



NATIONAL TECHNICAL UNIVERSITY OF ATHENS  
SCHOOL OF APPLIED MATHEMATICAL & PHYSICAL SCIENCES  
DEPARTMENT OF MATHEMATICS

# GRAPH DRAWING WITH EMPHASIS ON SECURITY APPLICATIONS

---

**EVMORFIA N. ARGYRIOU**

---

A THESIS PRESENTED TO THE NATIONAL TECHNICAL UNIVERSITY OF ATHENS IN FULFILMENT OF  
THE THESIS REQUIREMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

SUPERVISOR: PROF. ANTONIOS SYMVONIS

YEAR OF DOCTORAL DEFENCE: 2013



**European Union**  
European Social Fund



MINISTRY OF EDUCATION & RELIGIOUS AFFAIRS  
MANAGING AUTHORITY

Co- financed by Greece and the European Union



programme for development  
EUROPEAN SOCIAL FUND

This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.





NATIONAL TECHNICAL UNIVERSITY OF ATHENS  
SCHOOL OF APPLIED MATHEMATICS AND PHYSICAL SCIENCES  
DEPARTMENT OF MATHEMATICS

Ph.D. Degree Examination  
for  
**Evmorfia N. Argyriou**

---

**GRAPH DRAWING WITH EMPHASIS ON SECURITY APPLICATIONS**

---

**Thesis Committee:**

Antonios Symvonis, Supervisor  
Nektarios Koziris  
Michael Kaufmann

The Ph.D. Degree Proposal has been examined and approved on .....

.....  
A. Symvonis  
Professor  
NTUA  
(Supervisor)

.....  
N. Koziris  
Professor  
NTUA

.....  
M. Kaufmann  
Professor  
Un. of Tübingen

.....  
E. Zaxos  
Professor  
NTUA

.....  
V. Zisimopoulos  
Professor  
Un. of Athens

.....  
P. Rondogiannis  
Assoc. Professor  
Un. of Athens

.....  
D. Fotakis  
Lecturer  
NTUA

.....

Evmorfia N. Argyriou  
School of Applied Mathematics and Physical Sciences  
National Technical University of Athens

Copyright © Evmorfia N. Argyriou, 2013  
All rights reserved

No part of this thesis may be reproduced, stored in retrieval systems, or transmitted in any form or by any means -electronic, mechanical, photocopying, or otherwise- for profit or commercial advantage. It may be reprinted, stored or distributed for a non-profit, educational or research purpose, given that its source of origin and this notice are retained. Any questions concerning the use of this thesis for profit or commercial advantage should be addressed to the author. The opinions and conclusions stated in this thesis are expressing the author. They should not be considered as a pronouncement of the National Technical University of Athens.



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ & ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΟΜΕΑΣ ΜΑΘΗΜΑΤΙΚΩΝ

# ΑΠΕΙΚΟΝΙΣΗ ΓΡΑΦΗΜΑΤΩΝ ΜΕ ΕΜΦΑΣΗ ΣΕ ΕΦΑΡΜΟΓΕΣ ΑΣΦΑΛΕΙΑΣ

---

ΕΥΜΟΡΦΙΑ ΑΡΓΥΡΙΟΥ

---

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Επιβλέπων: Καθ. Αντώνιος Συμβώνης

Έτος Υποστήριξης Διδακτορικής Διατριβής: 2013



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Η παρούσα έρευνα έχει συγχρηματοδοτηθεί από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο - ΕΚΤ) και από εθνικούς πόρους μέσω του Επιχειρησιακού Προγράμματος "Εκπαίδευση και Δια Βίου Μάθηση" του Εθνικού Στρατηγικού Πλαισίου Αναφοράς (ΕΣΠΑ) - Ερευνητικό Χρηματοδοτούμενο Έργο: Ηράκλειτος ΙΙ. Επένδυση στην κοινωνία της γνώσης μέσω του Ευρωπαϊκού Κοινωνικού Ταμείου.





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ & ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΟΜΕΑΣ ΜΑΘΗΜΑΤΙΚΩΝ

Πρακτικό Εξέτασης της Διδακτορικής Διατριβής  
της  
**Ευμορφίας Ν. Αργυρίου**

---

**ΑΠΕΙΚΟΝΙΣΗ ΓΡΑΦΗΜΑΤΩΝ ΜΕ ΕΜΦΑΣΗ ΣΕ ΕΦΑΡΜΟΓΕΣ ΑΣΦΑΛΕΙΑΣ**

---

**Τριμελής Συμβουλευτική  
Επιτροπή:**

Αντώνιος Συμβώνης, Επιβλέπων  
Νεκτάριος Κοζύρης  
Michael Kaufmann

Εγκρίθηκε από την επταμελή εξεταστική επιτροπή την .....

.....  
Α. Συμβώνης  
Καθηγητής  
ΕΜΠ  
(Επιβλέπων)

.....  
Ν. Κοζύρης  
Καθηγητής  
ΕΜΠ

.....  
M. Kaufmann  
Καθηγητής  
Παν. Tübingen

.....  
Ε. Ζάχος  
Καθηγητής  
ΕΜΠ

.....  
Β. Ζησιμόπουλος  
Καθηγητής  
ΕΚΠΑ

.....  
Π. Ροντογιάννης  
Αναπ. Καθηγητής  
ΕΚΠΑ

.....  
Δ. Φωτάκης  
Λέκτορας  
ΕΜΠ

.....

Ευμορφία Ν. Αργυρίου

Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών

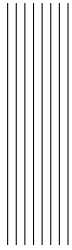
Εθνικό Μετσόβιο Πολυτεχνείο

Copyright © Ευμορφία Ν. Αργυρίου, 2013

Με επιφύλαξη παντός δικαιώματος - All rights reserved

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται η παρούσα σημείωση. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς το συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτή τη διατριβή εκφράζουν το συγγραφέα και δεν πρέπει να θεωρηθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.





## Abstract

Advances in technology and communications together with the widespread use of the internet have increased dramatically the volume of data that capture information of diverse types and thus, automated data analysis has become of tremendous importance. A challenge for data analysts is revealing possible relations among the data, as effectively and quick as possible. Information visualization can be promising in such a context, since it interprets the data into a visual image that provides a rich and clear presentation of the content. For this reason, visualizations have gained importance and find application in diverse areas (e.g., telecommunications, social networks and so on). The need for advanced visualization techniques led to the development of the *Graph Drawing* research area, which is concerned with the visualization of information as directed and undirected graphs. In this Ph.D. thesis, we have focused both on theoretical problems of graph drawing problems and on applications of graph drawing in security.

Over the last few decades, much research effort has been focused on the problem of reducing the number of edge crossings of a drawing, since it is commonly accepted that the edge crossings negatively affect the quality of a drawing. However, the computational complexity of the crossing minimization problem, which is  $\mathcal{NP}$ -complete in general, and recent cognitive experiments on large drawings of graphs showing that the negative impact of an edge crossing tends to be eliminated in the case where the crossing angles are large, resulted in a new research problem, referred to as the crossing resolution maximization problem (i.e., maximizing the minimum angle at which any two edges cross).

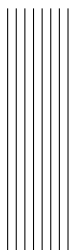
As a response to the problem of drawing graphs with optimal crossing resolution, a new class of graphs, referred to as right-angle crossing graphs, has been introduced by Didimo, Eades and Liotta [DEL09], in which every pair of crossing edges intersects at right angle. In this Ph.D. thesis, we study right-angle crossing drawings along the following research lines. Initially, we showed that the problem of determining whether a graph admits a straight-line right-angle crossing drawing is  $\mathcal{NP}$ -hard. We also propose the geometric right-angle crossing simultaneous drawing problem, which combines the study of right-angle crossing drawings and simultaneous graph drawing and we study classes of graphs that may admit such

drawings. Furthermore, we study the geometric graph-dual right-angle crossing simultaneous drawing problem, which dates back in 1963 to a question of W.T. Tutte [Tut63, p.p.767]. We also introduce a new graph drawing problem, in which the minimum of the crossing and the angular resolution (i.e., the minimum angle formed by two adjacent edges incident to a common node) should be maximized.

The second part of this Ph.D. thesis is devoted to applications of graph drawing in security, and more precisely, in fraud detection. Business fraud affects many companies worldwide causing them economic loss and liability issues towards their clients and other involved entities. Hence, the need for automated and immediate fraud detection is of particular interest. Even though information technology has been employed in fraud detection, the process of monitoring anti-fraud control systems is a problem of outstanding difficulty. There exist several factors that delay the immediate identification of fraud. Among them is the limited time and the huge amount of data that are in textual form, which renders their analysis in a daily basis (or even in a weekly basis) almost infeasible. For this reason, visualizations have gained significance, since they may facilitate the immediate identification of fraudulent activity.

Our motivation to work on fraud detection rests on real problems raised by internal auditors of a large telecommunication organization. We have mainly focused on the problem of detecting occupational fraud. The data-sets on which we have focused, stem from business systems involving pairs of entities (an employee and a client) such as billing systems, membership renewal systems and so on. Internal auditors suggested fraud patterns commonly detected in occupational fraud schemes based on which we have built our approaches. Among them is the identification of events that relate a pair of employee-client and appear in regular time basis. These periodic events may imply for instance that an employee deliberately falsifies the account of a client (e.g., deletes an invoice, etc).

Towards this direction, we present diverse visualization techniques such as visualizations of concentric circles, spiral visualizations and so on, which aim to identify periodic events related to pairs of entities. The systems that we have developed also support detection procedures based on fraud patterns suggested by the auditors. We also present an integrated fraud management visualization system combining pattern recognition and visualizations (heat-maps, circular drawings, parallel coordinates plots) in order to detect occupational fraud schemes. We have extended this particular system in order to monitor the behavior of the employees within the company network and identify possible suspicious activity such as unauthorized access to computers, systems or accounts.



## Περίληψη

Η πρόοδος της τεχνολογίας και των επικοινωνιών σε συνδυασμό με την ευρύτατα διαδεδωμένη χρήση του διαδικτύου έχουν αυξήσει δραματικά τον όγκο των δεδομένων που εμπιρεύουν πληροφορίες διαφόρων τύπων και καθιστούν την αυτοματοποιημένη ανάλυση δεδομένων ως εξαιρετικής σημασίας. Μία πρόκληση για τους αναλυτές δεδομένων αποτελεί η αποκάλυψη πιθανών συσχετίσεων μεταξύ των δεδομένων με γρήγορο και αποτελεσματικό τρόπο. Η οπτικοποίηση μπορεί να είναι πολλά υποσχόμενη σε αυτό πλαίσιο, καθώς μεταφράζει τα δεδομένα σε μία συνοπτική οπτική εικόνα η οποία παρέχει μία πλούσια και κατανοητή παρουσίαση του περιεχομένου. Ως αποτέλεσμα, η οπτικοποίηση έχει αποκτήσει σημαντικό ρόλο και βρίσκει εφαρμογές σε διάφορες περιοχές (π.χ. τηλεπικοινωνίες, κοινωνιολογικά δίκτυα και ούτω καθεξής). Η ανάγκη για προηγμένες τεχνικές οπτικοποίησης οδήγησε στην ανάπτυξη της περιοχής της *Απεικόνισης Γραφημάτων*, η οποία αφορά στην οπτικοποίηση πληροφοριών ως κατευθυνόμενα και μη-κατευθυνόμενα γραφήματα. Η παρούσα διδακτορική διατριβή, επικεντρώνει τόσο σε θεωρητικά προβλήματα της απεικόνισης γραφημάτων όσο και σε εφαρμογές της στην περιοχή της ασφάλειας.

Τα τελευταία χρόνια, σημαντική ερευνητική προσπάθεια έχει επικεντρωθεί στο πρόβλημα της ελαχιστοποίησης του αριθμού των τομών μεταξύ των ακμών μιας απεικόνισης ενός γραφήματος, καθώς είναι κοινά αποδεκτό ότι οι τομές επηρεάζουν αρνητικά την ποιότητα μιας απεικόνισης. Ωστόσο, η υπολογιστική πολυπλοκότητα του προβλήματος της ελαχιστοποίησης του αριθμού των τομών, το οποίο είναι  $\mathcal{NP}$ -πλήρες στη γενική περίπτωση, καθώς και πρόσφατα γνωστικά πειράματα σε μεγάλες απεικονίσεις γραφημάτων, τα οποία έδειξαν ότι η αρνητική επίδραση των τομών τείνει να εξαλειφθεί στην περίπτωση που οι γωνίες που σχηματίζονται στις τομές είναι μεγάλες, οδήγησαν σε ένα νέο ερευνητικό πρόβλημα, το οποίο ονομάζεται μεγιστοποίηση της γωνιακής ευκρίνειας τομών μιας απεικόνισης (δηλαδή τη μεγιστοποίηση της μικρότερης γωνίας υπό την οποία τέμνονται δύο ακμές).

Ως απάντηση στο πρόβλημα της απεικόνισης γραφημάτων με βέλτιστη γωνιακή ευκρίνεια τομών, μία νέα κλάση γραφημάτων, τα γραφήματα ορθογώνιων τομών ή εν συντομία RAC-γραφήματα, προτάθηκε από τους Didimo, Eades and Liotta [DEL09], στα οποία κάθε ζεύγος τεμνόμενων ακμών σχηματίζει ορθή γωνία. Στην παρούσα διδακτορική διατριβή, με-

λετήσαμε απεικονίσεις ορθογώνιων τομών στο ακόλουθο ερευνητικό πλαίσιο. Αρχικά, δείχνουμε ότι είναι  $\mathcal{NP}$ -δύσκολο να καθορίσουμε εάν ένα γράφημα επιδέχεται μία γεωμετρική RAC απεικόνιση. Προτείνουμε, επίσης, το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων, που συνδυάζει τη μελέτη των απεικονίσεων ορθογώνιων τομών και των ταυτόχρονων απεικονίσεων γραφημάτων και μελετάμε κλάσεις γραφημάτων που μπορεί να επιδέχονται τέτοιες απεικονίσεις. Επιπρόσθετα, μελετάμε το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης γραφημάτων, το οποίο χρονολογείται πίσω στο 1963 σε μία ερώτηση του W.T. Tutte [Tut63, p.p.767]. Εισάγουμε, τέλος, ένα νέο πρόβλημα απεικόνισης γραφημάτων, στο οποίο πρέπει να μεγιστοποιείται το ελάχιστο της γωνιακής ευκρίνειας (δηλαδή η μικρότερη γωνία που σχηματίζεται από δύο γειτονικές ακμές προσκείμενες στην ίδια κορυφή) και της γωνιακής ευκρίνειας τομών.

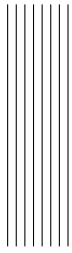
Το δεύτερο μέρος της παρούσας διδακτορικής διατριβής εστιάζει σε εφαρμογές της απεικόνισης γραφημάτων στην περιοχή της ασφάλειας και πιο συγκεκριμένα, στην ανίχνευση απάτης. Η εταιρική απάτη επηρεάζει πολλές εταιρείες, προκαλώντας τους οικονομικές απώλειες και νομικές ευθύνες έναντι πελατών και άλλων εμπλεκόμενων φορέων. Επομένως, η ανάγκη για αυτοματοποιημένη και άμεση ανίχνευση της απάτης αποκτά εξαιρετικό ενδιαφέρον. Παρόλο που προηγμένη τεχνολογία πληροφοριών έχει ήδη ενσωματωθεί στη διαδικασία ανίχνευσης απάτης, η παρακολούθηση των συστημάτων καταπολέμησης απάτης εξακολουθεί να αποτελεί ένα πρόβλημα αξιοσημείωτης δυσκολίας. Υπάρχουν διάφοροι παράγοντες που δυσχεραίνουν τον άμεσο εντοπισμό της απάτης. Μεταξύ αυτών είναι ο περιορισμένος χρόνος και ο τεράστιος όγκος δεδομένων που είναι συνήθως υπό τη μορφή κειμένου, τα οποία καθιστούν την ανάλυση τους σε καθημερινή βάση (αλλά και σε εβδομαδιαία βάση) σχεδόν ανέφικτη. Για το λόγο αυτό, οι οπτικοποιήσεις έχουν αποκτήσει σημαντικό ρόλο, καθώς διευκολύνουν τον άμεσο εντοπισμό “ύποπτης” δραστηριότητας.

Το κίνητρό μας να εργαστούμε σε προβλήματα ανίχνευσης απάτης βασίζεται σε πραγματικά προβλήματα, τα οποία μας τέθηκαν από τους εσωτερικούς ελεγκτές ενός μεγάλου οργανισμού τηλεπικοινωνιών. Έχουμε κυρίως εστιάσει στο πρόβλημα της ανίχνευσης εταιρικής απάτης. Τα σύνολα δεδομένων στα οποία επικεντρώσαμε, προέρχονται από εταιρικά συστήματα που εμπλέκουν ζεύγη οντοτήτων (υπάλληλος και πελάτης), όπως για παράδειγμα τα συστήματα τιμολόγησης/χρέωσης. Οι εσωτερικοί ελεγκτές πρότειναν πρότυπα απάτης, τα οποία συναντώνται συχνά σε σχήματα εταιρικής απάτης, πάνω στα οποία αναπτύξαμε τις προσεγγίσεις μας. Μεταξύ αυτών είναι η αναγνώριση γεγονότων τα οποία συσχετίζουν ένα ζεύγος υπαλλήλου-πελάτη και εμφανίζονται ανά τακτά χρονικά διαστήματα. Αυτά τα περιδικά γεγονότα, μπορεί να υπαινίσσονται, για παράδειγμα, ότι ένας υπάλληλος εσκεμμένα παραποιεί το λογαριασμό ενός πελάτη (διαγράφει/προσθέτει τιμολογήσεις, κλπ).

Προς αυτή την κατεύθυνση, παρουσιάζουμε διάφορες τεχνικές οπτικοποίησης, όπως

οπτικοποιήσεις ομόκεντρων κύκλων, σπειροειδείς οπτικοποιήσεις και ούτω καθεξής, που αποσκοπούν στην αναγνώριση περιοδικών γεγονότων που σχετίζονται με ζεύγη οντοτήτων. Παρουσιάζουμε, επίσης, ένα ολοκληρωμένο σύστημα ανίχνευσης απάτης, το οποίο συνδυάζει την αναγνώριση προτύπων με οπτικοποιήσεις (θερμικούς χάρτες, κυκλικές απεικονίσεις, διαγράμματα παράλληλων συντεταγμένων) αποσκοπώντας στην ανίχνευση σχημάτων εταιρικής απάτης. Έχουμε επεκτείνει το συγκεκριμένο σύστημα ώστε να εποπτεύει τη δραστηριότητα των υπαλλήλων στο δίκτυο της εταιρείας και να αναγνωρίζει πιθανή “ύποπτη” δραστηριότητα, όπως μη εξουσιοδοτημένη πρόσβαση σε υπολογιστές, εταιρικά συστήματα ή λογαριασμούς.





## Acknowledgements

This thesis completes a four-year journey in obtaining my Ph.D. from the National Technical University of Athens. Now, it is my pleasure to express my gratitude to all those who contributed to this study and supported me this period of my life.

My first debt of gratitude goes to my advisor, Prof. Antonios Symvonis, for his consistent and patient guidance, encouragement and insightful advice all these years that made possible the fulfillment of this Ph.D. thesis. I also wish to thank him for sharing his knowledge with me and for our cooperation since the time I was a postgraduate student. And, of course, for borrowing me his laptop for the last couple of months!

I also wish to express my thanks to Prof. Michael Kaufmann for allowing me to collaborate with him, sharing his knowledge with me and inviting me during my Ph.D. thesis to the University of Tuebingen. I would like also to thank my committee for reviewing this thesis.

I would like to acknowledge Vodafone Greece and, in particular, Kostas Panagos, Dimitris Peppes and Vassilis Vassiliou for introducing me the area of fraud detection and for providing me knowledge stemming from their professional experience that cannot be found in textbooks. I also thank them for their useful comments and trust during our cooperation.

This work would not have been possible without the support of my family and friends. I am really grateful to my husband Michalis Bekos for supporting me in many ways during this period of my life. I thank him not only for our scientific collaboration, but for his constant support, advice and encouragement (and for baby-sitting). I am not sure if I have managed to succeed without him. I also thank my daughter Maria-Artemis for her patience when her mom was at work.

I wish to express my gratitude to my parents, Nikos and Maria, and my brother, Panagiotis, for their love, support and encouragement to follow the path of learning. Also, I thank my uncle Vaggelis and his family, for their love and for always being here for me. I dedicate this thesis to Michalis, Maria-Artemis and my father Nikos, who left us very early and cannot share this joy with us.

I would like also to thank the members of our research group, Tamara Mchedlidze, Chrysanthi Raftopoulou and Kostas Mastakas not just for our collaboration, but for creating a nice at-

mosphere in the office with laughs, discussions and jokes. Also, I wish to express my thanks to all my friends who supported me all years of my academic life.

Finally, I acknowledge the financial support via the research funding program “Heracleitus II”. This thesis has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.





## Ευχαριστίες

Η παρούσα διατριβή ολοκληρώνει ένα ταξίδι τεσσάρων ετών για την απόκτηση του διδακτορικού μου από το Εθνικό Μετσόβιο Πολυτεχνείο. Τώρα, βρίσκομαι στην ευχάριστη θέση να ευχαριστήσω όλους όσους συνεισέφεραν σε αυτή τη διατριβή και με υποστήριξαν σε αυτή την περίοδο της ζωής μου.

Αρχικά, θα ήθελα να εκφράσω την ευγνωμοσύνη στον επιβλέποντα μου, Καθ. Αντώνιο Συμβώνη, για τη συνεχή και υπομονετική καθοδήγηση και ενθάρρυνση, καθώς και για τις συμβουλές του όλα αυτά τα χρόνια που κατέστησαν δυνατή την ολοκλήρωση αυτής της διδακτορικής διατριβής. Θα ήθελα επίσης να τον ευχαριστήσω που μοιράστηκε τη γνώση του μαζί μου, καθώς επίσης και για τη συνεργασία μας από τον καιρό ακόμα που ήμουν μεταπτυχιακή φοιτήτρια. Και φυσικά, τον ευχαριστώ που μου δάνεισε τον υπολογιστή του τους τελευταίους μήνες!

Θα ήθελα επίσης να εκφράσω τις ευχαριστίες μου στον Καθ. Michael Kaufmann που μου επέτρεψε να συνεργαστώ μαζί του, μοιράστηκε τη γνώση του μαζί μου και με προσκάλεσε στο Πανεπιστήμιο του Tuebingen κατά τη διάρκεια της διδακτορικής διατριβής. Ευχαριστώ επίσης τα μέλη της επιτροπής για τα σχόλια και τις συμβουλές τους πάνω σε αυτή τη διατριβή.

Θα ήθελα να ευχαριστήσω επίσης τη Vodafone Greece και ιδιαίτερα τους Κώστα Πανάγο, Δημήτρη Πεππέ και Βασίλη Βασιλείου που μου παρουσίασαν την περιοχή της ανίχνευσης απάτης και μου παρείχαν γνώση από την επαγγελματική τους εμπειρία που δεν είναι εύκολο να αναζητηθεί σε βιβλία. Τους ευχαριστώ θερμά για τα χρήσιμα σχόλια τους και την εμπιστοσύνη που μου έδειξαν κατά τη διάρκεια της συνεργασίας μας.

Η διατριβή αυτή δε θα μπορούσε να πραγματοποιηθεί χωρίς την υποστήριξη της οικογένειάς μου και των φίλων μου. Είμαι πραγματικά ευγνώμων στο σύζυγό μου, Μιχάλη Μπέκο για την υποστήριξή του με διάφορους τρόπους σε αυτή την περίοδο της ζωής μου. Τον ευχαριστώ, όχι μόνο για την επιστημονική συνεργασία μας, αλλά και για τη συνεχή του υποστήριξη και ενθάρρυνση, τις συμβουλές και τις συζητήσεις μας (αλλά και για το baby-sitting). Δεν είμαι σίγουρη ότι θα τα κατάφερα χωρίς αυτόν. Ευχαριστώ επίσης την κόρη μου Μαρία-Άρτεμις για την υπομονή της όταν η μαμά της έλειπε στη δουλειά.

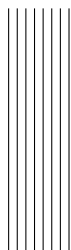
Θα ήθελα να εκφράσω την ευγνωμοσύνη μου στους γονείς μου, Νίκο και Μαρία, και στον αδελφό μου, Παναγιώτη, για την αγάπη τους, την υποστήριξη τους και την ενθάρρυνσή τους να ακολουθήσω το δρόμο της γνώσης. Θα ήθελα επίσης να ευχαριστήσω το θείο μου, Βαγγέλη και την οικογένειά του για την αγάπη τους και για το γεγονός ότι είναι πάντα εδώ για μένα. Αφιερώνω αυτή τη διατριβή στο Μιχάλη, στη Μαρία-Άρτεμις και στον πατέρα μου Νίκο, που μας άφησε νωρίς και δεν μπορεί να μοιραστεί αυτή τη χαρά μαζί μας.

Θα ήθελα επίσης να ευχαριστήσω τα μέλη της ερευνητικής μας ομάδας, Ταμμάρα Μτσε-ντλίντζε, Χρυσάνθη Ραυτοπούλου και Κώστα Μάστακα όχι μόνο για τη συνεργασία μας, αλλά και για την ωραία ατμόσφαιρα στο γραφείο μας με γέλια, συζητήσεις και αστεία. Επίσης, ευχαριστώ όλους τους φίλους που με στήριξαν όλα τα χρόνια της ακαδημαϊκής μου ζωής.

Τέλος, ευχαριστώ θερμά το πρόγραμμα “Ηράκλειτος II” για την οικονομική στήριξη που μου παρείχε. Η παρούσα έρευνα έχει συγχρηματοδοτηθεί από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο - ΕΚΤ) και από εθνικούς πόρους μέσω του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» του Εθνικού Στρατηγικού Πλαισίου Αναφοράς (ΕΣΠΑ) – Ερευνητικό Χρηματοδοτούμενο Έργο: Ηράκλειτος II. Επένδυση στην κοινωνία της γνώσης μέσω του Ευρωπαϊκού Κοινωνικού Ταμείου.

*To Michalis and Maria-Artemis  
and  
to the Memory of my father.*





# Περιεχόμενα

<b>I</b>	<b>Graph Drawing with Emphasis on Security Applications</b>	<b>27</b>
<b>1</b>	<b>Introduction</b>	<b>29</b>
1.1	Why Graph Drawing? . . . . .	29
1.2	A Brief Historic Overview . . . . .	30
1.3	Drawing Parameters . . . . .	31
1.3.1	Conventions . . . . .	32
1.3.2	Aesthetics . . . . .	33
1.4	Basic Graph Theory Definitions . . . . .	34
1.5	Common Graph Drawing Techniques . . . . .	37
1.5.1	Drawing Planar Graphs . . . . .	37
1.5.2	Force-directed Methods . . . . .	37
1.5.3	Orthogonal Drawings . . . . .	38
1.5.4	Layered Drawings . . . . .	39
1.5.5	Right-angle Crossing Drawings . . . . .	39
1.5.6	Simultaneous Drawings . . . . .	40
1.5.7	Clustered Drawings . . . . .	40
1.5.8	Dynamic Drawings . . . . .	41
1.5.9	Treemaps . . . . .	41
1.5.10	Other Common Drawings . . . . .	42
1.6	Graph Drawing with Applications in Security . . . . .	42
1.6.1	Applications in Network Monitoring . . . . .	45
1.6.2	Applications in Attack Graphs . . . . .	47
1.6.3	Applications in Fraud Detection . . . . .	48
1.6.4	Applications in Other Areas of Computer Security . . . . .	49
1.7	Thesis Contribution . . . . .	49
1.8	Thesis Overview . . . . .	51

<b>2</b>	<b>The Angular and Crossing Resolution Maximization Problems</b>	<b>53</b>
2.1	Motivation and Definitions . . . . .	53
2.2	Related Work . . . . .	55
2.2.1	Angular Resolution . . . . .	56
2.2.2	Crossing Resolution and RAC Graphs . . . . .	58
2.3	Preliminaries and Notation . . . . .	61
2.4	Simultaneous RAC drawings . . . . .	61
2.4.1	Motivation and Problem Definition . . . . .	61
2.4.2	Related Work . . . . .	63
<b>3</b>	<b>The Straight-line RAC drawing problem is NP-Hard</b>	<b>65</b>
3.1	Preliminary Notions and Notation . . . . .	65
3.2	A Class of Graphs with Unique RAC Combinatorial Embedding . . . . .	65
3.3	The Straight-Line RAC Drawing Problem is NP-hard . . . . .	87
3.3.1	Description of the Construction . . . . .	87
3.3.2	Properties of the Construction . . . . .	90
3.3.3	Reduction from 3-SAT . . . . .	90
<b>4</b>	<b>Geometric RAC Simultaneous Drawings of Graphs</b>	<b>95</b>
4.1	A Wheel and a Matching: A Negative Result . . . . .	95
4.2	A Cycle and a Matching: A Positive Result . . . . .	96
4.2.1	Algorithm Extensions . . . . .	105
4.3	A Planar Graph and its Dual: An Interesting Variation . . . . .	107
<b>5</b>	<b>Maximizing the Total Resolution of Graphs</b>	<b>113</b>
5.1	Preliminary Notions and Notation . . . . .	113
5.2	Drawings with Optimal Total Resolution for Complete and Complete Bipartite Graphs . . . . .	114
5.3	A Force-Directed Algorithm . . . . .	119
5.3.1	Time Complexity Analysis . . . . .	123
5.3.2	Similarities and Differences with Previous Techniques . . . . .	123
5.3.3	Experimental Results . . . . .	125
5.4	Sample Drawings . . . . .	129
<b>6</b>	<b>An Introduction to Occupational Fraud Detection</b>	<b>133</b>
6.1	What is Occupational Fraud? . . . . .	133
6.2	What is Fraud Detection? . . . . .	135
6.3	Motivation and Problem Definition . . . . .	138

6.4	Related Work . . . . .	139
6.5	Methodology and Notation . . . . .	141
<b>7</b>	<b>Periodicity Detection in Serial Data Through Visualization</b>	<b>143</b>
7.1	Periodicity Estimation . . . . .	143
7.2	Periodicity Visualization . . . . .	145
7.3	Single Period Visualization . . . . .	147
<b>8</b>	<b>A Spiral-Based Fraud Detection System</b>	<b>149</b>
8.1	Main Features of the System . . . . .	149
8.2	Overview of the Detection Procedure . . . . .	151
8.3	Ranking Procedure Description . . . . .	152
8.3.1	Client Ranking Function . . . . .	152
8.3.1.1	Distance from Billing Date . . . . .	153
8.3.1.2	Event-series Periodicity . . . . .	154
8.3.1.3	Events Occurring Outside Working Hours . . . . .	154
8.3.1.4	Number of Employees Related to a Client . . . . .	155
8.3.1.5	Action Name . . . . .	155
8.3.1.6	Client Status . . . . .	156
8.3.2	Employee Ranking . . . . .	156
8.4	Description of the System . . . . .	156
8.4.1	Off-line Mode . . . . .	157
8.4.1.1	The Spiral Visualization . . . . .	157
8.4.1.2	Supplementary Visualizations . . . . .	159
8.4.2	Semi-online Mode . . . . .	162
8.5	Case Study . . . . .	164
<b>9</b>	<b>A Visualization System Utilizing Radial Drawings and Heat-maps</b>	<b>167</b>
9.1	Main Features of the System . . . . .	167
9.2	Overview of the Detection Procedure . . . . .	168
9.3	Description of the System . . . . .	172
9.4	Case Study . . . . .	176
<b>10</b>	<b>Conclusions</b>	<b>187</b>
<b>II</b>	<b>Απεικόνιση Γραφημάτων με Έμφαση σε Εφαρμογές Ασφάλειας</b>	<b>189</b>
<b>11</b>	<b>Εισαγωγή</b>	<b>191</b>

11.1	Γιατί Απεικόνιση Γραφημάτων;	191
11.2	Μία σύντομη ιστορική αναδρομή	192
11.3	Παράμετροι απεικόνισης	194
11.3.1	Συμβάσεις	194
11.3.2	Αισθητικά κριτήρια	196
11.4	Βασικοί Ορισμοί της Θεωρίας Γραφημάτων	198
11.5	Συνήθεις Τεχνικές Απεικόνισης Γραφημάτων	201
11.5.1	Απεικόνιση επίπεδων γραφημάτων	201
11.5.2	Force-directed μέθοδοι	202
11.5.3	Ορθογώνιες απεικονίσεις	202
11.5.4	Πολυεπίπεδες απεικονίσεις	203
11.5.5	Απεικονίσεις ορθογώνιων τομών	204
11.5.6	Ταυτόχρονες απεικονίσεις	205
11.5.7	Ομαδοποιημένες απεικονίσεις	206
11.5.8	Δυναμικές απεικονίσεις	206
11.5.9	Δενδροειδείς απεικονίσεις	207
11.5.10	Άλλες συνήθεις απεικονίσεις	207
11.6	Απεικόνιση Γραφημάτων με Εφαρμογές στην Ασφάλεια	208
11.6.1	Εφαρμογές στην παρακολούθηση δικτύου	212
11.6.2	Εφαρμογές στα γραφήματα επίθεσης	214
11.6.3	Εφαρμογές στην ανίχνευση απάτης	215
11.6.4	Εφαρμογές σε άλλες περιοχές της ασφάλειας δικτύων	216
11.7	Συμβολή της Διδακτορικής Διατριβής	217
11.8	Οργάνωση της Διδακτορικής Διατριβής	219
<b>12</b>	<b>Τα Προβλήματα της Μεγιστοποίησης της Γωνιακής Ευκρίνειας και της Γωνιακής Ευκρίνειας Τομών</b>	<b>223</b>
12.1	Κίνητρο και Ορισμοί	223
12.2	Βιβλιογραφική Επισκόπηση	226
12.2.1	Γωνιακή Ευκρίνεια	226
12.2.2	Γωνιακή Ευκρίνεια τομών και RAC-γραφήματα	230
12.3	Εισαγωγικές Έννοιες και Ορολογία	233
12.4	Ταυτόχρονες RAC-απεικονίσεις	233
12.4.1	Κίνητρο και Ορισμοί	234
12.4.2	Συναφείς Εργασίες	236



<b>13 Το Πρόβλημα της Γεωμετρικής RAC Απεικόνισης είναι NP-δύσκολο</b>	<b>239</b>
13.1 Εισαγωγικές Έννοιες και Ορολογία . . . . .	239
13.2 Μία Κλάση Γραφημάτων με Μοναδική RAC Συνδυαστική Εμφύτευση . . . . .	239
13.3 Το Πρόβλημα της Γεωμετρικής RAC Απεικόνισης είναι NP-δύσκολο . . . . .	265
13.3.1 Περιγραφή της Κατασκευής . . . . .	266
13.3.2 Ιδιότητες της Κατασκευής . . . . .	270
13.3.3 Αναγωγή από το 3-SAT Πρόβλημα . . . . .	270
<b>14 Γεωμετρικές RAC Ταυτόχρονες Απεικονίσεις Γραφημάτων</b>	<b>273</b>
14.1 Γράφημα-Τροχός και Ταίριασμα: Ένα Αρνητικό Αποτέλεσμα . . . . .	273
14.2 Κύκλος και Ταίριασμα: Ένα Θετικό Αποτέλεσμα . . . . .	275
14.2.1 Αλγοριθμικές Επεκτάσεις . . . . .	286
14.3 Επίπεδο Γράφημα και Δυϊκό: Μία Ενδιαφέρουσα Παραλλαγή . . . . .	288
<b>15 Το Πρόβλημα της Μεγιστοποίησης της Συνολικής Γωνιακής Ευκρίνειας</b>	<b>295</b>
15.1 Προκαταρκτικές Έννοιες και Ορολογία . . . . .	295
15.2 Απεικονίσεις με Βέλτιστη Συνολική Γωνιακή Ευκρίνεια για Πλήρη και Πλήρη Διμερή Γραφήματα . . . . .	296
15.3 Ένας Force-Directed Αλγόριθμος . . . . .	302
15.3.1 Ανάλυση Χρονικής Πολυπλοκότητας . . . . .	306
15.3.2 Ομοιότητες και Διαφορές με Προηγούμενες Τεχνικές . . . . .	307
15.3.3 Πειραματικά Αποτελέσματα . . . . .	308
15.4 Παραδείγματα Απεικονίσεων . . . . .	313
<b>16 Μία Εισαγωγή στην Ανίχνευση Εταιρικής Απάτης</b>	<b>317</b>
16.1 Τι είναι η Εταιρική Απάτη? . . . . .	317
16.2 Τι είναι η Ανίχνευση Απάτης? . . . . .	320
16.3 Κίνητρο και Ορισμός του Προβλήματος . . . . .	323
16.4 Βιβλιογραφική Επισκόπηση . . . . .	324
16.5 Μεθοδολογία και Ορολογία . . . . .	327
<b>17 Ανίχνευση Περιοδικότητας σε Σειριακά Δεδομένα Μέσω Οπτικοποίησης</b>	<b>329</b>
17.1 Υπολογισμός Περιοδικότητας . . . . .	329
17.2 Οπτικοποίηση Περιοδικότητας . . . . .	331
17.3 Οπτικοποίηση Μοναδικής Περιόδου . . . . .	334

<b>18 Ένα Σύστημα Ανίχνευσης Απάτης Βασισμένο σε Σπειροειδείς Οπτικοποιήσεις</b>	<b>337</b>
18.1 Κύρια Χαρακτηριστικά του Συστήματος . . . . .	337
18.2 Επισκόπηση της Διαδικασίας Ανίχνευσης . . . . .	340
18.3 Περιγραφή της Διαδικασίας Κατάταξης . . . . .	340
18.3.1 Συνάρτηση Κατάταξης Πελατών . . . . .	340
18.3.1.1 Απόσταση από την Ημερομηνία Χρέωσης του Λογαριασμού .	341
18.3.1.2 Περιοδικότητα Σειράς Γεγονότων . . . . .	343
18.3.1.3 Γεγονότα Εκτός Ωρών Εργασίας . . . . .	344
18.3.1.4 Αριθμός των Υπαλλήλων που Σχετίζονται με έναν Πελάτη . .	344
18.3.1.5 Ενέργεια . . . . .	345
18.3.1.6 Κατάσταση Πελάτη . . . . .	345
18.3.2 Κατάταξη Υπαλλήλων . . . . .	346
18.4 Περιγραφή του Συστήματος . . . . .	346
18.4.1 Off-line Λειτουργία . . . . .	346
18.4.1.1 Η Σπειροειδής Οπτικοποίηση . . . . .	347
18.4.1.2 Συμπληρωματικές Οπτικοποιήσεις . . . . .	349
18.4.2 Semi-online Λειτουργία . . . . .	353
18.5 Μελέτη Περίπτωσης . . . . .	355
<b>19 Ένα Σύστημα Οπτικοποίησης με Κυκλικές Απεικονίσεις και Θερμικούς Χάρτες</b>	<b>359</b>
19.1 Κύρια Χαρακτηριστικά του Συστήματος . . . . .	359
19.2 Επισκόπηση της Διαδικασίας Ανίχνευσης . . . . .	361
19.3 Περιγραφή του Συστήματος . . . . .	365
19.4 Μελέτη Περίπτωσης . . . . .	370
<b>20 Συμπεράσματα</b>	<b>381</b>

# **Part I**

## **Graph Drawing with Emphasis on Security Applications**





# 1 Introduction

*“Use a picture. It’s worth a thousand words.”*

Arthur Brisbane, 1911

*“A good sketch is better than a long speech.”*

Napoleon Bonaparte

## 1.1 Why Graph Drawing?

Advances in technology, communications and computing together with the widespread use of the internet have increased dramatically the volume of data that capture information of diverse types (personal, business, scientific, security, etc) and have emerged information processing and data analysis to a crucial part of human life. Processing and revealing possible underlying relations in such data-sets is definitely a challenge for data analysts, since apart from the huge amount of data that they have to face, information is, in most cases, generated by different sources (usually in textual form and non-uniform formats). An immediate problem arising is to how to present the data such that a large amount of information is communicated as effectively as possible and in a well understandable form.

Visualization can be promising in such a context since it interprets the huge amount of data into a more comprehensive visual image that provides a rich and clear presentation of the content. In fact, people prefer to visualize information rather than describe in words, since human brain performs extremely better and quicker in image processing rather than text processing. Visual representations of large amounts of data enable the interpretation of information and may reveal inherent relations within the data. As a result, visualization has gained significance and finds application in diverse areas such as networks management (computer, telecommunication, social networks), business modeling, bioinformatics, database design, cartography, and so on. The need for advanced visualization techniques resulted in the development of the *Graph Drawing* area which is today an active research area in the computer sci-

ence community concerned with the visualization of combinatorial information as directed and undirected graphs.

A *graph* is an abstract mathematical object that models relational structures, consisting of entities and pairwise relationships between these entities. The entities correspond to the *vertices* of the graph, while the relations correspond to its *edges*. *Drawings of graphs* are quite common in our everyday life, even if people do not know it. Living in a city with a metro network, passengers search for directions in metro maps in which a graph illustrates information about metro lines. Each vertex of such graph corresponds to a metro station and is usually drawn as a circle. The lines connecting the stations correspond to the edges of the graph and are drawn as straight-line segments. Organizational charts are another example of graph drawing describing the hierarchical structure of an organization and the relationships between people within the organization. These examples of graph drawings are only a small excerpt from the vast range of possible applications. In fact, graph drawing offers its services to a great number of fields, such as software engineering (data flow diagrams, object-oriented class hierarchies), databases (entity-relationship diagrams), information systems (organization charts), VLSI design, network security, artificial intelligence (knowledge-representation diagrams), and it is not only restricted to computer science.

## 1.2 A Brief Historic Overview

It is difficult to trace the exact origins of graph drawing. Geometric drawings (i.e., drawings of graphs in which each edge is drawn as a line-segment) of various kinds have appeared extensively through the ages. The game-board of Morris games was probably the first form of graph drawing, which according to Henry Parker ([Hen09], p.p. 578), dates back in 1400 B.C. in Egypt. However, no images are available. This game-board is also presented in the "Book of Games" from the 13th-century [oCaL51]. Family trees are another type of ancient graph drawing (again without images) as described by Pliny the Elder and Seneca ([KZ91], p.p. 111). The first images of genealogical trees appeared in the Middle Ages [KZ91]. Squares of opposition have been also used in the Middle Ages representing logic diagrams [Mur84].

Although Euler relied on a drawing in order to solve the "Königsberger Brückenproblem" in his 1736 famous paper [Eul36], he actually did not use graph drawings to present his results. For this reason, even though one might think that the invention of graph theory and graph drawing coincide, in reality graph drawings started years after, in the late 18th century and early 19th century. Among them, Listing in 1847 worked on the problem of path tracing and presented a drawing that can be drawn with a single stroke [Lis47]. Cayley presented the first drawings of trees [Cay57]. For a more detailed presentation on the history of

graph drawing, the reader is referred to [KMBW02].

Even though the first paper with graph drawing algorithms can be traced back in W. T. Tutte's 1963 work on "How to Draw a Graph" [Tut63], it is only during the last two decades that graph drawing researchers have been organized into an active community (the first Graph Drawing Annual Symposium took place in Paris, in 1993). In these years, researchers developed a large number of graph drawing styles that are appropriate for several applications (organizational charts, database schemata, network layouts, hierarchies, planar drawings, trees, tree-maps, just to name a few), developed techniques to compare drawings and to evaluate their quality (number of crossings, number of edge-bends, resolution, aspect ratio, etc) and developed and analyzed algorithms that produce drawings of aesthetic appeal for a wide variety of drawing styles. For a detailed analysis on graph drawing algorithms, the reader is referred to classical textbooks on graph drawing [DETT99, KW01, NR04].

### 1.3 Drawing Parameters

From a human point of view, the usefulness of a graph drawing highly depends on its readability, i.e., its capability to convey the information contained in the graph quickly and clearly. A poor drawing may be confusing and misleading. However, how can we judge the quality of a graph drawing? If we try to draw a community of a social network (e.g., a facebook community) using an algorithm that was designed for drawing metro maps, we will be entirely disappointed by the result. The reason is that the metro map drawing algorithm takes into consideration some general knowledge that it is important for metro maps. On the other hand, a drawing of a community of a social network should reveal the relationships among community members and this requirement cannot be captured by a metro map drawing algorithm.

A reasonable question that arises from the above discussion is whether we need for each application domain, a different graph drawing algorithm, which takes into consideration more than the properties of the structure of a graph and may also use additional information about the semantics behind the graph structure. An affirmative answer to this question would imply the need of a tremendous number of different graph drawing algorithms; one for each application domain. Fortunately, this is not the case. There are only a few general concepts and techniques, which are capable of covering a wide variety of application domains. For this reason, general drawing conventions and aesthetics criteria have been introduced by the graph drawing community.

### 1.3.1 Conventions

A graph drawing has to satisfy a basic rule, referred to as *drawing convention*, in order to be admissible. For instance, in metro maps, a commonly used convention is the following: the vertices of the graph that correspond to stations are drawn as circles and all edges are drawn either as axis-aligned line-segments or as a sequence of axis-aligned line-segments. A drawing convention depends on the nature of the data that are visualized and may be very simple or very complex, involving less or more details of a drawing. We can also adopt different styles for the representation of a graph. For instance, vertices can be represented by boxes, dots, circles, labels or a mixture of styles. Information about vertices and edges can be further visualized by textual labels, different colors, different thickness, and so on. In the following, we present several common drawing conventions:

**Planar Drawings:** A drawing of a graph in which no two edges intersect is called *planar*. A graph that admits such a drawing is called a *planar graph*. Unfortunately, a small number of graphs admit a planar drawing. For instance, it is known that a planar graph with  $n$  vertices cannot have more than  $3n - 6$  edges. Note that, the maximum number of edges of a graph is  $n(n - 1)/2$  (*clique graphs*).

**Polyline Drawings:** A drawing in which each edge of the graph is drawn as a polygonal chain is called *polyline drawing*. The point at which an edge changes its direction is referred to as a *bend*. Polyline drawings are common in metro map production.

**Straight-line Drawings:** A drawing in which each edge is drawn as a straight-line segment is called *straight-line drawing*. Straight-line drawings are a special case of polyline drawings, where edges are drawn with no bends. Such drawings are also referred to as *geometric drawings*. It is known that any planar graph admits a (planar) straight-line drawing.

**Orthogonal Drawings:** A drawing in which each edge is represented as alternating sequences of axis-aligned line-segments is referred to as *orthogonal drawing*. Graphs that admit orthogonal drawings are usually of maximum degree four, where the degree of a graph is determined by the maximum number of edges incident to a vertex.

**Grid Drawings:** A drawing in which each vertex, crossing and/or bend occupies a point on the plane with integer coordinates is called *grid drawing*. Grid drawings are essential in many application scenarios e.g., in the case where a graph has to be displayed on a screen with low resolution.

**Upward (resp. Downward) Drawings:** A drawing in which every edge is a simple open curve monotonically increasing (resp. decreasing) in the vertical direction is referred



to as *upward (resp. downward) drawing*. For example, genealogical trees are usually drawn downwards to reveal the hierarchy.

### 1.3.2 Aesthetics

The quality of a drawing is measured based on *aesthetics criteria* that evaluate the niceness of drawings. Graph drawing algorithms typically conform to one or more aesthetic criteria with the assumption that the niceness of the drawing is improved. For instance, in a VLSI circuit bends should be avoided since they increase the manufacturing cost, and thus, we are interested in planar drawings with the minimum possible number of bends. A list of common aesthetic criteria [BFN85, PCA02], includes the following:

**Crossing minimization:** Crossings between edges affect negatively the quality of a graph drawing since it is widely accepted that they confuse the viewer. Ideally, one would prefer planar drawings. But as already stated, this is not always feasible. Hence, drawings with minimum number of crossings are desirable.

**Bend minimization:** The total number of bends and the number of bends per edge in a drawing should be kept as small as possible. Bends cause confusion to the human eye; they interrupt the eye movement and require sharp changes of direction. This is a crucial criterion in orthogonal drawings, where a large body of research has been devoted to this problem.

**Area minimization:** Minimizing the area of a drawing is crucial in many applications where the available space is limited, e.g., mobile devices are equipped with a relatively small screen and several readability issues may occur due to space limitations. On the other hand, a large drawing has to be scaled down or its resolution has to be decreased in order to fit in the available space and thus, its readability deteriorates.

**Angle maximization:** The *angular resolution* of a drawing of a graph is defined as the smallest angle formed by two adjacent edges incident to a common vertex. The *crossing resolution* of a drawing of a graph refers to the smallest angle formed by a pair of crossing edges. The maximization of the smallest angle in a drawing is essential in several applications, e.g., in mobile devices which are usually equipped with screens of low resolution.

**Length minimization:** In many applications, it is important to keep the length of the edges small. For instance, in VLSI circuits, the total edge length of the drawing corresponds to the length of the wires that have to be used, and thus, has to be kept small.

**Aspect Ratio minimization:** The *aspect ratio* of a drawing of a graph is defined as the ratio of the length of the longest side to the length of the shortest side of the smallest rectangle that encloses the drawing. The minimization of this criterion would provide the flexibility to fit the drawing in screen windows of arbitrary size.

**Symmetries:** Conveying symmetric properties of a graph is essential, but not an easy task. Usually, appropriate mathematical models must be introduced to cope with symmetries in graph drawings, see e.g., [Ead88, MAC<sup>+</sup>95].

**Clustering:** In many cases and especially when drawing large and dense graphs, it is important to place close to each other vertices of the drawing with similar properties in order to reveal the structure of the graph.

Unfortunately, there is no clear way to rank these criteria. Their importance heavily depends on the application domain. However, Purchase [Pur97] through cognitive experiments, observed that the crossing minimization seems to be the most important criterion for the readability of a drawing, while bend-minimization and symmetry-maximization seem to affect less its quality. She also prioritized the aesthetic criteria mentioned above (refer also to [PCJ97, Pur02]). On the other hand, other research papers state that the differences in the perception of a graph depends on its final layout [BMK96, MBK97, DC98].

Though these criteria are generally accepted, it is quite impossible to meet all aesthetic rules at the same time. Some of them conflict with each other. Some of them are computationally demanding. For instance, minimizing the number of crossings is  $\mathcal{NP}$ -complete [GJ83]. The same holds also, for the problem of determining whether a graph can be embedded on a grid of predefined size [MRK84]. Several results regarding the complexity of the optimization of the above criteria are presented in [GJ79]. Thus, practical graph drawings are usually the result of compromising among the aesthetics and general approaches are usually heuristics or approximation algorithms.

## 1.4 Basic Graph Theory Definitions

Typically, a *graph*  $G$  is an ordered pair  $(V, E)$ , where  $V$  is the set of *vertices* and  $E$  is the set of *edges* of  $G$ . Note that the vertex (edge) set of  $G$  is also commonly denoted by  $V(G)$  ( $E(G)$ ). A *subgraph* of graph  $G$  is a graph  $G' = (V', E')$  such that  $V' \subseteq V$  and  $E' \subseteq E$ . Given two graphs  $G$  and  $G'$ , we denote by  $G \cup G'$  the graph induced by the union of  $G$  and  $G'$ . An edge  $e \in E$  connecting a pair of vertices  $u, v \in V$  is usually denoted by  $(u, v)$ , where  $u$  and  $v$  are called the *endpoints* of edge  $e$ . Edge  $e$  is said to be *incident* to each of its endpoints.

By definition, mathematical sets are unordered. Hence, the edges of a graph have no direction. Such a graph is called *undirected*. However, in several applications, an edge has a direction that corresponds to a special property between a pair of vertices that is not bidirectional, e.g., one-way route in a road network. A graph in which the edges are ordered pairs of vertices is called *directed graph* or *digraph*. In a digraph, for an edge  $(u, v)$ , vertex  $u$  is called *source*, whereas vertex  $v$  is called *target*. Also, in graphs with directed edges, an edge of the form  $(u, u)$  is called *self-loop* or simply *loop*. Accordingly, a graph which has no self-loops is called *loop-less*. Edges that have the same endpoints are called *parallel edges*. A graph that has no self-loops and parallel edges is called *simple*.

In an undirected graph, the *degree* of a vertex  $v$ , usually denoted by  $d(v)$ , is the number of edges incident to vertex  $v$ . In a digraph, the *in-degree* of a vertex  $v$ , denoted by  $d^-(v)$ , is the number of edges having vertex  $v$  as their target vertex. Accordingly, the *out-degree* of a vertex  $v$ , denoted  $d^+(v)$ , is the number of edges having vertex  $v$  as their *source* vertex. A vertex of degree zero is called *isolated*. The *degree* of a graph, usually denoted by  $d(G)$ , is the maximum vertex degree of  $G$ . A *regular graph* is a graph where each vertex has the same degree. A regular directed graph must also satisfy the stronger condition that the in-degree and out-degree of each vertex are equal to each other. A regular graph with vertices of degree  $k$  is called a *k-regular graph* or *regular graph of degree k*. The *density* of a undirected graph  $G$  is the ratio of edges in  $G$  to the maximum possible number of edges.

A *path of length k* from vertex  $u$  to vertex  $v$  of graph  $G$  is a sequence  $\langle v_0, v_1, v_2, \dots, v_k \rangle$  of vertices such that  $u = v_0, v = v_k$  and  $(v_{i-1}, v_i) \in E$ , for each  $i = 1, 2, \dots, k$ . The *length* of the path corresponds to the number of its edges. A path is called *simple* if no vertex appears more than once. A path  $\langle v_0, v_1, v_2, \dots, v_k \rangle$  forms a *cycle* if  $v_0 = v_k$  and consists of at least one edge. A cycle is called *simple* if no vertex appears more than once. A graph that contains no cycles is called *acyclic*. A *hamiltonian path* of a graph  $G$  is a path that visits every vertex of  $G$ . A *hamiltonian cycle* of a graph  $G$  is a cycle that visits every vertex of  $G$ . A *tree* is an acyclic graph, in which any pair of vertices is connected through a path. If one vertex of a tree is distinguished from the other vertices then, the tree is called *rooted tree*. A *caterpillar* is a tree in which all the vertices are within distance one of a central path. A *wheel graph*  $W_n$  is a graph with  $n$  vertices (where  $n \geq 4$ ), formed by connecting a single vertex to all vertices of an  $(n - 1)$ -cycle. A *matching* of a graph  $G$  is a set of edges  $E' \subseteq E$  such that all edges in  $E'$  do not share a common endpoint.

An undirected graph is referred to as *complete graph* or *clique*, if every pair of vertices is connected by an edge. The complete graph on  $n$  vertices is denoted by  $K_n$ . A *bipartite graph* is a graph whose vertices can be divided into two disjoint sets  $V_1$  and  $V_2$  such that every edge has one endpoint in  $V_1$  and one endpoint in  $V_2$ . A *complete bipartite* graph is a bipartite graph

in which each vertex of  $V_1$  is joined to each vertex of  $V_2$  by exactly one edge. If  $r$  (resp.  $s$ ) is the number of vertices of  $V_1$  (resp.  $V_2$ ), then the complete bipartite is denoted by  $K_{r,s}$ . An  $n$ -star graph  $S_n$  is defined as the complete bipartite graph  $K_{1,n-1}$ . A *directed acyclic graph* (or DAG for short), is a directed graph with no directed cycles. Two graphs are called *isomorphic* if there exists a one-to-one correspondence between their vertex sets and adjacency is preserved.

An undirected graph is *connected* if any pair of vertices is connected by a path. The *connected components* of an undirected graph are equivalence classes under the relation is “reachable” on the set of vertices. A directed graph is called *strongly connected* if there is a path from each vertex in the graph to every other vertex. The strongly-connected components are equivalence classes under the relation is “both reachable” on the set of vertices. A *bridge* is an edge whose removal disconnects a graph. Accordingly, a *cut vertex* or *articulation point*, is a vertex whose removal disconnects the graph. A graph with no articulation points is called *biconnected*. A connected graph in which the deletion of any two vertices (and incident edges) results in a graph that is still connected is referred to as *triconnected* graph.

A *drawing* of a simple graph  $G$  (usually denoted by  $\Gamma(G)$ ) is a representation of  $G$  on the plane, where each vertex is a point and each edge is a closed Jordan arc between the points representing its endpoints. By  $x(v)$  and  $y(v)$ , we denote the  $x$ - and  $y$ -coordinate of  $v \in V$  in  $\Gamma(G)$ . Each drawing uniquely defines cyclic orders of edges incident to the same vertex and, therefore, specifies a *combinatorial embedding*.

A planar graph with a fixed planar embedding is called *plane graph*. In a planar drawing, any cycle that surrounds a region without any edges reaching from the cycle into the region forms a *face*. Let  $G$  be a plane graph. The *dual graph* of  $G$  is usually denoted by  $G^*$  has a vertex for each face of  $G$  and an edge between two faces  $f_1$  and  $f_2$  if and only if  $f_1$  and  $f_2$  share an edge. The subgraph of  $G^*$  induced by the deletion of the face-vertex of  $G^*$  corresponding to the unbounded face of  $G$  is called the *weak dual* of  $G$ . A planar graph is called *maximal planar* if no more edges can be added so that the resulting graph is still planar. A *triangular graph* is a maximal straight-line plane graph that has only three vertices on its convex hull.

A graph that can be drawn in the plane without crossings in such a way that all of the vertices belong to the unbounded face of the drawing is called *outerplanar*. An *outerpath* is an outerplanar graph whose weak dual is a path. The *geometric thickness* of a graph is the minimum number of layers required to draw the graph on the plane with straight-line edges and assign each edge to a layer such that no two edges of the same layer cross.

## 1.5 Common Graph Drawing Techniques

In this section, we will briefly review some common drawing techniques from the graph drawing literature. For a more comprehensive classification of these techniques, the reader is also referred to [MGB<sup>+</sup>98, HSMM00]. Detailed description of algorithms can be found in classical graph drawing textbooks [DETT99, KW01].

### 1.5.1 Drawing Planar Graphs

Planar drawings are of importance for several practical applications in which crossings among pair of edges are a nuisance such as VLSI design or road networks. However, since it is known that any planar graph with  $n$  vertices contains at most  $3n - 6$  edges, not all graphs are planar. Notably, much research effort has been focused on the planarity testing problem, i.e., the problem of testing whether a graph admits a planar drawing. The first polynomial-time algorithms to the planarity testing problem date back to sixties [AP61, Gol63, Bad64, DMP64, LEC67]. Few years later, Hopcroft and Tarjan [HT74] were the first to present a linear-time algorithm. Since then, several other linear time algorithms have been presented based on different approaches [BL76, CNA085, kSIH92, BT96, WKWL99, IH03a, dFdM96, dFdM12, Bra09]. For a survey on planar graph drawing refer to the classical book of Nishizeki and Rahman [NR04].

Since not all graphs admit planar drawings and the crossing minimization problem is, in general,  $\mathcal{NP}$ -complete [GJ83], a reasonable question that arises in this context is whether a graph can be made planar by removing the minimum number of edges (*graph planarization*). However, since the corresponding decision problem is  $\mathcal{NP}$ -complete [LY80], heuristics have been proposed for the planarization of graphs [BTT84, GMW05]. For a detailed overview on such algorithms the reader is referred to [Wei01].

### 1.5.2 Force-directed Methods

*Force-directed methods* are commonly employed for drawing graphs [Ead84, FR91, DH96, KK89]. In general, a force-directed method models the graph as a physical system with appropriate forces acting on it. Then, a good drawing is obtained from an equilibrium state of the system. Force-directed methods aim to produce “aesthetically pleasing” drawings. This is usually captured by having uniform edge lengths and conveying as much symmetry as possible. For the special case of planar graphs, these algorithms tend to result in crossing-free drawings. For an overview on force-directed methods and their variations the reader is referred to [DETT99, Bra01, KW01, Kob12].

One of the first force-directed graph drawing algorithms was presented by Tutte [Tut63] based on barycentric representations. The classical force-directed algorithm of Eades [Ead84] models the vertices of the graph as electrically charged particles that repel each other and its edges by springs attracting adjacent vertices. The algorithm of Fruchterman and Reinhold [FR91] also relies on spring forces (but also uses the notion of “temperature”). Several variations of the force-directed algorithm of Eades’ have been proposed over the years [KK89, FLM95, DH96, BHR96, Ber00]. In particular, force-directed algorithms enjoy the acceptance of the graph drawing community. It’s worth mentioning that, in each annual symposium on graph drawing, there exists at least one research work that models a specific graph drawing problem as a system of forces and suggests a corresponding algorithm to solve it.

### 1.5.3 Orthogonal Drawings

The study of orthogonal graphs can be traced back to VLSI design and floor-planning applications [Lei80, Val81, TT89]. An *orthogonal drawing* of a planar graph of maximum degree four (or 4-planar for short) is a drawing in which each vertex occupies a point on the integer grid and each edge is drawn as an alternating sequence of horizontal and vertical line-segments. In an orthogonal drawing, vertices of degree four are of optimal angular resolution, while vertices of lower degrees are within a factor of two from optimal. Orthogonal graph drawings are usually optimal with respect to an objective function. Typical objective functions include the minimization of the area [TT89, PT97], the minimization of the total number of bends [Tam87, FK95, GT01] or the minimization of the maximum number of bends per edge [BK94, LMS98]; for an overview see e.g. [EFK01].

In the field of orthogonal graph drawing and in particular, for the problem of minimizing the total number of bends, Tamassia [Tam87] laid important methodological foundations by the *topology-shape-metrics* approach (*TSM*), which works in three phases. In the first *planarization* phase a “planar” embedding is computed for a given (non) planar graph of maximum degree four by replacing edge crossings by dummy vertices (*planar representation*). In the next phase (*orthogonalization phase*), the angles and the bends of the drawing are computed, producing an *orthogonal representation*. In the third *compaction* phase the coordinates for the vertices and edges are computed. The core of the second step is a min-cost flow algorithm to minimize the number of bends of an embedded 4-planar graph [CK12]. Note that the general problem of minimizing the total number of bends of a 4-planar graph (over all planar embeddings) is  $\mathcal{NP}$ -hard [GT01].

### 1.5.4 Layered Drawings

Layered drawings are commonly used to represent hierarchical data. In a *layered* or *hierarchical drawing*, the vertices of a graph are drawn in a series of horizontal grid lines (*layers*). Dominant vertices are placed at the topmost layers; the remaining vertices lower in the hierarchy progressively further down. The most known framework for obtaining layered drawings of general graphs (not necessarily acyclic) is Sugiyama's one [STT81], according to which: (i) edges should point downwards, (ii) vertices should be evenly distributed in layers, (iii) long edges should be avoided, (iv) the number edge crossings is as small as possible, and, (v) many straight (i.e., bend-less) edges should occur.

Sugiyama's algorithm performs in four steps. In the first step (*cycle removal*), the digraph is made acyclic by reversing the direction of some edges. However, since the corresponding problem is  $\mathcal{NP}$ -hard [Kar72, GJ83], heuristics and approximation algorithms have been proposed and used [GJR85, Rei85, BS90, ELS93]. In the second step (*layer assignment*), vertices are assigned to appropriate layers. In addition, dummy vertices are placed at the intersections of edges (incident to vertices of non-consecutive layers) with layers, which will be eventually replaced by bends that control their inflection points. Common optimization functions for this step are the height minimization [JG72], the width minimization [BLME02, TNB04] or the minimization of the number of dummy vertices [GKNV93]. *Crossing reduction* is the next step of the algorithm in which an ordering of the vertices of each layer is computed such the total number of crossings is as small as possible. Heuristic algorithms [STT81, GKNV93] are also common for this step, since the corresponding decision problem is  $\mathcal{NP}$ -hard [MNKF90]. In the final step, the  $X$ -coordinates of the vertices are computed such that there exist no overlaps among vertices of the same layer and the total number of bends is minimized [STT81, GKNV93, San96, BK01].

### 1.5.5 Right-angle Crossing Drawings

Unfortunately, very few graphs are planar in practice and edge crossings are unavoidable in the vast majority of the application scenarios where relational data are visualized and analyzed by means of graph drawing techniques. This, together with the fact that the edge crossing minimization problem is computationally difficult [GJ83] apparently leads to the conclusion that the computation of high-quality drawings of non-planar graphs is significantly difficult and new approaches have to be introduced in order to handle the crossings that cannot be eliminated in a graph drawing.

Recent eye-tracking experiments by Huang et al. [Hua07, Hua08, HHE08] indicate that the negative impact of an edge crossing is eliminated in the case where the crossing angle is large. This hopefully implies that good drawings of graphs may have crossings, as long as the

crossings edges form large angles. The *crossing resolution problem* (i.e., the problem of determining a drawing in which the minimum angle at which any two edges cross is maximized) becomes as important for non-planar graphs, as the problem of avoiding crossings for planar graphs. The results of Huang et al. also motivated the study of a new class of drawings, called *right-angle crossing drawings* (or *RAC-drawings* for short) in which every pair of crossing edges intersects at right angle as a response to the problem of drawing graphs with optimal crossing resolution. Didimo, Eades and Liotta [DEL11] initiated the study of RAC-drawings and since then, much research effort has been devoted to the study of RAC-drawing problems [ACD<sup>+</sup>09, DDLM09, DEL10]. A large part of this Ph.D. thesis is devoted to the crossing resolution problem and in particular, to RAC-drawings which will be described in full detail in the Chapters 2-5.

### 1.5.6 Simultaneous Drawings

Unlike common graph drawing problems that involve a single graph, simultaneous graph drawing deals with the problem of drawing a series of planar graphs (e.g., a path and a tree, three paths, two planar graphs, etc) on the same set of vertices on the plane, such that each graph is drawn planar (i.e., only edges of different graphs are allowed to cross). Besides its independent theoretical interest, this problem arises in several application areas, such as software engineering, databases and social networks, where a visual analysis of evolving graphs, defined on the same set of vertices, is useful.

Known results for this problem are usually of three types: (i) positive results, i.e., algorithms that produce simultaneous drawings for given graphs [BCD<sup>+</sup>07, CvKL<sup>+</sup>11], (ii) negative results, i.e, counterexamples that prove that it is not feasible to produce a simultaneous drawing [EK05b, FKK09, GKV09, AGKN10] for given graphs, and (iii) results for the problem of deciding whether given graphs admit a simultaneous drawing [EBG]<sup>+</sup>07]. For an overview refer to [BKR07, BKR12]. Simultaneous drawings will be revisited in the Chapters 2,4, since they have been studied as part of this Ph.D. thesis in conjunction with RAC drawings. A more detailed overview of known results from the literature is postponed to a Chapter 2, as well.

### 1.5.7 Clustered Drawings

Large and complex graphs such as world wide web graphs, social networks or complex VLSI circuits are difficult to be drawn using classical graph drawing techniques. In such cases, in order to improve the readability of the drawing, it is advantageous to reduce the number of elements that are being visualized simultaneously. *Clustering* has been proved an effective approach towards this direction. In such a context, the vertices of the graph are divided into



a particular number of groups (*clusters*) such that vertices with similar “attributes” belong to the same group.

Existing clustering algorithms usually optimize fitness functions that measure the quality of the clustering such as *coverage* (i.e., the fraction of intra-cluster edges within the complete set of edges), *external* or *inter-cluster density* (i.e., the ratio of inter-cluster edges to the maximum number of inter-cluster edges possible), and so on. As expected, most of the clustering measures lead to  $\mathcal{NP}$ -hard problems and thus, heuristics and approximation algorithms have been proposed [SM91, SM93, Nor93, KMG88, BGW03, Sch05]. It’s worth mentioning that Eades et al. [EFLN99] and Feng et al. [FCE95] have proposed polynomial-time algorithms for straight-line drawings of clustered graphs which, however, require exponential drawing area. For an overview of clustering techniques refer to [JD88, JMF99, Sch07].

### 1.5.8 Dynamic Drawings

*Dynamic graphs* are graphs that evolve over the time (i.e., the properties of the graph change over the time, vertices and edges are being added or removed, vertices become more or less important, etc). Typical examples are graphs of computers on the internet, the world wide web graph, phone-call graphs, graphs of friends in a social network, etc. Drawings of dynamic graphs can be considered as a sequence of static drawings. However, in several application scenarios, in which a graph is repeatedly redesigned, it is time-consuming and computationally demanding to compute a new drawing from scratch each time that the graph changes. More problems usually occur if it is additionally required to preserve the *mental-map* of the user [ELMS91, MELS95].

Several techniques have been proposed for capturing the property of the preservation of the mental map and which can be classified in two main categories. In the first category, changes are restricted to specific parts of the drawing and usually deal with the addition of vertices and edges, maintaining the current placement of the vertices of the graph [BP90, MHT93, Nor96, Föß97, PT98]. The second category is defined based on various distance metrics [ELMS91, MELS95, BT98, LMR98]. For preserving the mental map of the user, the corresponding distance should be kept as small as possible when a change occurs.

### 1.5.9 Treemaps

A *treemap* is a common method to visualize tree-structured hierarchical data, in which each vertex has one parent (except for the root that has no parent) and zero or more children. In a treemap, the vertices are represented as nested rectangles such that the rectangle of a particular vertex nests inside its children. The area that a rectangle occupies is proportional to a specific attribute of the vertex that corresponds to it. Shneiderman [Shn92] was

the first who used this technique for the visualization of the files of a hard disk. He proposed a recursive algorithm that divided the drawing area into rectangles that alternated either horizontally or vertically. The size of each rectangle was proportional to the size of the file/folder that represented, while different colors or shadings were used to distinguish different regions.

Several variants of this method have been proposed [TJ92, VWvdW99, BHvW99, Sta00, BSW02]. Of particular interest, is the one Stasko [Sta00] who presented an algorithm for drawing “circular” treemaps that convey clearly the hierarchical structure of the underlying tree. Treemaps count numerous applications in various application domains such as network management [Hei03], bioinformatics [MJL02], business intelligence [Shn06] and media [Wes04, OCSZ05, CF07], just to name a few.

### 1.5.10 Other Common Drawings

Other common graph drawing techniques that are widely used include trees and circular drawings. The most common technique for drawing trees is the one of Reingold and Tilford [RT81] that produces drawings which clearly reflect the hierarchy of the underlying tree. Similar level-based drawing algorithms have been presented by [Wal90, Blo93, BJL02]. Radial drawings of trees [Ber81, Ead92, CPM<sup>+</sup>98, CPP00, BM03, Bac07] have also been used in various applications. For an overview, refer to [DETT99, KW01, Rus07].

Circular drawings are commonly used in applications in which the structure and the topology of the underlying graph is important and has to be revealed (telecommunication, social networks, etc). In a circular drawing, the vertices of the graph form clusters and are placed on the circumference of circles, while the edges are usually drawn as straight-line segments. Algorithms for producing circular drawings have been presented by [TX94, Kre96, DMM96, Bra97, KW02]. For more details refer to [ST07].

## 1.6 Graph Drawing with Applications in Security

Over the past few decades, computers have become a critical component of human life. In fact, it is impossible to imagine our lives without computers, mobile devices and internet. Daily activities may include internet banking, e-shopping, bill payments, and so forth. In this increasingly connected society, the need for security and privacy has emerged as a key issue since connected computers are being misused at a growing rate from attackers in a wide variety of ways (viruses, unauthorized access, password steals, etc). Dennis Hughes of the FBI is frequently cited for the quote “*The only secure computer is one that’s unplugged, locked in a safe, and buried 20 feet under the ground in a secret location... and I’m not even too sure about*

*that one*", which reflects the difficulty of securing a computer (or a network) from pervasive threats.

Typically, *computer security* refers to the techniques and policies adopted by an administrator such that a computer or a network is able to prevent and defend to any type of threat (unauthorized access, misuse, denial-of-service attack, etc). More formally, the term *computer security* is defined by the NIST Computer Security Handbook [GR95] as "*the protection afforded to an automated information system in order to attain the applicable objectives of preserving the confidentiality, integrity, and availability of information system resources (includes hardware, software, firmware, information/data, and telecommunications)*". The three key words in this definition are confidentiality, integrity, and availability, referred to as the *C.I.A triad*, upon which the field of computer security has been built.

*Confidentiality* refers to the limitation of information access and disclosure only to authorized users and, to the existing means for protecting personal privacy and proprietary information. *Integrity* refers to the trustworthiness of information resources, i.e., the data that have been sent, received or stored are complete, accurate and have not been changed inappropriately or corrupted, whether by accident or by deliberately malign activity. *Availability* assures that the information systems work promptly and services are not denied to authorized users. For basic background on computer security the reader is referred to [GR95, Sta99, GT11].

In order to protect a computer or a network from potential attacks in practice, administrators use several defense tools such as firewalls, intrusion detection systems and anti-virus softwares. In brief, *firewalls* are systems that aim to prevent unauthorized access to and from a private network, by filtering the incoming and outgoing traffic that flows through the network. *Intrusion detection systems (IDS)* are systems that monitor network traffic, detect suspicious activity and produce reports and alerts for the administrators. Finally, *anti-virus software* are designed to detect and disarm or remove malicious software programs, such as viruses, trojan horses, worms and so on.

However, the problem of maintaining the security of a computer or a network is a difficult task due to the complexity of the protocols being used and the fact that attackers employ sophisticated techniques that are usually a step forward from existing defense tools. Traditionally, the monitoring process is performed based on log-files produced by the defense tools that an administrator uses. However, log-files are in textual form and usually in non-uniform formats, since they are not produced by a single system and have to be analyzed with little or no automated support. This renders the monitoring process as a time-consuming task that hinders the immediate identification of potential network threats.

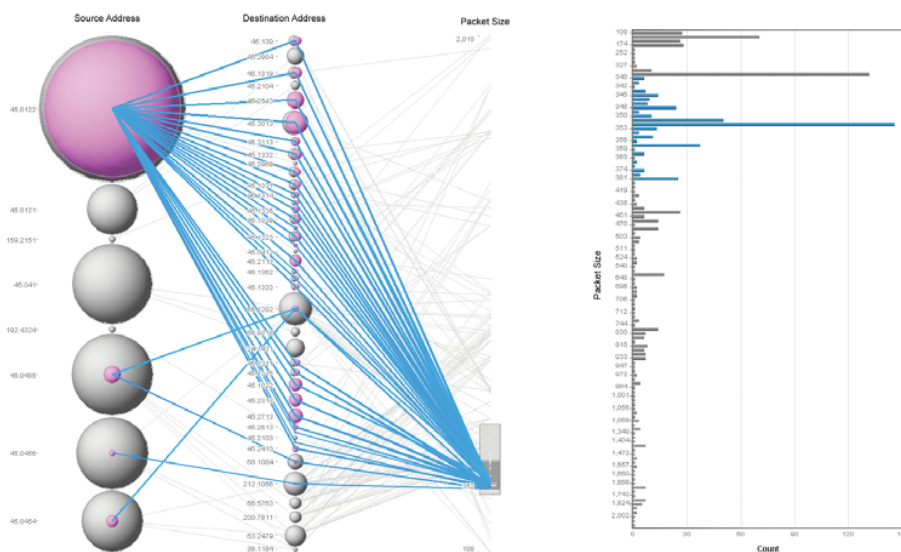
Due to the nature of data produced by such tools (large and complex), graph drawing

techniques can be effective in log-file analysis, since they facilitate the analysis of large-scale data. The visual representation of the information retrieved by log-files of a personal computer or a computer network is referred to as *security visualization* and is a fast-moving field [Con07, Mar08]. Security visualization aims at (i) discovering patterns or trends, (ii) detecting anomalous activity, (iii) understanding large-scale or small-scale features, and (iv) facilitating the corresponding decision making. Figure 1.1 represents a very small part of a firewall log-file stemming from a single personal computer in the National Technical University of Athens for a time-interval of two hours. Obviously, the analysis of such log files is a nuisance for a network administrator especially, in the case of larger time intervals. The problem increases significantly in the case where the analysis has to be performed not at a single host, but at the whole university network.

Rating	Date / Time	Type	Protocol	Source IP	Destination IP	Direction	Action Taken	Count	Source DHS	Destination DHS	Policy	Rule
Medi...	2009-01-29 14:20:34+2:00	Firewall	TCP (flags:S)	147.102.177.254:3...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:26:16+2:00	Firewall	TCP (flags:S)	147.102.37.1:43450	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:17:34+2:00	Firewall	TCP (flags:S)	147.102.37.1:24066	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:13:22+2:00	Firewall	TCP (flags:S)	147.102.177.254:4...	147.102.205.162:445	Incoming	Blocked	1		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:04:30+2:00	Firewall	TCP (flags:S)	147.102.177.254:3...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:00:24+2:00	Firewall	TCP (flags:S)	147.102.37.1:49950	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:41:16+2:00	Firewall	TCP (flags:S)	147.102.177.254:1...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:41:10+2:00	Firewall	TCP (flags:S)	147.102.177.254:1...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:30:00+2:00	Firewall	TCP (flags:S)	147.102.177.254:3...	147.102.205.162:445	Incoming	Blocked	1		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:07:24+2:00	Firewall	TCP (flags:S)	147.102.37.1:60220	147.102.205.162:445	Incoming	Blocked	1		ZEUS	callisto-nouplo...	ExtPortsIn

**Figure 1.1:** A log file produced by a firewall for a time interval of two hours.

Figure 1.2 is a great example of how visualizations can help in identifying potential attacks that are hidden between the lines of a log-file. The figure is retrieved by the book of Marty, “Applied Security Visualization” ([Mar08], pp. 264). The visualization is a *parallel coordinates plot* which shows different packets that are transmitted in a network. The vertices are ordered in three vertical layers. The vertices of the leftmost layer correspond to the IP addresses of senders (source IPs). The vertices of the middle layer correspond to the IP addresses of receivers (destination IPs). The vertices of the rightmost layer represent the size of the packets being sent during the communication. The size of each vertex of the first two layers is proportional to their degree of communication with the network hosts. From this particular figure, it is clear that the pink-colored vertex of the leftmost layer communicates with almost all vertices of the middle layer and in fact, the size of the packets being sent is almost the same in all these cases. Hence, a network administrator could easily imply that a *botnet attack* is in process, i.e., a network of compromised computers (refer to the vertices of the middle layer that communicate with the pink-colored vertex of the leftmost layer) is controlled by a third party (pink-colored vertex) and can be used for e.g., sending spam email messages, spreading malware, attacking computers, or committing other types of crime and fraud.



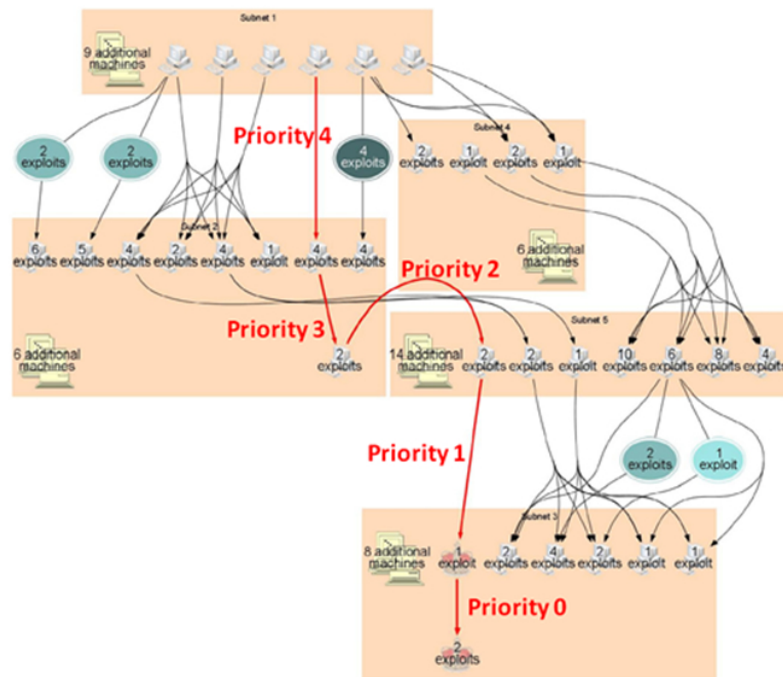
**Figure 1.2:** A visualization illustrating a botnet attack. Retrieved by the book “Applied Security Visualization” ([Mar08], pp. 264).

An important task in security is detecting and evaluating the vulnerabilities of the network. This can be accomplished by generating *attack graphs*. Figure 1.3 illustrates an attack graph in which vertices correspond to computers of the network (vertices with potential exploits are also labeled), while the edges represent how an attacker is able to gain access to a computer exploiting its vulnerabilities (reproduced by [NJ07]). The red-colored path represents one of the paths that an attacker can use in order to reach a specific host (destination). Obviously, attack graphs facilitate the identification of potential vulnerabilities in a network and can be used to assure the security of a network.

Several aspects in the field of security have been studied through visualizations. Among them are network traffic monitoring, intrusion detection, malware propagation, access control, attack graphs, border gateway protocol (BGP) and fraud detection. For an overview refer to [OPP07]. As expected, several standard graph drawing techniques such as layered drawings, force-directed algorithms or clustering have been employed to this end.

### 1.6.1 Applications in Network Monitoring

A fundamental issue in the area of computer security is *network monitoring*. This process includes both the inspection of the status of the devices of the network and the monitoring of the network traffic. The status of the devices can be captured by systems such as *network management softwares* that are able to detect failures or slowdowns and automatically inform the administrators. *Traffic monitoring* refers to the process of capturing network traffic



**Figure 1.3:** A visualization illustrating an attack graph [NJ07].

in order to identify anomalies and suspicious activity, such as denial of access attacks, port scanning, worms, overloads, and so on. This process is mainly performed by intrusion detection systems. Network monitoring includes also the monitoring of the activity of the hosts to reveal attacks or misuse stemming from within the network. However, even though intrusion detection systems are continuously developed in response to the increasing number of attacks, the detection of suspicious activity is becoming increasingly difficult, since attackers employ more and more sophisticated methods.

Many of the existing network monitoring software are equipped with visualizations in which graph drawing techniques are employed. One can divide these software in two categories. The first one includes softwares that support visualizations of data and alarms from intrusion detection systems, while the second one software that visualize the network traffic logs. The first IDS data visualizations focused on small scale data and on colored visualizations of the state and the forthcoming danger [DW01, FTM<sup>+</sup>98, VMF98], with limited capabilities and effectiveness. More recently, Tolle and Niggermann [TN05] presented an intrusion detection visualization system, based on a combination of force-directed and clustering algorithms that could control the state and evaluate the danger of the network. Through the resulting visualizations, false-positive states (abnormalities that do not correspond to intrusions) could also be recognized. Abdullah et al. [ALC<sup>+</sup>05] proposed a system that visualizes

IDS alarms on a large network, observing time patterns. Three-dimensional visualizations for the detection of abnormalities in a network have been used in [OR05, XMB<sup>+</sup>06]. Erbacher et al. [ECS05] studied visualization techniques for the analysis of network traffic data to ease the identification of sophisticated attacks above and beyond the ability for traditional network firewalls to detect and block.

In the area of traffic log analysis, Mansmann et al. [MMK08] presented an analytic tool that visualizes network host behavior through positional changes in a two dimensional space using a force-directed algorithm. Force-directed techniques have also been used by Girardin and Brodbeck [GB98] for the visualization of firewall or IDS log files, such that similar records are visualized relatively close to each other. Muelder et al. [MMB05] studied the problem of characterization network scans by using visualizations (force-directed and clustering algorithms) and statistics techniques. In the resulting visualizations, each vertex corresponds to a discrete scan, while the weights of the edges correspond to the degree of similarity between two vertices. Blue et al. [BDF<sup>+</sup>08] used force-directed techniques and treemaps to visualize computer network usage in real-time. Similar techniques have been used by Di Battista et al. [DSN12] that presented an animated visualization of migration of name servers between instances of the K-root server. Yin et al. [YYT<sup>+</sup>04] used graph visualizations, where the vertices are located on three parallel axes. The vertices that do not belong to the network are placed on the two external axes, while the vertices of the network in the middle one, using a heuristic algorithm that minimizes the number of crossings between edges. An edge between two vertices demonstrates a communication between them. Then, the identification of the intrusion is realized by pattern recognition. Similar visualizations with parallel axes have been used by Shabtai et al. [SKSE06] and Lakkaraju et al. [LYL04]. Vandenberghe [Van08] and Goodall et al. [GLRK05] used hierarchical approach techniques, whereas Toledo [Tol00], and Fink and North [FN05] used circular graph visualizations for monitoring network traffic.

### 1.6.2 Applications in Attack Graphs

The automated evaluation of the vulnerability of a network has emerged as a response to the continuously increasing number of network hosts and to the fact that networks are built upon multiple platforms, run various software and thus, it is likely that a number of vulnerabilities and security issues exist. A common technique employed by administrators in order to understand and evaluate the vulnerabilities of a network are the attack graphs that represent a series of exploits in a network (and the relationships among them) which an attacker can use to break into a network.

The generation of attack graphs was initially performed by Red Teams. However, graphs are produced by hand, usually contained errors and in fact, were impractical for large-scale

networks. The first steps towards the automation were performed by [PS98, RA00, SPEC01, SHJ<sup>+</sup>02], which however, faced problems due to the large number of data that had to be visualized. While most efforts focused mainly on the generation of an attack graph, Noel and Jacobs [NJ04] were the first that tried to produce a visualization that will be readable and comprehensive using layered drawings. Similar techniques have also been used by Noel et al. [NJK05, NJ07]. Homer et al. [HVOM08] presented methodologies that aim to identify parts of an attack graph that can be eliminated since they do not include useful information and group together similar attack steps into virtual vertices in order to identify the type of attack in which each computer of the network is exposed. O'Hare et al. [ONP08] presented a graph-theoretic approach to the problem of network risk visualization by generating attack graphs produced through Topological Vulnerability Analysis (TVA) [JN10]. Their approach was built upon the extensive functionality of yWorks [yWo] that facilitates the generation of various graph drawing layouts. Chu et al. [CIL<sup>+</sup>10] presented a tool for attack graph visualization using a modified algorithm for constructing treemaps [BSW02].

### 1.6.3 Applications in Fraud Detection

According to the 2012 Report to the Nation [AoCFE12] of the Association of Certified Fraud Examiners (ACFE), *fraud* is defined as any attempt that aims to deceive another party to gain a benefit. In business terms, fraud refers to *the intentional misuse or abuse of the assets of a company* and may include employees, clients, vendors, computer crime, or other entities [AoCFE12]. The definition of fraud implies the existence of an "intruder" (an outsider or an insider) that gains access to information and resources (probably unauthorized) and for this reason, fraud can be considered as a certain type of intrusion which falls into the wider area of network monitoring.

Among other techniques, fraud detection is also performed using fraud management systems (FMS) that operate in similar manner as the intrusion detection systems i.e., analyze log-files and detect unauthorized use of information and resources. During the past few years, several approaches have been proposed to detect fraud. Most of them, make use of data-mining and pattern-matching techniques [GS95, BH02, KLSH04, PLSMG10]. However, there exist several tools that support visualizations of various kinds adopting several graph drawing techniques [GS95, KSH<sup>+</sup>98, Sen00, SGS<sup>+</sup>02, HLN09, DDLP10, Lue10, Syn11]. In all of the systems mentioned above, one can easily identify several different graph drawing styles in order to produce the corresponding visualization such as force-directed algorithms, circular drawing, treemaps and so on. A great deal of work during this Ph.D. thesis has been devoted to visualizations techniques that aim to detect fraud, and in particular, employee fraud. For this reason, fraud detection will be described in detail in Chapters 6- 9 of this thesis.



### 1.6.4 Applications in Other Areas of Computer Security

Other areas of the computer security in which graph drawing techniques are widely employed include those of monitoring the activity of the border gateway protocol, access control and trust negotiation. *Border Gateway Protocol (BGP)* is a protocol used to exchange routing information between gateway hosts in a network of autonomous systems (ASes). An *autonomous system (AS)* is a portion of the internet under a single administrative authority. In fact, BGP is the protocol that determines the optimal routing paths for the packets exchanged in a network. Due to the fact that BGP supports routing between autonomous systems, it is important to detect quickly routing failures or attacks within an autonomous system that may propagate and cause severe damage to the Internet. Di Battista et al. [DBMPP04] visualize a portion of the BGP routing graph for a particular prefix and time interval using force-directed techniques. An animation is also developed to represent the evolution of this portion of the graph within a specific time interval. Same techniques along with circular drawings have been employed by Teoh et al. [TRNC06] aiming at discovering BGP anomalies in real time. Oberheide et al. [OKB06] used 3D drawings to visualize BGP routing topology of the internet.

In the fields of access control and trust negotiation, Yee [Yee06] presented a model that enables the identification of private information vulnerabilities using layered drawings. Montemayor et al. [MFG<sup>+</sup>06] studied various visualization techniques including layered drawings in order to gain access control situational awareness. Layered drawings have been also used by Yao et al. [YSTW05] to visualize automated trust negotiation. Heitzmann et al. [HPPT08] used treemaps to represent the access control permissions in a hierarchical file system, while Pan and Xu [PX12] analyzed multi-domain access control policies. Anwar and Fong [AF12] developed a visualization tool to support the assessment of access control policies in social network systems. Mazzia et al. [MLA12] presented a tool that facilitates the comprehension of the data stemming from a social network of a user, by employing force-directed techniques.

## 1.7 Thesis Contribution

The contribution of this thesis can be mainly focused on the following two topics:

**Crossing Resolution Maximization Problem:** *The problem of determining a drawing into which the minimum angle at which any two edges cross is maximized.* Maintaining the crossing angles of a drawing large is essential in many application scenarios, since it is widely accepted (and confirmed by cognitive experiments) that small crossing angles deteriorate the readability of a drawing. The situation becomes even more complex in the case where the available space is limited such as in mobile devices, where it is

required to fit the drawing in small screens, usually of low resolution. Therefore, the problem of maximizing the crossing resolution of a drawing is of particular interest.

In this Ph.D. Thesis, we have mainly focused on right-angle crossing drawings. Initially, we proved that the problem of determining whether a graph admits a straight-line RAC drawing is  $\mathcal{NP}$ -hard by introducing an infinite class of graphs with unique RAC combinatorial embedding. We also introduced the geometric RAC simultaneous drawing problem which combines the study of right-angle crossing drawings and simultaneous graph drawing. For this problem, we studied classes of graphs that may admit such drawings. In addition, we studied a closely related problem, the geometric graph-dual RAC simultaneous drawing problem which dates back in 1963 to a question of W.T. Tutte and presented an algorithm for constructing such drawings in the case where the input graph is an outerplanar embedded graph. Finally, we introduced the problem of maximizing the total resolution of graphs i.e., maximizing the minimum of the angular and crossing resolution. For this problem, we presented drawings of asymptotically optimal total resolution for special classes of graphs (complete, complete bipartite graphs) and a force-directed algorithm that constructs drawings of large total resolution.

**Fraud detection:** *Includes techniques and methods needed for the identification of actual or potential fraud within an organisation.* Fraud affects many companies worldwide, regardless of their size causing them economic loss and liability issues towards their customers and other involved entities. Our motivation to work on fraud detection rests on real problems raised by internal auditors of a large telecommunication organization. We have mainly focused on the problem of detecting fraud committed by employees, i.e., occupational or employee fraud. The data-sets stemmed from business systems involving pairs of entities (employee-client) such as billing systems, membership renewal systems and so on. Internal auditors suggested fraud patterns commonly detected in occupational fraud schemes based on which we have built our approaches. Among them is the identification of events and relate a pair of employee-client that appear in regular time basis. These events may imply for instance, that an employee deliberately falsifies the account of a client (deletes/add invoices, etc).

Towards this direction, we introduced a metric appropriately defined to reveal the frequency in which an event occurs and visualization techniques based on concentric circles that place close to the center of the system pairs of entities that appear to have periodic activity. In a second approach, we built a system that supports a detection procedure based on fraud patterns suggested by the auditors and appropriate spiral visualizations such that periodic events appear along the same radius or on close radii of the spiral. In our last approach, we developed an integrated fraud management visu-

alization system combining pattern recognition and visualizations (heat-maps, circular drawings, parallel coordinates plots) in order to detect occupational fraud schemes. We have extended the particular system in order to monitor employees' activity within the company network and identify suspicious activity.

## 1.8 Thesis Overview

In Chapter 2, we define the problems of maximizing the angular and the crossing resolution of a graph drawing. We also present the class of right-angle crossing graphs as the class of graphs admitting drawings of optimal crossing resolution. Finally, we introduce a problem called *geometric simultaneous right-angle crossing drawing* which combines the study of right-angle crossing drawings and simultaneous graph drawing.

In Chapter 3, we focus on straight-line RAC drawings (or geometric RAC drawings) and demonstrate an infinite class of graphs with unique RAC combinatorial embedding. We employ members of this class in order to show that it is  $\mathcal{NP}$ -hard to decide whether a graph admits a straight-line RAC drawing. Chapter 3 is joint work with Michael Bekos and Antonios Symvonis. A preliminary version of this work has been published in [ABS11], which was subsequently improved in [ABS12].

In Chapter 4, we study the geometric RAC simultaneous drawing problem. We first prove that two planar graphs admitting a geometric simultaneous drawing may not admit a geometric RAC simultaneous drawing. We further show that a cycle and a matching always admit a geometric RAC simultaneous drawing. We also study the geometric graph-dual RAC simultaneous drawing problem and we prove that it is always possible to construct such a drawing if the input graph is an outerplanar embedded graph. Chapter 4 is joint work with Michael Bekos, Michael Kaufmann and Antonios Symvonis. Preliminary versions of this work have been published in [ABKS11] and [ABKS12], and were subsequently improved in [ABKS13].

In Chapter 5, we introduce the problem of maximizing the total resolution of a graph drawing that is, the minimum of the angular and crossing resolution. We present drawings of asymptotically optimal total resolution for complete graphs (circular drawings) and for complete bipartite graphs (2-layered drawings). In addition, we present and experimentally evaluate a force-directed based algorithm that constructs drawings of large total resolution. Chapter 5 is joint work with Michael Bekos and Antonios Symvonis. A preliminary version of this work has been published in [ABS10], which was subsequently improved in [ABS13].

In Chapter 6, we present basic notions from the area of occupational fraud detection. We also present our motivation to work on fraud detection and some necessary notation that will be used in Chapters 7-9.

In Chapter 7, we present visualization techniques that aim to identify periodic patterns in serial data which involve pairs of entities (employee and client). The presented techniques can be applied to detect occupational fraud schemes. Towards this direction, we measure periodicity by introducing a new metric which is appropriately defined to reveal the frequency in which an event occurs within a time window. Our main visualization method results in drawings consisting of concentric circles whose radius correspond to the periodicity of the activity of each pair of employee and client. Events that are considered to be suspicious are easily identified since they are dragged towards the center of the circles. Chapter 7 is joint work with Antonios Symvonis. A version of this work has been published in [AS12].

In Chapter 8, we extend the system of Chapter 8 such that the visualization is not only limited to identifying periodic events that involve pairs of entities, but it provides a complete view of all the examined fraud patterns and the results of the examination on each pattern. Furthermore, the system monitors the activity of the employees aiming at detecting unusual employee activity such as unauthorized access to computers, business systems and accounts of employees or clients. Chapter 8 is joint work with Aikaterini Sotiraki and Antonios Symvonis. A version of this work has been published in [ASS13].

In Chapter 9, we present a system that exploits the advantages of information visualization and pattern recognition to detect suspicious patterns concerning billing fraud schemes in systems in which a pair of entities are involved. In the system, we have also incorporated a visualization that aims to monitor the activities of the employees and detect unusual employee activity such as unauthorized access to computers, business systems and accounts of employees or clients. Chapter 9 is joint work with Antonios Symvonis and Vassilis Vassiliou. A version of this work has been published in [ASV14].

In Chapter 10, we summarize the results presented in this Ph.D. thesis and we provide a report of some future extensions of our work.



## 2 The Angular and Crossing Resolution Maximization Problems

In this chapter, we formally define the problems of maximizing the angular and the crossing resolution of a graph drawing. We also present the class of right-angle crossing graphs as the class of graphs admitting drawings of optimal crossing resolution. Finally, we introduce a problem called *geometric simultaneous right-angle crossing drawing* which combines the study of right-angle crossing drawings and simultaneous graph drawing.

### 2.1 Motivation and Definitions

As already stated in Chapter 1, there exist several criteria that have been used to judge the quality of a graph drawing. From a human point of view, it is necessary to obtain drawings that are easy-to-read, i.e., they should nicely convey the structure of the objects and their relationships. From an algorithmic point of view, the quality of a drawing is usually evaluated by some objective function and the main task is to determine a drawing that minimizes or maximizes the specific objective function. Various such functions were presented in Chapter 1, which have been extensively studied by the graph drawing community, among them, the number of crossings among pairs of edges, the number of edge bends, the maximum edge length, the total area occupied by the drawing, and so on.

A property that may negatively influence the readability of a drawing is the presence of edges that are too close to each other, especially if these edges are incident to the same vertex. Various applications such as mobile devices or video screens, are usually equipped with a relatively small screen of low resolution and therefore several readability problems occur due to resolution limitations. In communication network design, it is also crucial to draw the networks with large angles in order to facilitate the transmission of signals. These motivated the study of drawings in which the angles among edges incident to the same vertex are as large as possible. Formally, we define the *angular resolution* of a graph drawing as follows:

**Definition 2.1.1. (Angular Resolution)** *The angular resolution of a graph drawing is the size of the minimum angle formed by any two edges incident to a common vertex.*

Accordingly, we define the *angular resolution of a graph* as the maximum angular reso-

lution of any (straight-line) drawing of the graph. An obvious upper bound on the angular resolution of a graph of maximum degree  $d$  is  $\frac{2\pi}{d}$ . However, this bound is not tight for many graphs. For instance, for  $K_3$  graph (i.e., the complete graph of three vertices), which is of maximum degree two, it is not difficult to see that any of its drawings is of angular resolution at most  $\pi/3$ . The corresponding maximization problem is referred to as the *angular resolution maximization problem* and can be stated as follows:

**Problem 2.1.1. (Angular resolution maximization problem)** *Given a graph  $G = (V, E)$  and a positive non-zero real number  $r \in \mathbb{R}^+$ , does there exist a drawing  $\Gamma(G)$  of  $G$  with angular resolution at least  $r$ ? In the case of an affirmative answer, we are interested in the actual drawing  $\Gamma(G)$ .*

Another undesired property that may negatively affect the quality of a graph drawing are the edge crossings, i.e., if too many edges cross each other, the human eye cannot distinguish which vertices are connected by an edge. This intuition has also scientific validity. Eye-tracking experiments have shown that performance of humans in path tracing tasks is negatively correlated to the number of edge crossings [Pur00, PCA02]. Therefore, if a graph can be drawn planar, then it is preferable to a drawing with edge crossings. To compute a planar drawing of a planar graph is not too difficult, even if we restrict ourselves to straight line edges. Unfortunately, as already stated in Chapter 1, very few graphs are planar in practice and edge crossings are unavoidable in the vast majority of the application scenarios. This, together with the computational complexity of the edge crossing minimization problem, which is  $\mathcal{NP}$ -complete in general [GJ83] apparently leads to the conclusion that the computation of high-quality drawings of non-planar graphs is significantly difficult and how to handle the crossings of non-planar graphs still remains partially answered.

The above conclusion was an obstacle in any graph drawing scenario that involves the visualization of non-planar graphs and subsequently of large and complex graphs. However, more recent cognitive experiments on large drawings of graphs indicated that the negative impact of an edge crossing on the human understanding of a graph drawing tends to be eliminated in the case where the crossing angles are large. More precisely, Huang et al. [Hua07, HHE08, HH10, Hua13] showed through eye-tracking experiments that crossings that occur at angles greater than  $70^\circ$  have little effect on the ability of humans to interpret graphs. These results motivated the study of the crossing resolution problem which is a relatively new graph drawing problem.

**Definition 2.1.2. (Crossing resolution)** *The crossing resolution of a graph drawing is defined as the size of the minimum angle formed by any two crossing edges.*

Accordingly, the *crossing resolution of a graph* is defined as the maximum crossing resolu-

tion over all (straight-line) drawings of the graph. The corresponding maximization problem is referred to as *crossing resolution maximization problem* and is formally defined as follows:

**Problem 2.1.2. (Crossing resolution maximization problem)** *Given a graph  $G = (V, E)$  and a positive non-zero real number  $r \in \mathbb{R}^+$ , does there exist a drawing  $\Gamma(G)$  of  $G$  with crossing resolution at least  $r$ ? In the case of an affirmative answer, we are interested in the actual drawing  $\Gamma(G)$ .*

As a response to the problem of drawing graphs with optimal crossing resolution, a new class of graphs, referred to as right-angle crossing graphs, were introduced by Didimo, Eades and Liotta [DEL09]. The formal definition follows:

**Definition 2.1.3. (Right-angle crossing graphs)** *A graph is called right-angle crossing (or RAC for short) graph if it admits a drawing in which every pair of crossing edges intersects at right angle. Accordingly, a right-angle crossing (or RAC for short) drawing of a graph is a polyline drawing in which every pair of crossing edges intersects at right angle.*

Combining the crossing and the angular resolution maximization problems, suggests a new interesting problem, referred to as the *total resolution maximization problem*, in the process of obtaining drawings of high total resolution. The formal definition of the total resolution is as follows:

**Definition 2.1.4. (Total resolution)** *The total resolution of a graph drawing is defined as minimum of the angular and the crossing resolution of the drawing.*

Accordingly, the *total resolution of a graph* is defined as the maximum total resolution over all (straight-line) drawings of the graph. The corresponding maximization problem is referred to as *total resolution maximization problem* and is formally defined as follows:

**Problem 2.1.3. (Total resolution maximization problem)** *Given a graph  $G = (V, E)$  and a positive non-zero real number  $r \in \mathbb{R}^+$ , does there exist a drawing  $\Gamma(G)$  of  $G$  with total resolution at least  $r$ ? In the case of an affirmative answer, we are interested in the actual drawing  $\Gamma(G)$ .*

## 2.2 Related Work

This section overviews existing results from the literature for the angular and crossing resolution maximization problems and known results for the class of right-angle crossing graphs.

### 2.2.1 Angular Resolution

Formann et al. [FHH<sup>+</sup>93] introduced the notion of the angular resolution of straight-line drawings in 1990. In their work, they proved that determining whether a graph of maximum degree  $d$  admits a drawing of angular resolution  $\frac{2\pi}{d}$  (i.e., the obvious upper bound) is  $\mathcal{NP}$ -hard. They also showed that a graph of maximum degree  $d$  has a straight-line drawing of angular resolution  $\Omega(\frac{1}{d^2})$  and that every planar graph of maximum degree  $d$  admits a straight-line (nonplanar) drawing of angular resolution  $\Omega(\frac{1}{d})$ . They also presented upper and lower bounds on the angular resolution for several types of graphs of maximum degree  $d$ , which are independent of the number of vertices of the graph.

Malitz and Papakostas [MP94] studied the angular resolution maximization problem of planar straight-line drawings and proved that for any planar graph of maximum degree  $d$ , it is possible to construct a planar straight-line drawing with angular resolution  $\Omega(\frac{1}{7d})$ . Again, the bound is independent of the number of vertices of the graph. Kant [Kan92, Kan96] showed that testing whether a biconnected planar graph admits a straight-line drawing with angular resolution greater than or equal to a given constant is  $\mathcal{NP}$ -hard. Furthermore, he showed that a triconnected planar graph of maximum degree  $d$  admits a planar polyline drawing with angular resolution  $\Theta(\frac{1}{d})$  in an  $O(n) \times O(n)$  integer grid, so that each edge has at most three bends.

Garg and Tamassia [GT94] showed the first nontrivial upper bound for the angular resolution maximization problem. In particular, they proved that there exists an infinite class of graphs of maximum degree  $d$  that are of angular resolution  $O(\sqrt{\log d/d^3})$  in any planar straight-line drawing. They also showed a continuous tradeoff between the area and the angular resolution of planar straight-line drawings and presented linear time algorithms for producing planar drawings of high angular resolution for certain classes of graphs.

Di Battista and Vismara [DV93] presented a nonlinear system of equalities and inequalities relating its angles to provide a characterization of all planar drawings of triangular graphs. In this manner, they showed that testing whether an assignment of angles between the edges of a planar triangular graph can result in a planar straight-line drawing can be done in linear time. They also showed that the problem of maximizing the minimum angle in a planar straight-line drawing of a triangular graph is reduced to a nonlinear optimization problem.

Gutwenger and Mutzel [GM99] presented a linear time algorithm that constructs planar polyline grid drawings of planar graphs with  $n$  vertices on a  $(2n - 5) \times (\frac{3}{2}n - \frac{7}{2})$  integer grid with at most  $5n - 15$  bends and minimum angle greater than  $\frac{2}{d}$ . Goodrich and Wagner [GW00] presented an algorithm that requires one bend less per edge, while asymptotically the angular resolution is maintained the same i.e.,  $\Theta(\frac{1}{d})$ . However, their algorithm requires larger grid area i.e., their drawings fit in a  $(20n - 48) \times (10n - 24)$  grid area. Cheng et al. [CDGK99]



improved the algorithm mentioned above by presenting an algorithm that also results in drawings of asymptotically optimal angular resolution, but uses at most one bend per edge. As expected, their drawings require larger drawing area, in particular, their drawings fit in a  $30n \times 15n$  area. Kobourov and Duncan [DK03] presented two algorithms that improved further the above results. The first one is a cartesian representation algorithm (CRA) that achieves one bend per edge,  $\frac{1}{2d}$  angular resolution and,  $5n \times 5n$  drawing area. The second algorithm is based on polar representation and guarantees one bend per edge,  $\frac{\pi}{4d}$  angular resolution and,  $9n \times \frac{9n}{2}$  drawing area.

More recently, Eppstein and Wortman [EW11] for a special type of drawings in which each bounded face of the graph is required to be drawn as centrally symmetric polygon presented a polynomial time algorithm to obtain drawings of optimal angular resolution.

Several methods also have been proposed for optimizing the angular resolution of tree drawings. Eades [Ead92] proposed a radial drawing algorithm which is suitable for drawing free trees (i.e., trees that do not represent hierarchies and have no special root) in two dimensions. Recursive radial drawing algorithms result in circular tree drawings that achieve high angular resolution [MH98]. Grivet et al. [GADM06] presented an algorithm for drawing general rooted trees in such a way that trees are drawn recursively and each subtree belongs to a circle disjoint from its siblings, but within the one of its parent. Balloon drawings follow similar approaches and aim to optimize the angular resolution as well as the ratio between the longest to the shortest edges in the tree [LY07].

Brandes and Schlieper [BS06] studied planar drawings of trees with constraints on the angles and on the distances between adjacent vertices. They showed that in linear time one can test whether these constraints are realizable, and they presented an algorithm that satisfies both constraints producing a polyline drawing in which edges have at most two bends per edge or are drawn as continuously differentiable curves. Didimo et al. [DKL<sup>+</sup>12] studied the problem of leveled tree drawings and tried to maximize the crossing angle resolution and/or the angular resolution.

Another approach towards the optimization of angular resolution are Lombardi drawings inspired by the American abstract artist Mark Lombardi [LH03b]. Circular-arc edges and their even distribution around vertices formally define a *Lombardi drawing* of a graph [DEG<sup>+</sup>10b, DEG<sup>+</sup>12]. Hence, Lombardi drawings achieve optimal angular resolution. However, not all graphs admit Lombardi drawings and the characterization of Lombardi graphs still remains an open problem. Non-Lombardi graphs can be visualized in a Lombardi-style by relaxing the circular-arc constraint and more precisely, by representing each edge as a smooth sequence of circular arcs. Duncan et al. [DEG<sup>+</sup>11] presented  $k$ -Lombardi drawings, in which edges consist of smooth sequences of  $k$  circular arcs. Duncan et al. [DEG<sup>+</sup>10a] presented an

algorithm that produces planar, straight-line drawings of unordered trees (with optimal angular resolution) in polynomial area (including Lombardi-style drawings). Matsakis [Mat10] presented a force-directed algorithm that results in near-Lombardi drawings. Chernobelski et al. [CCG<sup>+</sup>12] presented two force-directed algorithms that construct Lombardi-style drawings in which all edges are still circular arcs, but some vertices may not have optimal angular resolution.

Several force-directed algorithms have been proposed for the angular resolution maximization problem. Brandes and Wagner [BW98] presented a force-directed algorithm for visualizing interconnections in railway systems of high angular resolution in which the vertex positions are fixed but transitive edges are drawn as Bezier curves. Their work was extended by Finkel and Tamassia [FT04] who used curvilinear edges in their force-directed algorithm and relaxed the constraint of having fixed the vertex positions. Brandes et al. [BST00] visualized large-scale transportation networks by replacing straight-line and great-circle connections with cubic curves. Lin and Yen [LY05] presented a force-directed method that leads to drawings with high angular resolution. In their work, pairs of edges incident to a common vertex are modeled as charged springs, that repel each other, guaranteeing the absence of any overlap among edges incident to a common vertex.

Eppstein et al. [ELMN13] studied the problem of angular resolution in 3D drawings. They showed that every graph of maximum degree three can be drawn in three dimensions crossing-free with at most two bends per edge and, all angles between pairs of edge segments that meet at a vertex or a bend form  $120^\circ$  angles. They also showed how to draw graphs of maximum degree four in three dimensions with optimal angular resolution, three bends per edge, integer vertex coordinates, and cubic volume. Garg et al. [GTV96] studied 3D straight-line drawings considering various resolution criteria, but they did not constrain vertices to belong to a 2D surface. Goodrich and Pszona [GP13] studied arc diagrams and provided algorithms that achieve good angular resolution for 3D arc diagrams, even if arcs must project to a given 2D straight-line drawing of the input graph.

### 2.2.2 Crossing Resolution and RAC Graphs

The first work towards the crossing resolution maximization problem was published in 2009 by Didimo et al. [DEL09, DEL11] who combinatorially studied the ideal case of right-angle crossing drawings. In their work, they proved that the class of straight-line RAC drawings is a proper subset of the class of straight-line, *quasi-planar drawings*, in which no three edges are pairwise crossing and they further studied the interplay between the number of vertices and the *curve complexity* (i.e., the maximum number of bends per edge) in RAC drawings. They showed that any graph with  $n$  vertices, which admits a straight-line RAC drawing

(i.e., of zero curve complexity), has at most  $4n - 10$  edges and this bound is tight, i.e., there are infinitely many values of  $n$  for which there exists a straight-line RAC drawing of a graph with  $n$  vertices and  $4n - 10$  edges. Therefore, the requirement of having straight-line edges must be relaxed if one wishes to draw non-sparse, non-planar graphs with optimal crossing resolution. In fact, it is always possible to construct a RAC drawing of a given graph with curve complexity three, whereas if a RAC drawing of a graph with  $n$  vertices has curve complexity one or two, then its number of edges must be  $O(n^{7/4})$  and  $O(n^{4/3})$ , respectively [DEL09, DEL11]. The last two bounds were improved by Ariskushi et al. [AFK<sup>+</sup>10] who showed that if an  $n$ -vertices graph admits a RAC drawing with at most one (two bends, resp.) per edge, this graph has at most  $6.5n - 13$  ( $74.2n$ , resp.) edges. Upper and lower bounds on the crossing resolution of straight-line drawings of complete graphs have been presented by Di Giacomo et al. [DDE<sup>+</sup>12].

Angelini et al. [ACD<sup>+</sup>09] studied tradeoffs between optimal crossing resolution and other typical graph drawing constraints, including area requirement and edge upwardness for drawings of directed acyclic graphs. They proved that the problem of determining whether an acyclic planar digraph admits a straight-line upward RAC drawing is  $\mathcal{NP}$ -hard and they further constructed digraphs admitting straight-line upward RAC drawings which require exponential area. Didimo et al. [DEL10] showed that determining whether a bipartite graph admits a straight-line RAC drawing can be done in linear time, and they further presented an  $O(n^2 \log n)$  algorithm that computes the corresponding drawing, which simultaneously minimizes the total number of edge crossings. They also proved that the problem of determining the maximum subgraph of a given non-RAC bipartite graph that admits a straight-line RAC drawing is  $\mathcal{NP}$ -hard.

Eades and Liotta [EL13] introduced and studied properties of special subclass of RAC graphs called *maximally dense RAC graphs*, i.e., RAC graphs with  $n$  vertices and exactly  $4n - 10$  edges. They proved that a maximally dense graph is *1-planar*, i.e., it admits a drawing in which each edge is crossed at most once. They also showed that for every integer  $i \geq 0$ , there exists a 1-planar graph with  $n = 8 + 4i$  vertices and  $4n - 10$  edges that is not a RAC graph. On the other hand, Dehkordi and Eades [DE12] showed that every *outer 1-plane graph* (i.e., a topological graph in which each vertex appears on the outer face and each edge is crossed at most once) has a RAC drawing with the same embedding. The results mentioned above imply that the RAC graphs range between outer 1-plane and 1-planar graphs.

Van Kreveld [vK10] showed that the quality of a planar drawing of a planar graph, evaluated in terms of area required, edge-length and angular resolution, can be improved if one allows right-angle crossings. Fink et al. [FHM<sup>+</sup>11] studied the problem of crossing resolution in the case where vertices are at specified positions. In particular, they allow large crossing an-

gles and measure their effectiveness in terms of computational complexity, curve complexity with respect to the planar drawing. Nguyen et al. [NEHH10] presented a quadratic programming method that constructs a circular drawing with high crossing resolution. Dehkordi et al. [RDNEH13] provide a characterization of circular RAC graphs and present an algorithm that constructs a circular RAC drawing in linear time. They also propose a quadratic programming approach for the problem of maximizing the crossing resolution of a drawing.

Didimo et al. [DLR10] proposed a framework which combines force-directed and planarization approaches to obtain polyline drawings of non-planar graphs with a small number of crossings, high crossing resolution and good geodesic edge tendency<sup>1</sup>. In the context of polyline drawings again, crossing resolution and geodesic edge tendency are also the main optimization criteria of a force-directed algorithm of Didimo et al. [DLR11] that computes a simultaneous embedding of two input graphs in order to support web site traffic analysis by means of graph visualization techniques. Huang et al. [HEHL10] studied the effectiveness of compromises between aesthetics, and present a force-directed algorithm that maximizes the crossing resolution in a drawing.

Di Giacomo et al. [DDL09, DDE<sup>+</sup>12], Dujmovic et al. [DGMW10], and, Ackerman et al. [AFT10] studied the crossing resolution maximization problem by relaxing the right-angle constraint on the crossing resolution. Given a positive real number  $a$ , such that  $0 < a < \pi/2$ , a drawing is called aLAC (aEAC, resp.) if the minimum angle at which two edges cross is at least (exactly, resp.)  $a$ , whereas a graph that admits an aLAC (aEAC, resp.) drawing is called aLAC (aEAC, resp.) graph. Dujmovic et al. [DGMW10] proved that a straight-line aLAC drawing of a graph with  $n$  vertices has at most  $(3n - 6)\pi/a$  edges. If the requirement of having straight-line edges is relaxed, Di Giacomo et al. [DDL09, DDE<sup>+</sup>12] showed that it is always feasible to construct an aLAC drawing of a given graph with curve complexity one. Ackerman et al. [AFT10] proved that an aEAC drawing of a graph with  $n$  vertices and curve complexity one (two, resp.) has at most  $27n$  ( $477n$ , resp.) edges. Angelini et al. [ABD<sup>+</sup>12] provided techniques for constructing polyline drawings of planar graphs of large crossing resolution in subquadratic area. Figure 2.1 summarizes all bounds presented above. The shaded in gray cells correspond to bounds that are not known whether they are tight or not. For a survey on the problem of crossing resolution, the reader is referred to [DL13].

<sup>1</sup>For an edge  $e = (u, v)$  with bends, the term geodesic edge tendency refers to the maximum distance from a point of  $e$  to the straight-line segment connecting  $u$  and  $v$ .

Problem	Straight-line	1 bend/edge	2 bends/edge	3 bends/edge
RAC	$\leq 4n - 10$ [DEL09]	$\leq 6.5 - 13$ [AFK+10]	$\leq 74n$ [AFK+10]	No restrict [DEL09]
aLAC	$\leq (3n - 6)\pi/a$ [DGMW10]	No restrict [DDL09]	No restrict [DDL09]	No restrict [DDL09]
aEAC	$\leq 3(3n - 6)$ [AFT10]	$\leq 27n$ [AFT10]	$\leq 477n$ [AFT10]	No restrict [AFT10]

**Figure 2.1:** Known results on RAC and aLAC/aEAC drawing problems.

## 2.3 Preliminaries and Notation

In this section, we present some properties of right-angle crossing drawings that will be heavily used throughout this thesis.

**RAC-Property 1** (Didimo, Eades and Liotta [DEL11]). *In a straight-line RAC drawing there do not exist three mutually crossing edges.*

**RAC-Property 2** (Angelini et al. [ACD<sup>+</sup>09]). *Let  $(u, v)$  and  $(u, w)$  be a pair of non-overlapping edges incident to the same vertex. We say that  $(u, v)$  and  $(u, w)$  form a fan anchored at  $u$ . In a straight-line RAC drawing no edge can cross a fan.*

**RAC-Property 3** (Didimo, Eades and Liotta [DEL11]). *In a straight-line RAC drawing there does not exist a triangle  $\mathcal{T}$  formed by edges of the graph and two edges  $(a, b)$  and  $(a, b')$ , such that  $a$  lies outside  $\mathcal{T}$  and  $b, b'$  lie inside  $\mathcal{T}$ .*

**RAC-Property 4** (Eades and Liotta [EL11]). *The external face of any RAC drawing of a maximally dense RAC graph is bounded by three edges.*

## 2.4 Simultaneous RAC drawings

In this section, we present a problem that combines the study of right-angle crossing drawings and simultaneous graph drawing. We provide the necessary terminology and present an overview on existing literature.

### 2.4.1 Motivation and Problem Definition

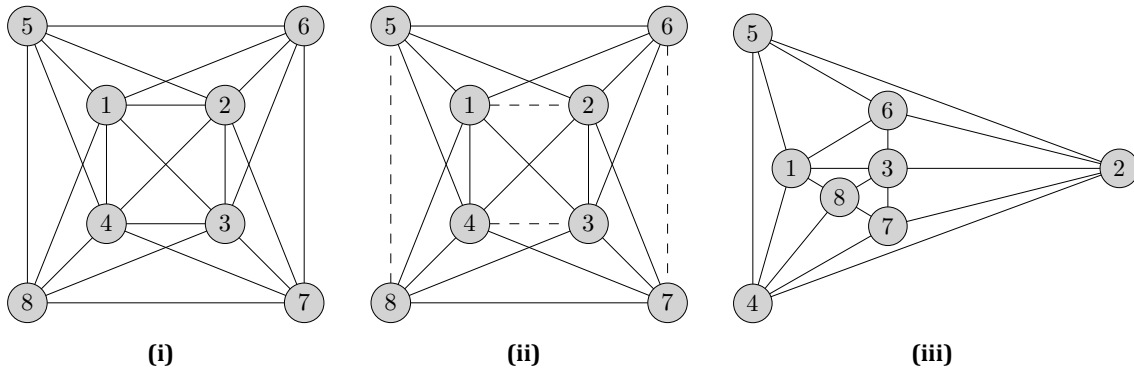
As already mentioned in Chapter 1, simultaneous graph drawing deals with the problem of drawing two (or more) planar graphs on the same set of vertices on the plane, such that each graph is drawn planar (i.e., only edges of different graphs are allowed to cross). The *geometric* version restricts the problem to straight-line drawings. The *geometric RAC simultane-*

ous drawing problem (or *GRacSim* drawing problem, for short) is a combination of right-angle crossing drawings and simultaneous graph drawing and is formally defined as follows:

**Problem 2.4.1. (Geometric RAC simultaneous drawing problem)** Let  $G_1 = (V, E_1)$  and  $G_2 = (V, E_2)$  be two planar graphs that share a common vertex set. The geometric RAC simultaneous drawing problem asks for a placement of the vertices on the plane such that, when the edges are drawn as straight-line segments, (i) each graph is drawn planar, (ii) there are no edge overlaps and (iii) crossings between edges in  $E_1$  and  $E_2$  occur at right angles.

The *GRacSim* drawing problem is of interest, since it combines two current research topics in graph drawing. Our motivation to study this problem rests on the work of Didimo et al. [DEL11] who proved that the crossing graph of a geometric RAC drawing is bipartite<sup>2</sup>. Thus, the edges of a geometric RAC drawing of a graph  $G = (V, E)$  can be partitioned into two sets  $E_1$  and  $E_2$ , such that no two edges of the same set cross. So, the problem we study is, in a sense, equivalent to the problem of finding a geometric RAC drawing of an input graph (if one exists), given its crossing graph.

Let  $G = (V, E_1 \cup E_2)$  be the graph induced by the union of  $G_1$  and  $G_2$ . Observe that  $G$  should be a RAC graph, which implies that  $|E_1 \cup E_2| \leq 4|V| - 10$  [DEL11]. We refer to this relationship as the *RAC-size constraint*.



**Figure 2.2:** (i) A graph with 8 vertices and 22 edges which does not admit a RAC drawing [EL11]. (ii) A decomposition of the graph of Figure 2.2i into a planar graph (solid edges; a planar drawing is given in Figure 2.2iii) and a matching (dashed edges), which implies that a planar graph and a matching do not always admit a *GRacSim* drawing; their union is not RAC.

If two graphs do not admit a geometric simultaneous drawing they, obviously, do not admit a *GRacSim* drawing. For instance, since it is known that there exists a planar graph and a

<sup>2</sup>This can be interpreted as follows: “If two edges of a geometric RAC drawing cross a third one, then these two edges must be parallel.”

matching that do not admit a geometric simultaneous drawing [CvKL<sup>+</sup>11], as a consequence, the same graph and matching do not admit a GRacSim drawing either. Figure 2.2 depicts an alternative and simpler technique to prove such negative results for GRacSim drawings, which is based on the fact that not all graphs that obey the RAC-size constraint are actually RAC graphs. On the other hand, as we will show in Chapter 4, two planar graphs admitting a geometric simultaneous drawing may not admit a GRacSim drawing.

A closely related problem to the GRacSim drawing problem is the *geometric Graph-Dual RAC simultaneous drawing problem* (or *GDual-GRacSim* for short) which is stated as follows:

**Problem 2.4.2. (Geometric Graph-Dual RAC simultaneous drawing problem)** *Given a planar embedded graph  $G$ , determine a geometric drawing of  $G$  and its weak dual  $G^*$  (i.e., without the face-vertex corresponding to the external face) such that: (i)  $G$  and  $G^*$  are drawn planar, (ii) each vertex of the dual is drawn inside its corresponding face of  $G$  and, (iii) the primal-dual edge crossings form right angles.*

Note that, the GDual-GRacSim drawing problem is not a new problem. Back in 1963, W.T. Tutte asked whether “Can we construct simultaneous straight representations . . . of  $G$  and  $G^*$  in which the . . . corresponding edges are represented by perpendicular segments ?” [Tut63, p.p.767].

### 2.4.2 Related Work

Brass et al. [BCD<sup>+</sup>07] presented algorithms for drawing simultaneously: (a) two paths in  $n \times n$  grid area, (b) a path and an extended star in  $O(n^2) \times O(n)$  area, (c) a caterpillar and a path in  $3n \times 2n$  area, (d) two caterpillars in  $3n \times 3n$  area, (e) two stars (resp.  $k$  stars) in  $3 \times (n - 1)$  area (resp.  $O(n) \times O(n)$ ) and, (f) two cycles in  $4n \times 4n$  grid area. On the negative side, they also proved that there exist (a) three paths and (b) two outerplanar graphs that do not admit a geometric simultaneous drawing. Erten and Kobourov [EK05b] showed that a planar graph and a path cannot always be drawn simultaneously. Frati, Kaufmann and Kobourov [FKK09] proved this negative result, even for the case where the planar graph and the path do not share any edges.

Duncan et al. [DEK04] showed that a graph of maximum degree four has geometric thickness two, where the geometric thickness is the minimum number of layers required to draw the graph on the plane with straight-line edges and assign each edge to a layer such that no two edges of the same layer cross. Hence, two graphs with maximum degree two always admit a simultaneous geometric drawing. The drawback of their algorithm is that the resulting drawings require large area. Cabello et al. [CvKL<sup>+</sup>11] showed that a geometric simultaneous drawing of a matching and (a) a wheel, (b) an outerpath or (c) a tree always exists. They also

showed that there exist (a) a planar graph and a matching and (b) six matchings that cannot be drawn simultaneously.

Geyer, Kaufmann and Vrt'o [GKV09] showed that a geometric simultaneous drawing of two trees does not always exist. Angelini et al. [AGKN10] proved that the same holds for a path and a tree. Cappos et al. [CEBFK09] presented a linear-time algorithm that constructs a geometric simultaneous drawing of a planar graph and a path in the case where the planar graph is level-planar with respect to the layering induced by the path. Estrella-Balderrama et al. [EBG]<sup>+</sup>07] proved that the problem of determining whether two planar graphs admit a geometric simultaneous drawing is  $\mathcal{NP}$ -hard. For a quick overview of known results on this research area refer to [BKR07, BKR12].

Brightwell and Scheinermann [BS93] proved that the geometric graph-dual RAC simultaneous drawing problem always admits a solution if the input graph is a triconnected planar graph. To the best of our knowledge, this is the only result which incorporates the requirement that the primal-dual edge crossings form right angles. Erten and Kobourov [EK05a] presented an  $O(n)$  time algorithm that results into a simultaneous drawing, but not a RAC drawing of a triconnected planar graph and its dual on an  $O(n^2)$  integer grid, where  $n$  is the total number of vertices in the graph and its dual.





## 3 The Straight-line RAC drawing problem is NP-Hard

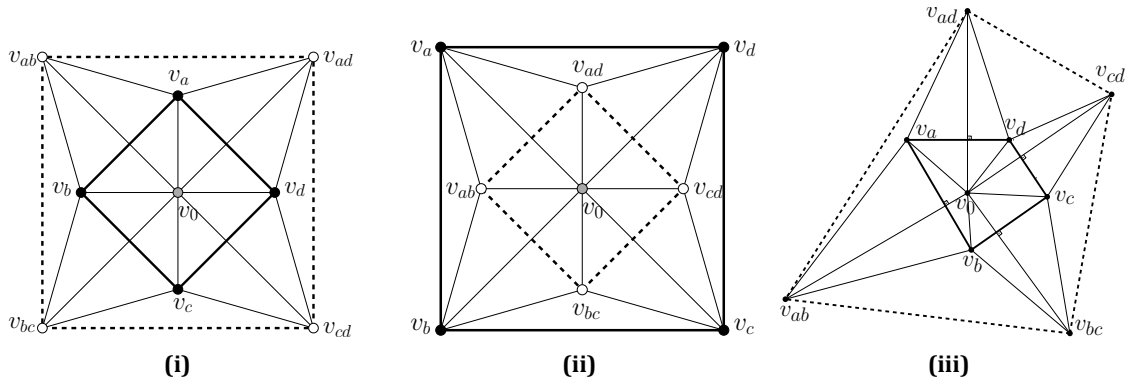
In this chapter, we focus on straight-line RAC drawings (or geometric RAC drawings) and demonstrate an infinite class of graphs with unique RAC combinatorial embedding. We employ members of this class in order to show that it is  $\mathcal{NP}$ -hard to decide whether a graph admits a straight-line RAC drawing.

### 3.1 Preliminary Notions and Notation

Before we proceed with the detailed description of our approach, we first introduce preliminary notions and notation that we heavily use in the remainder of this chapter. Given a drawing  $\Gamma(G)$  of a graph  $G$ , we denote by  $\ell_{u,v}$  the line passing through vertices  $u$  and  $v$ . By  $\ell'_{u,v}$ , we refer to the semi-line that emanates from vertex  $u$ , towards vertex  $v$ . Similarly, we denote by  $\ell_{u,v,w}$  ( $\ell'_{u,v,w}$ ) the line (semi-line) that passes through (emanates from) vertex  $u$  and is perpendicular to edge  $(v, w)$ .

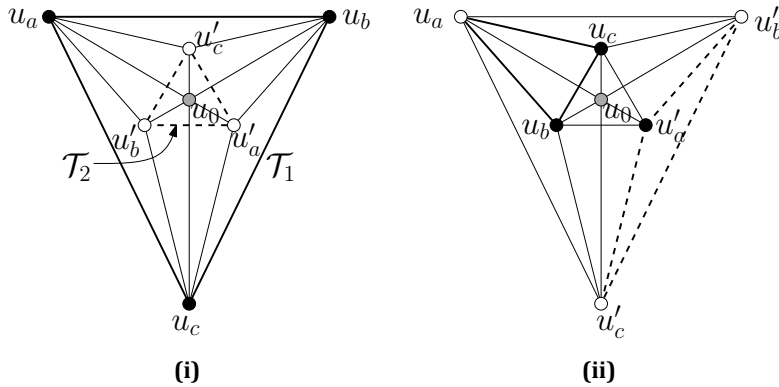
### 3.2 A Class of Graphs with Unique RAC Combinatorial Embedding

The main result of this chapter, i.e., the  $\mathcal{NP}$ -hardness of the straight line RAC drawing problem, employs a reduction from the well-known 3-SAT problem [GJ79]. Central in this reduction is a graph, referred to as *augmented square antiprism graph* (or *ASA graph* for short), which has the following property: “*The straight-line RAC drawings of the ASA graph define exactly two combinatorial embeddings*”. Figure 3.1 shows the ASA graph and its two combinatorial embeddings (the fact that these are the only two RAC combinatorial embeddings will be proved later in this section). Observe that the ASA graph consists of a “central” vertex  $v_0$ , which is incident to all vertices of the graph, and two quadrilaterals (refer to the dashed and bold drawn squares in Figure 3.1i), that are denoted by  $\mathcal{Q}_1$  and  $\mathcal{Q}_2$ . Removing the central vertex, the remaining graph corresponds to the skeleton of a square antiprism, and, it is commonly referred to as *square antiprism graph*.



**Figure 3.1:** (i),(ii) Two different RAC drawings of the ASA graph with different combinatorial embeddings. (i),(iii) Two different RAC drawings of the ASA graph with the same combinatorial embedding.

If, in the ASA graph, we replace the two quadrilaterals with two triangles, then the resulting graph is the *augmented triangular antiprism graph* (or *ATA graph* for short). Didimo, Eades and Liotta [DEL11], who showed that any  $n$ -vertex graph which admits a RAC-drawing can have at most  $4n - 10$  edges, used the augmented triangular antiprism graph, as an example of a graph that achieves the bound of  $4n - 10$  edges (see Figure 1.c in [DEL11]). In contrast to the augmented triangular antiprism graph, the augmented square antiprism graph does not achieve the upper bound on the number of edges. In general, the class of the *augmented  $k$ -gon antiprism graphs*,  $k \geq 3$ , is a class of non-planar graphs, that all admit RAC drawings. Recall that any planar  $n$ -vertex graph has at most  $3n - 6$  edges, and since an augmented  $k$ -gon antiprism graph has  $2k + 1$  vertices and  $6k$  edges, it is not planar for the entire class of these graphs.



**Figure 3.2:** (i)-(ii) Two different RAC drawings of the augmented triangle antiprism graph with different combinatorial embeddings.

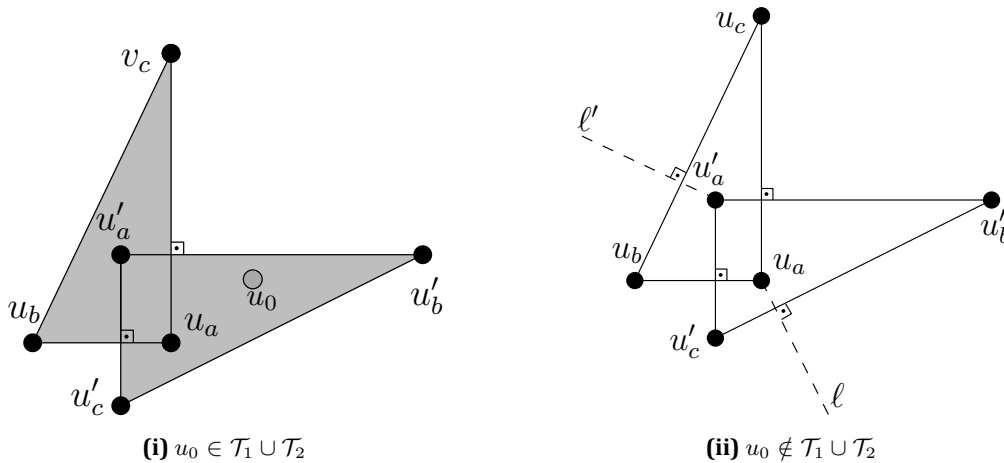
Observe that in the triangle antiprism graph of Figure 3.2i, (i)  $u_a$  and  $u'_a$ , (ii)  $u_b$  and  $u'_b$ , and (iii)  $u_c$  and  $u'_c$  share the same neighbors. Let  $\mathcal{T}_1$  (resp.  $\mathcal{T}_2$ ) be the triangle induced by  $u_a, u_b$  and  $u_c$  (resp.  $u'_a, u'_b$  and  $u'_c$ ). Our intention is to prove that the straight-lint RAC drawings of the augmented triangle antiprism graph define exactly eight different combinatorial embeddings, which are implied by the drawing of Figure 3.2i by mutually exchanging the positions of (i)  $u_a$  and  $u'_a$ , (ii)  $u_b$  and  $u'_b$ , and (iii)  $u_c$  and  $u'_c$ . In the following, we investigate some properties of the RAC drawings of the augmented triangle antiprism graph.

**Property 3.2.1.** *In any RAC drawing of the augmented triangle antiprism graph, the external face is bounded by three edges.*

*Proof.* Directly follows from RAC-Property 4 and the fact that the augmented triangle antiprism graph is maximally dense.  $\square$

**Property 3.2.2.** *In any RAC drawing of the augmented triangle antiprism graph, triangles  $\mathcal{T}_1$  and  $\mathcal{T}_2$  do not cross.*

*Proof.* Assume to the contrary that in a RAC drawing of the ATA graph, triangles  $\mathcal{T}_1$  and  $\mathcal{T}_2$  cross. Their crossing should involve four edges; two of each triangle (see Figure 3.3). As a consequence one vertex of  $\mathcal{T}_1$  should lie inside  $\mathcal{T}_2$  and one vertex of  $\mathcal{T}_2$  should lie inside  $\mathcal{T}_1$ . Without loss of generality, let this pair of vertices be  $u_a$  and  $u'_a$  (all other pairs are handled similarly). Then, edges  $(u_a, u_b), (u_a, u_c), (u'_a, u'_b)$  and  $(u'_a, u'_c)$  are the ones that are involved in the crossings. If vertex  $u_0$  lies in  $\mathcal{T}_1 \cup \mathcal{T}_2$  (refer to the gray-colored region of Figure 3.3i), then the external face is bounded by four vertices which is not permitted due to Property 3.2.1.



**Figure 3.3:** Triangles  $\mathcal{T}_1$  and  $\mathcal{T}_2$  cross.

Assume now that vertex  $u_0$  lies in  $(\mathcal{T}_1 \cup \mathcal{T}_2)^c$ . This case is depicted in Figure 3.3ii. Since vertex  $u_0$  is connected to  $u_a$ , it should lie along the selimine, say  $\ell$ , that emanates from vertex

$u_a$  and is perpendicular to edge  $(u'_b, u'_c)$ . On the other hand, vertex  $u_0$  should also be connected to  $u'_a$ . Hence,  $u_0$  should lie along the semiline, say  $\ell'$ , that emanates from vertex  $u_a$  and is perpendicular to edge  $(u_b, u_c)$ . Thus, semilines  $\ell$  and  $\ell'$  should cross, which is impossible due to the construction.  $\square$

**Property 3.2.3.** *There does not exist a RAC drawing of the augmented triangle antiprism graph in which the 3-cycle incident to the external face contains a vertex of  $\mathcal{T}_1$  and a vertex of  $\mathcal{T}_2$  that have the same neighbors.*

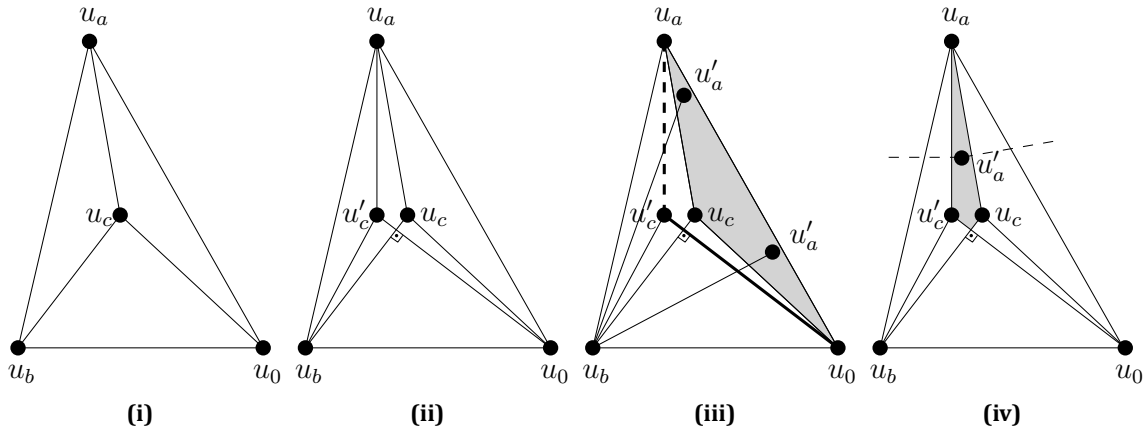
*Proof.* For the sake of contradiction, assume that  $u_a \in \mathcal{T}_1$  and  $u'_a \in \mathcal{T}_2$  (which have the same neighbors) are on the 3-cycle incident to the external face. Now observe that edge  $(u_a, u'_a)$  is not present in the augmented triangle antiprism, which leads to a contradiction Property 3.2.1. The cases where either  $u_b$  and  $u'_b$  or  $u_c$  and  $u'_c$  are on the 3-cycle incident to the external face are treated symmetrically.  $\square$

**Property 3.2.4.** *There does not exist a RAC drawing of the augmented triangle antiprism graph in which the 3-cycle incident to the external face contains vertex  $u_0$ .*

*Proof.* Assume to the contrary that in a RAC drawing of the augmented triangle antiprism graph vertices  $u_a$  and  $u_b$ <sup>1</sup> (which do not have the same neighbors) are on the 3-cycle incident to the external face together with vertex  $u_0$ . Since  $u_a, u_b$  and  $u_0$  form the external face, vertex  $u_c$  (which is incident to  $u_a, u_b$  and  $u_0$ ) should be an interior vertex and splits the interior of the 3-cycle into three triangular regions  $\Delta u_a u_b u_c$ ,  $\Delta u_a u_c u_0$  and  $\Delta u_b u_c u_0$  (see Figure 3.4i). We next consider vertex  $u'_c$ , which has the same neighborhood as  $u_c$ . Vertex  $u'_c$  may lie in the interior of one of  $\Delta u_a u_b u_c$ ,  $\Delta u_a u_c u_0$  or  $\Delta u_b u_c u_0$ . We prove in detail the case where  $u'_c$  lies in  $\Delta u_a u_b u_c$ , as Figure 3.4ii illustrates. The case where  $u'_c$  lies in the interior of  $\Delta u_a u_c u_0$  is symmetric, since  $u_c$  and  $u'_c$  have the same neighborhood. The case where  $u'_c$  lies in the interior of  $\Delta u_b u_c u_0$  can be proved similarly. Now observe that edge  $(u_0, u'_c)$  should cross either edge  $(u_b, u_c)$  or edge  $(u_a, u_c)$ . We prove the former case. The latter one can also be proved following similar arguments.

The desired contradiction will be implied by vertex  $u'_a$ , for which we will prove that there exists no feasible placement. Recall that vertex  $u'_a$  is incident to  $u_b, u'_b, u_c, u'_c$  and  $u_0$ . Initially, we prove that vertex  $u'_a$  cannot lie in a triangle formed by two vertices of the external face and either vertex  $u_c$  or vertex  $u'_c$ , i.e, a triangular region incident to the external face. To see this, assume to the contrary that  $u'_a$  lies in the interior of such a triangle, say at triangle  $\Delta u_0 u_a u_c$  (refer to the gray colored region of Figure 3.4iii). In this case, edge  $(u'_a, u_b)$  cannot cross edge  $(u'_c, u_0)$  (refer to the bold solid drawn edge of Figure 3.4iii), since that would lead to a situation where  $(u'_a, u_b)$  and  $(u_b, u_c)$  form a fan anchored at  $u_b$  crossed by  $(u_0, u'_c)$ , which is not

<sup>1</sup>If we use  $u'_b$  instead of  $u_b$  the proof is identical.



**Figure 3.4:** Vertices  $u_a, u_b$  and  $u_0$  form the external face, vertex  $u_c$  is an interior vertex (defining three triangular regions  $\triangle u_a u_b u_c, \triangle u_b u_c u_0$  and  $\triangle u_a u_c u_0$ ) and vertex  $u'_c$  is in  $\triangle u_a u_b u_c$ .

permitted due to RAC-Property 2. In addition,  $(u'_a, u_b)$  cannot cross edge  $(u_a, u'_c)$  (refer to the bold dashed drawn edge of Figure 3.4iii, since that would lead to a situation where  $(u_a, u'_c)$  and  $(u_a, u_c)$  form a fan anchored at  $u_a$  crossed by  $(u'_a, u_b)$ , which is not permitted due to RAC-Property 2. Hence,  $u'_a$  cannot be connected to  $u_b$ . Similarly, one can prove that  $u'_a$  cannot lie in  $\triangle u_a u_b u'_c, \triangle u_b u_c u_0$  and  $\triangle u_b u'_c u_0$ . Note that, the last two cases are easy to be proved based on RAC-Property 2 and the fact that edges  $(u_b, u_c)$  and  $(u_0, u'_c)$  are already involved in a right-angle crossing. Having excluded the cases where  $u'_a$  lies in a triangular region incident to the external face, the only case that remains to consider is the one where vertex  $u'_a$  lies in the region formed by vertices  $u_a, u_c, u'_c$  and the intersection of  $(u_0, u'_c)$  and  $(u_b, u_c)$  (refer to the gray colored region of Figure 3.4iv). In this case, edge  $(u'_a, u_b)$  cannot cross  $(u'_c, u_0)$  due to RAC-Property 2. For the same reason,  $(u'_a, u_0)$  cannot cross  $(u_b, u_c)$ . Hence, in order to realize the connections of  $u'_a$  to  $u_b$  and  $u_0$ , edges  $(u_a, u'_c)$  and  $(u_a, u_c)$  should be perpendicularly crossed by edges  $(u'_a, u_b)$  and  $(u'_a, u_0)$ , respectively, which cannot be done since the outface is a triangle. □

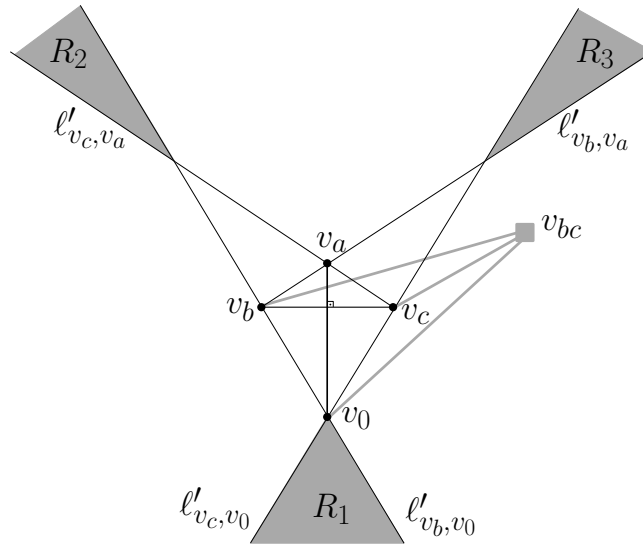
**Lemma 3.2.1.** *The straight-line RAC drawings of the augmented triangle antiprism graph define exactly eight combinatorial embeddings.*

*Proof.* Recall that (i)  $u_a$  and  $u'_a$ , (ii)  $u_b$  and  $u'_b$ , and, (iii)  $u_c$  and  $u'_c$  share the same neighbors. This combined with Properties 3.2.1, 3.2.2, 3.2.3 and 3.2.4 imply the eight different combinatorial embeddings of the augmented triangle antiprism graph (i.e., obtained by the drawing of Figure 3.2i by mutually exchanging the positions of (i)  $u_a$  and  $u'_a$ , (ii)  $u_b$  and  $u'_b$ , and, (iii)  $u_c$  and  $u'_c$ ). □

In the following, we prove that the straight-line RAC drawings of the ASA graph define exactly two combinatorial embeddings.

**Lemma 3.2.2.** *There does not exist a RAC drawing of the ASA graph in which (i) the central vertex  $v_0$  lies on the exterior of quadrilateral  $Q_i$  and (ii) an edge connecting  $v_0$  to a vertex of  $Q_i$  crosses another edge of  $Q_i$ ,  $i = 1, 2$ .*

*Proof.* Let  $Q$  be one of quadrilaterals  $Q_i$ ,  $i = 1, 2$  and let  $v_a, v_b, v_c$  and  $v_d$  be its vertices, consecutive along quadrilateral  $Q$  (refer to Figure 3.5). Assume, to the contrary, that vertex  $v_0$  lies on the exterior of quadrilateral  $Q$  and there exists an edge, say  $(v_0, v_a)$ , that emanates from vertex  $v_0$  towards vertex  $v_a$  of quadrilateral  $Q$ , such that it crosses an edge, say  $(v_b, v_c)$ <sup>2</sup>, of  $Q$  (see Figure 3.5). On the ASA graph, vertices  $v_b$  and  $v_c$  have the following properties: (a) they are both connected to vertex  $v_0$ , and, (b) they have a common neighbor  $v_{bc}$ , which is incident to vertex  $v_0$  and  $v_{cd} \notin Q$  (see Figure 3.1).

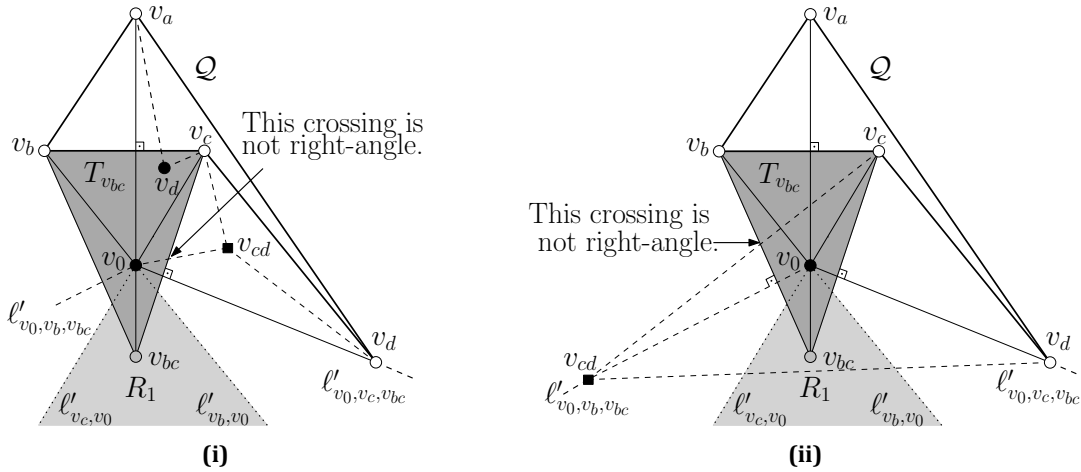


**Figure 3.5:** Configuration used in proof of Lemma 3.2.2: Vertex  $v_{bc}$  should lie in the interior of one of the regions  $R_1$ ,  $R_2$  and  $R_3$ .

Observe that if vertex  $v_{bc}$  lies in the non-shaded regions of Figure 3.5, then at least one of the edges incident to  $v_{bc}$  crosses either  $(v_0, v_a)$  or  $(v_b, v_c)$ , which are already involved in a right-angle crossing. This leads to a situation where three edges mutually cross, which, by RAC-Property 1 is not permitted. Hence, vertex  $v_{bc}$  should lie in the interior of the gray-shaded regions  $R_1$ ,  $R_2$  or  $R_3$  in Figure 3.5. In the following, we consider each of these cases separately. Depending on the position of  $v_a, v_b, v_c$ , and,  $v_0$ ,  $R_2$  or  $R_3$  or  $R_2 \cup R_3$  may be empty.

<sup>2</sup>The case where it crosses edge  $(v_c, v_d)$  is symmetric.

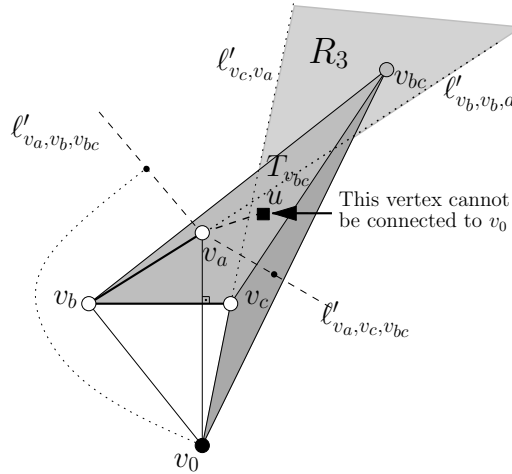
**Case i:** Vertex  $v_{bc}$  lies in the interior of  $R_1$ . This case is depicted in Figure 3.6. Let  $T_{v_{bc}}$  be the region formed by vertices  $v_{bc}, v_b$  and  $v_c$  (i.e., the dark-gray shaded region of Figure 3.6). Vertex  $v_d$ , which has to be connected to vertices  $v_a$  and  $v_c$ , and the central vertex  $v_0$ , cannot lie within  $T_{v_{bc}}$ , since  $(v_a, v_0)$  and  $(v_a, v_d)$  form a fan anchored at  $v_a$  and crossed by  $(v_b, v_c)$ , which by RAC-Property 2 is not permitted. Since vertex  $v_d$  has to be connected to vertex  $v_0$ , it has to lie either on semi-line  $\ell'_{v_0, v_c, v_{bc}}$  or on semi-line  $\ell'_{v_0, v_b, v_{bc}}$ . We consider only the former case. The latter one is handled symmetrically. However, under this restriction, the common neighbor  $v_{cd}$  of vertices  $v_c$  and  $v_d$  cannot be connected to vertex  $v_0$ , since edge  $(v_0, v_{cd})$  should be perpendicular to one of the edges of  $T_{v_{bc}}$ . To see this, observe that if edge  $(v_0, v_{cd})$  is perpendicular to  $(v_c, v_{bc})$ , then  $(v_0, v_{cd})$  and  $(v_0, v_d)$  form a fan anchored at  $v_0$  and crossed by  $(v_c, v_{bc})$ , which by RAC-Property 2 is not permitted (see Figure 3.6i). Consider now the case where edge  $(v_0, v_{cd})$  is perpendicular to edge  $(v_b, v_{bc})$  (and lies on  $\ell'_{v_0, v_b, v_{bc}}$ ; see Figure 3.6ii). Since angle  $\widehat{v_c v_b v_{bc}}$  is acute, edge  $(v_b, v_{bc})$  should be crossed by both edges  $(v_0, v_{cd})$  and  $(v_c, v_{cd})$ , which form a fan anchored at  $v_{cd}$ . This leads to a contradiction due to RAC-Property 2.



**Figure 3.6:** Configurations used in proof of Lemma 3.2.2: Vertex  $v_{bc}$  lies in the interior of  $R_1$ .

**Case ii:** Vertex  $v_{bc}$  lies in the interior of either  $R_2$  or  $R_3$ . Assume, without loss of generality, that vertex  $v_{bc}$  lies in the interior of  $R_3$ . This case is depicted in Figure 3.7. Let  $u$  be a vertex of the ASA graph (distinct from  $v_a, v_b, v_c$  and  $v_0$ ) and assume that  $u$  lies in the interior of the triangle  $T_{v_{bc}}$  formed by vertices  $v_b, v_c$  and  $v_{bc}$ . Vertex  $u$  has to be connected to the central vertex  $v_0$ . Edge  $(v_0, u)$  should not be involved in crossings with neither edge  $(v_b, v_c)$ , since  $(v_0, u)$  and  $(v_0, v_a)$  would form a fan anchored at  $v_0$  and crossed by  $(v_b, v_c)$ , nor edge  $(v_c, v_{bc})$ , since angle  $\widehat{v_b v_c v_{bc}}$  is smaller than  $180^\circ$ . Therefore, triangle  $T_{v_{bc}}$  cannot accommodate any other vertex (except  $v_a$ ). Now observe that each vertex of quadrilateral  $Q$  has degree five and there do not exist three vertices of quadrilateral  $Q$ , that have a common neighbor (see Figure 3.1). These

properties trivially hold for vertex  $v_a$ , since  $v_a \in \mathcal{Q}$ . Based on the above properties, each neighbor of vertex  $v_a$  can lie either in the interior of the dark-gray region of Figure 3.7, or, on the external face of the already constructed drawing (along the dashed semi-lines  $\ell'_{v_a, v_c, v_{bc}}$  and  $\ell'_{v_a, v_b, v_{bc}}$  of Figure 3.7, respectively). This implies that we can place only four vertices out of those incident to vertex  $v_a$ , i.e., one of them should lie in the interior of  $T_{v_{bc}}$  and thus, it cannot be connected to vertex  $v_0$ .



**Figure 3.7:** Configuration used in proof of Lemma 3.2.2: Vertex  $v_{bc}$  lies in the interior of  $R_3$ .

From the above case analysis, it follows that central vertex  $v_0$  cannot lie on the exterior of quadrilateral  $\mathcal{Q}$ , so that an edge connecting  $v_0$  to a vertex of  $\mathcal{Q}$  crosses another edge of  $\mathcal{Q}$ .  $\square$

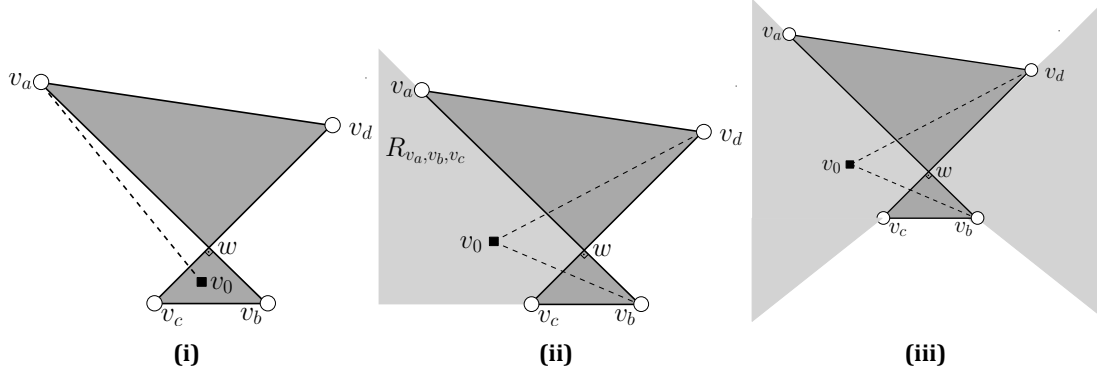
**Lemma 3.2.3.** *In any RAC drawing of the ASA graph, quadrilateral  $\mathcal{Q}_i$  is drawn planar, for each  $i = 1, 2$ .*

*Proof.* Let  $\mathcal{Q}$  be one of quadrilaterals  $\mathcal{Q}_i$ ,  $i = 1, 2$ , and let, as in the proof of the previous lemma,  $v_a, v_b, v_c$  and  $v_d$  be its vertices, consecutive along  $\mathcal{Q}$ . Note that, it is not feasible a non-planar, straight-line RAC drawing of a quadrilateral to contain more than one right-angle crossing (the drawing of  $\mathcal{Q}$  forms two orthogonal triangles). Assume to the contrary that in a RAC drawing of the ASA graph, quadrilateral  $\mathcal{Q}$  is not drawn planar, and say that edges  $(v_a, v_b)$  and  $(v_c, v_d)$  form a right-angle crossing. This case is illustrated in Figure 3.8i. In the following, we will consider the cases where the central vertex  $v_0$  lies either in one of the orthogonal triangles or in the area outside them. In both cases, we will reach a contradiction.

**Case i:** *Vertex  $v_0$  lies in the interior of one of the two triangles.* Assume, without loss of generality, that vertex  $v_0$  (which is incident to all vertices of quadrilateral  $\mathcal{Q}$ ) lies in the interior of the triangle formed by vertices  $v_b, v_c$  and the intersection point, say  $w$ , of edges  $(v_a, v_b)$  and



$(v_c, v_d)$ , as in Figure 3.8i. In this case, edges  $(v_a, v_0)$  and  $(v_a, v_b)$  form a fan anchored at  $v_a$ , which is crossed by  $(v_c, v_d)$ . This is not possible due to RAC-Property 2.



**Figure 3.8:** Configurations used in proof of Lemma 3.2.3: Quadrilateral  $\mathcal{Q}$  is not drawn planar. (i)  $v_0$  cannot lie within one of the two triangles, (ii)  $v_0$  cannot lie within  $R_{v_a, v_b, v_c}$ , and, (iii)  $v_0$  cannot lie within the light-gray open areas.

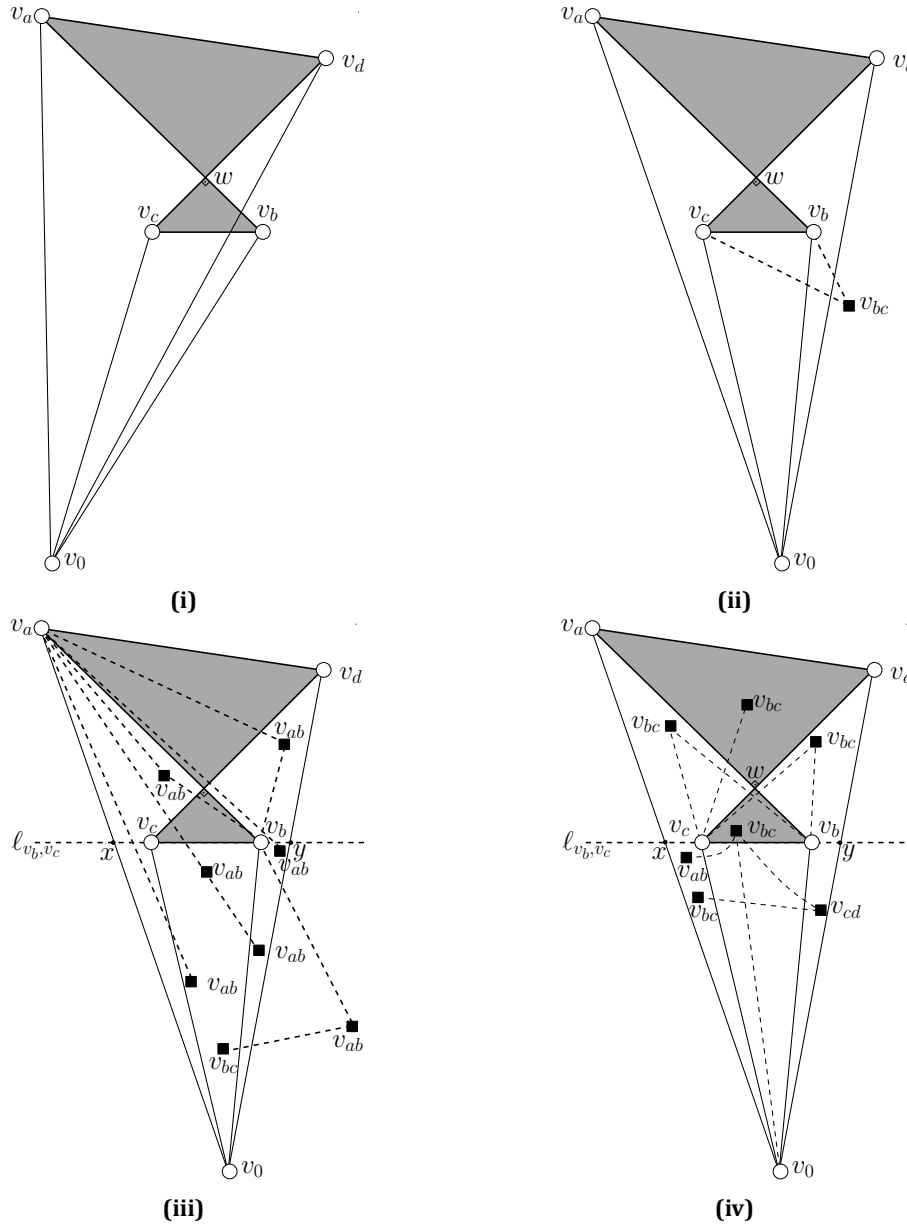
**Case ii:** Vertex  $v_0$  lies outside the two triangles. Without loss of generality, we assume that edge  $(v_b, v_c)$  is horizontal and that edge  $(v_a, v_d)$  is drawn above it. See Figure 3.8i. The two triangles are drawn dark-shaded.

Let  $R_{v_a, v_b, v_c}$  denote the light-gray shaded open area defined by angle  $\widehat{v_a v_b v_c}$  which excludes the area of triangle  $\triangle w v_b v_c$ . See Figure 3.8ii. We first show that vertex  $v_0$  cannot lie in area  $R_{v_a, v_b, v_c}$ . To see that, simply observe that if  $v_0$  lied within area  $R_{v_a, v_b, v_c}$ , then edges  $(v_0, v_b)$  and  $(v_0, v_d)$  would form a fan anchored at  $v_0$  and crossed by  $(v_c, v_d)$ , which is not possible due to RAC-Property 2. In the same way, we can define the open areas  $R_{v_b, v_c, v_d}$ ,  $R_{v_c, v_d, v_a}$  and  $R_{v_d, v_a, v_b}$  and show that vertex  $v_0$  cannot lie in any of them. Thus, it remains to examine the case where vertex  $v_0$  lies either in the white area below edge  $(v_b, v_c)$  or in the white area above edge  $(v_a, v_d)$ . See Figure 3.8iii. We only consider the former case, since the latter can be similarly handled by assuming that  $(v_a, v_d)$  is the horizontal edge. So, assume that vertex  $v_0$  lies in the white area of Figure 3.8iii below edge  $(v_b, v_c)$ . Given this assumption, we can state the following propositions regarding the placement of the remaining vertices in a RAC drawing of the ASA graph.

**Proposition 1.** Edge  $(v_b, v_c)$  should be inside triangle  $\triangle v_a v_0 v_d$ .

*Proof.* Refer to Figure 3.9i. Assume to the contrary that edge  $(v_b, v_c)$  is not inside  $\triangle v_a v_0 v_d$ .

Then, one of the triangle's edges incident at  $v_0$  would cross a fan formed by edges of  $\mathcal{Q}$  anchored either at  $v_b$  or  $v_c$ , which is a contradiction due to RAC-Property 2.  $\square$



**Figure 3.9:** Configurations used in proof of Lemma 3.2.3 (Propositions 1-5): (i) Edge  $(v_b, v_c)$  does not lie inside  $\triangle v_a v_0 v_d$ . (ii) Vertex  $v_{bc}$  does not lie inside  $\triangle v_a v_0 v_d$ , (iii) Potential placements of vertex  $v_{ab}$ , (iv) Potential placements of vertex  $v_{bc}$

**Proposition 2.** Vertex  $v_{bc}$  lies inside triangle  $\triangle v_a v_0 v_d$ .

*Proof.* Refer to Figure 3.9ii. Vertex  $v_{bc}$  is connected to both  $v_b$  and  $v_c$ , which by Proposition 1 are inside triangle  $\triangle v_a v_0 v_d$ . By RAC-Property 3, it follows that vertex  $v_{bc}$  must also lie inside triangle  $\triangle v_a v_0 v_d$ .  $\square$

**Proposition 3.** *Vertex  $v_{ab}$  lies inside triangle  $\triangle v_0 v_c x$ , where  $x$  is the intersection point of line  $\ell_{v_b, v_c}$  with edge  $(v_0, v_a)$ .*

*Proof.* Refer to Figure 3.9iii. In order to prove this proposition, we will prove that vertex  $v_{ab}$  (i) cannot lie outside triangle  $\triangle v_a v_0 v_d$ , (ii) must lie below line  $\ell_{v_b, v_c}$ , and, cannot lie in either (iii) triangle  $\triangle v_0 v_b y$ , or (iv) triangle  $\triangle v_0 v_b v_c$ .

- (i) *Vertex  $v_{ab}$  lies inside triangle  $\triangle v_a v_0 v_d$ :* Vertex  $v_{ab}$  is connected to  $v_b$  and  $v_{bc}$ , which by Propositions 1 and 2, respectively, lie within  $\triangle v_a v_0 v_d$ . This ensures that  $v_{ab}$  lies inside triangle  $\triangle v_a v_0 v_d$ , as well; otherwise RAC-Property 3 is violated.
- (ii) *Vertex  $v_{ab}$  must lie below line  $\ell_{v_b, v_c}$ :* Let  $y$  be the intersection point of line  $\ell_{v_b, v_c}$  with edge  $(v_0, v_d)$ . Assume to the contrary that  $v_{ab}$  lies above line  $\ell_{v_b, v_c}$ . We consider the following cases. First, assume that  $v_{ab}$  lies within quadrilateral  $v_a x v_c v_d$ . Then, edges  $(v_a, v_b)$  and  $(v_b, v_{ab})$  form a fan anchored at  $v_b$  which is crossed by  $(v_c, v_d)$ . In the case where  $v_{ab}$  lies within triangle  $\triangle v_c y v_d$ , then edges  $(v_a, v_b)$  and  $(v_a, v_{ab})$  form a fan anchored at  $v_a$  which is crossed by  $(v_c, v_d)$ . Since both cases lead to a contradiction due to RAC-Property 2, we conclude that  $v_{ab}$  must lie below line  $\ell_{v_b, v_c}$ .
- (iii) *Vertex  $v_{ab}$  cannot lie in triangle  $\triangle v_0 v_b y$ :* This is due to the fact that  $v_{ab}$  is connected with  $v_a$ . If edge  $(v_{ab}, v_a)$  passes from the “left” of  $v_b$  it must enter and exit triangle  $\triangle v_0 v_b v_c$ , forming three mutually crossing edges. If edge  $(v_{ab}, v_a)$  passes from the “right” of  $v_b$ , then  $(v_a, v_{ab})$  and  $(v_a, v_b)$  form a fan anchored at  $v_a$  and crossed by  $(v_c, v_d)$ . Since both cases lead to a contradiction (due to RAC-Properties 1 and 2, respectively), we conclude that  $v_{ab}$  cannot lie in triangle  $\triangle v_0 v_b y$ .
- (iv) *Vertex  $v_{ab}$  cannot lie in triangle  $\triangle v_0 v_b v_c$ :* This is, again, due to the fact that  $v_{ab}$  is connected with  $v_a$ . If edge  $(v_a, v_{ab})$  passes from the “right” of  $v_c$ , then it must enter and exit triangle  $\triangle v_b v_c w$ , forming three mutually crossing edges. However, this is not permitted due to RAC-Property 1. If edge  $(v_a, v_{ab})$  passes from the “left” of  $v_c$ , then it must cross at right angle edge  $(v_c, v_0)$ . But, since edge  $(v_a, v_b)$  is also perpendicular to edge  $(v_c, v_d)$  then quadrilateral  $v_a v_d v_c v_0$  must be convex, and thus,  $v_0$  must be above line  $\ell_{v_b, v_c}$ , a clear contradiction.  $\square$

**Proposition 4.** *Vertex  $v_{cd}$  lies inside triangle  $\triangle v_0 v_b y$ .*

*Proof.* Following symmetric arguments as in the proof of Proposition 3.  $\square$

**Proposition 5.** *Vertex  $v_{bc}$  lies within triangle  $\Delta v_0 y v_c$ .*

*Proof.* Recall that by Proposition 2, vertex  $v_{bc}$  lies within triangle  $\Delta v_0 v_a v_d$ . In order to establish that vertex  $v_{bc}$  lies within triangle  $\Delta v_0 y v_c$ , we will lead to a contradiction the cases where vertex  $v_{bc}$  lies within quadrilaterals (i)  $v_a w v_c x$  and (ii)  $v_d w v_b y$ , and within triangles (iii)  $\Delta v_a w v_d$ , (iv)  $\Delta v_b w v_c$  and (v)  $\Delta x v_c v_0$ . Refer to Figure 3.9iv.

- (i) *Vertex  $v_{bc}$  does not lie within quadrilateral  $v_a w v_c x$ :* If it does, then edges  $(v_a, v_b)$  and  $(v_b, v_{bc})$  form a fan anchored at  $v_b$  which is crossed by  $(v_c, v_d)$ .
- (ii) *Vertex  $v_{bc}$  does not lie within quadrilateral  $v_d w v_b y$ :* If it does, then edges  $(v_c, v_d)$  and  $(v_c, v_{bc})$  form a fan anchored at  $v_c$  which is crossed by  $(v_a, v_b)$ .
- (iii) *Vertex  $v_{bc}$  does not lie within triangle  $\Delta v_a w v_d$ :* If it does, then edges  $(v_c, v_d)$  and  $(v_c, v_{bc})$  form a fan anchored at  $v_c$  which is crossed by  $(v_a, v_b)$ .
- (iv) *Vertex  $v_{bc}$  does not lie within triangle  $\Delta v_b w v_c$ :* If it does, then it has three neighbors, namely  $v_0$ ,  $v_{ab}$  and  $v_{cd}$ , outside triangle  $\Delta v_b w v_c$ . Thus, it must be connected to them by edges that exit different sides of the triangle. Given that  $v_0$  lies below line  $\ell_{v_b, v_c}$ , then at least one of  $v_{ab}$  and  $v_{cd}$  must be above line  $\ell_{v_b, v_c}$ . A clear contradiction.
- (v) *Vertex  $v_{bc}$  does not lie within triangle  $\Delta x v_c v_0$ :* Recall that by Proposition 4, vertex  $v_{cd}$  lies inside triangle  $\Delta v_0 v_b y$ . However, vertex  $v_{bc}$  is connected to vertex  $v_{cd}$ . This implies that, if vertex  $v_{bc}$  lies within triangle  $\Delta x v_c v_0$ , then  $(v_{bc}, v_{cd})$  must enter end exit triangle  $\Delta v_0 v_b v_c$ , forming three mutually crossing edges.

□

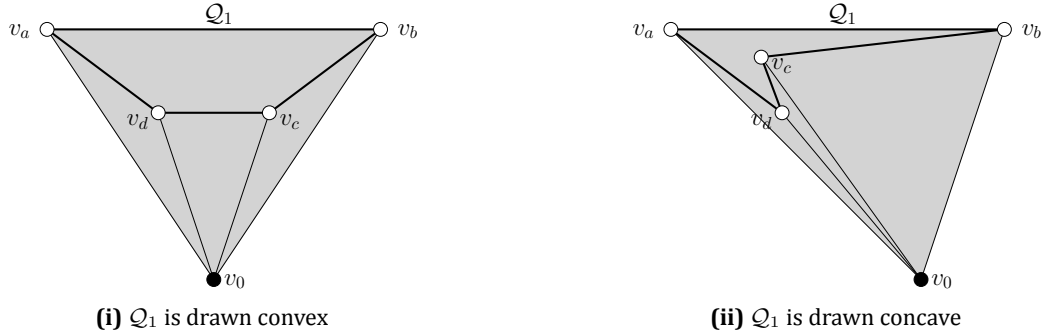
Since vertex  $v_{ab}$  lies inside triangle  $\Delta v_0 v_c x$  (Proposition 3) and vertex  $v_{bc}$  lies within triangle  $\Delta v_0 y v_c$  (Proposition 5), edges  $(v_{ab}, v_{bc})$  and  $(v_{ab}, v_b)$  form a fan anchored at  $v_{ab}$ , which is crossed by  $(v_0, v_c)$ , which violates RAC-Property 2. Thus, by assuming a legal RAC drawing of the ASA graph where vertex  $v_0$  is outside the two triangles, we concluded that it is not possible to find a legal placement for  $v_{bc}$ . This completes the proof of Case ii of this lemma.

□

**Lemma 3.2.4.** *In any RAC drawing of the ASA graph, the central vertex  $v_0$  lies in the interior of quadrilateral  $Q_i$ ,  $i = 1, 2$ .*

*Proof.* From Lemma 3.2.3, it follows that quadrilateral  $Q_i$  should be drawn planar, for each  $i = 1, 2$ . In order to prove this lemma, we assume to the contrary that central vertex  $v_0$  lies on the exterior of one of the two quadrilaterals, say w.l.o.g., on the exterior of quadrilateral  $Q_1$ . Let  $v_a, v_b, v_c$  and  $v_d$  be  $Q_1$ 's vertices, consecutive along quadrilateral  $Q_1$ . Then, by

Lemma 3.2.2, vertex  $v_0$  cannot contribute additional crossings on quadrilateral  $\mathcal{Q}_1$ . This suggests that the drawing of the graph induced by quadrilateral  $\mathcal{Q}_1$  and vertex  $v_0$  will be planar and resemble the ones depicted in Figure 3.10. We denote by  $T_{\mathcal{Q}_1}$  the triangle formed by vertex  $v_0$  and the two vertices, which are on the convex hull of  $\mathcal{Q}_1 \cup v_0$  (refer to the gray-shaded triangles of Figure 3.10).



**Figure 3.10:** Configurations used in proof of Lemma 3.2.4: Different drawings of the graph induced by quadrilateral  $\mathcal{Q}_1$  and vertex  $v_0$ .

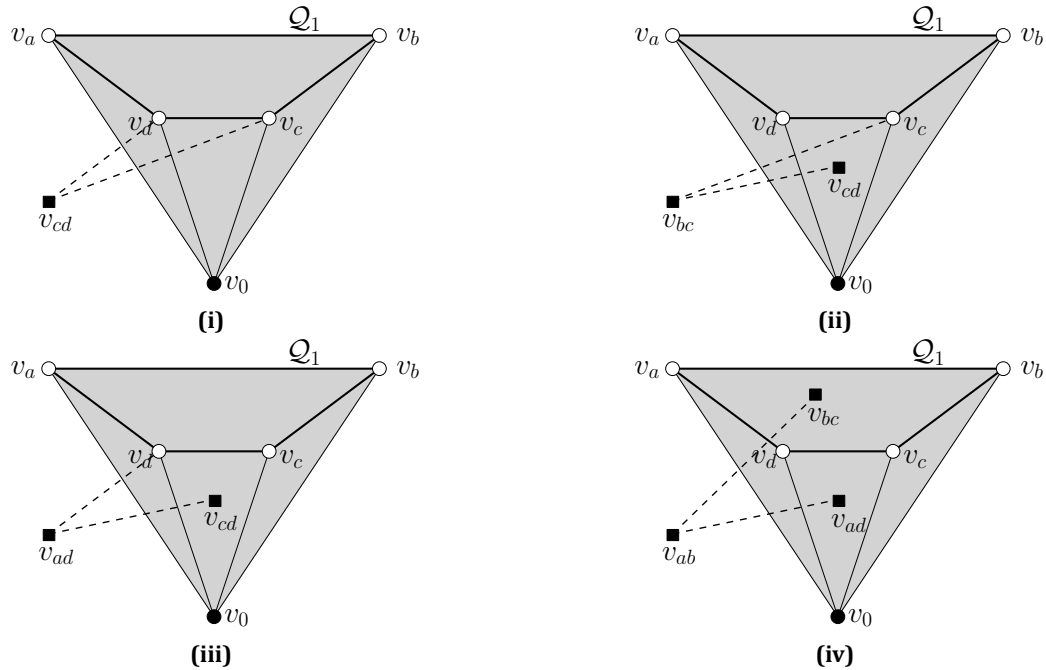
**Proposition 6.** *No vertex of  $\mathcal{Q}_2$  lies outside  $T_{\mathcal{Q}_1}$ .*

*Proof.* Refer to Figure 3.10 and assume w.l.o.g that  $T_{\mathcal{Q}_1}$  is defined by vertices  $v_0, v_a$  and  $v_b$ , i.e.,  $v_a$  and  $v_b$  are on the convex hull of  $\mathcal{Q}_1 \cup v_0$ . In the following, we prove that each of the vertices of  $\mathcal{Q}_2$  should lie inside  $T_{\mathcal{Q}_1}$ .

- (i) *Vertex  $v_{cd}$  lies within triangle  $T_{\mathcal{Q}_1}$ :* If not, RAC-Property 3 is violated (see Figure 3.11i);  $v_{cd}$  is connected to  $v_c$  and  $v_d$ , which both lie inside  $T_{\mathcal{Q}_1}$ .
- (ii) *Vertex  $v_{bc}$  lies within triangle  $T_{\mathcal{Q}_1}$ :* If not, RAC-Property 3 is violated (see Figure 3.11ii);  $v_{bc}$  is connected to  $v_c$  and  $v_{cd}$ , which both lie inside  $T_{\mathcal{Q}_1}$ .
- (iii) *Vertex  $v_{ad}$  lies within triangle  $T_{\mathcal{Q}_1}$ :* If not, RAC-Property 3 is violated (see Figure 3.11iii);  $v_{ad}$  is connected to  $v_d$  and  $v_{cd}$ , which both lie inside  $T_{\mathcal{Q}_1}$ .
- (iv) *Vertex  $v_{ab}$  lies within triangle  $T_{\mathcal{Q}_1}$ :* If not, RAC-Property 3 is violated (see Figure 3.11iv);  $v_{ab}$  is connected to  $v_{ad}$  and  $v_{bc}$ , which both lie inside  $T_{\mathcal{Q}_1}$ .

□

Since  $\mathcal{Q}_2$  lies inside  $T_{\mathcal{Q}_1}$ , vertex  $v_0$  is external to  $\mathcal{Q}_2$  as well. Therefore, in a way similar to that of Proposition 6, one can also prove that  $\mathcal{Q}_1$  lies inside  $T_{\mathcal{Q}_2}$ , where  $T_{\mathcal{Q}_2}$  is defined by



**Figure 3.11:** Configurations used in proof of Lemma 3.2.4 (Proposition 6): All vertices of  $Q_2$  should lie inside  $T_{Q_1}$

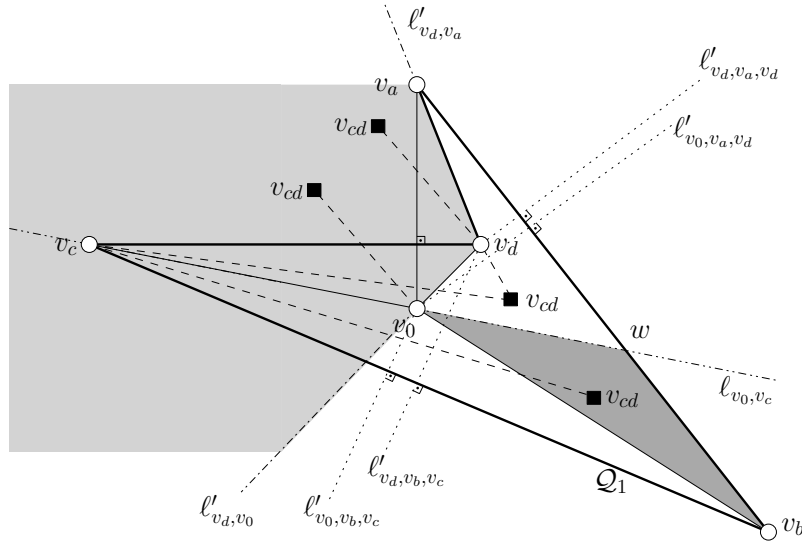
vertex  $v_0$  and the two vertices, which are on the convex hull of  $Q_2 \cup v_0$ . However, this leads to a contradiction, since two triangles (i.e.,  $T_{Q_1}$  and  $T_{Q_2}$ ) cannot be nested each other without being identical, and thus introducing vertex and edge overlaps.  $\square$

**Lemma 3.2.5.** *In any RAC drawing of the ASA graph, all edges from the central vertex  $v_0$  to quadrilateral  $Q_i$  are fully contained into  $Q_i$ ,  $i = 1, 2$ .*

*Proof.* By Lemma 3.2.4, vertex  $v_0$  should lie in the interior of quadrilateral  $Q_i$ ,  $i = 1, 2$ , which is drawn planar, due to Lemma 3.2.3. If both  $Q_1$  and  $Q_2$  are drawn convex, then the lemma trivially holds. For the sake of contradiction, assume that  $Q_1$  is drawn concave and an edge emanating from vertex  $v_0$  towards a vertex of quadrilateral  $Q_1$ , say  $v_a$ , crosses an edge, say  $(v_c, v_d)$ , of quadrilateral  $Q_1$  (see Figure 3.12).

**Proposition 7.** *The sum of the three acute internal angles of  $Q_1$  is less than  $\frac{\pi}{2}$*

*Proof.* Since  $Q_1$  is concave at  $v_d$ , the internal angle at  $v_d$  is greater than  $3\pi/2$ . Hence, the sum of the remaining internal angles is less than  $\pi/2$ .  $\square$



**Figure 3.12:** Configuration used in proof of Lemma 3.2.5 (Proposition 8): Potential placements of  $v_{cd}$ .

**Proposition 8.** Vertex  $v_{cd}$  should lie in the interior of  $Q_1$ .

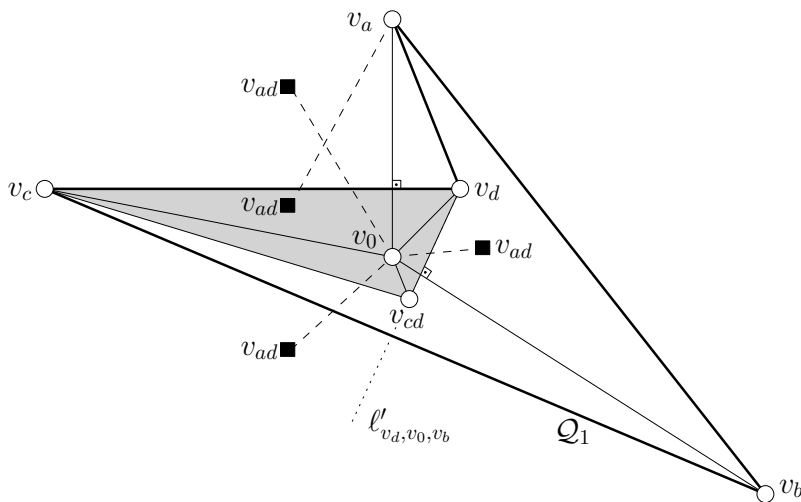
*Proof.* Recall that  $v_{cd}$  is connected to  $v_c$ ,  $v_d$  and  $v_0$  and assume, to the contrary, that vertex  $v_{cd}$  is not in the interior of  $Q_1$ . Then, the following hold (refer to Figure 3.12):

- i) Edge  $(v_0, v_{cd})$  is not entering  $Q_1$  from  $(v_c, v_d)$ : If it did, edges  $(v_0, v_{cd})$  and  $(v_0, v_a)$  form a fan anchored at  $v_0$  and crossed by  $(v_c, v_d)$ , which is not permitted due to RAC-Property 2.
- ii) Edge  $(v_0, v_{cd})$  is not entering  $Q_1$  from  $(v_a, v_d)$ : If it did, then it is implied that it is possible to draw from  $v_0$  perpendicular line segments (i.e.,  $(v_0, v_a)$  and  $(v_0, v_{cd})$ ) to two edges forming the concave angle of  $Q_1$ . However, this is not possible, since  $v_0$  is internal to  $Q_1$ .
- iii) Edge  $(v_0, v_{cd})$  is entering  $Q_1$  either from  $(v_a, v_b)$  or  $(v_b, v_c)$ : Trivially follows from (i) and (ii).
- iv) Vertex  $v_{cd}$  is not in region formed by  $l'_{v_d, v_a}$  and  $l'_{v_d, v_0}$  and contains  $v_c$ : In Figure 3.12, this region is shaded in light-gray. It follows from the fact that  $(v_d, v_{cd})$  cannot cross  $(v_0, v_a)$ , since otherwise edges  $(v_d, v_{cd})$  and  $(v_c, v_d)$  form a fan anchored at  $v_d$  and crossed by  $(v_0, v_a)$ , which is not permitted due to RAC-Property 2.
- v) Edge  $(v_d, v_{cd})$  is entering  $Q_1$  either from  $(v_a, v_b)$  or  $(v_b, v_c)$ : Trivially follows from (iv).

From (iii) and (v), it follows that  $v_{cd}$  should be at the intersection of two semi-lines (refer to the dotted semi-lines of Figure 3.12), which are perpendicular to  $(v_a, v_b)$  and/or  $(v_b, v_c)$ . However, this is a contradiction, since these lines do not intersect due to Proposition 7; they are perpendicular on two consecutive edges of  $\mathcal{Q}_1$  that form an acute angle.  $\square$

**Proposition 9.** Vertex  $v_{cd}$  lies in the interior of  $\triangle v_0 v_b v_c$ .

*Proof.* By Proposition 8, it is enough to prove that  $v_{cd}$  lies “below”  $\ell_{v_0, v_c}$  and not in  $\triangle v_0 v_b w$  (refer to the dark-gray shaded triangle of Figure 3.12), where  $w$  is the intersection point of  $\ell_{v_0, v_c}$  and  $(v_a, v_b)$ . The former property is obvious, since  $v_{cd}$  is connected to both  $v_c$  and  $v_d$ . Hence, if  $v_{cd}$  was “above”  $\ell_{v_0, v_c}$ , then  $(v_c, v_d)$  and either  $(v_c, v_{cd})$  or  $(v_d, v_{cd})$  would form a fan anchored at either  $v_c$  or  $v_d$ , respectively, that is crossed by  $(v_0, v_a)$ . If  $v_{cd}$  is in the interior of  $\triangle v_0 v_b w$ , then  $(v_c, v_{cd})$  should cross  $(v_0, v_b)$  at right angle. However, this is not possible since the perpendicular line from  $v_c$  to  $(v_0, v_b)$  is external to  $\triangle v_0 v_b w$  due to Proposition 7. Hence,  $v_{cd}$  is in the interior of  $\triangle v_0 v_b v_c$  (and along  $\ell'_{v_d, v_0, v_b}$ ; see Figure 3.13), as desired.  $\square$



**Figure 3.13:** Configuration used in proof of Lemma 3.2.5 (Propositions 10-12): Potential placements of  $v_{ad}$ .

**Proposition 10.** Vertex  $v_{ad}$  lies outside  $\triangle v_c v_d v_{cd}$ .

*Proof.* In Figure 3.13, triangle  $\triangle v_c v_d v_{cd}$  is shaded in gray. Assume, to the contrary, that  $v_{ad}$  lies in the interior of  $\triangle v_c v_d v_{cd}$ . Then, since  $v_0$  is internal as well, vertex  $v_a$ , which



is outside  $\triangle v_c v_d v_{cd}$ , should be connected to  $v_0$  and  $v_{ad}$ , that both lie in the interior of  $\triangle v_c v_d v_{cd}$ . This is a contradiction due to RAC-Property 3.  $\square$

**Proposition 11.** *Edge  $(v_0, v_{ad})$  cannot enter triangle  $\triangle v_c v_d v_{cd}$  neither from  $(v_c, v_d)$  nor from  $(v_d, v_{cd})$ .*

*Proof.* Follows from the fact that edges  $(v_c, v_d)$  and  $(v_d, v_{cd})$  are already crossed at right angle by edges  $(v_0, v_a)$  and  $(v_0, v_b)$ , respectively, which are incident to  $v_0$ , as is edge  $(v_0, v_{ad})$ .  $\square$

**Proposition 12.** *Vertex  $v_{ad}$  lies in the interior of  $\triangle v_0 v_b v_c$ .*

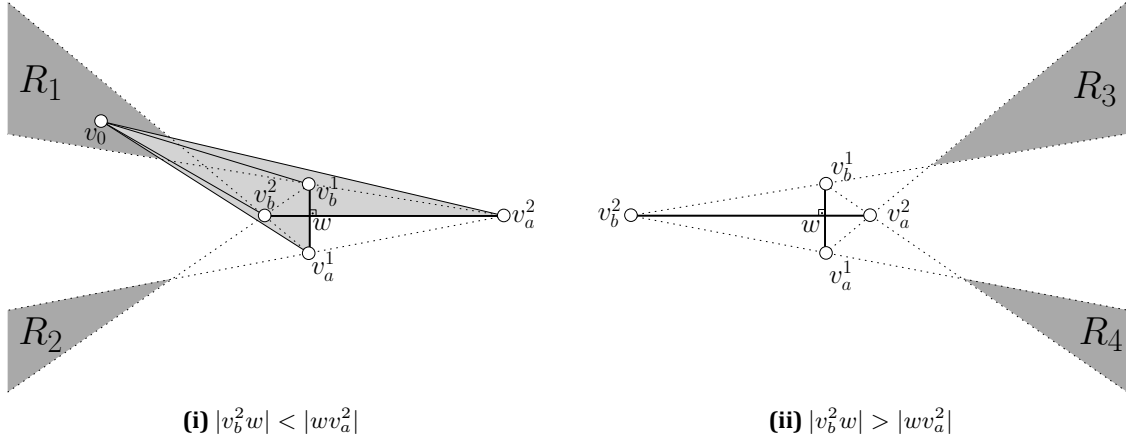
*Proof.* By Proposition 11, edge  $(v_0, v_{ad})$  enters triangle  $\triangle v_c v_d v_{cd}$  from  $(v_c, v_{cd})$ . If  $v_{ad}$  was external to  $\triangle v_0 v_b v_c$ , edges  $(v_c, v_{cd})$  and  $(v_b, v_c)$  should form a fan anchored at  $v_c$  and crossed by  $(v_0, v_{ad})$ , which is not permitted due to RAC-Property 2.  $\square$

Propositions 9 and 12 suggest that both  $v_{cd}$  and  $v_{ad}$  lie in the interior of  $\triangle v_0 v_b v_c$ . Based on this and following a similar reasoning scheme as in the proof of Proposition 6, we can prove that all vertices of  $\mathcal{Q}_2$  should lie in the interior of  $\triangle v_0 v_b v_c$ . However, this is a contradiction since vertex  $v_0$  should lie in the interior of  $\mathcal{Q}_2$ , due to Lemma 3.2.4.  $\square$

**Lemma 3.2.6.** *There does not exist a RAC drawing of the ASA graph, in which quadrilaterals  $\mathcal{Q}_1$  and  $\mathcal{Q}_2$  intersect.*

*Proof.* For the sake of contradiction, assume that  $\mathcal{Q}_1$  and  $\mathcal{Q}_2$  intersect. With slight abuse of notation, let  $v_a^i, v_b^i, v_c^i$  and  $v_d^i$  be  $\mathcal{Q}_i$ 's vertices consecutive along quadrilateral  $\mathcal{Q}_i$ ,  $i = 1, 2$ , i.e.,  $\{v_a, v_b, v_c, v_d\} = \{v_a^1, v_b^1, v_c^1, v_d^1\}$  and  $\{v_{ab}, v_{bc}, v_{cd}, v_{ad}\} = \{v_a^2, v_b^2, v_c^2, v_d^2\}$ . Let w.l.o.g.,  $(v_a^1, v_b^1) \in \mathcal{Q}_1$  and  $(v_a^2, v_b^2) \in \mathcal{Q}_2$  be a pair of edges that are involved in the crossing of  $\mathcal{Q}_1$  and  $\mathcal{Q}_2$  and let  $w$  be their intersection point. Consider a RAC drawing of the ASA graph in which  $(v_a^2, v_b^2)$  is drawn horizontal (and, hence,  $(v_a^1, v_b^1)$  is drawn vertical) and assume that  $v_b^2$  is to the left of  $v_a^2$ , whereas  $v_b^1$  is above  $v_a^1$  (see Figure 3.14).

Edges  $(v_0, v_a^1)$  and  $(v_0, v_b^1)$  should not cross  $(v_a^2, v_b^2)$ , since that would form a fan anchored at  $v_a^1$  and  $v_b^1$ , respectively, crossed by  $(v_a^2, v_b^2)$ , which is not permitted due to RAC-Property 2. Similarly, edges  $(v_0, v_a^2)$  and  $(v_0, v_b^2)$  should not cross  $(v_a^1, v_b^1)$ . This suggests that vertex  $v_0$  should lie in one of the dark-gray shaded unbounded regions of Figure 3.14. Assume w.l.o.g. that  $v_0$  lies in  $R_1$  (the remaining cases are treated symmetrically). Due to Lemma 3.2.4,  $v_0$  is internal to  $\mathcal{Q}_1$ . However, as we will show, there does not exist a legal placement of the vertices of  $\mathcal{Q}_2$ , such that  $\mathcal{Q}_1$  encloses  $v_0$ . In the following, we first prove that  $v_c^1$  should lie outside the light-gray shaded quadrilateral of Figure 3.14i.



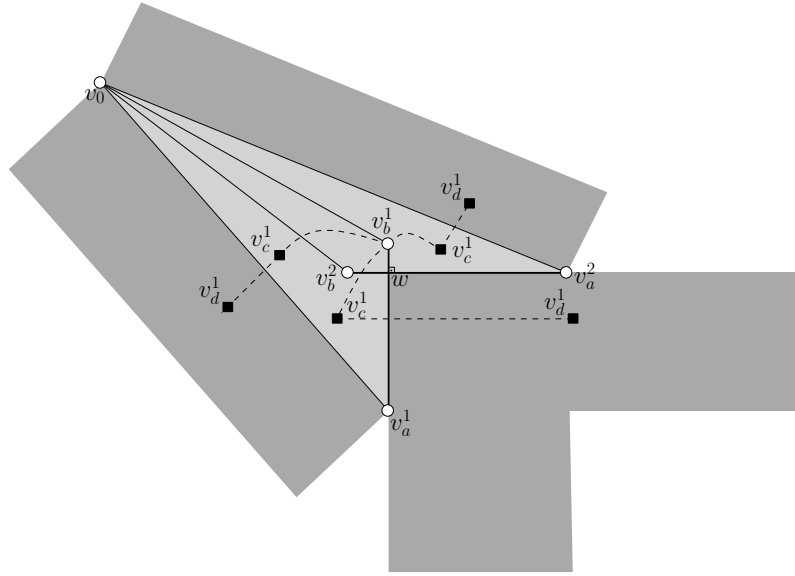
**Figure 3.14:** Configurations used in proof of Lemma 3.2.6: Vertex  $v_0$  should lie in the interior of one of the regions  $R_1, R_2, R_3$  and  $R_4$ .

**Proposition 13.** Vertex  $v_c^1$  lies outside quadrilateral  $wv_a^1v_0v_a^2$ , where  $w$  is the intersection of  $(v_a^1, v_b^1) \in \mathcal{Q}_1$  and  $(v_a^2, v_b^2) \in \mathcal{Q}_2$ .

*Proof.* Refer to Figure 3.15 and recall that quadrilateral  $\mathcal{Q}_1$  is formed by vertices  $v_a^1, v_b^1, v_c^1, v_d^1$  in this order. For the sake of contradiction, assume that  $v_c^1$  lies inside quadrilateral  $wv_a^1v_0v_a^2$ . If  $v_d^1$  lies inside quadrilateral  $wv_a^1v_0v_a^2$  as well, then  $\mathcal{Q}_1$  lies entirely within quadrilateral  $wv_a^1v_0v_a^2$ . Hence,  $v_0$  is not in the interior of  $\mathcal{Q}_1$ , which is a contradiction due to Lemma 3.2.4. Therefore,  $v_d^1$  should be outside quadrilateral  $wv_a^1v_0v_a^2$ . Since  $v_c^1$  is inside quadrilateral  $wv_a^1v_0v_a^2$  and  $v_d^1$  outside, edge  $(v_c^1, v_d^1)$  should be perpendicular to an edge of  $wv_a^1v_0v_a^2$ . This suggests that  $v_d^1$  should lie within a dark-gray shaded region of Figure 3.15, which implies again that  $v_0$  is not in the interior of  $\mathcal{Q}_1$ , since edge  $(v_d^1, v_a^1)$  forms a quadrilateral which does not enclose  $v_0$ . A clear contradiction due to Lemma 3.2.4.  $\square$

Proposition 13 and Lemma 3.2.4 suggest that quadrilateral  $\mathcal{Q}_1$  should be drawn as shown in Figure 3.16, i.e., edge  $(v_b^1, v_c^1)$  should be perpendicular to  $(v_0, v_a^2)$  whereas vertex  $v_d^1$  should be to the “left” of  $v_0$  such that edges  $(v_c^1, v_d^1)$  and  $(v_a^1, v_d^1)$  do not cross quadrilateral  $wv_a^1v_0v_a^2$  and  $v_0$  is in the interior of  $\mathcal{Q}_1$ . We proceed to investigate how  $\mathcal{Q}_2$  is drawn.

**Proposition 14.** Vertex  $v_c^2$  lies outside quadrilateral  $wv_a^1v_0v_a^2$ , where  $w$  is the intersection of  $(v_a^1, v_b^1) \in \mathcal{Q}_1$  and  $(v_a^2, v_b^2) \in \mathcal{Q}_2$ .



**Figure 3.15:** Configuration used in proof of Lemma 3.2.6 (Proposition 13): Potential placements of  $v_d^1$ , assuming that  $v_0$  is in the interior of  $R_1$  and  $v_c^1$  in the interior of  $wv_a^1v_0v_a^2$ .

*Proof.* Similarly to Proposition 13. □

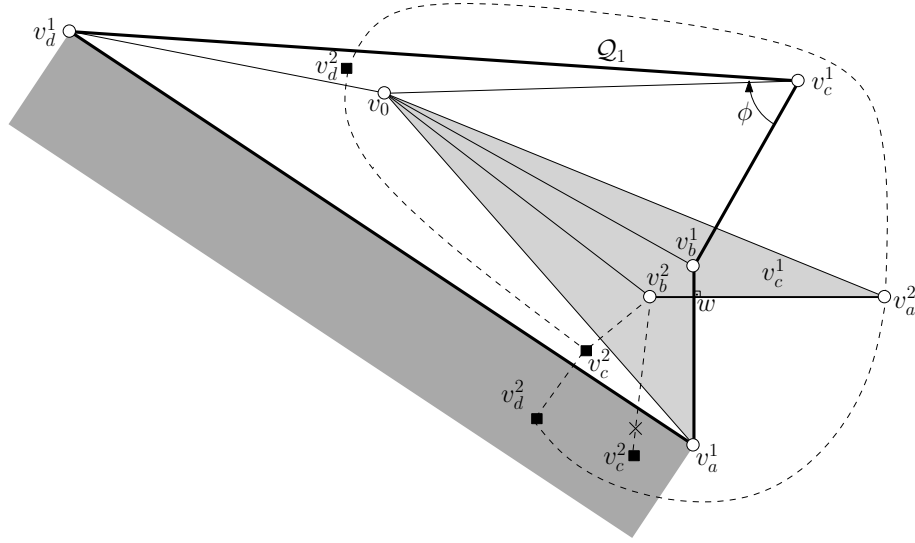
**Proposition 15.** Vertex  $v_c^2$  lies in the interior  $Q_1$ .

*Proof.* Proposition 14 implies that vertex  $v_c^2$  lies outside quadrilateral  $wv_a^1v_0v_a^2$ . If in addition  $v_c^2$  lies outside  $Q_1$ , then edges  $(v_0, v_a^1)$  and  $(v_d^1, v_a^1)$  form a fan at  $v_a^1$  crossed by  $(v_b^2, v_c^2)$ , which is not permitted by RAC-Property 2 (see Figure 3.16). □

From the above propositions it follows that vertex  $v_c^2$  should lie outside quadrilateral  $wv_a^1v_0v_a^2$ , but in the interior of  $Q_1$ . In the following, we will prove that  $v_d^2$  can neither lie in the interior of  $Q_1$  nor to its exterior, leading thus to a contradiction our initial hypothesis that  $Q_1$  and  $Q_2$  intersect. Refer to Figure 3.16.

**Proposition 16.** Vertex  $v_d^2$  lies in the interior of  $Q_1$ .

*Proof.* Assume to the contrary that  $v_d^2$  lies outside  $Q_1$ . Then, it should reside within the dark-gray shaded unbounded region of Figure 3.16, such that  $(v_c^2, v_d^2)$  is perpendicular to  $(v_a^1, v_d^1)$ . This implies that  $v_0$  cannot lie in the interior of  $Q_2$ , since edge  $(v_d^2, v_a^2)$  forms a quadrilateral which does not enclose  $v_0$ , which leads to a contradiction. Therefore,  $v_d^2$  should lie in the interior of  $Q_1$ . □



**Figure 3.16:** Configuration used in proof of Lemma 3.2.6: Potential placements of  $v_d^2$ , assuming that  $v_0$  is in the interior of  $R_1$  and  $v_c^2$  lies outside  $wv_a^1v_0v_a^2$ , but in the interior of  $Q_1$ .

By Proposition 16, edge  $(v_a^2, v_d^2)$  should perpendicularly cross an edge of  $Q_1$ , because  $v_d^2$  is inside it, whereas  $v_a^2$  outside it. First, observe that  $(v_a^2, v_d^2)$  can be perpendicular to neither  $(v_c^1, v_d^1)$  nor  $(v_a^1, v_d^1)$ . To see this assume, to the contrary, that  $(v_a^2, v_d^2)$  is perpendicular to  $(v_c^1, v_d^1)$ . Then, angle  $\widehat{v_b^1 v_c^1 v_d^1}$  (denoted by  $\phi$  in Figure 3.16) should be greater than  $\pi$ , which contradicts the fact that  $v_0$  is in the interior of  $Q_1$ . Similarly, we can prove that  $(v_a^2, v_d^2)$  cannot be perpendicular to  $(v_a^1, v_d^1)$ . Hence,  $(v_d^2, v_a^2)$  should be perpendicular to either  $(v_b^1, v_c^1)$  or  $(v_a^1, v_b^1)$ . In the case where  $(v_d^2, v_a^2)$  crosses  $(v_b^1, v_c^1)$ , edges  $(v_d^2, v_a^2)$  and  $(v_0, v_a^2)$  form a fan anchored at  $v$  and crossed by  $(v_b^1, v_c^1)$ , which by RAC-Property 2 is not permitted. Similarly,  $(v_d^2, v_a^2)$  cannot be perpendicular to  $(v_a^1, v_b^1)$ . Therefore,  $v_d^2$  cannot lie in the interior of  $Q_1$ , contradicting Proposition 16.  $\square$

**Theorem 3.2.1.** *The straight-line RAC drawings of the ASA graph define exactly two combinatorial embeddings.*

*Proof.* So far, we have managed to prove that both quadrilaterals  $Q_1$  and  $Q_2$  are drawn planar, do not cross, and have central vertex  $v_0$  to their interiors. This suggests that either quadrilateral  $Q_1$  is in the interior of  $Q_2$ , or quadrilateral  $Q_2$  is in the interior of  $Q_1$ . However, in both cases vertex  $v_0$ , which has to be connected to the four vertices of the “external” quadrilateral, should inevitably perpendicularly cross the four edges of the “internal” quadrilateral. This implies only two feasible combinatorial embeddings. The two combinatorial embeddings are shown in Figure 3.1i and 3.1ii.  $\square$

We extend the ASA graph by appropriately *glueing* multiple instances of it, the one next to the other. Figure 3.17i demonstrates how this operation is realized on two instances, say  $G$  and  $G'$ , of the ASA graph, i.e., by identifying two “external” vertices, say  $v$  and  $v'$ , of  $G$  with two “external” vertices of  $G'$  (refer to the gray-shaded vertices of Figure 3.17i), and by employing an additional edge (refer to the dashed drawn edge of Figure 3.17i), which connects an “internal” vertex, say  $u$ , of  $G$  with the corresponding “internal” vertex, say  $u'$ , of  $G'$ . Let  $G \oplus G'$  be the graph produced by the glueing operation on  $G$  and  $G'$ . Since the RAC drawings of  $G$  and  $G'$  define two combinatorial embeddings each, one would expect that the RAC drawings of  $G \oplus G'$  would define four possible combinatorial embeddings. We will show that this is not true and, more precisely, that there exists only a single combinatorial embedding.

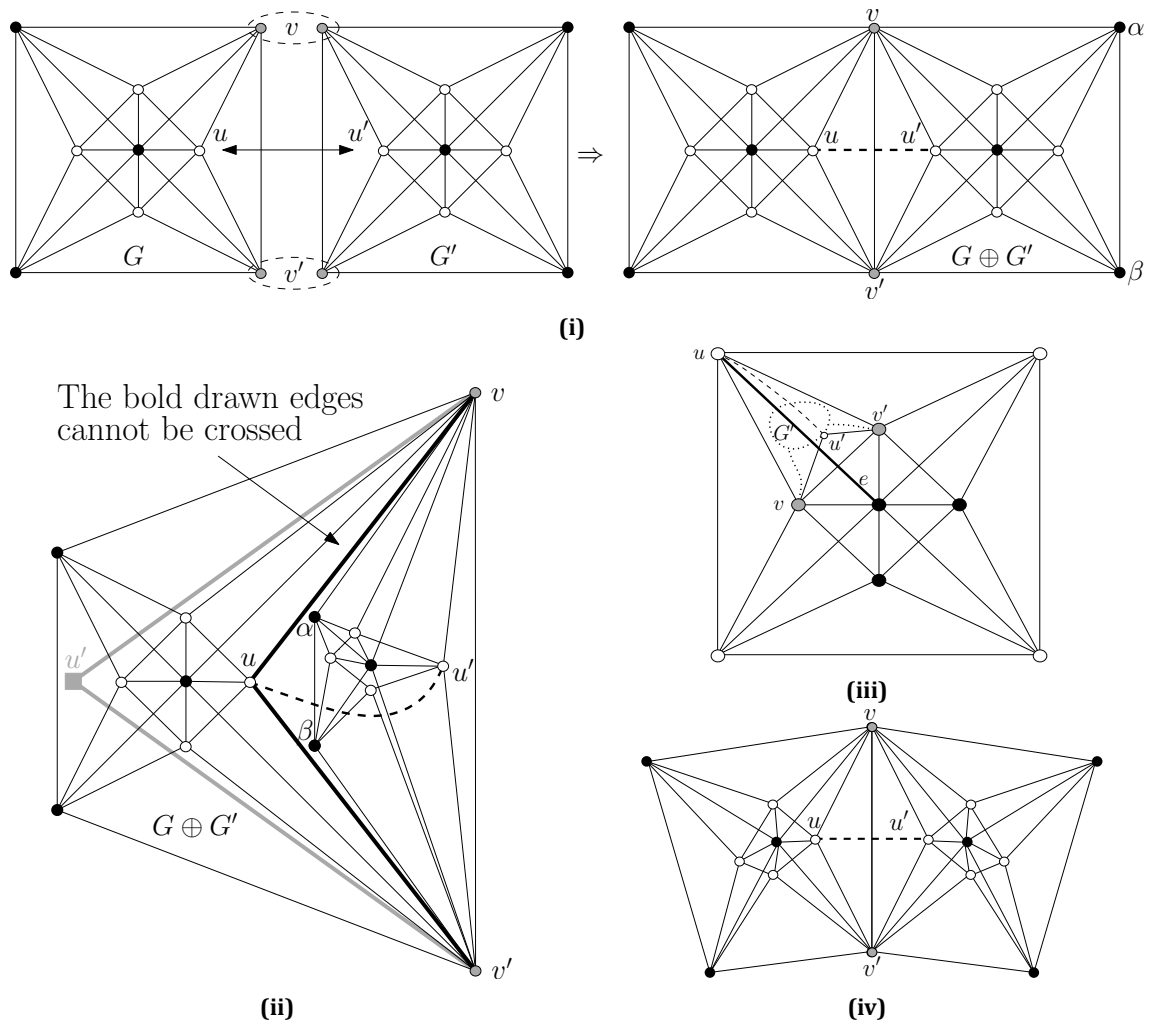
**Theorem 3.2.2.** *Let  $G$  and  $G'$  be two instances of the ASA graph. Then, the straight-line RAC drawings of  $G \oplus G'$  define a single RAC combinatorial embedding.*

*Proof.* Obviously, in any RAC drawing of  $G \oplus G'$ , both  $G$  and  $G'$  should be drawn RAC. Say that in a RAC drawing of  $G \oplus G'$ ,  $G$  is drawn such that vertices  $v$  and  $v'$  (that are identified during the glueing operation) are on the external face of  $\Gamma(G)$ .<sup>3</sup> Then, the common neighbor of  $v$  and  $v'$  in  $G'$  that is not identified with  $v_0$  (i.e.,  $u'$ ) can be either in the interior of  $\Gamma(G)$  or to its external face in the drawing of  $G \oplus G'$ .

We first consider the case where vertex  $u'$  is in the interior of  $\Gamma(G)$  in the drawing of  $G \oplus G'$  (see Figure 3.17ii). Vertex  $u'$ , which is incident to both  $v$  and  $v'$ , cannot reside to the “left” of both edges  $(u, v)$  and  $(u, v')$  (refer to the bold drawn edges of Figure 3.17ii), since this would lead to a situation where three edges mutually cross and, subsequently, to a violation of RAC-Property 1 (see the gray-shaded square vertex of Figure 3.17ii). Therefore, vertex  $u'$  should lie within the triangular face of  $G$  formed by vertices  $u$ ,  $v$  and  $v'$ . Similarly, the same holds for the central vertex of  $G'$ , which is also incident to vertices  $v$  and  $v'$ . By RAC-Property 3, any common neighbor of vertices  $u'$  and  $v$  should also lie within the same triangular face of  $G$ , which progressively implies that the entire graph  $G'$  should reside within this face, as in Figure 3.17ii. Since vertices  $v$  and  $v'$  are on the external face of  $\Gamma(G')$  in the drawing of  $G \oplus G'$  and  $G'$  should be drawn RAC, vertices  $\alpha$  and  $\beta$  in Figure 3.17ii should be on the external face of  $\Gamma(G')$  as well (due to Theorem 3.2.1). However, in this case and since  $u'$  is incident to  $v$  and  $v'$ , edge  $(u, u')$ , which is employed during the glueing operation, crosses the interior of  $G'$ , which is not permitted. This suggests that if vertex  $u'$  is in the interior of  $\Gamma(G)$  in the drawing of  $G \oplus G'$ , then there is no feasible embedding.

If one assumes that  $u'$  is on the external face of  $\Gamma(G)$  in the drawing of  $G \oplus G'$ , then it can be similarly proved that the entire graph  $G'$  should be on this face, too. Hence,  $v$  and  $v'$  are on

<sup>3</sup>The case where  $v$  and  $v'$  are not on the external face of  $\Gamma(G)$  will be examined later.



**Figure 3.17:** (i) Glueing two instances of the ASA graph, (ii) The additional (dashed) edge does not permit the second instance to be drawn in the interior of the first one. (iii) The vertices, which are identified during the glueing operation ( $v$  and  $v'$  in figure), should be on the external face of each ASA graph. (iv) Each glueing operation may introduce a “turn” in the corresponding RAC drawing.

the external face of  $\Gamma(G')$ , as well. From the discussion above, it follows that  $v$  and  $v'$  are on the external face of both  $\Gamma(G)$  and  $\Gamma(G')$  in the drawing of  $G \oplus G'$ , which implies a feasible embedding (see Figure 3.17iv).

We now examine the case where  $v$  and  $v'$  are not on the external face of  $\Gamma(G)$  in a RAC drawing of  $G \oplus G'$ , i.e.,  $v$  and  $v'$  are along the internal quadrilateral of  $G$  in the RAC drawing of  $G \oplus G'$ . This is illustrated in Figure 3.17iii. Let  $e$  be the edge of  $G$ , which perpendicularly crosses edge  $(v, v')$  and emanates from the external quadrilateral towards the central vertex

of  $G$  (refer to the bold solid edge of Figure 3.17iii). Edge  $e$  will be involved in crossings with  $G'$ . Let  $u'$  be the common neighbor of  $v$  and  $v'$  in  $G'$ . Then,  $e$ ,  $(v, v')$  and either  $(v, u')$  or  $(u', v')$  form three mutually crossing edges in the drawing of  $G \oplus G'$ , which is not permitted due to RAC-Property 1.

Therefore, the vertices that are identified during a glueing operation should always be on the external face of each ASA graph and, subsequently, any drawing of  $G \oplus G'$  has unique combinatorial embedding.  $\square$

Note that, the RAC drawing of  $G \oplus G'$  may differ from the drawing of Figure 3.17i. In the general case, each glueing operation may introduce a “turn” in the corresponding RAC drawing, as in Figure 3.17iv. However, the combinatorial embedding is still the same.

### 3.3 The Straight-Line RAC Drawing Problem is NP-hard

In this section, we will reduce the well-known 3-SAT problem [GJ79] to the straight-line RAC drawing problem. In a 3-SAT instance, we are given a formula  $\phi$  in conjunctive normal form with variables  $x_1, x_2, \dots, x_n$  and clauses  $C_1, C_2, \dots, C_m$ , each with three literals. We show how to construct a graph  $G_\phi$  that admits a straight-line RAC drawing  $\Gamma(G_\phi)$  if and only if formula  $\phi$  is satisfiable. Figure 3.20 shows an example of a graph which is build by our reduction based on a particular input 3-SAT formula. Our proof follows the general approach of Formann et al. [FHH<sup>+</sup>93] (used to prove that the angular resolution maximization problem is  $\mathcal{NP}$ -hard) using different gadgets to encode the variables and the clauses of the given formula.

#### 3.3.1 Description of the Construction

Figure 3.18 illustrates the gadgets of our construction. Each gray-shaded square in these drawings corresponds to an ASA graph. Adjacent gray squares correspond to glued ASA graphs (refer, for example, to the topmost gray squares of Figure 3.18i). There also exist gray squares that are not adjacent, but connected through edges. The legend in Figure 3.18 describes how these connections are realized.

The gadget that encodes variable  $x_i$  of formula  $\phi$  is given in Figure 3.18i. It consists of a combination of ASA graphs, and, “horizontal” and “vertical” edges, which form a tower, such that the RAC drawings of each tower define a single combinatorial embedding. One side of the tower accommodates multiple vertices that correspond to literal  $x_i$ , whereas its opposite side accommodates vertices that correspond to its negation  $\bar{x}_i$  (refer to vertices  $x_{i,1}, \dots, x_{i,m}$  and  $\bar{x}_{i,1}, \dots, \bar{x}_{i,m}$  in Figure 3.18i). These vertices are called *variable endpoints*. Then, based on whether on the final drawing the negated vertices will appear to the “left” or to the “right”

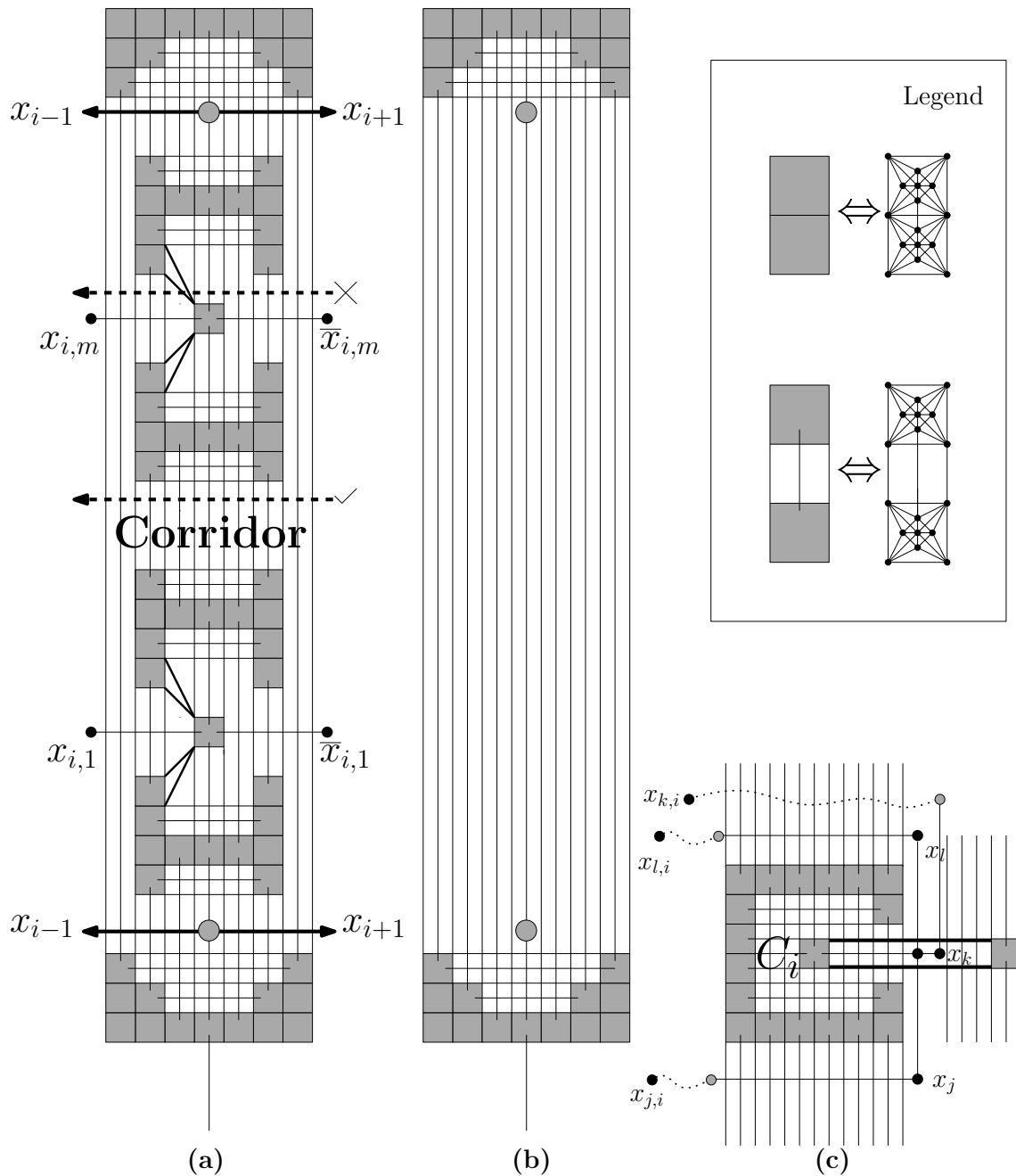


Figure 3.18: Gadgets of our construction: (i) Variable gadget, (ii) Dummy variable gadget, (iii) Clause gadget



side of the tower, we will assign a true or a false value to variable  $x_i$ , respectively. Pairs of consecutive endpoints  $x_{i,j}$  and  $x_{i,j+1}$  are separated by a *corridor* (marked by a “tick” in Figure 3.18i), which allows perpendicular edges to pass through it (see the bottommost dashed arrow of Figure 3.18i). Note that, no edge can pass through a “corridor” formed on a variable endpoint (marked by a cross in Figure 3.18i), since there exist four non-parallel edges that “block” any other edge passing through them (see the topmost dashed arrow of Figure 3.18i). The corridors can have variable height. In the variable gadget of variable  $x_i$ , there are also two vertices (drawn as gray circles in Figure 3.18i), which have degree four. These vertices serve as “connectors” among consecutive variable gadgets, i.e., these vertices should be connected to their corresponding vertices on the variable gadgets of variables  $x_{i-1}$  and  $x_{i+1}$ . Note that, the connector vertices of the variable gadgets associated with variables  $x_1$  and  $x_n$  are connected to connectors of the variable gadgets that correspond to variables  $x_2$  and  $x_{n-1}$ , respectively, and to connectors of *dummy variable gadgets*.

Figure 3.18ii illustrates a dummy variable gadget, which (similarly to the variable gadget) consists of a combination of ASA graphs, and, “horizontal” and “vertical” edges, which form a tower. The RAC drawings of this gadget also define a single combinatorial embedding. A dummy variable gadget does not support vertices that correspond to literals. However, it contains connector vertices (they are drawn as gray circles in Figure 3.18ii). In our construction, we use exactly two dummy variable gadgets. The connector vertices of each dummy variable gadget should be connected to their corresponding connector vertices on the variable gadgets associated with variables  $x_1$  and  $x_n$ , respectively.

The gadget that encodes the clauses of formula  $\phi$  is illustrated in Figure 3.18iii and resembles to a valve. Let  $C_i = (x_j \vee x_k \vee x_l)$  be a clause of  $\phi$ . As illustrated in Figure 3.18iii, the gadget which corresponds to clause  $C_i$  contains three vertices<sup>4</sup>, say  $x_j$ ,  $x_k$ , and  $x_l$ , such that:  $x_j$  has to be connected to  $x_{j,i}$ ,  $x_k$  to  $x_{k,i}$  and  $x_l$  to  $x_{l,i}$  by paths of length two. These vertices, referred to as the *clause endpoints*, encode the literals of each clause. Obviously, if a clause contains a negated literal, it should be connected to the negated endpoint of the corresponding variable gadget. The clause endpoints are incident to a vertex “trapped” within two parallel edges (refer to the bold drawn edges of Figure 3.18iii). Therefore, in a RAC drawing of  $G_\phi$ , only two of them can perpendicularly cross these edges, one from top (*top endpoint*) and one from bottom (*bottom endpoint*). The other one (*right endpoint*) should remain in the interior of the two parallel edges. The one that will remain “trapped” on the final drawing will correspond to the true literal of this clause.

The gadgets, which correspond to variables and clauses of  $\phi$ , are connected together by the skeleton of graph  $G_\phi$ , which is depicted in Figure 3.19i. The skeleton consists of two main

<sup>4</sup>With slight abuse of notation, the same term is used to denote variables of  $\phi$  and vertices of  $G_\phi$ .

parts, i.e., one “horizontal” and one “vertical”. The vertical part accommodates the clause gadgets (see Figure 3.19i). The horizontal part will be used in order to “plug” the variable gadgets. The long edges that perpendicularly cross (refer to the crossing edges slightly above the horizontal part in Figure 3.19i), imply that the vertical part should be perpendicular to the horizontal part. The horizontal part of the skeleton is separately illustrated in Figure 3.19ii. Observe that it contains one set of horizontal lines, which in conjunction with the vertical edges of the variable and clause gadgets do not allow it to bend.

Figure 3.20 shows how the variable gadgets are attached to the skeleton. More precisely, this is accomplished by a single edge, which should perpendicularly cross the set of the horizontal edges of the horizontal part. Therefore, each variable gadget is perpendicularly attached to the skeleton, as in Figure 3.20. Note that, each variable gadget should be drawn completely above these horizontal edges, since otherwise the connections among variable endpoints and clause endpoints would not be feasible. The connector vertices of the dummy variable gadgets, the variable gadgets and the vertical part of the construction, ensure that the variable gadgets will be parallel to each other (i.e., they are not allowed to bend) and parallel to the vertical part of the construction.

### 3.3.2 Properties of the Construction

We now proceed to investigate some properties of our construction. Any path of length two that emanates from a top- or bottom-clause endpoint can reach a variable endpoint either on the left or on the right side of its associated variable gadget. The first edge of this path should perpendicularly cross the vertical edges of the vertical part of the construction and pass through some corridors<sup>5</sup>, whereas the second edge will be used to realize the “final” connection with the variable gadget endpoint (see Figure 3.20).

The same doesn’t hold for the paths that emanate from a right-clause endpoint. These paths can only reach variable endpoints on the right side of their associated variable gadgets. More precisely, the first edge of the 2-length path should cross one of the two parallel edges (refer to the bold drawn edges of Figure 3.18c) that “trap” it, whereas the other one should be used to reach (passing through variable corridors) its variable endpoint (see Figure 3.20).

Our construction ensures that up to translations, rotations and stretchings any RAC drawing of  $G_\phi$  looks like the one of Figure 3.20.

### 3.3.3 Reduction from 3-SAT

**Theorem 3.3.1.** *It is NP-hard to decide whether an input graph admits a straight-line RAC drawing.*

<sup>5</sup>In Figure 3.20, the corridors are the gray-shaded regions that reside at each variable gadget.

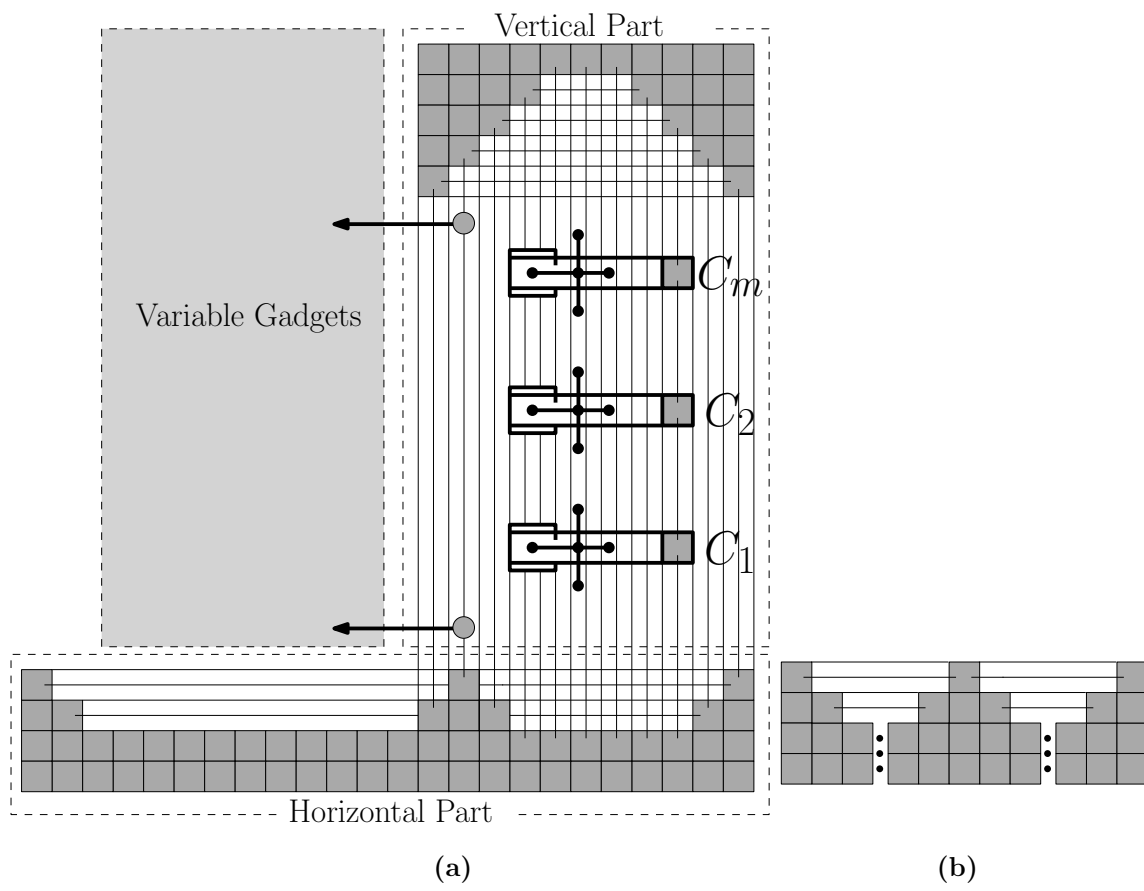
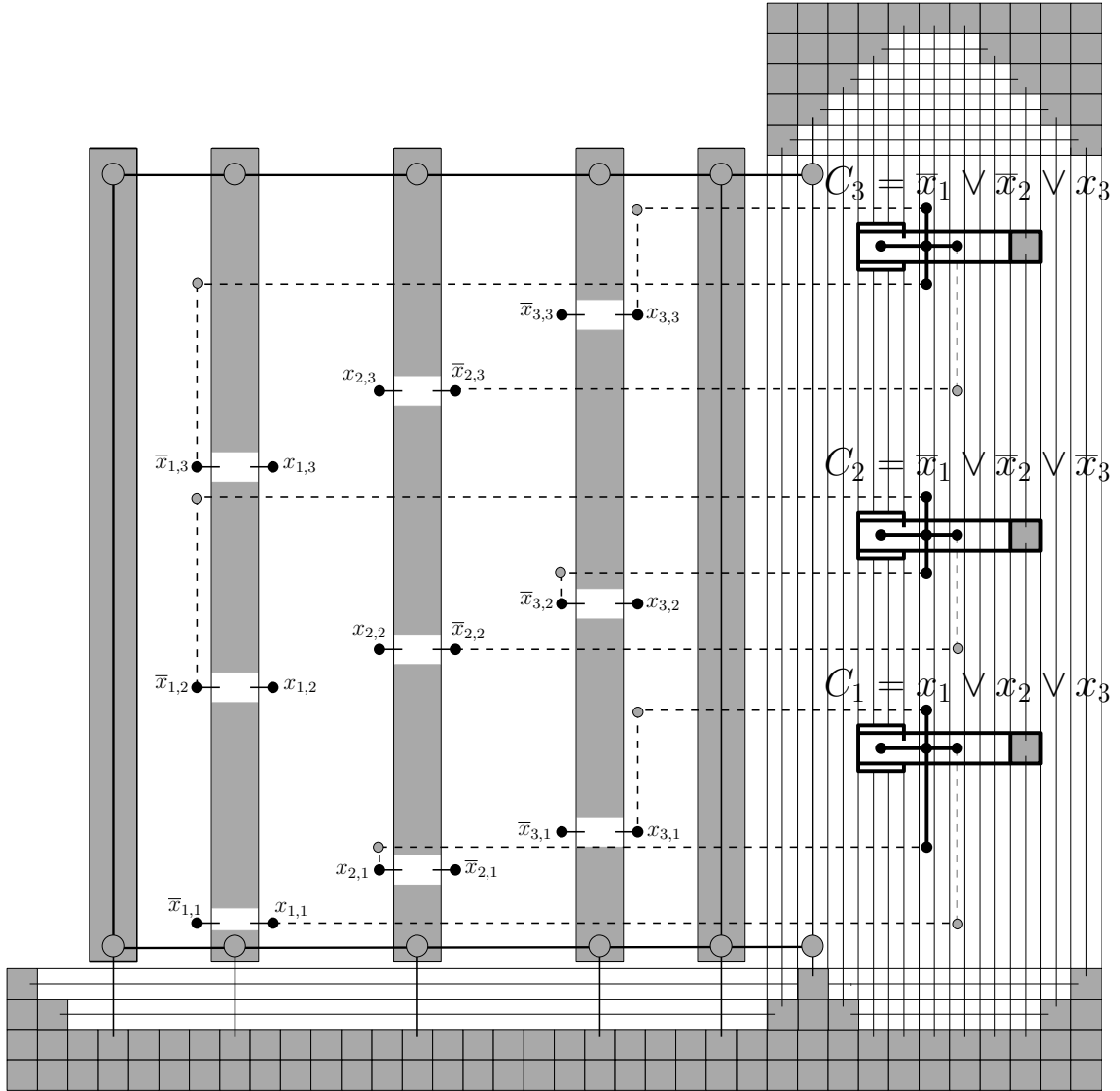


Figure 3.19: Illustration of the skeleton of the construction.



**Figure 3.20:** The reduction from 3-SAT to the straight-line RAC drawing problem. The input formula is  $\phi = (x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_3)$ . The drawing corresponds to the truth assignment  $x_1=x_3=\text{true}, x_2=\text{false}$ .

*Proof.* It is clear that the construction can be completed in  $O(nm)$  time. Assume now that there is a RAC drawing  $\Gamma(G_\phi)$  of  $G_\phi$ . If the negated vertices of the variable gadget that corresponds to  $x_i$ ,  $i = 1, 2, \dots, n$ , lie to the “left” side in  $\Gamma(G_\phi)$ , then variable  $x_i$  is set to true, otherwise  $x_i$  is set to false. We argue that this assignment satisfies  $\phi$ . To realize this, observe that there exist three paths that emanate from each clause gadget. The one that emanates from the right endpoint of each clause gadget can never reach a false value. Therefore, each clause of  $\phi$  must contain at least one true literal, which implies that  $\phi$  is satisfiable.

Conversely, suppose that there is a truth assignment that satisfies  $\phi$ . We proceed to construct a RAC drawing  $\Gamma(G_\phi)$  of  $G_\phi$ , as follows: In the case where, in the truth assignment, variable  $x_i$ ,  $i = 1, 2, \dots, n$  is set to true, we place the negated vertices of the variable gadget that corresponds to  $x_i$ , to its left side in  $\Gamma(G_\phi)$ , otherwise to its right side. Since each clause of  $\phi$  contains at least one true literal, we choose this as the right endpoint of its corresponding clause gadget. As mentioned above, it is always feasible to be connected to its variable gadgets by paths of length two. This completes our proof.  $\square$





## 4 Geometric RAC Simultaneous Drawings of Graphs

In this chapter, we study the geometric RAC simultaneous drawing problem (or GRacSim for short). Recall that according to the GRacSim problem, we are given two planar graphs that share a common vertex set, and our intention is to construct a straight-line drawing in which each graph is drawn planar, there are no edge overlaps, and, crossings between edges of the two graphs occur at right angles (refer to Section 2.4). We first prove that two planar graphs admitting a geometric simultaneous drawing may not admit a geometric RAC simultaneous drawing. We further show that a cycle and a matching always admit a geometric RAC simultaneous drawing. We also study a closely related problem, referred to as geometric graph-dual RAC simultaneous drawing problem (or GDual-GRacSim for short). We prove that it is always possible to construct such a drawing if the input graph is an outerplanar embedded graph.

### 4.1 A Wheel and a Matching: A Negative Result

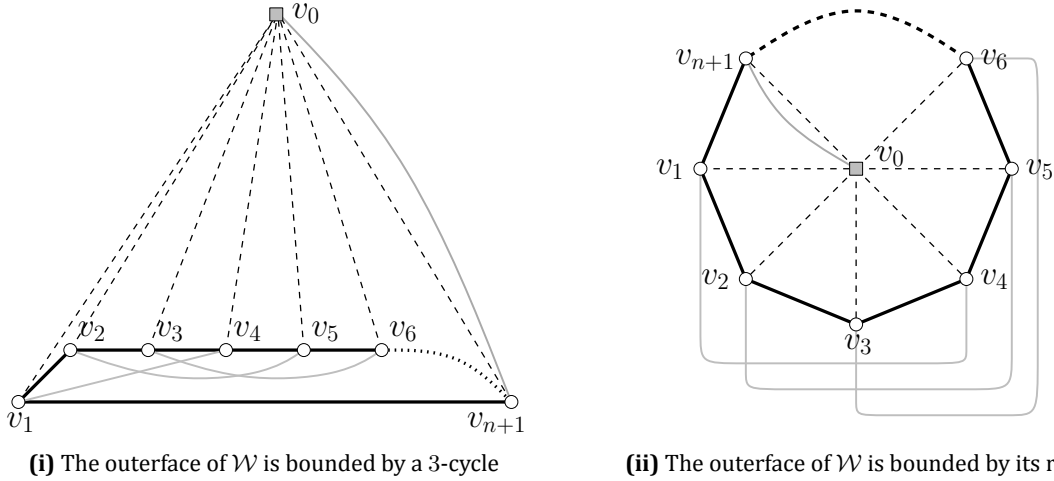
In this section, we show that there exists a pair of planar graphs that admits a geometric simultaneous drawing, their union meets the RAC size constraint and they do not admit a GRacSim drawing (i.e, the class of graphs that admit GRacSim drawings is a subset of the class of graphs for which a simultaneous drawing is possible). We achieve this by showing that there exists a wheel and a matching which do not admit a GRacSim drawing<sup>1</sup>. Cabello et al. [CvKL<sup>+</sup>11] have shown that a geometric simultaneous drawing of a wheel and a matching always exists.

**Theorem 4.1.1.** *There exists a wheel and a matching which do not admit a GRacSim drawing.*

*Proof.* We denote the wheel by  $\mathcal{W}$  and the matching by  $\mathcal{M}$ . Let the common vertex set of  $\mathcal{W}$  and  $\mathcal{M}$  be  $V = \{v_0, v_1, \dots, v_{n+1}\}$ , where  $n \geq 6$ . We further assume that  $n = 6k$ , for some  $k \in \mathbb{Z}$ . If  $v_0$  is the center of  $\mathcal{W}$  and  $v_1 \rightarrow \dots \rightarrow v_{n+1} \rightarrow v_1$  is the rim of  $\mathcal{W}$ , then  $E(\mathcal{W}) = \{(v_i, v_{i+1}); i = 1, \dots, n\} \cup \{(v_{n+1}, v_1)\} \cup \{(v_0, v_i); i = 1, \dots, n+1\}$ . Matching  $\mathcal{M}$

<sup>1</sup>A wheel and a matching on a vertex set of size  $n$  contribute  $5n/2 - 2$  edges, which meets the RAC size constraint.

contributes  $n/2 + 1$  edges; one edge of  $\mathcal{M}$  connects  $v_0$  with  $v_{n+1}$ . The  $6k$  vertices on the rim (excluding vertex  $v_{6k+1}$ ) are split into  $k$  groups, with group  $i$ ,  $1 \leq i \leq k$ , consisting of vertices  $v_{6i}, \dots, v_{6i+5}$ . Then, in each group  $i$ , vertex  $v_{6i+j}$  is matched with  $v_{6i+j+3}$ ,  $j = 0, 1, 2$ . More formally,  $E(\mathcal{M}) = \{(v_{6i-j-3}, v_{6i-j}); i = 1, \dots, n/6, j = 0, 1, 2\} \cup \{(v_0, v_{n+1})\}$ .



**Figure 4.1:** In both figures, the center of  $\mathcal{W}$  is marked by a box, the spokes of  $\mathcal{W}$  are drawn as dashed line-segments, the rim of  $\mathcal{W}$  is drawn in bold, while matching  $\mathcal{M}$  is drawn in gray. The examples correspond to the case where  $k = 1$ , i.e., wheel of rim-size seven. Wheels of rim-size  $6k + 1$ ,  $k > 1$  can be obtained by subdividing the dotted drawn edges and appropriately inserting wheel and matching edges.

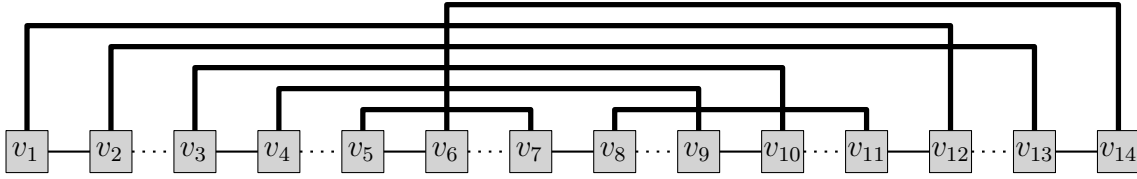
In any planar drawing of  $\mathcal{W}$ , the outerface is bounded either by a 3-cycle formed by  $v_0$  and two consecutive vertices of the rim of  $\mathcal{W}$  (see Figure 4.1i where we assume, without loss of generality, that  $(v_1, v_{n+1})$  is an edge of the boundary) or by the rim of  $\mathcal{W}$  itself (see Figure 4.1ii). Now observe that in both cases each edge of  $\mathcal{M}$  (except for edge  $(v_0, v_{n+1})$ ) connects two vertices that belong to two triangles of  $\mathcal{W}$  incident at  $v_0$  and are not consecutive around  $v_0$ , in the planar drawing of  $\mathcal{W}$ . Hence, by RAC-Property 2, it follows that the edges of  $\mathcal{M}$  cannot cross the spokes of  $\mathcal{W}$ . This implies that the edges of  $\mathcal{M}$  and the edges of  $\mathcal{W}$  cannot cross with each other, except for edge  $(v_0, v_{n+1})$  that belongs to both  $\mathcal{M}$  and  $\mathcal{W}$ . However, under this assumption,  $\mathcal{M}$  cannot be drawn planar.  $\square$

## 4.2 A Cycle and a Matching: A Positive Result

In this section, we first prove that a path and a matching always admit a GRacSim drawing and then, we show that a cycle and a matching always admit a GRacSim drawing, as well. Note that, the union of a path and a matching is not necessarily a planar graph. Cabello et al.



[CvKL<sup>+</sup>11] provide an example of a path and a matching, which form a subdivision of  $K_{3,3}$ . We denote the path by  $\mathcal{P}$  and the matching by  $\mathcal{M}$ . Let  $v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_n$  be the edges of  $\mathcal{P}$  (see Figure 4.2). In order to keep the description of our algorithm simple, we will initially assume that  $\mathcal{P}$  and  $\mathcal{M}$  do not share edges, i.e.,  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ . Since  $\mathcal{P}$  and  $\mathcal{M}$  are defined on the same vertex set,  $n$  should be even and  $|E(\mathcal{M})| = n/2$  (i.e.,  $\mathcal{M}$  is a perfect matching). Later on this section, we will describe how to cope with the cases where  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$  or  $\mathcal{M}$  is not a perfect matching.



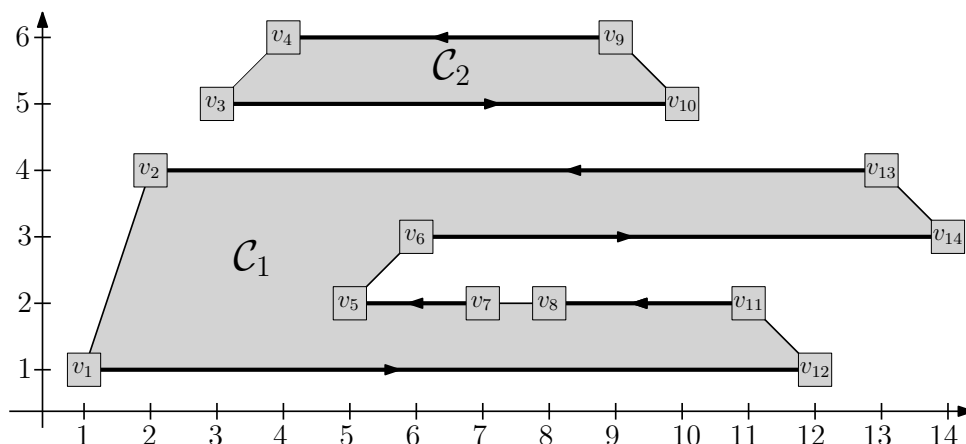
**Figure 4.2:** An example of a path  $\mathcal{P}$  and a matching  $\mathcal{M}$ . The path appears at the bottom of the figure. The edges of  $\mathcal{M}$  are drawn bold, with two bends each. The edges of path  $\mathcal{P}$  form two matchings, i.e.,  $\mathcal{P}_{odd}$  and  $\mathcal{P} - \mathcal{P}_{odd}$ . The edges of  $\mathcal{P}_{odd}$  are drawn solid, while the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  dotted.

The basic idea of our algorithm is to identify in the graph induced by the union of  $\mathcal{P}$  and  $\mathcal{M}$  a set of cycles  $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ ,  $k \leq n/4$ , such that: (i)  $|E(\mathcal{C}_1)| + |E(\mathcal{C}_2)| + \dots + |E(\mathcal{C}_k)| = n$ , (ii)  $\mathcal{M} \subseteq \mathcal{C}_1 \cup \mathcal{C}_2 \cup \dots \cup \mathcal{C}_k$ , and, (iii) the edges of cycle  $\mathcal{C}_i$ ,  $i = 1, 2, \dots, k$  alternate between edges of  $\mathcal{P}$  and  $\mathcal{M}$ . Note that, properties (i) and (ii) imply that the cycle collection will contain half of  $\mathcal{P}$ 's edges (in particular  $\lceil |E(\mathcal{P})|/2 \rceil$  edges) and all of  $\mathcal{M}$ 's edges. In our drawing, these edges will not cross with each other. The remaining edges of  $\mathcal{P}$  will introduce only right angle crossings with the edges of  $\mathcal{M}$ .

Let  $\mathcal{P}_{odd}$  be a subgraph of  $\mathcal{P}$  which contains each second edge of  $\mathcal{P}$ , starting from its first edge, i.e.,  $E(\mathcal{P}_{odd}) = \{(v_i, v_{i+1}); 1 \leq i < n, i \text{ is odd}\}$ . In Figure 4.2, the edges of  $\mathcal{P}_{odd}$  are drawn solid. Clearly,  $\mathcal{P}_{odd}$  is a matching. Since we have assumed that  $n$  is even,  $\mathcal{P}_{odd}$  contains exactly  $n/2$  edges. Hence,  $|E(\mathcal{P}_{odd})| = |E(\mathcal{M})|$ . In addition,  $\mathcal{P}_{odd}$  covers all vertices of  $\mathcal{P}$ , and,  $E(\mathcal{P}_{odd}) \cap E(\mathcal{M}) = \emptyset$ . The latter equation trivially follows from our initial hypothesis, which states that  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ . We conclude that  $\mathcal{P}_{odd} \cup \mathcal{M}$  is a 2-regular graph. Thus, each connected component of  $\mathcal{P}_{odd} \cup \mathcal{M}$  corresponds to a cycle of even length, which alternates between edges of  $\mathcal{P}_{odd}$  and  $\mathcal{M}$ . This is the cycle collection mentioned above (see Figure 4.3).

Initially, we fix the  $x$ -coordinate of each vertex<sup>2</sup> of  $\mathcal{P}$  by setting  $x(v_i) = i$ ,  $1 \leq i \leq n$ . This ensures that  $\mathcal{P}$  is  $x$ -monotone and hence planar. Later on, we will slightly change the

<sup>2</sup>Note that, the algorithm can be adjusted so that the  $x$  and  $y$  coordinates of each vertex are computed at the same time (without affecting neither the correctness of the algorithm nor its running time). We have chosen to compute them separately in order to simplify the presentation.



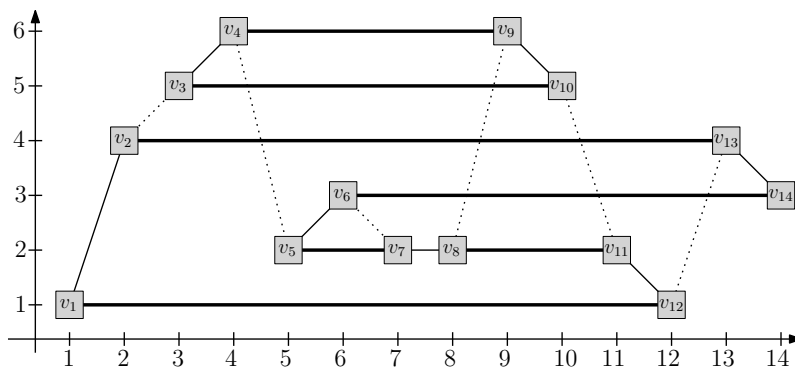
**Figure 4.3:**  $\mathcal{P}_{odd} \cup \mathcal{M}$  (of Figure 4.2) consists of cycles  $\mathcal{C}_1$  and  $\mathcal{C}_2$ . The edges of  $\mathcal{P}_{odd}$  are drawn solid, while the edges of  $\mathcal{M}$  are drawn bold.

$x$ -coordinate of some vertices of  $\mathcal{P}$  without affecting  $\mathcal{P}$ 's monotonicity. The  $y$ -coordinate of each vertex of  $\mathcal{P}$  is established by considering the cycles of  $\mathcal{P}_{odd} \cup \mathcal{M}$ .

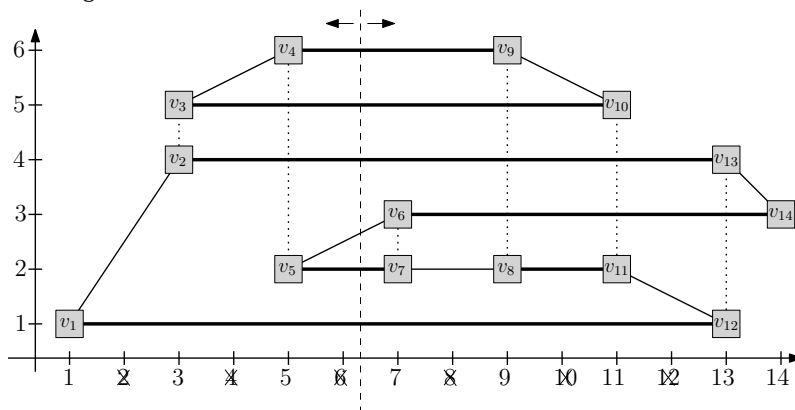
We draw each of these cycles in turn. More precisely, assume that zero or more cycles have been completely drawn and let  $\mathcal{C}$  be the cycle in the cycle collection which contains the leftmost vertex, say  $v_i$ , of  $\mathcal{P}$  that has not been drawn yet (initially,  $v_i$  is identified by  $v_1$ ). Then, vertex  $v_i$  should be an odd-indexed vertex and thus  $(v_i, v_{i+1})$  belongs to  $\mathcal{C}$ . Orient cycle  $\mathcal{C}$  so that vertex  $v_i$  is the first vertex of cycle  $\mathcal{C}$  and  $v_{i+1}$  is the last (see Figure 4.3). Based on this orientation, we will draw the edges of  $\mathcal{C}$  in a snake-like fashion, starting from vertex  $v_i$  and reaching vertex  $v_{i+1}$  last. The first edge to be drawn is incident to vertex  $v_i$  and belongs to  $\mathcal{M}$ . We draw it as a horizontal line-segment at the bottommost available layer in the produced drawing (initially,  $L_1 : y = 1$ ). Since cycle  $\mathcal{C}$  alternates between edges of  $\mathcal{P}_{odd}$  and  $\mathcal{M}$ , the next edge to be drawn belongs to  $\mathcal{P}_{odd}$  followed by an edge of  $\mathcal{M}$ . If we can draw both of them in the current layer without introducing edge overlaps, we do so. Otherwise, we employ an additional layer, i.e., the edge of  $\mathcal{P}_{odd}$  is drawn oblique starting from the current layer towards the next one and the edge of  $\mathcal{M}$  is drawn horizontally at the new layer. We continue in the same manner, until edge  $(v_i, v_{i+1})$  is reached in the traversal of cycle  $\mathcal{C}$ . This edge connects two consecutive vertices of  $\mathcal{P}$  that are the leftmost in the drawing of  $\mathcal{C}$ . Therefore, edge  $(v_i, v_{i+1})$  can be added in the drawing of  $\mathcal{C}$  without introducing any crossings. Thus, cycle  $\mathcal{C}$  is drawn planar.

So far, we have drawn all edges of  $\mathcal{M}$  and half of the edges of  $\mathcal{P}$  (i.e.,  $\mathcal{P}_{odd}$ ) and we have obtained a planar drawing in which all edges of  $\mathcal{M}$  are drawn as horizontal, non-overlapping line segments. In the worst case, this drawing occupies  $n/2$  layers.

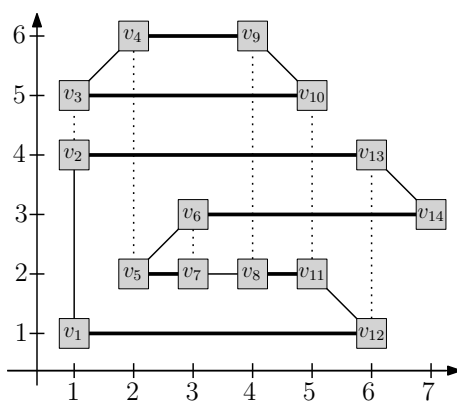
We proceed to incorporate the remaining edges of  $\mathcal{P}$ , i.e., the ones that belong to  $\mathcal{P} -$



(i) A drawing obtained by incorporating the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  into the drawing of Figure 4.3.



(ii) A drawing obtained by moving the even-indexed vertices of  $\mathcal{P}$  in the drawing of Figure 4.4i one unit to the right.



(iii) A compact GRacSim drawing.

**Figure 4.4:** In the drawings the edges of  $\mathcal{P}_{odd}$  are drawn solid, while the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  dotted. The edges of  $\mathcal{M}$  are drawn bold.

$\mathcal{P}_{odd}$ , into the drawing (refer to the dotted drawn edges of Figure 4.4i). Since  $x(v_i) = i$ ,  $i = 1, 2, \dots, n$ , the edges of  $\mathcal{P}$  do not cross with each other and therefore,  $\mathcal{P}$  is drawn planar. In contrast, an edge of  $\mathcal{P} - \mathcal{P}_{odd}$  may cross multiple edges of  $\mathcal{M}$ , and, these crossings do not form right angles (see Figure 4.4i). In order to fix these crossings, we move each even-indexed vertex of  $\mathcal{P}$  one unit to the right (keeping its  $y$ -coordinate unchanged), except for the last vertex of  $\mathcal{P}$ . Then, the endpoints of the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  have exactly the same  $x$ -coordinate and cross at right angles the edges of  $\mathcal{M}$ , which are drawn as horizontal line-segments. The path remains  $x$ -monotone (but not strictly anymore) and hence, planar. In addition, it is not possible to introduce vertex overlaps, since in the produced drawing each edge of  $\mathcal{M}$  has at least two units length (recall that  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ ). Since the vertices of the drawing do not occupy even  $x$ -coordinates, the width of the drawing can be reduced from  $n$  to  $n/2 + 1$  (see Figure 4.4ii). We can further reduce the width of the produced drawing by merging consecutive columns that do not interfere in  $y$ -direction into a common column (see Figure 4.4iii). However, this post-processing does not result into a drawing of asymptotically smaller area. This completes the description of our algorithm for the case where  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ .

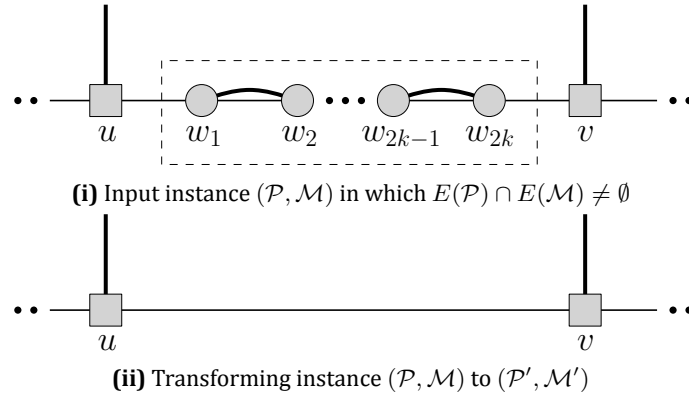
**Theorem 4.2.1.** *A path  $\mathcal{P}$  and a perfect matching  $\mathcal{M}$  on the same vertex set and such that  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$  always admit a GRacSim drawing on an  $(n/2 + 1) \times n/2$  integer grid, where  $n$  is the size of the vertex set. Moreover, the drawing can be computed in linear time.*

*Proof.* Finding the cycles of  $\mathcal{P}_{odd} \cup \mathcal{M}$  can be done in  $O(n)$  time, where  $n$  is the number of vertices of  $\mathcal{P}$ ; we identify the leftmost vertex of each cycle and then we traverse it. Having computed the cycle collection of  $\mathcal{P}_{odd} \cup \mathcal{M}$ , the coordinates of the vertices are computed in  $O(n)$  total time by a traversal of the cycle.  $\square$

For the case where  $\mathcal{P}$  and  $\mathcal{M}$  share edges (i.e.,  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$ ), our intention is to construct a drawing on an  $n \times n/2$  integer grid by extending the algorithm that supports Theorem 4.2.1. To achieve this, we utilize the fact that the resulting drawings for the case where  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$  are *stretchable* in the following sense: If we draw any vertical non-grid line that crosses part of the drawing (refer to the dashed drawn line of Figure 4.4ii), then we can shift to the left (right, resp.) the drawing that is to the left (right, resp.) of this line without affecting either the planarity of  $\mathcal{P}$  and  $\mathcal{M}$  or the angles in which  $\mathcal{P}$  and  $\mathcal{M}$  cross (since crossings always appear at grid points; horizontal stretching). Similarly, one can vertically stretch the drawing by employing a horizontal non-grid line.

We initially assume that the first and the last edge of  $\mathcal{P}$  do not appear in  $\mathcal{M}$ <sup>3</sup>, i.e., edges that are in both  $\mathcal{P}$  and  $\mathcal{M}$  are interior edges of  $\mathcal{P}$ . Let  $V_{com}(\mathcal{P})$  ( $V_{dis}(\mathcal{P})$ , resp.) be the set of

<sup>3</sup>Later on this section, we will describe how to cope with the degenerated cases where either the first or the last edge of  $\mathcal{P}$  appear in  $\mathcal{M}$ .



**Figure 4.5:** In the drawings the edges of  $\mathcal{P}$  are drawn plain, while the edges of  $\mathcal{M}$  are drawn bold. Vertices of  $V_{dis}(\mathcal{P})$  ( $V_{com}(\mathcal{P})$ , resp.) are drawn as squares (circles, resp.)

vertices of  $\mathcal{P}$  which are (are not, resp.) incident to an edge that belongs to both  $\mathcal{P}$  and  $\mathcal{M}$  (see Figure 4.5). More formally,  $V_{com}(\mathcal{P}) = \{u \in V(\mathcal{P}) : \exists v \in V(\mathcal{P}) \text{ s.t. } (u, v) \in E(\mathcal{P}) \cap E(\mathcal{M})\}$  and  $V_{dis}(\mathcal{P}) = V(\mathcal{P}) - V_{com}(\mathcal{P})$ . Since we have assumed that neither the first nor the last edge of  $\mathcal{P}$  appear in  $\mathcal{M}$ ,  $V_{com}(\mathcal{P}) \subset V(\mathcal{P})$ . Similarly, we define  $V_{com}(\mathcal{M})$  and  $V_{dis}(\mathcal{M})$ . Obviously,  $V_{com}(\mathcal{P}) = V_{com}(\mathcal{M})$ . Since  $\mathcal{P}$  and  $\mathcal{M}$  are defined on the same vertex set, it follows that  $V_{dis}(\mathcal{P}) = V_{dis}(\mathcal{M})$ .

If there exist edges that belong to both  $\mathcal{P}$  and  $\mathcal{M}$ , we momentarily remove them from both  $\mathcal{P}$  and  $\mathcal{M}$  as follows: If  $u \rightarrow w_1 \rightarrow \dots \rightarrow w_{2k} \rightarrow v$  is a sub-path of vertices of  $\mathcal{P}$  such that  $u, v \in V_{dis}(\mathcal{P})$  and  $w_i \in V_{com}(\mathcal{P})$ ,  $i = 1, 2, \dots, 2k$  (see Figure 4.5i), we replace it by a single edge  $(u, v)$  of  $\mathcal{P}$  (see Figure 4.5ii). This will result into a new path  $\mathcal{P}'$  of  $n'$  vertices and a new matching  $\mathcal{M}'$  with the following properties:

- i)  $V(\mathcal{P}') = V_{dis}(\mathcal{P})$
- ii)  $V(\mathcal{M}') = V_{dis}(\mathcal{M})$
- iii)  $E(\mathcal{P}') \cap E(\mathcal{M}') = \emptyset$
- iv)  $n'$  is even
- v)  $|E(\mathcal{M}')| = n'/2$

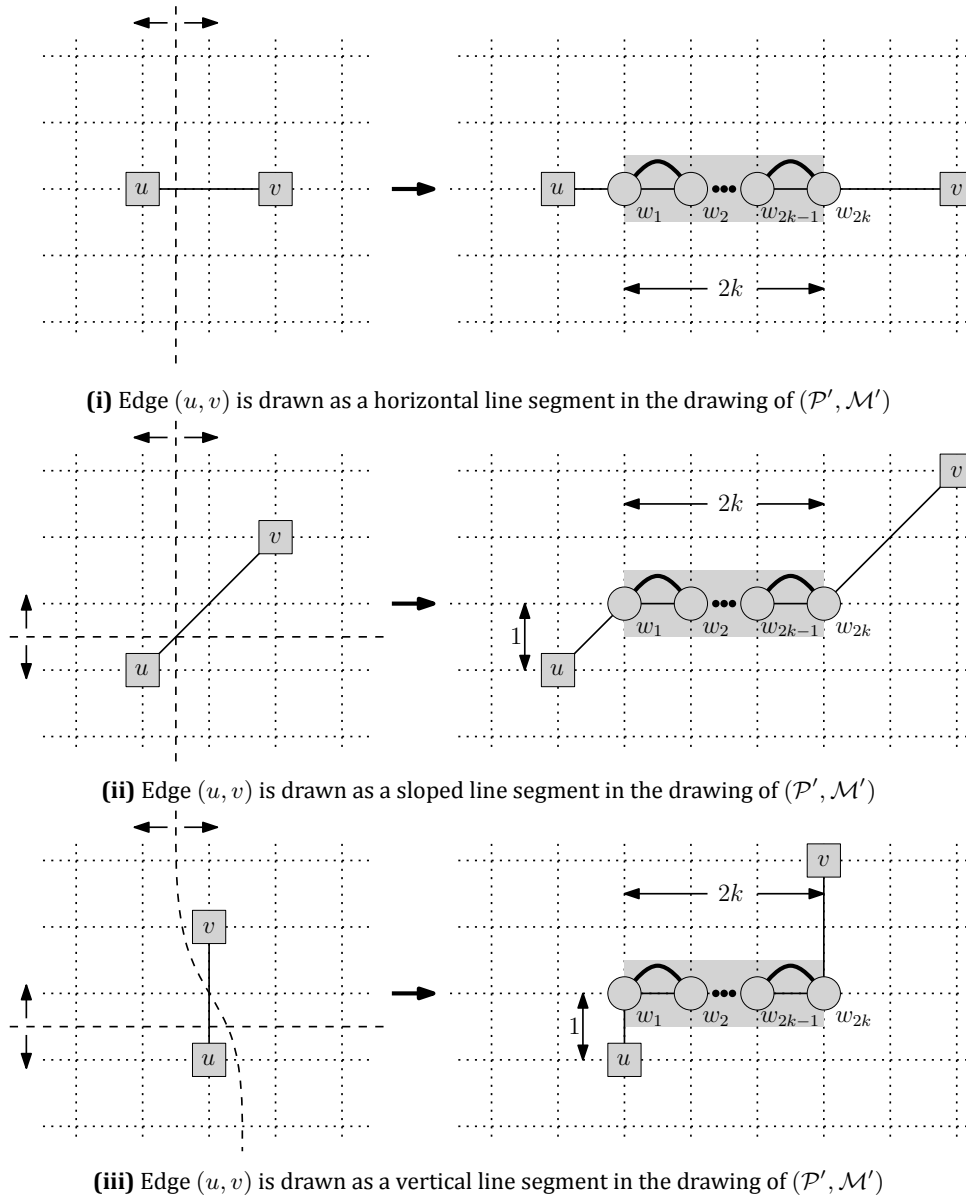
Hence,  $\mathcal{P}'$  and  $\mathcal{M}'$  can be drawn simultaneously due to Theorem 4.2.1. The width (height, resp.) of the produced drawing equals to  $|V_{dis}(\mathcal{P})|/2 + 1$  ( $|V_{dis}(\mathcal{M})|/2$ , resp.) In order to incorporate the removed vertices and edges in the produced drawing, we utilize the fact that the resulting drawing is stretchable.

Let  $u \rightarrow w_1 \rightarrow \dots \rightarrow w_{2k} \rightarrow v$  be a sub-path of degree-2 vertices of  $\mathcal{P}$  which was contracted into a single edge  $(u, v) \in E(\mathcal{P}')$ , when transforming instance  $(\mathcal{P}, \mathcal{M})$  to  $(\mathcal{P}', \mathcal{M}')$ . We distinguish the following cases:

- *Edge  $(u, v)$  is drawn as a horizontal line segment in the drawing of  $(\mathcal{P}', \mathcal{M}')$* : This case is illustrated in Figure 4.6i, in which we have assumed that in the drawing of  $\mathcal{P}'$  and  $\mathcal{M}'$  it holds that  $x(u) < x(v)$ . In this case, we apply a horizontal stretching in order to allocate  $2k$  units of length directly next to  $u$ . Then, vertices  $w_1, w_2, \dots, w_{2k}$  are drawn at consecutive  $x$ -coordinates along the line  $y = y(u)$  starting from  $x = x(u) + 1$  (i.e.,  $x(w_i) = x(u) + i + 1, i = 1, 2, \dots, 2k$ ).
- *Edge  $(u, v)$  is drawn as a sloped (neither vertical nor horizontal) line segment in the drawing of  $(\mathcal{P}', \mathcal{M}')$* : This case is illustrated in Figure 4.6ii, in which we have assumed that in the drawing of  $\mathcal{P}'$  and  $\mathcal{M}'$  it holds that  $x(u) < x(v)$  and  $y(u) < y(v)$ . In this case, we first apply a vertical stretching in order to allocate one unit of length directly above  $u$  followed by a horizontal stretching, in order to allocate  $2k$  units of length directly next to  $u$ . In the resulting drawing, vertices  $w_1, w_2, \dots, w_{2k}$  are drawn at consecutive  $x$ -coordinates along the line  $y = y(u) + 1$  starting from  $x = x(u) + 1$  (i.e.,  $x(w_i) = x(u) + i + 1, i = 1, 2, \dots, 2k$ ).
- *Edge  $(u, v)$  is drawn as a vertical line segment in the drawing of  $(\mathcal{P}', \mathcal{M}')$* : This case is illustrated in Figure 4.6i and is quite similar to the previous one. Again, without loss of generality, we have assume that in the drawing of  $\mathcal{P}'$  and  $\mathcal{M}'$  it holds that  $y(u) < y(v)$ . In this case, we first apply a vertical stretching in order to allocate one unit of length directly above  $u$ . Then, we apply a transformation similar to a horizontal stretching, in order to allocate  $2k$  units of length between  $u$  and  $v$  (see Figure 4.6iii). More formally, in order to achieve the transformation illustrated in Figure 4.6iii, we use a vertical line segment that coincides with edge  $(u, v)$ , and, we shift the top endpoint of edge  $(u, v)$ , while keeping its bottom endpoint in place. In the resulting drawing, vertices  $w_1, w_2, \dots, w_{2k}$  are drawn at consecutive  $x$ -coordinates along the line  $y = y(u) + 1$  starting from  $x = x(u)$  (i.e.,  $x(w_i) = x(u) + i, i = 1, 2, \dots, 2k$ ).

From the above, it follows that when we reinsert the  $2k$  vertices of a contracted sub-path of degree-2 vertices of  $\mathcal{P}$  the width of the drawing gets larger by  $2k$  units of length. So in total, the width of the drawing is at most  $n$ . On the other hand, all matching edges are drawn as horizontal line segments. In worst case, no two matching edges share the same horizontal grid line. So in total, the height of the drawing is at most  $n/2$ .

Recall that in order to cope with the case where  $\mathcal{P}$  and  $\mathcal{M}$  share edges, we had initially assumed that neither the first nor the last edge of  $\mathcal{P}$  appear in  $\mathcal{M}$ . The reason for this as-

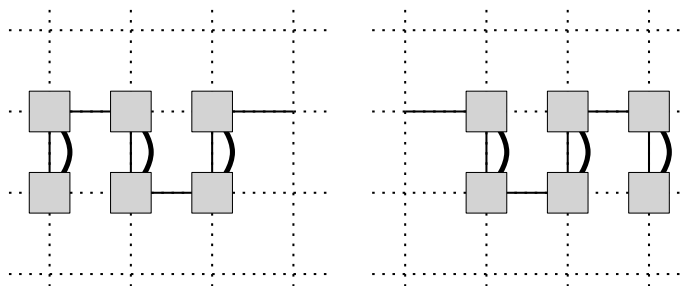


**Figure 4.6:** Different cases that occur when reinserting a contracted sub-path  $u \rightarrow w_1 \rightarrow \dots \rightarrow w_{2k} \rightarrow v$  of degree-2 vertices of  $\mathcal{P}$ . In the drawings the edges of the path are drawn plain, while the edges of the matching are drawn bold.

sumption is that the transformations that we have described so far (see Figure 4.6) require sub-paths that start from a vertex of  $V_{dis}(\mathcal{P})$  and (through an even number of vertices of  $V_{com}(\mathcal{P})$ ) end to another vertex of  $V_{dis}(\mathcal{P})$  again. If, for example, the first vertex of  $\mathcal{P}$  belongs to  $V_{com}(\mathcal{P})$  (i.e., the first edge of  $\mathcal{P}$  appears in  $\mathcal{M}$ ), then there might exist a whole sub-path of

vertices of  $V_{com}(\mathcal{P})$  at the beginning of  $\mathcal{P}$ <sup>4</sup>. However, such a sub-path is not supported by the transformations that we have described so far, since it does not start from a vertex of  $V_{dis}(\mathcal{P})$ .

If there exists a sub-path of vertices of  $V_{com}(\mathcal{P})$  at the beginning or at the end of  $\mathcal{P}$ , we momentarily remove it from  $\mathcal{P} \cup \mathcal{M}$  and draw it in a snake-like fashion as illustrated in Figure 4.7. The remaining part of  $\mathcal{P} \cup \mathcal{M}$  is either the empty graph or a graph with the property that neither the first nor the last edge of the path appear in the matching, which can be drawn with the developed algorithm. In the former case, the resulting drawing is a snake-like drawing of  $\mathcal{P}$  and  $\mathcal{M}$ . In the latter case, we simply plug the snake-like drawings of the removed parts to the first and last vertices of the drawing of the remaining part of  $\mathcal{P} \cup \mathcal{M}$ , which are drawn leftmost and rightmost, respectively. The height of the resulting drawing gets larger by at most two units of length, while its width is at most the number of vertices of the path. This ensures that the total area of the drawing is not affected. The following theorem summarizes our result.



**Figure 4.7:** Snake-like drawings in the case where the matching pairs each second tuple of vertices of the path.

**Theorem 4.2.2.** *A path  $\mathcal{P}$  and a perfect matching  $\mathcal{M}$  on the same vertex set always admit a GRacSim drawing on an  $n \times n/2$  integer grid, where  $n$  is the size of the vertex set. Moreover, the drawing can be computed in linear time.*

We extend the algorithm that produces a GRacSim drawing of a path and a matching to also cover the case of a cycle  $\mathcal{C}$  and a matching  $\mathcal{M}$ . Obviously, if we remove an edge from the input cycle (preferably one that belongs to  $E(\mathcal{C}) - E(\mathcal{M})$ ), the remaining graph is a path  $\mathcal{P}$  (see Figure 4.8). Then, we apply the developed algorithm and obtain a GRacSim drawing of  $\mathcal{P}$  and  $\mathcal{M}$ .

We describe in detail the case where  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ . The case where  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$  is treated similarly. By Theorem 4.2.1, the drawing fits in a  $(n/2 + 1) \times n/2$  integer grid. Additionally, the first vertex of  $\mathcal{P}$  is drawn at the bottommost layer (hence its incident edge in

<sup>4</sup>In worst case, all vertices of  $\mathcal{P}$  belong to  $V_{com}(\mathcal{P})$ , i.e.,  $\mathcal{M}$  matches each second pair of vertices of  $\mathcal{P}$ .



$\mathcal{M}$  is not crossed), and the last vertex of  $\mathcal{P}$  is drawn rightmost. With these two properties, we can add the removed edge between the first and the last vertex of  $\mathcal{P}$  without introducing new crossings. To achieve this, we move the first vertex of  $\mathcal{P}$  at most  $n/2 + 2$  units downwards (keeping its  $x$ -coordinate unchanged) and the last vertex of  $\mathcal{P}$  at most  $n/2 + 1$  units rightwards (keeping its  $y$ -coordinate unchanged). Then, the insertion in the drawing of the edge that closes the cycle does not introduce any crossings, as desired. The following theorem summarizes our result.

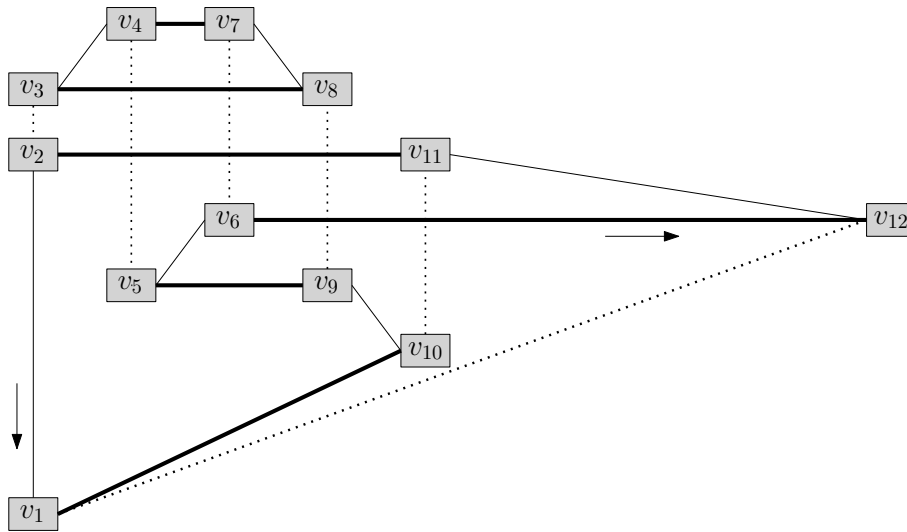


Figure 4.8: A GRacSim drawing of a cycle and a matching.

**Theorem 4.2.3.** *A cycle  $C$  and a perfect matching  $\mathcal{M}$  on the same vertex set and such that  $E(C) \cap E(\mathcal{M}) = \emptyset$  always admit a GRacSim drawing on an  $(n + 2) \times (n + 2)$  integer grid, where  $n$  is the size of the vertex set. Moreover, the drawing can be computed in linear time.*

As already stated, the case where  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$  is treated similarly. Since by Theorem 4.2.2 we have to reinsert the deleted edge in a drawing of size  $n \times n/2$ , the resulting drawing will be of size  $3n/2 \times 3n/2$ . Hence, we can state the following theorem.

**Theorem 4.2.4.** *A cycle  $C$  and a perfect matching  $\mathcal{M}$  on the same vertex set always admit a GRacSim drawing on an  $3n/2 \times 3n/2$  integer grid, where  $n$  is the size of the vertex set. Moreover, the drawing can be computed in linear time.*

### 4.2.1 Algorithm Extensions

According to the formal definition of the GRacSim drawing problem, the input graphs share the same vertex set. This immediately implies that  $\mathcal{M}$  is a perfect matching and  $n$  is

even. However, our algorithm can be adjusted to support the case where the input graphs are not necessarily on the same vertex set.

Consider the case where the input consists of a path  $\mathcal{P}$  and a matching  $\mathcal{M}$ , which have to be drawn simultaneously and assume without loss of generality that the union of  $\mathcal{P}$  and  $\mathcal{M}$  is a connected graph<sup>5</sup>. Let  $V$  be the union of  $V(\mathcal{P})$  and  $V(\mathcal{M})$  and denote by  $n$  the size of  $V$ . Without loss of generality, we assume that  $n$  is even<sup>6</sup>. In a first step, we augment  $\mathcal{P}$  so that it spans all vertices of  $V$ . Let  $\mathcal{P}_{aug}$  be the resulting path. Since  $n$  is even, we can augment  $\mathcal{M}$  so that it spans all vertices of  $V$ , as well. Denote by  $\mathcal{M}_{aug}$  the augmented matching. Obviously,  $\mathcal{P}_{aug}$  and  $\mathcal{M}_{aug}$  are defined on the same vertex set, which is of even size, and  $\mathcal{M}_{aug}$  is a perfect matching. Hence,  $\mathcal{P}_{aug}$  and  $\mathcal{M}_{aug}$  can be drawn simultaneously by the algorithm supporting Theorem 4.2.2. In order to obtain a GRacSim of  $\mathcal{P}$  and  $\mathcal{M}$ , it is enough to remove from the GRacSim drawing of  $\mathcal{P}_{aug}$  and  $\mathcal{M}_{aug}$  the extra edges that were used in order to augment  $\mathcal{P}$  and  $\mathcal{M}$  to  $\mathcal{P}_{aug}$  and  $\mathcal{M}_{aug}$ , respectively. The following theorem summarizes our result.

**Theorem 4.2.5.** *A path  $\mathcal{P}$  and a matching  $\mathcal{M}$  always admit a GRacSim drawing on an  $n \times n/2$  integer grid, where  $n$  is the size of the union of  $V(\mathcal{P})$  and  $V(\mathcal{M})$ . Moreover, the drawing can be computed in linear time.*

In the case where the input consists of a cycle  $\mathcal{C}$  and a matching  $\mathcal{M}$ , we follow a similar but slightly different approach, since  $\mathcal{C}$  cannot be further augmented. Again, we assume without loss of generality that the union of  $\mathcal{C}$  and  $\mathcal{M}$  is a connected graph. We proceed as follows. We monotonically remove the vertices that belong exclusively to  $\mathcal{M}$  (i.e., their incident edges belong exclusively to  $\mathcal{M}$ ) and their incident edges in  $\mathcal{M}$ . Then, we proceed to augment  $\mathcal{M}$ , so that it spans all vertices of  $\mathcal{C}$  (and obtain a new matching  $\mathcal{M}_{aug}$ , as previously). Observe that this is feasible, only when  $|V(\mathcal{C})|$  is even. If  $|V(\mathcal{C})|$  is odd, then unavoidably there exists a vertex of  $\mathcal{C}$  which is not covered by  $\mathcal{M}_{aug}$ . Say without loss of generality that this is the case. In order to conform with the assumptions of the algorithm supporting Theorem 4.2.4, we momentarily remove this particular vertex from  $\mathcal{C}$ , by connecting its two incident vertices in  $\mathcal{C}$  by an edge. The resulting cycle, say  $\mathcal{C}_{dec}$ , and  $\mathcal{M}_{aug}$  can be drawn simultaneously by means of the algorithm supporting Theorem 4.2.4, since they are defined on the same vertex set and  $\mathcal{M}_{aug}$  is a perfect matching.

In order to obtain a GRacSim of  $\mathcal{C}$  and  $\mathcal{M}$ , we first remove from the GRacSim drawing of  $\mathcal{C}_{dec}$  and  $\mathcal{M}_{aug}$  the extra edges that were used in order to augment  $\mathcal{M}$  to  $\mathcal{M}_{aug}$ . Then, we proceed to incorporate into the resulting drawing the vertex of  $\mathcal{C}$  that was previously removed, when transforming the odd-length cycle  $\mathcal{C}$  to the even-length cycle  $\mathcal{C}_{dec}$ . To achieve this, a horizontal stretching similar to the ones described in the proof of Theorem 4.2.2 is enough.

<sup>5</sup>If this is not the case, our algorithm treats each connected component separately.

<sup>6</sup>If  $n$  is odd, we add an isolated vertex in  $V$ . Hence, its size become even.

However, since both the width and the height of the drawing might become larger by one unit of length, special attention should be paid at the bottommost drawn edge of the cycle (i.e., the one which is drawn neither as a horizontal nor as a vertical line segment; see Figure 4.8). More precisely, potential crossings posed by this particular edge can be resolved by shifting the bottommost (rightmost, resp.) vertex of the drawing one unit downwards (rightwards, resp.).

Now it remains to explain how to incorporate into the resulting drawing the vertices that belong exclusively to  $\mathcal{M}$  and their corresponding edges of the matching (which we had removed at the beginning of this procedure). Since we have assumed that the union of  $\mathcal{C}$  and  $\mathcal{M}$  is a connected graph, each of these vertices should be incident to a vertex of  $\mathcal{C}$  (through an edge of  $\mathcal{M}$ ). Let  $v \in V(\mathcal{C})$  be such a vertex of  $\mathcal{C}$ . Obviously, in the drawing constructed so far  $v$  is not incident to a matching edge, which implies that either its left or right “port” is free. Hence, we can incorporate its incident matching edge into the drawing constructed so far, as a horizontal line segment of unit length incident to  $v$ ’s free port, through a horizontal stretching directly next to  $v$ ’s free port. Again, potential crossings posed by the presence of the bottommost drawn edge of the cycle can be resolved by shifting the bottommost (rightmost, resp.) vertex of the drawing one unit downwards (rightwards, resp.), each time a new vertex among those belonging exclusive to  $\mathcal{M}$  is incorporated. Note that, this does not affect the total area occupied by the drawing, which is still  $3n/2 \times 3n/2$ , where  $n$  is the size of the union of  $V(\mathcal{C})$  and  $V(\mathcal{M})$ . The following theorem summarizes our result.

**Theorem 4.2.6.** *A cycle  $\mathcal{C}$  and a matching  $\mathcal{M}$  always admit a GRacSim drawing on an  $3n/2 \times 3n/2$  integer grid, where  $n$  is the size of the union of  $V(\mathcal{C})$  and  $V(\mathcal{M})$ . Moreover, the drawing can be computed in linear time.*

From the above, it follows that we can identify a class of RAC graphs.

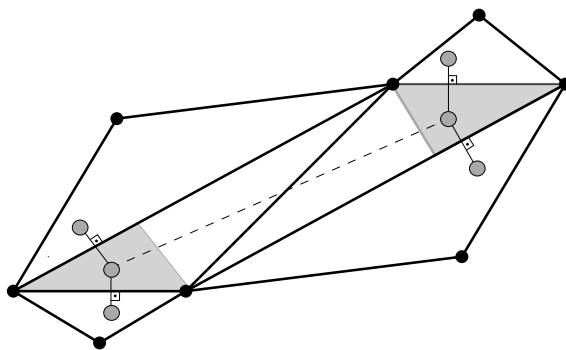
**Corollary 4.2.1.** *Let  $G$  be a simple connected graph that can be decomposed into a matching and either a path or a cycle. Then,  $G$  is a RAC graph.*

### 4.3 A Planar Graph and its Dual: An Interesting Variation

In this section, we examine the GDual-GRacSim drawing problem. This problem can be considered as a variation of the GRacSim drawing problem, where the first graph (i.e., the planar graph) determines the second one (i.e., the dual) and places restrictions on its layout. Recall that according to the GDual-GRacSim drawing problem, we are given a planar embedded graph  $G$  and the main task is to determine a geometric drawing of  $G$  and its weak dual  $G^*$  such that: (i)  $G$  and  $G^*$  are drawn planar, (ii) each vertex of the dual is drawn inside its corresponding face of  $G$  and, (iii) the primal-dual edge crossings form right angles.

As already stated in Section 2.4.2, Brightwell and Scheinermann [BS93] proved that the GDual-GRacSim problem always admits a solution if the input graph is a triconnected planar graph. For the general case of planar graphs, we demonstrate by an example that it is not always possible to compute such a drawing, and thus, we concentrate our study in the case of outerplanar graphs.

Initially, we consider the case where the planar drawing  $\Gamma(G)$  of graph  $G$  is specified as part of the input and it is required that it remains unchanged in the output. We demonstrate by an example that it is not always feasible to incorporate  $G^*$  into drawing  $\Gamma(G)$  and obtain a GDual-GRacSim drawing of  $G$  and  $G^*$ . The example is illustrated in Figure 4.9. In the following, we prove that if the input graph is a planar embedded graph, then the GDual-GRacSim drawing problem does not always admit a solution, as well.

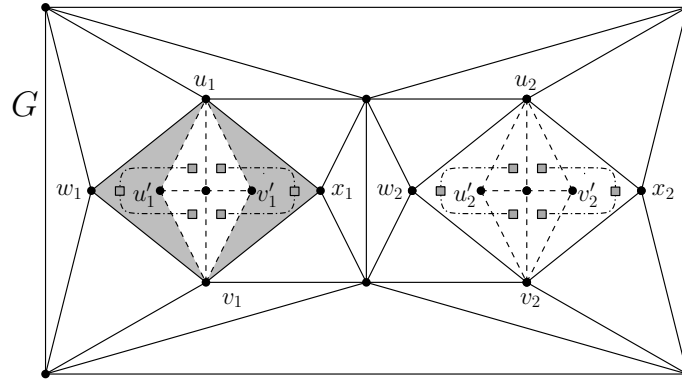


**Figure 4.9:** The input planar drawing of the primal graph  $G$  is sketched with black colored vertices and bold edges and should remain unchanged in the output. The vertices of the dual  $G^*$  are colored gray. Then, the dual's dashed drawn edge will inevitably introduce a non right angle crossing.

**Theorem 4.3.1.** *There exists a planar graph  $G$  with the following property: For none of the planar embeddings of  $G$  a GDual-GRacSim drawing of  $G$  and its weak dual  $G^*$  is possible.*

*Proof.* The graph  $G$  used to establish the theorem is depicted in Figure 4.10, where the vertices drawn as boxes belong to the dual graph  $G^*$ . Observe that if we replace the two subgraphs drawn with dashed edges by two edges, the resulting graph is a triconnected planar graph, which has unique planar drawing up to the choice of the outerface, translations, rotations and stretchings. This implies that in any planar drawing of  $G$ , either  $u_1w_1v_1x_1$  or  $u_2w_2v_2x_2$  is an internal face. Without loss of generality, we consider the case where  $u_1w_1v_1x_1$  is an internal face. Now, observe that the dual graph should have two vertices within each of the gray-colored faces of Figure 4.10 (refer to the vertices which are drawn as boxes). Each of these two vertices is incident to two vertices of the dual that lie within the triangular faces of

the dashed drawn subgraph of  $G$ , incident to the two gray-colored faces. Observe that in any RAC drawing of  $G$  and  $G^*$  both quadrilaterals  $u_1w_1v_1u'_1$  and  $u_1x_1v_1v'_1$  must be convex, which is impossible.



**Figure 4.10:** An example of a planar graph  $G$ , for which the GDual-GRacSim does not admit a solution. The problematic faces are drawn in gray.

□

**Corollary 4.3.1.** *There exists an infinite class of planar graphs  $\mathcal{F}$  with the following property: For any graph  $G \in \mathcal{F}$  and any planar embedding  $\mathcal{E}(G)$  of  $G$ , a GDual-GRacSim drawing of  $G$  and its weak dual  $G^*$  is not possible.*

We now proceed to study the GDual-GRacSim drawing problem in the case where the input graph is outerplanar. Notice that an outerplanar graph is not necessarily triconnected (i.e., by deleting any pair of vertices on the same interior face and non-consecutive on the outerface, the graph is disconnected). On the other hand, one can augment an outerplanar graph to a triconnected planar (but not necessarily outerplanar) graph, by introducing an additional vertex incident to all vertices of the outerplanar graph. This easily implies that an outerplanar embedded graph and its dual always admit a GDual-GRacSim drawing, due Brightwell and Scheinermann [BS93]. However, their constructive approach simply proves that such a drawing exists and cannot be utilized to construct the corresponding drawing, since the computation involves sequences that do not necessarily converge after a finite number steps.

**Theorem 4.3.2.** *Given an outerplane embedding of an outerplanar graph  $G$ , it is always possible to determine a GDual-GRacSim drawing of  $G$  and its weak dual  $G^*$ .*

*Proof.* We employ a recursive geometric construction which computes a GDual-GRacSim drawing of  $G$  and its dual  $G^*$ . Consider an arbitrary edge  $(u, v)$  of the outerplanar graph that does not belong to its external face and let  $f$  and  $g$  be the faces to its left and the right side,

respectively, as we move along  $(u, v)$  from vertex  $u$  to vertex  $v$ . Then,  $(f, g)$  is an edge of the dual graph  $G^*$ . Since the dual of an outerplanar graph is a tree, the removal of edge  $(f, g)$  results in two trees  $T_f$  and  $T_g$  that can be considered to be rooted at vertices  $f$  and  $g$  of  $G^*$ , respectively. For the recursive step of our drawing algorithm, we assume that we have already produced a GDual-GRacSim drawing for  $T_f$  and its corresponding subgraph of  $G$  that satisfies the following invariant properties:

- I-P1: *Edge  $(u, v)$  is drawn on the external face of the GDual-GSimRAC drawing of  $T_f$ . Let  $u$  and  $v$  be drawn at points  $p_u$  and  $p_v$ , respectively. Denote by  $\ell_{u,v}$  the line defined by  $p_u$  and  $p_v$ .*
- I-P2: *Let the face-vertex  $f$  be drawn at point  $p_f$ . The perpendicular from point  $p_f$  to line  $\ell_{u,v}$  intersects the line segment  $p_u p_v$ . Let  $p$  be the point of intersection.*
- I-P3: *There exists two parallel semi-lines  $\ell_u$  and  $\ell_v$  passing from  $p_u$  and  $p_v$ , respectively, that define a semi-strip to the right of segment  $p_u p_v$  that does not intersect the drawing constructed so far. Denote this empty semi-strip by  $R_{u,v}$ .*

We proceed to describe how to recursively produce a drawing for tree  $T_g$  and its corresponding subgraph of  $G$  so that the overall drawing is a GDual-GRacSim drawing for  $G$  and its dual. Refer to Figure 4.11. Let  $p_g$  be a point in semi-strip  $R_{u,v}$  that also belongs to the perpendicular line to line-segment  $p_u p_v$  that passes from point  $p$ . Thus, the segment corresponding to the edge  $(f, g)$  of the dual crosses at right angle the segment corresponding to the edge  $(u, v)$  of  $G$ , as required. If  $g$  is a leaf (i.e., all edges of  $g$  except  $(u, v)$  are edges of the external face), we can draw the remaining edges of face  $g$  as a polyline of appropriate number of points that goes around  $p_g$  and connects  $p_u$  and  $p_v$ .

Consider now the more interesting case where  $g$  is not a leaf in the dual tree of  $G$ . In this case, we draw two circles, say  $C_g$  and  $C'_g$ , centered at  $p_g$  such that both lie entirely within the semi-strip  $R_{u,v}$  and do not touch neither line  $\ell_u$  nor line  $\ell_v$ . Assume that circle  $C'_g$  is the external of the two circles. From point  $p_u$  draw the tangent to circle  $C_g$  and let  $a$  be the point where it touches  $C_g$  and  $a'$  be the point to the right of  $a$  where the tangent intersects circle  $C'_g$  (see Figure 4.11). Similarly, we define points  $b$  and  $b'$  based on the tangent from point  $p_v$  to circle  $C_g$ .

Let  $k \geq 4$  be the number of vertices defining face  $g$ . The case where  $k = 3$  will be examined later. Draw  $k - 4$  points on the  $(a', b')$  arc, which is furthest from segment  $p_u p_v$ . These points, say  $\{p_i \mid 1 \leq i \leq k - 4\}$ , together with points  $p_u, p_v, a'$  and  $b'$  form face  $g$ . Observe that from point  $p_g$ , we can draw perpendicular lines towards each edge of the face. Indeed, line segments  $p_g a$  and  $p_g b$  are perpendicular to  $p_u a'$  and  $p_v b'$ , respectively. In addition, the

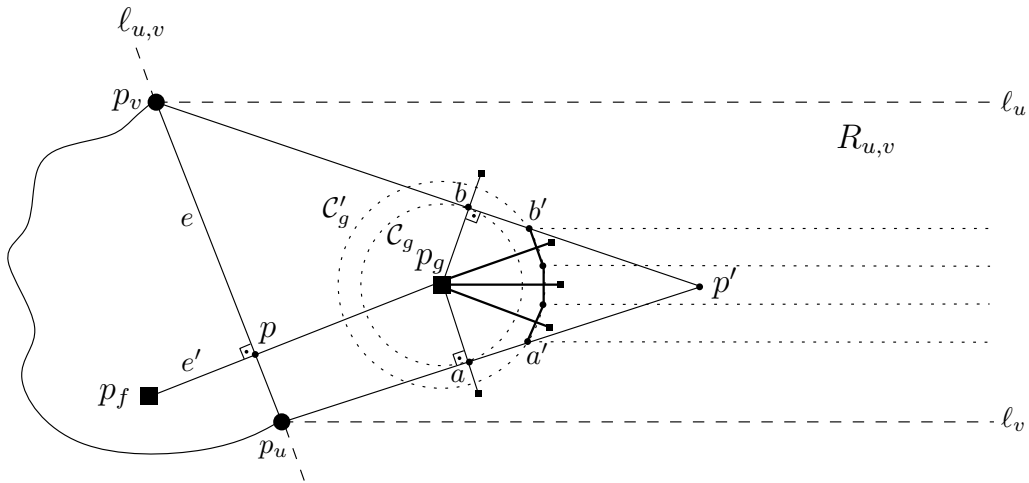


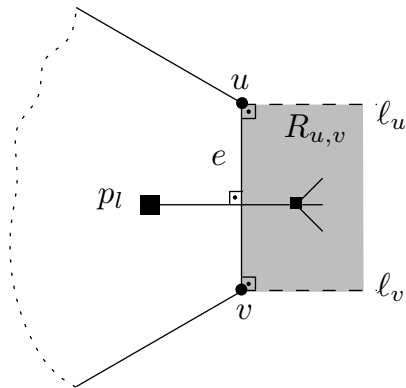
Figure 4.11: The recursion step of our algorithm.

remaining edges of the face are chords of circle  $C'_g$  and thus, we can always draw perpendicular lines to their midpoints from the center  $p_g$  of the circle. Now, from each of the newly inserted points of face  $g$  draw a semi-line that is parallel to semi-line  $\ell_u$  and lies entirely in the semi-strip  $R_{u,v}$ . We observe all invariant properties stated above hold for each child of face  $g$  in the subtree  $T_g$  of the dual of  $G$ . Thus, our algorithm can be applied recursively. The case where the number  $k$  of vertices defining face  $g$  is equal to 3 is treated as follows. We use the intersection of the two tangents, say  $p'$ , as the third point of the triangular face. We have to be careful so that  $p'$  lies inside the semi-strip. However, we can always select a point  $p_g$  close to segment  $p_u p_v$  and an appropriately small radius for circle  $C_g$ , so that  $p'$  is inside  $R_{u,v}$ .

Now that we have described the recursive step of the algorithm, it remains to define how the recursion begins (see Figure 4.12). We start from any face of  $G$  that is a leaf at its dual tree, say face  $l$ . We draw the face as regular polygon, with face-vertex  $l$  mapped at its center, say  $p_l$ . Let  $e = (u, v)$  be the only edge of the face that is internal to the outerplane embedding of  $G$ . Without loss of generality, assume that  $e$  is drawn vertically. Then, draw the horizontal semi-lines  $\ell_u$  and  $\ell_v$  from the endpoints of  $e$  in order to define the semi-strip  $R_{u,v}$ . From this point on, the algorithm can recursively draw the remaining faces of  $G$  and its dual  $G^*$ .

□

Note that, the produced GDual-GRacSim drawing of  $G$  and its dual proves that producing such drawings is possible. The drawing is not particularly appealing since the height of the strips quickly becomes very small. However, it is a starting point towards algorithms that produce better layouts. Also note that, the algorithm performs a linear number of “point computations” since for each face-vertex of the dual tree the performed computations are pro-



**Figure 4.12:** The initial step of our algorithm.

portional to the degree of the face-vertex. However, the coordinates of some points may be non-rational numbers.



## 5 Maximizing the Total Resolution of Graphs

In this chapter, we study the problem of maximizing the total resolution of a graph drawing that is, the minimum of the angular and crossing resolution. To the best of our knowledge, this is the first time where the problem of maximizing the total resolution of a graph over all its drawings is studied. The main contribution of the chapter consists of drawings of asymptotically optimal total resolution for complete graphs (circular drawings) and for complete bipartite graphs (2-layered drawings). In addition, we present and experimentally evaluate a force-directed based algorithm that constructs drawings of large total resolution.

### 5.1 Preliminary Notions and Notation

Given a drawing  $\Gamma(G)$  of an undirected graph  $G = (V, E)$ , we denote by  $p_u = (x(u), y(u))$  the position of vertex  $u \in V$  on the plane. The vector from  $p_u$  to  $p_v$  which has unit length is denoted by  $\overrightarrow{p_u p_v}$ , where  $u, v \in V$ .

Given a pair of points  $q_1, q_2 \in \mathbb{R}^2$ , with a slight abuse of notation, we denote by  $\|q_1 - q_2\|$  the Euclidean distance between  $q_1$  and  $q_2$ . We refer to the line segment defined by  $q_1$  and  $q_2$  as  $\overline{q_1 q_2}$ .

Let  $\vec{\alpha}$  and  $\vec{\gamma}$  be two vectors. The vector which bisects the angle between  $\vec{\alpha}$  and  $\vec{\gamma}$  is  $\frac{\vec{\alpha}}{\|\vec{\alpha}\|} + \frac{\vec{\gamma}}{\|\vec{\gamma}\|}$ . We denote by  $\text{Bsc}(\vec{\alpha}, \vec{\gamma})$  the corresponding unit length vector. Given a vector  $\vec{\beta}$ , we refer to the unit length vector which is perpendicular to  $\vec{\beta}$  and precedes it in the clockwise direction, as  $\text{Perp}(\vec{\beta})$ . Some of our proofs use the following elementary geometric properties:

$$\tan(\omega_1 \pm \omega_2) = \frac{\tan \omega_1 \pm \tan \omega_2}{1 \pm (-\tan \omega_1 \cdot \tan \omega_2)} \quad (5.1)$$

$$\tan(\omega/2) = \frac{\sin \omega}{1 + \cos \omega} \quad (5.2)$$

$$\omega \in (0, \frac{\pi}{2}) \Rightarrow \tan \omega > \omega \quad (5.3)$$

## 5.2 Drawings with Optimal Total Resolution for Complete and Complete Bipartite Graphs

In this section, we first define formally the total resolution of a drawing. Obviously, the problem of determining the maximum total resolution for the case of planar straight-line drawings of planar graphs is equivalent to the corresponding problem of determining the maximum angular resolution, which is a well studied problem in the graph drawing literature. Therefore, we concentrate our study on non-planar classes of graphs. As a first step towards solving the total resolution maximization problem on arbitrary non-planar graphs, we study the intuitively easier cases of complete and complete bipartite graphs, for which we seek to obtain circular and 2-layered drawings of maximum total resolution, respectively. Note that, such drawings are common for the visualization of complete and complete bipartite graphs. In particular, we present two methods for obtaining drawings of asymptotically optimal total resolution for complete and complete bipartite graphs. We prove that (i) any complete graph  $K_n$  admits a circular drawing of total resolution  $O(1/n)$ , which is asymptotically optimal, and, (ii) any complete bipartite graph  $K_{n,m}$  admits a layered drawing of total resolution  $O(1/\max\{n, m\})$ , which is also asymptotically optimal.

We first consider the case of complete graphs. Let  $K_n = (V, E)$  be a complete graph, where  $V = \{u_0, u_1, \dots, u_{n-1}\}$  and  $E = V \times V$ . Our aim is to construct a circular drawing of  $K_n$  of maximum total resolution. Our approach is constructive and quite common when dealing with complete graphs. A similar construction has been given by Formann et al. [FHH<sup>+</sup>93] for obtaining optimal drawings of complete graphs, in terms of angular resolution. Consider a circle  $\mathcal{C}$  of radius  $r_c > 0$  centered at  $(0, 0)$  and inscribe a regular  $n$ -polygon  $\mathcal{Q}$  on  $\mathcal{C}$ . In our construction, the vertices of  $K_n$  coincide with the vertices of  $\mathcal{Q}$ . W.l.o.g., we further assume that  $u_0, u_1, \dots, u_{n-1}$  appear in this order in the counter-clockwise direction around  $(0, 0)$ , as illustrated in Figure 5.1i.

**Theorem 5.2.1.** *A complete graph  $K_n$  admits a circular drawing of total resolution  $O(\frac{1}{n})$ , which is asymptotically optimal.*

*Proof.* We prove that the angular resolution of the presented drawing of  $K_n$  is  $\frac{\pi}{n}$ , whereas its crossing resolution is  $\frac{2\pi}{n}$ . First, observe that the length of the arc of circle  $\mathcal{C}$  that connects two consecutive vertices  $u_i$  and  $u_{(i+1) \bmod n}$  is equal to  $\frac{2\pi r_c}{n}$ , for each  $i = 0, 1, \dots, n-1$ . Therefore, the angular resolution of the drawing is  $\frac{\pi}{n}$ , as desired. Let now  $e_i = (u_i, u_{i'})$  and  $e_j = (u_j, u_{j'})$  be two crossing edges. Without loss of generality, we assume that  $i < j < i' < j'$ , as in Figure 5.1i. The crossing of  $e_i$  and  $e_j$  defines two angles  $\phi_c$  and  $\phi'_c$  such that  $\phi_c + \phi'_c = \pi$ . In Figure 5.1i,  $\phi_c$  is exterior to the triangle formed by the crossing of  $e_i$  and  $e_j$  and the vertices  $u_j$  and  $u_{i'}$  (refer to the dark-gray triangle of Figure 5.1i). Therefore:  $\phi_c = (j' - i')\frac{\pi}{n} + (j - i)\frac{\pi}{n}$ .

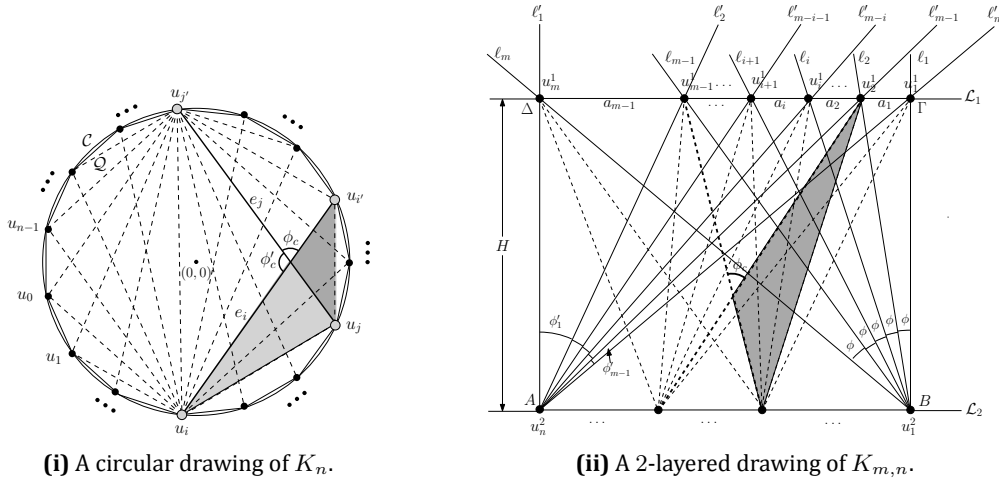


Figure 5.1: Illustrations of our constructions

Similarly,  $\phi'_c = (i' - j)\frac{\pi}{n} + (n - (j' - i))\frac{\pi}{n}$ . In the case in which  $j = (i + 1) \bmod n$  and  $j' = (i' + 1) \bmod n$  (i.e., the vertices  $u_i$  ( $u_{i'}$ , resp.) and  $u_j$  ( $u_{j'}$ , resp.) are consecutive), angle  $\phi_c$  receives its minimum value, which is equal to  $\frac{2\pi}{n}$ . Similarly, we can prove that the minimum value of  $\phi'_c$  is also  $\frac{2\pi}{n}$ . This establishes that the crossing resolution of the drawing is  $\frac{2\pi}{n}$ . Note that, the same upper bound for the crossing resolution of  $K_n$  has been independently proved by Di Giacomo et al. in [DDE<sup>+</sup>12].  $\square$

We now proceed to consider the class of complete bipartite graphs. Since an  $n$ -vertex complete bipartite graph is a subgraph of an  $n$ -vertex complete graph, a drawing of  $\Theta(\frac{1}{n})$  total resolution for the complete bipartite graph can be derived by the optimal drawing for the complete graph. However, if the vertices of the graph must have integer coordinates (i.e., we restrict ourselves on grid drawings), few results are known regarding the area needed for such a drawing. An upper bound of  $O(n^3)$  area can be derived by [BT04]. This motivates us to separately study the class of complete bipartite graphs, since we can drastically improve this bound. Note that, tradeoffs between (angular or crossing) resolution and area have been studied in the past. Malitz and Papakostas [MP94] showed that there exist graphs that always require exponential area for straight-line embeddings maintaining good angular resolution. The claim remains true, if circular arc edges are used instead of straight-lines [CDGK99]. More recently, Angelini et al. [ACD<sup>+</sup>09] constructively showed that there exist graphs whose straight-line upward RAC drawings require exponential area.

Let  $K_{m,n} = (V_1 \cup V_2, E)$  be a complete bipartite graph, where  $V_1 = \{u_1^1, u_2^1, \dots, u_m^1\}$ ,  $V_2 = \{u_1^2, u_2^2, \dots, u_n^2\}$  and  $E = V_1 \times V_2$ . W.l.o.g., we assume that  $m \geq n$ . As already stated, we seek to obtain 2-layered drawings in which the vertices of  $V_1$  and  $V_2$  lie on two parallel

lines, say  $\mathcal{L}_1$  and  $\mathcal{L}_2$ , respectively. For the sake of simplicity, we assume that both  $\mathcal{L}_1$  and  $\mathcal{L}_2$  are horizontal. We first provide a geometric construction according to which the vertices are not required to be on grid points and then we show how to convert it into a grid drawing.

Initially, we consider a square  $\mathcal{R} = AB\Gamma\Delta$ , whose top and bottom sides coincide with  $\mathcal{L}_1$  and  $\mathcal{L}_2$ , respectively (see Figure 5.1ii). Let  $H$  be the height (and width) of  $\mathcal{R}$ . According to our approach, the vertices of  $V_1$  ( $V_2$ , resp.) reside along side  $\Gamma\Delta$  ( $AB$ , resp.) of  $\mathcal{R}$ . In order to specify the exact positions of the vertices  $u_1^1, u_2^1, \dots, u_m^1$  along side  $\Gamma\Delta$ , we first construct a bundle of  $m$  semi-lines, say  $\ell_1, \dots, \ell_m$ , each of which emanates from vertex  $B$  and crosses side  $\Gamma\Delta$  of  $\mathcal{R}$ , so that the angle formed by  $B\Gamma$  and semi-line  $\ell_i$  is equal to  $\frac{(i-1) \cdot \widehat{\Delta B\Gamma}}{m-1}$ , for each  $i = 1, \dots, m$ . These semi-lines split angle  $\widehat{\Delta B\Gamma}$  into  $m - 1$  angles, each of which is equal to  $\frac{\pi}{4 \cdot (m-1)}$ , since  $\widehat{\Delta B\Gamma} = \pi/4$ . Say  $\phi = \frac{\pi}{4 \cdot (m-1)}$ . Then, we place vertices  $u_i^1$  at the intersection of semi-line  $\ell_i$  and  $\Gamma\Delta$ , for each  $i = 1, \dots, m$  (see Figure 5.1ii). Symmetrically, we compute the position of the vertices of  $V_2$  along side  $AB$  of  $\mathcal{R}$ .

We denote by  $a_i$  the horizontal distance between two consecutive vertices  $u_i^1$  and  $u_{i+1}^1$ ,  $i = 1, \dots, m - 1$ . We also define an additional bundle of  $m$  semi-lines, say  $\ell'_1, \dots, \ell'_m$ , that emanate from vertex  $A$ . More precisely, semi-line  $\ell'_i$  emanates from vertex  $A$  and passes through the intersection of  $\ell_{m-i}$  and  $\Gamma\Delta$  (i.e., vertex  $u_{m-i}^1$ ), for each  $i = 1, \dots, m$  (see Figure 5.1ii). Let  $\phi'_i$  be the angle formed by two consecutive semi-lines  $\ell'_i$  and  $\ell'_{i+1}$ , for each  $i = 1, \dots, m - 1$ . We are now ready to investigate some geometric properties of the proposed construction.

**Lemma 5.2.1.** *For each  $i = 1, 2, \dots, m - 1$ , it holds that  $a_{i-1} < a_i$ .*

*Proof.* By induction. For the base of the induction, we have to show that  $a_1 < a_2$ . First observe that  $a_1 = H \tan \phi$  and  $a_1 + a_2 = H \tan 2\phi$ . Therefore:

$$a_2 = H(\tan 2\phi - \tan \phi) \stackrel{(5.1)}{=} a_1 \cdot (1 + \tan 2\phi \cdot \tan \phi)$$

However, both  $\tan \phi$  and  $\tan 2\phi$  are greater than zero, which immediately implies that  $a_1 < a_2$ . For the induction hypothesis, we assume that  $\forall k$ , with  $k < m - 1$ , it holds that  $a_{k-1} < a_k$  and we prove that  $a_k < a_{k+1}$ . Obviously,  $a_1 + \dots + a_k = H \tan k\phi$ . Based on Equation 5.1 and similarly to the base of the induction, we have:

- $a_{k+1} = H \tan \phi \cdot (1 + \tan(k+1)\phi \cdot \tan k\phi)$
- $a_k = H \tan \phi \cdot (1 + \tan(k-1)\phi \cdot \tan k\phi)$

In order to complete the proof, observe that  $(k+1)\phi > (k-1)\phi$ . □

**Lemma 5.2.2.** *For each  $i = 2, \dots, m - 1$ , it holds that  $\phi'_{i-1} > \phi'_i$ .*

## 5.2 Drawings with Optimal Total Resolution for Complete and Complete Bipartite Graphs 117

*Proof.* By induction. For the base of the induction, we have to prove that  $\phi'_1 > \phi'_2$  or equivalently that  $\tan \phi'_1 > \tan \phi'_2$ . It holds that  $\tan \phi'_1 = a_{m-1}/H$  and  $\tan(\phi'_1 + \phi'_2) = (a_{m-1} + a_{m-2})/H$ . By combining these relationships with Equation 5.1, we have that  $\tan \phi'_2 = \frac{Ha_{m-2}}{H^2 + a_{m-1}^2 + a_{m-1}a_{m-2}}$ . Therefore:

$$\begin{aligned} \tan \phi'_1 > \tan \phi'_2 &\Leftrightarrow \\ H^2(a_{m-1} - a_{m-2}) + a_{m-1}^3 + a_{m-1}^2 a_{m-2} &> 0, \end{aligned}$$

which trivially holds due to Lemma 5.2.1. For the induction hypothesis, we assume that  $\forall k, k < m - 1$ , it holds that  $\phi'_{k-1} > \phi'_k$  and we have to show that  $\phi'_k > \phi'_{k+1}$ . Observe that:

$$\begin{aligned} \tan \phi'_k &= \frac{\tan(\phi'_1 + \dots + \phi'_k) - \tan(\phi'_1 + \dots + \phi'_{k-1})}{1 + \tan(\phi'_1 + \dots + \phi'_k) \cdot \tan(\phi'_1 + \dots + \phi'_{k-1})} \\ &= \frac{Ha_{m-k}}{H^2 + (a_{m-1} + \dots + a_{m-k})(a_{m-1} + \dots + a_{m-k+1})} \\ \tan(\phi'_{k+1}) &= \frac{Ha_{m-(k+1)}}{H^2 + (a_{m-1} + \dots + a_{m-(k+1)})(a_{m-1} + \dots + a_{m-k})} \end{aligned}$$

By Lemma 5.2.1, we have that  $(a_{m-1} + \dots + a_{m-(k-1)}) < (a_{m-1} + \dots + a_{m-(k+1)})$  and  $H \cdot a_{m-k} > H \cdot a_{m-(k+1)}$ . Therefore,  $\tan \phi'_k > \tan \phi'_{k+1}$ .  $\square$

**Lemma 5.2.3.** *It holds that  $\phi'_{m-1} \leq \phi$ .*

*Proof.* We equivalently prove that  $\tan \phi'_{m-1} \leq \tan \phi$ .

$$\begin{aligned} \tan \phi'_{m-1} \leq \frac{a_1}{H} &\Leftrightarrow \frac{\frac{a_1}{H}}{1 + \frac{a_1 + \dots + a_{m-1}}{H} \cdot \frac{a_2 + \dots + a_{m-1}}{H}} \leq \frac{a_1}{H} \\ &\Leftrightarrow \frac{1}{1 + \frac{H-a_1}{H}} \leq 1 \\ &\Leftrightarrow a_1 \leq H \end{aligned}$$

which obviously holds.  $\square$

**Lemma 5.2.4.** *Angle  $\phi'_{m-1}$  is the smallest angle among all the angles formed in the drawing.*

*Proof.* From Lemma 5.2.2, it follows that angle  $\phi'_{m-1}$  is the smallest angle among all  $\phi'_i, i = 1, \dots, m - 1$ . Additionally, each angle  $\phi'_i$  gets larger when the endpoint of the bundle (i.e., vertex  $u_n^2$ ) moves along side  $AB$  towards the intersection point of  $AB$  and the perpendicular bisector of line segment  $\overline{u_{m-(i-1)}^1 u_{m-i}^1}$  where it obtains its greatest value. From that point on and to its right, it continuously decreases, until it gets its minimum value when vertex  $u_1^2$  is reached. At this point, it becomes equal to  $\phi$ , which by Lemma 5.2.3 is greater than angle  $\phi'_{m-1}$ . Therefore,  $\phi'_{m-1}$  is the smallest angle among all the angles formed by pairs of consecutive edges incident to any vertex of  $V_2$ . Since  $m \geq n$ , the same holds for the vertices of  $V_1$ . Therefore,  $\phi'_{m-1}$  defines the angular resolution of the drawing.

We now proceed to show that the crossing resolution of the constructed drawing is always greater than the angular resolution. To realize this, consider two crossing edges (refer to the bold, crossing dashed-edges of Figure 5.1ii). Their crossing defines (a) a pair of angles that are smaller than  $90^\circ$  and (b) another pair of angles that are larger than  $90^\circ$ . Obviously, only the acute angles participate in the computation of the crossing resolution (see angle  $\phi_c$  in Figure 5.1ii). However, in a complete bipartite graph, the acute angles formed by two crossing edges are always exterior to a triangle having two of its vertices on  $V_1$  and  $V_2$ , respectively (refer to the gray-colored triangle of Figure 5.1ii). Therefore, the crossing resolution is always greater than the angular resolution, as desired.  $\square$

**Lemma 5.2.5.** *It holds that  $\phi'_{m-1} \geq \frac{\phi}{2}$ .*

*Proof.* We equivalently prove that  $\tan \phi'_{m-1} > \tan \frac{\phi}{2}$ . Let  $x$  be the length of edge  $(u_1^2, u_1^1)$  in the produced drawing. Then, by Equation 5.2 and since  $x > H$ , we have the following:

$$\tan \frac{\phi}{2} = \frac{\frac{a_1}{x}}{1 + \frac{H}{x}} < \frac{a_1}{2H}$$

From the above relationship, it follows that in order to complete the proof of this lemma, it is enough to prove that  $\tan \phi'_{m-1} > \frac{a_1}{2H}$ . So, we have:

$$\begin{aligned} \tan \phi'_{m-1} > \frac{a_1}{2H} &\Leftrightarrow \frac{\frac{a_1}{H}}{1 + \frac{a_1 + \dots + a_{m-1}}{H} \cdot \frac{a_2 + \dots + a_{m-1}}{H}} > \frac{a_1}{2H} \\ &\Leftrightarrow \frac{1}{1 + \frac{H - a_1}{H}} > \frac{1}{2} \\ &\Leftrightarrow a_1 > 0 \end{aligned}$$

which obviously holds.  $\square$

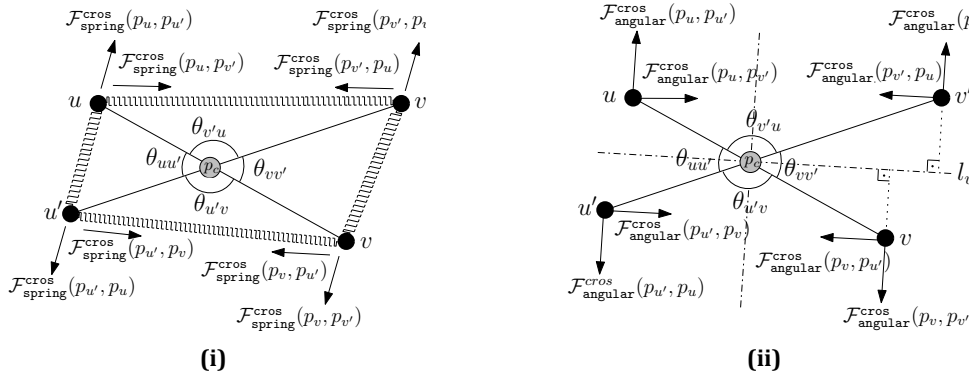
**Theorem 5.2.2.** *A complete bipartite graph  $K_{m,n}$  admits a 2-layered drawing of total resolution  $O(\frac{1}{\max\{m,n\}})$ , which is asymptotically optimal.*

*Proof.* Immediately follows from Lemmata 5.2.4 and 5.2.5.  $\square$

Assume, without loss of generality, that  $\mathcal{L}_1$  and  $\mathcal{L}_2$  are two horizontal lines,  $\mathcal{L}_2$  coincides with  $x$ -axis and the drawing produced by our algorithm has  $a_1 = 1$ . Then, we can express the height of drawing  $\Gamma(K_{m,n})$  as a function of  $\phi$ , as follows:

$$a_1 = 1 \iff \tan \phi \cdot H = 1 \iff H = 1/\tan \phi$$

Based on the above, the area of  $\Gamma(K_{m,n})$ , which is equal to  $H^2$ , is bounded by  $1/\tan^2 \phi$ . By Equation 5.3, this is further bounded by  $1/\phi^2$ . By Theorem 5.2.2, it holds that  $\phi = O(1/\max\{m, n\})$ .



**Figure 5.2:** Forces applied on vertices in order to maximize the crossing resolution. (i) Springs on vertices involved in crossing. (ii) Repelling or attractive forces based on the angles.

Therefore, the total area occupied by the drawing is  $O(\max\{m^2, n^2\})$ . However, the vertices of  $K_{m,n}$  in  $\Gamma(K_{m,n})$  are not necessarily located at grid points. To achieve this, we move the horizontal line  $\mathcal{L}_1$  to the horizontal grid line immediately above it and each vertex of both  $V_1$  and  $V_2$  to the rightmost grid-point to its left. In this manner, we obtain a new drawing  $\Gamma'(K_{m,n})$ , in which every vertex is located at a grid point. By Lemma 5.2.1, it follows that there are no two vertices sharing the same grid point, since  $a_1$  is equal to one grid unit. Since neither horizontal line  $\mathcal{L}_1$  nor any vertex of  $K_{m,n}$  moves more than one unit of length, the total resolution of  $\Gamma'(K_{m,n})$  is not asymptotically affected, and, in addition the height of the drawing is not significantly greater (i.e., asymptotically it remains the same). The following theorem summarizes this result.

**Theorem 5.2.3.** *A complete bipartite graph  $K_{m,n}$  admits a 2-layered grid drawing of  $\Theta(\frac{1}{\max\{m,n\}})$  total resolution and  $O(\max\{m^2, n^2\})$  area.*

### 5.3 A Force-Directed Algorithm

We present a force-directed algorithm that, starting from an initial drawing computed by a classical force-directed technique, results into a drawing of improved total resolution. The algorithm reinforces the classical force-directed algorithm of Eades [Ead84] with some additional forces exerted to the vertices of the graph. More precisely, these additional forces involve springs and some extra attractive or repulsive forces on vertices with degree greater than one and on endpoints of edges that are involved in an edge crossing. This aims to ensure that the angles between incident edges and the angles formed by pairs of crossing edges will be as large as possible in an equilibrium state of the model.

As already mentioned in Chapter 1, the classical force-directed algorithm of Eades [Ead84]

models the vertices of the graph as electrically charged particles that repel each other, and its edges by springs in order to attract adjacent vertices. Based on experimental results, we employ the attractive forces of the classical force-directed algorithm of Eades and omit the repulsive ones. In addition, we have chosen to use springs that follow the logarithmic law instead of the Hooke's law, in order to avoid exerting strong forces on distant vertices. More precisely, the attractive forces (denoted by  $\mathcal{F}_{\text{spring}}$ ) follow the formula:

$$\mathcal{F}_{\text{spring}}(p_u, p_v) = C_{\text{spring}} \cdot \log \frac{\|p_u - p_v\|}{\ell_{\text{spring}}} \cdot \overrightarrow{p_u p_v}, (u, v) \in E$$

where  $C_{\text{spring}}$  and  $\ell_{\text{spring}}$  capture the *stiffness* and the *natural length* of the springs, respectively. Recall that  $\overrightarrow{p_u p_v}$  denotes the unit length vector from  $p_u$  to  $p_v$ .

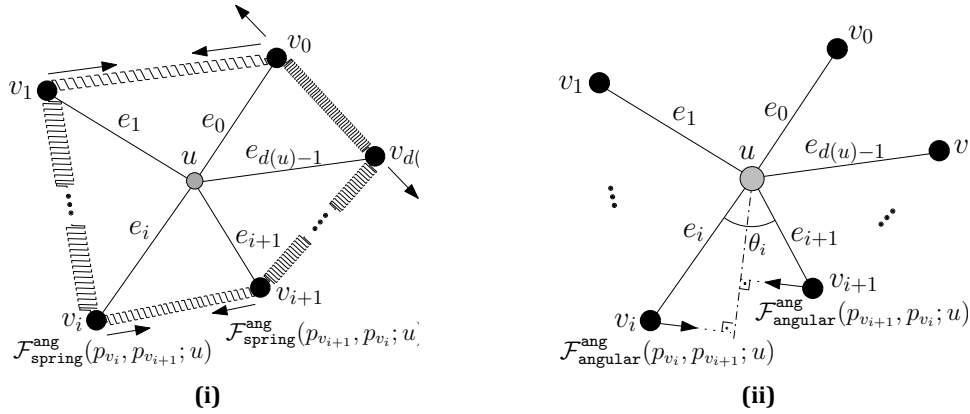
We first describe our approach for the case where two edges, say  $e = (u, v)$  and  $e' = (u', v')$ , are involved in a crossing. Let  $p_c$  be their intersection point (see the gray-colored point of Figure 5.2). W.l.o.g., we assume that  $u$  is to the left of  $v$ ,  $u'$  to the left of  $v'$ ,  $y_{u'} < y_u$  and  $y_v < y_{v'}$ , as in Figure 5.2. Let  $\theta_{vv'}$  be the angle formed by the line segments  $\overline{p_c p_v}$  and  $\overline{p_c p_{v'}}$  in counter-clockwise order around  $u$  from  $\overline{p_c p_v}$  to  $\overline{p_c p_{v'}}$ . In order to avoid confusion, we assume that  $\theta_{vv'} = \theta_{v'v}$ , i.e., we abuse the counter-clockwise measurement of the angles that would result in  $\theta_{vv'} = 2\pi - \theta_{v'v}$ . Similarly, we define the remaining angles of Figure 5.2. Obviously,  $\theta_{vv'} + \theta_{v'u} = \pi$ . Ideally, we would like  $\theta_{vv'} = \theta_{v'u} = \frac{\pi}{2}$ , i.e.,  $e$  and  $e'$  form a right angle crossing. As we will shortly see, the magnitude of the forces that we apply on the vertices  $u, u', v$  and  $v'$  depends on (a) the angles  $\theta_{vv'}$  and  $\theta_{v'u}$  and (b) the lengths of the line segments  $\overline{p_c p_u}, \overline{p_c p_{u'}}$ ,  $\overline{p_c p_v}$  and  $\overline{p_c p_{v'}}$ .

The physical model that describes our approach is illustrated in Figure 5.2. Initially, for each pair of crossing edges at point  $p_c$ , we place springs connecting consecutive vertices in the counter-clockwise order around  $p_c$ , as in Figure 5.2i. The magnitude of the forces due to these springs should capture our preference for right angles. Consider the spring connecting  $v$  and  $v'$ . The remaining ones are treated symmetrically. We set the natural length, say  $\ell_{\text{spring}}^{vv'}$ , of the spring connecting the vertices  $v$  and  $v'$  to be  $\sqrt{\|p_c - p_v\|^2 + \|p_c - p_{v'}\|^2}$ . This quantity corresponds to the length of the line segment that connects  $v$  and  $v'$  in the optimal case where  $\theta_{vv'} = \frac{\pi}{2}$ . So, in an equilibrium state of this model on a graph consisting only of  $e$  and  $e'$ , edges  $e$  and  $e'$  will form a right angle crossing. Concluding, the force on  $v$  due to the spring of  $v'$  is defined as follows:

$$\mathcal{F}_{\text{spring}}^{\text{cros}}(p_v, p_{v'}) = C_{\text{spring}}^{\text{cros}} \cdot \log \frac{\|p_v - p_{v'}\|}{\ell_{\text{spring}}^{vv'}} \cdot \overrightarrow{p_v p_{v'}}$$

The remaining forces of Figure 5.2i are defined similarly. Note that, in the formula above, the constant  $C_{\text{spring}}^{\text{cros}}$  is used to control the stiffness of the springs.





**Figure 5.3:** Forces applied on vertices in order to maximize the angular resolution. (i) Springs on consecutive edges around  $u$ . (ii) Repelling or attractive forces based on the angles.

So far, we have managed to express our preference for right angle crossings based on the lengths of the line segments  $\overline{p_c p_u}$ ,  $\overline{p_c p_{u'}}$ ,  $\overline{p_c p_v}$  and  $\overline{p_c p_{v'}}$ . The same can be achieved using the angles  $\theta_{vv'}$  and  $\theta_{v'u}$  (see Figure 5.2ii). We again restrict our description on the angle formed by the line segments  $\overline{p_c p_v}$  and  $\overline{p_c p_{v'}}$ . Ideally, we would like to exert forces on the vertices  $v$  and  $v'$  such that: (i) when  $\theta_{vv'} \rightarrow 0$ , the magnitude of the force is very large (in order to repel  $v$  and  $v'$ ), and (ii) when  $\theta_{vv'} \rightarrow \frac{\pi}{2}$ , the magnitude of the force is very small. A function, say  $f : \mathbb{R} \rightarrow \mathbb{R}$ , which captures this property is:  $f(\theta) = \frac{|\frac{\pi}{2} - \theta|}{\theta}$ . Having specified the magnitude of the forces, it remains to specify its direction. More precisely, we set the direction of the force on  $v$  (due to  $v'$ ) to be perpendicular to the line that bisects the angle  $\theta_{vv'}$  (refer to the dash-dotted line  $l_{vv'}$  of Figure 5.2ii), or equivalently parallel to the unit length vector  $\text{Perp}(\text{Bsc}(\overline{p_c p_v}, \overline{p_c p_{v'}}))$ . Recall that  $\text{Perp}$  and  $\text{Bsc}$  refer to the perpendicular and bisector vectors, respectively (see Section 5.1). It is clear that if  $\theta_{vv'} < \frac{\pi}{2}$ , the forces on  $v$  and  $v'$  should be repulsive (in order to enlarge the angle between them), otherwise attractive. This can be captured by the *sign* function. Summarizing the above, we use the following formula which expresses the force on  $v$  due to  $v'$ .

$$\mathcal{F}_{\text{angle}}^{\text{cross}}(p_v, p_{v'}) = C_{\text{angle}}^{\text{cross}} \cdot \text{sign}(\theta_{vv'} - \frac{\pi}{2}) \cdot f(\theta_{vv'}) \cdot \text{Perp}(\text{Bsc}(\overline{p_c p_v}, \overline{p_c p_{v'}}))$$

where constant  $C_{\text{angle}}^{\text{cross}}$  controls the strength of the force. Similarly, we define the remaining forces of Figure 5.2ii.

Consider now a vertex  $u \in V$  that is incident to  $d(u)$  edges, say  $e_0 = (u, v_0)$ ,  $e_1 = (u, v_1)$ ,  $\dots$ ,  $e_{d(u)-1} = (u, v_{d(u)-1})$ . Refer to Figure 5.3i. We assume that  $e_0, e_1, \dots, e_{d(u)-1}$  are consecutive in the counter-clockwise order around  $u$  in the drawing of the graph. Similarly to the case of two crossing edges, we proceed to connect the endpoints of consecutive edges around  $u$

by springs, as in Figure 5.3i. In this case, the natural length of each spring, should capture our preference for angles equal to  $\frac{2\pi}{d(u)}$  (i.e., the obvious upper bound). In order to achieve this, we proceed as follows: For each  $i = 0, 1, \dots, d(u) - 1$ , we set the natural length, say  $l_{\text{spring}}^i$ , of the spring connecting  $v_i$  with  $v_{(i+1) \bmod d(u)}$ , to be:

$$l_{\text{spring}}^i = \sqrt{\|a_i\|^2 + \|b_i\|^2 - 2\|a_i\|\|b_i\| \cdot \cos(2\pi/d(u))}$$

where  $a_i = e_i$  and  $b_i = e_{(i+1) \bmod d(u)}$ . The quantity  $l_{\text{spring}}^i$  corresponds to the length of the line segment that connects  $v_i$  with  $v_{(i+1) \bmod d(u)}$  in the optimal case where the angle formed by  $e_i$  and  $e_{(i+1) \bmod d(u)}$  is  $\frac{2\pi}{d(u)}$ , for each  $i = 0, 1, \dots, d(u) - 1$ . Therefore, the spring forces between consecutive edges follow the formula:

$$\mathcal{F}_{\text{spring}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod d(u)}}; u) = C_{\text{spring}}^{\text{angular}} \cdot \log \frac{\|p_{v_i} - p_{v_{(i+1) \bmod d(u)}}\|}{l_{\text{spring}}^i} \cdot \overrightarrow{p_{v_i} p_{v_{(i+1) \bmod d(u)}}}$$

where the quantity  $C_{\text{spring}}^{\text{angular}}$  is a constant which captures the stiffness of the spring.

Let now  $\theta_i$  be the angle formed by  $e_i$  and  $e_{(i+1) \bmod d(u)}$ , measured in counter-clockwise direction from  $e_i$  to  $e_{(i+1) \bmod d(u)}$ ,  $i = 0, 1, \dots, d(u) - 1$ . Similarly to the case of two crossing edges, we exert forces on  $v_i$  and  $v_{(i+1) \bmod d(u)}$  perpendicular to the bisector of  $\theta_i$ , as illustrated in Figure 5.3ii. However, in this case we need a magnitude function such that: (i) when  $\theta_i \rightarrow 0$ , the magnitude of the force is very large (in order to repel  $v_i$  and  $v_{(i+1) \bmod d(u)}$ ), and (ii) when  $\theta_i \rightarrow \frac{2\pi}{d(u)}$ , the magnitude of the force is very small. Such a function, say  $g : \mathbb{R} \times V \rightarrow \mathbb{R}$ , is:  $g(\theta; u) = \frac{|\frac{2\pi}{d(u)} - \theta|}{\theta}$ . Having fully specified the forces applied on the endpoints of consecutive edges and their directions, we are now ready to provide the exact formulas that the forces follow:

$$\mathcal{F}_{\text{angle}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod d(u)}}; u) = C_{\text{angle}}^{\text{angular}} \cdot \text{sign}(\theta_i - \frac{2\pi}{d(u)}) \cdot g(\theta_i; u) \cdot \text{Perp}(\text{Bsc}(\overrightarrow{p_u p_{v_i}}, \overrightarrow{p_u p_{v_{(i+1) \bmod d(u)}}}))$$

where  $C_{\text{angle}}^{\text{angular}}$  is a constant to control the strength of the force. We note that our experimental evaluation has shown that forces  $\mathcal{F}_{\text{cross}}^x$  and  $\mathcal{F}_{\text{angle}}^x$ , where  $x \in \{\text{cross}, \text{angular}\}$  have complementary effect and produce better drawings when they are both present. We stimulate that this is due to the fact that the movement of the vertices is determined as a resultant of a larger number of vectors. In addition, by setting zero values to either  $C_{\text{spring}}^{\text{cross}}$  and  $C_{\text{angle}}^{\text{cross}}$ , or,  $C_{\text{spring}}^{\text{angular}}$  and  $C_{\text{angle}}^{\text{angular}}$ , our algorithm can be configured to maximize only the angular or the crossing resolution, respectively.

### 5.3.1 Time Complexity Analysis

Our approach is outlined in Algorithm 5.1. On each iteration, the algorithm computes three types of forces. Type-1 forces correspond to the attractive forces of the classical force-directed model among pairs of adjacent vertices of the graph. Their computation requires  $O(E)$  time per iteration. In Type-2 forces of Algorithm 5.1, we compute forces due to edge crossings. In a straight-forward manner, the computation of all edges that are involved in crossings in Line 7 of Algorithm 5.1 needs  $O(E^2)$  time. Having computed all pairs of crossing edges, we can compute their associated forces in constant time for each pair of crossing edges. Therefore, Type-2 forces can be computed in  $O(E^2)$  time per iteration. In Type-3 forces of Algorithm 5.1, we compute forces due to the angles between consecutive edges. To cope with this case, initially we have to sort the incident edges of each vertex of the graph in cyclic order (see Line 16 of Algorithm 5.1). This can be done in  $O(d(G) \log d(G))$  time, where  $d(G)$  denotes the degree of the graph. Lines 18-20, where we compute the forces, need an extra of  $O(d(G))$  time. Thus, Type-3 forces needs  $O(E + Vd(G) \log d(G))$  time per iteration. Summarizing the above, each iteration of Algorithm 5.1 takes  $O(E^2 + V(V + d(G) \log d(G)))$  time.

The computational complexity of Algorithm 5.1 can be further improved using standard techniques adopted from computational geometry [dBvKOS00, O'R98, PS85]. More precisely, if  $K$  is the number of pairwise-crossing edges, then the  $K$  intersections in Type-2 force can be reported in  $O(K + E \log^2 E / \log \log E)$  time [PS85, pp.277], which leads to a total time complexity  $O(K + E \log^2 E / \log \log E + V(V + d(G) \log d(G)))$  per iteration.

### 5.3.2 Similarities and Differences with Previous Techniques

In the work of Lin and Yen [LY05], the technique which applies large repelling forces perpendicular to the bisectors of the angles formed by consecutive edges of a vertex, when these angles are small, is referred to as *edge-edge repulsion*. In their work, the function, say  $g : \mathbb{R} \rightarrow \mathbb{R}$ , which controls the magnitude of the force is:  $g(\theta) = \cot(\frac{\theta}{2})$ , where  $\theta$  is the angle formed by two consecutive edges incident to a common vertex. Observe that  $g$  has the same property as the one we used, i.e.,  $g(\theta)$  is very large, when  $\theta \rightarrow 0$ . We preferred to use a different function to control the magnitude of the forces in order to maintain a uniform approach in both crossing and angular cases.

Huang et al. [HEHL10], who have independently published a similar to our force-directed algorithm, use only “angular” forces (i.e., that depend only on the angles formed either by edge crossings or by consecutive edges incident to a common vertex) with different functions to control their magnitudes and different directions. They do not use “spring” forces to affect the angular or the crossing resolution (apart from the ones of the classical force-directed

**Algorithm 5.1:** FORCE-DIRECTED ALGORITHM**Input** : An undirected graph  $G$  and an initial placement  $P = (p_v)_{v \in V}$ .**Output** : A drawing of  $G$  of large angular and crossing resolution.**for** ( $t \leftarrow 1$  **to** ITERATIONS) **do**

{Type 1: Spring forces.}

**foreach**  $u \in V$  **do**

$$\mathcal{F}_u(t) \leftarrow \sum_{v:(u,v) \in E} \mathcal{F}_{\text{spring}}(p_v, p_u);$$

{Type 2: Forces applied on vertices in order to maximize the crossing resolution.}

**foreach** (pair of intersecting edges  $e = (u, v)$  and  $e' = (u', v')$ ) **do**    // The relative positions of edges  $e$  and  $e'$  are illustrated in Figure 5.2.;

$$\mathcal{F}_v(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_v, p_{v'}) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_v, p_{u'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_v, p_{v'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_v, p_{u'});$$

$$\mathcal{F}_{v'}(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{v'}, p_u) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{v'}, p_v) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{v'}, p_u) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{v'}, p_v);$$

$$\mathcal{F}_u(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_u, p_{v'}) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_u, p_{u'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_u, p_{v'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_u, p_{u'});$$

$$\mathcal{F}_{u'}(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{u'}, p_v) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{u'}, p_u) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{u'}, p_v) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{u'}, p_u);$$

{Type 3: Forces applied on vertices in order to maximize the angular resolution.}

**foreach** ( $u \in V$  with  $d(u) > 1$ ) **do**     $e_0, \dots, e_{d(u)-1} \leftarrow$  incident edges of  $u$  in counter-clockwise order (see Figure 5.3);    Let  $e_i = (u, v_i), i = 0, \dots, d(u) - 1$ ;    **for** ( $i \leftarrow 0$  **to**  $d(u) - 1$ ) **do**

$$\mathcal{F}_u(t) += \mathcal{F}_{\text{spring}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod d(u)}}; u) + \mathcal{F}_{\text{angle}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod d(u)}}; u);$$

**foreach**  $u \in V$  **do**

$$p_u \leftarrow p_u + \delta \cdot \mathcal{F}_u(t);$$

algorithm of Eades [Ead84]). In Section 5.3.3, we provide an experimental comparison of these techniques.

In [DLR10] and [DLR11], the crossing resolution is improved in a post-processing step in which every pair of crossing edges is associated with a disk centered at the crossing point. The intersection points of this disk with the edges participating at the crossing define four new vertices that form a four-cycle, called *cage*. Appropriate forces are then applied so that the cages are eventually drawn as close to squares as possible, which enforces their diagonals to cross at large angles. Since these techniques result into drawings with bends, they are not

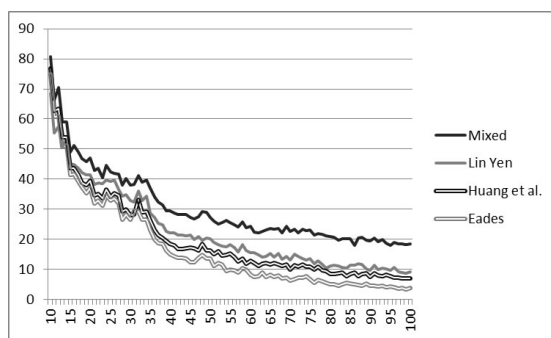
included in our experimental comparison.

### 5.3.3 Experimental Results

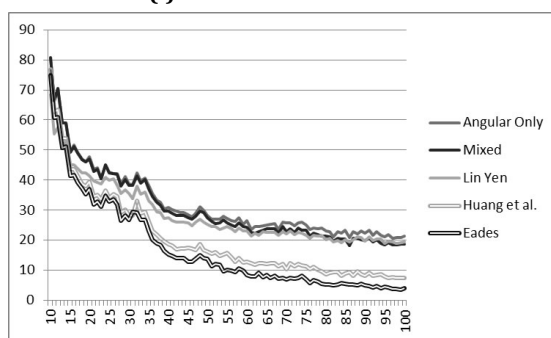
In this section, we present the results of the experimental evaluation of our algorithm. Apart from our algorithm, we have implemented the classical force-directed algorithm of Eades [Ead84], the algorithm of Lin and Yen [LY05] and the algorithm of Huang et al. [HEHL10]. The implementations are in Java using the yFiles library (<http://www.yworks.com>). The experiment was performed on a Linux machine with 2.00 GHz CPU and 2GB RAM using the Rome graphs (a collection of around 11.500 graphs) obtained from [graphdrawing.org](http://graphdrawing.org). Figure 5.8, illustrates a drawing of a Rome graph with 99 vertices and 135 edges produced by our force-directed algorithm. Even though our force-directed algorithm focuses on angular, crossing, and total resolution, we provide data on the performance of the algorithm with respect to total area, edge length and number of crossings (see Figure 5.5).

The experiment was performed as follows. First, each Rome graph was laid out using the SmartOrganic layouter of yFiles. This layout was the input layout of all algorithms, in order to speed up the experiment and overcome problems associated with local minimum traps, especially in large graphs. If both the angular and the crossing resolution between two consecutive iterations of each algorithm were not improved more than 0.001 degrees, we assumed that the algorithm has converged and we did not proceed any more. We note that the termination condition is quite strict and demands a large number of iterations. The maximum number of iterations that an algorithm could perform in order to converge was set to 100.000. However, very few graphs required the maximum number of iterations in order to converge to a layout. Once an algorithm converged, we measured its angular, crossing and total resolution, its average edge length, the number of crossings, the running time, and the required number of iterations to converge. We note that our algorithm is evaluated as (a) Angular-Only, (b) Crossing-Only and (c) Mixed. The results are illustrated in Figures 5.4, 5.5 and 5.6. The values plotted in Figures 5.4, 5.5 and 5.6 are average values over all graphs in the Rome database of a specific size (i.e., number of vertices).

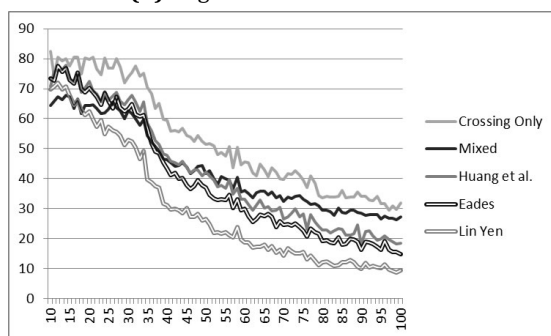
*The Total Resolution Maximization Problem:* The plot in Figure 5.4i indicates that our Mixed algorithm clearly outperforms all other drawing algorithms of the experiment. More specifically, in 80.46% (resp. 90.83%) of the graphs of the experiment, the Mixed algorithm yields a better solution compared to the algorithm of Lin-Yen (resp. Huang et. al) with an average improvement of  $10.21^\circ$  (resp.  $11.63^\circ$ ). The plot in Figure 5.4i also shows that the Mixed algorithm managed to produce drawings of total resolution of about 20 degrees for graphs of more than 50 vertices in the Rome database. An example of such a drawing is given in Figure 5.8.



(i) Total resolution results



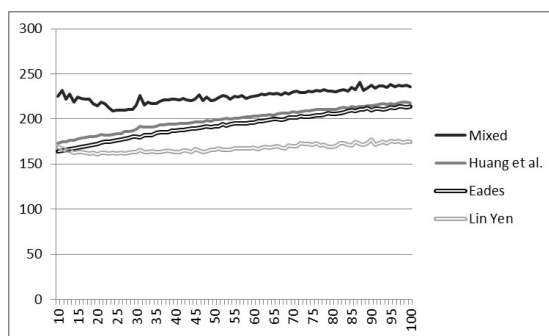
(ii) Angular resolution results



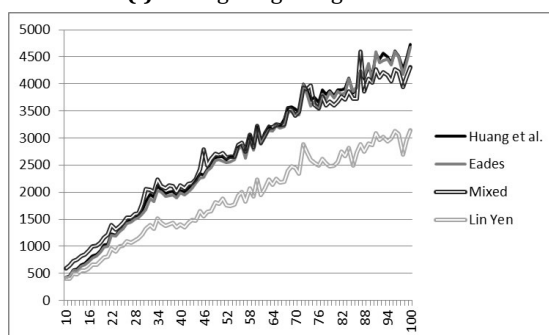
(iii) Crossing resolution results

**Figure 5.4:** A visual presentation of our experimental results on **resolution**: The  $X$ -axis corresponds to the number of the vertices of the graph and the  $Y$ -axis to resolution measured in degrees.

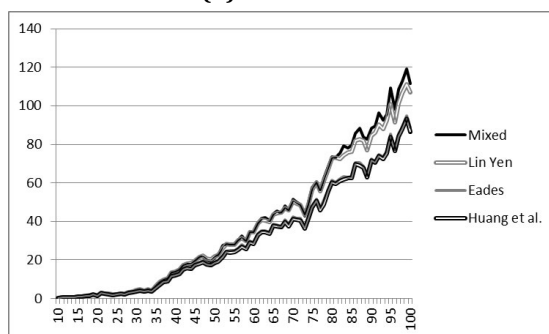
*The Angular Resolution Maximization Problem:* The plot in Figure 5.4ii shows that the Angular-Only algorithm yields drawings with better angular resolution compared to the other drawing algorithms of the experiment. More specifically, in 67.82% (resp. 92.12%) of the graphs of the experiment, the Angular-Only algorithm yields a better solution compared to the algorithm of Lin-Yen (resp. Huang et. al) with an average improvement of  $6.73^\circ$  (resp.  $12.75^\circ$ ). The angular resolution of the drawings produced by the Mixed algorithm is almost



(i) Average edge length results



(ii) Area results

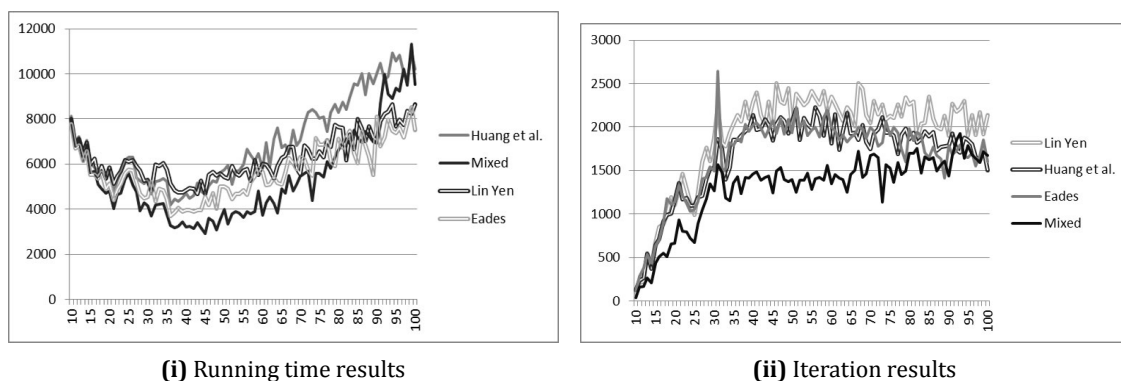


(iii) Number of crossings results

**Figure 5.5:** A visual presentation of our experimental results on **common graph drawing metrics:**

The  $X$ -axis indicates the number of the vertices of the graph. In Figures (i) and (ii), the  $Y$ -axis corresponds to the area and the average edge length of the produced drawings, respectively. In Figure (iii), the  $Y$ -axis corresponds to the number of crossings of the produced drawings.

equal to the ones produced by the algorithm of Lin and Yen. Note that, the algorithm of Lin and Yen, in contrast to ours, does not modify the embedding of the initial layout (refer to [LY05] for more details), i.e., it needs a close-to-final starting layout and improves on it. This explains why the Mixed algorithm achieves almost the same performance, in terms of angular resolu-



**Figure 5.6:** A visual presentation of our experimental results on **algorithm's efficiency**: The  $X$ -axis indicates the number of the vertices of the graph. In Figures (i) and (ii) the  $Y$ -axis corresponds to the running time measured in milliseconds and the iterations needed to converge respectively.

tion, as the one of Lin and Yen. More precisely, in 59.23% of the graphs, the Mixed algorithm yields a better solution compared to Lin-Yen's algorithm with an average improvement of  $6.49^\circ$ .

*The Crossing Resolution Maximization Problem:* In Figure 5.4iii the data were filtered to depict only the results of non-planar drawings produced by the algorithms and avoid infinity values in the case of planar ones. Clearly the Crossing-Only algorithm outperforms the other drawing algorithms of the experiment, yielding drawings with better crossing resolution. More specifically, in 92.26% (resp. 78.47%) of the graphs of the experiment, the Crossing-Only algorithm yields a better solution compared to the algorithm of Lin-Yen (resp. Huang et. al). The average improvement implied by the Crossing-Only algorithm is  $27.13^\circ$  (resp.  $17.08^\circ$ ) w.r.t. the algorithm of Lin-Yen (resp. Huang et. al). The Mixed algorithm yields a better solution in 85.23% (resp. 54.62%) of the graphs of the experiment compared to the algorithm of Lin-Yen (resp. Huang et. al) with an average improvement of  $18.75^\circ$  (resp.  $11.17^\circ$ ).

From the above discussion, it follows that our algorithm results into drawings with better resolution than the ones produced by the other algorithms of the experiment. However, the experimental evaluation also shows that the average edge length of the drawings produced by our algorithm is larger than the average edge length of the drawings produced by the other algorithms (see Figure 5.5i). The edge length is influenced by the natural spring length used by the force-directed algorithms. In the experiment, the natural length of the springs was set to 150 units. The average edge length of the drawings produced by the Mixed algorithm was 225.11 units, while the corresponding ones of Eades, Lin-Yen and Huang et al. were 191.61, 167.1 and 198.45 units, respectively. The experimental evaluation also shows that our algo-

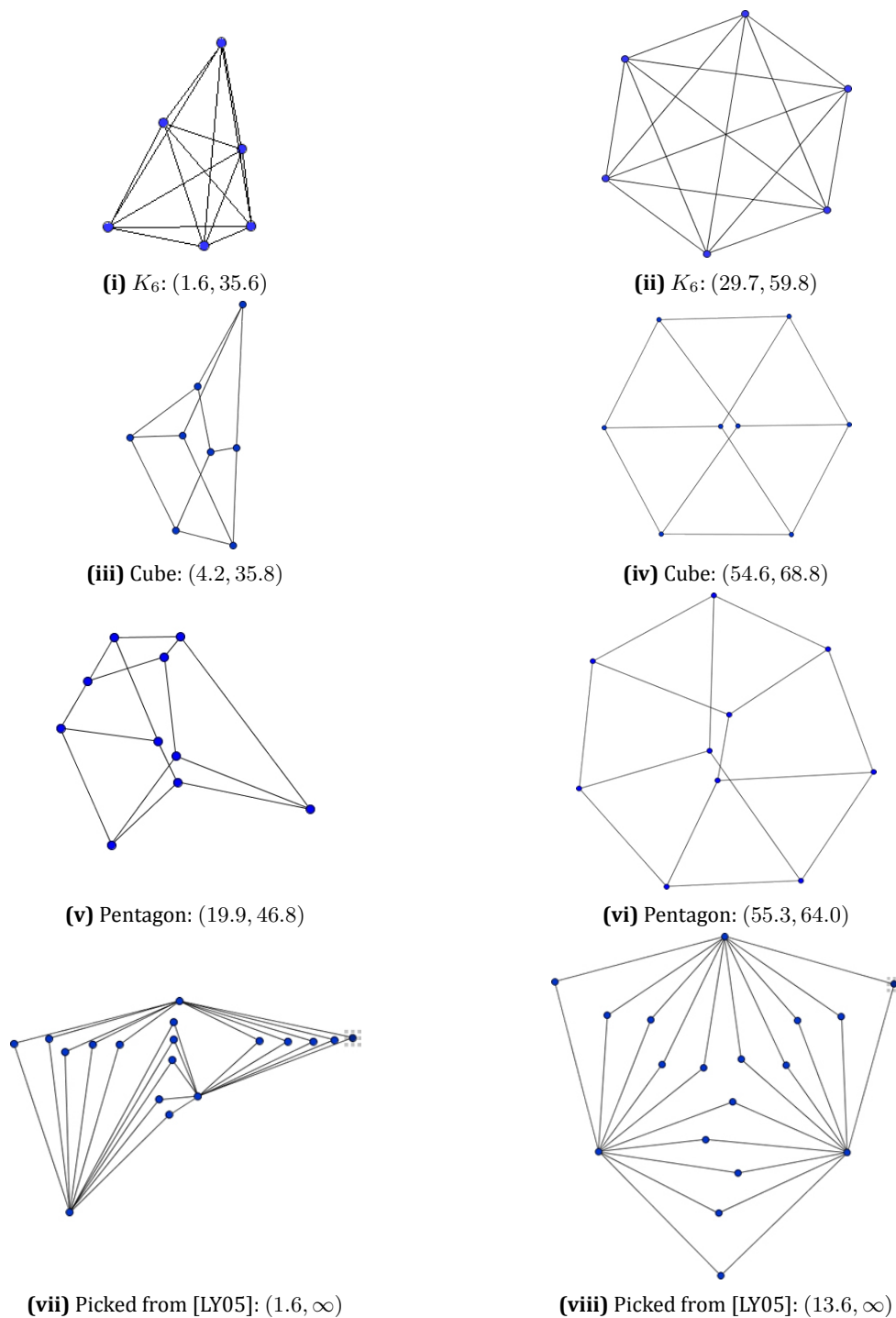


rithm produces drawings of comparable area to that required by the algorithms of Eades and Huang et al. All these algorithms are outperformed by the algorithm of Lin and Yen, which produced the most compact drawings. Despite the fact that the Mixed algorithm produces drawings of larger edge length compared to the drawings produced by the algorithms of Eades and Huang et al., its produced drawings are of comparable area. This may be attributed to the fact that, as the experimental analysis shows, it produces drawings with larger number of crossings. More precisely, Figure 5.5iii shows that the Mixed algorithm yields drawings of almost the same crossings as the one of Lin and Yen and slightly more than the algorithms of Eades and Huang et al.

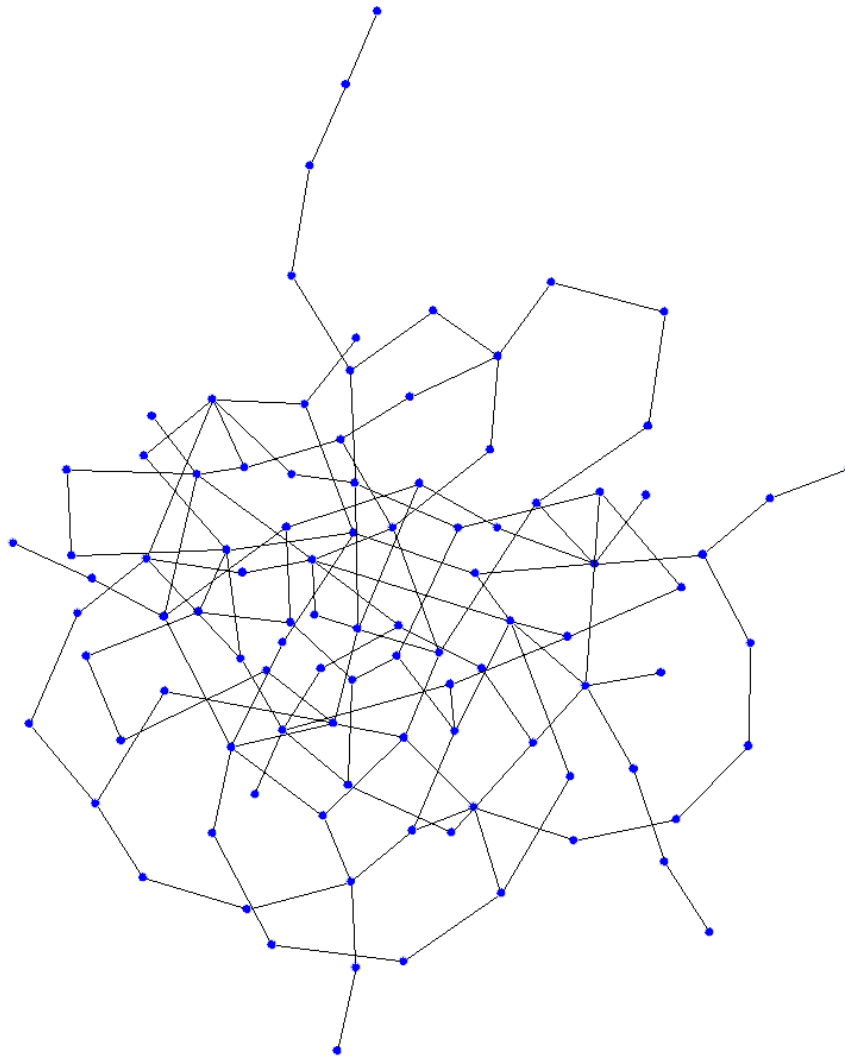
Finally, Figures 5.6i and 5.6ii summarize the running time performance of the algorithms and the iterations required to converge, respectively. The Mixed algorithm needs, on average, 5191 milliseconds and 1298 iterations to converge, while the algorithm of Huang et al. (which is of the same time complexity) 6626 and 1694, respectively. The algorithm of Lin and Yen needs, on average, 6123 milliseconds and 1920 iterations to converge. Note that, the time complexity of Lin-Yen's algorithm is better than ours. However, the termination condition takes into account the crossing improvement and therefore the algorithm of Lin and Yen needs more iterations to converge, which explains this contradiction. In the running time experiment, we exclude the algorithm of Eades, since it does not take into account the angular or the crossing resolution and therefore, it needs more iterations to converge, as depicted in Figure 5.6ii.

## 5.4 Sample Drawings

In this section, we present some sample drawings produced by our force-directed algorithm. Figure 5.7 depicts (some easy to produce) drawings of known graphs. Drawings of these graphs are also presented either in the classical work of Fruchterman and Reingold [FR91] or in the work of Lin and Yen [LY05]. In Figure 5.8, we present a drawing of a Rome graph with 99 vertices and 135 edges.



**Figure 5.7:** Sample drawings produced by our force-directed algorithm. The drawings on each line (of graphs) correspond to the same graph. The drawing on the left is the input of our algorithm, while the drawing on the right is its output. In the caption of each drawing  $(a, b)$  is translated as follows:  $a$  ( $b$ , resp.) is the angular (crossing, resp.) resolution measured in degrees. Drawings of these graphs are also presented either in the classical work of Fruchterman and Reingold [FR91] or in the work of Lin and Yen [LY05].



**Figure 5.8:** A drawing of Rome graph `grafo10129.99` consisting of 99 vertices and 135 edges with angular resolution  $20.15^\circ$  and crossing resolution  $26.12^\circ$ .





## 6 An Introduction to Occupational Fraud Detection

In this chapter, we present basic notions from the area of occupational fraud detection. We introduce the necessary definitions and present the problems which are studied in this Ph.D. thesis.

### 6.1 What is Occupational Fraud?

As already stated in Section 1.6.3, in business terms, *fraud* is defined as *the intentional misuse or abuse of the assets of a company* [AoCFE12]. Based on the relation of the fraudster with the company, fraud may be committed *internally* or *externally* from a company. *Internal fraud* refers to fraud cases that originate from the interior of an organization. *External fraud* refers to fraud schemes committed by outsiders; it involves cases in which the fraudster has no relation with the company.

Major fraud schemes based on 2003 KPMG's survey on fraud [KPM03] can be classified as follows (see Table 6.1): (i) occupational fraud, (ii) consumer fraud, (iii) financial reporting fraud, (iv) misconduct, (v) vendor or third-party fraud and (vi) medical/insurance fraud. Since fraud can be seen by many diverse aspects that are beyond the scope of this Ph.D. thesis, we will focus exclusively on occupational fraud.

Studies on business fraud schemes showed that most of the reported fraud cases have been committed by trusted associates. Figure 6.1 illustrates the results of a survey on the major types of fraud mentioned above [KPM03]. Among these categories, occupational fraud was almost twice as prevalent as consumer fraud which was the second in the row category. Formally, *occupational* or *employee* fraud can be defined as *the use of one's occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization's resources or assets* [AoCFE12]. The Association of Certified Fraud Examiners (ACFE) provides a fraud classification model, referred to as the *fraud tree* that lists diverse fraud schemes, grouped into categories and subcategories (see Figure 6.2). The three major fraud schemes in the fraud tree are the following:

**Falsification of financial statements:** This type of fraud includes cases in which an employee

<b>FRAUD CATEGORIES</b>	
<b><i>Consumer Fraud</i></b>	<b><i>Occupational Fraud</i></b>
<ul style="list-style-type: none"> <li>○ ATM theft</li> <li>○ Check fraud</li> <li>○ Credit card fraud</li> <li>○ Fraudulent classification of merchandise for customers</li> <li>○ Identity theft</li> </ul>	<ul style="list-style-type: none"> <li>○ Check fraud theft</li> <li>○ Expense account abuse</li> <li>○ Payroll fraud</li> <li>○ Pension theft</li> <li>○ Theft or misappropriation of assets</li> </ul>
<b><i>Computer Crime</i></b>	<b><i>Financial Reporting Fraud</i></b>
<ul style="list-style-type: none"> <li>○ Hacking</li> <li>○ Phishing</li> <li>○ Spamming</li> <li>○ Copyright crime</li> <li>○ Identity theft</li> </ul>	<ul style="list-style-type: none"> <li>○ Asset Revenue misstatement</li> <li>○ Concealed liability and expenses</li> <li>○ Improper revenue recognition</li> <li>○ Inadequate omissions</li> <li>○ Inappropriate disclosures</li> </ul>
<b><i>Vendor-Related/Other Third-Party</i></b>	<b><i>Medical/Insurance Fraud</i></b>
<ul style="list-style-type: none"> <li>○ Bid rigging and price fixing</li> <li>○ Bribery</li> <li>○ Diversion of sales</li> <li>○ Duplicate billings</li> <li>○ Extortion</li> <li>○ False invoices and phantom vendors</li> <li>○ Inventory theft</li> <li>○ Kickbacks and conflicts of interest</li> <li>○ Loan fraud</li> <li>○ Theft of intellectual property</li> </ul>	<ul style="list-style-type: none"> <li>○ Medical/insurance claims fraud</li> <li>○ Policy churning</li> <li>○ Worker's compensation fraud</li> </ul>
	<b><i>Misconduct</i></b>
	<ul style="list-style-type: none"> <li>○ Conflicts of interest</li> <li>○ Insider trading</li> <li>○ Payroll fraud</li> <li>○ Pension theft</li> <li>○ Theft or misappropriation of assets</li> </ul>

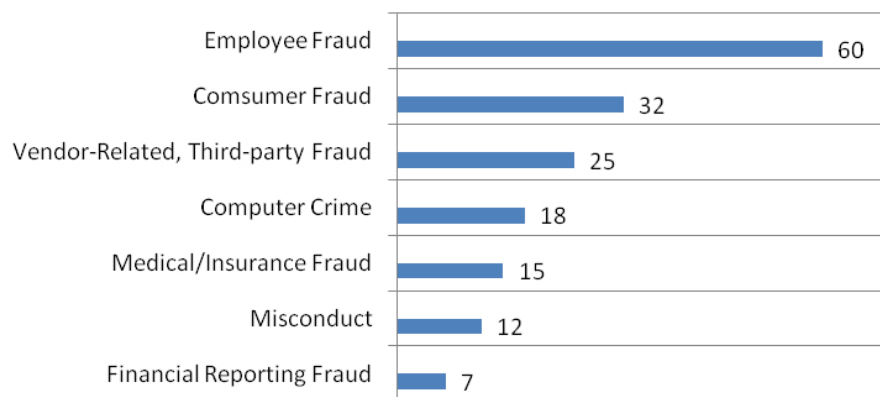
**Table 6.1:** Major fraud categories [KPM03]

intentionally manipulates financial accounts or reports by e.g., overstating assets, profit, or understating liabilities, expenses or losses. Such schemes produce great loss to a company and are mostly committed by employees in senior management or executives.

**Assets misappropriation:** This type of fraud includes cases in which an employee steals or misuses resources of a company, e.g., cash theft, billing fraud schemes and so on. In most

cases, they are committed by lower-level personnel and due to the fact that produce insignificant losses at an individual level, they cannot be easily traced. However, such fraud schemes are by far the most frequent schemes that companies experience. *Billing fraud* is the most frequent in this category and may cause a median loss of 100.000\$ [AoCFE12]. It includes any scheme in which a employee falsifies or deletes invoices, creates invoices for fictitious goods or services and so on.

**Corruption:** As the meaning of the word implies, *corruption* refers to the wrongful use of an employee's influence in a business violating her duty in order to gain a benefit. Such schemes include bribery, kickbacks, conflicts of interest and so on.

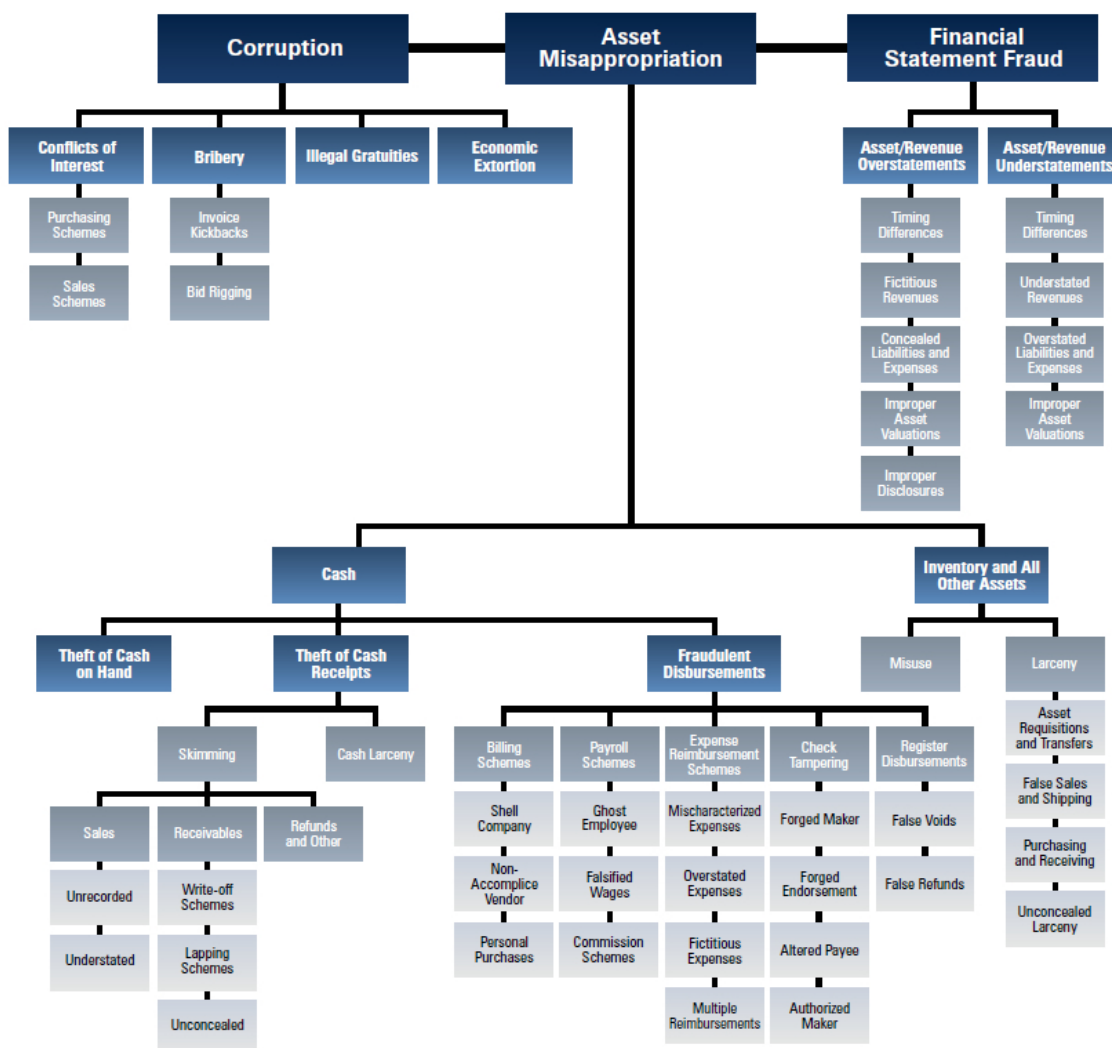


**Figure 6.1:** Types of fraud that companies experienced within a year [KPM03].

Fraud causes a 5% loss to companies' revenues each year, which applied to the estimated 2011 Gross World Product, leads to a potential global loss of more than \$3.5 trillion, while the median loss caused by the occupational fraud in the survey was \$140.000 [AoCFE12]. Apart from the economic loss and the fact that only a small percentage of losses from fraud can be eventually recovered, a company that is victim of fraud has to face many other non-negligible consequences. Among them, business reputation towards its customers, employees and other entities (financial institutions, vendors, etc.), company value, and public and client trust are negatively affected. Hence, it is crucial for a company to develop mechanisms against occupational fraud.

## 6.2 What is Fraud Detection?

Even though, most people might think that fraud "couldn't or wouldn't happen here", the truth is that any company regardless of size, type or country is susceptible to fraud and not



**Figure 6.2:** Occupational fraud and abuse classification system as presented in ACFE’s 2012 Report to the Nations [AoCFE12].

just banks and financial institutions. In particular, an ascendant trend in fraud activities has been recorded in the last couple of years, with a 13% increase only in 2011 [AoCFE12]. As a response to the rising incidents of fraud, many companies have focused on *fraud detection* which includes techniques and methods needed for the identification of actual or potential fraud within an organisation. Usually, anti-fraud strategy involves three steps: (i) fraud prevention, (ii) fraud detection and (iii) fraud investigation and response.

*Prevention techniques* encompass policies, controls and activities such as risk assessment, training and fraud awareness that will discourage any employee from committing fraud. It also includes the organization of internal audit departments and the publication of fraud oc-



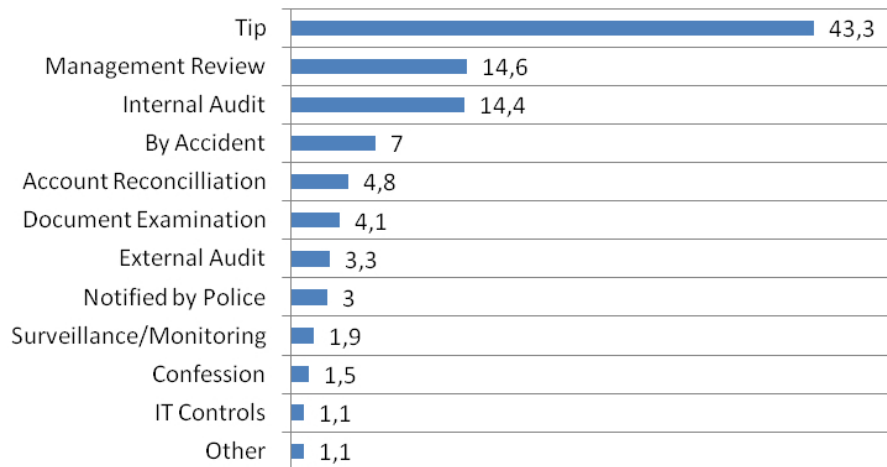
currences. The *investigation* and the *response* to fraud is the final step of a business anti-fraud strategy. During the investigation, auditors have to answer a series of questions of who, how, when and why, once fraud indications are present. In the case where evidence of fraud is revealed, the *response* (i.e., corrective control actions to remedy the harm caused by the fraud) has to be immediate.

However, since it is impossible to prevent all potential perpetrators, fraud schemes have to be identified as soon as possible in order to prevent money loss and protect a company, along with its employees and clients. Detection methods can be either *reactive* or *proactive*. Traditional fraud detection methods are considered to be reactive, since they are initiated by tips, unusual financial statement relationships, or other indicators that someone observes or hears and may include for instance, an anonymous reporting mechanism [AA12]. Proactive fraud detection methods target at predefined fraud schemes and include process controls such as identification of errors, symptoms or red-flags.

A proactive fraud detection strategy includes data analysis, identification of anomalies and trends within data-sets and reporting mechanisms that communicate information about potential fraudulent activity. Data analysis enables auditors to analyze data in order to gain insight into the effectiveness of internal controls and to identify fraudulent activity or the risk of fraud. Proactive detection also includes behavioral analysis in order to score personnel profiles and identify individuals that match to a known fraudster profile. Proactive methods should be flexible and continuously evolving in order to defend penetrators and adapt to their strategies [GTA09].

In addition, the anti-fraud strategy of each organization has to be adapted to the context that the organization operates. For instance, an organization that operates in a country with high level of corruption, has to take supplementary measures to prevent fraud. Continuous monitoring is also required such that the identification of fraud indications is immediate and effective, since the size of fraud increases over time as fraudsters gain confidence that they are not being detected. In particular, a KPMG Forensics' Fraud Risk Management report states that *unlike retrospective analyses, continuous transaction monitoring allows an organization to identify potentially fraudulent transactions on, for example, a daily, weekly, or monthly basis. Organizations frequently use continuous monitoring efforts to focus on narrow bands of transactions or areas that pose particularly strong risks* [KPM06].

As already stated, there exists a wide variety of occupational fraud schemes and unfortunately, no defense mechanism is able to prevent or detect all of them. For this reason, occupational fraud schemes may continue for quite a long time until fraud is confirmed (with a median of 18 months before being detected) and fraud is more likely to be detected by a tip rather than by any other detection method (see Figure 6.3).



**Figure 6.3:** Detection of occupational fraud as presented in ACFE's 2012 Report to the Nations [AoCFE12].

Even though advanced information technology has been incorporated into organizations to reduce the risk of internal fraud, monitoring of business anti-fraud control systems remains a problem of outstanding difficulty and requires severe internal control. There exist several factors that hinder the identification of fraud e.g., lack of fraud symptom expertise and fraud-specific tools, heavy use of sampling, limited time and so on. This latter factor combined with the fact that the data generated by these systems are in textual form and their amount is not easily manageable in daily basis (not even in a weekly basis) enlarges the problem of immediate fraud detection. Information visualization can be promising in such a context, since it facilitates the quick identification of fraudulent activity.

### 6.3 Motivation and Problem Definition

Our motivation to work on fraud-detection problems rests on real problems raised by internal auditors of a large telecommunication organization. The internal auditors had communicated to us the problems which they face during the fraud detection procedure and the fraud patterns that they examine. The main problem which we have asked to cope with is the following:

**Problem 6.3.1.** *We seek for a combination of methods and visualizations in order to detect occupational fraud schemes in business systems in which a pair of entities (employee and client) are involved (e.g., billing, membership renewal systems and so on).*

Examining occupational fraud schemes in such systems reveals that events that occur in

regular time basis may be indications of fraud. For instance, in a billing system of a company, if a specific employee appears to have a monthly activity (deletes/ adds invoices, etc) towards an account of a customer, this should be considered as a suspicious periodic series of events that has to be further examined. These events become more important in the case where they occur within the billing cycle of the account of a client or outside the working hours of employees. Hence, searching for periodic activity in data is of importance in revealing occupational fraud schemes.

**Problem 6.3.2.** *We seek for a combination of methods and visualizations that identify periodic events that may conceal occupational fraud schemes in serial data stemming from business systems in which a pair of entities (employee and client) are involved.*

A closely related problem to the problem of fraud detection is monitoring the activity of the employees in the network of an organization. We mainly focus on detecting unusual employee activity such as unauthorized access to computers, business systems and accounts of employees or clients and so on.

**Problem 6.3.3.** *We seek for visualizations that monitor the behavior of the employees of an organization and detect patterns of unusual employee activity.*

## 6.4 Related Work

Fraud detection has been studied enough in the literature. So far, there exists a significant amount of research works implementing a number of methods such as data mining and statistics [CFPS99, BC00, KSM07], artificial intelligence [FCS95, GC98, Kos04, LHB03] or genetic algorithms [WRW98, HKLS07, DO11]. However, we will focus our presentation only in research works that are supported by visualizations. For reviews in fraud detection, the reader is referred to [BH02, KLSH04, ZZ04, YWW<sup>+</sup>07, HRRV08, Wan10].

The NASD Regulation Detection System (ADS) [KSH<sup>+</sup>98, Sen00, SGS<sup>+</sup>02] monitors trades and quotations in the Nasdaq stock market and tries to detect patterns and practices of violative activity. The system is equipped with a detection component using pattern matching techniques and a discovery component that employs data-mining. The results are visualized using various techniques. The Financial Crimes Enforcement Network AI System (FAIS) [GS95, SGW<sup>+</sup>95] is a system that detects money laundering in cash transaction data. The system operates in two modes: (i) the user-driven mode, in which ad-hoc queries are allowed and (ii) the data-driven mode that visualizes suspicious subjects or accounts based on predefined rules.

Chang et al. [CLG<sup>+</sup>08] presented WireVis, a system that uses interactive visualization of financial wire transactions and aims to detect financial fraud. Their approach combines interactivity and coordinated multiviews (a network view, heat-maps, a search-by-example tool and a visualization component) in order to reveal the relationship between accounts, keywords, time and patterns of activity. Huang et al. [HLN09] presented a framework of visual analytics for stock market security. In particular, 3D-treemaps are used for the visualization of current stock prices and for the identification of stocks that produce an unusual trading pattern. Eberle and Holder [EH09] introduced the graph-based anomaly detection model which uses information theoretic, probabilistic and maximum partial substructure approaches to detect structural anomalies in business transactions and processes propagated by employees. Luell [Lue10] combines data-mining and visualization techniques to detect client advisor fraud in a major financial institution.

Di Giacomo et al. [DDL10] proposed a system based on network visualization techniques in order to discover financial crimes. In particular, the system enables the identification of investigative patterns and hidden truths in data stemming from suspicious transaction reports (STRs). Didimo et al. [DLMP11] developed a system that supports the analyst with effective tools in order to discover financial crimes, like money laundering and frauds. SynerScope [Syn11] is a visualization tool for analyzing “Big Data” capable of detecting financial fraud schemes. Didimo et al. [DLM12] presented VIS4AUI, a system that collects financial information with regard to ongoing bank relationships and high value transactions and tries to detect money laundering and financial crimes.

The Link Analysis Workbench (LAW) [WBH<sup>+</sup>03] is developed to detect cases of terrorist and other criminal activity in noisy and incomplete data. The system applies pattern matching and graph-pattern matching techniques and is based on the matching of known patterns of interest and the discovery of new patterns of interest. Xu and Chen [XC05] presented a framework for automated criminal network analysis and visualization that helps at extracting valuable criminal network knowledge from large volumes of data. Stasko et al. [SGL08] developed a visual analytic system that facilitates analysts to examine reports and documents more efficiently in order to identify potential embedded threats. Commercial tools that apply visual network analysis in the context of crime detection are I2 Analyst’s Notebook [IAN], Netmap [Net], XANALYS Link Explorer [XLE].

For the identification of periodic patterns in serial data, Carlis and Konstan [CK98] suggested an approach in which serial attributes of data are represented along a spiral axis, while periodic ones along the radii of the spiral. Weber et al. [WAM01] presented an approach which uses spirals to visualize large sets of time-series data and reveals periodic structures. In their approach, the time axis is represented by the spiral and other features of the data are depicted

by points, colors, bars or lines. Bertini et al. [BHL07] presented SpiralView, a tool that visualizes on a spiral the distribution of network alarms over time and helps revealing potential periodic patterns. Suntinger et al. [SOSG08] developed a visualization system that represents events from event-based systems on a cylindrical tunnel. The visualization detects several incidents, such as particular patterns or irregularities that might affect business performance. Regarding the visualization of time-series data and the available techniques, an overview can be found in [ABM<sup>+</sup>07, MS03, SC00].

## 6.5 Methodology and Notation

One of the major problems we had to deal with during this research was the non-uniform format of the log files produced by business systems (e.g., different databases schemata, different log files, etc). To cope with this problem, we developed parsers that transform in a uniform format the log files produced the vast majority of systems of the organization. Then, we designed a database in which we store the formatted records.

Another problem we had to face, was the huge volume of the data that had to be visualized. Since occupational fraud schemes are well-hidden in the amount of data, we were seeking for an approach that would present to the auditor all recorded events according to their severity. Since the simultaneous visualization of large data-sets is confusing and inefficient, in our systems we introduced a combination of fraud detection mechanisms based on fraud patterns suggested by the auditors and diverse visualizations that support animations.

As already mentioned, the events on which we focus involve two entities (i.e., an employee and a client). Thus, a record in our system consists (among other secondary fields) of a time-stamp, an employee ID, a client ID and an action. Each record may have been generated by a call between an employee and a client, a transaction involving both entities and so on.

An *event*  $e$  involves a pair of employee-client and is defined by a 4-tuple  $(t, u, c, a)$ , where:

- $t$  is the time-stamp of the occurrence of the event,
- $u$  is the id of the employee,
- $c$  is the id of the client,
- $a$  is the action taken by the employee.

For a particular 4-tuple  $(t, u, c, a)$ , we say that client  $c$  is *related to* event  $e$  and is also *related to* employee  $u$ .



## 7 Periodicity Detection in Serial Data Through Visualization

In this chapter, we present visualization techniques that aim to identify periodic patterns in serial data which involve pairs of entities (employee and client). The presented techniques can be applied to detect occupational fraud schemes. Towards this direction, we measure periodicity by introducing a new metric which is appropriately defined to reveal the frequency in which events related to the same pair of employee-client occur within a time window. Our main visualization method results in drawings consisting of concentric circles whose radius correspond to the periodicity of the activity of each pair of employee and client. Events that are considered to be suspicious are easily identified since they are dragged towards the center of the circles. Also, a force-directed approach is employed in order to provide a better configuration of the events over the visualization.

### 7.1 Periodicity Estimation

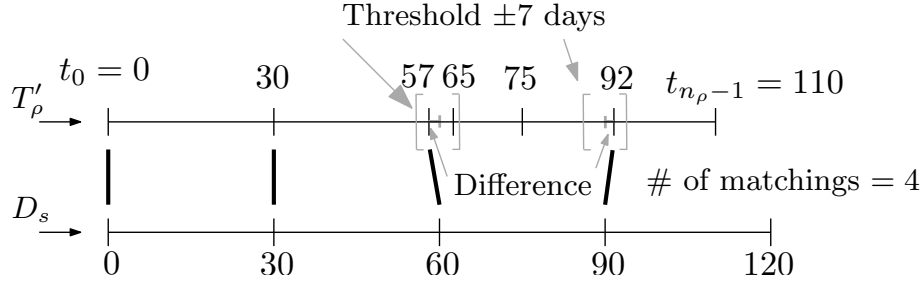
Let  $e = (t, u, c, a)$  be an event that involves employee  $u$  and client  $c$  (refer to Section 6.5). For a pair of employee-client  $(u, c)$ , an *event-series*  $T_{u,c} = \{e_{(u,c)}^1, e_{(u,c)}^2, \dots\}$  is a sequence of events  $e_{(u,c)}^i = (t_i, u, c, a_i)$  related to client  $c$  and employee  $u$  (we assume that it is ordered by date). In order to produce a visualization, the system preprocesses these series of events and estimates a proper period of activity for each pair of entities.

For each pair of entities (employee-client), we introduce a metric that estimates a periodicity value with a specific confidence degree. For simplicity reasons, let  $\rho = (u, c)$  be a pair of entities and  $n_\rho$  the number of events associated with pair  $\rho$ . Assume that  $e_\rho^i, e_\rho^{i+1}$ ,  $i = 1, \dots, n_\rho - 1$ , are two consecutive events and let  $d_\rho^{i,i+1}$  be their time distance (say measured in days). An event-series  $T'_\rho = (t_\rho^1, \dots, t_\rho^{n_\rho-1})$  is generated by assigning to each event  $e_\rho^i$  a value  $t_\rho^i$  according to the following formula:

$$t_\rho^i = \begin{cases} \sum_{j=1}^{i-1} d_\rho^{j,j+1} & \text{if } 1 \leq i < n_\rho \\ 0 & \text{if } i = 0 \end{cases}$$

For a given period value  $s$ , the ideal event-series  $D_s = (0, s, 2s, \dots)$  is defined by the time stamps that occur if the events between the entities of  $\rho$  appear in time intervals that

equal exactly to  $s$  (see Figure 7.1). For instance, in case of a period value of 30 days, the ideal event-series is  $D_{30} = (0, 30, 60, \dots)$ .



**Figure 7.1:** Line  $T'_\rho$  corresponds to the event-series of a pair of employee-client, whereas line  $D_s$  to the ideal event-series for a period of 30 days.

Let  $t \in T'_\rho$  and  $\lambda \in D_s$ . We say that  $t$  and  $\lambda$  *match each other with respect to a threshold value*  $\tau \in [1, s/2)$ , if it holds that  $t \in [\lambda - \tau, \lambda + \tau]$ . Let  $N_\rho^\tau$  be the set of time stamps of ideal event-series  $D_s$ , which can be matched with a time-stamp of event-series  $T'_\rho$ , i.e.,  $N_\rho^\tau = \{\lambda \in D_s : \exists t \in T'_\rho \text{ s.t., } t \text{ and } \lambda \text{ match}\}$ . With slight abuse of terminology, we refer to the cardinality of  $N_\rho^\tau$  as the *number of matchings* of pair  $\rho$ . Let also,  $diff_\rho^\tau : N_\rho^\tau \rightarrow \mathbb{R}$  with:

$$diff_\rho^\tau(\lambda) = \min\{|t - \lambda| : t \in T'_\rho, \lambda \in N_\rho^\tau \text{ and } (t, \lambda) \text{ match}\}$$

The confidence level of a pair  $\rho$  of entities with periodicity value  $s$  and threshold matching value  $\tau$  is given by the following formula:

$$confidence(\rho, s, \tau) = \frac{\sum_{\lambda \in N_\rho^\tau} 1 - \frac{diff_\rho^\tau(\lambda)}{\tau}}{|N_\rho^\tau|}$$

Observe that the confidence values belong in  $[0, 1]$ . Obviously, if the event-series  $T'_\rho$  is identified with the ideal event-series  $D_s$ , then  $confidence(\rho, s, \tau) = 1, \forall \tau \in [1, s/2)$ . In order to provide a more accurate estimation of the confidence value of a pair  $\rho$  for a given period  $s$ , with respect to a threshold value  $\tau$ , one can alternatively compute the confidence value of  $\rho$  for all ideal event-series  $D_s^i = (i, s + i, \dots), i = 0, \dots, s/2$  and, keep the one that maximizes the confidence value.

For a given pair of entities  $\rho$  and a predefined threshold value  $\tau$ , we measure its confidence for all periodicity values  $s \in [1, s_{max}]$ , where  $s_{max}$  corresponds to the maximum periodicity value defined by the auditor. Having determined all confidence values of  $\rho$ , the *periodicity* of pair  $\rho$  (with respect to the specified threshold value  $\tau$ ) equals to the periodicity value that maximizes its confidence value.



Note that, for each pair of entities, we produce a visualization similar to the one of Figure 7.1 in order to present the series of events associated with the specific pair and the matchings with the ideal event-series.

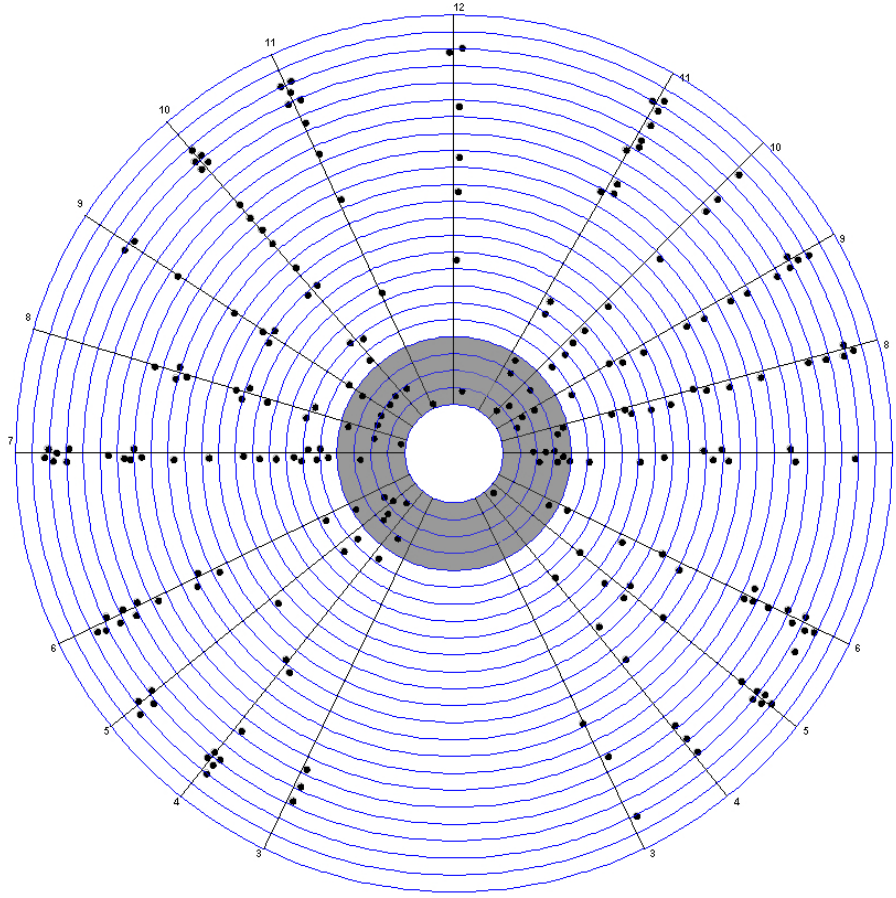
## 7.2 Periodicity Visualization

The main visualization is illustrated in Figure 7.2, where we seek for monthly periodic activity i.e., periodicity of 30 days. It consists of a system of concentric circles whose radius correspond to different periodicity values. The nodes of the visualization correspond to pairs of entities. The outermost circle corresponds to a period of 8 days, while the innermost to 31. We only compute periodicity values that are greater than the threshold value, which in the visualization of Figure 7.2 is set to 7 days. However, this is a value determined by the auditor. With this configuration, nodes with periodicity of 30 or 31 days are dragged towards the center of the system.

The system is also split in circular sectors that correspond to different number of matchings with the ideal event-series for each period, as discussed above. For a more uniform arrangement of the nodes in the system of circles, nodes whose number of matchings is greater than the median value lie on the upper semicircle, while the remaining ones at the bottom. The maximum number of matchings corresponds to the midpoint of each upper semicircle. In the visualization, we ignore nodes whose event-series had up to two matchings with the ideal event-series since otherwise, we always have a perfect sequence of value two. The gray-colored areas of Figure 7.2 contain nodes that appear to have suspicious activity (due to their periodicity values) and need to be further examined. We have also chosen to highlight the entire ring of periods greater than 27, even in cases with few matchings, since this may reveal a suspicious behavior that is about to start.

In the case where, the auditor selects a node, all nodes of the same employee are drawn by the same color or shape. In this manner, the auditor can identify whether the employee appears to have a continuous suspicious activity. Also, the most suspicious nodes can be drawn with different colors or shapes, such that they can be easily distinguished from the remaining ones, as in Figure 7.3.

In order to improve the legibility of the produced drawings, we have used a variation of the classical force-directed algorithm of Eades [Ead84] in conjunction with our visualization technique. As mentioned in Section 1.5.2, a force-directed algorithm models the vertices of the graph as electrically charged particles that repel each other and its edges by springs in order to attract adjacent vertices. However, before we proceed with the detailed description of the algorithm, we introduce some necessary notation. Given a drawing  $\Gamma(G)$  of graph  $G$ ,



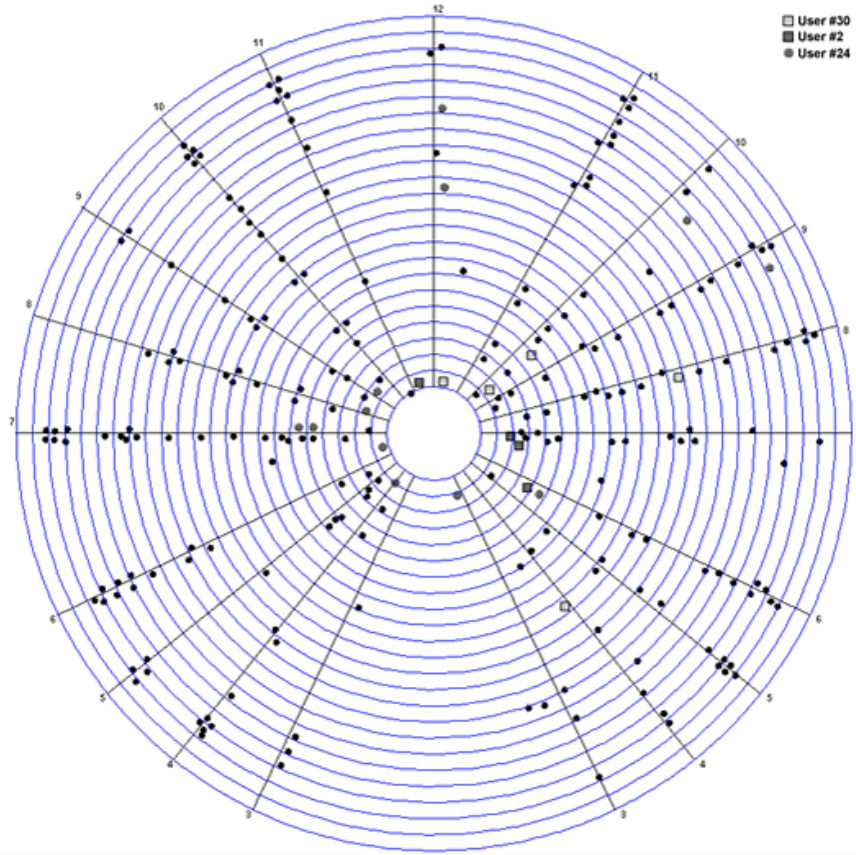
**Figure 7.2:** A concentric circle system in which each radius correspond to a periodicity value. The gray-colored areas are the ones that have to be examined first for suspicious activities.

we denote by  $p_u = (x(u), y(u))$  the position of node  $u \in V$  on the plane. As in Section 5.1, the unit length vector from  $p_u$  to  $p_v$  is denoted by  $\overrightarrow{p_u p_v}$ , where  $u, v \in V$ .

In our approach, we add dummy nodes on each circle and along the lines that splits these circles in circular sectors. Each dummy node corresponds to the number of matchings for a given periodicity. Then, we use springs to connect each node with the dummy node of its period circle that corresponds to its number of matchings. The springs follow the logarithmic law instead of the Hooke's law in order to avoid exerting strong forces on distant nodes. The attractive forces follow the formula:

$$\mathcal{F}_{spring}(p_u, p_v) = C \cdot \log \frac{\|p_u - p_v\|}{\ell} \cdot \overrightarrow{p_u p_v}, (u, v) \in E$$

where  $C$  and  $\ell$  capture the *stiffness* and the *natural length* of the springs, respectively. We also, use repulsive forces among the nodes of the visualization in order to avoid node overlaps. The



**Figure 7.3:** The top three most dangerous entities are illustrated with different colors and shapes.

repulsive forces are defined as follows:

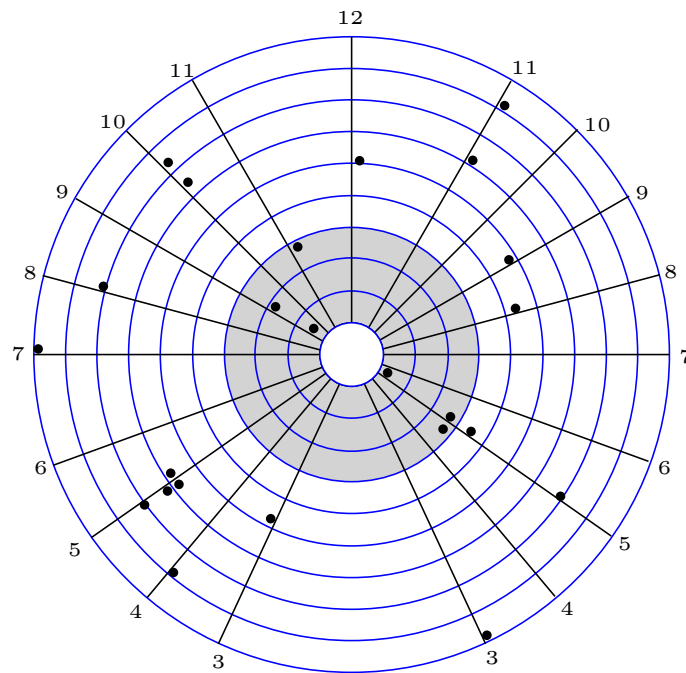
$$\mathcal{F}_{rep}(p_u, p_v) = \frac{C_p}{\|p_u - p_v\|^2} \cdot \overrightarrow{p_u p_v}, \quad u, v \in V$$

where  $C_p$  is a *repulsion* constant. The set of forces that were described assure that in an equilibrium state of the model, the nodes will be eventually drawn close to their associated periodicity circles and more precisely, close to the dummy nodes that “describe” their number of matchings. Note that, we do not apply forces on the dummy nodes. Hence, their positions remain unchanged.

### 7.3 Single Period Visualization

In order to have a better insight of the nodes that lie on a specific period ring (e.g., when examining activities in a period of 30 days), we propose a visualization based again on concentric circles (similar to the one mentioned above; see Figure 7.4) that contains nodes of a

specific period value. In this case, the radii of the concentric circles correspond to different degrees of confidence. The outermost circle corresponds to confidence value 0.1, while the innermost to 1. Hence, nodes for which the confidence value tends to 1 lie towards the center of the system. As above, the visualization is split in circular sectors based on the estimated number of matchings and simultaneously supports all functionalities of the previous visualization. Again, the final layout is computed using a force-directed algorithm that is a simple variation of the one described in Section 7.2.



**Figure 7.4:** A concentric circle visualization for a period of 30 days. The radii of each circle correspond to different confidence values. Nodes with confidence value 1 move towards the center of the system

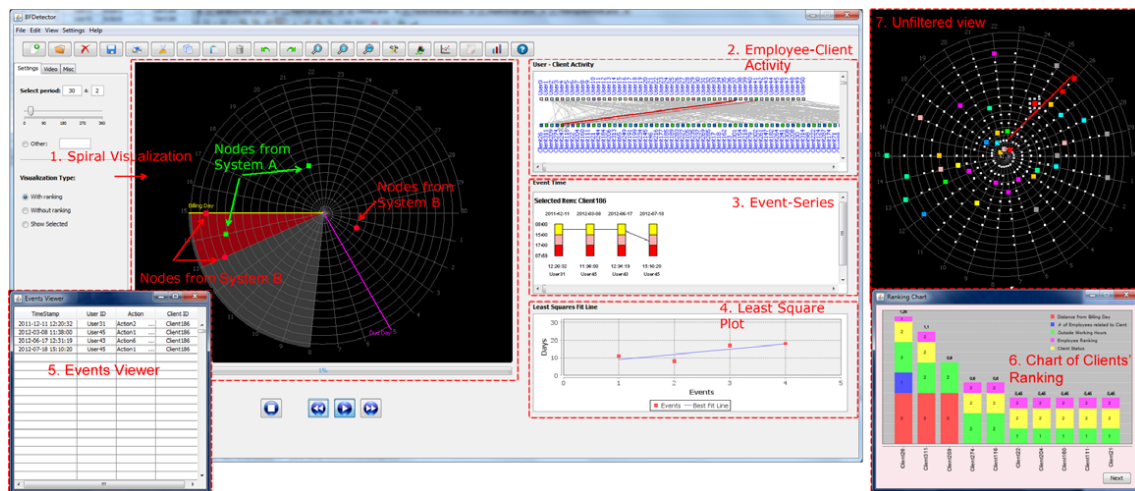
8

## A Spiral-Based Fraud Detection System

In this chapter, we exploit the advantages of information visualization and present a system that aims to detect (i) occupational fraud in systems which involve a pair of entities (e.g., an employee and a client) and (ii) periodic activity. The main visualization is based on a spiral system on which the events are drawn appropriately according to their time-stamp. Suspicious events are considered those which appear along the same radius or on close radii of the spiral. Before producing the visualization, the system ranks both involved entities according to the specifications of the internal auditor and generates a video file of the activity such that events with strong evidence of fraud appear first in the video. The system is also equipped with several different visualizations and mechanisms in order to meet the requirements of an internal fraud detection system.

### 8.1 Main Features of the System

Figure 8.1 illustrates a snapshot of our system. The base of our system is a spiral visualization on which the time-stamp of each event is appropriately represented. The main advantage of spiral visualizations is that potential periodic patterns can be quickly identified since they appear along a radius or on close radii of the spiral. Our system consists of several coordinated visualization windows, each dedicated to a particular aspect of audit data. Panel #2 of Figure 8.1 illustrates the total activity of employees and clients. Panel #3 of Figure 8.1 demonstrates the time and the dates for each event related to a client using different colors to identify access during specific time intervals of a day. This visualization contributes to the quick identification of events that appear outside the employees' working hours or on holidays, which may be indications of fraud. Panel #4 of Figure 8.1 is a least square plot where the  $y$ -axis corresponds to the days of a month and which aims to detect periodicity. In the case where the plotted line is "almost" parallel to  $x$ -axis and the points are "close" to this line, this implies that there exists a periodic pattern related to a day of the month. There also exists an event-viewer (refer to Panel #5 of Figure 8.1) that illustrates the initial input data and interacts with the visualization.



**Figure 8.1:** A snapshot of the interface of the system.

As already stated in Section 6.5, the proposed system has been developed based on feedback provided by internal auditors at a major telecommunication company. The main obstacle reported while trying to detect occupational fraud is the amount of data that is usually generated from more than one business systems and has to be investigated manually by writing and executing scripts. For this reason, we have tried to develop a system that quickly detects periodic patterns in data and incorporates several of the common patterns that are investigated by the auditors in order to detect occupational fraud. In addition, since occupational fraud is a sensitive issue, auditors have also emphasized the necessity to have a tool that is able to quickly confirm or reject their suspicions about the activity of an employee. The proposed system can also be used for this purpose.

The system is user-oriented and the visualization can be adapted appropriately such that it depicts the patterns that are investigated by the auditor. The innovation of this tool is that it aggregates the total activity of each employee and client and ranks both of them according to specifications defined by the internal auditor. Based on this ranking the system produces a video file. Frames are ordered such that those with strong indications of fraud appear first in the video.

The system is also equipped with supplementary visualizations that provide information about the activity of the employees and clients. In order to meet the requirements of an internal auditor, the system supports supplementary functionalities such as filtering, export log mechanisms, storing, reloading and post-processing of data. It provides also advanced graphic functionality, including popup menus, printing capabilities, custom zoom, fit-in window, selection, dragging and resizing of objects.

Their major requirement of the auditors was to design a system that reveals reoccurring

activity between pairs of employee-clients. For this reason, we have adopted the spiral visualization that enables serial data visualization. However, in contrast with the work of Bertini et al. [BHL07] that used the spiral in order to visualize events daily, the data-sets that we had to process cover a much larger time interval and had to be visualized simultaneously. This fact combined with the large volume of the data-set had to be faced efficiently in order to avoid cluttering the visualization. Also, unlike to the data-sets used for financial crime detection where there exists a “hint” on suspicious cases or transactions, in our data-sets, in most of the cases, it is highly unlikely to have prior knowledge on suspicious cases. Hence, it is difficult and maybe “risky” to try to filter the data before producing the visualization. This motivated us to design the ranking procedure in order to distinguish suspicious cases that have to be further examined and try to adapt it to the needs of the auditors. If compared with other existing systems for financial crime detection or employee fraud mentioned in Section 6.4, our system (i) produces a video file containing activity of clients-employees according to the time interval selected by the auditors, (ii) distinguishes suspicious events based on the ranking factors selected by the auditor and distributes them in the video frames such that they are quickly identified and (iii) supports multiple coordinated views that facilitate the investigation and reveal potential periodic activity.

## 8.2 Overview of the Detection Procedure

In this section, we present an overview of the fraud detection procedure as supported by our system. As already stated in Section 6.5, the system takes as input data consisting of log files or sets of database records generated by systems which involve pairs of entities. The system ranks the clients and the employees according to rules specified by the auditor and creates a video with all the activity of a client. By default, the ranking is calculated based on the entire database, unless the auditor specifies a desired time-interval. In the visualization, all potentially suspicious incidents are detected and can be further investigated by the auditor, who makes use of the system’s accompanied visualizations. Before we proceed with the detailed description of the system, we introduce some terminology necessary for the description of the ranking procedure. Let  $e = (t, u, c, a)$  be an event that involves employee  $u$  and client  $c$  (refer to Section 6.5). For a client  $c$ , an *event-series*  $T_c = \{e_1^c, e_2^c, \dots\}$  is a sequence of events  $e_i^c = (t_i, u_i, c, a_i)$  related to client  $c$ . Note that, such an event-series consists of events which may involve more than one employee.

## 8.3 Ranking Procedure Description

In general, the ranking of a client is based on several factors such as the number of events close to the billing date, the actions taken by the employees and so on, which are described in detail in Section 8.3.1. The ranking of an employee is based on the ranking of the clients that are related to the specific employee. Note that, since a company may have thousands of clients, in the ranking procedure we take into consideration only the ones for which there exists an event generated by a log mechanism (i.e., not all registered clients of the business systems of a company).

### 8.3.1 Client Ranking Function

In order to rank clients, the system analyzes the event-series that correspond to each client based on factors defined by the auditor. In the system, we have incorporated several of the queries commonly used by internal auditors, while seeking for occupational fraud.

Let  $N$  be the number of distinct factors considered in the ranking calculation. Let also  $a_f^c$  be the “performance” of client  $c$  at factor  $f$ . According to the severity (low, medium, high) of the corresponding event-series,  $a_f^c$  equals to zero, one or two, respectively. Ranking  $R_c$  of client  $c$  is defined as follows:

$$R_c = \sum_{f=1}^N a_f^c \cdot w_f,$$

where  $w_f$  is the weight of importance for factor  $f$ .

The weights of factors  $w_f, f = 1, \dots, N$  are determined by the auditor, who also specifies an ordering among them that best fits to what he/she is seeking for. For instance, if the auditor is interested in events that occur outside the employee’s working hours, the corresponding factor should be ranked first, which implies that the weight of the corresponding factor should be greater than the weights of other factors supported by the system. Given a factor ordering, weights  $w_f, f = 1, \dots, N$  of the factors are calculated based on a formula proposed by Stillwell et al. [SSE81], as follows:

$$w_f = \frac{N - r_f + 1}{\sum_{j=1}^N (N - r_j + 1)},$$

where  $r_f$  is the rank position of factor  $f$  in the factor ordering.

In the following, we describe the factors that are currently supported by the system. For each of these factors, we define three classes of clients according to the severity (low, medium, high) of event-series  $T_c$  corresponding to client  $c$ . Then, performance  $a_f^c$  of client  $c$  on factor  $f$  is defined by the following formula:



$$a_f^c = \begin{cases} 2, & T_c \in \text{High Severity Class for factor } f \\ 1, & T_c \in \text{Medium Severity Class for factor } f \\ 0, & T_c \in \text{Low Severity Class for factor } f \end{cases}$$

### 8.3.1.1 Distance from Billing Date

Experience on examining occupational fraud schemes has shown that events related to the same pair of employee-client that appear on a monthly basis and before the billing date of a client's invoice may be strong indications of fraud. Given an event-series  $T_c$  corresponding to client  $c$ , the system detects events whose time-stamp is close to the billing date of client  $c$ . Based on the number of such events, the system calculates the severity of event-series  $T_c$ . In the case where the time-stamp of an event occurs after the billing date of a particular month, we consider this as an incident that concerns next month's activity.

Let  $e$  be an arbitrary event and let  $t_e$  be its time-stamp. Let also  $t'_e$  be the billing date that immediately follows  $t_e$ . Then, the distance of event  $e$  from the billing date, denoted by  $d_e$ , is defined by the number of days between  $t_e$  and  $t'_e$ .

Let  $D_c^0$  be the set of events which occur within distance of three days from the billing date, i.e.,  $D_c^0 = \{e^c \in T_c : d_e \leq 3\}$  and  $|D_c^0|$  its cardinality. Similarly, we define the set of events which occur within distance greater than three and less than seven days from the billing date, i.e.,  $D_c^1 = \{e^c \in T_c : 3 < d_e \leq 7\}$  and the set of events  $D_c^2 = \{e^c \in T_c : d_e > 7\}$  with distance more than seven days from the billing date. The end-points of the above investigated time-intervals can be adjusted, if desired, by the auditor. The evaluation of the importance of this factor is based on a classification of event-series  $T_c$  in one of the following severity classes according to the number of events occurred close to the billing date. More precisely, event-series  $T_c$  related to client  $c$ , belongs to this class if:

*High Severity Class:* This class includes event-series with severe indications of fraud for which it holds one of the following:

- $|D_c^0| \geq 2$ : This implies that there exist at least two events related to client  $c$  too close to the billing date. To minimize false-positives, cases where there exists only one event too close to the billing date are not considered by the system of high-severity, since they may have occurred by coincidence. However, they are classified to a medium-severity class in order to be further investigated.
- $|D_c^1| \geq 3$ : We included this case in high severity class, since the specific event-series contains an "unusual" number of events within distance of one week from the billing date. Again, the system tries to minimize false-positives by taking into consideration

events that occurred within the interval of  $(3, 7]$  days from the billing date at least three times.

- $|D_c^0| + |D_c^1| \geq 2$ : In this case, the system takes into consideration the total number of events occurred within distance of one week from the billing date.

*Medium Severity Class:* In this class, we consider event-series for which it holds one of the following:

- $|D_c^0| = 1$  or  $|D_c^1| = 1$  or  $|D_c^1| = 2$ : These cases were excluded from the high-severity class in order to minimize false-positives. However, they have to be investigated since these may imply that malicious activity has just begun.
- $|D_c^2| > thres$ , where *thres* is a threshold defined by the auditor. By default, this value is 5. This implies that there exists a continuous activity concerning client  $c$ , which may have to be further investigated.

*Low Severity Class:* All other cases.

Similarly, one can define a factor regarding the due date of the invoice of a client. Again, in this case we are interested in events that occur before the due date. However, it is recommended that the auditor does not use simultaneously the “distance from billing date” factor and the “distance from due date” factor, since there exist overlaps between the investigated intervals which may create false-positives.

### 8.3.1.2 Event-series Periodicity

Given an event-series  $T_c$  related to client  $c$ , we calculate the *proper period of activity* based on the algorithm presented in Section 7.1 (assuming that the events of the event-series are ordered according to their time-stamps). Since internal auditors are interested in events that appear on monthly basis, the system evaluates the severity of the event-series as follows:

*High Severity Class:* Event-series with period  $p$  such that of  $27 \leq p \leq 30$  or 31 days.

*Medium Severity Class:* Event-series with period  $p$  such that of  $20 \leq p < 27$  days.

*Low Severity Class:* All other cases.

As previously, the auditor can adjust the values that determine the intervals for each of the above classes.

### 8.3.1.3 Events Occurring Outside Working Hours

Given an event-series related to a particular client, the system tries to detect suspicious cases occurred within a day. Events of high-severity are the ones that occur outside the employee’s working hours, on weekends or holidays. Furthermore, events of medium-severity

are considered those that occur at the end of the employee's shift. We have assumed that this case corresponds to the last two hours of employee's shift. However, this value can be appropriately adjusted by the auditor. The system takes as an input the working hours of each employee and also takes into consideration weekends and holidays. The classification of a client based on the time-stamps of the related events is performed as follows:

*High Severity Class:* There exists at least one event occurred outside working hours, on weekends or holidays.

*Medium Severity Class:* There exists at least two events at the end of employees' shift. As previously, the system requires at least two such events to include an event-series in this class in order to minimize false-positives.

*Low Severity Class:* All other cases.

Again, the auditor can appropriately adjust the values that define the above classes.

#### 8.3.1.4 Number of Employees Related to a Client

This factor indicates the number of employees that are related to a specific client. Normally, it is expected that distinct events are related to several distinct employees. Due to the fact that, usually a randomly selected employee handles a client request, having the same employee handling multiple requests of the same client may be an indication of fraud. The classification of the event-series related to client  $c$  based on this factor, is the following:

*High Severity Class:* An employee handles more than 50% of the events related to client  $c$ .

*Medium Severity Class:* Two or three employees handle more than 50% of the events related to client  $c$ .

*Low Severity Class:* All other cases.

The auditor can also adjust the percentages that define the above classes. However, there may exist cases where for instance, a client calls the company for an issue regarding his/her account and always asks for the same employee. This obviously does not consist fraud and the system would provide a false-positive if this factor was the only factor applied for ranking. Hence, it is recommended to be applied in conjunction with other factors.

#### 8.3.1.5 Action Name

Each company has its own rules regarding the employees that use the business systems and supports different privileges for different employees. Hence, there exists a list of actions that may be forbidden for all or for unauthorized employees. In addition, there exist actions which are suspicious, even though an employee may be authorized to perform. Furthermore, there exist actions that may be correlated (e.g., open-close action) and it is uncommon if one of them does not appear in the event-series. Thus, since there exist several rules in order

to detect fraud in diverse business systems, the auditor in our implementation has to adjust the rules of each severity class on the corresponding panel of the system. An overview of the classification based on this factor is the following:

*High Severity Class:* Actions that are forbidden for unauthorized employees.

*Medium Severity Class:* Actions that are considered to be suspicious, as described above.

*Low Severity Class:* All other actions.

### 8.3.1.6 Client Status

When ranking a client, it is important to take into consideration its corresponding background history. This implies that a client for which there existed evidence of fraud will be ranked higher. The auditor is able to mark a client as (i) blacklisted, if a previous investigation led to evidence of fraud, (ii) suspect, if a previous investigation is ongoing or unresolved, or (iii) cleared, if suspicions of fraud either do not exist in the system or were not confirmed. According to this marking, we define the following classification:

*High Severity Class:* The client is blacklisted.

*Medium Severity Class:* The client is a suspect.

*Low Severity Class:* The client is cleared.

### 8.3.2 Employee Ranking

As mentioned in Section 8.2, the system ranks the employees based on several aspects of their relation with their clients. Consider an arbitrary employee  $u$  and let  $S_u$  be the set of clients that are served by employee  $u$ , i.e.,  $S_u = \{c : \exists \text{ event } e = (t, u, c, a)\}$ . Let also  $R_u = \{R_c : c \in S_u\}$  be the set containing the rankings of these clients. By default, the system assigns to the employee the value that corresponds to the maximum ranking of set  $R_u$ . This implies that if an employee is related to a “suspicious” client, then he/she will be also considered as “suspicious”. An alternative could be to consider the clients with rankings greater than a threshold defined by the auditor.

## 8.4 Description of the System

In this section, we describe in detail our fraud detection system. The system operates in two modes, either *off-line* or *semi-online*. In brief, in the off-line mode the system parses static data concerning a period of time (e.g., the data of a week) and provides corresponding visualizations. The semi-online mode can be used on a daily basis in order to visualize the daily activity of the employees and clients. In both modes, the system provides interactive visualizations that help in the detection of suspicious events. Visualizations of large data-sets

may not be useful in certain cases. To cope with this problem, the auditor is able to specify a time-window and then, the system visualizes events whose time-stamp belongs to the query window. However, ranking can be estimated either based on the whole data-set that includes data from a much longer period of time or on data occurring in the specific time window, according to the specifications of the auditor.

### 8.4.1 Off-line Mode

As mentioned above, the system consists of multiple coordinated views and each of them visualizes a different aspect of the audit-data.

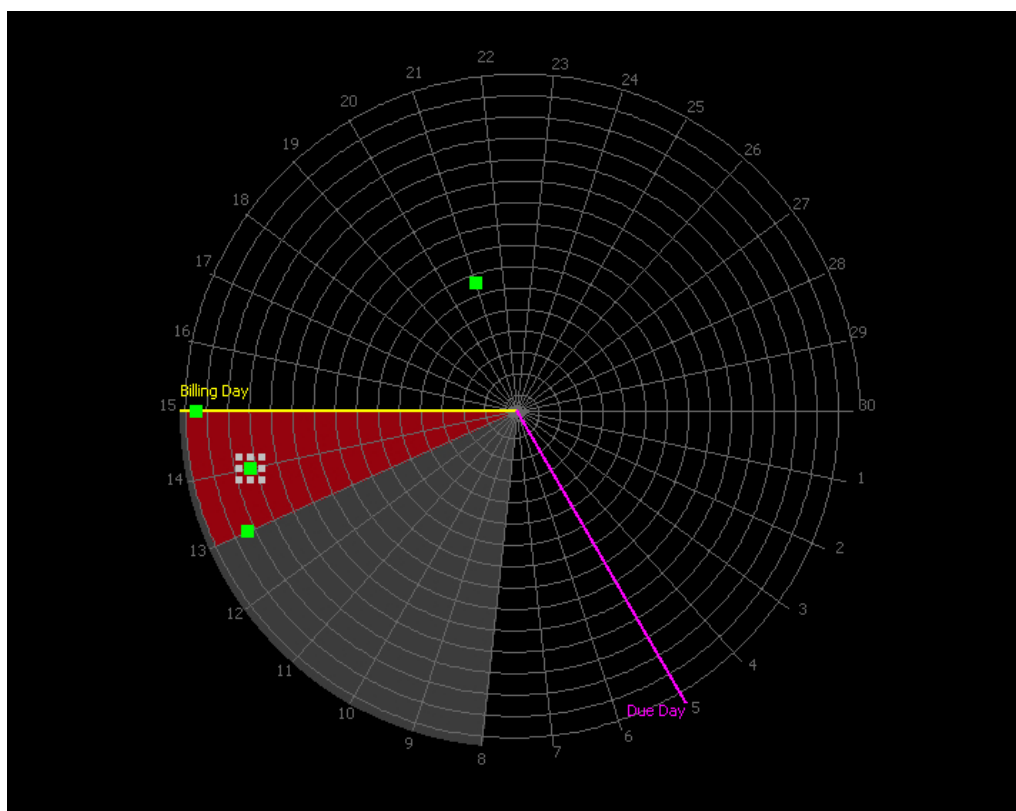
#### 8.4.1.1 The Spiral Visualization

A snapshot of the system in off-line mode is illustrated in Figure 8.2. In the spiral visualization, each spiral branch visualizes a period of one month, while the number of spiral branches is related to the first and last time-stamp of the input data (if not alternatively selected by the auditor), starting from the first month that coincides with the inner branch of the spiral. Each spiral branch is split by a number of lines according to the periodicity value that is examined (i.e., 7 days, 15 days, 30 days, etc.) and each line corresponds to a day of a month. The default value is 30 days, which implies that the auditor seeks for monthly suspicious activity (see Figure 8.2).

In the spiral window we place nodes, where each node represents an event related to a client and its position is determined based on the corresponding time-stamp. Nodes of different colors represent events related to different clients. To produce the spiral visualization, the system ignores multiple appearances of events that correspond to the same pair of employee-client at the same date. According to the spiral structure, events related to the same client and appearing along a radius of a spiral are considered suspicious and need to be further examined. However, examining cases of fraud has revealed that suspicious events may not always appear on the same date from month to month and thus, suspicious events may appear on close radii. These should also be considered as suspicious.

Employees and clients will be ranked according to the specifications of the auditor and the ranking function mentioned in Section 8.3. Based on the ranking, the system generates a video file in which each frame depicts the activity of a client within the specified time interval, giving priority to the ones with the higher ranking. The ordering of the frames guarantees that clients that are considered to be suspicious will not be skipped during processing; even in cases of large data sets they will be immediately distinguished.

The auditor is able to pause the video in order to further investigate the activity of a client. By default, on each frame the billing and the due date of the corresponding client are depicted



**Figure 8.2:** The main visualization when the system operates in off-line mode.

on the visualization (see Figure 8.2). The light-gray colored region of Figure 8.2 corresponds to the “dangerous” interval of a week before the billing date. The red-colored region of Figure 8.2 indicates that there exist events from month to month that differ by less than 3 days. If necessary, the auditor is able to visualize nodes that involve the same employee by the same color. By these features, the auditor quickly identifies events that occur close to the billing/-due date and potential periodic patterns for a specific client.

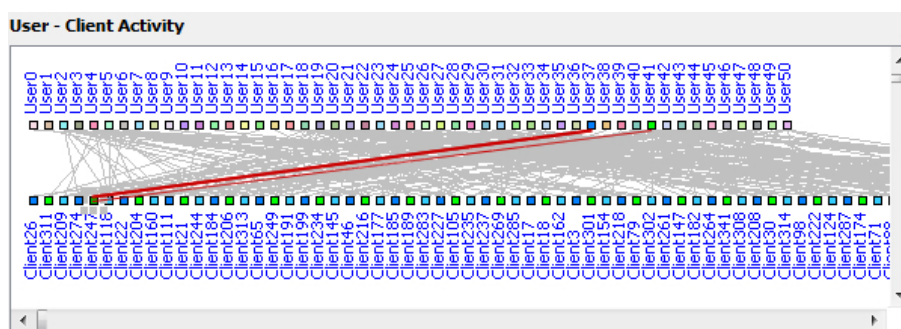
Filtering techniques are also supported by the system. The ranking factors described in Section 8.3 can also be used as filters while the results can be exported in separate log files. For instance, from a single frame, the auditor can select only the nodes representing events occurred outside the employee’s working hours. The auditor can also perform custom queries to the database which is a fundamental functionality for fraud detection. Optionally, the system is able to save a produced visualization in a file for the case where post-processing is required. It also maintains records about the employees/clients activities and their ranking.

As already mentioned, by default, the ordering of the frames is based on the ranking assignment. However, the nodes of the visualization can be distributed on the frames according

to an ordering specified by the auditor, which may be based on predefined knowledge about a client or on a list of clients already marked by the auditor from a previous investigation.

#### 8.4.1.2 Supplementary Visualizations

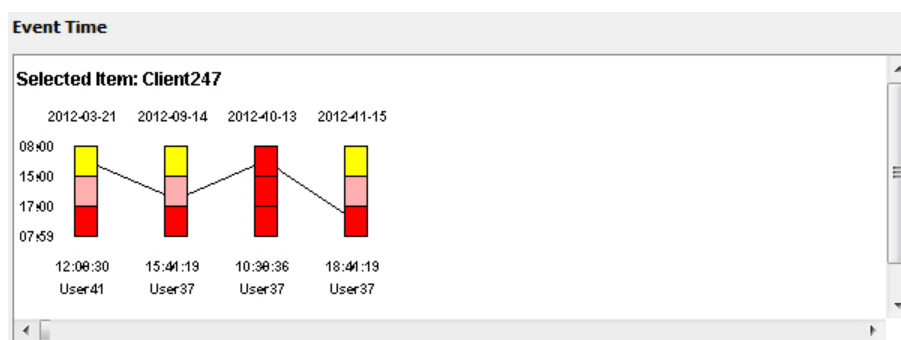
We proceed to describe the supplementary visualizations of the system. Figure 8.3 depicts a 2-layer visualization representing the total employee-client activity of the input data. The upper layer corresponds to distinct employees, while the bottom one to distinct clients. The ordering of the clients at the bottom layer is according to their ranking assignment. The node coloring follows the one used for each client in the spiral visualization. This visualization contributes to quickly identify pairs of employees-clients that appear to be involved in many events and simultaneously gives an overview of the total activity of the entities. Also, it demonstrates employees involved with many “suspicious” clients. The visualization interacts with the spiral drawing such that when a pair of employee-client is selected from the spiral drawing, it is also, marked in the layered visualization and vice-versa. In the case where there exist more than one event related to the same pair of employee-client the thickness of the corresponding edge becomes larger. Optionally, the visualization can be filtered such that only the client that is displayed in the video along with its related employees is visualized.



**Figure 8.3:** A 2-layered visualization representing the total activity of employees and clients.

In the visualization depicted in Figure 8.4, the event-series related to a specified client is represented by a line-graph. Each node of the drawing corresponds to the day of occurrence of an event. Each such node is split into time intervals that correspond to the hours of the day. The middle part (refer to the pink-colored region of Figure 8.4) corresponds to the end of the shift of the employee (i.e., two last hours of the shift). The upper part (refer to the yellow-colored region of Figure 8.4) corresponds to the employee’s time-shift having excluded the last two hours of the shift, while the bottom part (refer to red-colored regions of Figure 8.4) to non-working hours of a day. The endpoints of an edge touch the parts that correspond to the time-stamp of the events. Note that, for the spiral visualization the system

ignores multiple events that correspond to the same pair of employee-client. For the visualization of Figure 8.4, in the case where multiple events for the same pair of employee-client occur within a day, multiple nodes will be drawn and will be bounded by a rectangle in order to be distinguished. The system optionally takes as an input the shifts for each employee and makes the proper adjustments to the visualization. Also, since weekends and holidays can be taken under consideration by the system, if such cases occur the corresponding nodes are entirely colored in red (refer to the red-colored node of Figure 8.4).

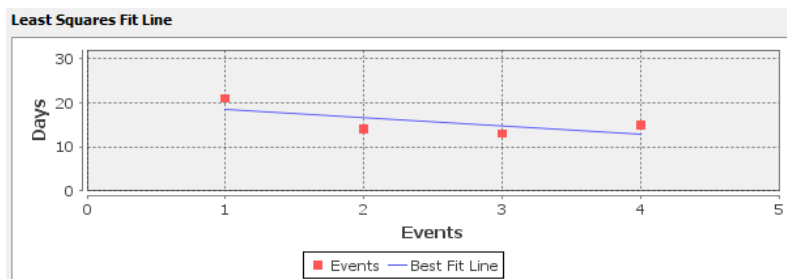


**Figure 8.4:** A visualization that depicts an event-series related to a client and distinguishes the time that an event occurs within a day.

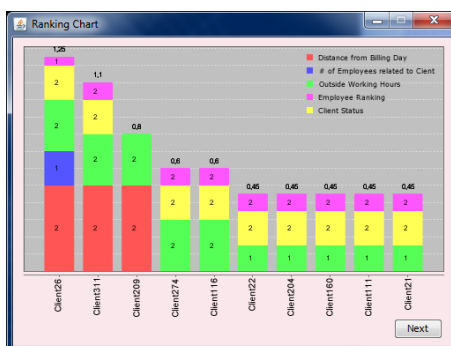
Given the event-series of a client, the system provides a plot where each point  $(x, y)$  represents the day  $y$  of the period interval that event  $x$  occurred (see Figure 8.5). Then, using the least squares method [DS81], the line that best fits to the data-set is calculated and plotted. Cases where the slope of the line tends to zero (i.e, almost parallel to  $x$ -axis) and points are close to the line (the model fits well to the point-set) indicate that most of the events appear close to the same day of the month. This implies that there exists a “suspicious” periodic pattern (e.g., close to day 15). Studying the least-square plot, we have to take into consideration cases where the calculated line is “almost” parallel to  $x$ -axis but, the points are not close to it. In this case, the impression of the periodicity is fictionally, since the line does not fit well to the point-set. However, these cases can be distinguished quickly either visually or by taking into consideration the least-square model error.

Note that, all the above visualizations are updated while the video frames change. This ensures that the auditor has a full view of the activity of each client without changing screens or drawbacks. A supplementary panel, as the one at the left-most part of Figure 8.1, demonstrates the database records related to the client in textual form such that the auditor does not have to recall database records in order to see the initial input. The panel also, interacts with the spiral visualization such that when selecting a node from the spiral the corresponding event in the panel is selected and vice-versa.





**Figure 8.5:** A least-square plot that indicates whether there exist periodic events within a time interval.

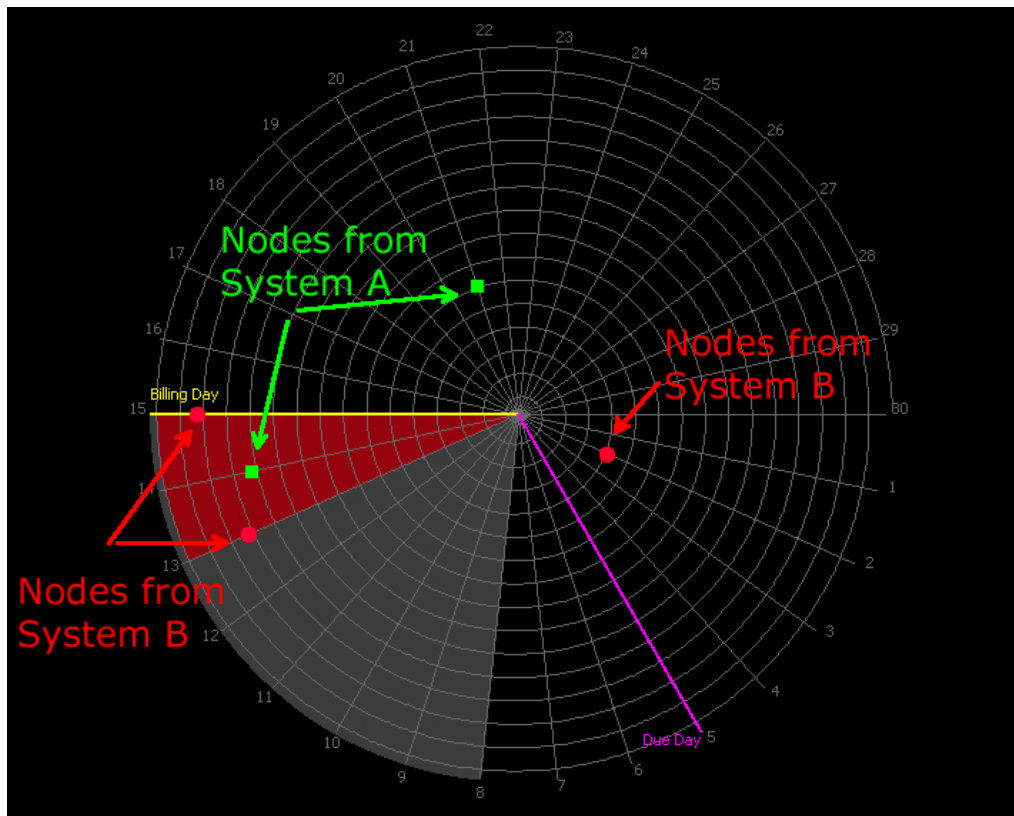


**Figure 8.6:** Client ranking based on 5 factors. Ten most highly-ranked clients are presented.

Another feature of the system is that it provides a stacked bar plot demonstrating the ranking of the clients as illustrated in Figure 8.6. Each bar corresponds to a client and is split into regions that represent each of the factors used for client ranking, while their lengths are proportional to the performance of the client on this factor. Also, in each of these regions the performance of the client on this factor (i.e., 0, 1, or 2) is illustrated. The plot of Figure 8.6, demonstrates the ten highly-ranked clients based on five factors. The first bar indicates that Client-26 was in the High-Severity Class (refer to Section 8.3) to three out of the five factors calculated, since client’s performance was 2 on Factors 1, 3 and 4.

Feedback provided from internal auditors added to our system another important functionality when searching for fraud. Sometimes, in order to detect fraud it is necessary to trace suspicious events to more than one business systems. For this reason, the auditor is able to select a client and load its activity from more than one such systems. In this case, the nodes of the visualization are drawn with different shapes and colors. In the visualization of Figure 8.7, green rectangular nodes correspond to a business system, say A, and seem to have no periodic activity. However, if red nodes stemming from a different business system, say B, are loaded to the visualization, it becomes clear that there exists periodicity in the events-series,

which may imply fraud.

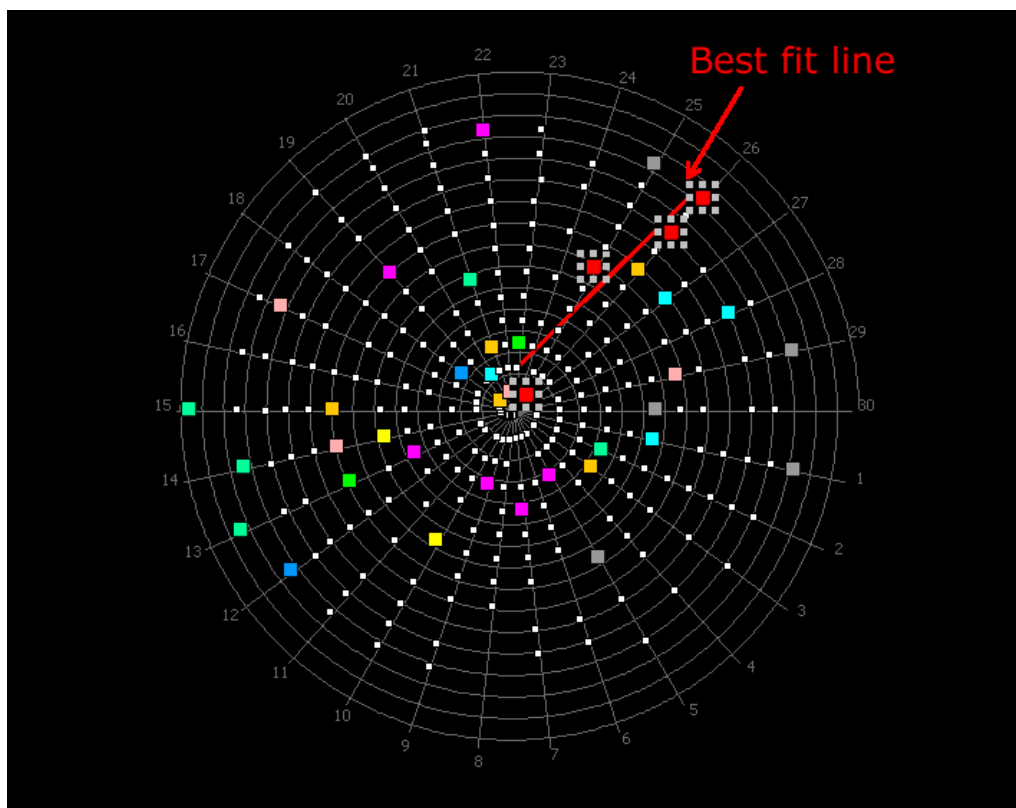


**Figure 8.7:** Different coloring and shape corresponds to events generated from different business systems. Their combination in the visualization reveals the periodic activity and probably fraud.

The system also provides an unfiltered view of the processed data, as illustrated in Figure 8.8. Colored nodes correspond to ten highly-ranked clients. Nodes that correspond to the same client will be represented by the same color. However, since it is difficult to distinguish suspicious behavior in such visualizations, the system optionally draws a line that best fits to the activity of the selected client. Filtering techniques like the ones mentioned above are also supported. By default, the visualization is filtered such that clients related to only one event are excluded.

#### 8.4.2 Semi-online Mode

When developing the system we have tried to incorporate functionalities for processing dynamic data. However, it would be impossible in real conditions to continuously have an auditor monitoring the employees' activity. Feedback provided by internal auditors also dis-

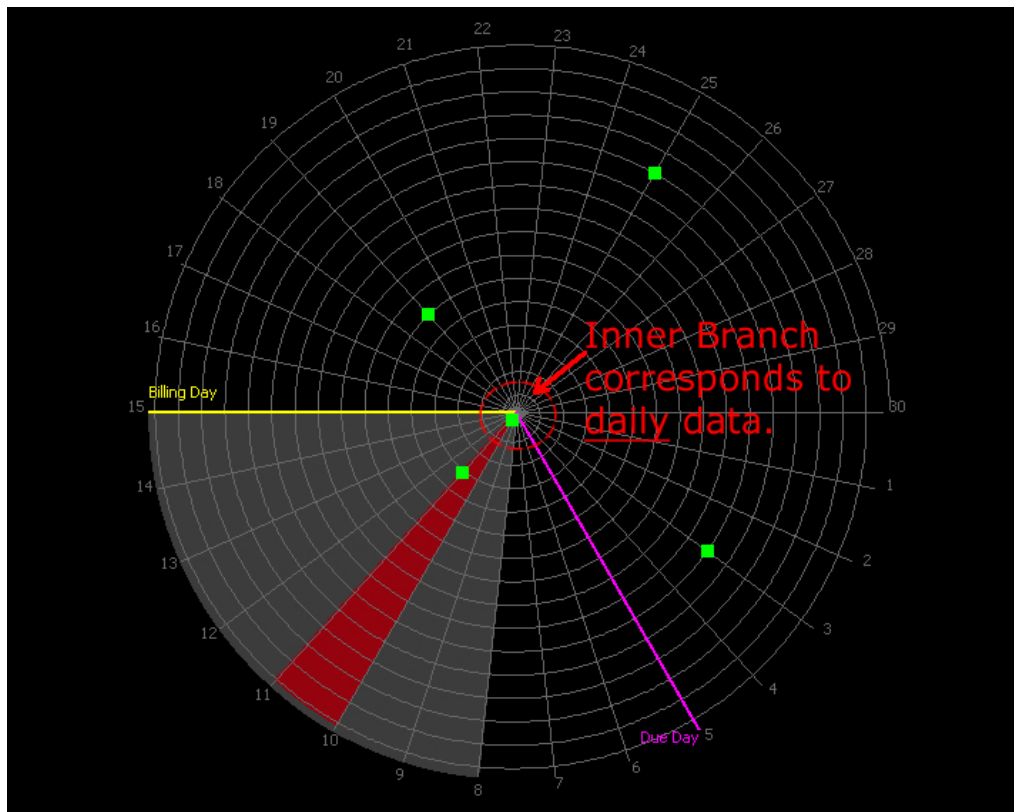


**Figure 8.8:** Spiral visualization when data are visualized without filtering.

couraged us, since fraud analysis takes place on a system different than the operational system which generates log records. For these reasons, we have developed functionalities for processing the data daily as soon as they are generated by the systems.

The main visualization of the semi-online mode is again a spiral and all visualization features are similar to the corresponding ones of the off-line mode (refer to Figure 8.9). In contrast to the off-line, in semi-online mode, the inner branch of the spiral corresponds to the events of current day. The other branches coincide to the first and last month of the input data. The spiral is split by lines representing the days of a month. In order to produce the visualization, the system re-ranks both the clients and employees based on the whole dataset (i.e., including both previous and new records) and defines their ordering in the video frames. In semi-online mode, each event related to a specific client is visualized at the appropriate position on the inner branch, according to its time-stamp. All previous events related to that client are also added in the visualization (on the appropriate branches of the spiral). Again, the auditor seeks for events that appear along a radius or on close radii of the spiral. This visualization facilitates the immediate detection of a periodic pattern that may have just

begun.



**Figure 8.9:** Spiral visualization when the system operates at semi-online mode.

## 8.5 Case Study

In this section, we present a case study on real data. For confidentiality reasons, the data presented in this case study were preprocessed and made anonymous. The data-set which we processed corresponds to a time interval of six months and consists of 35.000 entries involving 7200 distinct clients and 14 employees stemming from a single fraud management system of the company. Note that, the data include sensitive personal data and we were not allowed to process them for the purposes of our case study. The case study was performed while the system operated in off-line mode and without taking into consideration any prior client or employee ranking.

Since the case study was conducted in collaboration with the fraud experts of the company, the main question raised by them was to identify pairs of employee-client that appear to have more than ten events during the last six months, i.e., the time-interval of the data-

set. In order to become familiar with the data-set, we first ranked the clients based on the number of their events. The system identified 430 clients whose number of related events is larger than ten (about 6% of the number of clients in the initial data-set) and distributed them accordingly in the video frames. Even though this number is much smaller with respect to the total number of clients in the data-set, the investigation was still a hard task for the auditor (due to the number of clients to be examined). In the next step, we performed a second ranking (as described in Section 8.3) on the clients based on three factors: (i) the number of events related to a client, (ii) the number of actions that are highly unlikely to appear and are indications of possible fraudulent activity and (iii) the number of distinct employees serving the client. Precise details on the configuration values of each ranking factor are omitted due to a nondisclosure agreement. Also, since we were not communicated the information about the billing date of each client and the employees' shifts we had to ignore these ranking factors.

The ranking procedure distinguished 52 out of 430 clients (about 0.7% of the number of clients in the initial data-set) that were highly-ranked in the above factors and presented them in the first frames of the video. In the next frames, the system presented 62 clients (about 0.9% of the number of clients in the initial data-set) that were medium-ranked in the above factors. The results were further investigated by the auditors who tried to detect periodic patterns (daily or monthly) in the specific frames. The auditors also suggested to apply the periodicity factor described in Section 8.3.1.2 in order to detect events that appear within a time interval of (i) 28 days and (ii) 5 days. For the first case, the system identified 3 out of 52 clients that were highly-ranked and 6 out of 62 clients that were medium-ranked. For the second case, 11 out of 52 highly-ranked clients and 12 out of 62 medium-ranked clients were identified. For the final step of the investigation, the auditors used the supplementary visualizations and the log viewer of the system along with their experience and supplementary data that were not communicated to us in order to evaluate the severity of these events. It should be emphasized that the real-time investigation of the data performed by the auditors (i.e., the non-automated log file processing) did not identify any of these reoccurring activity. The results were taken into consideration by the auditors for further internal investigation. Fortunately, fraud was not confirmed to any of the above cases.





## **9 A Visualization System Utilizing Radial Drawings and Heat-maps**

In this chapter, we extend the system presented in Chapter 8 such that the visualization is not only limited to identifying periodic events that involve pairs of entities, but it provides a complete view of all the examined fraud patterns and the results of the examination on each pattern. Furthermore, the system monitors the activity of the employees aiming at detecting unusual employee activity such as unauthorized access to computers, business systems and accounts of employees or clients.

### **9.1 Main Features of the System**

The proposed system produces a multi-layer radial drawing (see Figure 9.1) representing the activity of employees and clients along with other significant information that enable the identification of possible fraud patterns. In order to produce the visualization, the system measures the similarity of the activity of the employees based on fraud detection patterns (suggested by auditors based on their experience and the framework of the company on internal fraud risk reduction) and appropriate heat-maps are generated and incorporated in the system. The produced visualization is presented as an animation. The system supports supplementary functionalities such as a database log viewer, filtering, export log mechanisms, storing, reloading and post-processing of data, plots and charts (see Figure 9.2). It provides also advanced graphic functionality, including popup menus, printing capabilities, custom zoom, fit-in window, selection, dragging and resizing of objects.

The system extends the work presented in Chapter 8 which detects periodic patterns that may conceal occupational fraud in several ways: (i) The current visualization provides a complete view of all the examined patterns and the results of the examination on each pattern (in Chapter 8, the detection procedure was a “black-box” determining the order of the presentation of the clients in a video representing their activity in which “suspicious” clients appeared first; partial results of the detection procedure are illustrated in additional plots and charts, which hindered the investigation), (ii) the detection mechanism is based on a decision tree even though we have incorporated most of the patterns presented in Chapter 8,

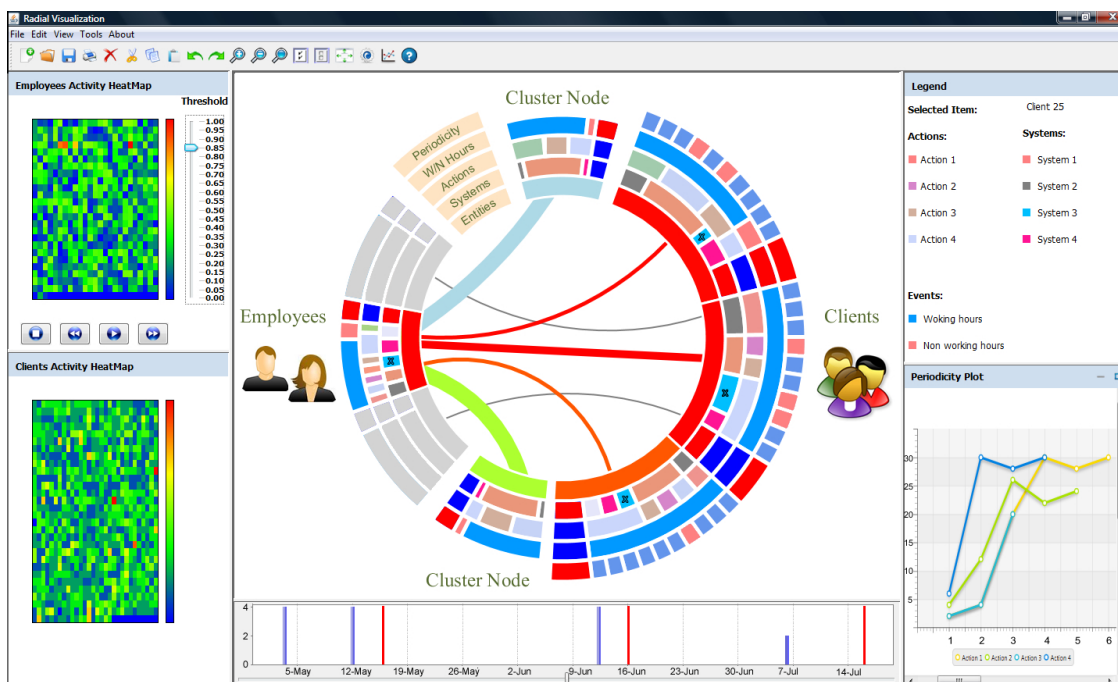


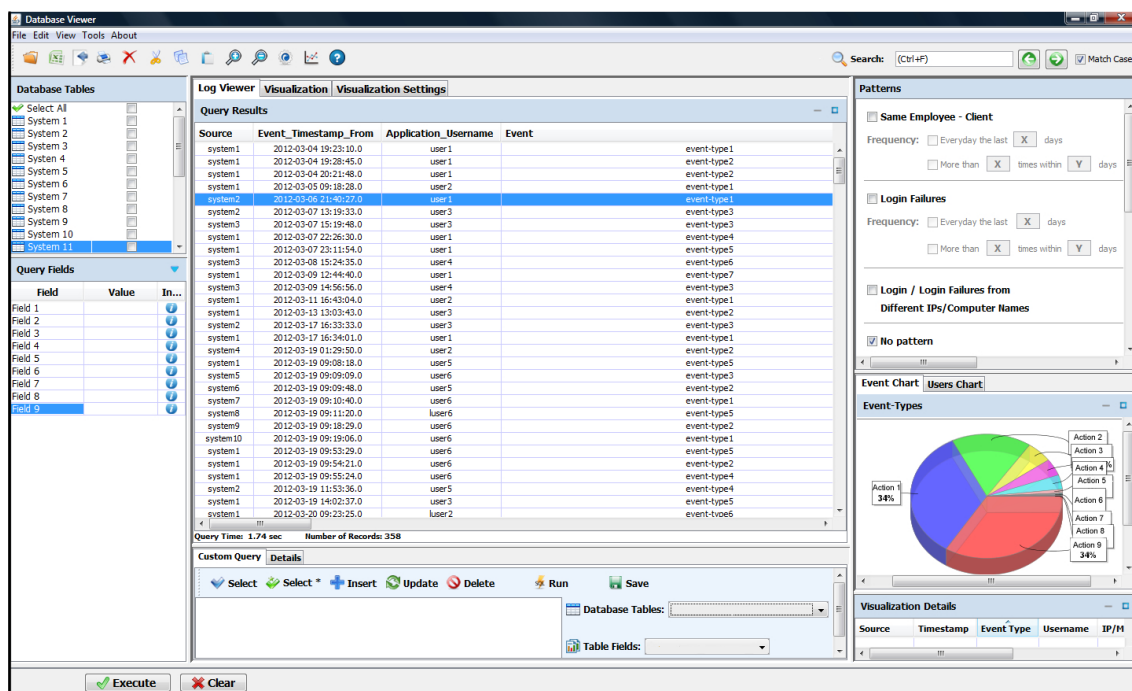
Figure 9.1: A snapshot of the interface of the system.

(iii) for periodicity detection, we apply a variation of the Longest Common Subsequence algorithm [VHGK03] that tackles noisy data, (iv) unusual employee behavior (unauthorized access to computers, business systems, etc) can also be detected using a parallel coordinates plot, and (v) the system provides a database-viewer to facilitate the investigation procedure. The main visualization of the system is similar to the one of Synerscope [Syn11], an industrial tool that represents the billing links and relations between the company and other related entities in order to detect fraud. However, our system is concentrated on detecting exclusively occupational fraud based on fraud patterns suggested by auditors and for this reason, it is equipped with a detection mechanism that preprocesses the data whereas the visualization is conformed to these patterns. Also, we utilize animations for the detection of fraud schemes to avoid cluttering the visualization.

## 9.2 Overview of the Detection Procedure

Again, the system takes as input log files from diverse control systems involving an employee and a client. Let  $e = (t, u, c, a)$  be an event that involves employee  $u$  and client  $c$  (refer to Section 6.5). We say that client  $c$  is related to  $e$  and is also related to employee  $u$ . For a pair of employee-client  $(u, c)$ , an event-series  $T_{u,c} = \{e_{(u,c)}^1, e_{(u,c)}^2, \dots\}$  is a sequence of events

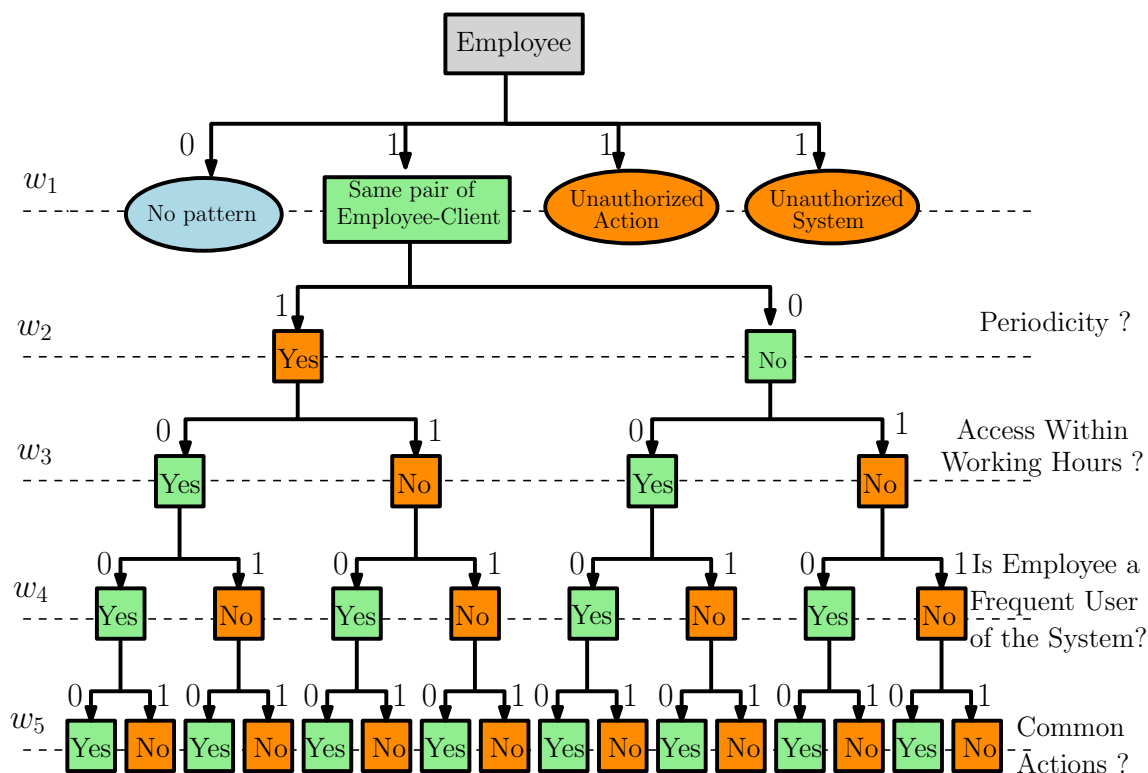




**Figure 9.2:** The startup screen of the system. Custom queries can be performed and results are presented in the log viewer. The data have been anonymized for security and data privacy reasons.

$e_{(u,c)}^i = (t_i, u, c, a_i)$  related to client  $c$  and employee  $u$ .

The visualization may be generated either based on the whole data-set of the database or on queries performed by the auditor in the startup screen of the system (see Figure 9.2). As mentioned above, data have to be preprocessed before producing the visualization such that employees with strong indication of fraud are distinguished. For this reason, for a given employee  $u$ , the event-series with each client related to  $u$  will be evaluated based on possible fraud patterns and a value indicating the severity of the related events (within range  $[0, 1]$ ) will be assigned first to the event-series and then, to the employee. If several fraud patterns are identified, employee  $u$  is assigned the maximum severity value of the already calculated event-series related to  $u$ . The evaluation is performed based on a decision tree generated by the following patterns suggested by the auditors (see Figure 9.3): (i) There exist more than  $X$  events related to employee  $u$  and client  $c$  within a time interval of  $Y$  days/months where  $X, Y$  are configurable by the auditor (refer to the green rectangular node of the first layer below the root of the decision tree in Figure 9.3), (ii) the employee has performed unauthorized actions (based on a list of actions provided by the auditors) and (iii) the employee operated in systems that she is not authorized to use. In the case where patterns (ii) or (iii) occur, the



**Figure 9.3:** The decision tree based on which the event-series of the employees are assigned a severity value.

event-series of the employee is assigned the maximum value (i.e., value 1) such that the employee is definitely distinguished in the visualization. However, in the case where pattern (i) occurs, the investigation has to proceed further. The patterns that are taken into consideration in this case include the following:

- **Event-series periodicity:** A common pattern while examining such fraud schemes is the occurrence of the events in regular time basis. For instance, an employee modifies intentionally the account of a client every month within the billing cycle of the account and more precisely, before its billing date. Assuming that event-series  $T_{u,c}$  related to employee  $u$  and client  $c$  is ordered according to the time-stamps of the events, the system aims to detect similarities between pattern time-series based on a variation of Longest Common Subsequence (LCSS) algorithm for time-series [VHGK03] which is robust under noisy conditions. The pattern time-series include the ideal time-series if the events between the entities appear in time intervals that equal exactly to 1, 7, 15, 30 days and other time-series identified in the past as fraud patterns. In the case where similarity with any of the above time-series is detected, we consider the event-series of the em-

ployee to be periodical.

- **Events occurring outside working hours:** Fraudulent activities usually occur outside working hours, on weekends, on holidays or at the end of the employee's shift. For this reason, if such events occur they have to be taken under consideration.
- **Employee frequency in recorded systems:** Each employee according to her responsibilities operates in specific business systems. If this is not the case, then the employee has to justify the recorded event. Also, in several systems such as fraud management systems (FMS), it is expected that an employee monitors the activity of a "suspicious" client. Hence, events stemming from these systems have to be given smaller weight.
- **Actions taken by the employee:** Similarly to the previous case, there exist some actions that an employee is unlikely, but not unauthorized to perform since they do not conform to her responsibilities.

The idea behind the decision tree was to correspond to each layer one fraud pattern and create a path according to the result of the examination on each pattern. We consider the importance of patterns based on their corresponding layer in the decision tree, such that the higher ones (i.e., closer to the root) are more important. Let  $x = [x_1, x_2, x_3, x_4, x_5]$  be the pattern vector examined (e.g.,  $x = [0, 0, 0, 0, 0]$  corresponds to non-fraudulent activity) and let  $y = [y_1, \dots, y_5]$  be the vector resulting from the traversal of the decision tree from its root to its leaves according to the evaluation of the events of pair  $(u, c)$  on each factor. If the examination of the events leads to "Unauthorized action" or "Unauthorized system" the event-series is directly assigned value 1. Else, each tree layer  $i$  is assigned a weight, say  $w_i, i = 1, \dots, 5$ , based on the formula  $w_i = (N - i + 1) / (\sum_{j=1}^N (N - i + 1))$ , where  $N$  is the number of decision tree layers [SSE81], such that dissimilarities between the two vectors that occur at higher levels of the tree will be more important. For this reason, the distance between vectors  $x$  and  $y$ , say  $d(x, y)$ , representing the dissimilarity by the pattern vector, is calculated by applying the normalized Weighted Euclidean Distance metric formula  $d(x, y) = \sqrt{\sum_{i=1}^5 (x_i - y_i)^2 * w_i} / \sqrt{\sum_{i=1}^5 w_i}$ . This value (or the maximum of the already calculated values, if more than one fraud patterns exist) corresponds to the severity value of the event-series of employee  $u$ , which is the value finally assigned to the employee. Based on these values the system generates a heat-map representing all employees by rectangular nodes and gradient colors from blue to red (refer to the upper-left heat-map of Figure 9.1), such that nodes with color "close to" color-red represent employees with strong indications of fraud, whereas blue colored nodes employees with no "suspicion" of fraud. Similarly to the severity calculation of the event-series of employees, the system assigns also a severity

value to clients based on the above patterns. The only difference is that for a given client  $c$ , the severity value is calculated on all the events related to  $c$  (not only the ones that concern a specific employee). In this manner, a client involved in “suspicious” activity with two or more employees will be distinguished.

### 9.3 Description of the System

The visualization window consists of two heat-maps representing the severity of the activity of employees and clients (refer to the upper-left and the bottom-left panel of Figure 9.1). Although a great deal of research has focused on the deficiencies of rainbow color maps [BTI07], they are still widely used in the visualizations, since their effectiveness depends on the nature of the data. In our data-sets, it was necessary to have three colors (red, green, blue) representing clearly the severity value of the examined entities (high, medium, low, resp), and thus, we have chosen this type of color map. We have tried different types of color maps, but the result was more misleading during the investigation. The particular selection of colors was not confusing to the auditors since fraud cases appear rarely in a company which implies that few red-colored rectangles appear eventually in the heat-maps.

At startup only the heat-map representing the activity of the employees is generated. The auditor selects a threshold value that determines the employees that will be presented based on their severity values. The visualization is animated and each time an employee together with her related clients is illustrated. Before an employee is presented, the heat-map that corresponds to the activity of related clients is generated.

The main visualization of the system is a multi-layer radial drawing where each layer  $L_1, \dots, L_5$  (see Figure 9.4) represents a different aspect of the audit data (systems, actions, within working hours or not, periodicity) and each circular sector corresponds to an entity (employee, client or cluster). Then, a graph is generated; its nodes correspond to each of the above entities, whereas its edges correspond to the connections among them. The inner-most layer of the visualization (layer  $L_1$  of Figure 9.4) accommodates the nodes of the graph (drawn as portions of a ring). Nodes representing employees are drawn to the left-part of the visualization where the ones representing their related clients to the right part. To avoid cluttering the visualization with nodes representing clients with no indication of fraud, the auditor is also able to specify thresholds that split clients in one or two clusters (low-severity cluster and/or medium-severity cluster) according to their severity values. These nodes are accommodated on the top and to the bottom part of the visualization. The color of the nodes (apart from the ones representing clusters and the gray-colored ones that will be explained later) follows the color of the corresponding entities in the heat-maps. The light-blue (green,

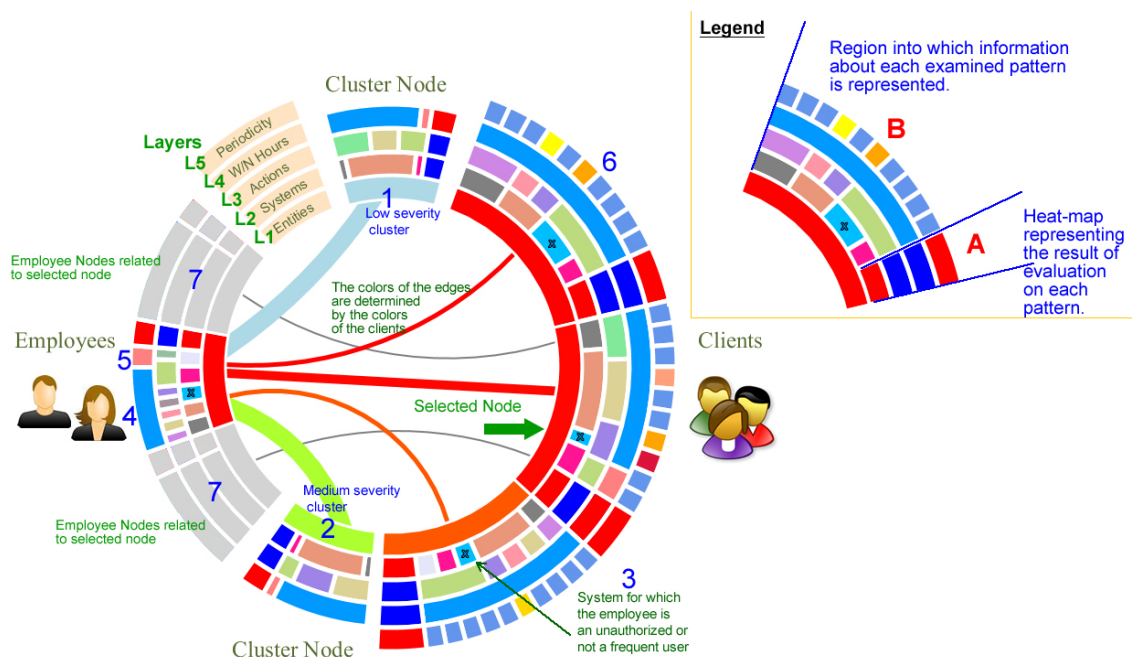


Figure 9.4: Description of the main-visualization of the system.

resp) colored cluster-node corresponds to the low (medium, resp) severity cluster (refer to reference points 1 and 2 of Figure 9.4, resp). Regarding the edges of the drawing, the system supports either circular arc edges or straight-line edges. The thickness of an edge is proportional to the number of connections between the employee and the client while its color is determined by the color of the client. In a fraud context, “fat” edges (unless referring to cluster nodes) and in particular, the red-colored ones will be indications of fraud that have to be further examined.

Subsequent layers (i.e., layers  $L_2 - L_5$ ) represent the patterns described in Section 9.2 and are split into two regions *A* and *B* (refer to the legend of Figure 9.4). Region *A* represents a heat-map indicating the result of the examination of the entity in the specific pattern. Red color indicates identification of “suspicious” pattern. Region *B* illustrates information about each corresponding examined pattern. In layer  $L_2$ , the different business systems related to the each of the entities are represented. Each such system is characterized by a specific color and occupies space proportional to the corresponding aggregate percentage of use by the entity. For each employee, this percentage is calculated based on the aggregated percentage of use on all clients that are currently drawn in the visualization, whereas for each client based on the percentage of use by the employee currently visualized (unless more than one employee related to a client are drawn simultaneously in the visualization). Similarly, for cluster nodes the aggregated percentage of use for all clients that belong to the cluster is calculated.

Systems for which the employee is an unauthorized or not a frequent user are marked by an  $X$  (refer to reference point 3 of Figure 9.4).

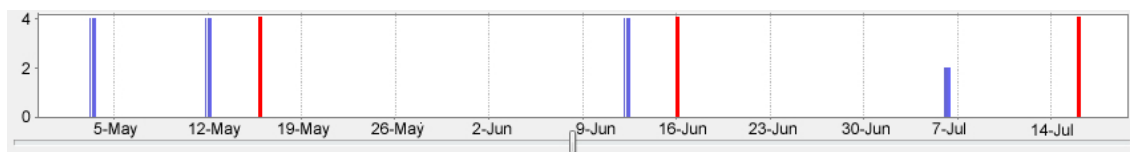
Layer  $L_3$  corresponds to the actions reported for each entity and are drawn in a similar manner as the ones in layer  $L_2$ . Again unauthorized or “suspicious” actions are marked by an  $X$ . Layer  $L_4$  represents the percentage of events that occur within or outside working hours. The light-blue colored parts represent events occurring within working hours, whereas the light-red colored parts indicate the existence of events occurring outside working hours (e.g., see reference points 4 and 5 of Figure 9.4, resp). For each client node there exists an additional layer (refer to  $L_5$  of Figure 9.4, e.g., see reference point 6) that indicates whether or not the event-series of the client is periodical. The event-series is compared with the pattern time-series currently stored in the system and a heat-map is generated indicating the degree of similarity with each pattern. Again, light-red colors indicate “suspicious” cases.

Regarding the investigation procedure, as already mentioned, the auditor specifies a threshold and the employees with assigned severity value above the threshold are presented in the visualization one by one, together with their related clients. The auditor is able to start, pause or stop the video and process the visualization. In the case where a client node is selected, additional employees related to the client can be added to the visualization which facilitates the possible identification of two or more employees that may cooperate in committing fraud (the case where an employee node is selected is treated similarly). In Figure 9.4, gray colored nodes (see reference point 7) represent nodes added during post-processing when a client node is selected (refer to the node pointed by the green arrow of Figure 9.4). In the case where more than one node representing employees exist simultaneously in the visualization, the auditor is able to select one of them and add the related clients to the visualization. Gray-color is utilized for non-selected employee nodes together with their edges and related clients (if they are not related also to the selected employee) to avoid distracting the auditor.

Since it is possible to switch between employees during the investigation, if the post-processing of a case is completed, the animation resumes from the last visualized employee before pausing the animation. In the case where a cluster node is selected, the corresponding rectangles in the heat-map representing clients are marked by an  $X$  such that they can be added (if desired) to the visualization. However, the system permits a specific number of additions of employee or client nodes in order to avoid cluttering the visualization area. If this number is exceeded, the system optionally is able to produce a visualization where only the inner-most layer of the radial drawing (i.e., the one corresponding to the clients and employees) is drawn along with the relations between them. Again, the width of the edges is proportional to the number of events that relate two entities. In the case where further investigation is needed the auditor selects the desired node (which becomes larger) and the other

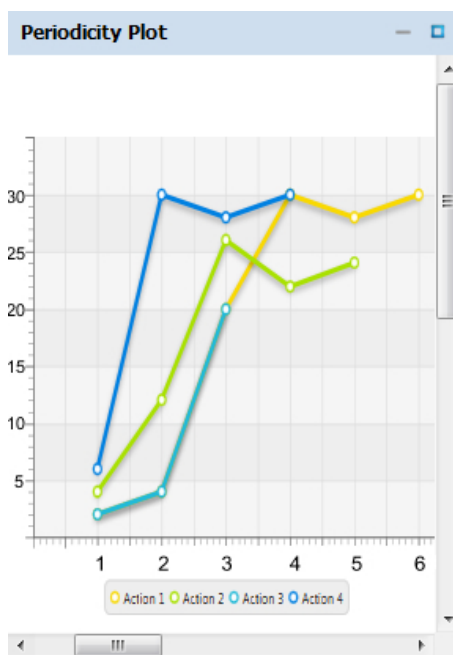
layers of the radial drawing (i.e.,  $L_2 - L_5$ ) that correspond to the particular node appear in the visualization. In this manner, the system is able to visualize simultaneously a larger set of entities and reveal the relations between them. However, this may slow down the investigation process and for this reason, we adopted both the animation approach and the simultaneous visualization of all layers for each node.

The ordering of the nodes representing the employees is performed according to their severity value (the more “suspicious” nodes are presented first in the animation). The clients appear in arbitrary order since no crossings between edges connecting a particular employee with her related clients can exist. The only crossings that may occur are caused by gray-colored employees related to clients already visualized and since these edges are also gray-colored, they do not confuse the auditor (see Figure 9.4). The gray-colored employees that are added in the visualization are placed either on the top or to the bottom of the already placed employee-nodes to maintain the relative positions of the already placed nodes. We also have chosen not to apply crossing minimization heuristics since, the addition of new nodes to the visualization may imply a rearrangement of the positions of the already placed nodes, disturbing the underlying mental map.



**Figure 9.5:** A time-line plot representing the event-series for the selected pair of entities.

Figure 9.5 accommodates a time-line plot representing the event-series for the selected pair of entities (refer to the blue series) where the  $x$ -axis corresponds to the date of the occurrence of each event and the  $y$ -axis to the number of events occurred during the specific date. The red drawn columns represent the billing dates of the account of the specific client. This plot facilitates the identification of possible periodic activity especially close to the billing date of the account of the client. In employee fraud schemes, it is also possible that the event-series related to a pair of entities is periodical only based on a specific action. For this reason, we have incorporated in the system a second plot (refer to Figure 9.6) that presents all reported actions by distinct series. In this plot, the  $y$ -axis corresponds to the day of the month that each event related to a specific action occurred (e.g., the first event related to an action that occurred on the 15th day of a month will be drawn on point  $(1, 15)$ ). For this case, we ignore multiple occurrences of events related to the same action occurred the same day of the month. In the case where periodicity occurs for a specific action, the corresponding series will have part (or the whole series) almost parallel to  $x$ -axis.



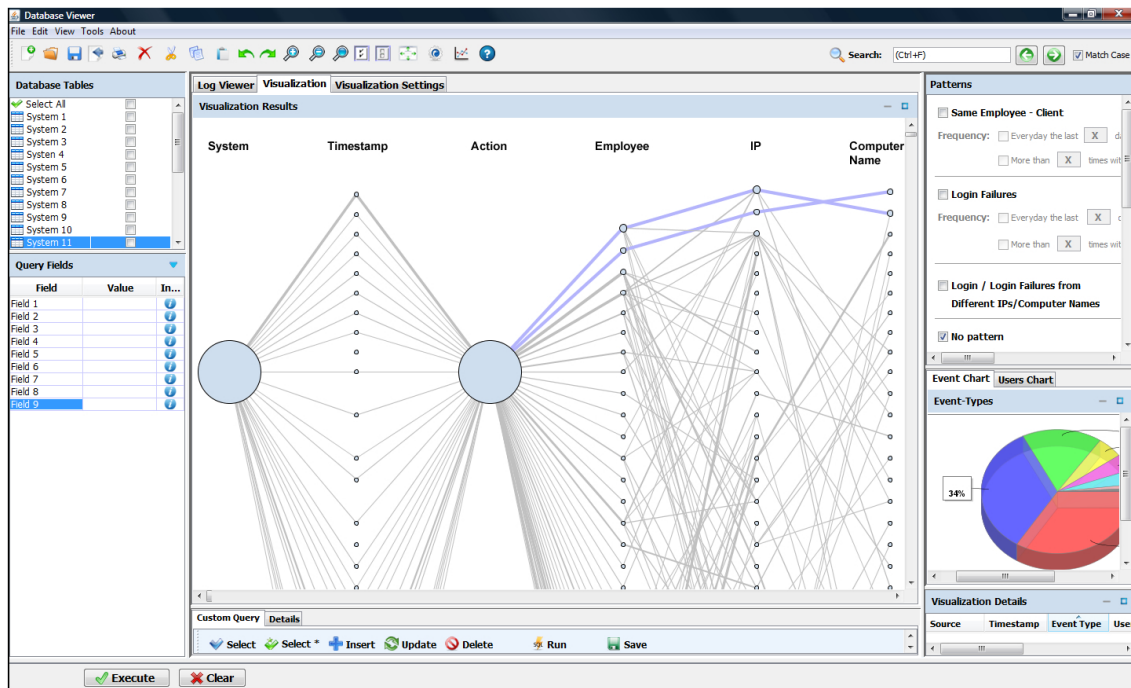
**Figure 9.6:** A periodicity-plot that presents all reported actions by distinct series for the selected pair of entities.

The system supports mechanisms to detect patterns of unusual employee behavior such as unauthorized access to computers, business systems and accounts of employees or clients by producing a parallel coordinates plot (see Figure 9.7). Each record consists of an employee, a time-stamp, an IP indicating the address of the employee's computer, a computer name and an action (i.e., login, login failure, etc). The size of the nodes and the edges is proportional to the number of their occurrences in the database. The nodes on each layer are ordered by their number of occurrences in the data-set. The patterns used in this scenario include the following: (i) More than  $X$  failed login attempts within a time interval of  $Y$  days/months, where  $X$  and  $Y$  are configurable by the auditor, and (ii) Login attempts or failed login attempts occurring from different IPs and/or computer names. The visualization of Figure 9.4 can also be adapted to this scenario -either with or without the preprocessing step (since the number of employees and actions is manageable in a radial drawing)- by substituting the nodes representing the clients by nodes that represent actions.

## 9.4 Case Study

In this section, we present the results of the evaluation of the system on real data-sets stemming from two control systems. All data provided to us were anonymous for security





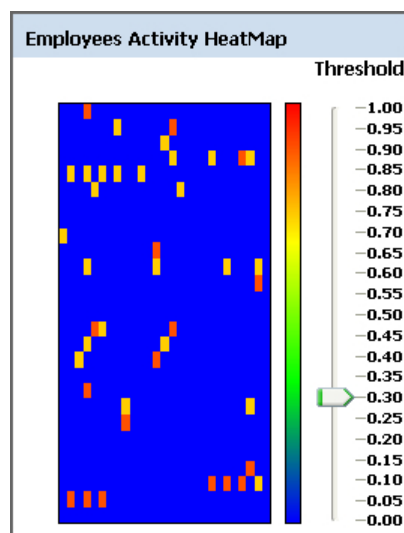
**Figure 9.7:** A parallel coordinates plot to monitor failed login attempts in a specific system.

and data privacy reasons. The data-set consists of 180.637 entries lying within a time interval of six months. The data-set consists of 710 distinct employees and 83.030 distinct clients. In the data-set, 66.2% of clients had only one occurrence, 31.6% between 2 and 5, 1.4% between 5 and 10 while the remaining ones (i.e., 0.8%) had more than 10 occurrences. The auditors have also included a set of entries corresponding to a “fictional” fraud case scenario where an employee modifies the account of a client. However, we were not communicated any information regarding the billing date of the accounts of the clients.

Since one of the data-sets stems from a fraud management system, it is expected that reoccurring activity between the same pair of employee and client will occur (a “suspicious” client reported by a fraud management system is expected to be supervised by an employee). For this reason, we concentrated our study in identifying pairs of employees and clients that appear to have more than 10 related events. The system identified 41 employees (5.8% of the total number of employees) that were related with the same client with more than 10 events (refer to the orange and red colored rectangles of the heat-map of Figure 9.8).

For each of the above employees we had to calculate the similarity of the event-series with pattern time-series (see Section 9.2) in order to detect periodicity. In particular, the auditors were interested for periodic events that occur monthly (i.e., periodicity of 28 up to 30 days). Thus, we could consider only the pattern time-series (refer to Section 9.2) that corre-

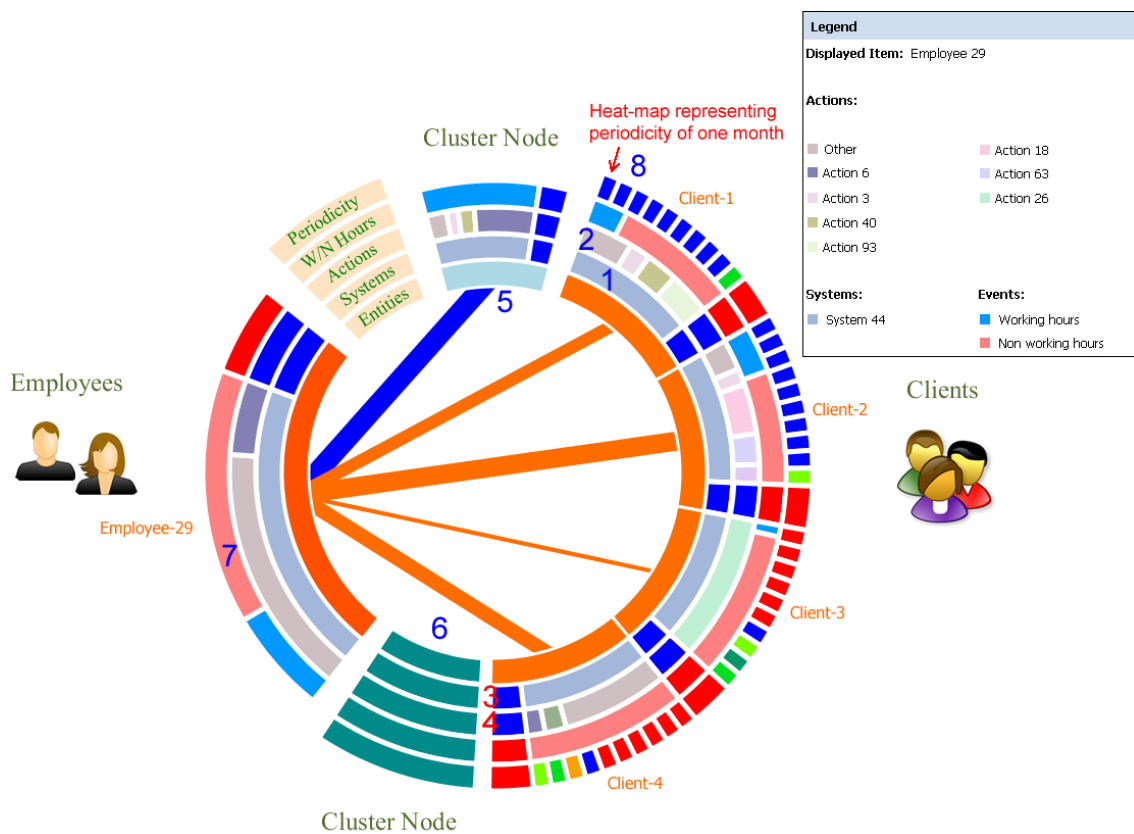
sponds to monthly activity. However, we decided to calculate the severity values based on all pattern time-series in order to distinguish any reoccurring activity and decide afterwards through the visualization if “suspicious” activity really exists. Even though, this approach creates more false positives that have to be investigated, it ensures that other possible periodic events will not be omitted. Among these 41 employees that were related with the same client with more than 10 events, 17 of them (2.4% of the total number of employees) appear to have periodic activity and events occurring outside working hours (refer to the red-colored rectangles of Figure 9.8), while the remaining ones (24 employees; 3.4% of the total number of employees) had events occurring outside working hours (refer to the orange-colored rectangles of Figure 9.8). Also, no employee had performed unauthorized access to systems or had used unauthorized actions, whereas only one employee had used non common actions. The results were communicated to the auditors who investigated whether there exist real indications of fraud.



**Figure 9.8:** A heat-map indicating the severity values for the employees participated in the case study.

In the following, we present some of the first frames of the animation in order to describe the investigation procedure. Figure 9.9 indicates a highly-ranked employee (i.e., Employee-29). Employee-29 was related to four clients, i.e., Client-1, Client-2, Client-3 and Client-4, more than 10 times using one system for which the employee was a frequent user and common actions (see reference points 1 and 2 of Figure 9.9; the parts representing the systems and the actions are not marked by an X, which implies that the user was frequent for the specific systems or had performed common actions, and the corresponding heat-maps are blue-colored; see reference points 3 and 4 of Figure 9.9). All other clients related to Employee-29, where placed in the low-severity cluster, whereas no client was placed in the medium-severity cluster.

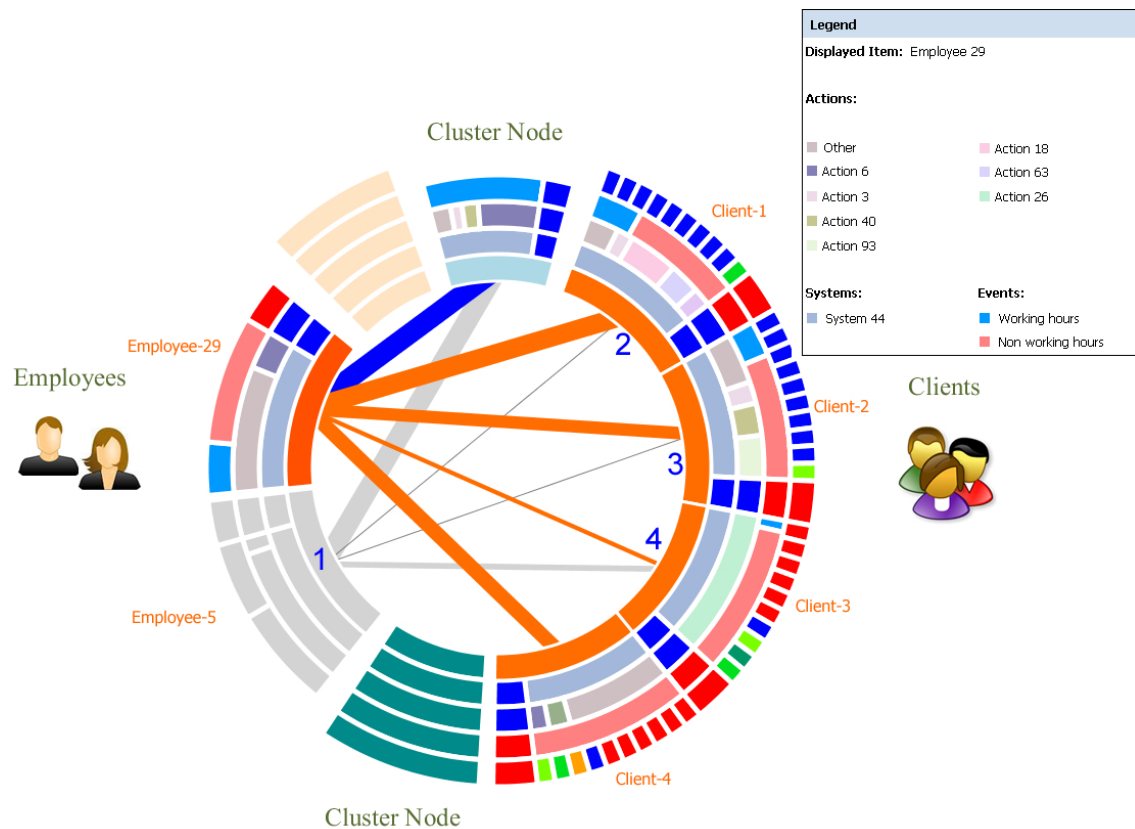
ter (see reference points 5 and 6 of Figure 9.9, resp.). Most of the events related to Employee-26 occur outside working hours (see reference point 7 of Figure 9.9). The layer of the visualization which accommodates the actions that Employee-26 has performed, is split in two parts representing “Action 6” and “Other”. In the “Other” action, we have clustered actions whose percentage of use was too small (< 0.1%) to be visualized. The problem was caused by the large number of possible actions in this particular system which does not permit the simultaneous visualization of all actions. However, if one of the clustered actions is not common for a particular employee, the red-colored heat-map in the corresponding layer and the X marking in the part representing the “Other” action would reveal the problem to the auditor.



**Figure 9.9:** A frame of the animation indicating the activity of highly-ranked Employee-26.

We will now proceed to describe the investigation procedure for each of the four clients. Client-1 and Client-2 have the majority of the events that relate them with Employee-29 occurring outside working hours (refer to the corresponding layer of Client-1 and Client-2). However, they appear to have a small similarity (less than 50%) with only one fraud pattern time-series. Note that, the pattern time-series that corresponds to periodicity of one month

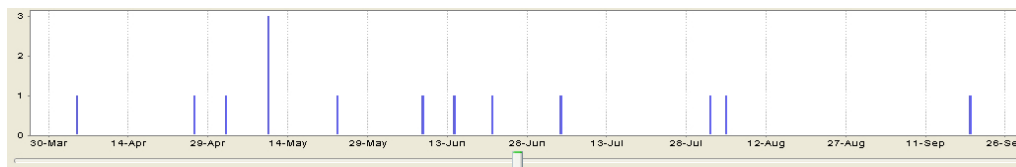
corresponds to the first heat-map of the each periodicity layer (see e.g., reference point 8 of Figure 9.9). Since the auditors were interested in events that occur monthly, these clients were not considered “suspicious”. Regarding Client-3 and Client-4, they have also the majority of the events that relate them with Employee-29 occurring outside working hours but, they appear to have strong periodic activity. In particular, there exist six heat-maps indicating strong similarities with pattern time-series including the one that represents monthly periodic activity. For this reason, these clients have to be further investigated.



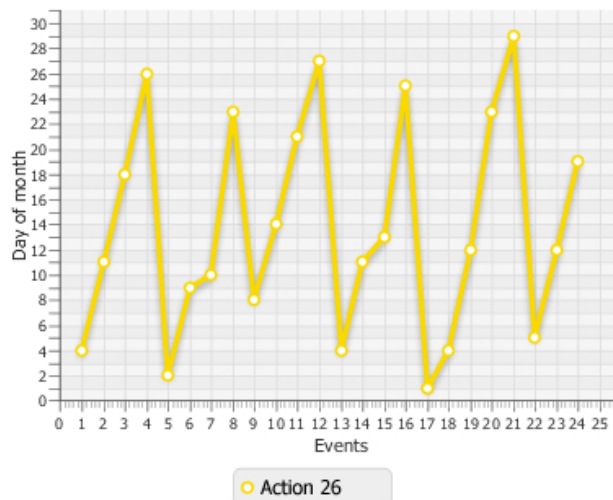
**Figure 9.10:** All employees related to Client-3 are added in the visualization. The gray-colored node corresponds to the new employee added (Employee-5) who surprisingly is related to almost all high-ranked clients of Employee-29.

Client-4 was related to no other employee (when we selected the node that corresponds to Client-4, no other employee was added to the visualization). However, when we selected Client-3, a new employee, referred to as Employee-5 (see the gray-colored node of the visualization; see reference point 1 of Figure 9.10), was added to the visualization. The interesting thing was that Employee-5 was related with the same high-ranked clients as Employee-29 (except for Client-4; see reference points 2, 3 and 4 of Figure 9.10). There exist two possible

explanations for this scenario. Either two employees are both responsible for monitoring the activity of the clients or these employees are accomplice to fraud. Thus, the investigation has to proceed further. The time-line plot of Figure 9.11i represents the time-stamps of the events relating Employee-29 and Client-3. Obviously, there exists a continuous activity between the two entities. However, this activity according to the auditors resembles more to monitoring activity rather than to a fraud pattern. This assumption is also reinforced by Figure 9.11ii which represents all reported actions for the selected pair of entities. In particular, only one action is performed by Employee-29 towards Client-3 and according to the auditors this action is part of a monitoring procedure. Studying, in a similar manner, the time-line plot and the periodicity plot for Client-4, the auditors claimed that there existed no indications of fraud (the recorded actions were also part of a monitoring procedure).



(i)



(ii)

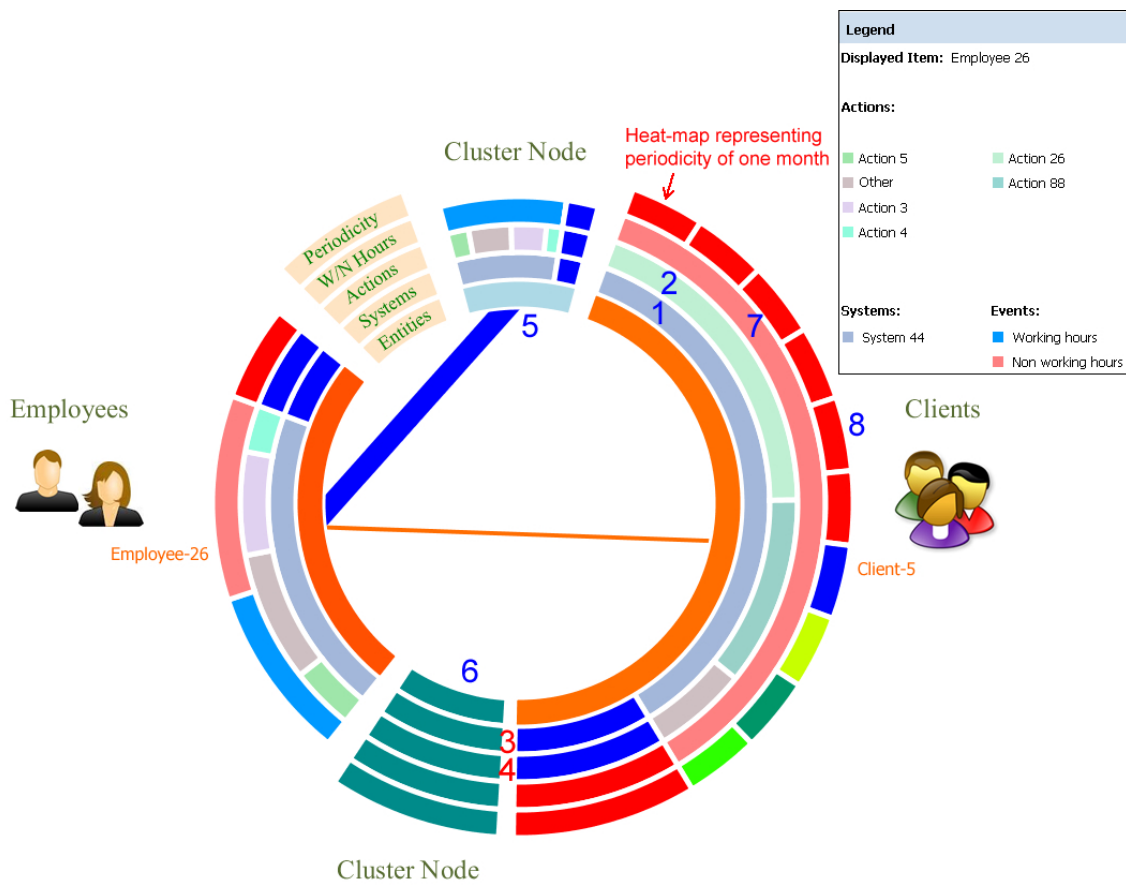
**Figure 9.11:** (i) The time-line plot for Employee-40 and Client-6 indicating an obvious monthly activity. (ii) A plot illustrating the periodicity of performed actions for Employee-40 and Client-6.

Figure 9.12 depicts another highly-ranked employee (i.e., Employee-26). Employee-26 was involved only with Client-5, more than 10 times using one system for which the employee was a frequent user and common actions (see reference points 1, 2 3 and 4 of Fig-

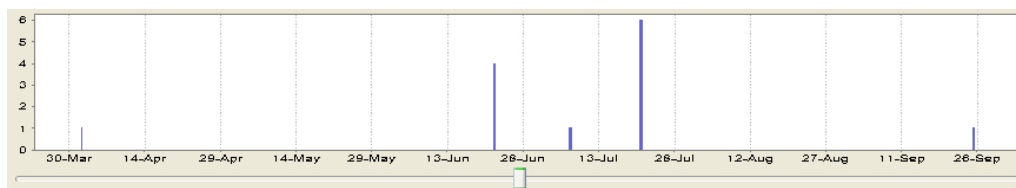
ure 9.12). Again, all other clients related to Employee-26, were placed in the low-severity cluster, whereas no client was placed in the medium-severity cluster (see reference points 5 and 6 of Figure 9.12, resp.). Also, Client-5 was related only to Employee-26 (when we selected the node that corresponds to Client-5, no other employee was added to the visualization). However, almost all events that relate the two entities appear outside the working hours (see reference point 7 of Figure 9.12). The visualization of Figure 9.12 also indicates a strong periodic activity (refer to the heat-maps in the last layer of the part of the visualization that corresponds to Client-5). In the next step, we examined the time-line plot for the two entities (refer to Figure 9.13; see reference point 8). One can distinguish a monthly periodic activity between June and September (i.e., the actual dates are 22/6, 21/7, 25/9), even with small gaps (no entries in August) and some noisy data (i.e., 7/7). The auditors that examined the case determined that there were no indications of fraud in this particular case mostly because of the actions performed by the employee which were again common monitoring actions. Client-5 was reported by a fraud management system as a “suspicious” client and thus, Employee-26 was monitored her in regular time basis.

Figure 9.14 illustrates the fictional fraud case which was added by the auditors. In this scenario, Employee-40 is related to Client-6 using two business systems for which she is a frequent user and common actions (see reference points 1, 2, 3 and 4 of Figure 9.14). All the events occurred within working hours (see reference point 5 of Figure 9.14). However, the event-series of Employee-40 appears to have strong similarity with six fraud pattern time-series (see reference point 6 of Figure 9.14). Also, Client-6 was not related to any other employee of the company (when selected, no other employee was added to the visualization). Again, all other clients related to Employee-40, were placed in the low-severity cluster, whereas no client was placed in the medium-severity cluster (see reference points 7 and 8 of Figure 9.14, resp.). Even though, this visualization resembles a lot to the one of Employee-26 (see Figure 9.12), the time-line plot (see Figure 9.15) and the periodicity plot (see Figure 9.16) explain why this case is considered as fraud. The first suspicions according to the auditors, are raised by the fact that there exists activity between the two entities stemming from two business systems (they take also into consideration the type of the systems; an information that was not communicated to us in full detail).

The auditors explained to us that the time-line plot (refer to Figure 9.15) matches to a fraud case scenario according to which there exists some activity at the beginning (between April - May) with no specific periodicity and then, appears periodic activity (from May to September). In the first time interval, the fraudster is trying to plan and organize her fraud by performing a number of actions. Once the fraud is organized, only periodic actions are required. Another suspicious fact in this case is that the events from May to September occur



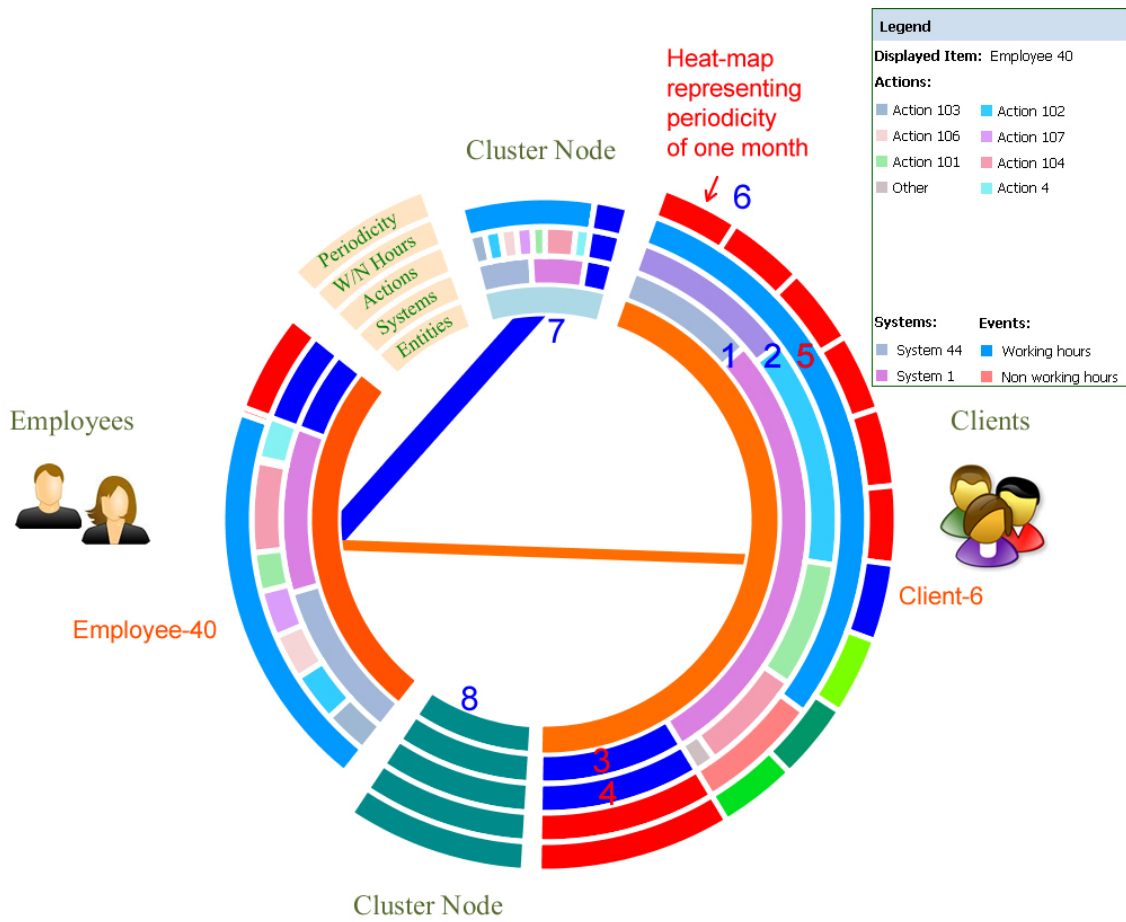
**Figure 9.12:** A frame of the animation indicating the activity of a Employee-26.



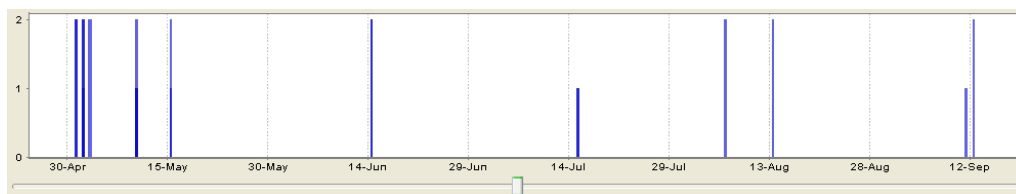
**Figure 9.13:** The corresponding time-line plot for Employee-26 and Client-5.

close to the same dates of each month (from 10th to 15th). Of course, there exist some “noisy” data that have to be excluded in order to understand the fraud pattern. These may have been caused either on purpose to cover up the fraud or were part of the duties of the employee.

The above assumption is reinforced by the plot of Figure 9.16 which reveals the periodic occurrence of each performed action. For instance, “Action 101” (see reference point 1) appears to have periodicity around the 15th day of the month from its second occurrence and



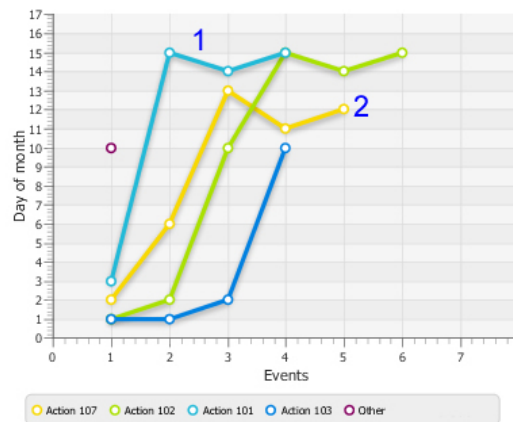
**Figure 9.14:** A frame of the animation illustrating the activity of Employee-40. This case corresponds to a fictional fraud case scenario.



**Figure 9.15:** The time-line plot for Employee-40 and Client-6 indicating an obvious monthly activity.

later. Also, “Action 107” (see reference point 2) appears to have periodicity around the 11th-13th day of the month from its third occurrence and later. In particular, the vast majority of events are recorded between the 10th and the 15th day of the month. We could be more convinced that this case consists fraud if we knew exactly the billing cycle of the account of the client.





**Figure 9.16:** A plot indicating the periodic pattern for the performed actions of Employee-40 and Client-6.

In a similar manner, the frames of the animation illustrating the other highly-ranked employees were investigated. In particular, we were given more attention to the first 17 frames of the animation containing periodic events. Since the data-sets provided for the case analysis were sensitive, we were not communicated many details about the final results of the investigation. Fortunately for the company, the only real evidence of fraud existed in the fictional data added by the auditors. However, the auditors had not identified all these cases while examining the data-sets manually and they had to make an additional investigation for them.





## 10 Conclusions

In this Ph.D. thesis, we have focused on two different topics of graph drawing i.e., the problem of maximizing the crossing resolution of a graph drawing and applications of graph drawing in fraud detection. For the former problem, we have mainly studied problems related to right-angle crossing drawings, i.e., drawings in which every pair of crossing edges intersects at right angle. We have also introduced a closely related problem, the total resolution maximization problem, according to which the minimum of the crossing and the angular resolution is maximized. For the problem of fraud detection, we presented a combination of methods and visualizations in order to detect occupational fraud schemes in business systems in which pairs of entities (employees and clients) are involved (e.g., billing, membership renewal systems, and so on). We presented two approaches that aim to detect periodical events that may conceal occupational fraud and an integrated fraud detection visualization system based on fraud patterns suggested by internal auditors of a large telecommunication company.

In Chapter 3, we proved that it is  $\mathcal{NP}$ -hard to decide whether a graph admits a straight-line RAC drawing. Didimo et al. [DEL11] proved that it is always feasible to construct a RAC drawing of a given graph with at most three bends per edge. If we permit one or two bends per edge, does the problem remain  $\mathcal{NP}$ -hard? The same question arises if the input graph has  $n$  vertices and exactly  $4n - 10$  edges, i.e., whether the problem of recognizing the class of maximally dense RAC graphs is also  $\mathcal{NP}$ -hard.

In Chapter 4, we introduced and examined geometric RAC simultaneous drawings. A reasonable problem that arises in this context is to identify other non-trivial classes of graphs, besides a matching and either a path or a cycle that admit a GRacSim drawing. Also, we considered only geometric RAC simultaneous drawings. For the classes where GRacSim drawings are not possible, it would be interesting to study drawings with bends or relax the optimality constraint on the crossing resolution of the produced drawings. A quite similar problem to the GRacSim drawing problem is the problem of drawing two (or more) graphs on the same vertex set on the plane, such that each graph is drawn RAC (i.e., only edges of different graphs may introduce non-right angle crossings). Note that, the class of graphs that admit such drawings

contains the class of graphs for which a simultaneous drawing is possible. Finally, it would be nice to obtain more appealing GDual-GRacSim drawings for an outerplanar graph and its dual. In this case, we could also study the required drawing area and provide a characterization of the classes, other than outerplanar graphs, admitting a GDual-GRacSim drawing.

In Chapter 5, we studied the total resolution maximization problem. It would be interesting to try to identify other classes of graphs that admit optimal drawings. Even the case of planar graphs is of interest, as by allowing some edges to cross (say at large angles), we may improve the angular resolution and therefore the total resolution. In [vK10], a class of graphs satisfying this property is given.

In Chapter 7, we presented a metric and visualization techniques that aim to detect periodic events in time-series data. It would be interesting to study alternative metrics to measure the confidence degree that can be used in order to obtain more accurate periodicity estimations. This may also affect the quality or the type of the produced visualizations. Identifying group of employees (instead of a particular employee) that appear to have similar suspicious behavior is also of interest. Standard clustering techniques adopted from Graph Drawing may be useful for the production of such visualizations.

In Chapter 8, we proposed a system that aims to detect periodic activity and occupational fraud in business systems in which pairs of entities, such as employees-clients are involved. It would be of interest to incorporate more factors in order to produce more accurate values at the ranking procedure. It would be also better for the auditor to add custom factors and define the appropriate functions through the graphical user interface of the system. Again, it is of interest to search for groups of collaborating employees/clients.

In this Chapter 9, we presented an integrated fraud management visualization system that aims to identify patterns that may conceal occupational fraud through a combination of pattern recognition and visualization. Our work opens several aspects for future work such as incorporation of more fraud patterns, use of more statistical methods and, extension of the system in order to identify more complicated fraud schemes (client fraud, telecommunication fraud, etc.) in a wider variety of business systems.

## **Part II**

**Απεικόνιση Γραφημάτων με Έμφαση σε Εφαρμογές Ασφάλειας**





## 11 Εισαγωγή

*“Use a picture. It’s worth a thousand words.”*

Arthur Brisbane, 1911

*“A good sketch is better than a long speech.”*

Napoleon Bonaparte

### 11.1 Γιατί Απεικόνιση Γραφημάτων;

Η πρόοδος της τεχνολογίας, των επικοινωνιών και του υπολογισμού σε συνδυασμό με την ευρύτατα διαδεδομένη χρήση του διαδικτύου έχουν αυξήσει δραματικά τον όγκο των δεδομένων που εμπεριέχουν πληροφορίες διαφόρων τύπων (προσωπικές, επαγγελματικές, επιστημονικές, ασφάλειας και ούτω καθεξής) και έχουν αναδείξει το πρόβλημα της επεξεργασίας πληροφοριών (information processing) και της ανάλυσης δεδομένων (data analysis) σε ένα πολύ σημαντικό κομμάτι της ανθρώπινης ζωής. Η επεξεργασία και η ανάδειξη πιθανών υποκείμενων σχέσεων σε τέτοια σύνολα δεδομένων αποτελούν μία αδιαμφισβήτητη πρόκληση για τους αναλυτές δεδομένων, καθώς εκτός από τον τεράστιο όγκο δεδομένων που έχουν να αντιμετωπίσουν, η πληροφορία προέρχεται τις περισσότερες φορές από διαφορετικές πηγές (συνήθως υπό τη μορφή κειμένου και σε μη-ομοιόμορφη μορφή). Ένα άμεσο πρόβλημα που ανακύπτει αφορά στην παρουσίαση των δεδομένων με τέτοιο τρόπο ώστε ένα μεγάλο μέρος της πληροφορίας να μεταδίδεται όσο πιο αποτελεσματικά γίνεται σε απλή και κατανοητή μορφή.

Η οπτικοποίηση (visualization) μπορεί να είναι πολλά υποσχόμενη σε αυτό πλαίσιο, καθώς μετατρέπει τον τεράστιο όγκο των δεδομένων σε μία συνοπτική οπτική εικόνα, η οποία παρέχει μία πλούσια και κατανοητή παρουσίαση του περιεχομένου. Πράγματι, οι άνθρωποι προτιμούν να οπτικοποιούν την πληροφορία παρά να την περιγράφουν με λέξεις, καθώς ο ανθρώπινος εγκέφαλος αποδίδει πολύ καλύτερα και γρηγορότερα στην επεξεργασία εικόνας παρά στην επεξεργασία κειμένου. Οι οπτικές αναπαραστάσεις μεγάλου όγκου δεδομέ-

νων καθιστούν εφικτή τη μετάφραση της πληροφορίας και την ανάδειξη πιθανών υποκείμενων σχέσεων στο σύνολο των δεδομένων. Ως αποτέλεσμα, η οπτικοποίηση έχει αποκτήσει σημαντικό ρόλο και βρίσκει εφαρμογές σε διάφορες περιοχές όπως η διαχείριση δικτύων (υπολογιστών, τηλεπικοινωνιακών, κοινωνικών), η επιχειρηματική μοντελοποίηση, η βιοπληροφορική, ο σχεδιασμός βάσεων δεδομένων, η χαρτογραφία και ούτω καθεξής. Η ανάγκη για προηγμένες τεχνικές οπτικοποίησης οδήγησε στην ανάπτυξη της περιοχής της *Απεικόνισης Γραφημάτων (Graph Drawing)*, η οποία είναι σήμερα μία ενεργή ερευνητική περιοχή της επιστήμης των υπολογιστών που αφορά στην οπτικοποίηση πληροφοριών με κατευθυνόμενα και μη-κατευθυνόμενα γραφήματα.

Ένα *γράφημα (graph)* είναι ένα αφηρημένο μαθηματικό μοντέλο, το οποίο μοντελοποιεί διμερείς σχεσιακές δομές, αποτελούμενες από οντότητες και διμερείς σχέσεις μεταξύ αυτών των οντοτήτων. Οι οντότητες αντιστοιχούν στις *κορυφές (vertices)* του γραφήματος, ενώ οι σχέσεις αντιστοιχούν στις *ακμές (edges)* του. Οι *απεικονίσεις γραφημάτων (drawings of graphs)* είναι αρκετά συνήθεις στην καθημερινή μας ζωή, ακόμα και αν οι άνθρωποι δεν το γνωρίζουν. Ζώντας σε μία πόλη με μετρό, οι επιβάτες αναζητούν κατευθύνσεις σε χάρτες-μετρό, στους οποίους το γράφημα απεικονίζει πληροφορίες σχετικά με τις γραμμές του μετρό. Κάθε κορυφή ενός τέτοιου γραφήματος αντιστοιχεί σε ένα σταθμό του δικτύου και συνήθως απεικονίζεται ως κύκλος. Οι γραμμές που συνδέουν τους σταθμούς αντιστοιχούν στις ακμές του γραφήματος και απεικονίζονται ως ευθύγραμμο τμήματα. Τα οργανογράμματα αποτελούν ένα ακόμα παράδειγμα της απεικόνισης γραφημάτων, τα οποία περιγράφουν την ιεραρχική δομή ενός οργανισμού και τις σχέσεις μεταξύ του προσωπικού που τον συναποτελεί. Τα παραπάνω παραδείγματα αποτελούν ένα πολύ μικρό δείγμα από το τεράστιο σύνολο των πιθανών εφαρμογών της απεικόνισης γραφημάτων. Πράγματι, η απεικόνιση γραφημάτων προσφέρει τις υπηρεσίες της σε ένα μεγάλο αριθμό πεδίων, όπως η μηχανική λογισμικού (διαγράμματα ροής δεδομένων, ιεραρχίες αντικειμενοστρεφών κλάσεων), οι βάσεις δεδομένων (σχεσιακά διαγράμματα), τα πληροφοριακά συστήματα (οργανογράμματα), ο σχεδιασμός VLSI κυκλωμάτων (σχηματικά διαγράμματα κυκλωμάτων), η ασφάλεια δικτύων, η τεχνητή νοημοσύνη (διαγράμματα αναπαράστασης γνώσης) και δεν περιορίζεται μόνο στην επιστήμη των υπολογιστών.

## 11.2 Μία σύντομη ιστορική αναδρομή

Είναι δύσκολο να εντοπιστεί η ακριβής προέλευση της απεικόνισης γραφημάτων. Γεωμετρικές απεικονίσεις (δηλαδή απεικονίσεις γραφημάτων στις οποίες κάθε ακμή απεικονίζεται σαν ένα ευθύγραμμο τμήμα) διαφόρων ειδών έχουν εμφανιστεί ευρέως ανά τους αιώνες. Το παιχνίδι του Morris αποτελεί πιθανόν την πρώτη μορφή απεικόνισης γραφημάτων,



το οποίο σύμφωνα με το Henry Parker ([Hen09], p.p. 578) χρονολογείται στο 1400 π.Χ. στην Αίγυπτο. Ωστόσο, δεν υπάρχουν διαθέσιμες εικόνες. Το συγκεκριμένο παιχνίδι παρουσιάζεται, επίσης, στο “Βιβλίο των Παιχνιδιών” (“Book of Games”) από τον 13ο αιώνα [oCaL51]. Τα οικογενειακά δέντρα αποτελούν έναν ακόμα τύπο αρχαίας απεικόνισης γραφημάτων (και πάλι χωρίς εικόνες), όπως περιγράφεται από τους Pliny the Elder and Seneca ([KZ91], p.p. 111). Οι πρώτες εικόνες γενεαλογικών δέντρων εμφανίζονται στο Μεσαίωνα [KZ91]. Τετράγωνα αντίθεσης (squares of opposition) έχουν, επίσης, χρησιμοποιεί στο Μεσαίωνα για την αναπαράσταση λογικών διαγραμμάτων [Mur84].

Παρόλο που ο Euler βασίστηκε σε ένα γράφημα για να λύσει το πρόβλημα των “Επτά Γέφυρων του Königsberg” (“Königsberger Brückenproblem”) στη διάσημη εργασία του [Eul36] το 1776, στην πραγματικότητα δε χρησιμοποίησε απεικονίσεις γραφημάτων για να παρουσιάσει τα αποτελέσματά του. Γι’ αυτό το λόγο, ακόμα και αν κάποιος μπορεί να θεωρήσει ότι η θεωρία γραφημάτων και η απεικόνιση γραφημάτων συμπίπτουν χρονικά, στην πραγματικότητα οι απεικονίσεις γραφημάτων ξεκινούν αρκετά χρόνια αργότερα, στα τέλη του 18ου και στις αρχές του 19ου αιώνα. Μεταξύ αυτών, ο Listing το 1847 μελέτησε το πρόβλημα του εντοπισμού μονοπατιών (path tracing problem) και παρουσίασε ένα γράφημα το οποίο μπορεί να σχεδιαστεί μονοκοντυλιά (single stroke), δηλαδή ακολουθώντας μία διαδρομή χωρίς επανάληψη ακμών. Ο Cayley παρουσίασε τις πρώτες απεικονίσεις δέντρων [Cay57]. Για μία πιο λεπτομερή παρουσίαση της ιστορίας της απεικόνισης γραφημάτων, ο αναγνώστης παραπέμπεται στην εργασία [KMBW02].

Παρόλο που η πρώτη εργασία με αλγορίθμους απεικόνισης γραφημάτων μπορεί να εντοπιστεί το 1963, στην εργασία του Tutte “How to Draw a Graph” [Tut63], οι ερευνητές της απεικόνισης γραφημάτων έχουν οργανωθεί σε μία ενεργή κοινότητα τις τελευταίες δύο δεκαετίες (το πρώτο ετήσιο συμπόσιο απεικόνισης γραφημάτων πραγματοποιήθηκε στο Παρίσι, το 1993). Σε αυτά τα χρόνια, οι ερευνητές έχουν αναπτύξει ένα μεγάλο αριθμό διαφορετικών τρόπων απεικόνισης, οι οποίες είναι κατάλληλες για αρκετές εφαρμογές (οργανογράμματα, βάσεις δεδομένων, δίκτυα, διαγράμματα ροών, ιεραρχίες, επίπεδες απεικονίσεις, δέντρα, δενδροειδείς απεικονίσεις, κτλ), έχουν αναπτύξει τεχνικές για τη σύγκριση απεικονίσεων και την αξιολόγηση της ποιότητάς τους (αριθμό τομών, αριθμό σημείων καμπής, γωνιακή ευκρίνεια, κτλ) και έχουν αναπτύξει και αναλύσει αλγορίθμους για την παραγωγή απεικονίσεων υψηλής αισθητικής. Για μία πιο λεπτομερή ανάλυση των αλγορίθμων απεικόνισης γραφημάτων, ο αναγνώστης παραπέμπεται σε κλασικά βιβλία απεικόνισης γραφημάτων [DETT99, KW01, NR04].

### 11.3 Παράμετροι απεικόνισης

Από την ανθρώπινη πλευρά, η χρησιμότητα μιας απεικόνισης ενός γραφήματος εξαρτάται σημαντικά από την αναγνωσιμότητα της, δηλαδή την ικανότητά της να μεταβιβάζει την πληροφορία που εμπεριέχεται σε ένα γράφημα γρήγορα και κατανοητά. Μία “φτωχή” απεικόνιση μπορεί να προκαλέσει σύγχυση και παραπλάνηση. Ωστόσο, πώς μπορούμε να κρίνουμε την ποιότητα μιας απεικόνισης; Εάν προσπαθήσουμε να απεικονίσουμε μία κοινότητα ενός κοινωνιολογικού δικτύου (π.χ. την κοινότητα του facebook) χρησιμοποιώντας έναν αλγόριθμο που έχει σχεδιαστεί για την απεικόνιση χαρτών δικτύων μετρό, θα απογοητευτούμε πλήρως από το αποτέλεσμα. Ο λόγος είναι ότι ένας αλγόριθμος που παράγει απεικονίσεις χαρτών δικτύων μετρό λαμβάνει υπόψη του μία γενική γνώση, η οποία είναι σημαντική για τους χάρτες μετρό. Από την άλλη μεριά, η απεικόνιση μιας κοινότητας ενός κοινωνιολογικού δικτύου θα πρέπει να αναδεικνύει τις σχέσεις μεταξύ των μελών της και αυτή η απαίτηση δεν μπορεί να υλοποιηθεί από έναν αλγόριθμο παραγωγής χαρτών δικτύων μετρό.

Ένα εύλογο ερώτημα το οποίο προκύπτει από την παραπάνω εισαγωγή είναι το αν χρειαζόμαστε για κάθε τομέα εφαρμογής, ένα διαφορετικό αλγόριθμο απεικόνισης, ο οποίος κοιτάει πέρα από τις δομικές ιδιότητες του γραφήματος και επιπλέον χρησιμοποιεί πρόσθετη γνώση για τη σημασιολογία πίσω από τη δομή του γραφήματος. Μία καταφατική απάντηση σε αυτή την ερώτηση θα υπονοούσε την ανάγκη ενός τεράστιου αριθμού διαφορετικών αλγορίθμων απεικόνισης γραφημάτων, έναν για κάθε πεδίο εφαρμογής. Ευτυχώς, στην πραγματικότητα δε συμβαίνει αυτό. Υπάρχουν μόνο μερικές έννοιες και τεχνικές, ικανές να καλύψουν ένα μεγάλο αριθμό πιθανών πεδίων εφαρμογής. Για το σκοπό αυτό, η κοινότητα της απεικόνισης γραφημάτων έχει εισάγει γενικές συμβάσεις απεικόνισης (*drawing conventions*) και αισθητικά κριτήρια (*aesthetics criteria*).

#### 11.3.1 Συμβάσεις

Μία απεικόνιση ενός γραφήματος θα πρέπει να ικανοποιεί ένα βασικό κανόνα που ονομάζεται *σύμβαση της απεικόνισης* (*drawing convention*) ώστε να είναι αποδεκτή. Για παράδειγμα, σε ένα χάρτη μετρό, μία σύμβαση που χρησιμοποιείται συχνά είναι η ακόλουθη: οι κορυφές του γραφήματος που αντιστοιχούν στους σταθμούς απεικονίζονται ως κύκλοι και όλες οι ακμές απεικονίζονται ως παράλληλα προς τους άξονες ευθύγραμμα τμήματα ή ως ακολουθίες από παράλληλα προς τους άξονες ευθύγραμμα τμήματα. Η σύμβαση της απεικόνισης εξαρτάται από τη φύση των δεδομένων που οπτικοποιούνται και μπορεί να είναι πολύ απλή ή αρκετά πολύπλοκη, ενώ μπορεί να περιέχει λίγες ή πολλές λεπτομέρειες για την απεικόνιση. Μπορούμε, επίσης, να υιοθετήσουμε διαφορετικούς τρόπους για την αναπαράσταση του γραφήματος. Για παράδειγμα, οι κορυφές μπορεί να απεικονίζονται σαν τε-

τράγωνα, τελείες, κύκλοι, ετικέτες ή σαν συνδυασμός διαφορετικών τρόπων απεικόνισης. Πληροφορίες σχετικά με τις κορυφές και τις ακμές μπορούν να οπτικοποιηθούν, επίσης, με ετικέτες κειμένου, διαφορετικά χρώματα, διαφορετικό πάχος και ούτω καθεξής. Παρακάτω παρουσιάζουμε ορισμένες συμβάσεις απεικόνισης:

**Επίπεδες απεικονίσεις:** Μία απεικόνιση ενός γραφήματος στην οποία οι ακμές δεν τέμνονται μεταξύ τους ονομάζεται *επίπεδη* (*planar*). Ένα γράφημα το οποίο επιδέχεται μία τέτοια απεικόνιση ονομάζεται *επίπεδο γράφημα* (*planar graph*). Δυστυχώς, ένας μικρός αριθμός γραφημάτων επιδέχονται επίπεδες απεικονίσεις. Για παράδειγμα, είναι γνωστό ότι ένα επίπεδο γράφημα με  $n$  κορυφές δεν μπορεί να έχει περισσότερες από  $3n - 6$  ακμές. Σημειώνεται ότι ο μέγιστος αριθμός κορυφών ενός γραφήματος είναι  $n(n - 1)/2$  (*γράφημα κλίκα*).

**Πολυγραμμικές απεικονίσεις:** Μία απεικόνιση στην οποία κάθε ακμή του γραφήματος απεικονίζεται σαν πολυγωνική αλυσίδα ονομάζεται *πολυγραμμική απεικόνιση* (*polyline drawing*). Το σημείο στο οποίο μία ακμή αλλάζει την κατεύθυνσή της αναφέρεται ως *σημείο καμπής* (*bend*). Οι πολυγραμμικές απεικονίσεις είναι συνήθεις στην παραγωγή χαρτών δικτύων μετρό.

**Απεικονίσεις με ευθείες γραμμές:** Μία απεικόνιση στην οποία κάθε ακμή απεικονίζεται σαν ένα ευθύγραμμο τμήμα ονομάζεται *απεικόνιση με ευθείες γραμμές* (*straight-line drawing*). Οι απεικονίσεις με ευθείες γραμμές είναι μία ειδική περίπτωση των πολυγραμμικών απεικονίσεων, στις οποίες κάθε ακμή απεικονίζεται χωρίς σημεία καμπής. Τέτοιες απεικονίσεις αναφέρονται και ως *γεωμετρικές απεικονίσεις* (*geometric drawings*). Είναι γνωστό ότι κάθε επίπεδο γράφημα επιδέχεται μία (επίπεδη) απεικόνιση με ευθείες γραμμές.

**Ορθογώνιες απεικονίσεις:** Μία απεικόνιση στην οποία κάθε ακμή αναπαρίστανται σαν μία εναλλασσόμενη ακολουθία από παράλληλα προς τους άξονες ευθύγραμμα τμήματα ονομάζεται *ορθογώνια απεικόνιση* (*orthogonal drawing*). Τα γραφήματα τα οποία επιδέχονται ορθογώνιες απεικονίσεις είναι συνήθως μεγίστου βαθμού τέσσερα, όπου ο βαθμός ενός γραφήματος ορίζεται ως ο μέγιστος αριθμός ακμών που προσπίπτουν σε μία κορυφή του.

**Απεικονίσεις σε πλέγματα:** Μία απεικόνιση στην οποία κάθε κορυφή, τομή και/ή σημείο καμπής καταλαμβάνει ένα σημείο στο επίπεδο με ακέραιες συντεταγμένες ονομάζεται *απεικόνιση σε πλέγμα* (*grid drawing*). Οι απεικονίσεις σε πλέγματα είναι σημαντικές σε πολλές εφαρμογές π.χ. στην περίπτωση που το γράφημα προβάλλεται σε οθόνη χαμηλής ανάλυσης.

**Ανερχόμενες (κατερχόμενες, αντίστοιχα) απεικονίσεις:** Μία απεικόνιση στην οποία κάθε ακμή είναι μία απλή ανοιχτή καμπύλη μονότονα αύξουσα (φθίνουσα, αντίστοιχα) στην κάθετη κατεύθυνση ονομάζεται *ανερχόμενη (κατερχόμενη, αντίστοιχα) απεικόνιση*. Για παράδειγμα, τα γενεαλογικά δέντρα συνήθως απεικονίζονται προς τα κάτω για να αναδείξουν την ιεραρχία.

### 11.3.2 Αισθητικά κριτήρια

Η ποιότητα ενός γραφήματος υπολογίζεται με βάση *αισθητικά κριτήρια (aesthetics criteria)* τα οποία χρησιμοποιούνται για την αξιολόγηση του πόσο “καλή” είναι μία απεικόνιση. Οι αλγόριθμοι απεικόνισης γραφημάτων τυπικά συμμορφώνονται με ένα ή περισσότερα αισθητικά κριτήρια με την προϋπόθεση ότι η ποιότητα της απεικόνισης βελτιώνεται. Για παράδειγμα, σε ένα VLSI κύκλωμα τα σημεία καμπής θα πρέπει να ελαχιστοποιούνται, καθώς αυξάνουν το κατασκευαστικό κόστος και συνεπώς, το ενδιαφέρον στρέφεται σε επίπεδες απεικονίσεις με τον ελάχιστο αριθμό σημείων καμπής. Μία λίστα από συνήθη αισθητικά κριτήρια [BFN85, PCA02] παρουσιάζεται παρακάτω:

**Ελαχιστοποίηση του αριθμού των τομών:** Οι τομές μεταξύ των ακμών επηρεάζουν αρνητικά την ποιότητα μιας απεικόνισης, καθώς είναι κοινά αποδεκτό ότι δημιουργούν σύγχυση στον αναγνώστη της. Ιδανικά, θα προτιμούσαμε τις επίπεδες απεικονίσεις. Ωστόσο, όπως έχει ήδη αναφερθεί, αυτές δεν είναι πάντοτε εφικτό να κατασκευαστούν. Σε αυτή την περίπτωση, θα επιθυμούσαμε απεικονίσεις με τον ελάχιστο αριθμό τομών.

**Ελαχιστοποίηση των σημείων καμπής:** Ο συνολικός αριθμός των σημείων καμπής και ο αριθμός των σημείων καμπής ανά ακμή σε μία απεικόνιση θα πρέπει να παραμένει μικρός. Τα σημεία καμπής προκαλούν σύγχυση, καθώς διακόπτουν την κίνηση του ματιού και απαιτούν απότομες αλλαγές κατεύθυνσης. Αυτό είναι ένα κρίσιμο κριτήριο στις ορθογώνιες απεικονίσεις, όπου ένα μεγάλο μέρος της έρευνας έχει επικεντρωθεί σε αυτό το πρόβλημα.

**Ελαχιστοποίηση του εμβαδού:** Η ελαχιστοποίηση του εμβαδού μιας απεικόνισης είναι σημαντική σε πολλές εφαρμογές, στις οποίες ο διαθέσιμος χώρος είναι περιορισμένος, π.χ. οι κινητές συσκευές είναι εφοδιασμένες με μία σχετικά μικρή οθόνη και αρκετά προβλήματα αναγνωσιμότητας μπορεί να προκύψουν εξαιτίας του περιορισμένου χώρου. Από την άλλη, για να μπορέσει να προβληθεί μία μεγάλη απεικόνιση στο διαθέσιμο χώρο θα πρέπει να γίνει σμίκρυνση ή να χαμηλώσει η ανάλυσή της, επηρεάζοντας έτσι αρνητικά την αναγνωσιμότητά της.

**Μεγιστοποίηση των γωνιών:** Η *γωνιακή ευκρίνεια* (*angular resolution*) ενός γραφήματος ορίζεται ως η μικρότερη γωνία που σχηματίζεται μεταξύ δύο γειτονικών ακμών που προσπίπτουν στην ίδια κορυφή. Η *γωνιακή ευκρίνεια τομών* (*crossing resolution*) αναφέρεται στη μικρότερη γωνία που σχηματίζεται από ένα ζεύγος τεμνόμενων ακμών. Η μεγιστοποίηση της μικρότερης γωνίας σε μία απεικόνιση είναι σημαντική σε αρκετές εφαρμογές, π.χ. σε κινητές συσκευές οι οποίες συχνά είναι εφοδιασμένες με οθόνες χαμηλής ανάλυσης.

**Ελαχιστοποίηση του μήκους:** Σε πολλές εφαρμογές, είναι σημαντικό να διατηρείται το μήκος των ακμών μικρό. Για παράδειγμα, στα κυκλώματα VLSI, το συνολικό μήκος των ακμών της απεικόνισης αντιστοιχεί στο μήκος των καλωδίων που χρειάζεται να χρησιμοποιηθεί, και συνεπώς, θα πρέπει να διατηρείται μικρό.

**Ελαχιστοποίηση αναλογίας διαστάσεων:** Η *αναλογία διαστάσεων* (*aspect ratio*) μιας απεικόνισης ενός γραφήματος ορίζεται ως ο λόγος του μήκους της μεγαλύτερης πλευράς προς το μήκος της μικρότερης πλευράς του μικρότερου ορθογωνίου που εσωκλείει την απεικόνιση. Η ελαχιστοποίηση αυτού του κριτηρίου παρέχει την απαιτούμενη ευελιξία στην προσαρμογή της απεικόνισης σε οθόνες τυχαίου μεγέθους.

**Συμμετρίες:** Η ανάδειξη των συμμετρικών ιδιοτήτων ενός γραφήματος είναι σημαντική, αλλά όχι εύκολη διαδικασία. Συνήθως εισάγονται κατάλληλα μαθηματικά μοντέλα για τη διαχείριση των συμμετριών σε απεικονίσεις γραφημάτων, π.χ. βλ. [Ead88, MAC<sup>+</sup>95].

**Ομαδοποίηση:** Σε πολλές περιπτώσεις και ιδιαίτερα σε μεγάλα και πυκνά γραφήματα, είναι σημαντικό οι κορυφές του γραφήματος με όμοιες ιδιότητες να τοποθετούνται κοντά, έτσι ώστε να αναδεικνύεται η δομή του γραφήματος.

Δυστυχώς, δεν υπάρχει σαφής τρόπος διάταξης των παραπάνω κριτηρίων (ως προς την σημαντικότητά τους). Η σημαντικότητά τους βασίζεται σε μεγάλο βαθμό από το πεδίο εφαρμογής τους. Ωστόσο, η Purchase [Pur97] μέσω γνωστικών πειραμάτων παρατήρησε ότι η ελαχιστοποίηση των τομών φαίνεται να είναι το πιο σημαντικό κριτήριο για την αναγνωσιμότητα μιας απεικόνισης, ενώ η ελαχιστοποίηση των σημείων καμπής και η μεγιστοποίηση των συμμετριών φαίνεται να επηρεάζουν λιγότερο την ποιότητά της. Επίσης, καθόρισε μία προτεραιότητα στα παραπάνω κριτήρια (βλέπε, επίσης, [PCJ97, Pur02]). Από την άλλη μεριά, άλλες ερευνητικές εργασίες διαπιστώνουν ότι οι διαφορές στην αντίληψη ενός γραφήματος εξαρτώνται από την τελική απεικόνιση [BMK96, MBK97, DC98].

Παρόλο που τα παραπάνω κριτήρια είναι γενικώς αποδεκτά, είναι σχεδόν αδύνατον να ικανοποιηθούν όλοι οι αισθητικοί κανόνες που ορίζουν ταυτοχρόνως. Μερικοί από αυτούς είναι αντικρουόμενοι. Άλλοι έχουν μεγάλες υπολογιστικές απαιτήσεις. Για παράδειγμα, η

ελαχιστοποίηση του αριθμού των τομών είναι ένα  $\mathcal{NP}$ -πλήρες πρόβλημα [GJ83]. Το ίδιο ισχύει και για το πρόβλημα του καθορισμού εάν ένα γράφημα μπορεί να εμφυτευτεί σε ένα πλέγμα προκαθορισμένου μεγέθους [MRK84]. Αρκετά αποτελέσματα σχετικά με την πολυπλοκότητα της βελτιστοποίησης των παραπάνω κριτηρίων έχουν παρουσιαστεί στο βιβλίο [GJ79]. Στην πράξη, οι απεικονίσεις γραφημάτων είναι συνήθως το αποτέλεσμα συμβιβασμών μεταξύ διαφορετικών αισθητικών κριτηρίων, ενώ γενικές προσεγγίσεις που χρησιμοποιούνται συνήθως για την παραγωγή τους περιλαμβάνουν τόσο ευρετικούς όσο και προσεγγιστικούς αλγορίθμους.

## 11.4 Βασικοί Ορισμοί της Θεωρίας Γραφημάτων

Τυπικά, ένα *γράφημα*  $G$  είναι ένα μη-διατεταγμένο ζεύγος  $(V, E)$ , όπου  $V$  είναι το σύνολο των *κορυφών* και  $E$  είναι το σύνολο των *ακμών* του γραφήματος  $G$ . Σημειώνεται ότι το σύνολο των κορυφών (ακμών) του  $G$  συμβολίζεται και ως  $V(G)$  ( $E(G)$ ). Ένα *υπογράφημα* (*subgraph*) ενός γραφήματος  $G$  είναι ένα γράφημα  $G' = (V', E')$  τέτοιο ώστε  $V' \subseteq V$  και  $E' \subseteq E$ . Δοθέντων δύο γραφημάτων  $G$  και  $G'$ , συμβολίζουμε με  $G \cup G'$  το γράφημα που προκύπτει από την ένωση των  $G$  και  $G'$ . Μία ακμή  $e \in E$  που συνδέει ένα ζεύγος κορυφών  $u, v \in V$  συνήθως συμβολίζεται με  $(u, v)$ , όπου οι κορυφές  $u$  και  $v$  ονομάζονται *άκρα* (*endpoints*) της ακμής  $e$ . Η ακμή  $e$  λέγεται ότι είναι *προσπίπτουσα* ή *προσκειμένη* (*incident*) σε καθένα από τα δύο άκρα της.

Εξ' ορισμού, τα μαθηματικά σύνολα είναι μη-διατεταγμένα. Επομένως, οι ακμές ενός γραφήματος δεν έχουν κατεύθυνση. Ένα τέτοιο γράφημα ονομάζεται *μη-κατευθυνόμενο* (*undirected*). Ωστόσο, σε πολλές εφαρμογές, μία ακμή έχει μία κατεύθυνση που αντιστοιχεί σε μία ειδική ιδιότητα μεταξύ ενός ζεύγους κορυφών, η οποία δεν είναι πάντα αμφίδρομη, π.χ. ένας μονόδρομος σε ένα δίκτυο συγκοινωνιών. Ένα γράφημα στο οποίο οι ακμές είναι διατεταγμένα ζεύγη κορυφών ονομάζεται *κατευθυνόμενο* (*directed*) γράφημα ή *διγράφημα* (*digraph*). Σε ένα διγράφημα, για μία ακμή  $(u, v)$ , η κορυφή  $u$  λέγεται *αφετηρία* (*source*), ενώ η κορυφή  $v$  λέγεται *προορισμός* (*target*). Επίσης, σε γραφήματα με κατευθυνόμενες ακμές, μία ακμή της μορφής  $(u, u)$  ονομάζεται *βρόγχος* (*self-loop* ή απλά *loop*). Ανάλογα, ένα γράφημα που δεν περιέχει βρόγχους ονομάζεται *γράφημα χωρίς βρόγχους* (*loop-less*). Ακμές που έχουν τα ίδια άκρα λέγονται *παράλληλες ακμές* (*parallel edges*). Ένα γράφημα χωρίς βρόγχους και παράλληλες ακμές ονομάζεται *απλό* (*simple*).

Σε ένα μη-κατευθυνόμενο γράφημα, ο *βαθμός* (*degree*) μιας κορυφής  $v$  συμβολίζεται συνήθως με  $d(v)$  και είναι ο αριθμός των ακμών που προσπίπτουν στην κορυφή  $v$ . Σε ένα κατευθυνόμενο γράφημα, ο *έσω-βαθμός* (*in-degree*) μιας κορυφής  $v$ , που συμβολίζεται με  $d^-(v)$ , είναι ο αριθμός των ακμών που έχουν την κορυφή  $v$  ως προορισμό τους. Αντίστοιχα, ο *έξω-*

*βαθμός (out-degree)* μιας κορυφής  $v$ , ο οποίος συμβολίζεται με  $d^+(v)$ , είναι ο αριθμός των ακμών που έχουν την κορυφή  $v$  ως αφετηρία τους. Μία κορυφή βαθμού μηδέν ονομάζεται *απομονωμένη (isolated)*. Ο *βαθμός (degree)* ενός γραφήματος συμβολίζεται συνήθως με  $d(G)$  και είναι ο μέγιστος βαθμός των κορυφών του  $G$ . Ένα *κανονικό γράφημα (regular graph)* είναι ένα γράφημα στο οποίο κάθε κορυφή έχει τον ίδιο βαθμό. Ένα κανονικό κατευθυνόμενο γράφημα θα πρέπει να ικανοποιεί μία επιπλέον ισχυρότερη συνθήκη, σύμφωνα με την οποία ο έσω-βαθμός και ο έξω-βαθμός κάθε κορυφής θα πρέπει να είναι ίσοι μεταξύ τους. Ένα κανονικό γράφημα με κορυφές βαθμού  $k$  ονομάζεται  *$k$ -κανονικό γράφημα ( $k$ -regular graph)* ή *κανονικό γράφημα βαθμού  $k$  (regular graph of degree  $k$ )*. Η *πυκνότητα (density)* ενός μη-κατευθυνόμενου γραφήματος  $G$  ισούται με το λόγο των ακμών του  $G$  προς το μέγιστο δυνατό αριθμό ακμών.

Ένα *μονοπάτι μήκους  $k$  (path of length  $k$ )* από μία κορυφή  $u$  προς μία κορυφή  $v$  του γραφήματος  $G$  είναι μία ακολουθία  $\langle v_0, v_1, v_2, \dots, v_k \rangle$  κορυφών τέτοια ώστε  $u = v_0, v = v_k$  και  $(v_{i-1}, v_i) \in E$ , για κάθε  $i = 1, 2, \dots, k$ . Το *μήκος (length)* του μονοπατιού αντιστοιχεί στον αριθμό των ακμών του. Ένα μονοπάτι λέγεται *απλό (simple)* αν καμία κορυφή δεν εμφανίζεται πάνω από μία φορά. Ένα μονοπάτι  $\langle v_0, v_1, v_2, \dots, v_k \rangle$  σχηματίζει *κύκλο (cycle)* εάν  $v_0 = v_k$  και αποτελείται από τουλάχιστον μία ακμή. Ένας κύκλος ονομάζεται *απλός (simple)* εάν καμία κορυφή δεν εμφανίζεται πάνω από μία φορά. Ένα γράφημα που δεν περιέχει κύκλους ονομάζεται *ακυκλικό (acyclic)*. Ένα *μονοπάτι hamilton (hamiltonian path)* ενός γραφήματος  $G$  είναι ένα μονοπάτι που περνάει από κάθε κορυφή του  $G$ . Ένας *κύκλος hamilton (hamiltonian cycle)* του γραφήματος  $G$  είναι ένας κύκλος που περνάει από κάθε κορυφή του  $G$ . Ένα *δέντρο (tree)* είναι ένα ακυκλικό γράφημα στο οποίο κάθε ζεύγος κορυφών συνδέεται με ένα μονοπάτι. Εάν μία κορυφή ενός δέντρου διακρίνεται από τις άλλες κορυφές, τότε το δέντρο ονομάζεται *δέντρο με ρίζα (rooted tree)*. Ένα *γράφημα-κάμπια (caterpillar)* είναι ένα δέντρο στο οποίο όλες οι κορυφές βρίσκονται σε απόσταση ένα από ένα κεντρικό μονοπάτι. Ένα *γράφημα-τροχός (wheel)  $W_n$*  είναι ένα γράφημα με  $n$  κορυφές (όπου  $n \geq 4$ ) το οποίο σχηματίζεται ενώνοντας μία κορυφή με όλες τις κορυφές ενός  $(n - 1)$ -κύκλου. Ένα *ταίριασμα (matching)* ενός γραφήματος  $G$  είναι ένα σύνολο ακμών  $E' \subseteq E$ , τέτοιο ώστε όλες οι ακμές του  $E'$  να μην έχουν ένα κοινό άκρο.

Ένα μη-κατευθυνόμενο γράφημα λέγεται *πλήρες (complete)* ή *κλίκα (clique)* αν κάθε ζεύγος κορυφών συνδέονται μεταξύ τους με μία ακμή. Το πλήρες γράφημα με  $n$  κορυφές συμβολίζεται με  $K_n$ . Ένα *διμερές γράφημα (bipartite graph)* είναι ένα γράφημα του οποίου οι κορυφές μπορούν να χωριστούν σε δύο ανεξάρτητα σύνολα  $V_1$  και  $V_2$ , τέτοια ώστε κάθε ακμή έχει ένα άκρο σημείο στο  $V_1$  και ένα άκρο στο  $V_2$ . Ένα *πλήρες διμερές γράφημα (complete bipartite graph)* είναι ένα διμερές γράφημα, στο οποίο κάθε κορυφή του  $V_1$  ενώνεται με κάθε κορυφή του  $V_2$  με ακριβώς μία ακμή. Αν  $r$  ( $s$ , αντίστοιχα) είναι ο αριθμός των κορυφών του

$V_1$  ( $V_2$ , αντίστοιχα), τότε το πλήρες διμερές γράφημα συμβολίζεται με  $K_{r,s}$ . Ένα *γράφημα-άστρο* (*star graph*)  $S_n$  με  $n$  κορυφές ορίζεται ως το πλήρες διμερές γράφημα  $K_{1,n-1}$ . Ένα *κατευθυνόμενο ακυκλικό γράφημα* (*directed acyclic graph* ή *DAG* εν συντομία) είναι ένα κατευθυνόμενο γράφημα χωρίς κατευθυνόμενους κύκλους. Δύο γραφήματα ονομάζονται *ισομορφικά* (*isomorphic*) αν υπάρχει μία ένα-προς-ένα αντιστοιχία μεταξύ των συνόλων των κορυφών και η γειτνίαση διατηρείται.

Ένα μη-κατευθυνόμενο γράφημα είναι *συνδεδεμένο* (*connected*) εάν κάθε ζεύγος κορυφών του συνδέεται με ένα μονοπάτι. Τα *συνδεδεμένα συστατικά* (*connected components*) ενός μη-κατευθυνόμενου γραφήματος είναι κλάσεις ισοδυναμίας των κορυφών του που ορίζονται από τη σχέση “είναι προσπελάσιμη από”. Ένα κατευθυνόμενο γράφημα ονομάζεται *ισχυρά συνδεδεμένο* (*strongly connected*) εάν υπάρχει ένα μονοπάτι από κάθε κορυφή του γραφήματος προς κάθε άλλη κορυφή. Τα ισχυρά συνδεδεμένα συστατικά ενός κατευθυνόμενου γραφήματος είναι κλάσεις ισοδυναμίας των κορυφών του γραφήματος που ορίζονται από τη σχέση “είναι αμοιβαία προσπελάσιμοι”. Μία *γέφυρα* (*bridge*) είναι μία ακμή της οποίας η αφαίρεση αποσυνδέει ένα γράφημα. Αναλόγως, μία *κορυφή τομή* ή *σημείο άρθρωσης* (*cut-vertex* ή *articulation point*) είναι μία κορυφή της οποίας η αφαίρεση αποσυνδέει το γράφημα. Ένα γράφημα χωρίς σημεία άρθρωσης ονομάζεται *διπλά συνδεδεμένο* ή *δισυνδεδεμένο* (*biconnected*). Ένα συνδεδεμένο γράφημα στο οποίο η διαγραφή οποιονδήποτε δύο κορυφών (και των προσκείμενων ακμών) καταλήγει σε ένα γράφημα που εξακολουθεί να είναι συνδεδεμένο, ονομάζεται *τριπλά συνδεδεμένο* ή *τρισυνδεδεμένο* (*triconnected*) γράφημα.

Μία *απεικόνιση* (*drawing*) ενός απλού γραφήματος  $G$  (συνήθως συμβολίζεται με  $\Gamma(G)$ ) είναι μία αναπαράσταση του  $G$  στο επίπεδο, όπου κάθε κορυφή είναι ένα σημείο και κάθε ακμή είναι μία κλειστή καμπύλη Jordan μεταξύ των σημείων που αναπαριστούν τα άκρα της. Με  $x(v)$  και  $y(v)$  ορίζουμε τη  $x$ - και τη  $y$ -συντεταγμένη μιας κορυφής  $v \in V$  στην απεικόνιση  $\Gamma(G)$ . Κάθε επίπεδη απεικόνιση ορίζει μοναδικά μία κυκλική διάταξη των ακμών που προσπίπτουν στην ίδια κορυφή και επομένως, καθορίζει μία *συνδυαστική εμφύτευση* (*combinatorial embedding*).

Ένα επίπεδο γράφημα με καθορισμένη συνδυαστική εμφύτευση ονομάζεται *επιπεδικό γράφημα* (*plane graph*). Σε μία επίπεδη απεικόνιση, κάθε κύκλος που περιβάλλει μία περιοχή, η οποία δεν περιέχει ακμές με άκρα από τον κύκλο προς την περιοχή, σχηματίζει μία *όψη* (*face*). Έστω  $G$  ένα επιπεδικό γράφημα. Το *δυϊκό γράφημα* (*dual graph*) του  $G$ , το οποίο συνήθως συμβολίζεται με  $G^*$ , έχει μία κορυφή για κάθε όψη του  $G$  και μία ακμή μεταξύ δύο όψεων  $f_1$  και  $f_2$  αν και μόνο αν οι  $f_1$  και  $f_2$  μοιράζονται μία ακμή. Το υπογράφημα του  $G^*$ , που παράγεται από τη διαγραφή της κορυφής-όψης του  $G^*$  που αντιστοιχεί στη μη-φραγμένη όψη του  $G$ , ονομάζεται *ασθενής δυϊκός* (*weak dual*) του  $G$ . Ένα επίπεδο γράφημα καλείται *μέγιστο επίπεδο* (*maximal planar*) αν δεν μπορούν να προστεθούν άλλες ακ-



μές ώστε το προκύπτον γράφημα να εξακολουθεί να είναι επίπεδο. Ένα *τριγωνικό γράφημα* (*triangular graph*) είναι ένα μέγιστο επιπεδικό γράφημα το οποίο έχει μόνο τρεις κορυφές στο κυρτό περίβλημα (convex hull) του.

Ένα γράφημα το οποίο μπορεί να απεικονιστεί στο επίπεδο χωρίς τομές, με τέτοιο τρόπο ώστε όλες οι κορυφές να ανήκουν στη μη φραγμένη όψη της απεικόνισης, ονομάζεται *εξω-επίπεδο* (*outerplanar*). Ένα *εξωμονοπάτι* (*outerpath*) είναι ένα εξωεπίπεδο γράφημα τέτοιο ώστε ο ασθενής δυϊκός του είναι ένα μονοπάτι. Το *γεωμετρικό πάχος* (*geometric thickness*) ενός γραφήματος είναι ο ελάχιστος αριθμός επιπέδων που απαιτούνται για να παραχθεί μία γεωμετρική απεικόνιση του γραφήματος και να ανατεθεί κάθε ακμή σε ένα επίπεδο, έτσι ώστε ακμές που βρίσκονται στο ίδιο επίπεδο να μην τέμνονται.

## 11.5 Συνήθειες Τεχνικές Απεικόνισης Γραφημάτων

Σε αυτή την ενότητα, θα αναλύσουμε εν συντομία κάποιες συνήθειες τεχνικές απεικόνισης γραφημάτων από τη βιβλιογραφία. Για μία πιο εκτενή κατηγοριοποίηση αυτών των τεχνικών, ο αναγνώστης παραπέμπεται στις εργασίες [MGB<sup>+</sup>98, HSMM00]. Λεπτομερής περιγραφή των αντίστοιχων αλγορίθμων γίνεται παραγωγής τους σε κλασικά βιβλία της απεικόνισης γραφημάτων [DETT99, KW01].

### 11.5.1 Απεικόνιση επίπεδων γραφημάτων

Οι επίπεδες απεικονίσεις είναι σημαντικές σε πολλές πρακτικές εφαρμογές, στις οποίες οι τομές μεταξύ ακμών προκαλούν προβλήματα, όπως στο σχεδιασμό VLSI κυκλωμάτων ή οδικών δικτύων. Ωστόσο, είναι γνωστό ότι κάθε επίπεδο γράφημα με  $n$  κορυφές περιέχει το πολύ  $3n - 6$  ακμές. Συνεπώς, δεν είναι όλα τα γραφήματα επίπεδα και ένα μεγάλο μέρος της ερευνητικής προσπάθειας έχει επικεντρωθεί στο πρόβλημα του ελέγχου της επιπεδότητας ενός γραφήματος (planarity testing problem), δηλαδή στο πρόβλημα του ελέγχου εάν ένα γράφημα επιδέχεται μία επίπεδη απεικόνιση. Οι πρώτοι αλγόριθμοι πολυωνυμικού χρόνου για τον έλεγχο της επιπεδότητας ενός γραφήματος χρονολογούνται στη δεκαετία του '60 [AP61, Gol63, Bad64, DMP64, LEC67]. Μερικά χρόνια αργότερα, οι Hopcroft και Tarjan [HT74] ήταν οι πρώτοι που παρουσίασαν αλγόριθμο γραμμικού χρόνου για το πρόβλημα αυτό. Από τότε, πολλοί άλλοι αλγόριθμοι γραμμικού χρόνου έχουν προταθεί, βασισμένοι σε διαφορετικές προσεγγίσεις [BL76, CNA085, kSIH92, BT96, WKWL99, IH03a, dFdM96, dFdM12, Bra09]. Για μία επισκόπηση σχετική με τις επίπεδες απεικονίσεις γραφημάτων, ο αναγνώστης παραπέμπεται στο κλασικό βιβλίο των Nishizeki και Rahman [NR04].

Δεδομένου ότι όλα τα γραφήματα δεν επιδέχονται επίπεδες απεικονίσεις και το πρόβλημα της ελαχιστοποίησης των τομών είναι στη γενική του περίπτωση  $\mathcal{NP}$ -πλήρες [GJ83],

ένα ερώτημα που εύλογα προκύπτει είναι το εάν ένα γράφημα μπορεί να γίνει επίπεδο αφαιρώντας τον ελάχιστο αριθμό ακμών (*επιπεδοποίηση γραφήματος; graph planarization*). Σημειώνεται ωστόσο ότι, το αντίστοιχο πρόβλημα απόφασης είναι  $\mathcal{NP}$ -πλήρες [LY80] και συνεπώς, αρκετοί ευρετικοί αλγόριθμοι έχουν προταθεί [BTT84, GMW05]. Για μία πιο πλήρη επισκόπηση αυτών των αλγορίθμων, ο αναγνώστης παραπέμπεται στην εργασία [Wei01].

### 11.5.2 Force-directed μέθοδοι

Οι *force-directed αλγόριθμοι* χρησιμοποιούνται ευρέως για την απεικόνιση γραφημάτων [Ead84, FR91, DH96, KK89]. Ένας *force-directed αλγόριθμος* μοντελοποιεί το γράφημα σαν ένα φυσικό σύστημα με κατάλληλες δυνάμεις που δρουν σε αυτό. Στη συνέχεια, μία καλή απεικόνιση παράγεται από μία κατάσταση ισορροπίας του συστήματος. Οι *force-directed μέθοδοι* αποσκοπούν στην παραγωγή “αισθητικά καλών” απεικονίσεων. Αυτό συνήθως επιτυγχάνεται μέσω της διατήρησης ομοιόμορφου μήκους ακμών και της ανάδειξης τυχόν συμμετριών κατά τον καλύτερο τρόπο. Για την ειδική περίπτωση των επίπεδων γραφημάτων, οι αλγόριθμοι αυτοί τείνουν να οδηγούν σε απεικονίσεις χωρίς τομές μεταξύ των ακμών.

Ένας από τους πρώτους *force-directed αλγόριθμους* απεικόνισης γραφημάτων παρουσιάστηκε από τον Tutte [Tut63], ο οποίος βασίζεται σε βαρυκεντρικές αναπαραστάσεις. Ο κλασικός *force-directed αλγόριθμος* του Eades [Ead84] μοντελοποιεί τις κορυφές ενός γραφήματος ως ηλεκτρικά φορτισμένα σωματίδια που απωθούνται μεταξύ τους και τις ακμές ως ελατήρια που έλκουν γειτονικές κορυφές. Ο αλγόριθμος των Fruchterman και Reingold [FR91] στηρίζεται, επίσης, στις δυνάμεις των ελατηρίων (αλλά επιπλέον χρησιμοποιεί την έννοια της “θερμοκρασίας”). Αρκετές παραλλαγές του κλασικού αλγορίθμου του Eades έχουν προταθεί τα τελευταία χρόνια [KK89, FLM95, DH96, BHR96, Ber00]. Σημειώνεται ότι οι *force-directed αλγόριθμοι* απολαμβάνουν την αποδοχή της κοινότητας της Απεικόνισης Γραφημάτων. Αξίζει να σημειωθεί ότι, σε κάθε ετήσιο συνέδριο Απεικόνισης Γραφημάτων, υπάρχει τουλάχιστον μία εργασία, η οποία μοντελοποιεί ένα συγκεκριμένο πρόβλημα ως ένα σύστημα δυνάμεων και προτείνει έναν αντίστοιχο αλγόριθμο που το επιλύει. Μία επισκόπηση *force-directed* των μεθόδων και παραλλαγών τους μπορεί να αναζητηθεί σε κλασικά βιβλία απεικόνισης γραφημάτων [DETT99, Bra01, KW01, Kob12].

### 11.5.3 Ορθογώνιες απεικονίσεις

Η μελέτη ορθογώνιων απεικονίσεων έχει τις ρίζες της στο σχεδιασμό VLSI και στις floor-planning εφαρμογές [Lei80, Val81, TT89]. Μία *ορθογώνια απεικόνιση (orthogonal drawing)* ενός επίπεδου γραφήματος με μέγιστο βαθμό τέσσερα (ή 4-επίπεδου για συντομία) είναι μία απεικόνιση στην οποία κάθε κορυφή καταλαμβάνει ένα σημείο στο ακέραιο πλέγμα (integer grid) και κάθε ακμή απεικονίζεται ως μία εναλλασσόμενη ακολουθία από οριζόντια

και κάθετα ευθύγραμμα τμήματα. Σε μία ορθογώνια απεικόνιση, οι κορυφές βαθμού τέσσερα έχουν βέλτιστη γωνιακή ευκρίνεια, ενώ οι κορυφές χαμηλότερου βαθμού είναι κατά ένα παράγοντα της τάξης του δύο μακριά από το βέλτιστο. Οι ορθογώνιες απεικονίσεις γραφημάτων είναι συνήθως βέλτιστες ως προς μία αντικειμενική συνάρτηση, όπως για παράδειγμα, η ελαχιστοποίηση του συνολικού αριθμού των σημείων καμπής [Tam87, FK95, GT01] ή η ελαχιστοποίηση του μέγιστου αριθμού των σημείων καμπής ανά ακμή [BK94, LMS98]. Για μία πλήρη επισκόπηση, ο αναγνώστης παραπέμπεται στην εργασία [EFK01].

Στο πεδίο των ορθογώνιων απεικονίσεων και πιο συγκεκριμένα, για το πρόβλημα της ελαχιστοποίησης του συνολικού αριθμού σημείων καμπής, ο Tamassia [Tam87] εισήγαγε μία αρκετά σημαντική μεθοδολογική τεχνική με την προσέγγιση της “τοπολογίας-σχήματος-μετρικής” (*topology-shape-metrics; TSM*), η οποία αποτελείται από τρεις φάσεις. Στην πρώτη φάση (*επιπεδοποίηση; planarization*), υπολογίζεται μία “επίπεδη” εμφύτευση ενός δοθέντος (μη) επίπεδου γραφήματος μέγιστου βαθμού τέσσερα, αντικαθιστώντας τις τομές των ακμών με ψευδοκορυφές (*επίπεδη αναπαράσταση; planar representation*). Στην επόμενη φάση (*φάση της ορθογωνιοποίησης; orthogonalization phase*), υπολογίζονται οι γωνίες και τα σημεία καμπής της απεικόνισης, παράγοντας μία *ορθογώνια αναπαράσταση (orthogonal representation)*. Στην τρίτη φάση (*φάση συμπίεσης; compaction phase*), υπολογίζονται οι συντεταγμένες των κορυφών και των ακμών. Ο πυρήνας του δεύτερου βήματος είναι ένας ελαχίστου κόστους αλγόριθμος ροής (min-cost flow algorithm) ο οποίος ελαχιστοποιεί τον αριθμό των σημείων καμπής ενός εμφυτευμένου 4-επίπεδου γραφήματος [CK12]. Σημειώνεται ότι το γενικό πρόβλημα της ελαχιστοποίησης του συνολικού αριθμού των σημείων καμπής ενός 4-επίπεδου γραφήματος (σ.σ. για όλες τις πιθανές επίπεδες εμφυτεύσεις του) είναι  $\mathcal{NP}$ -δύσκολο [GT01].

#### 11.5.4 Πολυεπίπεδες απεικονίσεις

Οι πολυεπίπεδες απεικονίσεις χρησιμοποιούνται συνήθως για την αναπαράσταση ιεραρχικών δεδομένων. Σε μία *πολυεπίπεδη (layered)* ή *ιεραρχική (hierarchical)* απεικόνιση, οι κορυφές ενός γραφήματος απεικονίζονται σε μία σειρά από οριζόντιες γραμμές πλέγματος (*επίπεδα; layers*). Οι σημαντικές κορυφές (dominant vertices) τοποθετούνται στα υψηλότερα επίπεδα, ενώ οι υπόλοιπες κορυφές χαμηλότερα στην ιεραρχία, σταδιακά προς τα κάτω. Η πιο γνωστή προσέγγιση για την παραγωγή πολυεπίπεδων απεικονίσεων γενικών γραφημάτων (όχι κατ’ ανάγκη ακυκλικών) είναι αυτή του Sugiyama [STT81], σύμφωνα με την οποία: (i) οι ακμές θα πρέπει να προσανατολίζονται προς τα κάτω, (ii) οι κορυφές θα πρέπει να είναι ομοιόμορφα κατανεμημένες στα επίπεδα, (iii) οι ακμές μεγάλου μήκους θα πρέπει να αποφεύγονται, (iv) ο αριθμός των τομών μεταξύ των ακμών θα πρέπει να είναι μικρός και (v) ο αριθμός των γεωμετρικών ακμών (δηλ. χωρίς σημεία καμπής) θα πρέπει να μεγί-

στοποποιείται.

Ο αλγόριθμος του Sugiyama αποτελείται από τέσσερα στάδια. Στο πρώτο στάδιο (*αφαίρεση κύκλων; cycle removal*), το γράφημα γίνεται ακυκλικό αντιστρέφοντας την κατεύθυνση ορισμένων ακμών. Ωστόσο, καθώς το αντίστοιχο πρόβλημα είναι  $\mathcal{NP}$ -δύσκολο [Kar72, GJ83], ευρετικοί και προσεγγιστικοί αλγόριθμοι έχουν προταθεί και χρησιμοποιηθεί [GJR85, Rei85, BS90, ELS93]. Στο δεύτερο στάδιο (*ανάθεση επιπέδου; layer assignment*), οι κορυφές τοποθετούνται σε κατάλληλα επίπεδα. Επιπλέον, ψευδοκορυφές τοποθετούνται στις τομές των ακμών (προσκειμένων σε κορυφές μη-διαδοχικών επιπέδων) με τα επίπεδα, οι οποίες τελικά θα αντικατασταθούν με σημεία καμψής. Συνήθεις συναρτήσεις βελτιστοποίησης σε αυτό το βήμα είναι η ελαχιστοποίηση του ύψους [JG72], η ελαχιστοποίηση του πλάτους [BLME02, TNB04] ή η ελαχιστοποίηση του αριθμού των ψευδοκορυφών [GKNV93]. Η *μείωση των τομών (crossing reduction)* είναι το επόμενο βήμα του αλγορίθμου, κατά το οποίο υπολογίζεται μία διάταξη των κορυφών κάθε επιπέδου, έτσι ώστε ο αριθμός των τομών να είναι όσο το δυνατόν μικρότερος. Ευρετικοί αλγόριθμοι [STT81, GKNV93] χρησιμοποιούνται και σε αυτό το βήμα, δεδομένου ότι το αντίστοιχο πρόβλημα απόφασης είναι  $\mathcal{NP}$ -δύσκολο [MNKF90]. Στο τελικό βήμα, υπολογίζονται οι  $x$ -συντεταγμένες των κορυφών έτσι ώστε να μην υπάρχουν επικαλύψεις μεταξύ των κορυφών του ίδιου επιπέδου και ο συνολικός αριθμός των σημείων καμψής να ελαχιστοποιείται [STT81, GKNV93, San96, BK01].

### 11.5.5 Απεικονίσεις ορθογώνιων τομών

Δυστυχώς, στην πράξη πολύ λίγα γραφήματα είναι επίπεδα και οι τομές μεταξύ ακμών είναι αναπόφευκτες στη μεγαλύτερη πλειοψηφία εφαρμογών στις οποίες σχεσιακά δεδομένα οπτικοποιούνται και αναλύονται υιοθετώντας τεχνικές απεικόνισης γραφημάτων. Αυτό σε συνδυασμό με το γεγονός ότι το πρόβλημα της ελαχιστοποίησης των τομών είναι υπολογιστικά δύσκολο [GJ83] εμφανώς οδηγεί στο συμπέρασμα ότι ο υπολογισμός υψηλής ποιότητας απεικονίσεων μη-επίπεδων γραφημάτων είναι εξαιρετικά δύσκολος και νέες προσεγγίσεις θα πρέπει να εισαχθούν ώστε να είναι διαχειρίσιμες οι τομές που δεν μπορούν να εξαλειφθούν από την απεικόνιση ενός γραφήματος.

Πρόσφατα πειράματα με την τεχνική του eye-tracking από τους Huang et al. [Hua07, Hua08, HHE08] δείχνουν ότι οι αρνητικές επιπτώσεις των τομών μεταξύ ακμών τείνουν να εξαλειφθούν στην περίπτωση όπου οι γωνίες που σχηματίζονται από τις τεμνόμενες ακμές είναι μεγάλες. Από αυτό έπεται ότι καλές απεικονίσεις γραφημάτων μπορούν να έχουν τομές μεταξύ ακμών, με την προϋπόθεση ότι οι τεμνόμενες ακμές θα σχηματίζουν μεγάλες γωνίες. Το *πρόβλημα της γωνιακής ευκρίνειας τομών (crossing resolution problem)*, δηλαδή το πρόβλημα της εύρεσης μιας απεικόνισης, στην οποία η ελάχιστη γωνία κατά την οποία δύο ακμές τέμνονται μεγιστοποιείται, γίνεται τόσο σημαντικό για τα μη-επίπεδα γραφήματα,

όσο το πρόβλημα της αποφυγής τομών για τα επίπεδα γραφήματα. Τα αποτελέσματα των Huang et al. έδωσαν το κίνητρο για τη μελέτη μιας νέας κλάσης απεικονίσεων, που ονομάζονται *απεικονίσεις ορθογώνιων τομών* (*right-angle crossing drawings*) ή *RAC-απεικονίσεις* (*RAC-drawings*) για συντομία, στις οποίες κάθε ζεύγος ακμών τέμνεται υπό ορθή γωνία (ως απάντηση στο πρόβλημα της απεικόνισης γραφημάτων με βέλτιστη γωνιακή ευκρίνεια τομών). Οι Didimo, Eades και Liotta [DEL11] ξεκίνησαν τη μελέτη των RAC-απεικονίσεων και από τότε, μεγάλη ερευνητική προσπάθεια έχει αφιερωθεί στη μελέτη προβλημάτων που αφορούν σε RAC-απεικονίσεις [ACD<sup>+</sup>09, DDLM09, DEL10]. Ένα μεγάλο μέρος αυτής της διδακτορικής διατριβής είναι αφιερωμένο στο πρόβλημα της γωνιακής ευκρίνειας τομών και ιδιαίτερα στις RAC-απεικονίσεις, οι οποίες θα περιγραφούν λεπτομερώς στα Κεφάλαια 12-15.

### 11.5.6 Ταυτόχρονες απεικονίσεις

Σε αντίθεση με τα κοινά προβλήματα απεικόνισης γραφημάτων, τα οποία εμπλέκουν ένα μόνο γράφημα, η ταυτόχρονη απεικόνιση γραφημάτων αφορά στο πρόβλημα της απεικόνισης μιας σειράς επίπεδων γραφημάτων (π.χ. ένα μονοπάτι και ένα δέντρο, τρία μονοπάτια, δύο επίπεδα γραφήματα, κλπ) του ίδιου συνόλου κορυφών στο επίπεδο, έτσι ώστε κάθε γράφημα να απεικονίζεται επίπεδο (δηλαδή επιτρέπονται μόνο οι τομές μεταξύ ακμών που προέρχονται από διαφορετικά γραφήματα). Εκτός από το ανεξάρτητο θεωρητικό ενδιαφέρον του, το πρόβλημα αυτό συναντάται σε πολλά πεδία εφαρμογής, όπως η μηχανική λογισμικού, οι βάσεις δεδομένων και τα κοινωνιολογικά δίκτυα, όπου είναι χρήσιμη μία οπτική ανάλυση εξελισσόμενων γραφημάτων, που ορίζονται πάνω στο ίδιο σύνολο κορυφών.

Τα ήδη γνωστά αποτελέσματα για το πρόβλημα της ταυτόχρονης απεικόνισης γραφημάτων είναι συνήθως τριών ειδών: (i) θετικά αποτελέσματα, δηλαδή αλγόριθμοι που παράγουν ταυτόχρονες απεικονίσεις για δοθέντα γραφήματα [BCD<sup>+</sup>07, CvKL<sup>+</sup>11], (ii) αρνητικά αποτελέσματα, δηλαδή αντιπαραδείγματα που αποδεικνύουν ότι δεν είναι εφικτό να παραχθεί μία ταυτόχρονη απεικόνιση για δοθέντα γραφήματα [EK05b, FKK09, GKV09, AGKN10] και (iii) αποτελέσματα για το πρόβλημα του καθορισμού εάν δοθέντα γραφήματα επιδέχονται ταυτόχρονη απεικόνιση [EBG]<sup>+</sup>07]. Για μία πιο πλήρη επισκόπηση, ο αναγνώστης παραπέμπεται σε κλασικά βιβλία απεικόνισης γραφημάτων [BKR07, BKR12]. Οι ταυτόχρονες απεικονίσεις θα επανεξεταστούν στα Κεφάλαια 12, 14, δεδομένου ότι έχουν μελετηθεί ως μέρος αυτής της διδακτορικής διατριβής, σε συνδυασμό με τις RAC-απεικονίσεις. Μία πιο λεπτομερή επισκόπηση των γνωστών αποτελεσμάτων από τη βιβλιογραφία αναβάλλεται για το Κεφάλαιο 12.

### 11.5.7 Ομαδοποιημένες απεικονίσεις

Μεγάλα και πολύπλοκα γραφήματα, όπως γραφήματα του διαδικτύου (world wide web graphs), κοινωνιολογικά δίκτυα ή πολύπλοκα VLSI κυκλώματα είναι δύσκολο να απεικονιστούν χρησιμοποιώντας κλασικές τεχνικές απεικόνισης γραφημάτων. Σε αυτές τις περιπτώσεις, προκειμένου να βελτιωθεί η αναγνωσιμότητα μιας απεικόνισης, είναι αναγκαίο να μειωθεί ο αριθμός των στοιχείων που πρέπει να οπτικοποιηθούν ταυτόχρονα. Η *ομαδοποίηση* (*clustering*) έχει αποδειχθεί ως μία αποτελεσματική προσέγγιση προς αυτή την κατεύθυνση. Πιο συγκεκριμένα με αυτή την τεχνική, οι κορυφές του γραφήματος διαμερίζονται σε ένα συγκεκριμένο αριθμό *ομάδων* (*groups*), με τρόπο τέτοιο ώστε κορυφές με παρόμοια “χαρακτηριστικά” να ανήκουν στην ίδια ομάδα.

Υπάρχοντες αλγόριθμοι ομαδοποίησης συνήθως βελτιστοποιούν συναρτήσεις που αξιολογούν την ποιότητα της ομαδοποίησης, όπως η *κάλυψη*; *coverage*, (δηλ. ο λόγος των ακμών στο εσωτερικό μιας ομάδας προς το συνολικό αριθμό των ακμών), η *εξωτερική ή εσωτερική πυκνότητα*; *external or inter-cluster density* (δηλ. ο λόγος των εσωτερικών ακμών μιας ομάδας προς το μέγιστο δυνατό αριθμό ακμών της ομάδας) και ούτω καθεξής. Ωστόσο, οι περισσότερες μετρικές ομαδοποίησης οδηγούν συνήθως σε αντίστοιχα  $\mathcal{NP}$ -δύσκολα προβλήματα και ως εκ τούτου, έχουν προταθεί ευρετικοί και προσεγγιστικοί αλγόριθμοι [SM91, SM93, Nor93, KMG88, BGW03, Sch05]. Αξίζει να σημειωθεί ότι οι Eades et al. [EFLN99] και Feng et al. [FCE95] έχουν προτείνει αλγόριθμους πολυωνυμικού χρόνου για γεωμετρικές απεικονίσεις ομαδοποιημένων γραφημάτων οι οποίες, ωστόσο, απαιτούν εκθετικό εμβαδόν. Για μία επισκόπηση των τεχνικών ομαδοποίησης, ο αναγνώστης παραπέμπεται στο βιβλίο [JD88] και στις εργασίες [JMF99, Sch07].

### 11.5.8 Δυναμικές απεικονίσεις

Τα *δυναμικά γραφήματα* (*dynamic graphs*) είναι γραφήματα τα οποία εξελίσσονται με την πάροδο του χρόνου (δηλαδή οι ιδιότητες του γραφήματος μεταβάλλονται στο χρόνο, κορυφές και ακμές προστίθενται ή αφαιρούνται, κορυφές γίνονται περισσότερο ή λιγότερο σημαντικές, κλπ.). Τυπικά παραδείγματα είναι τα γραφήματα των υπολογιστών στο διαδίκτυο, το γράφημα του διαδικτύου (world wide web graph), γραφήματα τηλεπικοινωνιακών κλήσεων (phone-call graphs), κοινωνιολογικά δίκτυα, κλπ. Οι απεικονίσεις δυναμικών γραφημάτων μπορούν να θεωρηθούν ως μία ακολουθία στατικών απεικονίσεων. Ωστόσο, σε διάφορες εφαρμογές, ο υπολογισμός μιας νέας απεικόνισης από την αρχή, κάθε φορά που το γράφημα μεταβάλλεται, μπορεί να είναι μία χρονοβόρα και υπολογιστικά απαιτητική διαδικασία. Περισσότερα προβλήματα εμφανίζονται συνήθως στην περίπτωση που είναι επιπλέον αναγκαία η διατήρηση του *νοητού χάρτη* (*mental map*) του χρήστη [ELMS91, MELS95].

Αρκετές τεχνικές έχουν προταθεί για τη διατήρηση του νοητού χάρτη, οι οποίες μπορούν να ταξινομηθούν σε δύο κύριες κατηγορίες. Στην πρώτη κατηγορία, οι αλλαγές περιορίζονται σε συγκεκριμένα μέρη της απεικόνισης και συνήθως αφορούν στην προσθήκη κορυφών και ακμών, διατηρώντας την τρέχουσα τοποθέτηση των κορυφών του γραφήματος αμετάβλητη [BP90, MHT93, Nor96, FöB97, PT98]. Η δεύτερη κατηγορία ορίζεται με βάση διάφορες μετρικές απόστασης [ELMS91, MELS95, BT98, LMR98], ενώ για τη διατήρηση του νοητού χάρτη του χρήστη, η αντίστοιχη απόσταση θα πρέπει να διατηρείται όσο δυνατόν μικρότερη, όταν επέρχεται μία αλλαγή στο γράφημα.

### 11.5.9 Δενδροειδείς απεικονίσεις

Μία *δενδροειδής απεικόνιση* (*treemap*) είναι μία κοινή μέθοδος για την απεικόνιση δέντροδομημένων ιεραρχικών δεδομένων στα οποία κάθε κορυφή έχει ένα πρόγονο (εκτός από τη ρίζα που δεν έχει πρόγονο) και μηδέν ή περισσότερους απογόνους. Σε μία δενδροειδή απεικόνιση, οι κορυφές αναπαρίστανται ως φωλιασμένα ορθογώνια (*nested rectangles*), με τρόπο τέτοιο ώστε το ορθογώνιο κάθε κορυφής να εμπεριέχει στο εσωτερικό του όλους τους απογόνους της. Το εμβαδόν κάθε ορθογωνίου είναι ανάλογο προς ένα συγκεκριμένο χαρακτηριστικό της κορυφής που του αντιστοιχεί. Ο Schneiderman [Shn92] ήταν ο πρώτος που χρησιμοποίησε αυτή την τεχνική για την οπτικοποίηση των αρχείων ενός σκληρού δίσκου. Πρότεινε έναν αναδρομικό αλγόριθμο που διαμέριζε την περιοχή απεικόνισης σε ορθογώνια που εναλλάσσονταν είτε οριζόντια ή κάθετα. Το μέγεθος του κάθε ορθογωνίου ήταν ανάλογο με το μέγεθος του αρχείου/φακέλου που αναπαριστούσε, ενώ διαφορετικά χρώματα ή σκιάσεις χρησιμοποιήθηκαν για τη διάκριση των διαφορετικών περιοχών.

Αρκετές παραλλαγές αυτής της μεθόδου έχουν προταθεί με την πάροδο του χρόνου [TJ92, VWvdW99, BHvW99, Sta00, BSW02]. Ιδιαίτερο ενδιαφέρον παρουσιάζει η εργασία του Stasko [Sta00], ο οποίος παρουσίασε έναν αλγόριθμο για την παραγωγή “κυκλικών” δενδροειδών απεικονίσεων που αναδεικνύουν με σαφήνεια την ιεραρχική δομή του υποκείμενου δέντρου. Οι δενδροειδείς απεικονίσεις μετράνε πολυάριθμες εφαρμογές σε διάφορα πεδία εφαρμογών, όπως η διαχείριση δικτύων [Hei03], η βιοπληροφορική [MJL02], η επιχειρηματική νοημοσύνη (*business intelligence*) [Shn06] και τα μέσα ενημέρωσης [Wes04, OCSZ05, CF07].

### 11.5.10 Άλλες συνήθειες απεικονίσεις

Άλλες συνήθειες τεχνικές της απεικόνισης γραφημάτων που χρησιμοποιούνται ευρέως περιλαμβάνουν τις δεντρικές και τις κυκλικές απεικονίσεις. Η πιο συνηθισμένη τεχνική για την απεικόνιση δέντρων είναι αυτή των Reingold και Tilford [RT81], η οποία παράγει απεικονίσεις που αντικατοπτρίζουν με σαφήνεια την ιεραρχία του υποκείμενου δέντρου. Παρόμοιοι αλγόριθμοι απεικόνισης με επίπεδα (*level-based algorithms*) έχουν παρουσιαστεί στις

εργασίες [Wal90, Blo93, BJL02]. Κυκλικές απεικονίσεις δέντρων [Ber81, Ead92, CPM<sup>+</sup>98, CPP00, BM03, Bac07] έχουν, επίσης, χρησιμοποιηθεί σε διάφορες εφαρμογές. Για μία επισκόπηση, ο αναγνώστης παραπέμπεται στα κλασικά βιβλία απεικόνισης γραφημάτων [DETT99, KW01, Rus07].

Οι κυκλικές απεικονίσεις χρησιμοποιούνται συνήθως σε εφαρμογές στις οποίες η δομή και η τοπολογία του υποκείμενου γραφήματος είναι σημαντικές και πρέπει να αναδειχτούν (τηλεπικοινωνίες, κοινωνιολογικά δίκτυα, κλπ). Σε μία κυκλική απεικόνιση, οι κορυφές του γραφήματος σχηματίζουν ομάδες (clusters) και τοποθετούνται στην περιφέρεια κύκλων, ενώ οι ακμές συνήθως απεικονίζονται ως ευθύγραμμα τμήματα. Αλγόριθμοι για την παραγωγή κυκλικών απεικονίσεων έχουν προταθεί στις εργασίες [TX94, Kre96, DMM96, Bra97, KW02]. Για περισσότερες λεπτομέρειες, βλέπε [ST07].

## 11.6 Απεικόνιση Γραφημάτων με Εφαρμογές στην Ασφάλεια

Τις τελευταίες δεκαετίες, οι υπολογιστές έχουν εξελιχθεί ως ένα καθοριστικό συστατικό της ανθρώπινης ζωής. Στην πραγματικότητα, είναι αδύνατο να φανταστούμε τη ζωή μας χωρίς υπολογιστές, κινητές συσκευές και διαδίκτυο. Καθημερινές δραστηριότητες μπορεί να περιλαμβάνουν τραπεζικές συναλλαγές μέσω διαδικτύου, ηλεκτρονικές αγορές, πληρωμές λογαριασμών και ούτω καθεξής. Σε αυτή την ολοένα και περισσότερο “συνδεδεμένη” κοινωνία, η ανάγκη για ασφάλεια και ιδιωτικότητα έχουν αναδειχθεί σε ένα καίριο ζήτημα, καθώς οι συνδεδεμένοι υπολογιστές γίνονται ολοένα και περισσότερο αντικείμενα κατάχρησης από “εισβολείς” με ποικίλους τρόπους (ιοί, μη-εξουσιοδοτημένη πρόσβαση, κλοπή κωδικών πρόσβασης, κλπ). Στον Dennis Hughes του FBI αποδίδεται συχνά το παρακάτω απόσπασμα: *“Ο μόνος ασφαλής υπολογιστής είναι αυτός που είναι αποσυνδεδεμένος, κλειδωμένος σε ένα χρηματοκιβώτιο και θαμμένος 20 πόδια κάτω από το έδαφος σε μία μυστική τοποθεσία ... χωρίς να είμαι πολύ σίγουρος ακόμα και γι’ αυτό”*, η οποία αντικατοπτρίζει τη δυσκολία της ασφάλειας ενός υπολογιστή (ή ενός δικτύου) από απειλές διαφόρων ειδών.

Τυπικά, η *ασφάλεια υπολογιστών (computer security)* περιλαμβάνει τις τεχνικές και τις πολιτικές που υιοθετούνται από ένα διαχειριστή, ώστε ένας υπολογιστής ή ένα δίκτυο να είναι σε θέση να αποστρέψει και να αμυνθεί σε κάθε είδους απειλή (μη-εξουσιοδοτημένη πρόσβαση, άρνηση εξυπηρέτησης, κλπ). Τυπικά, ο όρος *ασφάλεια υπολογιστών* ορίζεται στο NIST Computer Security Handbook [GR95] ως *“η προστασία που παρέχεται σε ένα αυτοματοποιημένο πληροφοριακό σύστημα, προκειμένου να επιτευχθούν οι εφαρμοστέοι στόχοι για τη διατήρηση της εμπιστευτικότητας, της ακεραιότητας και της διαθεσιμότητας των πόρων του πληροφοριακού συστήματος (και περιλαμβάνει hardware, software, firmware, πληροφορίες/δεδομένα και τηλεπικοινωνίες)”*. Οι τρεις λέξεις-κλειδιά σε αυτόν τον ορισμό είναι η εμπιστευ-



τικότητα, η ακεραιότητα και η διαθεσιμότητα, οι οποίες αναφέρονται ως *E.A.Δ τριάδα* (*C.I.A triad*), πάνω στις οποίες χτίστηκε το πεδίο της ασφάλειας υπολογιστών.

Η *εμπιστευτικότητα* (*confidentiality*) αναφέρεται στον περιορισμό της πρόσβασης σε πληροφορίες και στη γνωστοποίηση τους μόνο σε εξουσιοδοτημένους χρήστες, καθώς και στα υπάρχοντα μέσα για την προστασία της ιδιωτικότητας και ιδιωτικών πληροφοριών. Η *ακεραιότητα* (*integrity*) αναφέρεται στην αξιοπιστία των πόρων της πληροφορίας, δηλαδή τα δεδομένα που έχουν αποσταλεί, παραληφθεί ή αποθηκευτεί είναι πλήρη, ακριβή και δεν έχουν τροποποιηθεί ακατάλληλα ή καταστραφεί, είτε από ατύχημα ή από εσκεμμένη κακόβουλη δραστηριότητα. Η *διαθεσιμότητα* (*availability*) εξασφαλίζει ότι τα πληροφοριακά συστήματα ανταποκρίνονται άμεσα και οι υπηρεσίες δεν αρνούνται σε εξουσιοδοτημένους χρήστες. Για μία πιο πλήρη επισκόπηση της ασφάλειας υπολογιστών, ο αναγνώστης παραπέμπεται στα βιβλία [GR95, Sta99, GT11].

Για την προστασία ενός υπολογιστή ή ενός δικτύου από πιθανές επιθέσεις, στην πράξη οι διαχειριστές χρησιμοποιούν διάφορα εργαλεία άμυνας, όπως τα τείχη προστασίας (*firewalls*), τα συστήματα ανίχνευσης εισβολών (*intrusion detection systems*) και τα αντιϊκά προγράμματα (*anti-virus softwares*). Εν συντομία, τα *τείχη προστασίας* είναι συστήματα που αποσκοπούν στην αποτροπή μη-εξουσιοδοτημένης πρόσβασης από και προς ένα ιδιωτικό δίκτυο, φιλτράροντας την εισερχόμενη και εξερχόμενη κίνηση του δικτύου. Τα *συστήματα ανίχνευσης εισβολών* (*IDS*) είναι συστήματα που παρακολουθούν την κίνηση του δικτύου, ανιχνεύουν “ύποπτη” δραστηριότητα και παράγουν αναφορές και ειδοποιήσεις προς τους διαχειριστές. Τέλος, τα *αντιϊκά προγράμματα* έχουν σχεδιαστεί για την ανίχνευση και τον αφοπλισμό ή την αφαίρεση κακόβουλων προγραμμάτων, όπως ιοί (*viruses*), δούρειοι ίπποι (*trojan horses*), σκουλήκια (*worms*), κλπ.

Ωστόσο, το πρόβλημα της διατήρησης της ασφάλειας ενός υπολογιστή ή ενός δικτύου είναι μία δύσκολη διαδικασία, λόγω της πολυπλοκότητας των πρωτοκόλλων που χρησιμοποιούνται, αλλά και του γεγονότος ότι οι επιτιθέμενοι χρησιμοποιούν εξελιγμένες τεχνικές που συχνά είναι ένα βήμα μπροστά από τα υπάρχοντα εργαλεία άμυνας. Παραδοσιακά, η διαδικασία παρακολούθησης πραγματοποιείται με βάση τα αρχεία καταγραφής (*log-files*) που παράγονται από τα εργαλεία άμυνας, τα οποία χρησιμοποιεί ένας διαχειριστής. Ωστόσο, τα αρχεία καταγραφής είναι σε μορφή κειμένου και συνήθως σε μη-ομοιόμορφη μορφή, καθώς δεν παράγονται από ένα ενιαίο σύστημα και πρέπει να αναλυθούν με ελάχιστη ή καμία αυτοματοποιημένη υποστήριξη. Αυτό καθιστά την παρακολούθηση ως μία χρονοβόρα διαδικασία που δυσχεραίνει τον άμεσα εντοπισμό πιθανών απειλών.

Λόγω της φύσης των δεδομένων που παράγονται από τέτοια εργαλεία (μεγάλου όγκου και πολύπλοκα), οι τεχνικές της απεικόνισης γραφημάτων μπορεί να είναι αποτελεσματικές στην ανάλυση των αρχείων καταγραφής, δεδομένου ότι διευκολύνουν την ανάλυση δε-

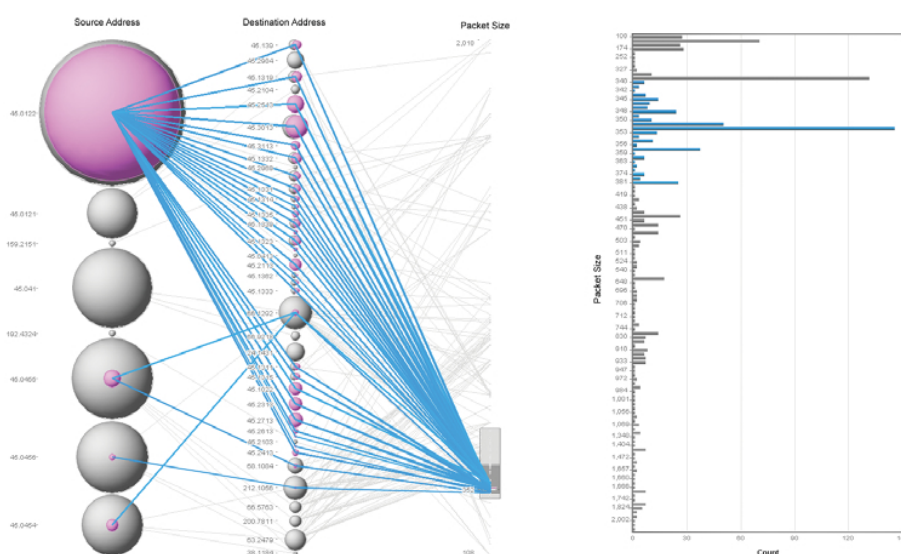
δομένων μεγάλης κλίμακας. Η οπτική αναπαράσταση των πληροφοριών που εσωκλείονται στα αρχεία καταγραφής ενός προσωπικού υπολογιστή ή ενός δικτύου υπολογιστών αναφέρεται ως *οπτικοποίηση δεδομένων ασφάλειας (security visualization)* και αποτελεί ένα γρήγορα εξελισσόμενο πεδίο έρευνας [Con07, Mar08]. Η οπτικοποίηση δεδομένων ασφάλειας στοχεύει στην (i) ανακάλυψη μοτίβων ή τάσεων, (ii) στην ανίχνευση ανώμαλης δραστηριότητας, (iii) στην κατανόηση χαρακτηριστικών μεγάλης ή μικρής κλίμακας και (iv) στη διευκόλυνση της αντίστοιχης λήψης αποφάσεων. Η Εικόνα 11.1 παρουσιάζει ένα πολύ μικρό μέρος ενός αρχείου καταγραφής από το τείχος προστασίας ενός ηλεκτρονικού υπολογιστή του Εθνικού Μετσόβιου Πολυτεχνείου για ένα χρονικό διάστημα δύο ωρών. Προφανώς, η ανάλυση τέτοιων αρχείων καταγραφής αποτελεί “πρόβλημα” για το διαχειριστή ενός δικτύου ιδίως, στην περίπτωση μεγαλύτερων χρονικών διαστημάτων. Το μέγεθος του προβλήματος αυξάνει σημαντικά στην περίπτωση που η ανάλυση πρέπει να πραγματοποιηθεί όχι σε ένα μόνο υπολογιστή, αλλά στο σύνολο του δικτύου του πανεπιστημίου.

Rating	Date / Time	Type	Protocol	Source IP	Destination IP	Direction	Action Taken	Count	Source DHS	Destination DHS	Policy	Rule
Medi...	2009-01-29 14:20:34+2:00	Firewall	TCP (flags:S)	147.102.177.254:3...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:26:16+2:00	Firewall	TCP (flags:S)	147.102.37.1:43450	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:17:34+2:00	Firewall	TCP (flags:S)	147.102.37.1:24066	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:13:22+2:00	Firewall	TCP (flags:S)	147.102.177.254:4...	147.102.205.162:445	Incoming	Blocked	1		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:04:30+2:00	Firewall	TCP (flags:S)	147.102.177.254:3...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 13:00:24+2:00	Firewall	TCP (flags:S)	147.102.37.1:49950	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:41:16+2:00	Firewall	TCP (flags:S)	147.102.177.254:1...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:41:10+2:00	Firewall	TCP (flags:S)	147.102.177.254:1...	147.102.205.162:445	Incoming	Blocked	2		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:30:00+2:00	Firewall	TCP (flags:S)	147.102.177.254:3...	147.102.205.162:445	Incoming	Blocked	1		ZEUS	callisto-nouplo...	ExtPortsIn
Medi...	2009-01-29 12:07:24+2:00	Firewall	TCP (flags:S)	147.102.37.1:60220	147.102.205.162:445	Incoming	Blocked	1		ZEUS	callisto-nouplo...	ExtPortsIn

**Εικόνα 11.1:** Ένα αρχείο καταγραφής που παράγεται από ένα τείχος ασφάλειας για ένα χρονικό διάστημα δύο ωρών.

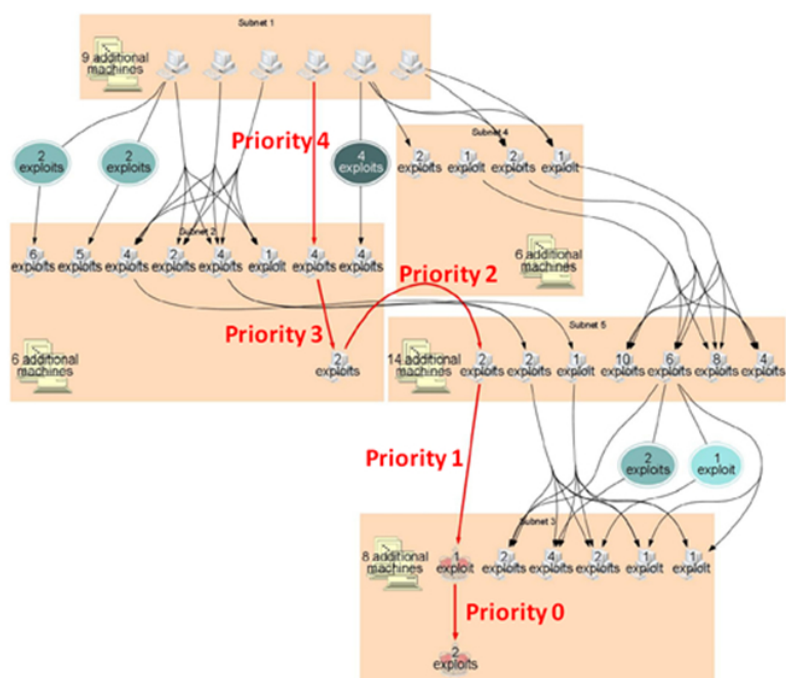
Η Εικόνα 11.2 αποτελεί ένα εξαιρετικό παράδειγμα για τον τρόπο με τον οποίο οι οπτικοποιήσεις μπορούν να βοηθήσουν στον εντοπισμό πιθανών επιθέσεων “κρυμμένων πίσω από τις γραμμές” ενός αρχείου καταγραφής. Η εικόνα αναπαράγεται από το βιβλίο του Marty “Applied Security Visualization” ([Mar08], pp. 264). Η συγκεκριμένη οπτικοποίηση είναι ένα *διάγραμμα παράλληλων συντεταγμένων (parallel coordinates plot)* το οποίο παρουσιάζει τα διαφορετικά πακέτα που μεταδίδονται σε ένα δίκτυο. Οι κόμβοι ταξινομούνται σε τρία κάθετα επίπεδα. Οι κόμβοι του αριστερότερου επιπέδου αντιστοιχούν στις IP-διευθύνσεις των αποστολέων (source IPs). Οι κόμβοι του μεσαίου επιπέδου αντιστοιχούν στις IP-διευθύνσεις των παραληπτών (destination IPs). Οι κόμβοι του δεξιότερου επιπέδου αντιστοιχούν στο μέγεθος των πακέτων που αποστέλλονται κατά τη διάρκεια της επικοινωνίας. Το μέγεθος των κόμβων των δύο πρώτων επιπέδων είναι ανάλογο του βαθμού επικοινωνίας τους με τους υπολογιστές του δικτύου. Από τη συγκεκριμένη οπτικοποίηση είναι εμφανές ότι ο ροζ-χρωματισμένος κόμβος του αριστερότερου επιπέδου επικοινωνεί με σχεδόν όλους τους κόμβους του μεσαίου επιπέδου και στην πραγματικότητα το μέγεθος των πακέτων που απο-

στέλλονται είναι σχεδόν το ίδιο σε όλες τις περιπτώσεις. Ως εκ τούτου, ο διαχειριστής του δικτύου μπορεί εύκολα να αντιληφθεί ότι μία επίθεση από ένα δίκτυο υπολογιστών ρομπότ/ζόμπι (*botnet attack*) βρίσκεται σε εξέλιξη, δηλαδή ένα δίκτυο προσβεβλημένων υπολογιστών (βλ. τους κόμβους του μεσαίου επιπέδου που επικοινωνούν με το ροζ-χρωματισμένο κόμβο του αριστερότερου επιπέδου) ελέγχεται από έναν τρίτο (το ροζ-χρωματισμένο κόμβο) και μπορεί να χρησιμοποιηθεί π.χ. για την αποστολή ανεπιθύμητων μηνυμάτων ηλεκτρονικού ταχυδρομείου (spam e-mails), τη διάδοση κακόβουλου λογισμικού (malware), την επίθεση σε άλλους υπολογιστές ή τη διάπραξη κακόβουλων ενεργειών και απάτης.



**Εικόνα 11.2:** Μία οπτικοποίηση που παρουσιάζει μία επίθεση από ένα δίκτυο υπολογιστών ρομπότ/ζομπι. Αναπαράγεται από το βιβλίο “Applied Security Visualization” ([Mar08], pp. 264).

Μία σημαντική διεργασία στον τομέα της ασφάλειας είναι η ανίχνευση και η αξιολόγηση των τρωτών σημείων ενός δικτύου. Αυτό μπορεί να επιτευχθεί με την παραγωγή *γραφημάτων επίθεσης* (attack graphs). Η Εικόνα 11.3, η οποία αναπαράγεται από την εργασία [NJ07], παρουσιάζει ένα γράφημα επίθεσης του οποίου οι κόμβοι αντιστοιχούν στους υπολογιστές του δικτύου (επισημαίνονται, επίσης, και οι κόμβοι με τρωτά σημεία), ενώ οι ακμές αναπαριστούν τον τρόπο με τον οποίο ένας “εισβολέας” δύναται να αποκτήσει πρόσβαση σε έναν υπολογιστή εκμεταλλευόμενος τις ευπάθειες του δικτύου. Το μονοπάτι που απεικονίζεται με κόκκινο χρώμα, αντιστοιχεί σε ένα από τα μονοπάτια τα οποία ο “εισβολέας” μπορεί να χρησιμοποιήσει για να φτάσει σε ένα συγκεκριμένο υπολογιστή (προορισμός). Προφανώς, τα γραφήματα επίθεσης διευκολύνουν τον εντοπισμό πιθανών τρωτών σημείων ενός δικτύου και δύναται να χρησιμοποιηθούν για την προστασία του.



Εικόνα 11.3: Μία οπτικοποίηση που παρουσιάζει ένα γράφημα επίθεσης [N]07].

Αρκετά πεδία της ασφάλειας υπολογιστών έχουν μελετηθεί μέσω των απεικονίσεων. Μεταξύ αυτών είναι η παρακολούθηση της κίνησης του δικτύου (network traffic monitoring), η ανίχνευση επιθέσεων (intrusion detection), η διάδοση κακόβουλου λογισμικού, ο έλεγχος πρόσβασης, τα γραφήματα επίθεσης, το πρωτόκολλο δρομολόγησης BGP (border gateway protocol) και η ανίχνευση απάτης. Για μία επισκόπηση, ο αναγνώστης παραπέμπεται στην εργασία [ORP07]. Όπως είναι αναμενόμενο, αρκετές κλασικές τεχνικές απεικόνισης γραφημάτων, όπως απεικονίσεις σε επίπεδα, force-directed αλγόριθμοι ή τεχνικές ομαδοποίησης (clustering) έχουν ήδη χρησιμοποιηθεί για το σκοπό αυτό.

### 11.6.1 Εφαρμογές στην παρακολούθηση δικτύου

Ένα θεμελιώδες ζήτημα στον τομέα της ασφάλειας υπολογιστών είναι η παρακολούθηση δικτύου (network monitoring). Αυτή η διαδικασία περιλαμβάνει τόσο την επιθεώρηση της κατάστασης των συσκευών του δικτύου, όσο και την παρακολούθηση της κίνησης του δικτύου. Η κατάσταση των συσκευών μπορεί να ελεγχθεί με τη βοήθεια συστημάτων όπως τα λογισμικά διαχείρισης δικτύου (network management softwares), τα οποία είναι σε θέση να ανιχνεύουν βλάβες ή καθυστερήσεις και αυτόματα να ενημερώνουν τους διαχειριστές. Η παρακολούθηση της κίνησης (traffic monitoring) αναφέρεται στη διαδικασία της συλλογής δικτυακής κίνησης με στόχο τον εντοπισμό ανωμαλιών και "ύποπτης" δραστηριότητας,

όπως οι επιθέσεις άρνησης πρόσβασης (denial of access attacks), οι σαρώσεις θυρών (port scanning), τα σκουλήκια (worms), οι υπερφόρτωσεις (overloads) και ούτω καθεξής. Η συγκεκριμένη διαδικασία πραγματοποιείται κυρίως από συστήματα ανίχνευσης εισβολών. Η παρακολούθηση δικτύου περιλαμβάνει, επίσης, την παρακολούθηση της δραστηριότητας των χρηστών ενός δικτύου με στόχο την ανίχνευση επιθέσεων ή την κατάχρηση που προέρχεται από τους ίδιους τους χρήστες του δικτύου. Ωστόσο, ακόμη αν τα συστήματα ανίχνευσης εισβολών συνεχώς εξελίσσονται ως απάντηση στο συνεχώς αυξανόμενο αριθμό επιθέσεων, η ανίχνευση “ύποπτης” δραστηριότητας καθίστανται ολοένα και πιο δύσκολη, δεδομένου ότι οι επιτιθέμενοι χρησιμοποιούν όλο και περισσότερο εξελιγμένες μεθόδους.

Πολλά από τα υπάρχοντα λογισμικά παρακολούθησης δικτύου είναι εφοδιασμένα με απεικονίσεις στις οποίες εφαρμόζονται τεχνικές απεικόνισης γραφημάτων. Τα λογισμικά αυτά μπορούν να διαχωριστούν σε δύο κατηγορίες. Η πρώτη περιλαμβάνει λογισμικά εφοδιασμένα με απεικονίσεις δεδομένων και ειδοποιήσεων που προέρχονται από συστήματα ανίχνευσης εισβολών (IDS απεικονίσεις, IDS ειδοποιήσεις), ενώ η δεύτερη περιλαμβάνει λογισμικά τα οποία οπτικοποιούν τα αρχεία καταγραφής της δικτυακής κίνησης. Οι πρώτες IDS απεικονίσεις επικεντρώθηκαν σε δεδομένα μικρής κλίμακας και σε έγχρωμες οπτικοποιήσεις της κατάστασης και του επικείμενου κινδύνου [DW01, FTM<sup>+</sup>98, VMF98], με περιορισμένες όμως δυνατότητες και αποτελεσματικότητα. Πιο πρόσφατα, οι Tolle και Niggermann [TN05] παρουσίασαν ένα σύστημα ανίχνευσης εισβολών με απεικονίσεις, το οποίο βασίζεται σε ένα συνδυασμό force-directed αλγορίθμων και αλγορίθμων ομαδοποίησης, που δύναται να οπτικοποιούν την τρέχουσα κατάσταση και να αξιολογούν τον κίνδυνο του δικτύου. Μέσω της παραγόμενης απεικόνισης, μπορούν επιπλέον να αναγνωριστούν και οι ψευδώς θετικές καταστάσεις (ανωμαλίες που δεν αντιστοιχούν σε πραγματικές “εισβολές”; false-positives). Οι Abdullah et al. [ALC<sup>+</sup>05] παρουσίασαν ένα σύστημα που οπτικοποιεί IDS-ειδοποιήσεις σε ένα μεγάλο δίκτυο, παρατηρώντας χρονικά μοτίβα. Τρισδιάστατες απεικονίσεις για την ανίχνευση ανωμαλιών σε ένα δίκτυο έχουν προταθεί στις εργασίες [OR05, XMB<sup>+</sup>06]. Οι Erbacher et al. [ECS05] μελέτησαν τεχνικές απεικόνισης για την ανάλυση των δεδομένων κίνησης του δικτύου, οι οποίες διευκολύνουν την αναγνώριση επιθέσεων σε εξέλιξη πάνω και πέρα από την ικανότητα εντοπισμού και αποκλεισμού των παραδοσιακών τειχών προστασίας.

Στην περιοχή της ανάλυσης της δικτυακής κίνησης, οι Mansmann et al. [MMK08] παρουσίασαν ένα αναλυτικό εργαλείο που οπτικοποιεί τη συμπεριφορά των χρηστών του δικτύου μέσω της αλλαγής της θέσης τους σε ένα δισδιάστατο χώρο, χρησιμοποιώντας ένα force-directed αλγόριθμο. Force-directed τεχνικές έχουν, επίσης, χρησιμοποιηθεί και από τους Girardin και Brodbeck [GB98] για την οπτικοποίηση αρχείων καταγραφής από τείχη προστασίας ή συστήματα ανίχνευσης εισβολών, έτσι ώστε παρόμοιες εγγραφές να οπτικο-

ποιούνται σχετικά κοντά η μία στην άλλη. Οι Muelder et al. [MMB05] μελέτησαν το πρόβλημα του χαρακτηρισμού των σαρώσεων του δικτύου (network scans) χρησιμοποιώντας απεικονίσεις (force-directed αλγορίθμους και αλγορίθμους ομαδοποίησης) και στατιστικές τεχνικές. Στις παραγόμενες απεικονίσεις, κάθε κόμβος αντιστοιχεί σε μία διακριτή σάρωση, ενώ τα βάρη των ακμών αντιστοιχούν στο βαθμό ομοιότητας μεταξύ δύο κόμβων. Οι Blue et al. [BDF<sup>+</sup>08] χρησιμοποίησαν force-directed τεχνικές και δενδροειδείς απεικονίσεις για την οπτικοποίηση της χρήσης του δικτύου σε πραγματικό χρόνο. Παρόμοιες τεχνικές έχουν χρησιμοποιηθεί από τους Di Battista et al. [DSN12], οι οποίες παρουσιάζουν μία κινούμενη απεικόνιση (animation) των αλλαγών των εξυπηρετητών ονομάτων (name servers) μεταξύ στιγμιότυπων του εξυπηρετητή K-root. Οι Yin et al. [YYT<sup>+</sup>04] χρησιμοποίησαν απεικονίσεις γραφημάτων, στις οποίες οι κόμβοι τοποθετούνται σε τρεις παράλληλους άξονες. Οι κόμβοι που δεν ανήκουν στο δίκτυο τοποθετούνται στους δύο εξωτερικούς άξονες, ενώ οι κόμβοι του δικτύου στο μεσαίο άξονα, χρησιμοποιώντας έναν ευρετικό αλγόριθμο που ελαχιστοποιεί τον αριθμό των τομών μεταξύ των ακμών. Μία ακμή μεταξύ δύο κόμβων υποδηλώνει μία επικοινωνία μεταξύ τους. Στη συνέχεια, η αναγνώριση μιας “εισβολής” πραγματοποιείται με χρήση γνωστών προτύπων. Παρόμοιες απεικονίσεις με παράλληλους άξονες έχουν χρησιμοποιηθεί από τους Shabtai et al. [SKSE06] και Lakkaraju et al. [LYL04]. Οι Vandenberghe [Van08] και Goodall et al. [GLRK05] χρησιμοποίησαν ιεραρχικές τεχνικές, ενώ οι Toledo [Tol00] και Fink και North [FN05] χρησιμοποίησαν κυκλικές απεικονίσεις για την παρακολούθηση της κίνησης του δικτύου.

### 11.6.2 Εφαρμογές στα γραφήματα επίθεσης

Η αυτοματοποιημένη αξιολόγηση της ευπάθειας ενός δικτύου έχει αναδειχθεί ως απάντηση στον ολοένα αυξανόμενο αριθμό δικτυακών χρηστών και στο γεγονός ότι, τα δίκτυα είναι βασισμένα σε πολλαπλές πλατφόρμες, τρέχουν διαφορετικά λογισμικά και ως εκ τούτου, είναι πιθανό να ανακύπτουν ζητήματα ευπάθειας και ασφάλειας. Μία κοινή τεχνική που χρησιμοποιείται από τους διαχειριστές, προκειμένου να κατανοήσουν και να αξιολογήσουν τα τρωτά σημεία ενός δικτύου είναι τα γραφήματα επίθεσης, που αναπαριστούν μία σειρά από τρωτά σημεία (και τις σχέσεις μεταξύ τους), τα οποία ένας “εισβολέας” δύναται να χρησιμοποιήσει για να εισβάλει σε ένα δίκτυο.

Τα γραφήματα επίθεσης αρχικά κατασκευάζονταν από τις Red Teams. Ωστόσο, τα γραφήματα παράγονταν με το χέρι, συνήθως περιείχαν λάθη και στην πραγματικότητα, δεν ήταν πρακτικά για δίκτυα μεγάλης κλίμακας. Τα πρώτα βήματα προς την κατεύθυνση της αυτοματοποιημένης παραγωγής γραφημάτων επίθεσης έγιναν στις εργασίες [PS98, RA00, SPEC01, SHJ<sup>+</sup>02], οι οποίες ωστόσο αντιμετώπιζαν προβλήματα λόγω του μεγάλου αριθμού δεδομένων που έπρεπε να οπτικοποιηθούν. Ενώ οι περισσότερες προσπάθειες επικε-

ντρώνονταν κυρίως στην παραγωγή ενός γραφήματος επίθεσης, οι Noel και Jacobs [NJ04] ήταν οι πρώτοι που προσπάθησαν να παράγουν μία οπτικοποίηση, η οποία θα ήταν αναγνώσιμη και κατανοητή, χρησιμοποιώντας πολυεπίπεδες απεικονίσεις. Πολυεπίπεδες απεικονίσεις έχουν, επίσης, χρησιμοποιηθεί από τους Noel et al. [NJK05, NJ07]. Οι Homer et al. [HVOM08] παρουσίασαν μεθοδολογίες που έχουν ως στόχο τον προσδιορισμό τμημάτων ενός γραφήματος επίθεσης, τα οποία μπορούν να παραλειφθούν, εφόσον δεν περιλαμβάνουν χρήσιμες πληροφορίες, καθώς και την ομαδοποίηση παρόμοιων βημάτων επίθεσης σε εικονικούς κόμβους, προκειμένου να προσδιοριστεί το είδος της επίθεσης στην οποία κάθε υπολογιστής του δικτύου είναι εκτεθειμένος. Οι O'Hare et al. [ONP08] παρουσίασαν μία γραφο-θεωρητική προσέγγιση για το πρόβλημα της οπτικοποίησης του κινδύνου του δικτύου, κατασκευάζοντας γραφήματα επίθεσης μέσω της Τοπολογικής Ανάλυσης Ευπάθειας (Topological Vulnerability Analysis; TVA) [JN10]. Η προσέγγιση τους βασίζεται στην εκτεταμένη λειτουργικότητα της yWorks [yWo] που διευκολύνει την παραγωγή διαφορετικών τύπων απεικονίσεων γραφημάτων. Οι Chu et al. [CIL<sup>+</sup>10] παρουσίασαν ένα εργαλείο για την οπτικοποίηση γραφημάτων επίθεσης χρησιμοποιώντας έναν τροποποιημένο αλγόριθμο για την κατασκευή δενδροειδών απεικονίσεων [BSW02].

### 11.6.3 Εφαρμογές στην ανίχνευση απάτης

Σύμφωνα με την έκθεση της Ένωσης Πιστοποιημένων Ελεγκτών κατά της Απάτης (Association of Certified Fraud Examiners; ACFE) προς τα έθνη [AoCFE12], η *απάτη (fraud)* ορίζεται ως κάθε προσπάθεια, η οποία έχει ως στόχο την παραπλάνηση ενός ατόμου με σκοπό την απόκτηση κέρδους. Σε επιχειρηματικό επίπεδο, η απάτη αφορά στην “*εσκεμμένα κακή χρήση ή κατάχρηση των περιουσιακών στοιχείων μιας εταιρείας*”, στην οποία δύναται να εμπλέκονται υπάλληλοι, πελάτες, πωλητές ή άλλες οντότητες [AoCFE12]. Ο ορισμός της απάτης προϋποθέτει την ύπαρξη ενός “*εισβολέα*” (εξωτερικού ή εσωτερικού), ο οποίος αποκτά πρόσβαση σε πληροφορίες και πόρους (πιθανώς μη-εξουσιοδοτημένη) και για το λόγο αυτό, η απάτη μπορεί να θεωρηθεί ως ένας τύπος “*εισβολής*”, ο οποίος εμπίπτει στην ευρύτερη περιοχή της παρακολούθησης δικτύου.

Μεταξύ άλλων τεχνικών, η ανίχνευση της απάτης πραγματοποιείται, επίσης, με τη χρήση συστημάτων διαχείρισης απάτης (fraud management systems; FMS), τα οποία λειτουργούν με τρόπο παρόμοιο με αυτό των συστημάτων ανίχνευσης εισβολών, δηλαδή αναλύουν αρχεία καταγραφής και εντοπίζουν μη-εξουσιοδοτημένη χρήση πληροφοριών και πόρων. Κατά τη διάρκεια των τελευταίων ετών, έχουν προταθεί πολλές προσεγγίσεις για την ανίχνευση απάτης. Οι περισσότερες κάνουν χρήση τεχνικών εξόρυξης δεδομένων (data mining) και αναγνώρισης προτύπων (pattern-matching) [GS95, BH02, KLSH04, PLSMG10]. Πολλά εργαλεία υποστηρίζουν, ωστόσο, οπτικοποιήσεις διαφόρων ειδών, υιοθετώντας διαφορετικές

τεχνικές απεικόνισης γραφημάτων [GS95, KSH<sup>+</sup>98, Sen00, SGS<sup>+</sup>02, HLN09, DDLP10, Lue10, Syn11]. Σε όλα τα συστήματα που αναφέρονται παραπάνω, μπορεί κανείς να εντοπίσει εύκολα πολλούς διαφορετικούς τρόπους απεικόνισης, προκειμένου να παραχθεί η αντίστοιχη οπτικοποίηση δεδομένων συμπεριλαμβανομένων force-directed αλγορίθμων, κυκλικών ή δενδροειδών απεικονίσεων και ούτω καθεξής. Ένα μεγάλο μέρος αυτής της διδακτορικής διατριβής έχει αφιερωθεί στην ανάπτυξη ενός πρωτότυπου συστήματος που στοχεύει στην ανίχνευση εταιρικής απάτης. Για το λόγο αυτό, η ανίχνευση απάτης θα περιγραφεί με λεπτομέρειες στα Κεφάλαια 6-9 της παρούσας διατριβής.

#### 11.6.4 Εφαρμογές σε άλλες περιοχές της ασφάλειας δικτύων

Άλλοι τομείς της ασφάλειας υπολογιστών, στους οποίους χρησιμοποιούνται ευρέως τεχνικές απεικόνισης γραφημάτων, είναι η παρακολούθηση της δραστηριότητας του BGP πρωτοκόλλου (border gateway protocol), ο έλεγχος πρόσβασης (access control) και η διαπραγμάτευση εμπιστοσύνης (trust negotiation). Το πρωτόκολλο BGP είναι ένα πρωτόκολλο, το οποίο χρησιμοποιείται για την ανταλλαγή πληροφοριών δρομολόγησης μεταξύ χρηστών σε δίκτυα αυτόνομων συστημάτων (autonomous systems; ASes). Ένα αυτόνομο σύστημα (AS) είναι ένα τμήμα του διαδικτύου υπό τον έλεγχο μιας ενιαίας διαχειριστικής αρχής. Στην πραγματικότητα, το BGP είναι το πρωτόκολλο που καθορίζει τα βέλτιστα μονοπάτια δρομολόγησης για τα πακέτα που ανταλλάσσονται σε ένα δίκτυο. Λόγω του γεγονότος ότι το BGP υποστηρίζει τη δρομολόγηση μεταξύ αυτόνομων συστημάτων, είναι σημαντικό να ανιχνεύει γρήγορα τυχόν αποτυχίες δρομολόγησης ή επιθέσεις σε ένα αυτόνομο σύστημα που μπορεί να μεταδοθούν και να προκαλέσουν σοβαρή βλάβη στο διαδίκτυο. Οι Di Battista et al. [DBMPP04] οπτικοποιούν ένα τμήμα του BGP γραφήματος δρομολόγησης για ένα συγκεκριμένο πρόθεμα και χρονικό διάστημα, με τη χρήση force-directed τεχνικών. Η αναπαράσταση της εξέλιξης του συγκεκριμένου τμήματος εντός ενός χρονικού διαστήματος πραγματοποιείται με τη μορφή κινούμενων εικόνων (animation). Παρόμοιες τεχνικές σε συνδυασμό με κυκλικές απεικονίσεις έχουν εφαρμοστεί από τους Teoh et al. [TRNC06] με στόχο την ανακάλυψη BGP ανωμαλιών σε πραγματικό χρόνο. Οι Oberheide et al. [OKB06] χρησιμοποίησαν 3D απεικονίσεις για την οπτικοποίηση της BGP τοπολογίας δρομολόγησης του διαδικτύου.

Στην περιοχή του ελέγχου πρόσβασης και της διαπραγμάτευσης εμπιστοσύνης, ο Yee [Yee06] παρουσιάζει ένα μοντέλο που επιτρέπει την αναγνώριση τρωτών σημείων σε ιδιωτικές πληροφορίες (private information vulnerabilities), χρησιμοποιώντας πολυεπίπεδες απεικονίσεις. Οι Montemayor et al. [MFG<sup>+</sup>06] μελέτησαν διάφορες τεχνικές οπτικοποίησης, συμπεριλαμβανομένων πολυεπίπεδων απεικονίσεων, προκειμένου να αποκτήσουν επίγνωση της κατάστασης του ελέγχου πρόσβασης. Πολυεπίπεδες απεικονίσεις έχουν, επίσης, χρησιμοποιηθεί από τους Yao et al. [YSTW05] για την οπτικοποίηση της αυτοματοποιημένης δια-



πραγμάτευσης εμπιστοσύνης (automated trust negotiation). Οι Heitzmann et al. [HPPT08] χρησιμοποίησαν δένδροειδείς απεικονίσεις για την αναπαράσταση των δικαιωμάτων πρόσβασης σε ένα ιεραρχικό σύστημα αρχείων. Οι Anwar και Fong [AF12] ανέπτυξαν ένα εργαλείο απεικόνισης για την υποστήριξη της αξιολόγησης των πολιτικών ελέγχου πρόσβασης σε συστήματα κοινωνιολογικών δικτύων. Οι Mazzia et al. [MLA12] παρουσίασαν ένα εργαλείο που διευκολύνει την κατανόηση των δεδομένων που απορρέουν από το κοινωνιολογικό δίκτυο ενός χρήστη, χρησιμοποιώντας force-directed τεχνικές.

## 11.7 Συμβολή της Διδακτορικής Διατριβής

Η συμβολή της παρούσας διδακτορικής διατριβής εντοπίζεται κυρίως στα παρακάτω προβλήματα:

### **Το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών μιας απεικόνισης:**

*Το πρόβλημα του καθορισμού ενός γραφήματος στο οποίο η ελάχιστη γωνία υπό την οποία τέμνονται δύο ακμές μεγιστοποιείται.* Η διατήρηση μεγάλων γωνιών στα σημεία τομών μιας απεικόνισης είναι θεμελιώδης σε πολλά σενάρια εφαρμογών, καθώς είναι κοινά αποδεκτό (και επιβεβαιωμένο από γνωστικά πειράματα) ότι οι μικρές γωνίες στα σημεία τομής των ακμών δυσχεραίνουν την αναγνωσιμότητα μιας απεικόνισης. Η κατάσταση περιπλέκεται ακόμη περισσότερο στην περίπτωση που ο διαθέσιμος χώρος είναι περιορισμένος όπως στις κινητές συσκευές, όπου είναι αναγκαίο να προβληθεί η απεικόνιση σε μικρές οθόνες, συχνά χαμηλής ανάλυσης. Επομένως, το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών μιας απεικόνισης αποκτά ιδιαίτερο ενδιαφέρον.

Στην παρούσα διδακτορική διατριβή, επικεντρώνουμε κυρίως, σε απεικονίσεις ορθογώνιων τομών (right-angle crossing drawings ή RAC-απεικονίσεις εν συντομία). Αρχικά, αποδεικνύουμε ότι το πρόβλημα του καθορισμού εάν ένα γράφημα επιδέχεται μία γεωμετρική RAC-απεικόνιση είναι  $\mathcal{NP}$ -δύσκολο εισάγοντας μία άπειρη κλάση γραφημάτων με μοναδική RAC συνδυαστική εμφύτευση. Επίσης, εισάγουμε το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης (geometric RAC simultaneous drawing problem), το οποίο συνδυάζει τη μελέτη απεικονίσεων ορθογώνιων τομών και ταυτόχρονων απεικονίσεων. Για το πρόβλημα αυτό, μελετάμε κλάσεις γραφημάτων οι οποίες επιδέχονται τέτοιες απεικονίσεις. Περαιτέρω, μελετάμε ένα σχετικό πρόβλημα, το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης (geometric graph-dual RAC simultaneous drawing problem) και παρουσιάζουμε έναν αλγόριθμο για την κατασκευή μιας τέτοιας απεικόνισης στην περίπτωση που το γράφημα εισόδου είναι ένα εξωεπίπεδο εμφυτευμένο γράφημα. Τέλος, εισάγουμε το πρόβλημα

της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας μιας απεικόνισης ενός γραφήματος, δηλαδή της μεγιστοποίησης του ελαχίστου της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών. Για το πρόβλημα αυτό, παρουσιάζουμε ασυμπτωτικά βέλτιστες ως προς τη συνολική γωνιακή ευκρίνεια απεικονίσεις για ειδικές κλάσεις γραφημάτων (πλήρη, πλήρη διμερή γραφήματα) και ένα force-directed αλγόριθμο για την κατασκευή γραφημάτων υψηλής συνολικής γωνιακής ευκρίνειας.

**Ανίχνευση απάτης:** Περιλαμβάνει τεχνικές και μεθόδους που απαιτούνται για την αναγνώριση πραγματικών ή δυνητικών περιπτώσεων απάτης σε έναν οργανισμό. Η απάτη επηρεάζει πολλές εταιρείες σε όλο τον κόσμο (ανεξάρτητα από το μέγεθός τους) προκαλώντας τους οικονομική ζημιά και θέματα ευθύνης έναντι στους πελάτες τους, αλλά και σε άλλους εμπλεκόμενους φορείς. Το κίνητρό μας για να μελετήσουμε την ανίχνευση απάτης στηρίζεται σε πραγματικά προβλήματα που εγείρονται από τους εσωτερικούς ελεγκτές ενός μεγάλου οργανισμού τηλεπικοινωνιών. Στην παρούσα διδακτορική διατριβή έχουμε εστιάσει κυρίως στην ανίχνευση εταιρικής απάτης (occupational ή employee fraud). Τα σύνολα των δεδομένων που εξετάζουμε προέρχονται κυρίως από συστήματα που περιλαμβάνουν ζεύγη οντοτήτων (υπάλληλος-πελάτης), όπως συστήματα χρέωσης (billing systems), συστήματα ανανέωσης συνδρομών μελών και ούτω καθεξής. Οι εσωτερικοί ελεγκτές μας παρουσίασαν κάποια συχνά εμφανιζόμενα πρότυπα απάτης, πάνω στα οποία στηρίχτηκαν οι προσεγγίσεις που προτείνουμε. Μεταξύ αυτών είναι και η αναγνώριση γεγονότων που αφορούν σε ένα συγκεκριμένο ζεύγος υπαλλήλου-πελάτη, τα οποία όμως εμφανίζονται ανά τακτά χρονικά διαστήματα. Αυτά τα γεγονότα μπορεί να υποκρύπτουν επί παραδείγματι, ότι ένας υπάλληλος παραποιεί σκόπιμα το λογαριασμό ενός πελάτη (διαγράφει/προσθέτει τιμολογήσεις, κλπ).

Προς την κατεύθυνση αυτή σαν μία πρώτη προσέγγιση, προτείνουμε (α) μία μετρική ορισμένη κατάλληλα έτσι ώστε να υπολογίζεται η συχνότητα εμφάνισης των γεγονότων που αφορούν σε ένα συγκεκριμένο ζεύγος και (β) τεχνικές απεικόνισης με βάση ομόκεντρους κύκλους, οι οποίες τοποθετούν κοντά στο κέντρο του συστήματος ζεύγη οντοτήτων που φαίνεται να έχουν περιοδική δραστηριότητα. Σε μία δεύτερη προσέγγιση, περιγράφουμε ένα σύστημα που υποστηρίζει τη διαδικασία ανίχνευσης απάτης με βάση πρότυπα απάτης που προτάθηκαν από τους ελεγκτές και κατάλληλες σπειροειδείς οπτικοποιήσεις τέτοιες ώστε περιοδικά γεγονότα να εμφανίζονται κατά μήκος της ίδιας ακτίνας ή σε κοντινές ακτίνες της σπειροειδούς οπτικοποίησης. Στην τελευταία προσέγγισή μας, περιγράφουμε ένα ολοκληρωμένο σύστημα διαχείρισης απάτης που συνδυάζει τεχνικές αναγνώρισης προτύπων με οπτικοποιήσεις (heat-maps, κυκλικές οπτικοποιήσεις, διαγράμματα παράλληλων συντεταγμένων), με σκοπό τον

εντοπισμό εταιρικής απάτης. Έχουμε επεκτείνει το συγκεκριμένο σύστημα προκειμένου να υποστηρίξει επιπλέον την παρακολούθηση της δραστηριότητας των υπαλλήλων στο δίκτυο της εταιρείας και τον προσδιορισμό τυχόν “ύποπτης” δραστηριότητας.

## 11.8 Οργάνωση της Διδακτορικής Διατριβής

Στο Κεφάλαιο 12, ορίζουμε τα προβλήματα της μεγιστοποίησης της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών μιας απεικόνισης ενός γραφήματος. Παρουσιάζουμε, επίσης, την κλάση των γραφημάτων ορθογώνιων τομών, ως μία κλάση γραφημάτων η οποία επιδέχεται απεικονίσεις βέλτιστης γωνιακής ευκρίνειας τομών και εισάγουμε το πρόβλημα της *γεωμετρικής ταυτόχρονης απεικόνισης ορθογώνιων τομών*, το οποίο συνδυάζει τη μελέτη των απεικονίσεων ορθογώνιων τομών και των ταυτόχρονων απεικονίσεων.

Στο Κεφάλαιο 13, επικεντρωνόμαστε σε γεωμετρικές RAC απεικονίσεις και παρουσιάζουμε μία άπειρη κλάση γραφημάτων με μοναδική RAC συνδυαστική εμφύτευση. Χρησιμοποιούμε μέλη της κλάσης αυτής για να δείξουμε ότι είναι  $\mathcal{NP}$ -δύσκολο να αποφασίσουμε εάν ένα γράφημα επιδέχεται μία γεωμετρική RAC-απεικόνιση. Το Κεφάλαιο 13 αποτελεί προϊόν συνεργασίας με τους Michael Bekos και Antonios Symvonis. Μέρος των αποτελεσμάτων έχει δημοσιευτεί στην εργασία [ABS11], η οποία βελτιώνεται στην εργασία [ABS12].

Στο Κεφάλαιο 14, μελετάμε το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης. Αρχικά αποδεικνύουμε ότι δύο επίπεδα γραφήματα, τα οποία επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση δεν επιδέχονται γεωμετρική RAC ταυτόχρονη απεικόνιση. Επιπλέον, δείχνουμε ότι ένας κύκλος και ένα ταίριασμα πάντα επιδέχονται γεωμετρική RAC ταυτόχρονη απεικόνιση. Μελετάμε, επίσης, το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC απεικόνισης και αποδεικνύουμε ότι είναι πάντα εφικτό να κατασκευαστεί ένα τέτοιο γράφημα, εάν το γράφημα εισόδου είναι ένα εξωεπίπεδο εμφυτευμένο γράφημα. Το Κεφάλαιο 14 αποτελεί προϊόν συνεργασίας με τους Michael Bekos, Michael Kaufmann και Antonios Symvonis. Μέρος των αποτελεσμάτων έχει δημοσιευτεί στην εργασία [ABKS11], η οποία βελτιώνεται στην εργασία [ABKS12].

Στο Κεφάλαιο 5, εισάγουμε το πρόβλημα της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας μιας απεικόνισης, δηλαδή της μεγιστοποίησης του ελαχίστου της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών. Παρουσιάζουμε απεικονίσεις ασυμπτωτικά βέλτιστης συνολικής γωνιακής ευκρίνειας για πλήρη γραφήματα (κυκλικές απεικονίσεις) και πλήρη διμερή γραφήματα (απεικονίσεις 2-επιπέδων). Επιπρόσθετα, παρουσιάζουμε και αξιολογούμε πειραματικά ένα force-directed αλγόριθμο, ο οποίος κατασκευάζει γραφήματα υψηλής συνολικής γωνιακής ευκρίνειας. Το Κεφάλαιο 15 αποτελεί προϊόν συνεργασίας με τους Michael Bekos και Antonios Symvonis. Μέρος των αποτελεσμάτων έχει δημοσιευτεί

στην εργασία [ABS10], η οποία βελτιώνεται στην εργασία [ABS13].

Στο Κεφάλαιο 16, παρουσιάζουμε κάποιες βασικές έννοιες από την περιοχή της ανίχνευσης εταιρικής απάτης. Επιπλέον, παρουσιάζουμε το κίνητρο μας να εργαστούμε στη συγκεκριμένη περιοχή και εισάγουμε την απαραίτητη ορολογία που θα χρησιμοποιηθεί στα Κεφάλαια 17-19.

Στο Κεφάλαιο 17, παρουσιάζουμε τεχνικές οπτικοποίησης που έχουν σκοπό την ανίχνευση περιοδικών μοτίβων σε σειριακά δεδομένα, τα οποία προέρχονται από συστήματα που εμπλέκουν ζεύγη οντοτήτων (υπάλληλος και πελάτης). Οι τεχνικές που παρουσιάζουμε μπορούν να εφαρμοστούν για την ανίχνευση εταιρικής απάτης. Προς αυτή την κατεύθυνση, μετράμε την περιοδικότητα εισάγοντας μία νέα μετρική κατάλληλα ορισμένη για τον υπολογισμό της συχνότητας με την οποία εμφανίζονται γεγονότα που σχετίζονται με ένα συγκεκριμένο ζεύγος οντοτήτων. Η κύρια μέθοδος οπτικοποίησης καταλήγει σε απεικονίσεις που αποτελούνται από ομόκεντρους κύκλους, των οποίων οι ακτίνες αντιστοιχούν στην περιοδικότητα της δραστηριότητας κάθε ζεύγους. Γεγονότα που θεωρούνται ύποπτα διακρίνονται εύκολα, καθώς μετακινούνται προς το κέντρο του συστήματος των κύκλων. Το Κεφάλαιο 17 αποτελεί προϊόν συνεργασίας με τον Antonios Symvonis και έχει δημοσιευτεί στην εργασία [AS12].

Στο Κεφάλαιο 18, παρουσιάζουμε ένα σύστημα που έχει στόχο (α) την ανίχνευση απάτης σε συστήματα που εμπλέκουν ζεύγη οντοτήτων και (β) την ανίχνευση περιοδικής δραστηριότητας. Η κύρια μέθοδος οπτικοποίησης βασίζεται σε ένα σπειροειδές σύστημα στο οποίο τα γεγονότα απεικονίζονται με βάση τη χρονική στιγμή που συνέβησαν. Ύποπτα θεωρούνται τα γεγονότα τα οποία εμφανίζονται κατά μήκος της ίδιας ακτίνας ή σε κοντινές ακτίνες του σπειροειδούς συστήματος. Για την παραγωγή της οπτικοποίησης, το σύστημα ιεραρχεί τις δύο οντότητες με βάση πρότυπα που ορίζουν οι εσωτερικοί ελεγκτές και παράγει ένα βίντεο της δραστηριότητας, στο οποίο γεγονότα με υψηλές ενδείξεις απάτης εμφανίζονται στα πρώτα καρέ του. Το Κεφάλαιο 18 αποτελεί προϊόν συνεργασίας με τους Aikaterini Sotiraki και Antonios Symvonis και έχει δημοσιευτεί στην εργασία [ASS13].

Στο Κεφάλαιο 19 εξελίσσουμε περαιτέρω το σύστημα του Κεφαλαίου 18 ώστε η οπτικοποίηση να μην περιορίζεται μόνο στον εντοπισμό περιοδικών γεγονότων που εμπλέκουν ζεύγη οντοτήτων, αλλά να παρέχει μία ολοκληρωμένη εικόνα όλων των υπό εξέταση προτύπων απάτης, καθώς και των αποτελεσμάτων της εξέτασης σε κάθε πρότυπο. Το σύστημα υποστηρίζει, επίσης, την παρακολούθηση της δραστηριότητας των υπαλλήλων στο δίκτυο της εταιρείας αποσκοπώντας στην ανίχνευση ασυνήθιστης δραστηριότητας, όπως μη εξουσιοδοτημένη πρόσβαση σε εταιρικά συστήματα και λογαριασμούς υπαλλήλων ή πελατών. Το Κεφάλαιο 19 αποτελεί προϊόν συνεργασίας με τους Antonios Symvonis και Vassilis Vassiliou και έχει δημοσιευτεί στην εργασία [ASV14].

Στο Κεφάλαιο 10, παρουσιάζουμε μία περίληψη των αποτελεσμάτων που παρουσιάστη-

καν στην παρούσα διδακτορική διατριβή και μία αναφορά σε μελλοντικές ερευνητικές επεκτάσεις.



## 12 Τα Προβλήματα της Μεγιστοποίησης της Γωνιακής Ευκρίνειας και της Γωνιακής Ευκρίνειας Τομών

Σε αυτό το κεφάλαιο, ορίζουμε τυπικά τα προβλήματα της μεγιστοποίησης της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών. Παρουσιάζεται, επίσης, η κλάση των RAC-γραφημάτων, η οποία περιέχει όλα εκείνα τα γραφήματα που επιδέχονται απεικονίσεις βέλτιστης γωνιακής ευκρίνειας τομών. Τέλος, εισάγεται και ορίζεται τυπικά το πρόβλημα της γεωμετρικής ταυτόχρονης RAC-απεικόνισης (*geometric simultaneous right-angle crossing drawing*), το οποίο συνδυάζει τη μελέτη των RAC-απεικονίσεων και των ταυτόχρονων απεικονίσεων γραφημάτων.

### 12.1 Κίνητρο και Ορισμοί

Όπως ειπώθηκε στο Κεφάλαιο 11, υπάρχουν αρκετά κριτήρια τα οποία έχουν χρησιμοποιηθεί για την αξιολόγηση της ποιότητας μιας απεικόνισης ενός γραφήματος. Από τη σκοπιά του ανθρώπου ως τελικού αποδέκτη της απεικόνισης, είναι απαραίτητη η ανάπτυξη απεικονίσεων εύκολων-στην-ανάγνωση, δηλαδή απεικονίσεων που αναδεικνύουν τα αντικείμενα και τις σχέσεις μεταξύ τους. Από αλγοριθμικής άποψης, η ποιότητα μιας απεικόνισης συνήθως μετράται από κάποια αντικειμενική συνάρτηση και σκοπός είναι η εύρεση μιας απεικόνισης η οποία να ελαχιστοποιεί ή να μεγιστοποιεί την αντικειμενική συνάρτηση. Διάφορες τέτοιες συναρτήσεις, οι οποίες έχουν μελετηθεί εκτενώς από την επιστημονική κοινότητα της Απεικόνισης Γραφημάτων, παρουσιάστηκαν στο Κεφάλαιο 11 μεταξύ των οποίων, ο αριθμός των τομών μεταξύ των ακμών, ο αριθμός των σημείων καμπής των ακμών, το μέγιστο μήκος ακμής, το συνολικό εμβαδόν που καταλαμβάνει η απεικόνιση και ούτω καθεξής.

Μία ιδιότητα η οποία μπορεί να επηρεάσει αρνητικά την αναγνωσιμότητα μιας απεικόνισης είναι η παρουσία ακμών που είναι πάρα πολύ κοντά η μία στην άλλη, ειδικά στην περίπτωση που οι ακμές αυτές προσπίπτουν στην ίδια κορυφή. Διάφορες εφαρμογές, όπως κινητές συσκευές ή οθόνες βίντεο είναι συνήθως εξοπλισμένες με σχετικά μικρές οθόνες χαμηλής ανάλυσης και ως εκ τούτου, αρκετά προβλήματα αναγνωσιμότητας προκύπτουν συχνά λόγω περιορισμών στην ανάλυση. Στο σχεδιασμό δικτύων επικοινωνίας είναι, επίσης, καίριας σημασίας η εύρεση απεικονίσεων με μεγάλες γωνίες, προκειμένου να διευκολυνθεί

η μετάδοση των σημάτων. Τα παραπάνω παραδείγματα έδωσαν κίνητρο για τη μελέτη απεικονίσεων στις οποίες οι γωνίες μεταξύ των ακμών που προσπίπτουν στην ίδια κορυφή είναι όσο το δυνατόν μεγαλύτερες. Τυπικά, η *γωνιακή ευκρίνεια* (*angular resolution*) μιας απεικόνισης ενός γραφήματος ορίζεται ως εξής:

**Ορισμός 12.1.1. (Γωνιακή Ευκρίνεια)** Η γωνιακή ευκρίνεια (*angular resolution*) μιας απεικόνισης ενός γραφήματος ορίζεται ως το μέγεθος της μικρότερης γωνίας που σχηματίζεται από οποιοσδήποτε δύο ακμές προσπίπτουσες στην ίδια κορυφή.

Αναλόγως, ορίζεται η *γωνιακή ευκρίνεια ενός γραφήματος* ως η μέγιστη γωνιακή ευκρίνεια οποιασδήποτε (γεωμετρικής) απεικόνισης του. Ένα προφανές άνω φράγμα της γωνιακής ευκρίνειας ενός γραφήματος μέγιστου βαθμού  $d$  είναι το  $\frac{2\pi}{d}$ . Ωστόσο, το φράγμα αυτό δεν είναι αυστηρό (*tight*) σε αρκετές περιπτώσεις γραφημάτων. Επί παραδείγματι, για το γράφημα  $K_3$  (δηλ. το πλήρες γράφημα τριών κορυφών), το οποίο έχει μέγιστο βαθμό δύο, δεν είναι δύσκολο να διαπιστωθεί ότι κάθε απεικόνιση του έχει μέγιστη γωνιακή ευκρίνεια  $\pi/3$ . Το αντίστοιχο πρόβλημα μεγιστοποίησης αναφέρεται ως *πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας* (*angular resolution maximization problem*) και τυπικά διατυπώνεται ως εξής:

**Πρόβλημα 12.1.1. (Το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας)** Δοθέντος ενός γραφήματος  $G = (V, E)$  και ενός μη-αρνητικού πραγματικού αριθμού  $r \in \mathbb{R}^+$ , υπάρχει απεικόνιση  $\Gamma(G)$  του γραφήματος  $G$  με γωνιακή ευκρίνεια τουλάχιστον  $r$ ? Στην περίπτωση καταφατικής απάντησης, το ενδιαφέρον εντοπίζεται στην εύρεση της απεικόνισης  $\Gamma(G)$ .

Μία ακόμη ανεπιθύμητη ιδιότητα, η οποία μπορεί να επηρεάσει αρνητικά την ποιότητα μιας απεικόνισης ενός γραφήματος είναι οι τομές μεταξύ των ακμών, δηλαδή εάν πάρα πολλές ακμές τέμνονται μεταξύ τους, το ανθρώπινο μάτι δεν μπορεί να διακρίνει τις κορυφές που συνδέονται μεταξύ τους με μία ακμή. Αυτή η διαίσθηση έχει και επιστημονική εγκυρότητα. Πειράματα έχουν δείξει ότι οι επιδόσεις του ανθρώπου σε ασκήσεις εντοπισμού μονοπατιών (*path-tracing tasks*) συσχετίζονται αρνητικά με το πλήθος των τομών μεταξύ ακμών [Pur00, PCA02]. Ως εκ τούτου, εάν ένα γράφημα μπορεί να απεικονισθεί επίπεδο, τότε μία τέτοιου τύπου απεικόνιση είναι προτιμότερη από μία αντίστοιχη με διασταυρώσεις ακμών. Ο υπολογισμός μιας επίπεδης απεικόνισης ενός επίπεδου γραφήματος δεν είναι ιδιαίτερα δύσκολος, ακόμη και στην περίπτωση που οι ακμές πρέπει να απεικονιστούν ως ευθύγραμμα τμήματα. Δυστυχώς, όπως ήδη αναφέρθηκε στο Κεφάλαιο 11, πολύ λίγα γραφήματα είναι επίπεδα στην πράξη και οι διασταυρώσεις μεταξύ ακμών είναι αναπόφευκτες στη συντριπτική πλειοψηφία των πραγματικών εφαρμογών. Αυτό, σε συνδυασμό με την υπολογιστική πολυπλοκότητα του προβλήματος της ελαχιστοποίησης των διασταυρώσεων με-



ταξύ των ακμών ενός γραφήματος, που είναι  $\mathcal{NP}$ -πλήρες στη γενική του περίπτωση [GJ83], προφανώς οδηγεί στο συμπέρασμα ότι ο υπολογισμός υψηλής ποιότητας απεικονίσεων μη-επίπεδων γραφημάτων είναι ιδιαίτερα δύσκολος και ο τρόπος χειρισμού των διασταυρώσεων των γραφημάτων αυτών παραμένει -εν μέρει- χωρίς απάντηση.

Το παραπάνω συμπέρασμα ήταν ένα αρκετά μεγάλο εμπόδιο σε οποιοδήποτε σενάριο απεικόνισης μη-επίπεδων γραφημάτων και κατ'επέκταση, σε μεγάλα και πολύπλοκα γραφήματα. Ωστόσο, πιο πρόσφατα γνωστικά πειράματα (cognitive experiments) σε απεικονίσεις μεγάλων γραφημάτων καταδεικνύουν ότι η αρνητική επίπτωση των τομών μεταξύ των ακμών στην ανθρώπινη ικανότητα κατανόησης μιας απεικόνισης τείνει να εξαλειφθεί στην περίπτωση που οι γωνίες που σχηματίζονται στις διασταυρώσεις είναι μεγάλες. Πιο συγκεκριμένα, οι Huang et al. [Hua07, HHE08, HH10, Hua13] έδειξαν μέσω πειραμάτων εντοπισμού μονοπατιών ότι οι διασταυρώσεις που σχηματίζουν γωνίες μεγαλύτερες των 70 μοιρών έχουν μικρή επίδραση στην ικανότητα του ανθρώπου να ερμηνεύει απεικονίσεις γραφημάτων. Τα αποτελέσματα αυτά έδωσαν το κίνητρο για τη μελέτη του προβλήματος της μεγιστοποίησης της γωνιακής ευκρίνειας τομών, το οποίο είναι ένα σχετικά νέο πρόβλημα απεικόνισης γραφημάτων.

**Ορισμός 12.1.2. (Γωνιακή Ευκρίνεια Τομών)** Η γωνιακή ευκρίνεια τομών (crossing resolution) μιας απεικόνισης ενός γραφήματος ορίζεται ως το μέγεθος της μικρότερης γωνίας που σχηματίζεται από οποιοσδήποτε δύο τεμνόμενες ακμές.

Αναλόγως, ορίζεται η γωνιακή ευκρίνεια τομών ενός γραφήματος ως η μέγιστη γωνιακή ευκρίνεια τομών οποιασδήποτε (γεωμετρικής) απεικόνισης του. Το αντίστοιχο πρόβλημα μεγιστοποίησης αναφέρεται ως πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών (crossing resolution maximization problem) και τυπικά διατυπώνεται ως εξής:

**Πρόβλημα 12.1.2. (Το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών)** Δοθέντος ενός γραφήματος  $G = (V, E)$  και ενός μη-αρνητικού πραγματικού αριθμού  $r \in \mathbb{R}^+$ , υπάρχει απεικόνιση  $\Gamma(G)$  του γραφήματος  $G$  με γωνιακής ευκρίνειας τομών τουλάχιστον  $r$ ? Στην περίπτωση καταφατικής απάντησης, το ενδιαφέρον εντοπίζεται στην εύρεση της απεικόνισης  $\Gamma(G)$ .

Ως απάντηση στο πρόβλημα της απεικόνισης γραφημάτων με βέλτιστη γωνιακή ευκρίνεια τομών, μία νέα κλάση γραφημάτων, που αναφέρονται ως γραφήματα ορθογώνιων τομών (right-angle crossing graphs) ή RAC-γραφήματα για συντομία, εισήχθη και μελετήθηκε αρχικά από τους Didimo, Eades και Liotta [DEL09]. Ο τυπικός ορισμός ενός RAC-γραφήματος έχει ως εξής:

**Ορισμός 12.1.3. (RAC-Γράφημα)** Ένα γράφημα ονομάζεται γράφημα ορθογώνιων τομών (right-angle crossing graph) ή RAC-γράφημα για συντομία, εάν επιδέχεται μία απεικόνιση

στην οποία όλες οι διασταυρώσεις μεταξύ ακμών σχηματίζουν ορθές γωνίες. Αναλόγως, μία απεικόνιση ορθογώνιων τομών (right-angle crossing drawing) ή RAC-απεικόνιση ενός γραφήματος είναι μία πολυγραμμική απεικόνιση στην οποία όλες οι διασταυρώσεις μεταξύ ακμών σχηματίζουν ορθές γωνίες.

Ο συνδυασμός των προβλημάτων της μεγιστοποίησης της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών προτείνει ένα νέο και ενδιαφέρον πρόβλημα στη διαδικασία εύρεσης απεικονίσεων υψηλής συνολικής γωνιακής ευκρίνειας, το οποίο αναφέρεται ως *πρόβλημα μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας (total resolution maximization problem)*. Ο τυπικός ορισμός της συνολικής γωνιακής ευκρίνειας μιας απεικόνισης διατυπώνεται ως εξής:

**Ορισμός 12.1.4. (Συνολική Γωνιακή Ευκρίνεια)** Η συνολική γωνιακή ευκρίνεια (total resolution) μιας απεικόνισης ενός γραφήματος ορίζεται ως το ελάχιστο της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών.

Αναλόγως, ορίζεται η *συνολική γωνιακή ευκρίνεια ενός γραφήματος* ως η μέγιστη συνολική γωνιακή ευκρίνεια οποιασδήποτε (γεωμετρικής) απεικόνισης του. Το αντίστοιχο πρόβλημα μεγιστοποίησης αναφέρεται ως *πρόβλημα της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας (total resolution maximization problem)* και τυπικά διατυπώνεται ως εξής:

**Πρόβλημα 12.1.3. (Το πρόβλημα της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας)** Δοθέντος ενός γραφήματος  $G = (V, E)$  και ενός μη-αρνητικού πραγματικού αριθμού  $r \in \mathbb{R}^+$ , υπάρχει απεικόνιση  $\Gamma(G)$  του γραφήματος  $G$  με συνολική γωνιακή ευκρίνεια τουλάχιστον  $r$ ? Στην περίπτωση καταφατικής απάντησης, το ενδιαφέρον εντοπίζεται στην εύρεση της απεικόνισης  $\Gamma(G)$ .

## 12.2 Βιβλιογραφική Επισκόπηση

Στην ενότητα αυτή, γίνεται επισκόπηση γνωστών αποτελεσμάτων από τη βιβλιογραφία σχετικά με τα προβλήματα της μεγιστοποίησης της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών. Παρουσιάζονται, επίσης, γνωστά αποτελέσματα σχετικά με την κλάση των RAC-γραφημάτων.

### 12.2.1 Γωνιακή Ευκρίνεια

Οι Formann et al. [FHH<sup>+</sup>93] εισήγαγαν την έννοια της γωνιακής ευκρίνειας γεωμετρικών απεικονίσεων γραφημάτων το 1990. Στην εργασία τους, απέδειξαν ότι το πρόβλημα του καθορισμού εάν ένα γράφημα μέγιστου βαθμού  $d$  επιδέχεται απεικόνιση με γωνιακή ευκρίνεια

τουλάχιστον ίση με  $\frac{2\pi}{d}$  (δηλαδή το προφανές άνω φράγμα) είναι  $\mathcal{NP}$ -δύσκολο. Επίσης, έδειξαν ότι ένα γράφημα μέγιστου βαθμού  $d$  επιδέχεται γεωμετρική απεικόνιση γωνιακής ευκρίνειας  $\Omega(\frac{1}{d^2})$ , ενώ εάν το γράφημα είναι επίπεδο μέγιστου βαθμού  $d$ , τότε επιδέχεται γεωμετρική (μη-επίπεδη) απεικόνιση γωνιακής ευκρίνειας  $\Omega(\frac{1}{d})$ . Παρουσίασαν, επίσης, άνω και κάτω φράγματα για τη γωνιακή ευκρίνεια διάφορων τύπων γραφημάτων μέγιστου βαθμού  $d$ , τα οποία είναι ανεξάρτητα από τον αριθμό των κορυφών του γραφήματος.

Οι Malitz και Papakostas [MP94] μελέτησαν το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας επίπεδων γεωμετρικών απεικονίσεων και απέδειξαν ότι για κάθε επίπεδο γράφημα μέγιστου βαθμού  $d$ , είναι δυνατόν να κατασκευαστεί μία επίπεδη γεωμετρική απεικόνιση με γωνιακή ευκρίνεια  $\Omega(\frac{1}{7d})$ . Και πάλι, το φράγμα είναι ανεξάρτητο από τον αριθμό των κορυφών του γραφήματος. Ο Kant [Kan92, Kan96] έδειξε ότι το πρόβλημα του καθορισμού εάν ένα δισυνδεδεμένο επίπεδο γράφημα επιδέχεται γεωμετρική απεικόνιση με γωνιακή ευκρίνεια μεγαλύτερη ή ίση μιας δοθείσας σταθεράς είναι  $\mathcal{NP}$ -δύσκολο. Περαιτέρω, έδειξε ότι ένα τρισυνδεδεμένο επίπεδο γράφημα μέγιστου βαθμού  $d$  επιδέχεται επίπεδη πολυγωνική απεικόνιση με γωνιακή ευκρίνεια  $\Theta(\frac{1}{d})$  σε ένα  $O(n) \times O(n)$  ακέραιο πλέγμα (integer grid), στην οποία κάθε ακμή έχει το πολύ τρία σημεία καμπής.

Οι Garg και Tamassia [GT94] έδειξαν το πρώτο μη-τετριμμένο άνω φράγμα του προβλήματος της μεγιστοποίησης της γωνιακής ευκρίνειας. Ειδικότερα, απέδειξαν ότι υπάρχει μία άπειρη κλάση γραφημάτων μέγιστου βαθμού  $d$ , των οποίων η γωνιακή ευκρίνεια είναι  $O(\sqrt{\log d/d^3})$  σε οποιαδήποτε επίπεδη γεωμετρική απεικόνισή τους. Μελέτησαν, επίσης, τη σχέση ανάμεσα στο εμβαδόν και τη γωνιακή ευκρίνεια σε επίπεδες γεωμετρικές απεικονίσεις και παρουσίασαν γραμμικούς αλγόριθμους για την εύρεση επίπεδων απεικονίσεων υψηλής γωνιακής ευκρίνειας για αρκετές κλάσεις γραφημάτων.

Οι Di Battista και Vismara [DV93] παρουσίασαν ένα μη-γραμμικό σύστημα ισοτήτων και ανισοτήτων που καθορίζει τις γωνίες της απεικόνισης, με σκοπό το χαρακτηρισμό όλων των επίπεδων απεικονίσεων των τριγωνικών (triangular) γραφημάτων. Με τον τρόπο αυτό, έδειξαν ότι σε γραμμικό χρόνο μπορεί να ελεγχθεί εάν μία ανάθεση γωνιών μεταξύ των ακμών ενός επίπεδου τριγωνικού γραφήματος αντιστοιχεί σε μία επίπεδη γεωμετρική απεικόνισή του. Απέδειξαν, επίσης, ότι το πρόβλημα της μεγιστοποίησης της ελάχιστης γωνίας μιας επίπεδης γεωμετρικής απεικόνισης ενός τριγωνικού γραφήματος αντιστοιχεί στην επίλυση ενός μη-γραμμικού προβλήματος βελτιστοποίησης.

Οι Gutwenger και Mutzel [GM99] παρουσίασαν ένα γραμμικό αλγόριθμο που κατασκευάζει επίπεδες πολυγραμμικές απεικονίσεις επίπεδων γραφημάτων με  $n$  κορυφές σε ένα  $(2n - 5) \times (\frac{3}{2}n - \frac{7}{2})$  ακέραιο πλέγμα με το πολύ  $5n - 15$  σημεία καμπής και ελάχιστη γωνιακή ευκρίνεια  $\frac{2}{d}$ . Οι Goodrich και Wagner [GW00] παρουσίασαν έναν αλγόριθμο που απαιτεί ένα σημείο καμπής λιγότερο ανά ακμή, ενώ ασυμπτωτικά διατηρεί αναλλοίωτη τη γωνιακή ευκρί-

νεια, δηλ.,  $\Theta(\frac{1}{d})$ . Ωστόσο, ο αλγόριθμος τους απαιτεί μεγαλύτερο εμβαδόν, καθώς οι τελικές απεικονίσεις χωρούν σε ένα  $(20n - 48) \times (10n - 24)$  ακέραιο πλέγμα. Οι Cheng et al. [CDGK99] βελτίωσαν τον αλγόριθμο που αναφέρθηκε παραπάνω με την παρουσίαση ενός αλγορίθμου που, επίσης, κατασκευάζει απεικονίσεις με ασυμπτωτικά βέλτιστη γωνιακή ευκρίνεια, αλλά απαιτεί το πολύ ένα σημείο καμπής ανά ακμή. Όπως είναι αναμενόμενο, οι απεικονίσεις τους απαιτούν μεγαλύτερη περιοχή σχεδίασης, καθώς χωρούν σε ένα  $30n \times 15n$  ακέραιο πλέγμα. Οι Kobourov και Duncan [DK03] παρουσίασαν δύο αλγορίθμους που βελτιώνουν περαιτέρω τους παραπάνω. Ο πρώτος είναι ένας καρτεσιανής αναπαράστασης αλγόριθμος (cartesian representation algorithm; CRA) που επιτυγχάνει  $\frac{1}{2d}$  γωνιακή ευκρίνεια με ένα σημείο καμπής ανά ακμή σε ένα  $5n \times 5n$  ακέραιο πλέγμα. Ο δεύτερος αλγόριθμος βασίζεται σε πολικές αναπαραστάσεις (polar representations) και εγγυάται  $\frac{\pi}{4d}$  γωνιακή ευκρίνεια πάλι με ένα σημείο καμπής ανά ακμή αλλά σε ένα  $9n \times \frac{9n}{2}$  ακέραιο πλέγμα.

Πιο πρόσφατα, οι Eppstein και Wortman [EW11] για έναν ειδικό τύπο απεικονίσεων, στις οποίες κάθε φραγμένη όψη του γραφήματος απαιτείται να απεικονιστεί ως ένα κεντρικά συμμετρικό πολύγωνο, παρουσίασαν ένα πολυωνυμικού χρόνου αλγόριθμο για την παραγωγή απεικονίσεων βέλτιστης γωνιακής ευκρίνειας.

Αρκετές μέθοδοι έχουν, επίσης, προταθεί για τη βελτιστοποίηση της γωνιακής ευκρίνειας των δεντρικών απεικονίσεων. Ο Eades [Ead92] πρότεινε έναν αλγόριθμο παραγωγής κυκλικών απεικονίσεων κατάλληλο για την απεικόνιση ελεύθερων δένδρων (free trees) στις δύο διαστάσεις (δηλαδή δέντρων που δεν αναδεικνύουν ιεραρχίες και δεν έχουν ρίζα). Γενικά, οι αναδρομικοί αλγόριθμοι παραγωγής κυκλικών απεικονίσεων επιτυγχάνουν υψηλή γωνιακή ευκρίνεια [MH98]. Οι Grivet et al. [GADM06] παρουσίασαν έναν αλγόριθμο για την απεικόνιση δέντρων με ρίζα, ο οποίος είναι, επίσης, αναδρομικός και στον οποίο κάθε υπό-δένδρο ανήκει σε ένα κύκλο διαφορετικό από αυτόν των αδερφών του, αλλά στο εσωτερικό του κύκλου του γονέα του. Οι απεικονίσεις τύπου balloon ακολουθούν παρόμοιες προσεγγίσεις και έχουν ως στόχο τη βελτιστοποίηση της γωνιακής ευκρίνειας, καθώς και του λόγου της μεγαλύτερης σε μήκος ακμής προς την μικρότερη σε μήκος ακμή [LY07].

Οι Brandes και Schlieper [BS06] μελέτησαν επίπεδες απεικονίσεις δένδρων με σταθερές (constant) γωνίες και αποστάσεις μεταξύ γειτονικών κορυφών. Απέδειξαν ότι σε γραμμικό χρόνο μπορεί να ελεγχθεί εάν οι σταθερές αυτές είναι πραγματοποιήσιμες και παρουσίασαν έναν αλγόριθμο ο οποίος ικανοποιεί τις σταθερές και στις δύο περιπτώσεις (δηλαδή τόσο στις γωνίες όσο και στις αποστάσεις) και παράγει πολυγραμμικές απεικονίσεις, στις οποίες η κάθε ακμή είτε έχει το πολύ δύο σημεία καμπής ή απεικονίζεται ως μία συνεχώς διαφορίσιμη καμπύλη (continuously differentiable curve). Οι Didimo et al. [DKL<sup>+</sup>12] μελέτησαν το πρόβλημα της εύρεσης δεντρικών απεικονίσεων σε επίπεδα (leveled tree drawings) μεγιστοποιώντας τη γωνιακή ευκρίνεια ή/και την γωνιακή ευκρίνεια τομών.

Μία ακόμη προσέγγιση του προβλήματος της μεγιστοποίησης της γωνιακής ευκρίνειας είναι οι Lombardi απεικονίσεις, οι οποίες είναι εμπνευσμένες από τα έργα του Αμερικάνου αφηρημένου καλλιτέχνη Mark Lombardi [LH03b]. Ακμές οι οποίες αντιστοιχούν σε κυκλικά τόξα, καθώς και η ομοιόμορφη κατανομή τους γύρω από τις κορυφές τυπικά ορίζουν μία *Lombardi απεικόνιση* ενός γραφήματος [DEG<sup>+</sup>10b, DEG<sup>+</sup>12]. Κατά συνέπεια, οι Lombardi απεικονίσεις είναι βέλτιστης γωνιακής ευκρίνειας εξ'ορισμού. Ωστόσο, δεν επιδέχονται όλα τα γραφήματα Lombardi απεικονίσεις και ο χαρακτηρισμός των Lombardi γραφημάτων παραμένει ακόμη ανοικτό πρόβλημα. Κατά μία έννοια, τα μη-Lombardi γραφήματα μπορούν να οπτικοποιηθούν στο ίδιο στυλ που ορίζουν τα Lombardi, εάν κάποιος χαλαρώσει τον περιορισμό των κυκλικών τόξων στις αναπαραστάσεις των ακμών και πιο συγκεκριμένα, εάν απαιτηθεί κάθε ακμή να αντιστοιχεί σε μία ομαλή (smooth) ακολουθία κυκλικών τόξων. Οι Duncan et al. [DEG<sup>+</sup>11] παρουσίασαν τις *k*-Lombardi απεικονίσεις, στις οποίες κάθε ακμή αποτελείται από μία ομαλή ακολουθία *k* κυκλικών τόξων. Οι Duncan et al. [DEG<sup>+</sup>10a] παρουσίασαν έναν αλγόριθμο, ο οποίος κατασκευάζει επίπεδες Lombardi απεικονίσεις (βέλτιστης γωνιακής ευκρίνειας) για δένδρα χωρίς ρίζα σε πολυωνυμικό χώρο. Ο Matsakis [Mat10] παρουσίασε ένα force-directed αλγόριθμο, ο οποίος οδηγεί σε απεικονίσεις που μοιάζουν στις Lombardi απεικονίσεις. Οι Chernobelski et al. [CCG<sup>+</sup>12] παρουσίασαν δύο force-directed αλγόριθμους, οι οποίοι επίσης οδηγούν σε απεικονίσεις που μοιάζουν στις Lombardi απεικονίσεις, αλλά κάποιες κορυφές ίσως να μην έχουν βέλτιστη γωνιακή ευκρίνεια.

Αρκετοί force-directed αλγόριθμοι έχουν προταθεί για το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας. Οι Brandes και Wagner [BW98] παρουσίασαν ένα force-directed αλγόριθμο για την οπτικοποίηση των διασυνδέσεων στον τομέα των σιδηροδρομικών συστημάτων, ο οποίος οδηγεί σε απεικονίσεις υψηλής γωνιακής ευκρίνειας, υποθέτοντας ότι οι θέσεις των κορυφών στο επίπεδο είναι αμετάβλητες, ενώ οι ακμές αντιστοιχούν σε Bezier καμπύλες. Η εργασία του συνεχίστηκε από τους Finkel και Tamassia [FT04], οι οποίοι χρησιμοποίησαν καμπύλες γραμμικής καμπυλότητας για την αναπαράσταση των ακμών στο αλγόριθμο τους, αίροντας επίσης τον περιορισμό ότι οι κορυφές είναι σε αμετάβλητες θέσεις. Οι Brandes et al. [BST00] απεικόνισαν μεγάλα δίκτυα μεταφορών χρησιμοποιώντας καμπύλες κυβικής καμπυλότητας. Οι Lin και Yen [LY05] παρουσίασαν ένα force-directed αλγόριθμο, ο οποίος οδηγεί σε αναπαραστάσεις γραφημάτων υψηλής γωνιακής ευκρίνειας. Στην εργασία τους, ζεύγη ακμών που προσπίπτουν στην ίδια κορυφή μοντελοποιούνται ως φορτισμένα ελατήρια, τα οποία απωθούνται μεταξύ τους και με τον τρόπο αυτό εγγυώνται την αποφυγή επικαλύψεων μεταξύ ακμών που προσπίπτουν στην ίδια κορυφή.

Οι Eppstein et al. [ELMN13] μελέτησαν το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας στις τρεις διαστάσεις. Απέδειξαν ότι κάθε γράφημα μέγιστου βαθμού τρία μπορεί να απεικονιστεί χωρίς τομές στις τρεις διαστάσεις, αν κάθε ακμή έχει το πολύ δύο σημεία κα-

μής και όλες οι γωνίες μεταξύ τμημάτων-ακμών που προσπίπτουν στην ίδια κορυφή ή στο ίδιο σημείο καμπής είναι ίσες με 120 μοίρες. Έδειξαν, επίσης, πως ένα γράφημα μέγιστου βαθμού τέσσερα μπορεί να απεικονιστεί στις τρεις διαστάσεις σε κυβικό χώρο με βέλτιστη γωνιακή ευκρίνεια, υποθέτοντας ότι κάθε ακμή έχει τρία σημεία καμπής και οι κορυφές έχουν ακέραιες συντεταγμένες. Οι Garg et al. [GTV96] μελέτησαν τρισδιάστατες γεωμετρικές απεικονίσεις υποθέτοντας αρκετά κριτήρια ανάλυσης, χωρίς ωστόσο να απαιτούν οι κορυφές να ανήκουν σε μία δισδιάστατη επιφάνεια. Οι Goodrich και Pszona [GP13] μελέτησαν διαγράμματα τόξου (arc diagrams) και πρότειναν αλγόριθμους που επιτυγχάνουν καλή γωνιακή ευκρίνεια στις τρεις διαστάσεις, ακόμα και αν τα τόξα πρέπει να προβληθούν σε μία δοθείσα 2D ευθεία γραμμή της απεικόνισης του γραφήματος εισόδου.

### 12.2.2 Γωνιακή Ευκρίνεια τομών και RAC-γραφήματα

Η πρώτη εργασία για το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών δημοσιεύθηκε το 2009 από τους Didimo et al. [DEL09, DEL11], οι οποίοι συνδυαστικά μελέτησαν την ιδανική περίπτωση στην οποία οι γωνίες που σχηματίζονται στις διασταυρώσεις μεταξύ των ακμών είναι ακριβώς ενενήντα μοίρες. Στην εργασία τους, απέδειξαν ότι η κλάση των γεωμετρικών RAC απεικονίσεων είναι υποκλάση των γεωμετρικών *quasi-planar απεικονίσεων*, στις οποίες δεν επιτρέπεται τρεις ακμές να τέμνονται ανά δύο μεταξύ τους και περαιτέρω μελέτησαν τη σχέση μεταξύ του πλήθους των κορυφών του γραφήματος και της *πολυπλοκότητας καμπύλης* (*curve complexity*, δηλαδή του μέγιστου αριθμού σημείων καμπής ανά ακμή) σε RAC απεικονίσεις. Απέδειξαν ότι κάθε γράφημα με  $n$  κορυφές, το οποίο επιδέχεται μία γεωμετρική RAC-απεικόνιση (δηλαδή μηδενικής πολυπλοκότητας καμπύλης), έχει το πολύ  $4n - 10$  ακμές και ότι το φράγμα αυτό είναι αυστηρό, δηλαδή υπάρχουν απείρως πολλές τιμές του  $n$  για τις οποίες υπάρχει μία γεωμετρική RAC απεικόνιση ενός γραφήματος με  $n$  κορυφές και ακριβώς  $4n - 10$  ακμές. Κατά συνέπεια, ο περιορισμός των γεωμετρικών απεικονίσεων θα πρέπει να αρθεί, αν είναι επιθυμητή η απεικόνιση πυκνών, μη-επίπεδων γραφημάτων με βέλτιστη γωνιακή ευκρίνεια τομών. Πράγματι, είναι πάντοτε δυνατόν να κατασκευαστεί μία RAC-απεικόνιση ενός γραφήματος, εάν κάθε ακμή έχει πολυπλοκότητα καμπύλης τρία, ενώ εάν μία RAC-απεικόνιση ενός γραφήματος με  $n$  κορυφές έχει πολυπλοκότητα καμπύλης ένα ή δύο, τότε ο αριθμός των ακμών του γραφήματος πρέπει να είναι  $O(n^{7/4})$  και  $O(n^{4/3})$ , αντίστοιχα [DEL09, DEL11]. Τα τελευταία δύο φράγματα βελτιώθηκαν από τους Ariskushi et al. [AFK<sup>+</sup>10], οι οποίοι έδειξαν ότι εάν ένα γράφημα με  $n$  κορυφές επιδέχεται μία RAC-απεικόνιση με το πολύ ένα (δύο, αντίστοιχα) σημεία καμπής ανά ακμή, τότε το γράφημα αυτό έχει το πολύ  $6.5n - 13$  ( $74.2n$ , αντίστοιχα) ακμές. Αντίστοιχα, άνω και κάτω φράγματα στη γωνιακή ευκρίνεια τομών γεωμετρικών απεικονίσεων πλήρων γραφημάτων έχουν παρουσιαστεί από τους Di Giacomo et al. [DDE<sup>+</sup>12].

Οι Angelini et al. [ACD<sup>+</sup>09] μελέτησαν τη σχέση μεταξύ της βέλτιστης γωνιακής ευκρίνειας τομών και άλλων κλασσικών αισθητικών κριτηρίων απεικόνισης γραφημάτων, συμπεριλαμβανομένων του εμβαδού και του προσανατολισμού των ακμών σε απεικονίσεις κατευθυνόμενων ακυκλικών γραφημάτων. Απέδειξαν ότι το πρόβλημα του καθορισμού εάν ένα ακυκλικό, κατευθυνόμενο επίπεδο γράφημα επιδέχεται μία γεωμετρική ανερχόμενη RAC-απεικόνιση είναι  $\mathcal{NP}$ -δύσκολο και περαιτέρω, κατασκεύασαν κατευθυνόμενα γραφήματα τα οποία επιδέχονται γεωμετρικές ανερχόμενες RAC-απεικονίσεις, αλλά σε εκθετικό χώρο. Οι Didimo et al. [DEL10] έδειξαν ότι το πρόβλημα του καθορισμού εάν ένα διμερές γράφημα επιδέχεται γεωμετρική RAC-απεικόνιση μπορεί να απαντηθεί σε γραμμικό χρόνο και περαιτέρω, παρουσίασαν έναν αλγόριθμο της τάξης του  $O(n^2 \log n)$ , ο οποίος υπολογίζει την αντίστοιχη απεικόνιση και παράλληλα ελαχιστοποιεί το συνολικό αριθμό των διασταυρώσεων. Απέδειξαν, επίσης, ότι το πρόβλημα της εύρεσης του μέγιστου υπό-γραφήματος ενός μη-RAC διμερούς γραφήματος, το οποίο επιδέχεται γεωμετρικής RAC-απεικόνισης, είναι  $\mathcal{NP}$ -δύσκολο.

Οι Eades και Liotta [EL13] εισήγαγαν και μελέτησαν ιδιότητες μιας υπό-κλάσης των RAC-γραφημάτων, η οποία περιέχει *μεγιστικώς πυκνά RAC γραφήματα* (*maximally dense RAC graphs*), δηλαδή RAC-γραφήματα με  $n$  κορυφές και ακριβώς  $4n - 10$  ακμές. Απέδειξαν ότι ένα μεγιστικώς πυκνό RAC γράφημα είναι 1-επίπεδο (*1-planar*), δηλαδή επιδέχεται απεικόνιση στην οποία κάθε ακμή εμπλέκεται σε -το πολύ- μία διασταύρωση. Επίσης, έδειξαν ότι για κάθε ακέραιο αριθμό  $i \geq 0$ , υπάρχει ένα 1-επίπεδο γράφημα με  $n = 8 + 4i$  κορυφές και  $4n - 10$  ακμές, το οποίο ωστόσο δεν είναι RAC-γράφημα. Αφετέρου, οι Dehkordi και Eades [DE12] έδειξαν ότι κάθε εξωτερικώς 1-επιπεδικό γράφημα (*outer 1-plane graph*), δηλαδή ένα τοπολογικό γράφημα στο οποίο κάθε κορυφή εμφανίζεται στη μη-φραγμένη όψη του και κάθε ακμή εμπλέκεται σε -το πολύ- μία διασταύρωση, επιδέχεται RAC-απεικόνιση με την ίδια εμφύτευση. Τα αποτελέσματα που περιγράφηκαν παραπάνω οδηγούν στο συμπέρασμα ότι τα RAC-γραφήματα βρίσκονται μεταξύ των εξωτερικώς 1-επιπεδικών γραφημάτων και των 1-επίπεδων γραφημάτων.

Ο Van Kreveld [vK10] έδειξε ότι η ποιότητα μιας επίπεδης απεικόνισης ενός επίπεδου γραφήματος, μετρούμενη σε όρους απαιτούμενου εμβαδού, μήκους-ακμών και γωνιακής ευκρίνειας, μπορεί να βελτιωθεί εάν επιτραπούν ορθογώνιες διασταυρώσεις ακμών. Οι Fink et al. [FHM<sup>+</sup>11] μελέτησαν το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών στην ειδική περίπτωση που οι κορυφές του γραφήματος βρίσκονται σε σταθερές - προκαθορισμένες θέσεις. Πιο συγκεκριμένα, επιτρέποντας το σχηματισμό μεγάλων γωνιών στις διασταυρώσεις, αξιολόγησαν την αποδοτικότητα των μεθόδων τους σε όρους υπολογιστικής πολυπλοκότητας και πολυπλοκότητας καμπύλης ως προς τις αντίστοιχες επίπεδες απεικονίσεις. Οι Nguyen et al. [NEHH10] παρουσίασαν μία μέθοδο τετραγωνικού προγραμματι-

σμού, η οποία κατασκευάζει κυκλικές απεικονίσεις υψηλής γωνιακής ευκρίνειας τομών. Οι Dehkordi et al. [RDNEH13] πρότειναν ένα χαρακτηρισμό των κυκλικών RAC-γραφημάτων, καθώς και έναν αλγόριθμο ο οποίος κατασκευάζει κυκλικές RAC-απεικονίσεις σε γραμμικό χρόνο. Εισάγουν, επίσης, μία μέθοδο τετραγωνικού προγραμματισμού για το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών μιας απεικόνισης.

Οι Didimo et al. [DLR10] εισήγαγαν ένα πλαίσιο (framework), το οποίο συνδυάζει μεθόδους επιπεδοποίησης (planarization methods) και force-directed αλγόριθμους για την παραγωγή πολυγραμμικών απεικονίσεων υψηλής γωνιακής ευκρίνειας τομών μη-επίπεδων γραφημάτων με σχετικά μικρό αριθμό διασταυρώσεων και καλή γεωδεδτική τάση ακμών<sup>1</sup>. Για την παραγωγή πολυγραμμικών απεικονίσεων, η γωνιακή ευκρίνεια τομών και η γεωδεδτική τάση ακμών είναι τα βασικά κριτήρια βελτιστοποίησης ενός force-directed αλγορίθμου των Didimo et al. [DLR11], ο οποίος υπολογίζει ταυτόχρονες εμφυτεύσεις δύο γραφημάτων στο επίπεδο, με σκοπό την ανάλυση της επισκεψιμότητας ιστοσελίδων με τεχνικές απεικόνισης γραφημάτων. Οι Huang et al. [HEHL10] μελέτησαν την αποδοτικότητα των συμβιβασμών μεταξύ διαφορετικών αισθητικών κριτηρίων και παρουσίασαν ένα force-directed αλγόριθμο, ο οποίος τείνει να μεγιστοποιεί την γωνιακή ευκρίνεια τομών μιας απεικόνισης.

Problem	Straight-line	1 bend/edge	2 bends/edge	3 bends/edge
RAC	$\leq 4n - 10$ [DEL09]	$\leq 6.5 - 13$ [AFK+10]	$\leq 74n$ [AFK+10]	No restrict [DEL09]
aLAC	$\leq (3n - 6)\pi/a$ [DGMW10]	No restrict [DDL09]	No restrict [DDL09]	No restrict [DDL09]
aEAC	$\leq 3(3n - 6)$ [AFT10]	$\leq 27n$ [AFT10]	$\leq 477n$ [AFT10]	No restrict [AFT10]

**Εικόνα 12.1:** Γνωστά αποτελέσματα για τα RAC και aLAC/aEAC προβλήματα απεικόνισης.

Οι Di Giacomo et al. [DDL09, DDE<sup>+</sup>12], Dujmovic et al. [DGMW10], και Ackerman et al. [AFT10] μελέτησαν το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών χαλαρώνοντας τον περιορισμό της βέλτιστης γωνιακής ευκρίνειας τομών. Δοθέντος ενός θετικού πραγματικού αριθμού  $a$ , τέτοιο ώστε  $0 < a < \pi/2$ , μία απεικόνιση ονομάζεται aLAC (aEAC, αντίστοιχα) εάν η ελάχιστη γωνία η οποία σχηματίζεται από δύο οποιοσδήποτε τεμνόμενες ακμές του είναι τουλάχιστον (ακριβώς, αντίστοιχα)  $a$ , ενώ ένα γράφημα το οποίο επιδέχεται μία aLAC (aEAC, αντίστοιχα) απεικόνιση ονομάζεται aLAC (aEAC, αντίστοιχα) γράφημα. Οι Dujmovic et al. [DGMW10] απέδειξαν ότι ένα γράφημα με  $n$  κορυφές, το οποίο επιδέχεται γεωμετρικής aLAC απεικόνισης, έχει το πολύ  $(3n - 6)\pi/a$  ακμές. Οι Di Giacomo et al.

<sup>1</sup>Για μία ακμή  $e = (u, v)$  με σημεία καμψής, ο όρος γεωδεδτική τάση ακμών (geodesic edge tendency) αναφέρεται στη μέγιστη απόσταση ενός σημείου της ακμής  $e$  από το ευθύγραμμο τμήμα που ενώνει τα άκρα  $u$  και  $v$  της ακμής  $e$ .



[DDL09, DDE<sup>+</sup>12] απέδειξαν ότι είναι πάντοτε εφικτή η κατασκευή μιας aLAC απεικόνισης δοθέντος ενός γραφήματος με πολυπλοκότητα καμπύλης ένα. Οι Ackerman et al. [AFT10] απέδειξαν ότι ένα γράφημα με  $n$  κορυφές, το οποίο επιδέχεται μιας aEAC απεικόνισης με πολυπλοκότητα καμπύλης ένα (δύο, αντίστοιχα), έχει το πολύ  $27n$  ( $477n$ , αντίστοιχα) ακμές. Οι Angelini et al. [ABD<sup>+</sup>12] παρουσίασαν τεχνικές για την παραγωγή πολυγραμμικών απεικονίσεων επίπεδων γραφημάτων υψηλής γωνιακής ευκρίνειας τομών σε υπό-τετραγωνικό χώρο. Στην Εικόνα 12.1 συνοψίζονται όλα τα φράγματα που παρουσιάστηκαν παραπάνω. Τα γκρι-σκιασμένα κελιά αντιστοιχούν σε φράγματα τα οποία δεν είναι γνωστό εάν είναι αυστηρά. Για μία πιο πλήρη βιβλιογραφική επισκόπηση του προβλήματος της μεγιστοποίησης της γωνιακής ευκρίνειας τομών, ο αναγνώστης παραπέμπεται στην εργασία [DL13].

## 12.3 Εισαγωγικές Έννοιες και Ορολογία

Στην ενότητα αυτή, παρουσιάζονται ορισμένες γνωστές ιδιότητες των RAC απεικονίσεων, οι οποίες θα χρησιμοποιηθούν αρκετά στο υπόλοιπο αυτής της διατριβής.

**RAC-Ιδιότητα 1** (Didimo, Eades και Liotta [DEL11]). *Σε μία γεωμετρική RAC-απεικόνιση, δεν υπάρχουν τρεις ακμές, οι οποίες ανά δύο να τέμνονται μεταξύ τους.*

**RAC-Ιδιότητα 2** (Angelini et al. [ACD<sup>+</sup>09]). *Λέμε ότι ένα ζεύγος μη-επικαλυπτόμενων ακμών  $(u, v)$  και  $(u, v)$  προσπίπτουσες στην ίδια κορυφή σχηματίζουν μία έλικα (fan) με κέντρο την κορυφή  $u$ . Σε μία γεωμετρική RAC-απεικόνιση, μία ακμή δεν μπορεί να διασταυρωθεί με μία έλικα.*

**RAC-Ιδιότητα 3** (Didimo, Eades και Liotta [DEL11]). *Σε μία γεωμετρική RAC-απεικόνιση, δεν υπάρχει τρίγωνο  $T$  σχηματιζόμενο από ακμές του γραφήματος και δύο ακμές  $(a, b)$  και  $(a, b')$ , έτσι ώστε η κορυφή  $a$  να βρίσκεται εξωτερικά του  $T$ , ενώ οι κορυφές  $b$  και  $b'$  στο εσωτερικό του.*

**RAC-Ιδιότητα 4** (Eades και Liotta [EL11]). *Η μη-φραγμένη όψη κάθε RAC-απεικόνισης ενός μεγιστικώς πυκνού RAC-γραφήματος φράσσεται από τρεις ακμές του.*

## 12.4 Ταυτόχρονες RAC-απεικονίσεις

Στην ενότητα αυτή, παρουσιάζεται ένα πρόβλημα, το οποίο συνδυάζει τη μελέτη των RAC-απεικονίσεων και των ταυτόχρονων απεικονίσεων γραφημάτων. Εισάγεται, επίσης, η αντίστοιχη ορολογία και γίνεται η απαραίτητη βιβλιογραφική επισκόπηση της περιοχής.

### 12.4.1 Κίνητρο και Ορισμοί

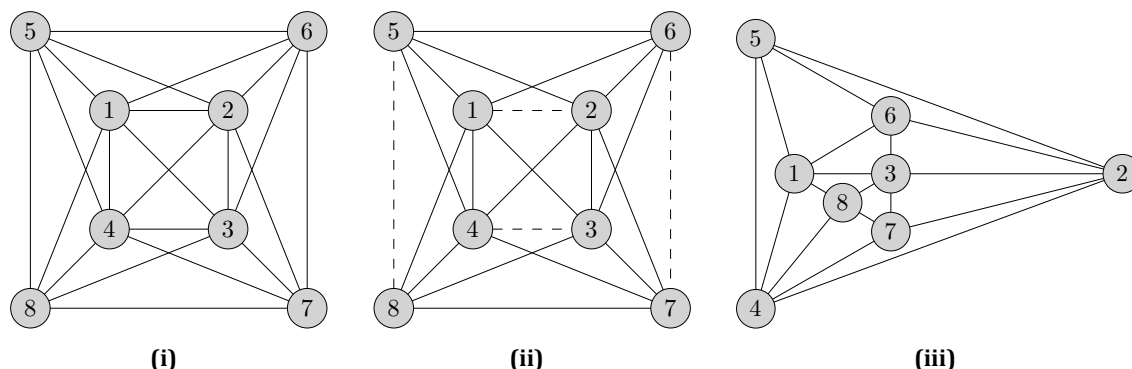
Όπως ειπώθηκε στο Κεφάλαιο 11, οι ταυτόχρονες απεικονίσεις γραφημάτων αφορούν στην απεικόνιση δύο (ή περισσότερων) επίπεδων γραφημάτων του ίδιου συνόλου κορυφών, έτσι ώστε καθένα γράφημα να απεικονίζεται επίπεδο (δηλαδή επιτρέπονται μονάχα διασταυρώσεις μεταξύ ακμών διαφορετικών γραφημάτων). Η γεωμετρική (*geometric*) εκδοχή του προβλήματος περιορίζει το πρόβλημα σε γεωμετρικές απεικονίσεις. Το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων (*geometric RAC simultaneous drawing problem* ή GRacSim πρόβλημα απεικόνισης γραφημάτων) αποτελεί συνδυασμό των προβλημάτων της εύρεσης RAC απεικονίσεων και ταυτόχρονων απεικονίσεων και τυπικά ορίζεται ως εξής:

**Πρόβλημα 12.4.1. (Το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων)** Έστω  $G_1 = (V, E_1)$  και  $G_2 = (V, E_2)$  δύο επίπεδα γραφήματα του ίδιου συνόλου κορυφών. Το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων αφορά στον καθορισμό των συντεταγμένων των κορυφών των γραφημάτων  $G_1$  και  $G_2$  στο επίπεδο, έτσι ώστε όταν οι ακμές τους απεικονίζονται ως ευθύγραμμα-τμήματα, (i) κάθε γράφημα απεικονίζεται επίπεδο, (ii) δεν υπάρχουν επικαλύψεις ακμών και (iii) οι διασταυρώσεις μεταξύ ακμών των  $E_1$  και  $E_2$  σχηματίζουν ορθές γωνίες.

Το GRacSim πρόβλημα απεικόνισης γραφημάτων παρουσιάζει ενδιαφέρον, καθώς αποτελεί συνδυασμό δύο ενεργών ερευνητικών περιοχών της απεικόνισης γραφημάτων. Το δικό μας κίνητρο για τη μελέτη του προβλήματος αυτού στηρίζεται στην εργασία των Didimo et al. [DEL11], οι οποίοι απέδειξαν ότι το γράφημα-τομών (crossing graph) μιας γεωμετρικής RAC-απεικόνισης είναι διμερές<sup>2</sup>. Συνεπώς, οι ακμές μιας γεωμετρικής RAC απεικόνισης ενός γραφήματος  $G = (V, E)$  μπορούν να διαχωριστούν σε δύο σύνολα  $E_1$  και  $E_2$ , έτσι ώστε οι ακμές που ανήκουν σε καθένα από τα δύο σύνολα  $E_1$  και  $E_2$  να μην τέμνονται μεταξύ τους. Κατά μία έννοια λοιπόν, το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων είναι ισοδύναμο με το πρόβλημα της εύρεσης μιας γεωμετρικής RAC-απεικόνισης ενός δοθέντος γραφήματος (εάν μία τέτοια απεικόνιση υπάρχει), υποθέτοντας ότι το γράφημα-τομών της απεικόνισης δίνεται ως μέρος της εισόδου του προβλήματος.

Έστω  $G = (V, E_1 \cup E_2)$  το γράφημα το οποίο επάγεται από την ένωση των  $G_1$  και  $G_2$ . Προφανώς, το γράφημα  $G$  θα πρέπει να είναι RAC, το οποίο συνεπάγεται ότι  $|E_1 \cup E_2| \leq 4|V| - 10$  [DEL11]. Αναφερόμαστε σε αυτή τη σχέση ως τον *περιορισμό του RAC-μεγέθους (RAC-size constraint)*.

<sup>2</sup>Αυτό ερμηνεύεται ως εξής: "Εάν δύο ακμές μιας γεωμετρικής RAC-απεικόνισης διασταυρώνονται με μία τρίτη ακμή, τότε οι δύο αυτές ακμές είναι παράλληλες μεταξύ τους."



**Εικόνα 12.2:** (α) Ένα γράφημα με 8 κορυφές και 22 ακμές το οποίο δεν επιδέχεται RAC-απεικόνιση [EL11]. (β) Η αποσύνθεση του γραφήματος της Εικόνας 12.2i σε ένα επίπεδο γράφημα (solid ακμές; μία επίπεδη αναπαράσταση του δίνεται στην Εικόνα 12.2iii) και ένα ταίριασμα (διακεκομμένες ακμές) συνεπάγεται ότι ένα επίπεδο γράφημα και ένα ταίριασμα δεν επιδέχονται απαραίτητα γεωμετρικής RAC ταυτόχρονη απεικόνιση, καθώς η ένωση τους δεν ικανοποιεί τον περιορισμό του RAC-μεγέθους.

Εάν δύο γραφήματα δεν επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση τότε, προφανώς, δεν επιδέχονται και γεωμετρική RAC ταυτόχρονη απεικόνιση. Επί παραδείγματι, είναι γνωστό ότι υπάρχει ένα επίπεδο γράφημα και ένα ταίριασμα τα οποία δεν επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση. Ως εκ τούτου, δεν επιδέχονται και γεωμετρική RAC ταυτόχρονη απεικόνιση. Η Εικόνα 12.2 απεικονίζει μία εναλλακτική και πιο απλή τεχνική για την απόδειξη τέτοιου τύπου αρνητικών αποτελεσμάτων μη-ύπαρξης γεωμετρικών RAC ταυτόχρονων απεικονίσεων, η οποία βασίζεται στο γεγονός ότι όλα τα γραφήματα τα οποία ικανοποιούν τον περιορισμό του RAC-μεγέθους δεν είναι απαραίτητα RAC-γραφήματα. Από την άλλη πλευρά, δύο επίπεδα γραφήματα τα οποία επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση δεν επιδέχονται απαραίτητα και γεωμετρική RAC ταυτόχρονη απεικόνιση (η απόδειξη δίνεται στο Κεφάλαιο 14).

Ένα αρκετά συναφές πρόβλημα με αυτό της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων είναι το πρόβλημα της *γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης γραφημάτων* (*geometric Graph-Dual RAC simultaneous drawing problem* ή εν συντομία *GDual-GRacSim* πρόβλημα απεικόνισης γραφημάτων), το οποίο τυπικά ορίζεται ως εξής:

**Πρόβλημα 12.4.2. (Το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης γραφημάτων)** Δοθέντος ενός επίπεδου εμφυτευμένου γραφήματος  $G$ , το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης γραφημάτων αφορά στον καθορισμό μιας γεωμετρικής απεικόνισης του  $G$  και του ασθενούς δυϊκού  $G^*$  του  $G$  έτσι ώστε: (i) τα γραφήματα  $G$  και  $G^*$  απεικονίζονται επίπεδα, (ii) κάθε κορυφή του δυϊκού απεικονίζεται στο εσωτερικό της αντίστοιχης όψης του πρωτεύοντος γραφήματος  $G$  και, (iii) οι διασταυ-

ρώσεις μεταξύ των ακμών του πρωτεύοντος και του δυϊκού γραφήματος σχηματίζουν ορθές γωνίες.

Σημειώνεται ότι, το GDual-GRacSim πρόβλημα απεικόνισης γραφημάτων δεν είναι νέο πρόβλημα. Το 1963, ο W.T. Tutte ρώτησε εάν “Μπορούμε να κατασκευάσουμε γεωμετρικές ταυτόχρονες αναπαραστάσεις . . . των  $G$  και  $G^*$  στις οποίες . . . οι αντίστοιχες ακμές αναπαρίστανται ως κάθετα τμήματα;” [Tut63, p.p.767].

### 12.4.2 Συναφείς Εργασίες

Οι Brass et al. [BCD<sup>+</sup>07] παρουσιάζουν αλγορίθμους για την ταυτόχρονη απεικόνιση: (α) δύο μονοπατιών σε ένα  $n \times n$  ακέραιο πλέγμα, (β) ενός μονοπατιού και ενός extended-star γραφήματος σε ένα  $O(n^2) \times O(n)$  ακέραιο πλέγμα, (γ) ενός μονοπατιού και ενός caterpillar γραφήματος σε ένα  $3n \times 2n$  ακέραιο πλέγμα, (δ) δύο caterpillar γραφημάτων σε ένα  $3n \times 3n$  ακέραιο πλέγμα, (ε) δύο γράφημα-άστρων ( $k$  star graph, αντίστοιχα) σε ένα  $3 \times (n - 1)$  ( $O(n) \times O(n)$ , αντίστοιχα) ακέραιο πλέγμα. και (στ) δύο κύκλων σε ένα  $4n \times 4n$  ακέραιο πλέγμα. Αποδεικνύουν, επίσης, ότι υπάρχουν (α) τρία μονοπάτια και (β) δύο εξωεπίπεδα γραφήματα τα οποία δεν επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση. Οι Erten και Kobourov [EK05b] έδειξαν ότι ένα επίπεδο γράφημα και ένα μονοπάτι δεν επιδέχονται πάντοτε γεωμετρική ταυτόχρονη απεικόνιση. Οι Frati, Kaufmann και Kobourov [FKK09] απέδειξαν ότι το αρνητικό αυτό αποτέλεσμα ισχύει ακόμη και στην περίπτωση που το επίπεδο γράφημα και το μονοπάτι δεν έχουν κοινές ακμές.

Οι Duncan et al. [DEK04] έδειξαν ότι ένα γράφημα μέγιστου βαθμού τέσσερα έχει γεωμετρικό πάχος (geometric thickness) δύο, όπου το γεωμετρικό πάχος ενός γραφήματος ορίζεται ως ο ελάχιστος αριθμός επιπέδων (layers) που απαιτούνται για τη γεωμετρική απεικόνιση του γραφήματος και την αντιστοίχιση κάθε ακμής του σε ένα από τα διαθέσιμα επίπεδα, έτσι ώστε ακμές του ίδιου επιπέδου να μην διασταυρώνονται. Κατά συνέπεια, δύο γραφήματα μέγιστου βαθμού δύο επιδέχονται πάντοτε γεωμετρική ταυτόχρονη απεικόνιση. Το μειονέκτημα του αλγορίθμου τους είναι ότι οι προκύπτουσες απεικονίσεις απαιτούν μεγάλο χώρο σχεδίασης. Οι Cabello et al. [CvKL<sup>+</sup>11] έδειξαν ότι υπάρχει πάντοτε μία γεωμετρική ταυτόχρονη απεικόνιση ενός ταιριάσματος και (α) ενός γραφήματος-τροχού (wheel), (β) ενός εξωμονοπατιού (outerpath) ή (γ) ενός δένδρου. Απέδειξαν, επίσης, ότι υπάρχουν (α) ένα επίπεδο γράφημα και ένα ταιριασμα και (β) έξι ταιριάσματα τα οποία δεν επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση.

Οι Geyer, Kaufmann και Vrťo [GKV09] έδειξαν ότι μία γεωμετρική ταυτόχρονη απεικόνιση δύο δένδρων δεν υπάρχει πάντοτε. Οι Angelini et al. [AGKN10] έδειξαν ότι το ίδιο ισχύει και στην περίπτωση ενός δένδρου και ενός μονοπατιού. Οι Cappos et al. [CEBFK09] παρουσίασαν ένα γραμμικό αλγόριθμο, ο οποίος δοθέντος ενός επίπεδου γραφήματος και ενός μο-

νοπατιού κατασκευάζει μία γεωμετρική ταυτόχρονη απεικόνιση τους, στην περίπτωση που το επίπεδο γράφημα είναι επιπεδοποιημένο (level-planar), ως προς κάποια επιπεδοποίηση (layering) η οποία επάγεται από το μονοπάτι. Οι Estrella-Balderrama et al. [EBG]<sup>+</sup>07] απέδειξαν ότι το πρόβλημα του καθορισμού εάν δύο επίπεδα γραφήματα επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση είναι  $\mathcal{NP}$ -δύσκολο. Για μία πιο πλήρη επισκόπηση γνωστών αποτελεσμάτων αυτής της ερευνητικής περιοχής, ο αναγνώστης παραπέμπεται στις εργασίες [BKR07, BKR12].

Οι Brightwell και Scheinermann [BS93] έδειξαν ότι το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης γραφημάτων επιδέχεται πάντοτε λύση, εάν το γράφημα εισόδου είναι ένα τρισυνδεδεμένο επίπεδο γράφημα. Από όσο είμαστε σε θέση να γνωρίζουμε, αυτό είναι και το μόνο γνωστό αποτέλεσμα για το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης γραφημάτων. Οι Erten και Kobourov [EK05a] παρουσιάζουν έναν αλγόριθμο πολυπλοκότητας  $O(n)$  ο οποίος κατασκευάζει μία γεωμετρική γραφο-δυϊκή ταυτόχρονη απεικόνιση (αλλά όχι απαραίτητα RAC) ενός τρισυνδεδεμένου επίπεδου γραφήματος και του δυϊκού του σε ένα  $O(n^2)$  ακέραιο πλέγμα, όπου  $n$  είναι το πλήθος των κορυφών τόσο του πρωτεύοντος, όσο και του δυϊκού γραφήματος.



## 13 Το Πρόβλημα της Γεωμετρικής RAC Απεικόνισης είναι NP-δύσκολο

Σε αυτό το κεφάλαιο, επικεντρώνουμε στις γεωμετρικές RAC απεικονίσεις (ή RAC απεικονίσεις με ευθείες γραμμές) και παρουσιάζουμε μία άπειρη κλάση γραφημάτων με μοναδική RAC συνδυαστική εμφύτευση. Χρησιμοποιούμε μέλη της κλάσης αυτής για να δείξουμε ότι είναι NP-δύσκολο να καθορίσουμε εάν ένα γράφημα επιδέχεται μία γεωμετρική RAC απεικόνιση.

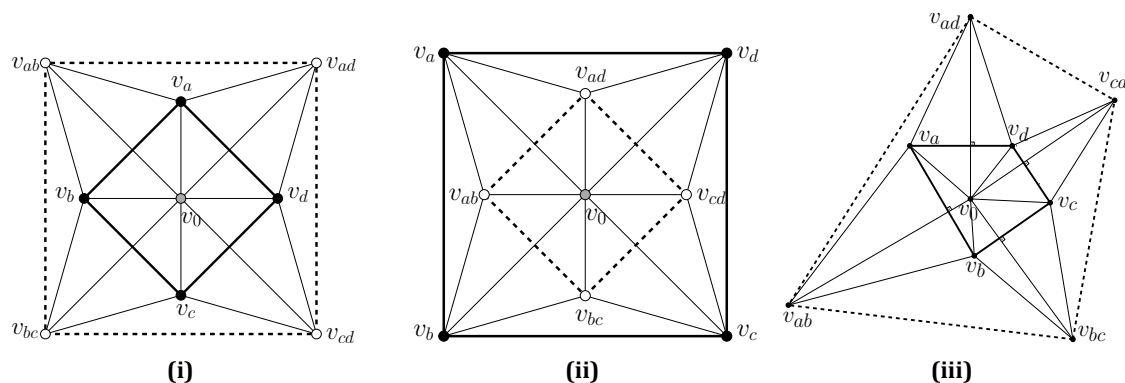
### 13.1 Εισαγωγικές Έννοιες και Ορολογία

Πριν προχωρήσουμε στη λεπτομερή περιγραφή της προσέγγισής μας, εισάγουμε προκαταρκτικές έννοιες και ορολογία που θα χρησιμοποιηθούν αρκετά στη συνέχεια του κεφαλαίου. Δοθείσας μιας απεικόνισης  $\Gamma(G)$  ενός γραφήματος  $G$ , συμβολίζουμε με  $\ell_{u,v}$  την ευθεία που περνάει από τις κορυφές  $u$  και  $v$ . Με  $\ell'_{u,v}$ , συμβολίζουμε την ημιευθεία που ξεκινάει από την κορυφή  $u$  με κατεύθυνση προς την κορυφή  $v$ . Όμοια, συμβολίζουμε με  $\ell_{u,v,w}$  ( $\ell'_{u,v,w}$ ) την ευθεία (ημιευθεία) που περνάει (ξεκινάει) από την κορυφή  $u$  και είναι κάθετη στην ακμή  $(v, w)$ .

### 13.2 Μία Κλάση Γραφημάτων με Μοναδική RAC Συνδυαστική Εμφύτευση

Το βασικό αποτέλεσμα αυτού του κεφαλαίου, δηλαδή η απόδειξη ότι το πρόβλημα της γεωμετρικής RAC απεικόνισης είναι NP-δύσκολο, βασίζεται σε μία αναγωγή από το γνωστό πρόβλημα 3-SAT [GJ79]. Βασικό στοιχείο αυτής της αναγωγής είναι ένα γράφημα, που ονομάζεται *επαυξημένο τετραγωνικό αντιπρισματικό γράφημα* (*augmented square antiprism graph*) ή *ASA γράφημα* για συντομία, το οποίο έχει την ακόλουθη ιδιότητα: “Οι γεωμετρικές RAC απεικονίσεις ενός ASA γραφήματος ορίζουν ακριβώς δύο συνδυαστικές εμφυτεύσεις”. Η Εικόνα 13.1 παρουσιάζει το ASA γράφημα και τις δύο συνδυαστικές του εμφυτεύσεις (το γεγονός ότι αυτές είναι οι μοναδικές δύο RAC συνδυαστικές εμφυτεύσεις θα αποδειχθεί αργότερα σε αυτό το κεφάλαιο). Παρατηρήστε ότι το ASA γράφημα αποτελείται από μία

“κεντρική κορυφή”  $v_0$ , η οποία είναι γειτονική με όλες τις κορυφές του γραφήματος και δύο τετράπλευρα (βλ. το διακεκομμένο και το έντονο ζωγραφισμένο τετράγωνο της Εικόνας 13.1i), τα οποία συμβολίζονται με  $\mathcal{Q}_1$  και  $\mathcal{Q}_2$ . Αφαιρώντας την κεντρική κορυφή, το προκύπτων γράφημα αντιστοιχεί στο σκελετό ενός τετραγωνικού αντιπρίσματος, το οποίο συνήθως αναφέρεται ως *τετραγωνικό αντιπρισματικό γράφημα* (*square antiprism graph*).

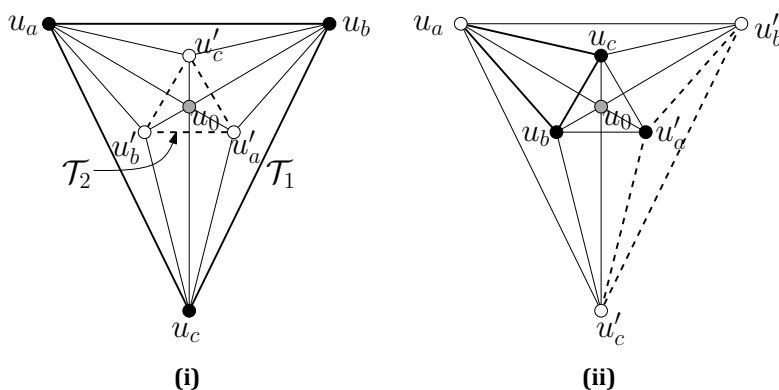


**Εικόνα 13.1:** (i),(ii) Οι δύο διαφορετικές RAC απεικονίσεις του ASA γραφήματος με διαφορετικές συνδυαστικές εμφυτεύσεις. (i),(iii) Δύο διαφορετικές RAC απεικονίσεις του ASA γραφήματος με την ίδια συνδυαστική εμφύτευση.

Εάν στο ASA γράφημα αντικαταστήσουμε τα δύο τετράπλευρα με δύο τρίγωνα, τότε το γράφημα που προκύπτει είναι το *επαυξημένο τριγωνικό αντιπρισματικό γράφημα* (*augmented triangular antiprism graph*) ή *ATA γράφημα* για συντομία. Οι Didimo, Eades και Liotta [DEL11], οι οποίοι απέδειξαν ότι κάθε γράφημα με  $n$ -κορυφές το οποίο επιδέχεται μία RAC απεικόνιση μπορεί να έχει το πολύ  $4n - 10$  ακμές, χρησιμοποίησαν το επαυξημένο τριγωνικό αντιπρισματικό γράφημα ως ένα παράδειγμα ενός γραφήματος που επιτυγχάνει το φράγμα των  $4n - 10$  ακμών (βλ. Εικόνα. 1.c στην εργασία [DEL11]). Σε αντίθεση με το επαυξημένο τριγωνικό αντιπρισματικό γράφημα, το επαυξημένο τετραγωνικό αντιπρισματικό γράφημα δεν επιτυγχάνει το άνω φράγμα για τον αριθμό των ακμών. Γενικά, η κλάση των *επαυξημένων  $k$ -γωνικών αντιπρισματικών γραφημάτων*,  $k \geq 3$ , είναι μία κλάση μη-επίπεδων γραφημάτων, τα οποία επιδέχονται όλα RAC απεικονίσεις. Υπενθυμίζουμε ότι κάθε επίπεδο γράφημα  $n$ -κορυφών έχει το πολύ  $3n - 6$  ακμές, και δεδομένου ότι ένα επαυξημένο  $k$ -γωνικό αντιπρισματικό γράφημα έχει  $2k + 1$  κορυφές και  $6k$  ακμές, δεν είναι επίπεδο για όλη την κλάση αυτών των γραφημάτων.

Παρατηρείστε ότι στο τριγωνικό αντιπρισματικό γράφημα της Εικόνας 13.2i, οι κορυφές (i)  $u_a$  και  $u'_a$ , (ii)  $u_b$  και  $u'_b$  και (iii)  $u_c$  και  $u'_c$  μοιράζονται τους ίδιους γείτονες. Έστω  $\mathcal{T}_1$  ( $\mathcal{T}_2$ , αντίστοιχα) το τρίγωνο που σχηματίζεται από τις κορυφές  $u_a, u_b$  και  $u_c$  ( $u'_a, u'_b$  και  $u'_c$ , αντίστοιχα). Η πρόθεσή μας είναι να αποδείξουμε ότι οι γεωμετρικές RAC απεικονίσεις





**Εικόνα 13.2:** (i)-(ii) Δύο διαφορετικές RAC απεικονίσεις του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος με διαφορετικές συνδυαστικές εμφυτεύσεις.

του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος ορίζουν ακριβώς οκτώ συνδυαστικές εμφυτεύσεις, οι οποίες έπονται από την απεικόνιση της Εικόνας 13.2i με αμοιβαίες αλλαγές των θέσεων των κορυφών (i)  $u_a$  και  $u'_a$ , (ii)  $u_b$  και  $u'_b$  και (iii)  $u_c$  και  $u'_c$ . Στη συνέχεια, εξετάζονται κάποιες ιδιότητες των RAC απεικονίσεων του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος.

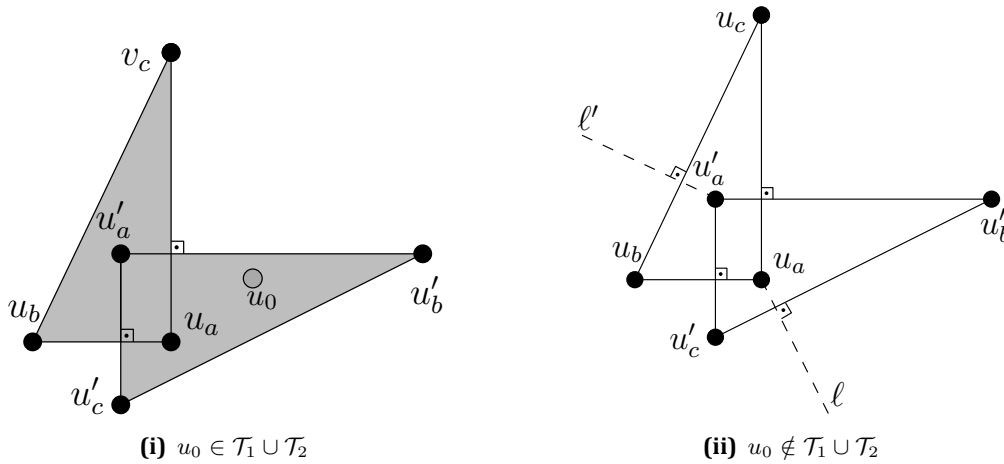
**Ιδιότητα 13.2.1.** Σε κάθε RAC απεικόνιση του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος, η εξωτερική όψη φράσσεται από τρεις ακμές.

*Απόδειξη.* Έπεται άμεσα από τη RAC-Ιδιότητα 4 και από το γεγονός ότι, το επαυξημένο τριγωνικό αντιπρισματικό γράφημα είναι μεγιστικόως πυκνό. □

**Ιδιότητα 13.2.2.** Σε κάθε RAC απεικόνιση του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος, τα τρίγωνα  $T_1$  και  $T_2$  δεν τέμνονται.

*Απόδειξη.* Έστω ότι υπάρχει μία RAC απεικόνιση του ATA γραφήματος στην οποία τα τρίγωνα  $T_1$  και  $T_2$  τέμνονται. Η τομή τους θα εμπλέκει τέσσερις ακμές, δύο από κάθε τρίγωνο (βλ. Εικόνα 13.3). Συνεπώς, μία κορυφή του  $T_1$  θα πρέπει να βρίσκεται στο εσωτερικό του  $T_2$  και μία κορυφή του  $T_2$  θα πρέπει να βρίσκεται στο εσωτερικό του  $T_1$ . Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι αυτό το ζεύγος αποτελείται από τις κορυφές  $u_a$  και  $u'_a$  (όλα τα υπόλοιπα ζεύγη αντιμετωπίζονται με τον ίδιο τρόπο). Συνεπώς, οι ακμές  $(u_a, u_b)$ ,  $(u_a, u_c)$ ,  $(u'_a, u'_b)$  και  $(u'_a, u'_c)$  είναι οι μόνες που εμπλέκονται σε διασταυρώσεις. Εάν η κορυφή  $u_0$  βρίσκεται στο εσωτερικό του  $T_1 \cup T_2$  (βλ. τη γκρι-σκιασμένη περιοχή της Εικόνας 13.3i), τότε η εξωτερική όψη φράσσεται από τέσσερις ακμές, το οποίο αντιβαίνει στην Ιδιότητα 13.2.1.

Ας υποθέτουμε τώρα ότι η κορυφή  $u_0$  βρίσκεται στο εσωτερικό του  $(T_1 \cup T_2)^c$ . Η περίπτωση αυτή περιγράφεται στην Εικόνα 13.3ii. Δεδομένου ότι η κορυφή  $u_0$  συνδέεται με την



**Εικόνα 13.3:** Τα τρίγωνα  $T_1$  και  $T_2$  τέμνονται.

κορυφή  $u_a$ , θα πρέπει να βρίσκεται κατά μήκος της ημιευθείας, έστω  $\ell$ , η οποία ξεκινάει από την κορυφή  $u_a$  και είναι κάθετη στην ακμή  $(u'_b, u'_c)$ . Από την άλλη, η κορυφή  $u_0$  θα πρέπει, επίσης, να ενωθεί με την κορυφή  $u'_a$ . Επομένως, η κορυφή  $u_0$  θα πρέπει να βρίσκεται κατά μήκος της ημιευθείας, έστω  $\ell'$ , η οποία ξεκινάει από την κορυφή  $u_a$  και είναι κάθετη στην ακμή  $(u_b, u_c)$ . Συνεπώς, οι ημιευθείες  $\ell$  και  $\ell'$  θα πρέπει να τέμνονται, το οποίο οδηγεί σε άτοπο λόγω της κατασκευής.  $\square$

**Ιδιότητα 13.2.3.** Δεν υπάρχει RAC απεικόνιση του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος στην οποία ο 3-κύκλος που πρόσκειται στην εξωτερική όψη να περιέχει μία κορυφή του  $T_1$  και μία κορυφή του  $T_2$  που έχουν τους ίδιους γείτονες.

*Απόδειξη.* Έστω ότι οι κορυφές  $u_a \in T_1$  και  $u'_a \in T_2$  (οι οποίες έχουν τους ίδιους γείτονες) βρίσκονται στον 3-κύκλο που πρόσκειται στην εξωτερική όψη. Παρατηρείστε τώρα, ότι η ακμή  $(u_a, u'_a)$  δεν είναι παρούσα στο ATA γράφημα, το οποίο αντιβαίνει στη RAC-Ιδιότητα 13.2.1. Οι περιπτώσεις όπου είτε οι κορυφές  $u_b$  και  $u'_b$  ή οι κορυφές  $u_c$  και  $u'_c$  ανήκουν στον 3-κύκλο που πρόσκειται στην εξωτερική όψη, αντιμετωπίζονται συμμετρικά.  $\square$

**Ιδιότητα 13.2.4.** Δεν υπάρχει RAC απεικόνιση του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος στην οποία ο 3-κύκλος που πρόσκειται στην εξωτερική όψη να περιέχει την κορυφή  $u_0$ .

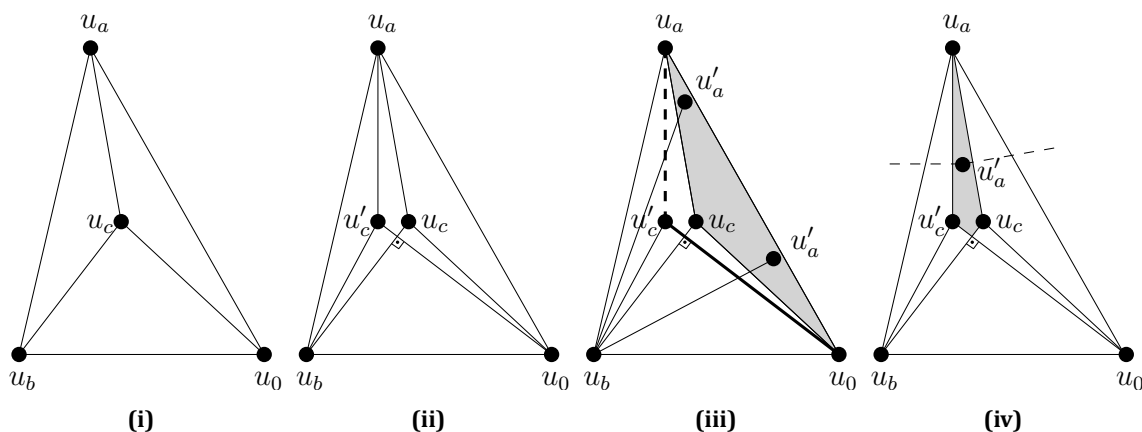
*Απόδειξη.* Έστω ότι σε μία RAC απεικόνιση του ATA γραφήματος, οι κορυφές  $u_a$  και  $u_b$ <sup>1</sup> (οι οποίες έχουν τους ίδιους γείτονες) ανήκουν στον 3-κύκλο που πρόσκειται στην εξωτερική

<sup>1</sup>Εάν χρησιμοποιήσουμε την κορυφή  $u'_b$  αντί της  $u_b$ , η απόδειξη είναι πανομοιότυπη.

όψη μαζί με την κορυφή  $u_0$ . Δεδομένου ότι οι κορυφές  $u_a$ ,  $u_b$  και  $u_0$  σχηματίζουν την εξωτερική όψη, η κορυφή  $u_c$  (που είναι γειτονική με τις κορυφές  $u_a$ ,  $u_b$  και  $u_0$ ) θα πρέπει να είναι μία εσωτερική κορυφή και θα χωρίζει το εσωτερικό του 3-κύκλου σε τρεις τριγωνικές περιοχές  $\Delta u_a u_b u_c$ ,  $\Delta u_a u_c u_0$  και  $\Delta u_b u_c u_0$  (βλ. Εικόνα 13.4i). Στη συνέχεια, θεωρούμε την κορυφή  $u'_c$ , η οποία έχει τους ίδιους γείτονες με την κορυφή  $u_c$ . Η κορυφή  $u'_c$  θα πρέπει να βρίσκεται στο εσωτερικό ενός εκ των τριγώνων  $\Delta u_a u_b u_c$ ,  $\Delta u_a u_c u_0$  ή  $\Delta u_b u_c u_0$ . Θα αποδείξουμε τυπικά την περίπτωση όπου η κορυφή  $u'_c$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta u_a u_b u_c$ , όπως περιγράφεται στην Εικόνα 13.4ii. Η περίπτωση όπου η κορυφή  $u'_c$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta u_a u_c u_0$  είναι συμμετρική, καθώς οι κορυφές  $u_c$  και  $u'_c$  έχουν τους ίδιους γείτονες. Η περίπτωση όπου η κορυφή  $u'_c$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta u_b u_c u_0$  μπορεί να αποδειχθεί όμοια. Παρατηρείστε τώρα ότι η ακμή  $(u_0, u'_c)$  θα πρέπει να τέμνει είτε την ακμή  $(u_b, u_c)$  ή την ακμή  $(u_a, u_c)$ . Θα αποδείξουμε την πρώτη περίπτωση. Η δεύτερη μπορεί, επίσης, να αποδειχθεί με παρόμοια επιχειρήματα.

Το άτοπο έπεται από την κορυφή  $u'_a$ , για την οποία θα αποδείξουμε ότι δεν υπάρχει εφικτή τοποθέτηση. Υπενθυμίζεται ότι η κορυφή  $u'_a$  είναι γειτονική με τις κορυφές  $u_b$ ,  $u'_b$ ,  $u_c$ ,  $u'_c$  και  $u_0$ . Αρχικά, αποδεικνύεται ότι η κορυφή  $u'_a$  δεν μπορεί να βρίσκεται στο εσωτερικό ενός τριγώνου που σχηματίζεται από δύο κορυφές της εξωτερικής όψης και είτε την κορυφή  $u_c$  ή την κορυφή  $u'_c$ , δηλαδή σε μία τριγωνική περιοχή που πρόσκειται στην εξωτερική όψη. Για να το αποδείξουμε αυτό, έστω ότι η κορυφή  $u'_a$  βρίσκεται στο εσωτερικό ενός τέτοιου τριγώνου, έστω στο εσωτερικό του τριγώνου  $\Delta u_0 u_a u_c$  (βλ. τη γκρι-σκιασμένη περιοχή της Εικόνας 13.4iii). Σε αυτή την περίπτωση, η ακμή  $(u'_a, u_b)$  δεν μπορεί να τέμνει την ακμή  $(u'_c, u_0)$  (βλ. την έντονα ζωγραφισμένη ακμή της Εικόνας 13.4iii), καθώς αυτό θα οδηγούσε σε μία κατάσταση στην οποία οι ακμές  $(u'_a, u_b)$  και  $(u_b, u_c)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $u_b$  που τέμνεται από την  $(u_0, u'_c)$ , το οποίο αντιβαίνει στη RAC-Ιδιότητα 2. Επιπρόσθετα, η ακμή  $(u'_a, u_b)$  δεν μπορεί να τέμνει την ακμή  $(u_a, u'_c)$  (βλ. την έντονα διακεκομμένη ακμή της Εικόνας 13.4iii), καθώς αυτό θα οδηγούσε σε μία κατάσταση όπου οι ακμές  $(u_a, u'_c)$  και  $(u_a, u_c)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $u_a$  που τέμνεται από την ακμή  $(u'_a, u_b)$ , το οποίο αντιβαίνει στη RAC-Ιδιότητα 2. Επομένως, η κορυφή  $u'_a$  δεν μπορεί να ενωθεί με την κορυφή  $u_b$ . Όμοια, μπορεί να αποδειχθεί ότι η  $u'_a$  δεν μπορεί να βρίσκεται στο εσωτερικό των τριγώνων  $\Delta u_a u_b u'_c$ ,  $\Delta u_b u_c u_0$  και  $\Delta u_b u'_c u_0$ . Σημειώνεται ότι οι τελευταίες δύο περιπτώσεις μπορούν να αποδειχθούν εύκολα με βάση τη RAC-Ιδιότητα 2 και το γεγονός ότι οι ακμές  $(u_b, u_c)$  και  $(u_0, u'_c)$  έχουν ήδη εμπλακεί σε μία τομή υπό ορθή γωνία. Έχοντας αποκλείσει την περίπτωση η κορυφή  $u'_a$  να βρίσκεται στο εσωτερικό μιας τριγωνικής περιοχής που πρόσκειται στην εξωτερική όψη, η μόνη περίπτωση που απομένει είναι να θεωρήσουμε ότι η κορυφή  $u'_a$  βρίσκεται στο εσωτερικό μιας περιοχής που σχηματίζεται από τις κορυφές  $u_a$ ,  $u_c$ ,  $u'_c$  και το σημείο τομής των ακμών  $(u_0, u'_c)$  και  $(u_b, u_c)$  (βλ. τη γκρι-σκιασμένη περιοχή

της Εικόνας 13.4iv). Σε αυτή την περίπτωση, η ακμή  $(u'_a, u_b)$  δεν μπορεί να τέμνει την ακμή  $(u'_c, u_0)$ , λόγω της RAC-Ιδιότητας 2. Για τον ίδιο λόγο, η ακμή  $(u'_a, u_0)$  δεν μπορεί να τέμνει την ακμή  $(u_b, u_c)$ . Επομένως, για να πραγματοποιηθούν οι ενώσεις της κορυφής  $u'_a$  με τις κορυφές  $u_b$  και  $u_0$ , οι ακμές  $(u_a, u'_c)$  και  $(u_a, u_c)$  θα πρέπει να τέμνουν κάθετα τις ακμές  $(u'_a, u_b)$  και  $(u'_a, u_0)$ , αντίστοιχα, το οποίο οδηγεί σε άτοπο, καθώς η εξωτερική όψη είναι τρίγωνο.



**Εικόνα 13.4:** Οι κορυφές  $u_a, u_b$  και  $u_0$  σχηματίζουν την εξωτερική όψη, η κορυφή  $u_c$  είναι μία εσωτερική κορυφή (που ορίζει τρεις τριγωνικές περιοχές  $\Delta u_a u_b u_c$ ,  $\Delta u_b u_c u_0$  και  $\Delta u_a u_c u_0$ ) και η κορυφή  $u'_c$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta u_a u_b u_c$ .

□

**Λήμμα 13.2.1.** Οι γεωμετρικές RAC απεικονίσεις του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος ορίζουν ακριβώς οκτώ συνδυαστικές εμφυτεύσεις.

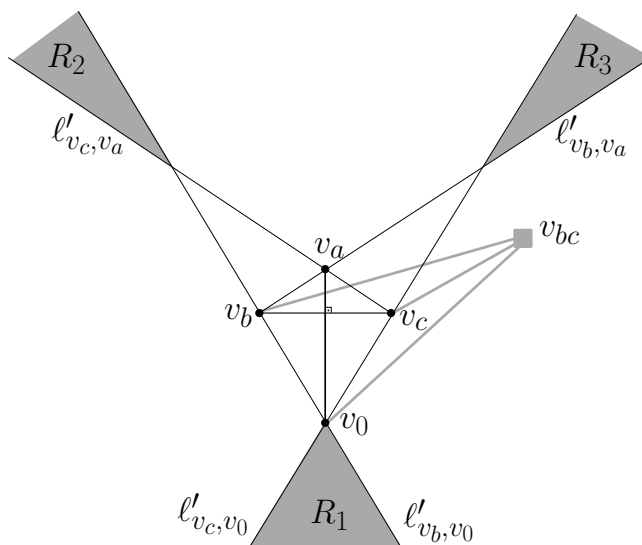
*Απόδειξη.* Υπενθυμίζεται ότι οι κορυφές (i)  $u_a$  και  $u'_a$ , (ii)  $u_b$  και  $u'_b$ , και (iii)  $u_c$  και  $u'_c$  μοιράζονται τους ίδιους γείτονες. Αυτό σε συνδυασμό με τις Ιδιότητες 13.2.1, 13.2.2, 13.2.3 και 13.2.4 οδηγεί στις οκτώ συνδυαστικές εμφυτεύσεις του επαυξημένου τριγωνικού αντιπρισματικού γραφήματος (που έπονται από την απεικόνιση της Εικόνας 13.2i με αμοιβαίες αλλαγές των θέσεων των κορυφών (i)  $u_a$  και  $u'_a$ , (ii)  $u_b$  και  $u'_b$  και (iii)  $u_c$  και  $u'_c$ ). □

Στη συνέχεια, αποδεικνύεται ότι οι γεωμετρικές απεικονίσεις ενός τετραγωνικού αντιπρισματικού γραφήματος ορίζουν ακριβώς δύο συνδυαστικές εμφυτεύσεις.

**Λήμμα 13.2.2.** Δεν υπάρχει RAC απεικόνιση του ASA γραφήματος στην οποία (i) η κεντρική κορυφή  $v_0$  να βρίσκεται εξωτερικά του τετραπλεύρου  $Q_i$  και (ii) μία ακμή που συνδέει την κορυφή  $v_0$  με μία κορυφή του  $Q_i$  να τέμνει μία άλλη ακμή του  $Q_i$ ,  $i = 1, 2$ .

*Απόδειξη.* Έστω  $Q$  ένα από τα τετράπλευρα  $Q_i$ ,  $i = 1, 2$  και  $v_a, v_b, v_c$  και  $v_d$  οι κορυφές του, διαδοχικές κατά μήκος του τετραπλεύρου  $Q$  (βλ. Εικόνα 13.5). Έστω τώρα ότι η κορυφή  $v_0$

βρίσκεται εξωτερικά του τετραπλεύρου  $\mathcal{Q}$  και υπάρχει μία ακμή, έστω  $(v_0, v_a)$ , που ξεκινάει από την κορυφή  $v_0$  προς την κορυφή  $v_a$  του τετραπλεύρου  $\mathcal{Q}$ , τέτοια ώστε να τέμνει μία ακμή, έστω την  $(v_b, v_c)^2$ , του  $\mathcal{Q}$  (βλ. Εικόνα 13.5). Στο ASA γράφημα, οι κορυφές  $v_b$  και  $v_c$  έχουν τις ακόλουθες ιδιότητες: (α) είναι και οι δύο ενωμένες με την κορυφή  $v_0$  και (β) έχουν ένα κοινό γείτονα την κορυφή  $v_{bc}$ , η οποία είναι γείτονας των κορυφών  $v_0$  και  $v_{cd} \notin \mathcal{Q}$  (βλ. Εικόνα 13.1).



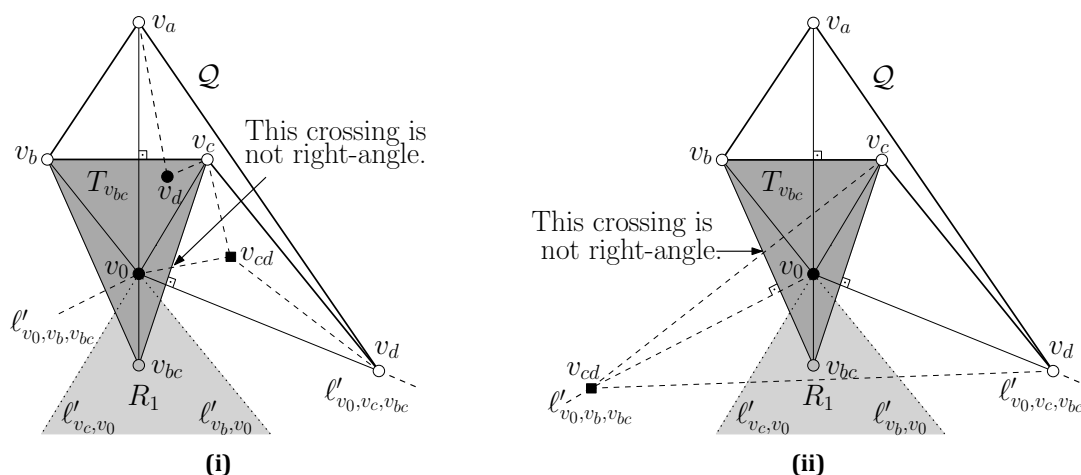
**Εικόνα 13.5:** Ο σχηματισμός που χρησιμοποιείται για την απόδειξη του Λήμματος 13.2.2: Η κορυφή  $v_{bc}$  θα πρέπει να βρίσκεται στο εσωτερικό μιας εκ των περιοχών  $R_1, R_2$  και  $R_3$ .

Παρατηρείστε ότι, εάν η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό των μη-σκιασμένων περιοχών της Εικόνας 13.5, τότε τουλάχιστον μία από τις ακμές που πρόσκεινται στην κορυφή  $v_{bc}$  θα τέμνει είτε την ακμή  $(v_0, v_a)$  ή την ακμή  $(v_b, v_c)$ , οι οποίες έχουν ήδη εμπλακεί σε μία τομή υπό ορθή γωνία. Αυτό οδηγεί σε μία κατάσταση στην οποία τρεις ακμές τέμνονται μεταξύ τους ανά δύο, που αντιβαίνει στη RAC-Ιδιότητα 1. Επομένως, η κορυφή  $v_{bc}$  θα πρέπει να βρίσκεται στο εσωτερικό των γκρι-σκιασμένων περιοχών  $R_1, R_2$  ή  $R_3$  της Εικόνας 13.5. Στη συνέχεια, θεωρούμε καθεμία από αυτές τις περιπτώσεις ξεχωριστά. Σημειώνεται ότι ανάλογα με τις θέσεις των κορυφών  $v_a, v_b, v_c$  και  $v_0$ , οι περιοχές  $R_2$  ή  $R_3$  ή  $R_2 \cup R_3$  μπορεί να είναι κενές.

**Περίπτωση i:** Η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό της  $R_1$ . Η περίπτωση αυτή περιγράφεται στην Εικόνα 13.6. Έστω  $T_{v_{bc}}$  η περιοχή που σχηματίζεται από τις κορυφές  $v_{bc}, v_b$  και  $v_c$  (βλ. τη σκούρα-γκρι σκιασμένη περιοχή της Εικόνας 13.6). Η κορυφή  $v_a$  που πρέπει να ενωθεί με τις κορυφές  $v_a, v_c$  και την κεντρική κορυφή  $v_0$  δεν μπορεί να βρίσκεται στο εσωτερικό

<sup>2</sup>Η περίπτωση όπου τέμνει την ακμή  $(v_c, v_a)$  είναι συμμετρική.

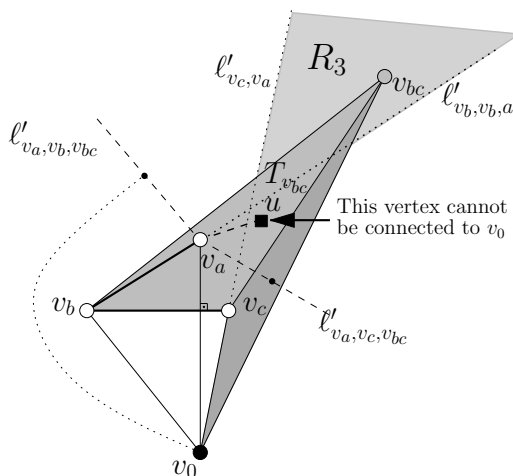
της  $T_{v_{bc}}$ , καθώς οι ακμές  $(v_a, v_0)$  και  $(v_a, v_d)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_a$  που τέμνεται από την ακμή  $(v_b, v_c)$ , το οποίο αντιβαίνει στη RAC-Ιδιότητα 2. Δεδομένου ότι η κορυφή  $v_d$  πρέπει να ενωθεί με την κορυφή  $v_0$ , θα πρέπει να βρίσκεται είτε κατά μήκος της ημιευθείας  $\ell'_{v_0, v_c, v_{bc}}$  ή κατά μήκος της ημιευθείας  $\ell'_{v_0, v_b, v_{bc}}$ . Θα εξετάσουμε μόνο την πρώτη περίπτωση. Η δεύτερη αντιμετωπίζεται συμμετρικά. Ωστόσο, υπό αυτό τον περιορισμό, η κοινή γειτονική κορυφή  $v_{cd}$  των  $v_c$  και  $v_d$  δεν μπορεί να ενωθεί με την κορυφή  $v_0$ , καθώς η ακμή  $(v_0, v_{cd})$  θα πρέπει να είναι κάθετη σε μία από τις ακμές της περιοχής  $T_{v_{bc}}$ . Για να δειχθεί αυτό, παρατηρήστε ότι εάν η ακμή  $(v_0, v_{cd})$  είναι κάθετη στην  $(v_c, v_{bc})$ , τότε οι ακμές  $(v_0, v_{cd})$  και  $(v_0, v_d)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_0$  που τέμνεται από την ακμή  $(v_c, v_{bc})$ , το οποίο αντιβαίνει στη RAC-Ιδιότητα 2 (βλ. Εικόνα 13.6i). Θεωρήστε τώρα την περίπτωση όπου η ακμή  $(v_0, v_{cd})$  είναι κάθετη στην ακμή  $(v_b, v_{bc})$  (και βρίσκεται κατά μήκος της ημιευθείας  $\ell'_{v_0, v_b, v_{bc}}$ , βλ. Εικόνα 13.6ii). Δεδομένου ότι η γωνία  $\widehat{v_c v_b v_{bc}}$  είναι οξεία, η ακμή  $(v_b, v_{bc})$  θα τέμνεται και από τις δύο ακμές  $(v_0, v_{cd})$  και  $(v_c, v_{cd})$ , που σχηματίζουν μία έλικα με κέντρο την  $v_{cd}$ . Αυτό οδηγεί σε άτοπο, καθώς παραβιάζεται η RAC-Ιδιότητα 2.



**Εικόνα 13.6:** Ο σχηματισμός που χρησιμοποιείται για την απόδειξη του Λήμματος 13.2.2: Η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό της περιοχής  $R_1$ .

**Περίπτωση ii:** Η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό είτε της περιοχής  $R_2$  ή της περιοχής  $R_3$ . Υποθέτουμε χωρίς βλάβη της γενικότητας, ότι η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό της περιοχής  $R_3$ . Η περίπτωση αυτή απεικονίζεται στην Εικόνα 13.7. Έστω  $u$  μία κορυφή του ASA γραφήματος (διάφορη των  $v_a, v_b, v_c$  και  $v_0$ ) και υποθέστε ότι η κορυφή  $u$  βρίσκεται στο εσωτερικό του τριγώνου  $T_{v_{bc}}$  που σχηματίζεται από τις κορυφές  $v_b, v_c$  και  $v_{bc}$ . Η κορυφή  $u$  πρέπει να ενωθεί με την κεντρική κορυφή  $v_0$ . Η ακμή  $(v_0, u)$  δεν μπορεί να εμπλακεί σε τομές ούτε με την ακμή  $(v_b, v_c)$ , καθώς οι ακμές  $(v_0, u)$  και  $(v_0, v_a)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_0$  που τέμνεται από την  $(v_b, v_c)$  ούτε με την ακμή  $(v_c, v_{bc})$ , καθώς η γωνία  $\widehat{v_b v_c v_{bc}}$  είναι

μικρότερη των  $180^\circ$ . Επομένως, το τρίγωνο  $T_{v_{bc}}$  δεν μπορεί να εμπεριέχει καμία άλλη κορυφή (εκτός από την κορυφή  $v_a$ ). Παρατηρείστε τώρα, ότι κάθε κορυφή του τετραπλεύρου  $\mathcal{Q}$  έχει βαθμό πέντε και δεν υπάρχουν τρεις κορυφές του τετραπλεύρου  $\mathcal{Q}$ , οι οποίες έχουν κοινή γειτονική κορυφή (βλ. Εικόνα 13.1). Αυτές οι ιδιότητες τετριμμένα ισχύουν και για την κορυφή  $v_a$ , καθώς  $v_a \in \mathcal{Q}$ . Με βάση τις παραπάνω ιδιότητες, κάθε γειτονική κορυφή της κορυφής  $v_a$  μπορεί να βρίσκεται είτε στο εσωτερικό της σκούρας-γκρι σκιασμένης περιοχής της Εικόνας 13.7 ή στην εξωτερική όψη της ήδη κατασκευασμένης απεικόνισης (κατά μήκος των διακεκομμένων ημιευθειών  $\ell'_{v_a, v_c, v_{bc}}$  και  $\ell'_{v_a, v_b, v_{bc}}$  της Εικόνας 13.7, αντίστοιχα). Αυτό συνεπάγεται ότι μπορούμε να τοποθετήσουμε μόνο τέσσερις από τις πέντε γειτονικές κορυφές της κορυφής  $v_a$ , δηλαδή μία από αυτές θα πρέπει να βρίσκεται στο εσωτερικό του τριγώνου  $T_{v_{bc}}$  και επομένως, δεν μπορεί να ενωθεί με την κορυφή  $v_0$ .



**Εικόνα 13.7:** Ο σχηματισμός που χρησιμοποιείται για την απόδειξη του Λήμματος 13.2.2: Η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό της περιοχής  $R_3$ .

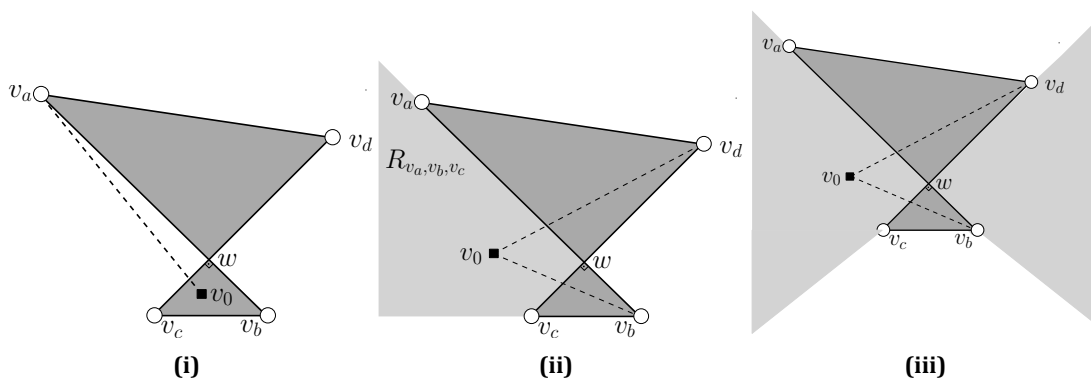
Από την παραπάνω ανάλυση, έπεται ότι η κεντρική κορυφή  $v_0$  δεν μπορεί να βρίσκεται εξωτερικά του τετραπλεύρου  $\mathcal{Q}$ , έτσι ώστε μία ακμή που ενώνει την κορυφή  $v_0$  με μία κορυφή του  $\mathcal{Q}$  να τέμνει μία ακμή του  $\mathcal{Q}$ . □

**Λήμμα 13.2.3.** Σε κάθε RAC απεικόνιση του ASA γραφήματος, το τετράπλευρο  $\mathcal{Q}_i$  απεικονίζεται επίπεδο, για κάθε  $i = 1, 2$ .

*Απόδειξη.* Έστω  $\mathcal{Q}$  ένα από τα τετράπλευρα  $\mathcal{Q}_i$ ,  $i = 1, 2$  και έστω, όπως και στην απόδειξη του προηγούμενου λήμματος,  $v_a, v_b, v_c$  και  $v_d$  οι κορυφές του, διαδοχικές κατά μήκος του τετραπλεύρου  $\mathcal{Q}$ . Σημειώνεται ότι δεν είναι δυνατόν μία μη-επίπεδη, γεωμετρική RAC απεικόνιση του τετραπλεύρου να περιέχει περισσότερες από μία ορθογώνιες τομές (η απεικόνιση

του  $\mathcal{Q}$  σχηματίζει δύο ορθογώνια τρίγωνα). Έστω τώρα, ότι σε μία RAC απεικόνιση του ASA γραφήματος, το τετράπλευρο  $\mathcal{Q}$  δεν είναι επίπεδο και έστω ότι οι ακμές  $(v_a, v_b)$  και  $(v_c, v_d)$  σχηματίζουν μία ορθογώνια τομή. Η περίπτωση αυτή απεικονίζεται στην Εικόνα 13.8i. Στη συνέχεια, θα θεωρήσουμε τις περιπτώσεις στις οποίες η κεντρική κορυφή  $v_0$  βρίσκεται είτε στο εσωτερικό ενός εκ των δύο ορθογωνίων τριγώνων ή στην περιοχή εξωτερικά τους. Και στις δύο περιπτώσεις θα καταλήξουμε σε άτοπο.

**Περίπτωση i:** Η κορυφή  $v_0$  βρίσκεται στο εσωτερικό ενός εκ των δύο τριγώνων. Υποθέτουμε χωρίς βλάβη της γενικότητας, ότι η κορυφή  $v_0$  (η οποία είναι γειτονική με όλες τις κορυφές του  $\mathcal{Q}$ ) βρίσκεται στο εσωτερικό του τριγώνου που σχηματίζεται από τις κορυφές  $v_b, v_c$  και το σημείο τομής, έστω  $w$ , των ακμών  $(v_a, v_b)$  και  $(v_c, v_d)$ , όπως στην Εικόνα 13.8i. Σε αυτή την περίπτωση, οι ακμές  $(v_a, v_0)$  και  $(v_a, v_b)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_a$ , η οποία τέμνεται από την ακμή  $(v_c, v_d)$ . Αυτό δεν είναι αποδεκτό εξαιτίας της RAC-Ιδιότητας 2.



**Εικόνα 13.8:** Ο σχηματισμός που χρησιμοποιείται για την απόδειξη του Λήμματος 13.2.3: Το τετράπλευρο  $\mathcal{Q}$  δεν είναι επίπεδο. (i) η κορυφή  $v_0$  δεν μπορεί να βρίσκεται στο εσωτερικό ενός εκ των δύο τριγώνων, (ii) η κορυφή  $v_0$  δεν μπορεί να βρίσκεται στο εσωτερικό της  $R_{v_a, v_b, v_c}$  και (iii) η κορυφή  $v_0$  δεν μπορεί να βρίσκεται στις ανοιχτές περιοχές που είναι σκιασμένες με ανοιχτό-γκρι.

**Περίπτωση ii:** Η κορυφή  $v_0$  βρίσκεται στο εξωτερικό των δύο τριγώνων. Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι η ακμή  $(v_b, v_c)$  είναι οριζόντια και η ακμή  $(v_a, v_d)$  απεικονίζεται από “πάνω” της (βλ. Εικόνα 13.8i). Τα δύο τρίγωνα απεικονίζονται με σκούρο-γκρι χρώμα.

Έστω  $R_{v_a, v_b, v_c}$  η ανοιχτή περιοχή που απεικονίζεται με ανοιχτό-γκρι χρώμα και ορίζεται από τη γωνία  $\widehat{v_a v_b v_c}$ , η οποία δε συμπεριλαμβάνει την περιοχή του τριγώνου  $\Delta v_b v_c$  (βλ. Εικόνα 13.8ii). Αρχικά, θα αποδειχθεί ότι η κορυφή  $v_0$  δεν μπορεί να βρίσκεται στην περιοχή  $R_{v_a, v_b, v_c}$ . Για να δειχθεί αυτό, παρατηρούμε ότι εάν η κορυφή  $v_0$  βρίσκεται στο εσωτερικό της περιοχής  $R_{v_a, v_b, v_c}$ , τότε οι ακμές  $(v_0, v_b)$  και  $(v_0, v_d)$  θα σχηματίζουν μία έλικα με κέντρο



την κορυφή  $v_0$  που τέμνεται από την ακμή  $(v_c, v_d)$ , το οποίο αντιβαίνει στη RAC-Ιδιότητα 2. Με τον ίδιο τρόπο, μπορούν να οριστούν οι περιοχές  $R_{v_b, v_c, v_d}$ ,  $R_{v_c, v_d, v_a}$  και  $R_{v_d, v_a, v_b}$  και να δειχθεί ότι η κορυφή  $v_0$  δεν μπορεί να βρίσκεται σε καμία από αυτές. Επομένως, απομένει να εξεταστεί η περίπτωση όπου η κορυφή  $v_0$  βρίσκεται είτε στην άσπρη-σκιασμένη περιοχή “κάτω” από την ακμή  $(v_b, v_c)$  ή στην άσπρη-σκιασμένη περιοχή “πάνω” από την ακμή  $(v_a, v_d)$ . Βλέπε Εικόνα 13.8iii. Θα εξετάσουμε μόνο την πρώτη περίπτωση, καθώς η δεύτερη αντιμετωπίζεται παρόμοια υποθέτοντας ότι η  $(v_a, v_d)$  είναι η οριζόντια ακμή.

Επομένως, θεωρούμε ότι η κορυφή  $v_0$  βρίσκεται στο εσωτερικό της άσπρης-σκιασμένης περιοχής της Εικόνας 13.8iii “κάτω” από την ακμή  $(v_b, v_c)$ . Με βάση αυτή την υπόθεση, μπορούμε να διατυπώσουμε τις παρακάτω προτάσεις σχετικά με την τοποθέτηση των υπολοίπων κορυφών σε μία RAC απεικόνιση του ASA γραφήματος.

**Πρόταση 13.2.1.** Η ακμή  $(v_b, v_c)$  θα πρέπει να βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ .

*Απόδειξη.* Βλέπε Εικόνα 13.9i. Έστω ότι η ακμή  $(v_b, v_c)$  δε βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ . Τότε, μία από τις ακμές του τριγώνου που είναι γειτονικές της κορυφής  $v_0$  θα τέμνει μία έλικα που σχηματίζεται από τις ακμές του  $\mathcal{Q}$  με κέντρο είτε την κορυφή  $v_b$  ή την κορυφή  $v_c$ , το οποίο αντιβαίνει στη RAC-Ιδιότητα 2.  $\square$

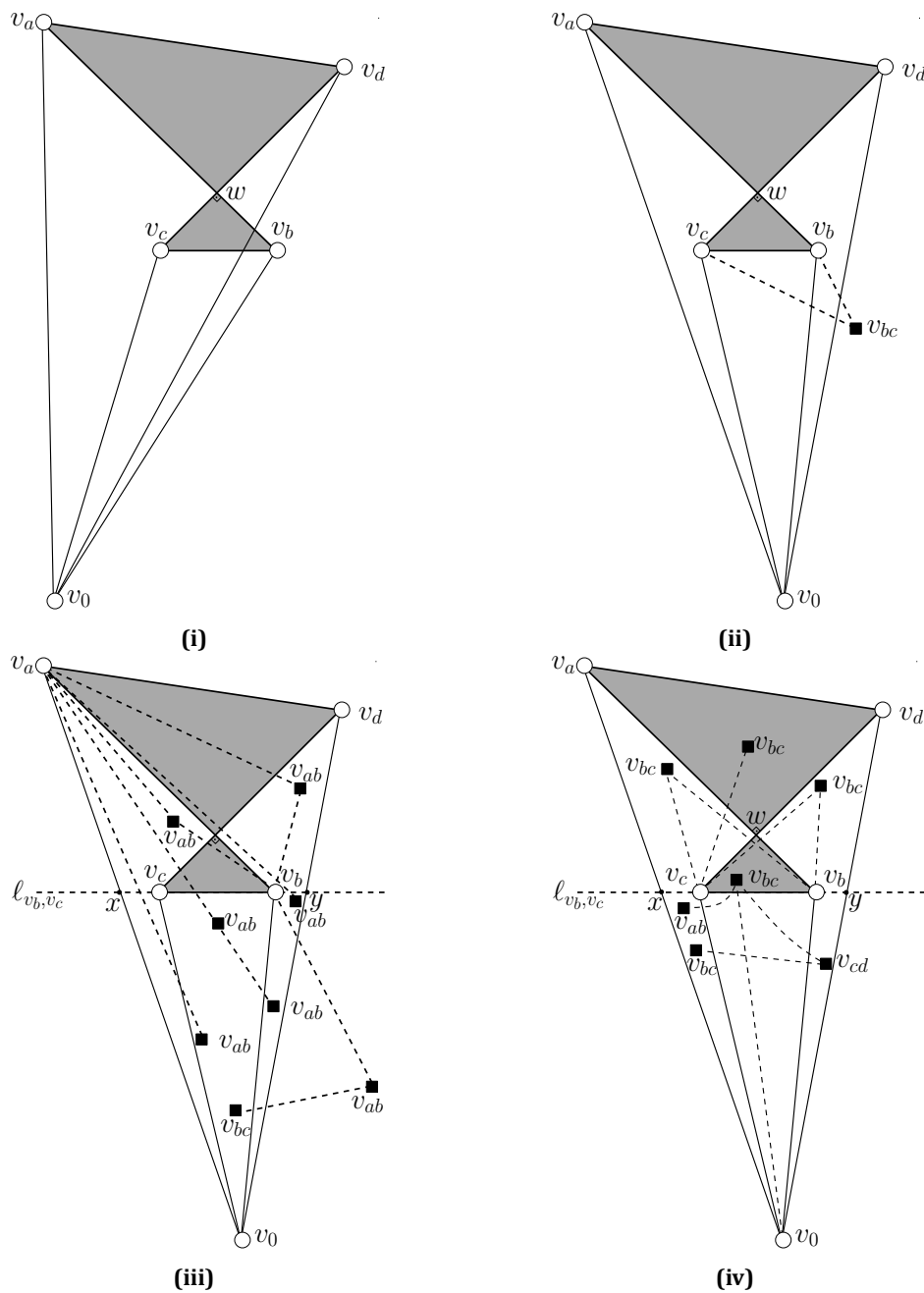
**Πρόταση 13.2.2.** Η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ .

*Απόδειξη.* Βλέπε Εικόνα 13.9ii. Η κορυφή  $v_{bc}$  ενώνεται με τις κορυφές  $v_b$  και  $v_c$ , οι οποίες από την Πρόταση 13.2.1 βρίσκονται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ . Από τη RAC-Ιδιότητα 3, έπεται ότι η κορυφή  $v_{bc}$  πρέπει, επίσης, να βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ .  $\square$

**Πρόταση 13.2.3.** Η κορυφή  $v_{ab}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_c x$ , όπου  $x$  είναι το σημείο τομής της ευθείας  $\ell_{v_b, v_c}$  με την ακμή  $(v_0, v_a)$ .

*Απόδειξη.* Βλέπε Εικόνα 13.9iii. Για την απόδειξη αυτής της πρότασης, θα δείξουμε ότι η κορυφή  $v_{ab}$  ( $\alpha'$ ) δεν μπορεί να βρίσκεται εξωτερικά του τριγώνου  $\Delta v_a v_0 v_d$ , ( $\beta'$ ) πρέπει να βρίσκεται “κάτω” από την ευθεία  $\ell_{v_b, v_c}$  και δεν μπορεί να βρίσκεται στο εσωτερικό ( $\gamma'$ ) του τριγώνου  $\Delta v_0 v_b y$  ή ( $\delta'$ ) του τριγώνου  $\Delta v_0 v_b v_c$ .

( $\alpha'$ ) Η κορυφή  $v_{ab}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ : Η κορυφή  $v_{ab}$  ενώνεται με τις κορυφές  $v_b$  και  $v_{bc}$ , οι οποίες από τις Προτάσεις 13.2.1 και 13.2.2, αντίστοιχα, βρίσκονται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ . Αυτό εξασφαλίζει ότι και η κορυφή  $v_{ab}$  θα βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_a v_0 v_d$ . Διαφορετικά, παραβιάζεται η RAC-Ιδιότητα 3.



**Εικόνα 13.9:** Σχηματισμοί που χρησιμοποιούνται για την απόδειξη του Λήμματος 13.2.3 (Προτάσεις 13.2.1-13.2.5): (i) Η ακμή  $(v_b, v_c)$  δε βρίσκεται στο εσωτερικό του τριγώνου  $\triangle v_a v_0 v_d$ , (ii) Η κορυφή  $v_{bc}$  δε βρίσκεται στο εσωτερικό του τριγώνου  $\triangle v_a v_0 v_d$ , (iii) Δυνατές τοποθετήσεις της κορυφής  $v_{ab}$ , (iv) Δυνατές τοποθετήσεις της κορυφής  $v_{bc}$

- (β') Η κορυφή  $v_{ab}$  πρέπει να βρίσκεται "κάτω" από την ευθεία  $\ell_{v_b, v_c}$ : Έστω ότι  $y$  είναι το σημείο τομής της ευθείας  $\ell_{v_b, v_c}$  με την ακμή  $(v_0, v_d)$ . Υποθέτουμε τώρα ότι η κορυφή  $v_{ab}$  βρίσκεται "πάνω" από την ευθεία  $\ell_{v_b, v_c}$ . Θεωρούμε τις ακόλουθες περιπτώσεις. Αρχικά υποθέτουμε ότι η κορυφή  $v_{ab}$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $v_a x v_c v_d$ . Τότε, οι ακμές  $(v_a, v_b)$  και  $(v_b, v_{ab})$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_b$ , η οποία τέμνεται από την ακμή  $(v_c, v_d)$ . Στην περίπτωση που η κορυφή  $v_{ab}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_c y v_d$ , οι ακμές  $(v_a, v_b)$  και  $(v_a, v_{ab})$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_a$ , η οποία τέμνεται από την ακμή  $(v_c, v_d)$ . Εφόσον και οι δύο περιπτώσεις οδηγούν σε άτοπο εξαιτίας της RAC-Ιδιότητας 2, καταλήγουμε ότι η κορυφή  $v_{ab}$  πρέπει να βρίσκεται "κάτω" από την ευθεία  $\ell_{v_b, v_c}$ .
- (γ') Η κορυφή  $v_{ab}$  δεν μπορεί να βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b y$ : Αυτό συνεπάγεται από το γεγονός ότι η κορυφή  $v_{ab}$  ενώνεται με την κορυφή  $v_a$ . Εάν η ακμή  $(v_{ab}, v_a)$  διέρχεται "αριστερά" της κορυφής  $v_b$ , θα πρέπει να εισέλθει και εξέλθει από το τρίγωνο  $\Delta v_0 v_b v_c$ , σχηματίζοντας τρεις ακμές οι οποίες τέμνονται ανά δύο μεταξύ τους. Εάν η ακμή  $(v_{ab}, v_a)$  διέρχεται "δεξιά" της κορυφής  $v_b$ , τότε οι ακμές  $(v_a, v_{ab})$  και  $(v_a, v_b)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_a$ , η οποία τέμνεται από την ακμή  $(v_c, v_d)$ . Εφόσον και οι δύο περιπτώσεις οδηγούν σε άτοπο (εξαιτίας των RAC-Ιδιοτήτων 1 και 2, αντίστοιχα), καταλήγουμε ότι η κορυφή  $v_{ab}$  δεν μπορεί να βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b y$ .
- (δ') Η κορυφή  $v_{ab}$  δεν μπορεί να βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b v_c$ : Αυτό συνεπάγεται και πάλι, από το γεγονός ότι η κορυφή  $v_{ab}$  ενώνεται με την κορυφή  $v_a$ . Εάν η ακμή  $(v_a, v_{ab})$  διέρχεται "δεξιά" της κορυφής  $v_c$ , θα πρέπει να εισέλθει και εξέλθει από το τρίγωνο  $\Delta v_b v_c w$ , σχηματίζοντας τρεις ακμές οι οποίες τέμνονται ανά δύο μεταξύ τους. Ωστόσο, αυτό παραβιάζει τη RAC-Ιδιότητα 1. Εάν η ακμή  $(v_a, v_{ab})$  διέρχεται "αριστερά" της κορυφής  $v_c$ , θα πρέπει να τμήσει υπό ορθή γωνία την ακμή  $(v_c, v_0)$ . Αλλά, δεδομένου ότι η ακμή  $(v_a, v_b)$  είναι, επίσης, κάθετη στην ακμή  $(v_c, v_d)$ , το τετράπλευρο  $v_a v_d v_c v_0$  πρέπει να είναι κυρτό, και επομένως, η κορυφή  $v_0$  θα πρέπει να βρίσκεται "πάνω" από την ευθεία  $\ell_{v_b, v_c}$ , που οδηγεί σε άτοπο.

□

**Πρόταση 13.2.4.** Η κορυφή  $v_{cd}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b y$ .

*Απόδειξη.* Έπεται χρησιμοποιώντας συμμετρικά επιχειρήματα με αυτά της απόδειξης της Πρότασης 13.2.3. □

**Πρόταση 13.2.5.** Η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 y v_c$ .

*Απόδειξη.* Υπενθυμίζεται ότι από την Πρόταση 13.2.2, η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_a v_d$ . Για να αποδείξουμε ότι η κορυφή  $v_{bc}$  βρίσκεται στο εσω-

τερικό του τριγώνου  $\Delta v_0 y v_c$ , θα οδηγήσουμε σε άτοπο τις περιπτώσεις στις οποίες η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό των τετράπλευρων (α')  $v_a w v_c x$  και (β')  $v_d w v_b y$ , καθώς και στο εσωτερικό των τριγώνων (γ')  $\Delta v_a w v_d$ , (δ')  $\Delta v_b w v_c$  και (ε')  $\Delta x v_c v_0$ . Βλέπε Εικόνα 13.9iv.

- (α') Η κορυφή  $v_{bc}$  δε βρίσκεται στο εσωτερικό του τετραπλεύρου  $v_a w v_c x$ : Εάν ισχύει αυτό, τότε οι ακμές  $(v_a, v_b)$  και  $(v_b, v_{bc})$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_b$ , η οποία τέμνεται από την ακμή  $(v_c, v_d)$ .
- (β') Η κορυφή  $v_{bc}$  δε βρίσκεται στο εσωτερικό του τετραπλεύρου  $v_d w v_b y$ : Εάν ισχύει αυτό, τότε οι ακμές  $(v_c, v_d)$  και  $(v_c, v_{bc})$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_c$ , η οποία τέμνεται από την ακμή  $(v_a, v_b)$ .
- (γ') Η κορυφή  $v_{bc}$  δε βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_a w v_d$ : Εάν ισχύει αυτό, τότε οι ακμές  $(v_c, v_d)$  και  $(v_c, v_{bc})$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_c$ , η οποία τέμνεται από την ακμή  $(v_a, v_b)$ .
- (δ') Η κορυφή  $v_{bc}$  δε βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_b w v_c$ : Εάν ισχύει αυτό, τότε θα έχει τρεις γειτονικές κορυφές, τις  $v_0, v_{ab}$  και  $v_{cd}$  εξωτερικά του τριγώνου  $\Delta v_b w v_c$ . Επομένως, θα πρέπει να ενωθεί με αυτές τις κορυφές με ακμές που εξέρχονται από διαφορετικές πλευρές του τριγώνου. Δοθέντος ότι η κορυφή  $v_0$  βρίσκεται "κάτω" από την ευθεία  $\ell_{v_b, v_c}$ , έπεται ότι τουλάχιστον μία εκ των κορυφών  $v_{ab}$  και  $v_{cd}$  θα πρέπει να βρίσκεται "πάνω" από την ευθεία  $\ell_{v_b, v_c}$ , το οποίο οδηγεί σε άτοπο.
- (ε') Η κορυφή  $v_{bc}$  δε βρίσκεται στο εσωτερικό του τριγώνου  $\Delta x v_c v_0$ : Υπενθυμίζεται ότι λόγω της Πρότασης 13.2.4, η κορυφή  $v_{cd}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b y$ . Ωστόσο, η κορυφή  $v_{bc}$  ενώνεται με την κορυφή  $v_{cd}$ . Αυτό συνεπάγεται ότι στην περίπτωση που η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta x v_c v_0$ , τότε η ακμή  $(v_{bc}, v_{cd})$  θα πρέπει να εισέλθει και να εξέλθει από το τρίγωνο  $\Delta v_0 v_b v_c$ , σχηματίζοντας τρεις ακμές οι οποίες τέμνονται ανά δύο μεταξύ τους.

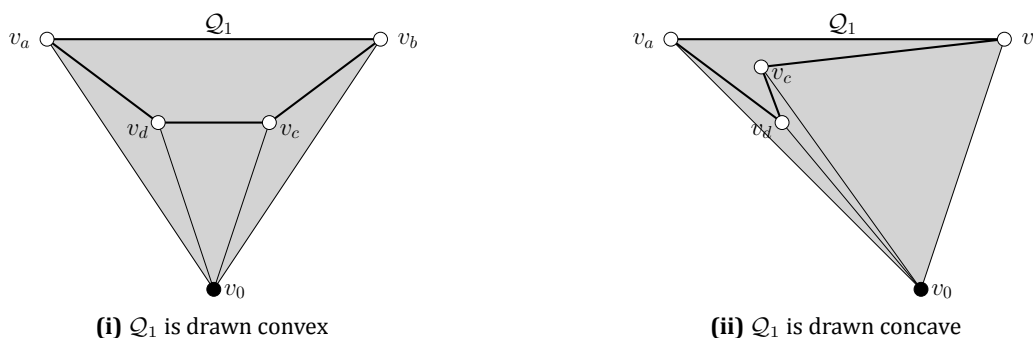
□

Δεδομένου ότι η κορυφή  $v_{ab}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_c x$  (Πρόταση 13.2.3) και η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 y v_c$  (Πρόταση 13.2.5), οι ακμές  $(v_{ab}, v_{bc})$  και  $(v_{ab}, v_b)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_{ab}$ , η οποία τέμνεται από την ακμή  $(v_0, v_c)$ . Αυτό είναι αδύνατο εξαιτίας της RAC-Ιδιότητας 2. Επομένως, θεωρώντας μία νόμιμη RAC απεικόνιση του ASA γραφήματος, όπου η κορυφή  $v_0$  βρίσκεται εξωτερικά των δύο τριγώνων, καταλήξαμε ότι δεν είναι δυνατόν να βρεθεί μία νόμιμη τοποθέτηση για την κορυφή  $v_{bc}$ , το οποίο οδηγεί σε άτοπο. Αυτό ολοκληρώνει την απόδειξη της Περίπτωσης ii του παρόντος λήμματος.

□

**Λήμμα 13.2.4.** Σε κάθε RAC απεικόνιση του ASA γραφήματος, η κεντρική κορυφή  $v_0$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_i$ ,  $i = 1, 2$ .

*Απόδειξη.* Από το Λήμμα 13.2.3 έπεται ότι το τετράπλευρο  $Q_i$  πρέπει να απεικονιστεί επίπεδο, για κάθε  $i = 1, 2$ . Για την απόδειξη αυτού του λήμματος, υποθέτουμε ότι η κεντρική κορυφή  $v_0$  βρίσκεται εξωτερικά των δύο τετράπλευρων, έστω χωρίς βλάβη της γενικότητας, εξωτερικά του τετράπλευρου  $Q_1$ . Έστω  $v_a, v_b, v_c$  και  $v_d$  οι κορυφές του  $Q_1$ , διαδοχικές κατά μήκος του τετράπλευρου  $Q_1$ . Από το Λήμμα 13.2.2 έπεται ότι η κορυφή  $v_0$  δεν μπορεί να συνεισφέρει επιπλέον τομές στο τετράπλευρο  $Q_1$ . Αυτό σημαίνει ότι η απεικόνιση του γραφήματος που παράγεται από το τετράπλευρο  $Q_1$  και την κορυφή  $v_0$  θα είναι επίπεδη και θα μοιάζει με αυτή της Εικόνας 13.10. Συμβολίζουμε με  $T_{Q_1}$  το τρίγωνο που σχηματίζεται από την κορυφή  $v_0$  και τις δύο κορυφές, που βρίσκονται στο κυρτό περίβλημα του  $Q_1 \cup v_0$  (βλ. τα γκρι-σκιασμένα τρίγωνα της Εικόνας 13.10).



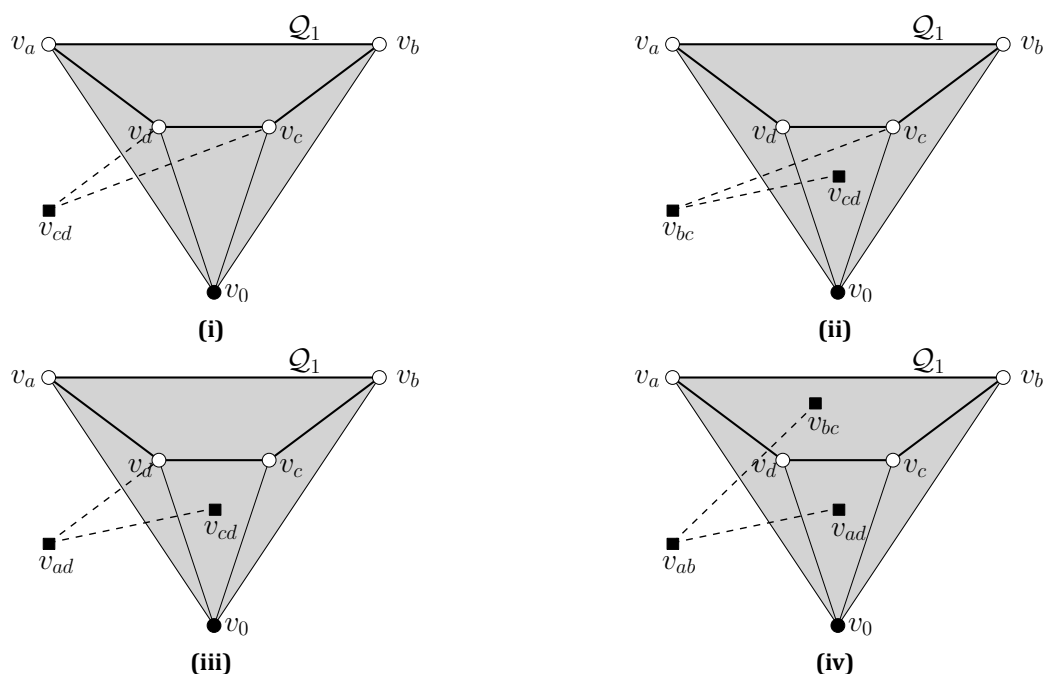
**Εικόνα 13.10:** Σχηματισμοί που χρησιμοποιούνται στην απόδειξη του Λήμματος 13.2.4: Διαφορετικές απεικονίσεις του γραφήματος drawings που παράγεται από το τετράπλευρο  $Q_1$  και την κορυφή  $v_0$ .

**Πρόταση 13.2.6.** Καμία κορυφή του  $Q_2$  δε βρίσκεται εξωτερικά του  $T_{Q_1}$ .

*Απόδειξη.* Βλέπε Εικόνα 13.10. Υποθέτουμε χωρίς βλάβη της γενικότητας ότι το τρίγωνο  $T_{Q_1}$  ορίζεται από τις κορυφές  $v_0, v_a$  και  $v_b$ , δηλαδή οι κορυφές  $v_a$  και  $v_b$  ανήκουν στο κυρτό περίβλημα του  $Q_1 \cup v_0$ . Ακολούθως, θα αποδείξουμε ότι καθεμία από τις κορυφές του  $Q_2$  θα πρέπει να βρίσκεται στο εσωτερικό του τριγώνου  $T_{Q_1}$ .

(α') Η κορυφή  $v_{cd}$  βρίσκεται στο εσωτερικό του τριγώνου  $T_{Q_1}$ : Εάν δεν ισχύει αυτό, η RAC-Ιδιότητα 3 παραβιάζεται (βλ. Εικόνα 13.11i), καθώς η κορυφή  $v_{cd}$  ενώνεται με τις  $v_c$  και  $v_d$ , οι οποίες βρίσκονται και οι δύο στο εσωτερικό του τριγώνου  $T_{Q_1}$ .

- (β') Η κορυφή  $v_{bc}$  βρίσκεται στο εσωτερικό του τριγώνου  $T_{Q_1}$ : Εάν δεν ισχύει αυτό, η RAC-Ιδιότητα 3 παραβιάζεται (βλ. Εικόνα 13.11ii), καθώς η κορυφή  $v_{bc}$  ενώνεται με τις  $v_c$  και  $v_{cd}$ , οι οποίες βρίσκονται και οι δύο στο εσωτερικό του τριγώνου  $T_{Q_1}$ .
- (γ') Η κορυφή  $v_{ad}$  βρίσκεται στο εσωτερικό του τριγώνου  $T_{Q_1}$ : Εάν δεν ισχύει αυτό, η RAC-Ιδιότητα 3 παραβιάζεται (βλ. Εικόνα 13.11iii), καθώς η κορυφή  $v_{ad}$  ενώνεται με τις  $v_d$  και  $v_{cd}$ , οι οποίες βρίσκονται και οι δύο στο εσωτερικό του τριγώνου  $T_{Q_1}$ .
- (δ') Η κορυφή  $v_{ab}$  βρίσκεται στο εσωτερικό του τριγώνου  $T_{Q_1}$ : Εάν δεν ισχύει αυτό, η RAC-Ιδιότητα 3 παραβιάζεται (βλ. Εικόνα 13.11iv), καθώς η κορυφή  $v_{ab}$  ενώνεται με τις  $v_{ad}$  και  $v_{bc}$ , οι οποίες βρίσκονται και οι δύο στο εσωτερικό του τριγώνου  $T_{Q_1}$ .



**Εικόνα 13.11:** Σχηματισμοί που χρησιμοποιούνται στην απόδειξη του Λήμματος 13.2.4 (Πρόταση 13.2.6): Όλες οι κορυφές του  $Q_2$  βρίσκονται στο εσωτερικό του τριγώνου  $T_{Q_1}$

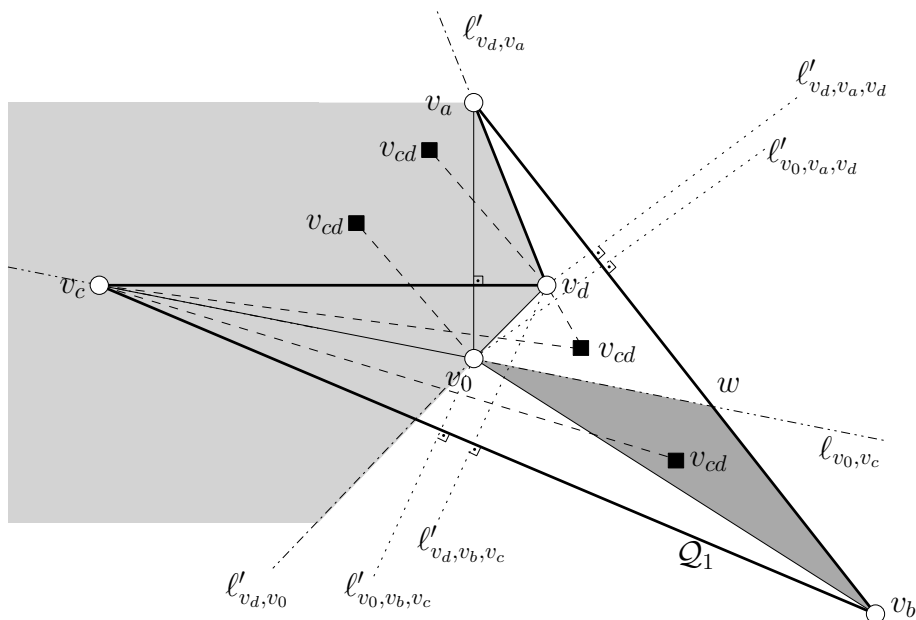
□

Εφόσον το τετράπλευρο  $Q_2$  βρίσκεται στο εσωτερικό του τριγώνου  $T_{Q_1}$ , η κορυφή  $v_0$  είναι, επίσης, εξωτερικά και του τετραπλεύρου  $Q_2$ . Επομένως, με επιχειρήματα παρόμοια με αυτά της Πρότασης 13.2.6, μπορεί να αποδειχθεί ότι το τετράπλευρο  $Q_1$  βρίσκεται εσωτερικά του τριγώνου  $T_{Q_2}$ , όπου το τρίγωνο  $T_{Q_2}$  ορίζεται από την κορυφή  $v_0$  και τις δύο κορυφές, οι οποίες βρίσκονται στο κυρτό περίβλημα του  $Q_2 \cup v_0$ . Ωστόσο, αυτό οδηγεί σε άτοπο,

καθώς δύο τρίγωνα (σ.σ., τα τρίγωνα  $T_{Q_1}$  και  $T_{Q_2}$ ) δεν μπορεί να περικλείονται το ένα μέσα στο άλλο χωρίς να ταυτίζονται και επομένως, χωρίς να εισάγουν επικαλύψεις κορυφών και ακμών. □

**Λήμμα 13.2.5.** Σε κάθε RAC απεικόνιση ενός ASA γραφήματος, όλες οι ακμές από την κεντρική κορυφή  $v_0$  προς το τετράπλευρο  $Q_i$  περιέχονται πλήρως στο τετράπλευρο  $Q_i, i = 1, 2$ .

*Απόδειξη.* Από το Λήμμα 13.2.4 έπεται ότι η κορυφή  $v_0$  θα πρέπει να βρίσκεται στο εσωτερικό του τετράπλευρου  $Q_i, i = 1, 2$ , το οποίο απεικονίζεται επίπεδο εξαιτίας του Λήμματος 13.2.3. Εάν τα δύο τετράπλευρα  $Q_1$  και  $Q_2$  απεικονίζονται κυρτά, τότε το λήμμα τετριμμένα ισχύει. Έστω τώρα, ότι το  $Q_1$  απεικονίζεται μη-κυρτό και μία ακμή που ξεκινάει από την κορυφή  $v_0$  προς μία κορυφή του  $Q_1$ , έστω την κορυφή  $v_a$ , τέμνει μία ακμή, έστω την ακμή  $(v_c, v_d)$ , του τετραπλεύρου  $Q_1$  (βλ. Εικόνα 13.12).



**Εικόνα 13.12:** Σχηματισμός που χρησιμοποιείται στην απόδειξη του Λήμματος 13.2.5 (Πρόταση 13.2.8): Δυνατές τοποθετήσεις της κορυφής  $v_{cd}$ .

**Πρόταση 13.2.7.** Το άθροισμα των τριών οξείων γωνιών του τετραπλεύρου  $Q_1$  είναι μικρότερο από  $\frac{\pi}{2}$ .

*Απόδειξη.* Καθώς το τετράπλευρο  $Q_1$  είναι μη-κυρτό στην κορυφή  $v_d$ , η εσωτερική γωνία στην κορυφή  $v_d$  είναι μεγαλύτερη από  $3\pi/2$ . Επομένως, το άθροισμα των υπολοίπων τριών γωνιών είναι μικρότερο από  $\pi/2$ . □

**Πρόταση 13.2.8.** Η κορυφή  $v_{cd}$  πρέπει να βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ .

*Απόδειξη.* Υπενθυμίζεται ότι η κορυφή  $v_{cd}$  ενώνεται με τις κορυφές  $v_c, v_d$  και  $v_0$  και έστω ότι, η κορυφή  $v_{cd}$  δε βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ . Τότε, ισχύουν τα ακόλουθα (βλ. Εικόνα 13.12):

- i) Η ακμή  $(v_0, v_{cd})$  δεν εισέρχεται στο  $Q_1$  από την ακμή  $(v_c, v_d)$ : Εάν συνέβαινε αυτό, οι ακμές  $(v_0, v_{cd})$  και  $(v_0, v_a)$  θα σχημάτιζαν μία έλικα με κέντρο την κορυφή  $v_0$ , η οποία θα διασταυρωνόταν με την ακμή  $(v_c, v_d)$ , που αντιβαίνει στη RAC-Ιδιότητα 2.
- ii) Η ακμή  $(v_0, v_{cd})$  δεν εισέρχεται στο  $Q_1$  από την ακμή  $(v_a, v_d)$ : Εάν συνέβαινε αυτό, τότε θα ήταν δυνατόν να φέρουμε από την κορυφή  $v_0$  κάθετα ευθύγραμμα τμήματα (σ.σ.,  $(v_0, v_a)$  και  $(v_0, v_{cd})$ ) προς τις δύο ακμές που σχηματίζουν τη μη-κυρτή γωνία του τετραπλεύρου  $Q_1$ . Ωστόσο, αυτό δεν είναι εφικτό, καθώς η κορυφή  $v_0$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ .
- iii) Η ακμή  $(v_0, v_{cd})$  εισέρχεται στο  $Q_1$  είτε από την ακμή  $(v_a, v_b)$  είτε από την ακμή  $(v_b, v_c)$ : Τετριμμένα έπεται από τα (i) και (ii).
- iv) Η κορυφή  $v_{cd}$  δεν ανήκει στην περιοχή που σχηματίζεται από τις  $\ell'_{v_d, v_a}$  και  $\ell'_{v_d, v_0}$  και περιέχει την κορυφή  $v_c$ : (βλ. την περιοχή που έχει σκιαγραφηθεί με ανοιχτό-γκρι χρώμα στην Εικόνα. 13.12). Η απόδειξη της περίπτωσης αυτής έπεται από το γεγονός ότι η ακμή  $(v_d, v_{cd})$  δεν μπορεί να τμήσει την ακμή  $(v_0, v_a)$ , καθώς σε διαφορετική περίπτωση, οι ακμές  $(v_d, v_{cd})$  και  $(v_c, v_d)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v_d$ , η οποία τέμνεται από την ακμή  $(v_0, v_a)$ , παραβιάζοντας τη RAC-Ιδιότητα 2.
- v) Η ακμή  $(v_d, v_{cd})$  εισέρχεται στο  $Q_1$  είτε από την ακμή  $(v_a, v_b)$  είτε από την ακμή  $(v_b, v_c)$ : Τετριμμένα έπεται από το (iv).

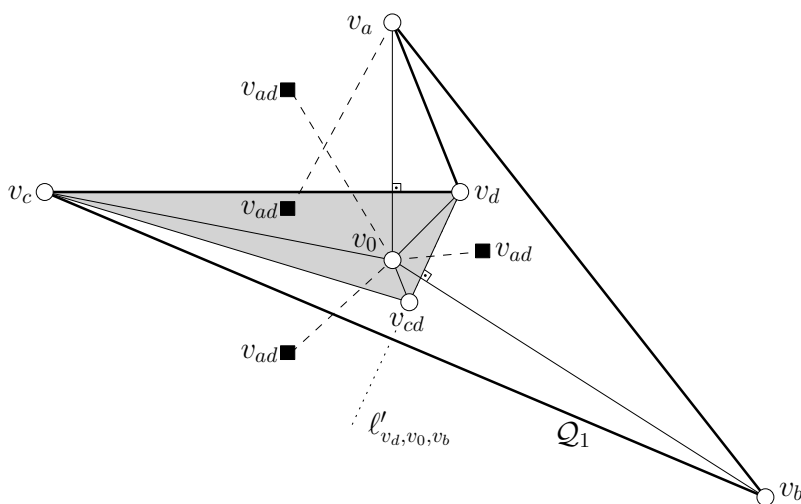
Από τα (iii) και (v), έπεται ότι η κορυφή  $v_{cd}$  θα πρέπει να βρίσκεται στην τομή των δύο ημιευθειών (βλ. τις ημιευθείες που απεικονίζονται διάστικτες (dotted) στην Εικόνα 13.12), οι οποίες είναι κάθετες στις ακμές  $(v_a, v_b)$  και/ή  $(v_b, v_c)$ . Ωστόσο, αυτό οδηγεί σε άτοπο, καθώς αυτές οι ημιευθείες δεν τέμνονται λόγω της Πρότασης 13.2.7, ως κάθετες σε δύο διαδοχικές ακμές του  $Q_1$  οι οποίες όμως, σχηματίζουν οξεία γωνία.  $\square$

**Πρόταση 13.2.9.** Η κορυφή  $v_{cd}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b v_c$ .

*Απόδειξη.* Σύμφωνα με την Πρόταση 13.2.8, αρκεί να δείξουμε ότι η κορυφή  $v_{cd}$  βρίσκεται "κάτω" από την ευθεία  $\ell_{v_0, v_c}$  και εξωτερικά του τριγώνου  $\Delta v_0 v_b v_c$  (βλ. το σκούρο-



γκρι σκιασμένο τρίγωνο της Εικόνας 13.12), όπου το  $w$  είναι το σημείο τομής της ευθείας  $\ell_{v_0, v_c}$  και της ακμής  $(v_a, v_b)$ . Η πρώτη ιδιότητα είναι προφανής, καθώς η κορυφή  $v_{cd}$  ενώνεται και με τις δύο κορυφές  $v_c$  και  $v_d$ . Ως εκ τούτου, εάν η κορυφή  $v_{cd}$  ήταν “πάνω” από την ευθεία  $\ell_{v_0, v_c}$ , τότε η ακμή  $(v_c, v_d)$  και είτε η ακμή  $(v_c, v_{cd})$  είτε η ακμή  $(v_d, v_{cd})$  θα σχημάτιζαν μία έλικα με κέντρο είτε την κορυφή  $v_c$  είτε την κορυφή  $v_d$ , αντίστοιχα, η οποία τέμνεται από την ακμή  $(v_0, v_a)$ . Εάν η κορυφή  $v_{cd}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b w$ , τότε η ακμή  $(v_c, v_{cd})$  θα πρέπει να τέμνει την ακμή  $(v_0, v_b)$  υπό ορθή γωνία. Ωστόσο, αυτό δεν είναι εφικτό, καθώς η κάθετη ευθεία από την κορυφή  $v_c$  προς την ακμή  $(v_0, v_b)$  είναι εξωτερικά του τριγώνου  $\Delta v_0 v_b w$ , λόγω της Πρότασης 13.2.7. Επομένως, η κορυφή  $v_{cd}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b v_c$  (και κατά μήκος της ημιευθείας  $\ell'_{v_d, v_0, v_b}$ ), όπως περιγράφεται στην Εικόνα 13.13).  $\square$



**Εικόνα 13.13:** Ο σχηματισμός που χρησιμοποιείται για την απόδειξη του Λήμματος 13.2.5 (Προτάσεις 13.2.10-13.2.12): Δυνατές τοποθετήσεις της κορυφής  $v_{ad}$ .

**Πρόταση 13.2.10.** Η κορυφή  $v_{ad}$  βρίσκεται εξωτερικά του τριγώνου  $\Delta v_c v_d v_{cd}$ .

*Απόδειξη.* Στην Εικόνα 13.13, το τρίγωνο  $\Delta v_c v_d v_{cd}$  είναι σκιασμένο με γκρι χρώμα. Έστω ότι η κορυφή  $v_{ad}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_c v_d v_{cd}$ . Τότε, δεδομένου ότι η κορυφή  $v_0$  είναι επίσης εσωτερική, η κορυφή  $v_a$  η οποία βρίσκεται εξωτερικά του τριγώνου  $\Delta v_c v_d v_{cd}$ , θα πρέπει να ενωθεί με τις κορυφές  $v_0$  και  $v_{ad}$ , οι οποίες βρίσκονται και οι δύο στο εσωτερικό του τριγώνου  $\Delta v_c v_d v_{cd}$ . Αυτό οδηγεί σε άτοπο, λόγω της RAC-Ιδιότητας 3.  $\square$

**Πρόταση 13.2.11.** Η ακμή  $(v_0, v_{ad})$  δεν μπορεί να εισέλθει στο τρίγωνο  $\Delta v_c v_d v_{cd}$  ούτε από την ακμή  $(v_c, v_d)$  ούτε από την ακμή  $(v_d, v_{cd})$ .

*Απόδειξη.* Έπεται από το γεγονός ότι, οι ακμές  $(v_c, v_d)$  και  $(v_d, v_{cd})$  τέμνονται ήδη υπό ορθές γωνίες από τις ακμές  $(v_0, v_a)$  και  $(v_0, v_b)$ , αντίστοιχα, οι οποίες είναι προσκείμενες στην κορυφή  $v_0$ , όπως και η ακμή  $(v_0, v_{ad})$ .  $\square$

**Πρόταση 13.2.12.** Η κορυφή  $v_{ad}$  βρίσκεται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b v_c$ .

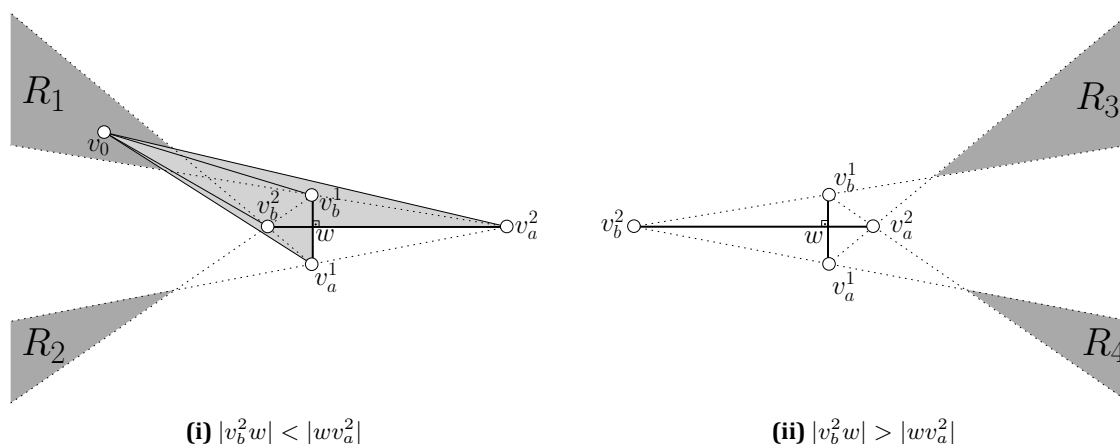
*Απόδειξη.* Από την Πρόταση 13.2.11, η ακμή  $(v_0, v_{ad})$  εισέρχεται στο τρίγωνο  $\Delta v_c v_d v_{cd}$  από την ακμή  $(v_c, v_{cd})$ . Εάν η κορυφή  $v_{ad}$  ήταν εξωτερική του τριγώνου  $\Delta v_0 v_b v_c$ , οι ακμές  $(v_c, v_{cd})$  και  $(v_b, v_c)$  θα σχημάτιζαν μία έλικα με κέντρο την κορυφή  $v_c$ , η οποία τέμνεται από την ακμή  $(v_0, v_{ad})$ , παραβιάζοντας τη RAC-Ιδιότητα 2.  $\square$

Οι Προτάσεις 13.2.9 και 13.2.12 υποδηλώνουν ότι και οι δύο κορυφές  $v_{cd}$  και  $v_{ad}$  βρίσκονται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b v_c$ . Βασιζόμενοι σε αυτό και ακολουθώντας παρόμοια συλλογιστική με αυτή της απόδειξης της Πρότασης 13.2.6, μπορούμε να δείξουμε ότι όλες οι κορυφές του τετραπλεύρου  $\mathcal{Q}_2$  θα πρέπει να βρίσκονται στο εσωτερικό του τριγώνου  $\Delta v_0 v_b v_c$ . Ωστόσο, αυτό οδηγεί σε άτοπο, καθώς η κορυφή  $v_0$  θα πρέπει να βρίσκεται στο εσωτερικό του τετραπλεύρου  $\mathcal{Q}_2$  λόγω του Λήμματος 13.2.4.  $\square$

**Λήμμα 13.2.6.** Δεν υπάρχει RAC απεικόνιση του ASA γραφήματος στην οποία τα τετράπλευρα  $\mathcal{Q}_1$  και  $\mathcal{Q}_2$  τέμνονται.

*Απόδειξη.* Έστω ότι τα τετράπλευρα  $\mathcal{Q}_1$  και  $\mathcal{Q}_2$  τέμνονται. Παραβιάζοντας ελαφρώς τη σημειολογία, έστω  $v_a^i, v_b^i, v_c^i$  και  $v_d^i$  οι κορυφές του τετραπλεύρου  $\mathcal{Q}_i$ , διαδοχικές κατά μήκος του, δηλαδή  $\{v_a, v_b, v_c, v_d\} = \{v_a^1, v_b^1, v_c^1, v_d^1\}$  και  $\{v_{ab}, v_{bc}, v_{cd}, v_{ad}\} = \{v_a^2, v_b^2, v_c^2, v_d^2\}$ . Έστω, χωρίς βλάβη της γενικότητας ότι,  $(v_a^1, v_b^1) \in \mathcal{Q}_1$  και  $(v_a^2, v_b^2) \in \mathcal{Q}_2$  είναι ένα ζεύγος ακμών, οι οποίες εμπλέκονται στην τομή των τετράπλευρων  $\mathcal{Q}_1$  και  $\mathcal{Q}_2$  και έστω  $w$  το σημείο τομής τους. Θεωρούμε μία RAC απεικόνιση του ASA γραφήματος στην οποία η ακμή  $(v_a^2, v_b^2)$  απεικονίζεται οριζόντια (και επομένως, η ακμή  $(v_a^1, v_b^1)$  απεικονίζεται κάθετη) και υποθέτουμε ότι η κορυφή  $v_b^2$  βρίσκεται στα “αριστερά” της κορυφής  $v_a^2$ , ενώ η κορυφή  $v_b^1$  βρίσκεται “πάνω” από την κορυφή  $v_a^1$  (βλ. Εικόνα 13.14).

Οι ακμές  $(v_0, v_a^1)$  και  $(v_0, v_b^1)$  δεν μπορούν να τέμνουν την ακμή  $(v_a^2, v_b^2)$ , καθώς θα σχημάτιζαν έλικα με κέντρο τις κορυφές  $v_a^1$  και  $v_b^1$ , αντίστοιχα, η οποία θα τεμνόταν από την ακμή  $(v_a^2, v_b^2)$ , παραβιάζοντας τη RAC-Ιδιότητα 2. Όμοια, οι ακμές  $(v_0, v_a^2)$  και  $(v_0, v_b^2)$  δεν μπορούν να τέμνουν την ακμή  $(v_a^1, v_b^1)$ . Συνεπώς, η κορυφή  $v_0$  θα πρέπει να βρίσκεται στο εσωτερικό των σκούρων-γκρι σκιασμένων μη-φραγμένων περιοχών της Εικόνας 13.14. Υποθέτουμε χωρίς βλάβη της γενικότητας, ότι η κορυφή  $v_0$  βρίσκεται εσωτερικά της περιοχής  $R_1$  (οι εναπομείναντες περιπτώσεις αντιμετωπίζονται συμμετρικά). Από το Λήμμα 13.2.4, η κορυφή  $v_0$  είναι εσωτερική του τετραπλεύρου  $\mathcal{Q}_1$ . Ωστόσο, όπως θα δείξουμε, δεν υπάρχει νόμιμη



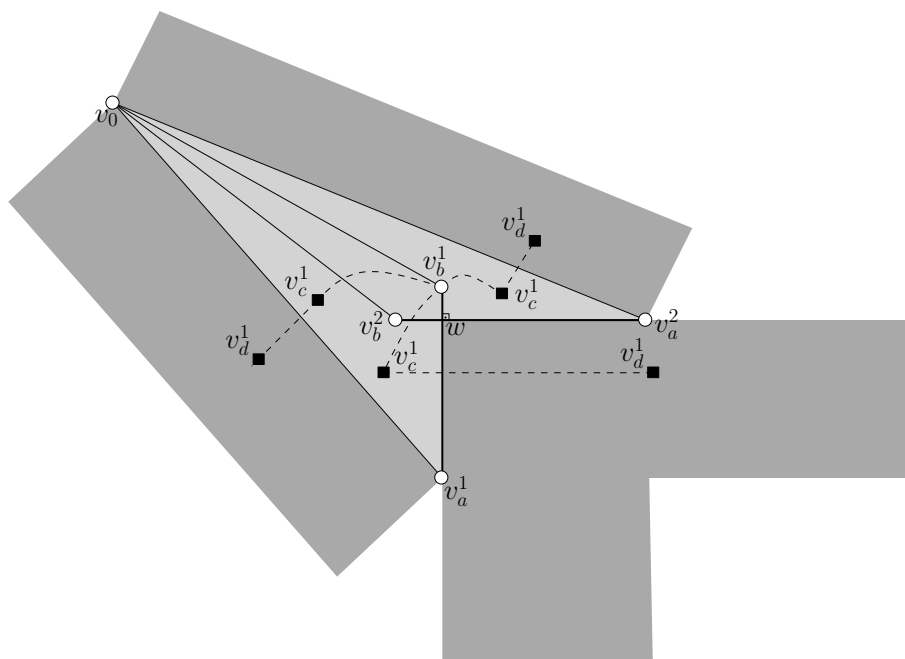
**Εικόνα 13.14:** Σχηματισμοί που χρησιμοποιούνται στην απόδειξη του Λήμματος 13.2.6: Η κορυφή  $v_0$  θα πρέπει να βρίσκεται στο εσωτερικό μιας εκ των περιοχών  $R_1, R_2, R_3$  και  $R_4$ .

τοποθέτηση των κορυφών του τετραπλεύρου  $\mathcal{Q}_2$ , τέτοια ώστε το τετράπλευρο  $\mathcal{Q}_1$  να περικλείει την κορυφή  $v_0$ . Ακολούθως, θα αποδείξουμε αρχικά, ότι η κορυφή  $v_c^1$  θα πρέπει να βρίσκεται εξωτερικά του τετραπλεύρου που είναι σκιασμένο με ανοιχτό-γκρι χρώμα στην Εικόνα 13.14i.

**Πρόταση 13.2.13.** Η κορυφή  $v_c^1$  βρίσκεται εξωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ , όπου το  $w$  είναι το σημείο τομής των ακμών  $(v_a^1, v_b^1) \in \mathcal{Q}_1$  και  $(v_a^2, v_b^2) \in \mathcal{Q}_2$ .

*Απόδειξη.* Βλέπε Εικόνα 13.15. Υπενθυμίζεται ότι, το τετράπλευρο  $\mathcal{Q}_1$  σχηματίζεται από τις κορυφές  $v_a^1, v_b^1, v_c^1$  και  $v_d^1$  με αυτή τη σειρά. Έστω ότι η κορυφή  $v_c^1$  βρίσκεται εσωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ . Εάν και η κορυφή  $v_d^1$  βρίσκεται, επίσης, εσωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ , τότε το τετράπλευρο  $\mathcal{Q}_1$  βρίσκεται ολόκληρο εσωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ . Επομένως, η κορυφή  $v_0$  δε βρίσκεται εσωτερικά του τετραπλεύρου  $\mathcal{Q}_1$ , το οποίο αντιβαίνει στο Λήμμα 13.2.4. Ως εκ τούτου, η κορυφή  $v_d^1$  βρίσκεται εξωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ . Δεδομένου ότι η κορυφή  $v_c^1$  είναι εσωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ , ενώ η κορυφή  $v_d^1$  εξωτερικά, η ακμή  $(v_c^1, v_d^1)$  θα πρέπει να είναι κάθετη σε μία ακμή του τετραπλεύρου  $wv_a^1v_0v_a^2$ . Αυτό υποδηλώνει ότι η κορυφή  $v_d^1$  θα πρέπει να βρίσκεται στη σκιασμένη με σκούρο-γκρι περιοχή της Εικόνας 13.15, το οποίο συνεπάγεται και πάλι ότι η κορυφή  $v_0$  δε βρίσκεται στο εσωτερικό του τετραπλεύρου  $\mathcal{Q}_1$ , καθώς η ακμή  $(v_d^1, v_a^1)$  σχηματίζει ένα τετράπλευρο που δεν περιέχει την κορυφή  $v_0$ . Αυτό οδηγεί σε άτοπο λόγω του Λήμματος 13.2.4. □

Η Πρόταση 13.2.13 και το Λήμμα 13.2.4 συνεπάγονται ότι το τετράπλευρο  $\mathcal{Q}_1$  θα πρέπει



**Εικόνα 13.15:** Ο σχηματισμός που χρησιμοποιείται για την απόδειξη του Λήμματος 13.2.6 (Πρόταση 13.2.13): Δυνατές τοποθετήσεις της κορυφής  $v_d^1$ , υποθέτοντας ότι η κορυφή  $v_0$  βρίσκεται στο εσωτερικό της περιοχής  $R_1$  και η κορυφή  $v_c^1$  στο εσωτερικό του τετραπλεύρου  $wv_a^1v_0v_a^2$ .

να απεικονιστεί όπως στην Εικόνα 13.16, δηλαδή η ακμή  $(v_b^1, v_c^1)$  θα πρέπει να είναι κάθετη στην ακμή  $(v_0, v_a^2)$ , ενώ η κορυφή  $v_d^1$  θα πρέπει να είναι στα “αριστερά” της κορυφής  $v_0$ , έτσι ώστε οι ακμές  $(v_c^1, v_d^1)$  και  $(v_a^1, v_d^1)$  να μην τέμνουν το τετράπλευρο  $wv_a^1v_0v_a^2$  και η κορυφή  $v_0$  να βρίσκεται εσωτερικά του  $\mathcal{Q}_1$ . Στη συνέχεια, θα εξετάσουμε τον τρόπο με τον οποίο απεικονίζεται το τετράπλευρο  $\mathcal{Q}_2$ .

**Πρόταση 13.2.14.** Η κορυφή  $v_c^2$  βρίσκεται εξωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ , όπου το  $w$  είναι το σημείο τομής των ακμών  $(v_a^1, v_b^1) \in \mathcal{Q}_1$  και  $(v_a^2, v_b^2) \in \mathcal{Q}_2$ .

Απόδειξη. Όμοια με την Πρόταση 13.2.13. □

**Πρόταση 13.2.15.** Η κορυφή  $v_c^2$  βρίσκεται εσωτερικά του τετραπλεύρου  $\mathcal{Q}_1$ .

Απόδειξη. Από την Πρόταση 13.2.14 έπεται ότι η κορυφή  $v_c^2$  βρίσκεται εξωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ . Εάν και η κορυφή  $v_c^2$  βρίσκεται εξωτερικά του τετραπλεύρου  $\mathcal{Q}_1$ , τότε οι ακμές  $(v_0, v_a^1)$  και  $(v_d^1, v_a^1)$  σχηματίζουν μία έλικα με κέντρο την κορυφή

$v_a^1$ , η οποία τέμνεται από την ακμή  $(v_b^2, v_c^2)$ , παραβιάζοντας τη RAC-Ιδιότητα 2 (βλ. Εικόνα 13.16). □

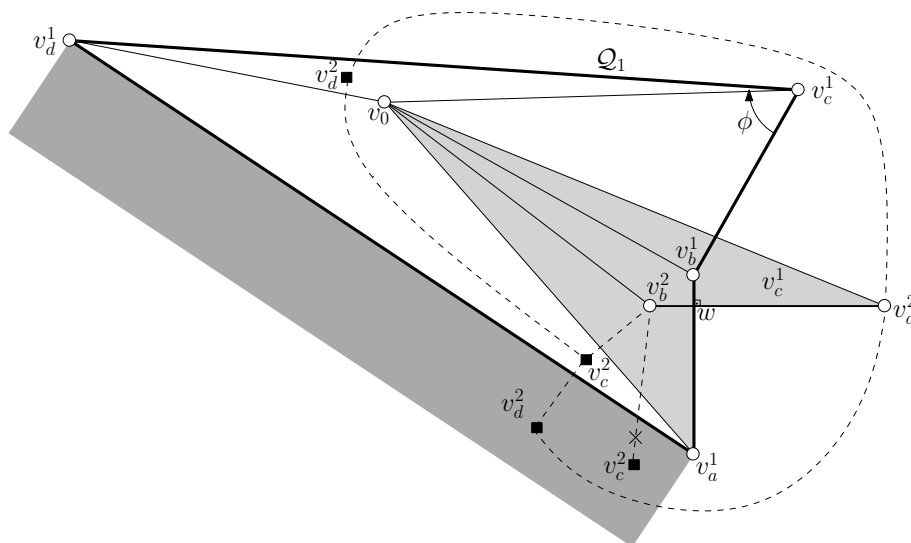
Από τις παραπάνω προτάσεις, έπεται ότι η κορυφή  $v_c^2$  βρίσκεται εξωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ , αλλά στο εσωτερικό του τετραπλεύρου  $Q_1$ . Ακολούθως, θα αποδείξουμε ότι η κορυφή  $v_d^2$  δεν μπορεί να βρίσκεται ούτε στο εσωτερικό του τετραπλεύρου  $Q_1$  ούτε στο εξωτερικό του, οδηγώντας σε άτοπο την αρχική μας υπόθεση ότι τα τετράπλευρα  $Q_1$  και  $Q_2$  τέμνονται (βλ. Εικόνα 13.16).

**Πρόταση 13.2.16.** *Η κορυφή  $v_d^2$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ .*

*Απόδειξη.* Έστω ότι η κορυφή  $v_d^2$  βρίσκεται εξωτερικά του τετραπλεύρου  $Q_1$ . Σε αυτή την περίπτωση, θα πρέπει να βρίσκεται στο εσωτερικό της μη-φραγμένης περιοχής που είναι σκιασμένη με σκούρο-γκρι στην Εικόνα 13.16, έτσι ώστε η ακμή  $(v_c^2, v_d^2)$  να είναι κάθετη στην ακμή  $(v_a^1, v_d^1)$ . Αυτό συνεπάγεται ότι η κορυφή  $v_0$  δεν μπορεί να βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_2$ , καθώς η ακμή  $(v_d^2, v_a^2)$  σχηματίζει ένα τετράπλευρο που δεν περικλείει την κορυφή  $v_0$ , το οποίο οδηγεί σε άτοπο. Επομένως, η κορυφή  $v_d^2$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ . □

Από την Πρόταση 13.2.16, η ακμή  $(v_a^2, v_d^2)$  θα πρέπει να τέμνει κάθετα μία ακμή του τετραπλεύρου  $Q_1$ , καθώς η κορυφή  $v_d^2$  είναι στο εσωτερικό του, ενώ η κορυφή  $v_a^2$  στο εξωτερικό του. Αρχικά, παρατηρούμε ότι η ακμή  $(v_a^2, v_d^2)$  δεν μπορεί να είναι κάθετη ούτε στην ακμή  $(v_c^1, v_d^1)$  ούτε στην ακμή  $(v_a^1, v_d^1)$ . Για να το δείξουμε αυτό, υποθέτουμε ότι η ακμή  $(v_a^2, v_d^2)$  είναι κάθετη στην ακμή  $(v_c^1, v_d^1)$ . Τότε, η γωνία  $\widehat{v_b^1v_c^1v_d^1}$  (που συμβολίζεται με  $\phi$  στην Εικόνα 13.16) θα πρέπει να είναι μεγαλύτερη του  $\pi$ , το οποίο αντιβαίνει στο γεγονός ότι η κορυφή  $v_0$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ . Όμοια, μπορεί να δειχθεί ότι η ακμή  $(v_a^2, v_d^2)$  δεν μπορεί να είναι κάθετη στην ακμή  $(v_a^1, v_d^1)$ . Ως εκ τούτου, η ακμή  $(v_d^2, v_a^2)$  δεν μπορεί να είναι κάθετη ούτε στην  $(v_b^1, v_c^1)$  ούτε στην ακμή  $(v_a^1, v_b^1)$ . Στην περίπτωση όπου η ακμή  $(v_d^2, v_a^2)$  τέμνει την ακμή  $(v_b^1, v_c^1)$ , οι ακμές  $(v_d^2, v_a^2)$  και  $(v_0, v_a^2)$  σχηματίζουν μία έλικα με κέντρο την κορυφή  $v$ , η οποία τέμνεται από την ακμή  $(v_b^1, v_c^1)$ , παραβιάζοντας τη RAC-Ιδιότητα 2. Όμοια, η ακμή  $(v_d^2, v_a^2)$  δεν μπορεί να είναι κάθετη στην ακμή  $(v_a^1, v_b^1)$ . Συμπερασματικά, η κορυφή  $v_d^2$  δε βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ , αντιβαίνοντας στην Πρόταση 13.2.16. □

**Θεώρημα 13.2.1.** *Οι γεωμετρικές RAC απεικονίσεις ενός ASA γραφήματος ορίζουν ακριβώς δύο συνδυαστικές εμφυτεύσεις.*

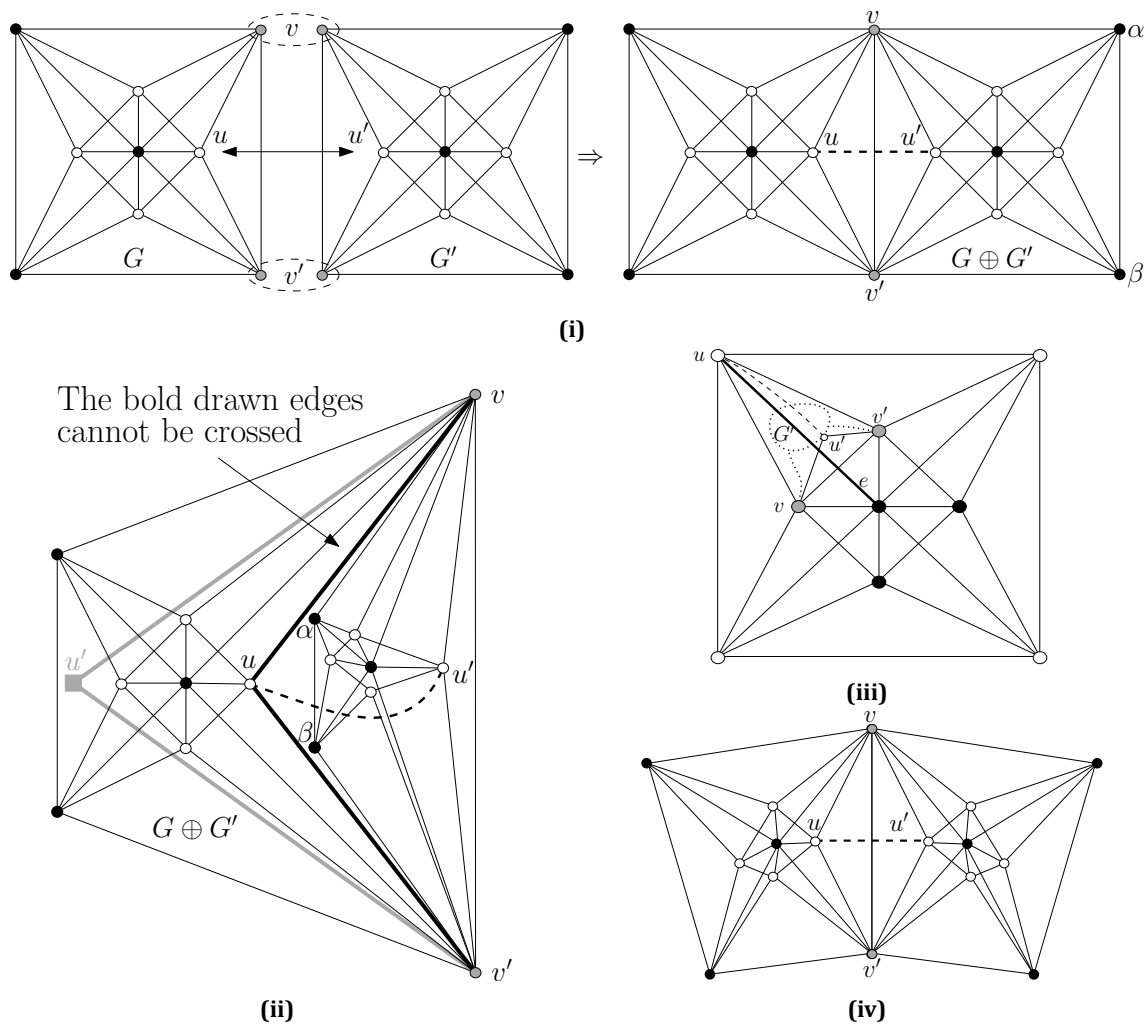


**Εικόνα 13.16:** Ο σχηματισμός που χρησιμοποιείται για την απόδειξη του Λήμματος 13.2.6: Δυνατές τοποθετήσεις της κορυφής  $v_d^2$ , υποθέτοντας ότι η κορυφή  $v_0$  βρίσκεται στο εσωτερικό της περιοχής  $R_1$  και η κορυφή  $v_c^2$  εξωτερικά του τετραπλεύρου  $wv_a^1v_0v_a^2$ , αλλά στο εσωτερικό του τετραπλεύρου  $Q_1$ .

*Απόδειξη.* Μέχρι τώρα, έχειδειχθεί ότι και τα δύο τετράπλευρα  $Q_1$  και  $Q_2$  απεικονίζονται επίπεδα, δεν τέμνονται και έχουν την κεντρική κορυφή  $v_0$  στο εσωτερικό τους. Αυτό συνεπάγεται ότι είτε το τετράπλευρο  $Q_1$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_2$  είτε το τετράπλευρο  $Q_2$  βρίσκεται στο εσωτερικό του τετραπλεύρου  $Q_1$ . Ωστόσο και στις δύο περιπτώσεις, η κορυφή  $v_0$ , η οποία πρέπει να ενωθεί με τις τέσσερις κορυφές του “εξωτερικού” τετραπλεύρου, θα πρέπει αναπόφευκτα να τέμνει τις τέσσερις ακμές του “εσωτερικού” τετραπλεύρου. Συνεπώς, υπάρχουν μόνο δύο εφικτές συνδυαστικές εμφυτεύσεις όπως παρουσιάζονται στις Εικόνες 13.1i και 13.1ii.  $\square$

Το ASA γράφημα επεκτείνεται “συγκολλώντας” (*glueing*) κατάλληλα πολλαπλά στιγμιότυπα του, το ένα δίπλα στο άλλο. Η Εικόνα 13.17i περιγράφει τον τρόπο με τον οποίο πραγματοποιείται αυτή η διαδικασία σε δύο στιγμιότυπα, έστω  $G$  και  $G'$ , του ASA γραφήματος, δηλαδή ταυτίζοντας δύο “εξωτερικές” κορυφές, έστω  $v$  και  $v'$ , του στιγμιότυπου  $G$  με δύο “εξωτερικές” κορυφές του στιγμιότυπου  $G'$  (βλ. τις γκρι-χρωματισμένες κορυφές της Εικόνας 13.17i) και εισάγοντας μία επιπλέον ακμή (βλ. τη διακεκομμένη ακμή της Εικόνας 13.17i), η οποία συνδέει μία “εσωτερική” κορυφή, έστω  $u$ , του στιγμιότυπου  $G$  με την αντίστοιχη “εσωτερική” κορυφή, έστω  $u'$ , του στιγμιότυπου  $G'$ . Έστω  $G \oplus G'$  το γράφημα που παράγεται από τη διαδικασία της “συγκόλλησης” των στιγμιότυπων  $G$  και  $G'$ . Δεδομένου ότι οι RAC απεικονίσεις των στιγμιότυπων  $G$  και  $G'$  ορίζουν η καθεμία από δύο συνδυαστικές εμφυ-

φυτεύσεις, κάποιος θα πιθανολογούσε ότι οι RAC απεικονίσεις του γραφήματος  $G \oplus G'$  θα όριζαν τέσσερις δυνατές συνδυαστικές εμφυτεύσεις. Στη συνέχεια, θα δειχθεί ότι αυτό δεν ισχύει και πιο συγκεκριμένα, θα δειχθεί ότι υπάρχει ακριβώς μία συνδυαστική εμφύτευση.



**Εικόνα 13.17:** (i) “Συγκολλώντας” δύο στιγμιότυπα του ASA γραφήματος, (ii) Η επιπλέον (διακεκομμένη) ακμή δεν επιτρέπει στο δεύτερο στιγμιότυπο να απεικονιστεί στο εσωτερικό του πρώτου. (iii) Οι κορυφές, οι οποίες ταυτίζονται κατά τη διαδικασία “συγκόλλησης” (κορυφές  $v$  και  $v'$  της εικόνας), θα πρέπει να βρίσκονται στην εξωτερική όψη κάθε ASA γραφήματος. (iv) Κάθε διαδικασία “συγκόλλησης” μπορεί να εισάγει μία “στροφή” στην αντίστοιχη RAC απεικόνιση.

**Θεώρημα 13.2.2.** Έστω  $G$  και  $G'$  δύο στιγμιότυπα του ASA γραφήματος. Τότε, οι γεωμετρικές RAC απεικονίσεις του γραφήματος  $G \oplus G'$  ορίζουν ακριβώς μία συνδυαστική εμφύτευση.

*Απόδειξη.* Προφανώς, σε κάθε RAC απεικόνιση του γραφήματος  $G \oplus G'$  και τα δύο στιγμιότυπα  $G$  και  $G'$  θα πρέπει να απεικονιστούν RAC. Έστω ότι σε μία RAC απεικόνιση του γραφήματος  $G \oplus G'$ , το στιγμιότυπο  $G$  απεικονίζεται έτσι ώστε οι κορυφές  $v$  και  $v'$  (οι οποίες ταυτίζονται κατά τη διαδικασία “συγκόλλησης”) να βρίσκονται στην εξωτερική όψη της απεικόνισης  $\Gamma(G)$ .<sup>3</sup> Τότε, ο κοινός γείτονας των κορυφών  $v$  και  $v'$  στο στιγμιότυπο  $G'$ , ο οποίος δεν ταυτίζεται με την κορυφή  $v_0$  (σ.σ., η κορυφή  $u'$ ), μπορεί να βρίσκεται είτε στο εσωτερικό της απεικόνισης  $\Gamma(G)$  ή στην εξωτερική όψη της απεικόνισης του γραφήματος  $G \oplus G'$ .

Εξετάζουμε πρώτα την περίπτωση που η κορυφή  $u'$  βρίσκεται στο εσωτερικό της απεικόνισης  $\Gamma(G)$  του γραφήματος  $G \oplus G'$  (βλ. Εικόνα 13.17ii). Η κορυφή  $u'$ , η οποία είναι γειτονική και με τις δύο κορυφές  $v$  και  $v'$ , δεν μπορεί να βρίσκεται στα “αριστερά” και των δύο ακμών  $(u, v)$  και  $(u, v')$  (βλ. τις έντονα ζωγραφισμένες ακμές της Εικόνας 13.17ii), καθώς αυτό θα οδηγούσε σε μία περίπτωση όπου τρεις ακμές τέμνονται ανά δύο μεταξύ τους, το οποίο αντιβαίνει στη RAC-Ιδιότητα 1 (βλ. τη γκρι-σκιασμένη τετράγωνη κορυφή της Εικόνας 13.17ii). Επομένως, η κορυφή  $u'$  θα βρίσκεται στην τριγωνική όψη της απεικόνισης του  $G$  που σχηματίζεται από τις κορυφές  $u, v$  και  $v'$ . Όμοια, το ίδιο ισχύει και για την κεντρική κορυφή του στιγμιότυπου  $G'$ , η οποία είναι επίσης γειτονική των κορυφών  $v$  και  $v'$ . Από τη RAC-Ιδιότητα 3, κάθε κοινός γείτονας των κορυφών  $u'$  και  $v$  θα πρέπει, επίσης, να βρίσκεται στην ίδια τριγωνική όψη του στιγμιότυπου  $G$ , το οποίο προοδευτικά συνεπάγεται ότι ολόκληρο το στιγμιότυπο  $G'$  θα πρέπει να βρίσκεται σε αυτή την όψη, όπως φαίνεται στην Εικόνα 13.17ii. Δεδομένου ότι οι κορυφές  $v$  και  $v'$  βρίσκονται στην εξωτερική όψη της απεικόνισης  $\Gamma(G')$  στη RAC απεικόνιση του γραφήματος  $G \oplus G'$  και το στιγμιότυπο  $G'$  θα πρέπει να απεικονιστεί RAC, οι κορυφές  $\alpha$  και  $\beta$  της Εικόνας 13.17ii θα πρέπει, επίσης, να βρίσκονται στην εξωτερική όψη της απεικόνισης  $\Gamma(G')$  (λόγω του Θεωρήματος 13.2.1). Ωστόσο, σε αυτή την περίπτωση, καθώς η κορυφή  $u'$  είναι γειτονική των  $v$  και  $v'$ , η ακμή  $(u, u')$  που εισάγεται κατά τη διαδικασία “συγκόλλησης”, τέμνει το εσωτερικό του στιγμιότυπου  $G'$ , κάτι το οποίο δεν επιτρέπεται. Αυτό σημαίνει ότι εάν η κορυφή  $u'$  βρίσκεται στο εσωτερικό της απεικόνισης  $\Gamma(G)$  σε μία RAC απεικόνιση του γραφήματος  $G \oplus G'$ , τότε δεν υπάρχει δυνατή εμφύτευση.

Εάν υποθεθεί ότι η κορυφή  $u'$  βρίσκεται στην εξωτερική όψη της απεικόνισης  $\Gamma(G)$  σε μία RAC απεικόνιση του γραφήματος  $G \oplus G'$ , τότε μπορεί όμοια να δειχθεί ότι ολόκληρο το στιγμιότυπο  $G'$  θα πρέπει επίσης να βρίσκεται σε αυτή την όψη. Επομένως, οι κορυφές  $v$  και  $v'$  βρίσκονται επίσης στην εξωτερική όψη της απεικόνισης  $\Gamma(G')$ . Από τα παραπάνω έπεται ότι οι κορυφές  $v$  και  $v'$  βρίσκονται στην εξωτερική όψη και των δύο απεικονίσεων  $\Gamma(G)$  και

<sup>3</sup>Η περίπτωση όπου οι κορυφές  $v$  και  $v'$  δεν είναι στην εξωτερική όψη της απεικόνισης  $\Gamma(G)$  θα εξεταστεί αργότερα.



$\Gamma(G')$  σε μία RAC απεικόνιση του γραφήματος  $G \oplus G'$ , το οποίο συνεπάγεται μία δυνατή εμφύτευση (βλ. Εικόνα 13.17iv).

Στη συνέχεια, εξετάζεται η περίπτωση όπου οι κορυφές  $v$  και  $v'$  δε βρίσκονται στην εξωτερική όψη της απεικόνισης  $\Gamma(G)$  σε μία RAC απεικόνιση του γραφήματος  $G \oplus G'$ , δηλαδή οι κορυφές  $v$  και  $v'$  βρίσκονται κατά μήκος του εσωτερικού τετράπλευρου του στιγμιότυπου  $G$  σε μία RAC απεικόνιση του γραφήματος  $G \oplus G'$ . Η περίπτωση αυτή περιγράφεται στην Εικόνα 13.17iii. Έστω  $e$  η ακμή του στιγμιότυπου  $G$  η οποία τέμνει κάθετα την ακμή  $(v, v')$  και ξεκινάει από το εξωτερικό τετράπλευρο προς την κεντρική κορυφή του στιγμιότυπου  $G$  (βλ. την έντονη-συμπαγή ακμή της Εικόνας 13.17iii). Η ακμή  $e$  θα εμπλακεί σε τομές με το στιγμιότυπο  $G'$ . Έστω  $u'$  ο κοινός γείτονας των κορυφών  $v$  και  $v'$  στο στιγμιότυπο  $G'$ . Τότε, οι ακμές  $e$ ,  $(v, v')$  και είτε η ακμή  $(v, u')$  είτε η ακμή  $(u', v')$  σχηματίζουν τρεις ακμές που τέμνονται ανά δύο μεταξύ τους στην απεικόνιση του γραφήματος  $G \oplus G'$ , αντιβαίνοντας στη RAC-Ιδιότητα 1.

Επομένως, οι κορυφές που ταυτίζονται κατά τη διαδικασία “συγκόλλησης” θα πρέπει πάντοτε να βρίσκονται στην εξωτερική όψη σε καθένα από τα ASA γραφήματα και επομένως, κάθε απεικόνιση του γραφήματος  $G \oplus G'$  έχει μοναδική συνδυαστική εμφύτευση.  $\square$

Σημειώνεται ότι η RAC απεικόνιση του γραφήματος  $G \oplus G'$  μπορεί να διαφέρει από την απεικόνιση της Εικόνας 13.17i. Στη γενική περίπτωση, κάθε διαδικασία “συγκόλλησης” μπορεί να εισάγει μία “στροφή” (turn) στην αντίστοιχη RAC απεικόνιση, όπως φαίνεται στην Εικόνα 13.17iv. Ωστόσο, η συνδυαστική εμφύτευση παραμένει η ίδια.

### 13.3 Το Πρόβλημα της Γεωμετρικής RAC Απεικόνισης είναι NP-δύσκολο

Σε αυτή την ενότητα, θα ανάγουμε το γνωστό πρόβλημα 3-SAT [GJ79] στο πρόβλημα της γεωμετρικής RAC απεικόνισης. Σε ένα στιγμιότυπο του 3-SAT προβλήματος, δίνεται μία λογική πρόταση  $\phi$  σε συζευκτική κανονική μορφή (conjunctive normal form; CNF) με μεταβλητές (variables)  $x_1, x_2, \dots, x_n$  και όρους (clauses)  $C_1, C_2, \dots, C_m$ , καθεμία με τρία σύμβολα (literals). Θα παρουσιαστεί ο τρόπος με τον οποίο κατασκευάζεται ένα γράφημα  $G_\phi$ , το οποίο επιδέχεται μία γεωμετρική RAC απεικόνιση  $\Gamma(G_\phi)$ , αν και μόνο αν η πρόταση  $\phi$  είναι ικανοποιήσιμη. Η Εικόνα 13.20 παρουσιάζει ένα παράδειγμα ενός γραφήματος, το οποίο κατασκευάζεται από την αναγωγή μας με βάση μία συγκεκριμένη 3-SAT πρόταση που δίνεται ως είσοδος. Η απόδειξη μας ακολουθεί τη γενική προσέγγιση μιας αντίστοιχης πρότασης των Formann et al. [FHH<sup>+</sup>93] (η οποία χρησιμοποιήθηκε για την απόδειξη ότι το πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας είναι NP-δύσκολο) χρησιμοποιώντας διαφορετικά gadgets για την κωδικοποίηση των μεταβλητών και των όρων της δοθείσας πρότα-

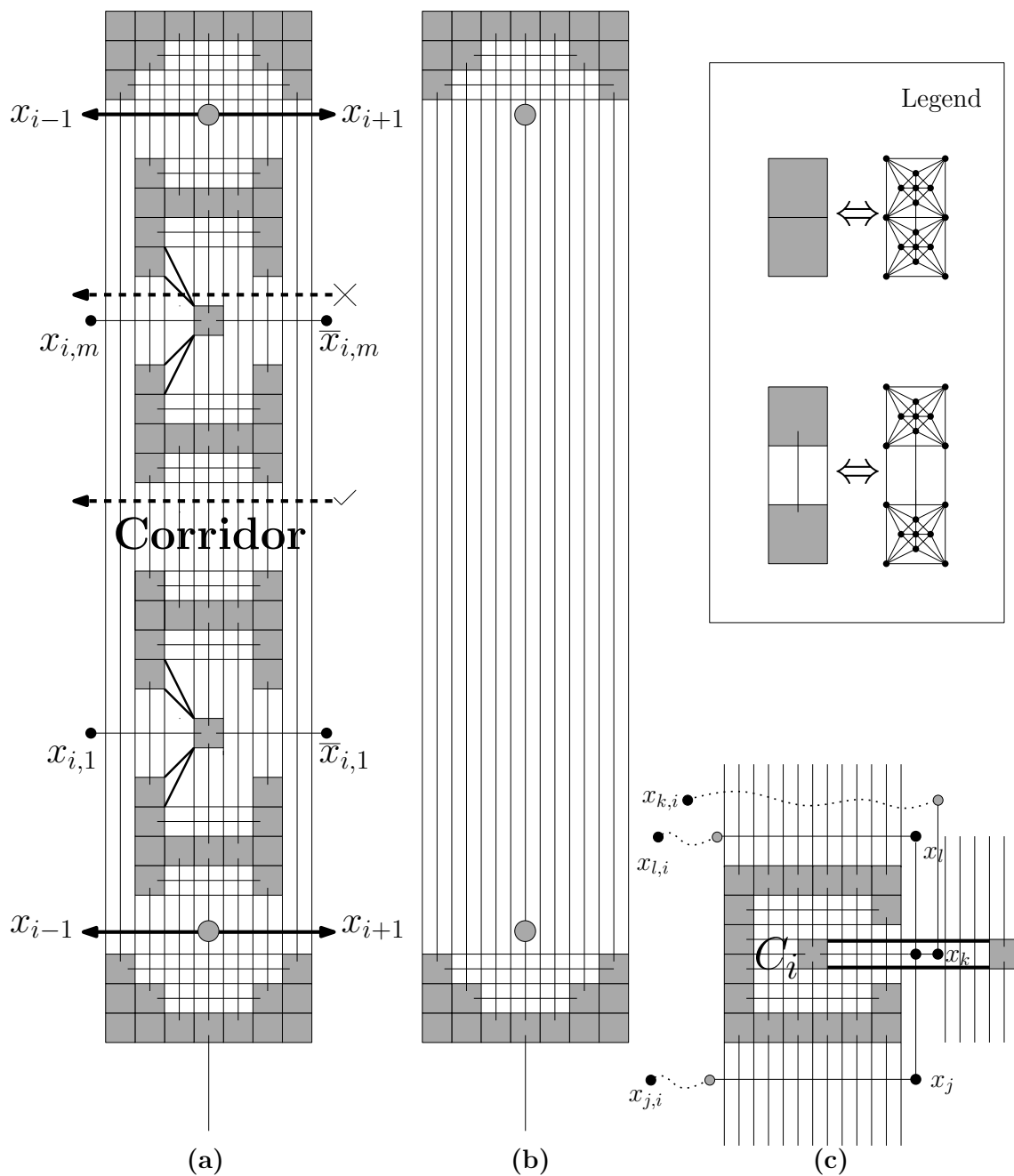
σης.

### 13.3.1 Περιγραφή της Κατασκευής

Η Εικόνα 13.18 παρουσιάζει τα gadgets της κατασκευής μας. Κάθε γκρι-σκιασμένο τετράγωνο σε αυτές τις απεικονίσεις αντιστοιχεί σε ένα ASA γραφήμα. Γειτονικά γκρι-σκιασμένα τετράγωνα αντιστοιχούν σε “κολλημένα” ASA γραφήματα (βλέπε, για παράδειγμα, τα πάνω γκρι-σκιασμένα τετράγωνα της Εικόνας 13.18a). Υπάρχουν, επίσης, γκρι-σκιασμένα τετράγωνα, τα οποία δεν είναι γειτονικά, αλλά είναι ενωμένα με ακμές. Η λεζάντα της Εικόνας 13.18 περιγράφει τον τρόπο με τον οποίο πραγματοποιούνται αυτές οι συνδέσεις.

Το gadget που κωδικοποιεί τις μεταβλητές  $x_i$  της πρότασης  $\phi$  δίνεται στην Εικόνα 13.18i. Αποτελείται από ένα συνδυασμό ASA γραφημάτων, καθώς και “οριζόντιες” και “κάθετες” ακμές οι οποίες σχηματίζουν έναν πύργο τέτοιο ώστε οι RAC απεικονίσεις κάθε πύργου να ορίζουν μοναδική συνδυαστική εμφύτευση. Στη μία πλευρά κάθε πύργου τοποθετούνται πολλαπλές κορυφές που αντιστοιχούν στο σύμβολο  $x_i$ , ενώ στην απέναντι πλευρά, τοποθετούνται οι κορυφές που αντιστοιχούν στην άρνηση του  $\bar{x}_i$  (βλ. τις κορυφές  $x_{i,1}, \dots, x_{i,m}$  και  $\bar{x}_{i,1}, \dots, \bar{x}_{i,m}$  της Εικόνας 13.18i). Αυτές οι κορυφές ονομάζονται *άκρο-μεταβλητών* (*variable endpoints*). Εάν στην τελική απεικόνιση οι κορυφές που αντιστοιχούν στην άρνηση εμφανίζονται στην “αριστερή” ή στη “δεξιά” πλευρά κάθε πύργου, τότε θα ανατεθεί η τιμή “αληθής” ή “ψευδής” στη μεταβλητή  $x_i$ , αντίστοιχα. Ζεύγη διαδοχικών άκρων  $x_{i,j}$  και  $x_{i,j+1}$  διαχωρίζονται από ένα “διάδρομο” (*corridor*), ο οποίος σημειώνεται με “✓” στην Εικόνα 13.18i, που επιτρέπει τη διέλευση κάθετων προς τον πύργο ακμών (βλ. το κάτω διακεκομμένο βέλος της Εικόνας 13.18i). Σημειώνεται ότι καμία ακμή δεν μπορεί να περάσει μέσω ενός “διαδρόμου” που σχηματίζεται σε ένα άκρο-μεταβλητών (σημειώνεται με “X” στην Εικόνα 13.18i), καθώς υπάρχουν τέσσερις μη-παράλληλες ακμές που “παρεμποδίζουν” οποιαδήποτε άλλη ακμή τις διαπεράσει (βλ. το πάνω διακεκομμένο βέλος της Εικόνας 13.18i). Οι διάδρομοι μπορεί να έχουν μεταβλητό ύψος. Στο gadget-μεταβλητών της  $x_i$ , υπάρχουν, επίσης, δύο κορυφές (απεικονίζονται ως γκρι-σκιασμένοι κύκλοι στην Εικόνα 13.18i), οι οποίες έχουν βαθμό τέσσερα. Οι κορυφές αυτές χρησιμεύουν ως “σύνδεσμοι” (*connectors*) μεταξύ διαδοχικών gadget-μεταβλητών, δηλαδή οι κορυφές αυτές θα πρέπει να ενωθούν με τις αντίστοιχες κορυφές των gadget-μεταβλητών των  $x_{i-1}$  και  $x_{i+1}$ . Σημειώνεται ότι οι σύνδεσμοι των gadget-μεταβλητών που συσχετίζονται με τις μεταβλητές  $x_1$  και  $x_n$  ενώνονται με συνδέσμους των gadget-μεταβλητών που αντιστοιχούν στις  $x_2$  και  $x_{n-1}$ , αντίστοιχα και με συνδέσμους των *εικονικών gadget-μεταβλητών* (*dummy variable gadgets*).

Η Εικόνα 13.18ii περιγράφει ένα εικονικό gadget-μεταβλητών, το οποίο (όμοια με το gadget-μεταβλητών) αποτελείται από ένα συνδυασμό ASA γραφημάτων, καθώς και “οριζόντιες” και “κάθετες” ακμές, οι οποίες σχηματίζουν ένα πύργο. Οι RAC απεικονίσεις αυ-



Εικόνα 13.18: Τα gadgets της κατασκευής: (i) Gadget των μεταβλητών, (ii) Εικονικά gadgets των μεταβλητών, (iii) Gadget των όρων

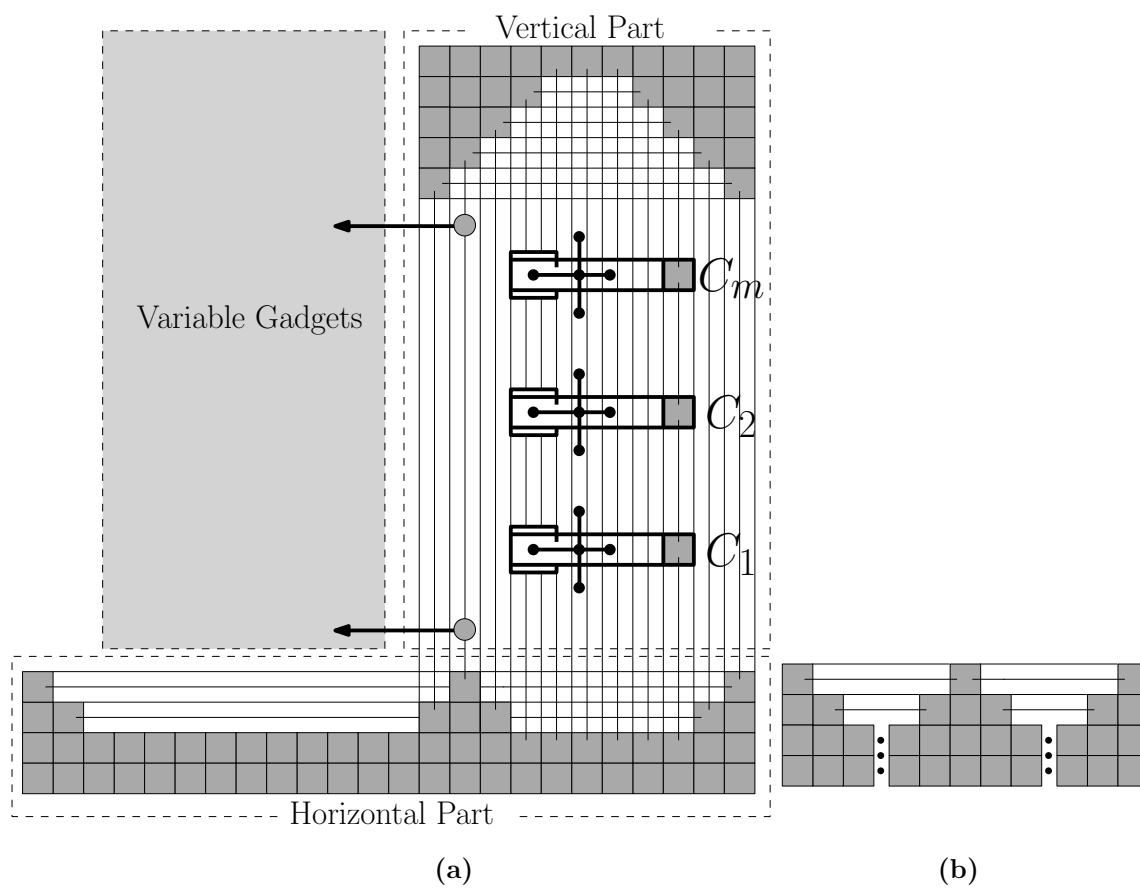
τού του gadget ορίζουν επίσης μία μοναδική συνδυαστική εμφύτευση. Ένα εικονικό gadget-μεταβλητών δεν υποστηρίζει κορυφές που αντιστοιχούν σε σύμβολα. Ωστόσο, περιέχει συνδέσμους (απεικονίζονται ως γκρι-σκιασμένοι κύκλοι στην Εικόνα 13.18ii). Στην κατασκευή μας, χρησιμοποιούμε ακριβώς δύο εικονικά gadget-μεταβλητών. Οι σύνδεσμοι καθενός από τα εικονικά gadget-μεταβλητών θα πρέπει να ενωθούν με τους αντίστοιχους συνδέσμους στα gadget-μεταβλητών που συσχετίζονται με τις μεταβλητές  $x_1$  και  $x_n$ , αντίστοιχα.

Το gadget που κωδικοποιεί τους όρους της πρότασης  $\phi$  περιγράφεται στην Εικόνα 13.18iii και μοιάζει με βαλβίδα. Έστω  $C_i = (x_j \vee x_k \vee x_l)$  ένας όρος της πρότασης  $\phi$ . Όπως φαίνεται στην Εικόνα 13.18iii, το gadget που αντιστοιχεί στον όρο  $C_i$  περιέχει τρεις κορυφές<sup>4</sup>, έστω  $x_j$ ,  $x_k$  και  $x_l$ , τέτοιες ώστε: η κορυφή  $x_j$  πρέπει να ενωθεί με τη κορυφή  $x_{j,i}$ , η κορυφή  $x_k$  με την κορυφή  $x_{k,i}$  και η κορυφή  $x_l$  με την κορυφή  $x_{l,i}$  με μονοπάτια μήκους δύο. Αυτές οι κορυφές, τις οποίες αναφέρουμε ως *άκρα-όρων* (*clause endpoints*), κωδικοποιούν τα σύμβολα κάθε όρου. Προφανώς, εάν ένας όρος περιέχει την άρνηση ενός συμβόλου θα πρέπει να ενωθεί με το άκρο του αντίστοιχου gadget-μεταβλητών που αντιστοιχεί στην άρνηση. Τα άκρα-όρων είναι γειτονικά σε μία κορυφή “παγιδευμένη” από δύο παράλληλες ακμές (βλ. τις έντονα ζωγραφισμένες ακμές της Εικόνας 13.18iii). Επομένως, σε μία RAC απεικόνιση του γραφήματος  $G_\phi$ , μόνο δύο από αυτά μπορούν να τέμνουν κάθετα αυτές τις ακμές, ένα από “πάνω” (*πάνω άκρο*) και ένα από “κάτω” (*κάτω άκρο*). Το άλλο (*δεξί άκρο*) θα πρέπει να παραμείνει στο εσωτερικό των δύο παράλληλων ακμών. Αυτό που θα παραμείνει “παγιδευμένο” στην τελική απεικόνιση θα αντιστοιχεί στο “αληθές” σύμβολο κάθε όρου.

Τα gadgets τα οποία αντιστοιχούν στις μεταβλητές και στους όρους της πρότασης  $\phi$  ενώνονται μεταξύ τους στο σκελετό του γραφήματος  $G_\phi$  που παρουσιάζεται στην Εικόνα 13.19i. Ο σκελετός αποτελείται από δύο κύρια τμήματα, ένα “οριζόντιο” και ένα “κάθετο”. Στο κάθετο τμήμα τοποθετούνται τα gadgets των όρων (βλ. Εικόνα 13.19i). Το οριζόντιο μέρος χρησιμοποιείται για τη “σύνδεση” των gadgets των μεταβλητών. Οι μακριές ακμές, οι οποίες τέμνουν κάθετα (βλ. τις τεμνόμενες ακμές ακριβώς πάνω από το οριζόντιο τμήμα της Εικόνας 13.19i), έπονται ότι το κάθετο τμήμα του σκελετού θα είναι πάντα κάθετο στο οριζόντιο. Το οριζόντιο τμήμα του σκελετού παρουσιάζεται ξεχωριστά στην Εικόνα 13.19ii. Σημειώνεται ότι το οριζόντιο τμήμα περιέχει ένα σύνολο οριζόντιων ευθειών, οι οποίες σε συνδυασμό με τις κάθετες ακμές των gadgets των μεταβλητών και των όρων, δεν του επιτρέπουν να κάνει οποιαδήποτε καμπή.

Η Εικόνα 13.20 παρουσιάζει τον τρόπο με τον οποίο τα gadgets των μεταβλητών προσκολλώνται στο σκελετό. Πιο συγκεκριμένα, αυτό επιτυγχάνεται με μία ακμή, η οποία τέμνει κάθετα το σύνολο των οριζόντιων ακμών του οριζόντιου τμήματος. Επομένως, κάθε

<sup>4</sup>Παραβιάζοντας ελαφρώς την ορολογία, ο ίδιος όρος χρησιμοποιείται για να δηλώσουμε τις μεταβλητές της πρότασης  $\phi$  και τις κορυφές του γραφήματος  $G_\phi$ .



Εικόνα 13.19: Παρουσίαση του σκελετού της κατασκευής.

gadget-μεταβλητών είναι κάθετα προσκολλημένο στο σκελετό, όπως στην Εικόνα 13.20. Σημειώνεται ότι, κάθε gadget-μεταβλητών θα πρέπει να απεικονίζεται εξολοκλήρου “πάνω” από αυτές τις οριζόντιες ακμές, καθώς σε διαφορετική περίπτωση οι συνδέσεις μεταξύ των άκρων-μεταβλητών και των άκρων-όρων δε θα είναι υπό ορθές γωνίες. Οι σύνδεσμοι των εικονικών gadgets των μεταβλητών, τα gadgets των μεταβλητών και το κάθετο τμήμα της κατασκευής διασφαλίζουν ότι τα gadgets των μεταβλητών θα είναι παράλληλα μεταξύ τους (σ.σ., δεν επιτρέπεται να κάνουν καμπή) και επιπλέον παράλληλα στο κάθετο τμήμα της κατασκευής.

### 13.3.2 Ιδιότητες της Κατασκευής

Στη συνέχεια, θα εξεταστούν μερικές ιδιότητες της κατασκευής μας. Κάθε μονοπάτι μήκους δύο, το οποίο ξεκινάει από ένα πάνω ή κάτω άκρο-όρων, μπορεί να συνδεθεί με ένα άκρο-μεταβλητών είτε από την αριστερή ή από τη δεξιά πλευρά του αντίστοιχου gadget-μεταβλητών. Η πρώτη ακμή του συγκεκριμένου μονοπατιού θα πρέπει να τέμνει τις κάθετες ακμές του κάθετου τμήματος της κατασκευής και να διαπεράσει κάποιους διαδρόμους<sup>5</sup>, ενώ η δεύτερη ακμή θα χρησιμοποιηθεί για να πραγματοποιηθεί η “τελική” σύνδεση με το άκρο του gadget-μεταβλητών.

Ωστόσο, δεν ισχύει το ίδιο για τα μονοπάτια που ξεκινούν από το δεξί άκρο-όρων. Αυτά τα μονοπάτια μπορούν να προσεγγίσουν μόνο άκρα-μεταβλητών που βρίσκονται στη δεξιά πλευρά των αντίστοιχων gadgets-μεταβλητών. Πιο συγκεκριμένα, η πρώτη ακμή του μονοπατιού μήκους δύο θα πρέπει να τμήσει μία από τις παράλληλες ακμές (βλ. τις έντονα ζωγραφισμένες ακμές της Εικόνας 13.18iii) που την “παγιδεύουν”, ενώ η άλλη θα χρησιμοποιηθεί για να προσεγγίσει (περνώντας μέσω διαδρόμων) το τελικό άκρο-μεταβλητών (βλ. Εικόνα 13.20).

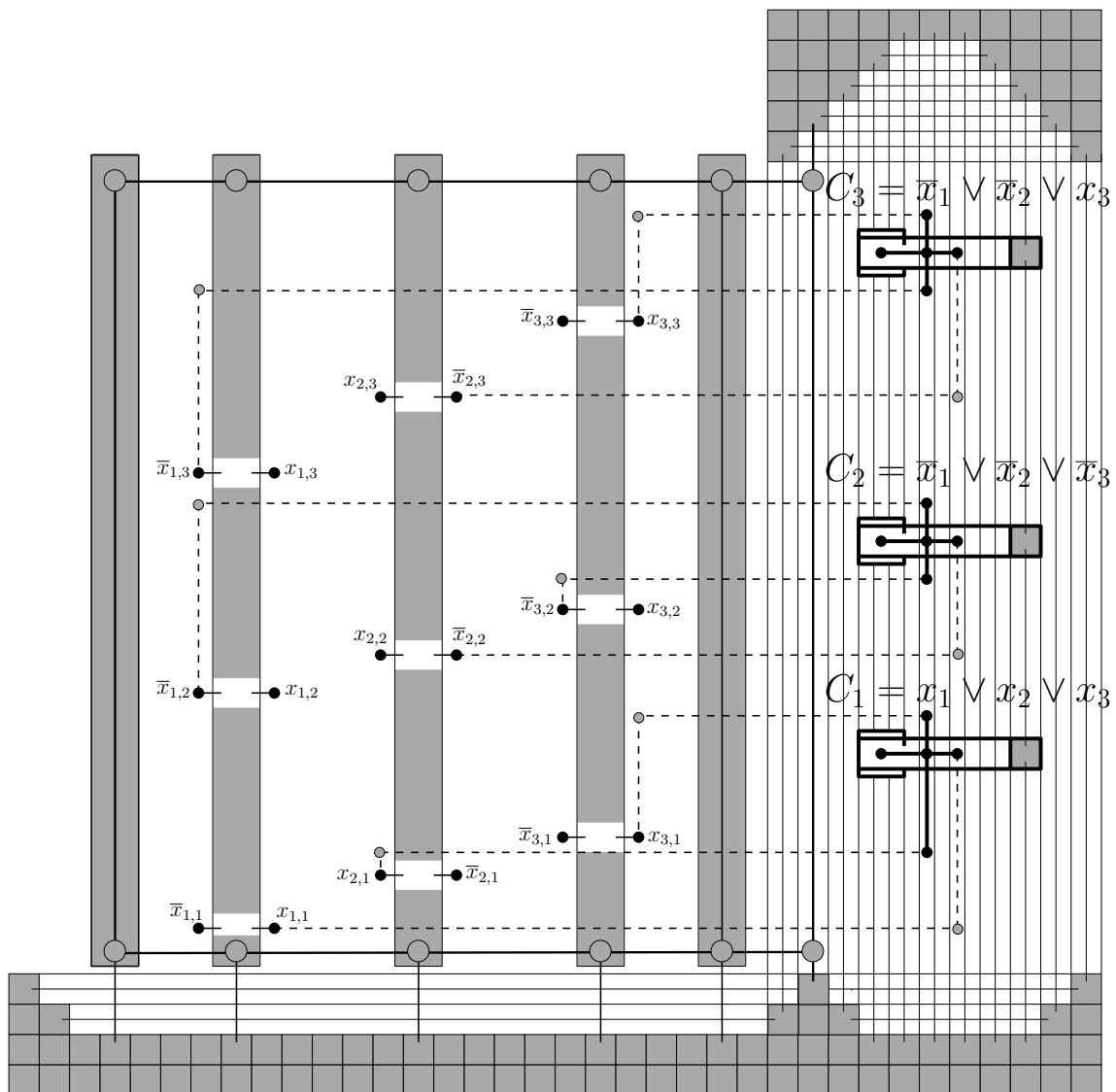
Η κατασκευή μας εξασφαλίζει ότι κάτω από μετατοπίσεις (translations), περιστροφές (rotations) και επεκτάσεις (stretchings), κάθε RAC απεικόνιση του γραφήματος  $G_\phi$  θα μοιάζει με αυτή της Εικόνας 13.20.

### 13.3.3 Αναγωγή από το 3-SAT Πρόβλημα

**Θεώρημα 13.3.1.** *Είναι NP-δύσκολο να καθοριστεί εάν ένα γράφημα επιδέχεται μία γεωμετρική RAC απεικόνιση.*

*Απόδειξη.* Είναι προφανές ότι η κατασκευή μπορεί να πραγματοποιηθεί σε  $O(nm)$  χρόνο. Υποθέτουμε τώρα ότι υπάρχει μία RAC απεικόνιση  $\Gamma(G_\phi)$  του γραφήματος  $G_\phi$ . Εάν οι κορυ-

<sup>5</sup>Στην Εικόνα 13.20, οι διάδρομοι είναι οι γκρι-σκιασμένες περιοχές που βρίσκονται σε κάθε gadget μεταβλητών.



**Εικόνα 13.20:** Η αναγωγή του 3-SAT στο πρόβλημα της γεωμετρικής RAC απεικόνισης. Η πρόταση εισόδου είναι  $\phi = (x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_3)$ . Η απεικόνιση αντιστοιχεί στην αληθή ανάθεση τιμών  $x_1=x_3=\text{true}$ ,  $x_2=\text{false}$ .

φές που αντιστοιχούν στις αρνήσεις του gadget-μεταβλητών της  $x_i$ ,  $i = 1, 2, \dots, n$ , βρίσκονται στην “αριστερή” πλευρά στην απεικόνιση  $\Gamma(G_\phi)$ , τότε στη μεταβλητή  $x_i$  ανατίθεται η τιμή “αληθής”, διαφορετικά της ανατίθεται η τιμή “ψευδής”. Υποστηρίζουμε ότι η συγκεκριμένη ανάθεση ικανοποιεί την πρόταση  $\phi$ . Για γίνει αυτό κατανοητό, παρατηρούμε ότι υπάρχουν τρία μονοπάτια που ξεκινούν από κάθε gadget-όρων. Αυτό που ξεκινάει από το δεξί άκρο κάθε τέτοιου gadget δεν μπορεί ποτέ να προσεγγίσει μία ψευδή τιμή. Επομένως, κάθε όρος της πρότασης  $\phi$  θα περιέχει τουλάχιστον ένα αληθές σύμβολο, το οποίο συνεπάγεται ότι η πρόταση  $\phi$  είναι ικανοποιήσιμη.

Αντιστρόφως, υποθέτουμε ότι υπάρχει μία αληθής ανάθεση τιμών, η οποία ικανοποιεί την πρόταση  $\phi$ . Θα προχωρήσουμε στην κατασκευή της RAC απεικόνισης ως εξής: Στην περίπτωση που στην αληθή ανάθεση τιμών η μεταβλητή  $x_i$ ,  $i = 1, 2, \dots, n$  έχει λάβει την τιμή “αληθής”, τοποθετούμε τις κορυφές που αντιστοιχούν στην άρνηση στο gadget-μεταβλητών της  $x_i$ , στην αριστερή πλευρά της απεικόνισης  $\Gamma(G_\phi)$ , διαφορετικά στη δεξιά της. Δεδομένου ότι κάθε όρος της πρότασης  $\phi$  περιέχει τουλάχιστον ένα αληθές σύμβολο, επιλέγουμε αυτό να είναι το δεξί άκρο του αντίστοιχου gadget-όρων. Όπως αναφέρθηκε παραπάνω, είναι πάντα εφικτό αυτό να ενωθεί με το gadget-μεταβλητών του με μονοπάτια μήκους δύο, ολοκληρώνοντας την απόδειξη του θεωρήματος.  $\square$



## 14 Γεωμετρικές RAC Ταυτόχρονες Απεικονίσεις Γραφημάτων

Στο κεφάλαιο αυτό, μελετάται το πρόβλημα της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων (ή εν συντομία GRacSim πρόβλημα απεικόνισης γραφημάτων). Υπενθυμίζεται ότι σύμφωνα με το GRacSim πρόβλημα, δοθέντων δύο επίπεδων γραφημάτων του ίδιου συνόλου κορυφών, ο στόχος είναι η εύρεση μιας γεωμετρικής απεικόνισης στην οποία κάθε γράφημα απεικονίζεται επίπεδο, δεν υπάρχουν επικαλύψεις ακμών και οι τομές μεταξύ ακμών των δύο γραφημάτων σχηματίζουν ορθές γωνίες (βλ. Ενότητα 12.4). Αρχικά αποδεικνύεται ότι δύο επίπεδα γραφήματα, τα οποία επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση, δεν επιδέχονται απαραίτητα και γεωμετρική RAC ταυτόχρονη απεικόνιση. Ακολούθως, αποδεικνύεται ότι ένας κύκλος και ένα ταίριασμα επιδέχονται γεωμετρική RAC ταυτόχρονη απεικόνιση. Μελετάται, επίσης, ένα αρκετά συναφές πρόβλημα με αυτό της γεωμετρικής RAC ταυτόχρονης απεικόνισης γραφημάτων, το οποίο αναφέρεται ως το πρόβλημα της γεωμετρικής γραφο-δυϊκής RAC ταυτόχρονης απεικόνισης γραφημάτων (ή εν συντομία *GDual-GRacSim* πρόβλημα απεικόνισης γραφημάτων). Αποδεικνύεται ότι είναι πάντοτε εφικτή η κατασκευή μιας τέτοιας απεικόνισης, εάν το γράφημα εισόδου είναι ένα εξωεπίπεδο εμφυτευμένο γράφημα.

### 14.1 Γράφημα-Τροχός και Ταίριασμα: Ένα Αρνητικό Αποτέλεσμα

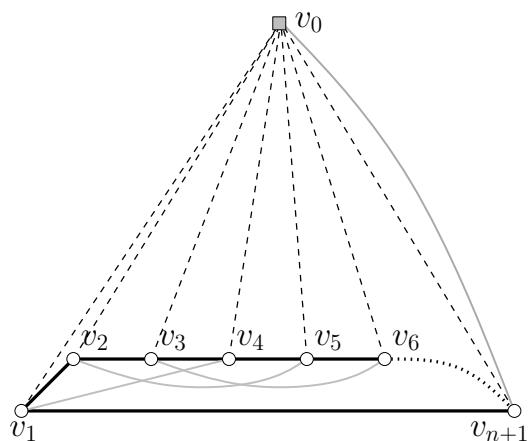
Στην ενότητα αυτή, αποδεικνύεται η ύπαρξη ενός ζεύγους επίπεδων γραφημάτων τα οποία επιδέχονται γεωμετρική ταυτόχρονη απεικόνιση, η ένωση τους ικανοποιεί τον περιορισμό του RAC-μεγέθους, αλλά δεν επιδέχονται μία GRacSim απεικόνιση (δηλαδή η κλάση των γραφημάτων τα οποία επιδέχονται GRacSim απεικονίσεις είναι υποσύνολο της κλάσης των γραφημάτων για τα οποία μία γεωμετρική ταυτόχρονη απεικόνιση είναι εφικτή). Αυτό επιτυγχάνεται, δείχνοντας ότι υπάρχει ένα γράφημα-τροχός και ένα ταίριασμα, τα οποία δεν επιδέχονται μία GRacSim απεικόνιση<sup>1</sup>. Οι Cabello et al. [CvKL<sup>+</sup>11] έχουν δείξει ότι μία

<sup>1</sup>Ένα γράφημα-τροχός και ένα ταίριασμα σε ένα σύνολο κορυφών μεγέθους  $n$  συνεισφέρουν  $5n/2 - 2$  ακμές, συνεπώς ο περιορισμός του RAC-μεγέθους ικανοποιείται.

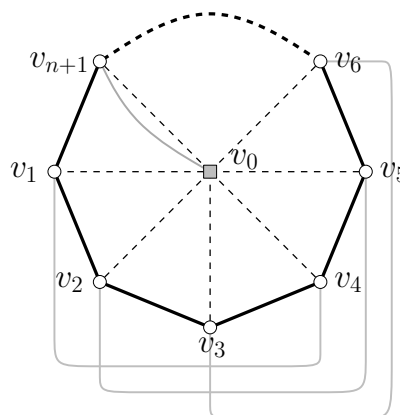
γεωμετρική ταυτόχρονη απεικόνιση ενός γραφήματος-τροχού και ενός ταιριάσματος είναι πάντοτε εφικτή.

**Θεώρημα 14.1.1.** Υπάρχει ένα γράφημα-τροχός και ένα ταιρίασμα τα οποία δεν επιδέχονται μία GRacSim απεικόνιση.

*Απόδειξη.* Έστω  $\mathcal{W}$  το γράφημα-τροχός,  $\mathcal{M}$  το ταιρίασμα και  $V = \{v_0, v_1, \dots, v_{n+1}\}$  το κοινό σύνολο κορυφών των  $\mathcal{W}$  και  $\mathcal{M}$ , όπου  $n \geq 6$ . Υποθέτουμε επίσης ότι  $n = 6k$ , για κάποιο  $k \in \mathbb{Z}$ . Εάν η κορυφή  $v_0$  είναι το κέντρο του γραφήματος-τροχού  $\mathcal{W}$  και οι κορυφές  $v_1 \rightarrow \dots \rightarrow v_{n+1} \rightarrow v_1$  είναι η περιφέρεια του, τότε  $E(\mathcal{W}) = \{(v_i, v_{i+1}); i = 1, \dots, n\} \cup \{(v_{n+1}, v_1)\} \cup \{(v_0, v_i); i = 1, \dots, n+1\}$ . Το ταιρίασμα  $\mathcal{M}$  συνεισφέρει  $n/2 + 1$  ακμές. Μία ακμή του  $\mathcal{M}$  συνδέει την κορυφή  $v_0$  με την κορυφή  $v_{n+1}$ . Για τον καθορισμό των υπολοίπων ακμών του  $\mathcal{M}$ , οι  $6k$  κορυφές της περιφέρειας του  $\mathcal{W}$  (εξαιρουμένης της κορυφής  $v_{6k+1}$ ) χωρίζονται σε  $k$  ομάδες, όπου η ομάδα  $i$ ,  $1 \leq i \leq k$ , αποτελείται από τις κορυφές  $v_{6i}, \dots, v_{6i+5}$ . Ακολουθώντας, σε κάθε ομάδα  $i$ , η κορυφή  $v_{6i+j}$  ταιριάζει με την κορυφή  $v_{6i+j+3}$ , για κάθε  $j = 0, 1, 2$ . Πιο τυπικά, το σύνολο ακμών του ταιριάσματος ορίζεται ως εξής:  $E(\mathcal{M}) = \{(v_{6i-j-3}, v_{6i-j}); i = 1, \dots, n/6, j = 0, 1, 2\} \cup \{(v_0, v_{n+1})\}$ .



(i) Η εξωτερική όψη του γραφήματος-τροχού  $\mathcal{W}$  φράσσεται από έναν 3-κύκλο.



(ii) Η εξωτερική όψη του γραφήματος-τροχού  $\mathcal{W}$  φράσσεται από την περιφέρεια του.

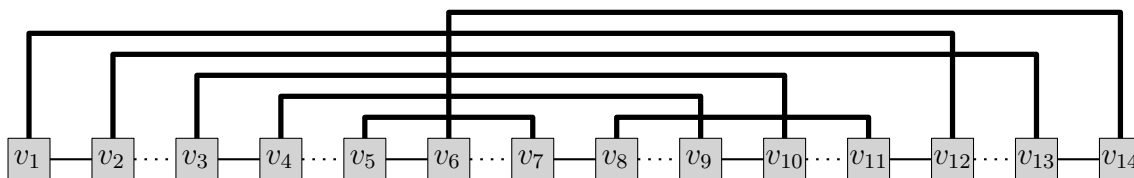
**Εικόνα 14.1:** Και στις δύο εικόνες, το κέντρο του  $\mathcal{W}$  απεικονίζεται ως ένα γκρι-σκιασμένο τετράγωνο, οι ακτίνες του  $\mathcal{W}$  είναι διακεκομμένες, η περιφέρεια του  $\mathcal{W}$  είναι έντονη, ενώ το ταιρίασμα  $\mathcal{M}$  απεικονίζεται με γκρι χρώμα. Το παράδειγμα αντιστοιχεί στην περίπτωση όπου  $k = 1$ , δηλαδή η περιφέρεια του γραφήματος-τροχού αποτελείται από επτά κορυφές.

Σε κάθε επίπεδη απεικόνιση του γραφήματος-τροχού  $\mathcal{W}$ , η εξωτερική του όψη φράσσεται είτε από έναν 3-κύκλο που σχηματίζεται από την κορυφή  $v_0$  και δύο διαδοχικές κορυφές της περιφέρειας του (βλ. Εικόνα 14.1i, στην οποία, χωρίς βλάβη της γενικότητας,

η ακμή  $(v_1, v_{n+1})$  είναι στην εξωτερική όψη) είτε από την ίδια την περιφέρεια του (βλ. Εικόνα 14.1ii). Και στις δύο περιπτώσεις, κάθε ακμή του ταιριάσματος  $\mathcal{M}$  (εξαιρουμένης της ακμής  $(v_0, v_{n+1})$ ) συνδέει δύο κορυφές οι οποίες ανήκουν σε δύο τρίγωνα του  $\mathcal{W}$  προσκείμενα στην κορυφή  $v_0$ , αλλά όχι διαδοχικά γύρω από την κορυφή  $v_0$ , στην επίπεδη απεικόνιση του γραφήματος-τροχού  $\mathcal{W}$ . Συνεπώς, από τη RAC-Ιδιότητα 2 συνεπάγεται ότι οι ακμές του ταιριάσματος  $\mathcal{M}$  δεν μπορούν να τέμνουν τις ακτίνες του γραφήματος-τροχού  $\mathcal{W}$ . Επομένως, οι ακμές του ταιριάσματος  $\mathcal{M}$  και οι ακμές του γραφήματος-τροχού  $\mathcal{W}$  δεν μπορούν να τέμνονται μεταξύ τους, εξαιρουμένης και πάλι της ακμής  $(v_0, v_{n+1})$ , η οποία ανήκει και στο ταίριασμα  $\mathcal{M}$  και στο γράφημα-τροχό  $\mathcal{W}$ . Ωστόσο, υπό αυτή την προϋπόθεση το ταίριασμα  $\mathcal{M}$  δεν μπορεί να απεικονιστεί επίπεδο.  $\square$

### 14.2 Κύκλος και Ταίριασμα: Ένα Θετικό Αποτέλεσμα

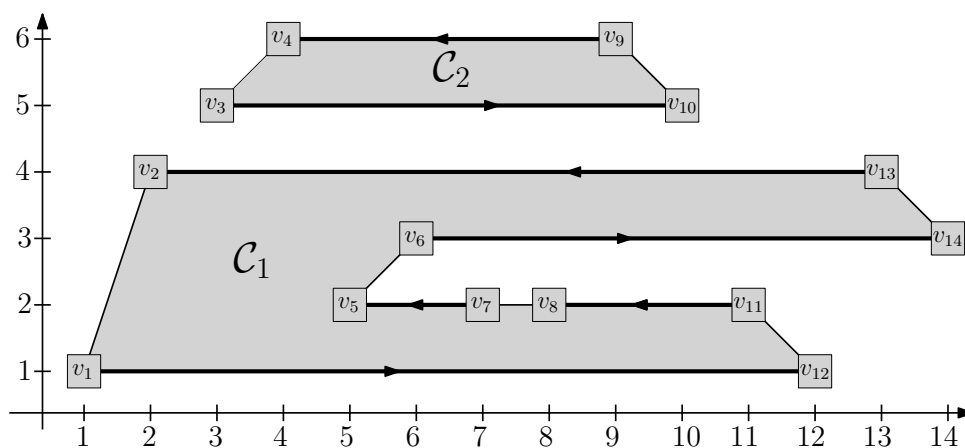
Στην ενότητα αυτή, αρχικά αποδεικνύεται ότι ένα μονοπάτι και ένα ταίριασμα επιδέχονται πάντοτε μία GRacSim απεικόνιση και ακολούθως, δείχνεται ότι το ίδιο ισχύει και στην περίπτωση ενός κύκλου και ενός ταιριάσματος. Σημειώνεται ότι, η ένωση ενός μονοπατιού και ενός ταιριάσματος δεν είναι απαραίτητα ένα επίπεδο γράφημα. Οι Cabello et al. [CvKL<sup>+</sup>11] παρουσιάζουν ένα παράδειγμα ενός μονοπατιού και ενός ταιριάσματος, τα οποία σχηματίζουν μία υποδιαίρεση (subdivision) του  $K_{3,3}$ . Έστω  $\mathcal{P}$  το μονοπάτι και  $\mathcal{M}$  το ταίριασμα. Έστω επίσης  $v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_n$  οι ακμές του  $\mathcal{P}$  (βλ. Εικόνα 14.2). Προκειμένου να διατηρηθεί η περιγραφή του αλγόριθμου απλή, αρχικά υποθέτουμε ότι το μονοπάτι  $\mathcal{P}$  και το ταίριασμα  $\mathcal{M}$  δεν έχουν κοινές ακμές, δηλαδή  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ . Αφού το μονοπάτι  $\mathcal{P}$  και το ταίριασμα  $\mathcal{M}$  ορίζονται στο ίδιο σύνολο κορυφών, ο ακέραιος αριθμός  $n$  θα πρέπει να είναι άρτιος και επιπλέον  $|E(\mathcal{M})| = n/2$  (δηλαδή το ταίριασμα  $\mathcal{M}$  είναι τέλειο (perfect)). Αργότερα σε αυτή την ενότητα, θα περιγραφούν αναλυτικά οι περιπτώσεις όπου  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$  ή όπου το  $\mathcal{M}$  δεν είναι τέλειο ταίριασμα.



**Εικόνα 14.2:** Ένα παράδειγμα ενός μονοπατιού  $\mathcal{P}$  και ενός ταιριάσματος  $\mathcal{M}$ . Το μονοπάτι απεικονίζεται στο κάτω μέρος της εικόνας. Οι ακμές του ταιριάσματος  $\mathcal{M}$  απεικονίζονται έντονες, με δύο σημεία καμπής η καθεμία. Οι ακμές του μονοπατιού  $\mathcal{P}$  ορίζουν δύο ταιριάσματα, δηλαδή  $\mathcal{P}_{odd}$  και  $\mathcal{P} - \mathcal{P}_{odd}$ . Οι ακμές του ταιριάσματος  $\mathcal{P}_{odd}$  απεικονίζονται σταθερές (solid), ενώ οι ακμές του  $\mathcal{P} - \mathcal{P}_{odd}$  διάστικτες (dotted).

Η κεντρική ιδέα του αλγορίθμου είναι ο εντοπισμός στο γράφημα, το οποίο επάγεται από την ένωση των  $\mathcal{P}$  και  $\mathcal{M}$ , ενός συνόλου κύκλων  $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k, k \leq n/4$ , τέτοιων ώστε: (i)  $|E(\mathcal{C}_1)| + |E(\mathcal{C}_2)| + \dots + |E(\mathcal{C}_k)| = n$ , (ii)  $\mathcal{M} \subseteq \mathcal{C}_1 \cup \mathcal{C}_2 \cup \dots \cup \mathcal{C}_k$  και (iii) οι ακμές του κύκλου  $\mathcal{C}_i, i = 1, 2, \dots, k$  εναλλάσσονται μεταξύ ακμών του  $\mathcal{P}$  και του  $\mathcal{M}$ . Σημειώνεται ότι από τις ιδιότητες (i) και (ii) συνεπάγεται ότι οι κύκλοι θα πρέπει να περιέχουν μισές από τις ακμές του μονοπατιού  $\mathcal{P}$  (και πιο συγκεκριμένα,  $\lceil |E(\mathcal{P})|/2 \rceil$  ακμές) και όλες τις ακμές του ταιριάσματος  $\mathcal{M}$ . Στην τελική απεικόνιση, οι ακμές αυτές δε θα διασταυρώνονται μεταξύ τους. Οι υπόλοιπες ακμές του  $\mathcal{P}$  θα διασταυρώνονται με τις ακμές του ταιριάσματος  $\mathcal{M}$  υπό ορθές γωνίες.

Έστω  $\mathcal{P}_{odd}$  το υπογράφημα του  $\mathcal{P}$ , το οποίο περιέχει κάθε δεύτερη ακμή του  $\mathcal{P}$ , ξεκινώντας από την πρώτη του ακμή, δηλαδή  $E(\mathcal{P}_{odd}) = \{(v_i, v_{i+1}); 1 \leq i < n, i \text{ περιττός}\}$ . Στην Εικόνα 14.2, οι ακμές του  $\mathcal{P}_{odd}$  απεικονίζονται σταθερές (solid). Προφανώς, το γράφημα  $\mathcal{P}_{odd}$  είναι ένα ταίριασμα. Αφού ο ακέραιος αριθμός  $n$  είναι άρτιος, το γράφημα  $\mathcal{P}_{odd}$  περιέχει ακριβώς  $n/2$  ακμές. Ως εκ τούτου,  $|E(\mathcal{P}_{odd})| = |E(\mathcal{M})|$ . Επιπλέον, το  $\mathcal{P}_{odd}$  καλύπτει όλες τις κορυφές του  $\mathcal{P}$  και  $E(\mathcal{P}_{odd}) \cap E(\mathcal{M}) = \emptyset$ . Από την τελευταία σχέση, τετριμμένα συνεπάγεται η αρχική μας υπόθεση ότι  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ . Επομένως, το συμπέρασμα στο οποίο καταλήγουμε είναι ότι το γράφημα  $\mathcal{P}_{odd} \cup \mathcal{M}$  είναι ένα 2-κανονικό γράφημα. Κατά συνέπεια, κάθε συνδεδεμένο συστατικό του  $\mathcal{P}_{odd} \cup \mathcal{M}$  αντιστοιχεί σε έναν κύκλο άρτιου μήκους, ο οποίος εναλλάσσεται μεταξύ ακμών του γραφήματος  $\mathcal{P}_{odd}$  και του μονοπατιού  $\mathcal{M}$ . Αυτό ακριβώς είναι το σύνολο των κύκλων που περιγράφηκε παραπάνω (βλ. Εικόνα 14.3).



**Εικόνα 14.3:** Το γράφημα  $\mathcal{P}_{odd} \cup \mathcal{M}$  (της Εικόνας 14.2) αποτελείται από τους κύκλους  $\mathcal{C}_1$  και  $\mathcal{C}_2$ . Οι ακμές του ταιριάσματος  $\mathcal{P}_{odd}$  απεικονίζονται σταθερές (solid), ενώ οι ακμές του ταιριάσματος  $\mathcal{M}$  απεικονίζονται έντονες (bold).

Αρχικά, καθορίζονται οι  $x$ -συντεταγμένες κάθε κορυφής<sup>2</sup> του μονοπατιού  $\mathcal{P}$  ως εξής:

<sup>2</sup>Σημειώνεται ότι, ο αλγόριθμος μπορεί εύκολα να τροποποιηθεί έτσι ώστε ο υπολογισμός τόσο της  $x$  και της

$x(v_i) = i, 1 \leq i \leq n$ . Συνεπώς, το μονοπάτι  $\mathcal{P}$  είναι  $x$ -μονότονο και συνεπώς, επίπεδο. Αργότερα, θα αλλαχθούν κατά κάτι οι  $x$ -συντεταγμένες ορισμένων κορυφών του  $\mathcal{P}$ , χωρίς ωστόσο να επηρεαστεί η μονοτονία του  $\mathcal{P}$ . Οι  $y$ -συντεταγμένες κάθε κορυφής του  $\mathcal{P}$  καθορίζονται από τους κύκλους του γραφήματος  $\mathcal{P}_{odd} \cup \mathcal{M}$ .

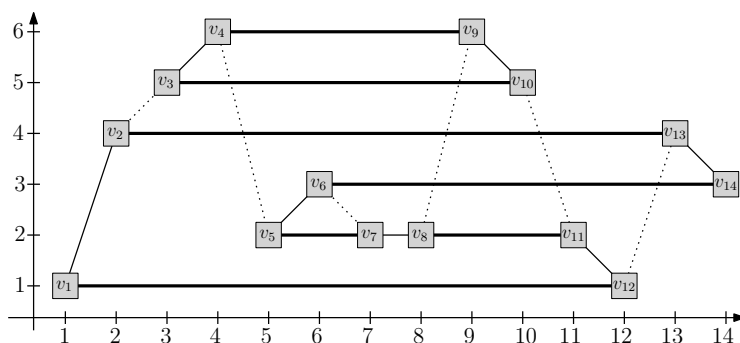
Οι κύκλοι του γραφήματος  $\mathcal{P}_{odd} \cup \mathcal{M}$  απεικονίζεται σειριακά όπως περιγράφεται παρακάτω. Πιο συγκεκριμένα, υποθέτοντας ότι έχουν απεικονιστεί μηδέν ή περισσότεροι κύκλοι, συμβολίζουμε με  $\mathcal{C}$  τον κύκλο του  $\mathcal{P}_{odd} \cup \mathcal{M}$  που περιέχει την αριστερότερη κορυφή, έστω  $v_i$ , του  $\mathcal{P}$  που δεν έχει ακόμη απεικονιστεί (αρχικά, η κορυφή  $v_i$  ταυτίζεται με την κορυφή  $v_1$ ). Η κορυφή  $v_i$  θα πρέπει να είναι περιττού-δείκτη και κατά συνέπεια, η ακμή  $(v_i, v_{i+1})$  θα πρέπει να ανήκει στον κύκλο  $\mathcal{C}$ . Ο κύκλος  $\mathcal{C}$  προσανατολίζεται έτσι ώστε η κορυφή  $v_i$  να είναι η πρώτη στον κύκλο  $\mathcal{C}$ , ενώ η κορυφή  $v_{i+1}$  η τελευταία του (βλ. Εικόνα 14.3). Με βάση αυτόν τον προσανατολισμό, θα απεικονιστούν οι ακμές του  $\mathcal{C}$ , ξεκινώντας από την κορυφή  $v_i$  και τερματίζοντας στην κορυφή  $v_{i+1}$ . Η πρώτη ακμή, η οποία θα πρέπει να απεικονιστεί, πρόσκεινται στην κορυφή  $v_i$  και ανήκει στο ταίριασμα  $\mathcal{M}$ . Επιλέγουμε λοιπόν να την απεικονίσουμε ως ένα οριζόντιο ευθύγραμμο τμήμα στο κατώτατο διαθέσιμο επίπεδο (layer) της απεικόνισης που έχει κατασκευαστεί έως τώρα (αρχικά, στο επίπεδο  $L_1 : y = 1$ ). Αφού ο κύκλος  $\mathcal{C}$  εναλλάσσεται μεταξύ ακμών του  $\mathcal{P}_{odd}$  και του  $\mathcal{M}$ , η επόμενη ακμή που θα πρέπει να απεικονιστεί ανήκει στο  $\mathcal{P}_{odd}$  ακολουθούμενη από μία ακμή του  $\mathcal{M}$ . Αν και οι δύο μπορούν να απεικονιστούν στο τρέχον επίπεδο χωρίς να εισάγουν επικαλύψεις ακμών, τότε απεικονίζονται με τον τρόπο αυτό. Σε διαφορετική περίπτωση, ένα νέο επίπεδο εισάγεται και η ακμή του  $\mathcal{P}_{odd}$  απεικονίζεται πλαγίως από το τρέχον στο επόμενο επίπεδο, ενώ η ακμή του  $\mathcal{M}$  απεικονίζεται ως οριζόντιο ευθύγραμμο τμήμα στο νεοεισαχθέν επίπεδο. Συνεχίζοντας κατά τον ίδιο τρόπο, κάποια στιγμή θα κληθούμε να απεικονίσουμε την ακμή  $(v_i, v_{i+1})$  του κύκλου  $\mathcal{C}$ . Η ακμή αυτή ενώνει δύο διαδοχικές ακμές του μονοπατιού  $\mathcal{P}$ , οι οποίες είναι οι αριστερότερες στην απεικόνιση του κύκλου  $\mathcal{C}$ . Κατά συνέπεια, η ακμή  $(v_i, v_{i+1})$  μπορεί να προστεθεί στην απεικόνιση του κύκλου  $\mathcal{C}$  χωρίς να εισάγει διασταυρώσεις. Συνεπώς, ο κύκλος  $\mathcal{C}$  απεικονίζεται επίπεδος.

Μέχρι στιγμής, έχουμε απεικονίσει όλες τις ακμές του ταίριασματος  $\mathcal{M}$  και τις μισές ακμές του μονοπατιού  $\mathcal{P}$  (δηλαδή τις ακμές του  $\mathcal{P}_{odd}$ ) και το αποτέλεσμα είναι μία επίπεδη απεικόνιση, στην οποία όλες οι ακμές του ταίριασματος  $\mathcal{M}$  αντιστοιχούν σε οριζόντια μη-επικαλυπτόμενα ευθύγραμμα τμήματα. Στη χειρότερη περίπτωση, η απεικόνιση αυτή καταλαμβάνει  $n/2$  επίπεδα.

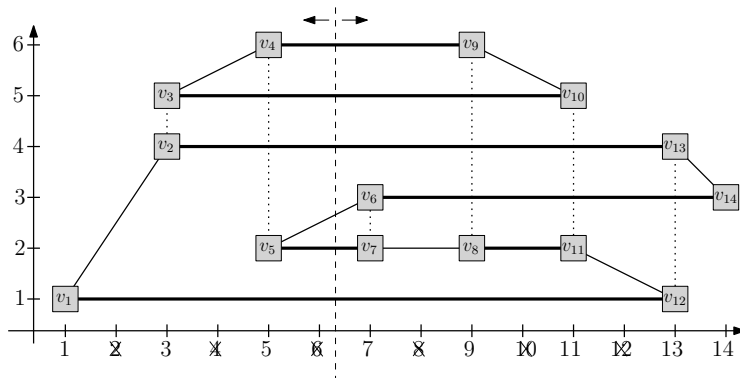
Ο αλγόριθμος ακολούθως, ενσωματώνει τις υπόλοιπες ακμές του μονοπατιού  $\mathcal{P}$  (δηλαδή αυτές που ανήκουν στο  $\mathcal{P} - \mathcal{P}_{odd}$ ) στην απεικόνιση (βλ. τις διάστικτες ακμές της Εικόνας 14.4i).

---

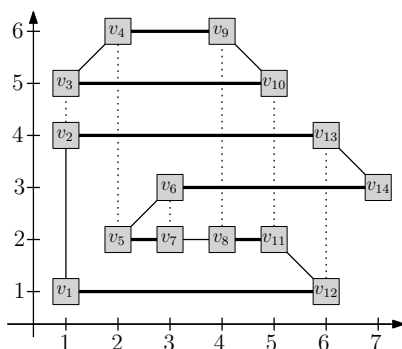
$y$  συντεταγμένης κάθε κορυφής να γίνεται την ίδια χρονική στιγμή (χωρίς να επηρεαστούν τόσο η ορθότητα όσο και η πολυπλοκότητα του αλγορίθμου). Έχουμε επιλέξει ο υπολογισμός τους να γίνεται ξεχωριστά, ώστε να διατηρηθεί η παρουσίαση του αλγορίθμου απλή.



(i) Μία απεικόνιση η οποία προκύπτει εισάγοντας τις ακμές του ταιριάσματος  $\mathcal{P} - \mathcal{P}_{odd}$  στην απεικόνιση της Εικόνας 14.3.



(ii) Μία απεικόνιση η οποία προκύπτει μετακινώντας κατά μία μονάδα μήκους δεξιά κάθε κορυφή άρτιου-δείκτη του μονοπατιού  $\mathcal{P}$  της απεικόνισης της Εικόνας 14.4i.



(iii) Μία ακόμη μικρότερη GRacSim απεικόνιση.

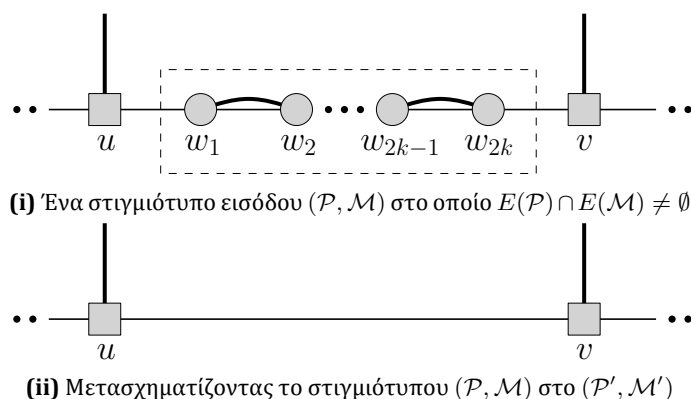
**Εικόνα 14.4:** Στις παραπάνω απεικονίσεις, οι ακμές του ταιριάσματος  $\mathcal{P}_{odd}$  απεικονίζονται σταθερές (solid), ενώ οι ακμές του ταιριάσματος  $\mathcal{P} - \mathcal{P}_{odd}$  διάστικτες (dotted). Οι ακμές του ταιριάσματος  $\mathcal{M}$  απεικονίζονται έντονες (bold).

Το γεγονός ότι  $x(v_i) = i, i = 1, 2, \dots, n$ , εξασφαλίζει ότι οι ακμές του μονοπατιού  $\mathcal{P}$  δεν τέμνονται μεταξύ τους και κατά συνέπεια, ότι το μονοπάτι  $\mathcal{P}$  απεικονίζεται επίπεδο. Ωστόσο, μία ακμή του  $\mathcal{P} - \mathcal{P}_{odd}$  πιθανότατα διασταυρώνεται με πολλές ακμές του ταιριάσματος  $\mathcal{M}$  και οι διασταυρώσεις αυτές δε σχηματίζουν ορθές γωνίες (βλ. Εικόνα 14.4i). Για την “επιδιόρθωση” αυτών των διασταυρώσεων, μετακινούμε κάθε άρτιου-δείκτη κορυφή του μονοπατιού  $\mathcal{P}$  κατά μία μονάδα μήκους δεξιά (διατηρώντας την  $y$ -συντεταγμένη τους αμετάβλητη), εξαιρουμένης της τελευταίας κορυφής του μονοπατιού  $\mathcal{P}$ . Με τον τρόπο αυτό, τα άκρα των ακμών του  $\mathcal{P} - \mathcal{P}_{odd}$  έχουν ακριβώς την ίδια  $x$ -συντεταγμένη και τέμνουν υπό ορθές γωνίες τις αντίστοιχες ακμές του ταιριάσματος  $\mathcal{M}$ , οι οποίες απεικονίζονται ως οριζόντια ευθύγραμμα τμήματα. Σημειώνεται ότι το μονοπάτι παραμένει  $x$ -μονότονο (όχι όμως γνησίως μονότονο) και συνεπώς, επίπεδο. Επιπρόσθετα, δεν είναι δυνατή η εισαγωγή επικαλύψεων μεταξύ κορυφών, καθώς στην παραχθείσα απεικόνιση κάθε ακμή του ταιριάσματος  $\mathcal{M}$  έχει μήκος τουλάχιστον δύο (υπενθυμίζεται ότι  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ ). Επίσης, στην παραχθείσα απεικόνιση οι κορυφές του γραφήματος δεν καταλαμβάνουν άρτιες  $x$ -συντεταγμένες. Συνεπώς, το μήκος της μπορεί να μειωθεί από  $n$  σε  $n/2+1$  (βλ. Εικόνα 14.4ii). Το μήκος της απεικόνισης μπορεί να μειωθεί περαιτέρω, συγχωνεύοντας διαδοχικές στήλες της απεικόνισης, οι οποίες δεν αλληλεπιδρούν στην  $y$ -κατεύθυνση σε μία κοινή στήλη (βλ. Εικόνα 14.4iii). Ωστόσο, η παραχθείσα απεικόνιση δεν απαιτεί απαραίτητα ασυμπτωτικά μικρότερη περιοχή σχεδίασης. Έτσι, ολοκληρώνεται η περιγραφή του αλγορίθμου για την περίπτωση όπου  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ .

**Θεώρημα 14.2.1.** Ένα μονοπάτι  $\mathcal{P}$  και ένα τέλει ταίριασμα  $\mathcal{M}$  στο ίδιο σύνολο κορυφών και τέτοιο ώστε  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$  επιδέχονται πάντοτε μία GRacSim απεικόνιση σε ένα  $(n/2 + 1) \times n/2$  ακέραιο πλέγμα, όπου  $n$  είναι το μέγεθος του συνόλου κορυφών. Επιπλέον, η απεικόνιση μπορεί να υπολογιστεί σε γραμμικό χρόνο.

*Απόδειξη.* Η εύρεση των κύκλων του  $\mathcal{P}_{odd} \cup \mathcal{M}$  μπορεί να γίνει σε  $O(n)$  χρόνο, όπου  $n$  είναι ο αριθμός των κορυφών του μονοπατιού  $\mathcal{P}$ , αρχικά εντοπίζοντας την αριστερότερη κορυφή του κάθε κύκλου και εν συνεχεία, διασχίζοντας τον. Έχοντας υπολογίσει τους κύκλους του γραφήματος  $\mathcal{P}_{odd} \cup \mathcal{M}$ , οι συντεταγμένες των κορυφών μπορούν να υπολογιστούν σε  $O(n)$  συνολικό χρόνο διασχίζοντας κάθε κύκλο.  $\square$

Για την περίπτωση στην οποία το μονοπάτι  $\mathcal{P}$  και το ταίριασμα  $\mathcal{M}$  μοιράζονται ακμές (δηλαδή  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$ ), ο σκοπός μας είναι η κατασκευή μιας απεικόνισης σε ένα  $n \times n/2$  ακέραιο πλέγμα επεκτείνοντας τον αλγόριθμο του Θεωρήματος 14.2.1. Για την επίτευξη του σκοπού αυτού, θα χρησιμοποιήσουμε το γεγονός ότι οι παραχθείσες απεικονίσεις για την περίπτωση όπου  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$  είναι *επεκτάσιμες* (*stretchable*) υπό την εξής έννοια: Αν θεωρήσουμε μία κάθετη ευθεία, η οποία δεν ταυτίζεται με μία από τις ευθείες πλέγματος και



**Εικόνα 14.5:** Στις παραπάνω απεικονίσεις, οι ακμές του μονοπατιού  $\mathcal{P}$  απεικονίζονται σταθερές (solid), ενώ οι ακμές του ταίριασματος  $\mathcal{M}$  έντονες (bold). Οι κορυφές του συνόλου  $V_{dis}(\mathcal{P})$  ( $V_{com}(\mathcal{P})$ , αντίστοιχα) απεικονίζονται ως τετράγωνα (κύκλοι, αντίστοιχα)

τέμνει μέρος της απεικόνισης (βλ. τη διακεκομμένη ευθεία της Εικόνας 14.4ii), τότε είναι δυνατόν να μετατοπίσουμε προς τα αριστερά (δεξιά, αντίστοιχα) το μέρος της απεικόνισης που βρίσκεται στα αριστερά (δεξιά, αντίστοιχα) της ευθείας αυτής χωρίς να μεταβληθούν ούτε η επιπεδότητα της απεικόνισης ούτε οι γωνίες που σχηματίζονται στις διασταυρώσεις των ακμών (καθώς οι διασταυρώσεις λαμβάνουν χώρα σε ακέραιες συντεταγμένες). Παρομοίως, μπορεί κανείς να επεκτείνει κάθετα την απεικόνιση με τη χρησιμοποίηση μιας οριζόντιας ευθείας που δεν ταυτίζεται με κάποια από τις ευθείες πλέγματος.

Αρχικά, θα υποθέσουμε ότι η πρώτη και η τελευταία ακμή του μονοπατιού  $\mathcal{P}$  δεν περιέχονται στο ταίριασμα  $\mathcal{M}$ <sup>3</sup>, δηλαδή οι ακμές που ανήκουν τόσο στο μονοπάτι  $\mathcal{P}$  όσο και στο ταίριασμα  $\mathcal{M}$ , είναι εσωτερικές ακμές του  $\mathcal{P}$ . Έστω  $V_{com}(\mathcal{P})$  ( $V_{dis}(\mathcal{P})$ , αντίστοιχα) το σύνολο των ακμών του  $\mathcal{P}$  που πρόσκεινται (δεν πρόσκεινται, αντίστοιχα) σε μία ακμή που ανήκει τόσο στο μονοπάτι  $\mathcal{P}$ , όσο και στο ταίριασμα  $\mathcal{M}$  (βλ. Εικόνα 14.5). Πιο τυπικά, τα σύνολα  $V_{com}(\mathcal{P})$  και  $V_{dis}(\mathcal{P})$  ορίζονται ως εξής:  $V_{com}(\mathcal{P}) = \{u \in V(\mathcal{P}) : \exists u \in V(\mathcal{P}) \text{ s.t. } (u, v) \in E(\mathcal{P}) \cap E(\mathcal{M})\}$  και  $V_{dis}(\mathcal{P}) = V(\mathcal{P}) - V_{com}(\mathcal{P})$ . Το γεγονός ότι ούτε η πρώτη ούτε η τελευταία ακμή του μονοπατιού  $\mathcal{P}$  εμφανίζονται στο ταίριασμα  $\mathcal{M}$  εξασφαλίζει ότι  $V_{com}(\mathcal{P}) \subset V(\mathcal{P})$ . Ομοίως, ορίζουμε τα σύνολα  $V_{com}(\mathcal{M})$  και  $V_{dis}(\mathcal{M})$ . Προφανώς,  $V_{com}(\mathcal{P}) = V_{com}(\mathcal{M})$ . Εφόσον, τόσο το μονοπάτι  $\mathcal{P}$  όσο και το ταίριασμα  $\mathcal{M}$ , ορίζονται στο ίδιο σύνολο κορυφών, ισχύει επίσης ότι  $V_{dis}(\mathcal{P}) = V_{dis}(\mathcal{M})$ .

Εάν υπάρχουν ακμές που ανήκουν τόσο στο μονοπάτι  $\mathcal{P}$  όσο και στο ταίριασμα  $\mathcal{M}$ , τις διαγράφουμε προσωρινά και από το  $\mathcal{P}$  και από το  $\mathcal{M}$  ως εξής: Εάν  $u \rightarrow w_1 \rightarrow \dots \rightarrow w_{2k} \rightarrow v$  είναι ένα υπό-μονοπάτι κορυφών του  $\mathcal{P}$  τέτοιο ώστε  $u, v \in V_{dis}(\mathcal{P})$  και  $w_i \in V_{com}(\mathcal{P})$ ,

<sup>3</sup>Αργότερα, θα περιγραφεί πως αντιμετωπίζονται οι τετριμμένες περιπτώσεις που είτε η πρώτη είτε η τελευταία ακμή του μονοπατιού  $\mathcal{P}$  εμφανίζονται στο ταίριασμα  $\mathcal{M}$ .



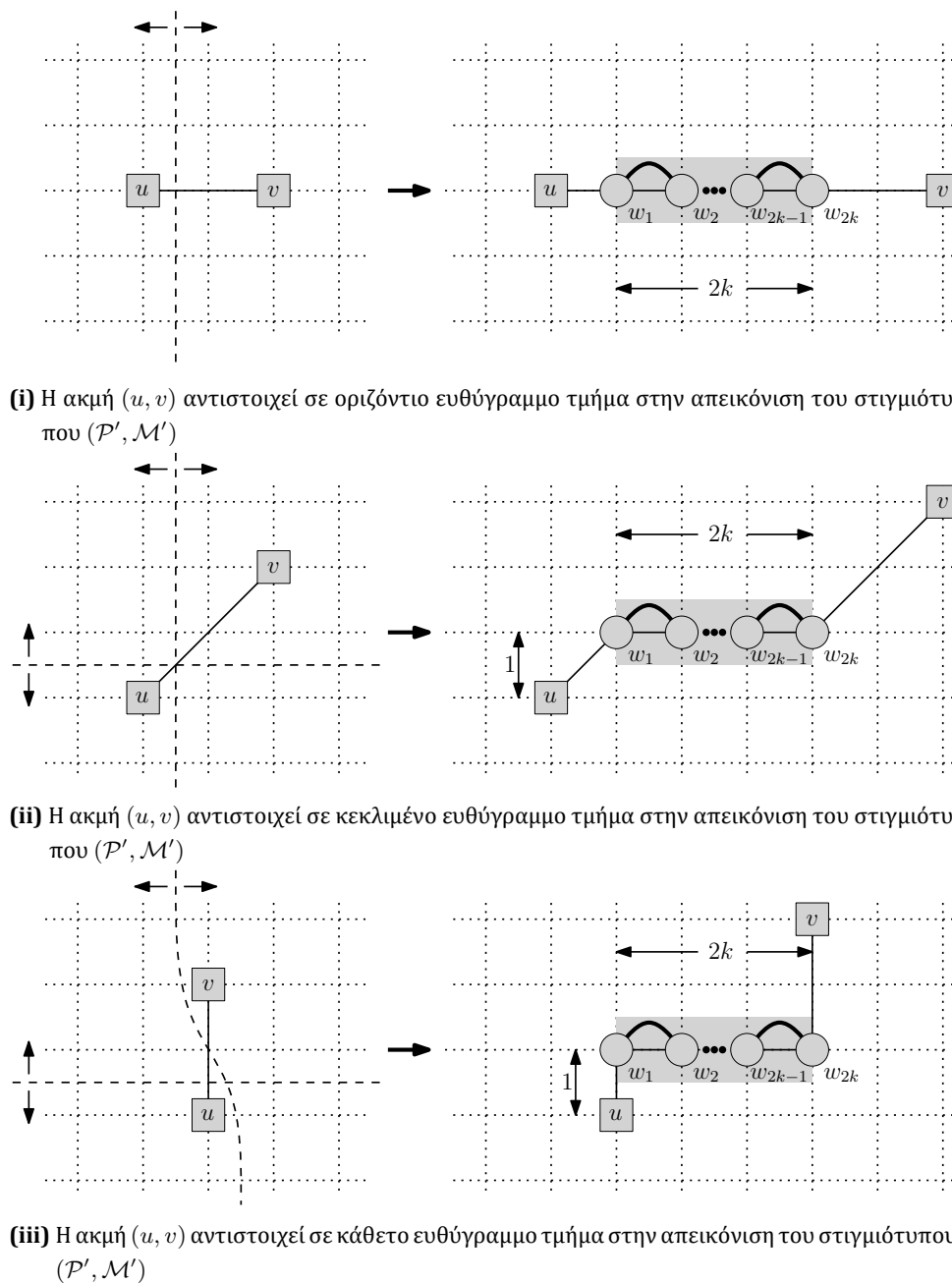
$i = 1, 2, \dots, 2k$  (βλ. Εικόνα 14.5i), τότε το αντικαθιστούμε με μία ακμή  $(u, v)$  του  $\mathcal{P}$  (βλ. Εικόνα 14.5ii). Με τον τρόπο αυτό, δημιουργούμε ένα νέο μονοπάτι  $\mathcal{P}'$  με  $n'$  κορυφές και ένα νέο ταίριασμα  $\mathcal{M}'$  με τις παρακάτω ιδιότητες:

- i)  $V(\mathcal{P}') = V_{dis}(\mathcal{P})$
- ii)  $V(\mathcal{M}') = V_{dis}(\mathcal{M})$
- iii)  $E(\mathcal{P}') \cap E(\mathcal{M}') = \emptyset$
- iv) Ο αριθμός  $n'$  είναι άρτιος
- v)  $|E(\mathcal{M}')| = n'/2$

Από τα παραπάνω έπεται ότι τα  $\mathcal{P}'$  και  $\mathcal{M}'$  μπορούν λόγω του Θεωρήματος 14.2.1 να απεικονιστούν ταυτόχρονα. Το μήκος (ύψος, αντίστοιχα) της παραγόμενης απεικόνισης ισούται με  $|V_{dis}(\mathcal{P})|/2 + 1$  ( $|V_{dis}(\mathcal{M})|/2$ , αντίστοιχα). Για να επαναφέρουμε τις κορυφές που προσωρινά διεγράφησαν, θα χρησιμοποιήσουμε το γεγονός ότι η παραχθείσα απεικόνιση είναι επεκτάσιμη.

Έστω  $u \rightarrow w_1 \rightarrow \dots \rightarrow w_{2k} \rightarrow v$  ένα υπό-μονοπάτι κορυφών βαθμού 2 του  $\mathcal{P}$ , το οποίο είχε αντικατασταθεί από μία ακμή  $(u, v) \in E(\mathcal{P}')$ , όταν το στιγμιότυπο  $(\mathcal{P}, \mathcal{M})$  μετασχηματίστηκε στο  $(\mathcal{P}', \mathcal{M}')$ . Διακρίνουμε τις εξής περιπτώσεις:

- Η ακμή  $(u, v)$  αντιστοιχεί σε οριζόντιο ευθύγραμμο τμήμα στην απεικόνιση του στιγμιότυπου  $(\mathcal{P}', \mathcal{M}')$ : Η περίπτωση αυτή περιγράφεται στην Εικόνα 14.6i. Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι  $x(u) < x(v)$ . Στην περίπτωση αυτή, επεκτείνουμε οριζόντια την απεικόνιση των  $\mathcal{P}'$  και  $\mathcal{M}'$  εισάγοντας  $2k$  μονάδες μήκους ακριβώς δίπλα στην κορυφή  $u$ . Έτσι, οι κορυφές  $w_1, w_2, \dots, w_{2k}$  απεικονίζονται σε διαδοχικές  $x$ -συντεταγμένες κατά μήκος της ευθείας  $y = y(u)$  ξεκινώντας από το σημείο  $x = x(u) + 1$  (δηλαδή  $x(w_i) = x(u) + i + 1, i = 1, 2, \dots, 2k$ ).
- Η ακμή  $(u, v)$  αντιστοιχεί σε κεκλιμένο (δηλαδή ούτε οριζόντιο ούτε κάθετο) ευθύγραμμο τμήμα στην απεικόνιση του στιγμιότυπου  $(\mathcal{P}', \mathcal{M}')$ : Η περίπτωση αυτή περιγράφεται στην Εικόνα 14.6ii. Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι  $x(u) < x(v)$  και  $y(u) < y(v)$ . Στην περίπτωση αυτή, επεκτείνουμε κάθετα την απεικόνιση των  $\mathcal{P}'$  και  $\mathcal{M}'$  εισάγοντας μία μονάδα μήκους ακριβώς πάνω από την κορυφή  $u$  και ακολούθως, οριζόντια εισάγοντας  $2k$  μονάδες μήκους ακριβώς δίπλα στην κορυφή  $u$ . Έτσι, οι κορυφές  $w_1, w_2, \dots, w_{2k}$  απεικονίζονται σε διαδοχικές  $x$ -συντεταγμένες κατά μήκος της ευθείας  $y = y(u) + 1$  ξεκινώντας από το σημείο  $x = x(u) + 1$  (δηλαδή  $x(w_i) = x(u) + i + 1, i = 1, 2, \dots, 2k$ ).



**Εικόνα 14.6:** Διαφορετικές περιπτώσεις που ενδεχομένως εμφανίζονται κατά την επανεισαγωγή ενός υπό-μονοπατιού  $u \rightarrow w_1 \rightarrow \dots \rightarrow w_{2k} \rightarrow v$  κορυφών βαθμού 2 του μονοπατιού  $\mathcal{P}$ . Σε όλες τις παραπάνω απεικονίσεις, οι ακμές του μονοπατιού απεικονίζονται σταθερές (solid), ενώ οι αντίστοιχες του ταιριάσματος έντονες (bold).

- Η ακμή  $(u, v)$  αντιστοιχεί σε κάθετο ευθύγραμμο τμήμα στην απεικόνιση του στιγμιότυπου  $(\mathcal{P}', \mathcal{M}')$ : Η περίπτωση αυτή περιγράφεται στην Εικόνα 14.6i και αντιμετωπίζεται παρόμοια με την προηγούμενη περίπτωση. Εκ νέου, υποθέτουμε, χωρίς βλάβη της γενικότητας, ότι στην απεικόνιση των  $\mathcal{P}'$  και  $\mathcal{M}'$  ισχύει ότι  $y(u) < y(v)$ . Στην περίπτωση αυτή, αρχικά επεκτείνουμε κάθετα την απεικόνιση εισάγοντας μία μονάδα μήκους ακριβώς πάνω από την κορυφή  $u$ . Ακολούθως, εφαρμόζουμε ένα μετασχηματισμό παρόμοιο με αυτό της οριζόντιας επέκτασης, για να απελευθερώσουμε  $2k$  μονάδες μήκους μεταξύ των κορυφών  $u$  και  $v$  (βλ. Εικόνα 14.6iii). Τυπικά, για να επιτύχουμε το μετασχηματισμό που περιγράφεται στην Εικόνα 14.6iii, χρησιμοποιούμε ένα κάθετο ευθύγραμμο τμήμα το οποίο συντρέχει με την ακμή  $(u, v)$  και μετατοπίζουμε το πάνω άκρο της ακμής  $(u, v)$ , διατηρώντας τη θέση του κάτω άκρου της αμετάβλητη. Έτσι, οι κορυφές  $w_1, w_2, \dots, w_{2k}$  απεικονίζονται σε διαδοχικές  $x$ -συντεταγμένες κατά μήκος της ευθείας  $y = y(u) + 1$  ξεκινώντας από το σημείο  $x = x(u)$  (δηλαδή  $x(w_i) = x(u) + i$ ,  $i = 1, 2, \dots, 2k$ ).

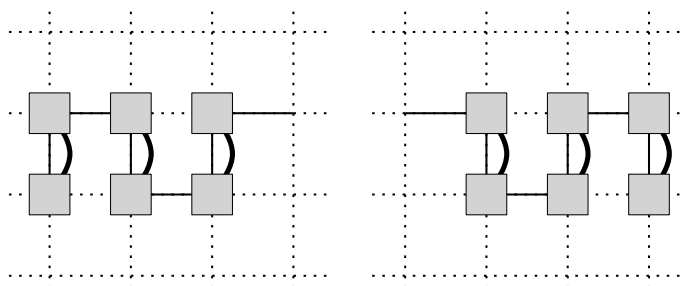
Από τα παραπάνω έπεται ότι, όταν επανεισάγονται οι  $2k$  κορυφές ενός υπό-μονοπατιού κορυφών βαθμού 2 του  $\mathcal{P}$ , το μήκος της απεικόνισης αυξάνεται κατά  $2k$  μονάδες μήκους. Συνεπώς, το συνολικό μήκος της απεικόνισης είναι το πολύ  $n$ . Από την άλλη μεριά, όλες οι ακμές του ταιριάσματος απεικονίζονται ως οριζόντια ευθύγραμμα τμήματα. Στη χειρότερη περίπτωση, δεν υπάρχουν δύο ακμές του ταιριάσματος στην ίδια ευθεία πλέγματος. Άρα, συνολικά το ύψος της απεικόνισης είναι το πολύ  $n/2$ .

Υπενθυμίζεται ότι για να αντιμετωπίσουμε την περίπτωση που το μονοπάτι  $\mathcal{P}$  και το ταιρίασμα  $\mathcal{M}$  μοιράζονται ακμές, αρχικά υποθέσαμε ότι ούτε η πρώτη ούτε η τελευταία ακμή του μονοπατιού  $\mathcal{P}$  περιέχονται στο ταιρίασμα  $\mathcal{M}$ . Ο λόγος για τον οποίο έγινε αρχικώς αυτή η παραδοχή είναι ότι οι μετασχηματισμοί που έχουμε περιγράψει έως τώρα (βλ. Εικόνα 14.6) απαιτούν υπό-μονοπάτια τα οποία ξεκινούν από μία κορυφή του  $V_{dis}(\mathcal{P})$  και (μέσω ενός άρτιου πλήθους κορυφών του  $V_{com}(\mathcal{P})$ ) τερματίζουν σε κάποια άλλη κορυφή του  $V_{dis}(\mathcal{P})$ . Εάν, επί παραδείγματι, η πρώτη κορυφή του  $\mathcal{P}$  ανήκει στο σύνολο  $V_{com}(\mathcal{P})$  (δηλαδή η πρώτη ακμή του  $\mathcal{P}$  περιέχεται στο  $\mathcal{M}$ ), τότε μπορεί να υπάρχει ένα ολόκληρο υπό-μονοπάτι κορυφών του  $V_{com}(\mathcal{P})$  στην αρχή του  $\mathcal{P}$ <sup>4</sup>. Ωστόσο, ένα τέτοιο υπό-μονοπάτι δεν υποστηρίζεται από τους μετασχηματισμούς που έχουμε περιγράψει έως τώρα, καθώς δεν ξεκινάει από μία κορυφή του  $V_{dis}(\mathcal{P})$ .

Εάν υπάρχει λοιπόν ένα τέτοιο υπό-μονοπάτι κορυφών του  $V_{com}(\mathcal{P})$  στην αρχή ή στο τέλος του μονοπατιού  $\mathcal{P}$ , τότε προσωρινά το διαγράφουμε από το  $\mathcal{P} \cup \mathcal{M}$  και το απεικονίζουμε ανεξάρτητα όπως φαίνεται στην Εικόνα 14.7. Το υπογράφημα του  $\mathcal{P} \cup \mathcal{M}$  το οποίο επάγε-

<sup>4</sup>Στη χειρότερη περίπτωση, όλες οι ακμές του  $\mathcal{P}$  ανήκουν στο  $V_{com}(\mathcal{P})$ , δηλαδή το  $\mathcal{M}$  ταιριάζει κάθε δεύτερο ζεύγος κορυφών του  $\mathcal{P}$ .

ται μετά τη διαγραφή που μόλις περιγράφηκε είναι είτε το κενό γράφημα ή ένα γράφημα με την ιδιότητα ότι ούτε η πρώτη ούτε η τελευταία κορυφή του εμφανίζεται στο ταίριασμα και κατά συνέπεια, μπορεί να απεικονιστεί με τον αλγόριθμο που έχουμε ήδη αναπτύξει. Στην πρώτη περίπτωση, η τελική απεικόνιση είναι αυτή της Εικόνας 14.7. Στη δεύτερη περίπτωση, “κολλάμε” τις απεικονίσεις των διαγραμμένων μερών στην πρώτη και στην τελευταία κορυφή, αντίστοιχα, της απεικόνισης του εναπομείναντος γραφήματος του  $\mathcal{P} \cup \mathcal{M}$ , οι οποίες καταλαμβάνουν την αριστερότερη και τη δεξιότερη θέση στην απεικόνιση του, αντίστοιχα. Το ύψος της τελικής απεικόνισης αυξάνει το πολύ κατά δύο μονάδες μήκους, ενώ το μήκος του στη χειρότερη περίπτωση ισούται με το μήκος του μονοπατιού. Συνεπώς, το συνολικό εμβαδόν της απεικόνισης δεν επηρεάζεται. Το παρακάτω θεώρημα συνοψίζει το αποτέλεσμα μας.



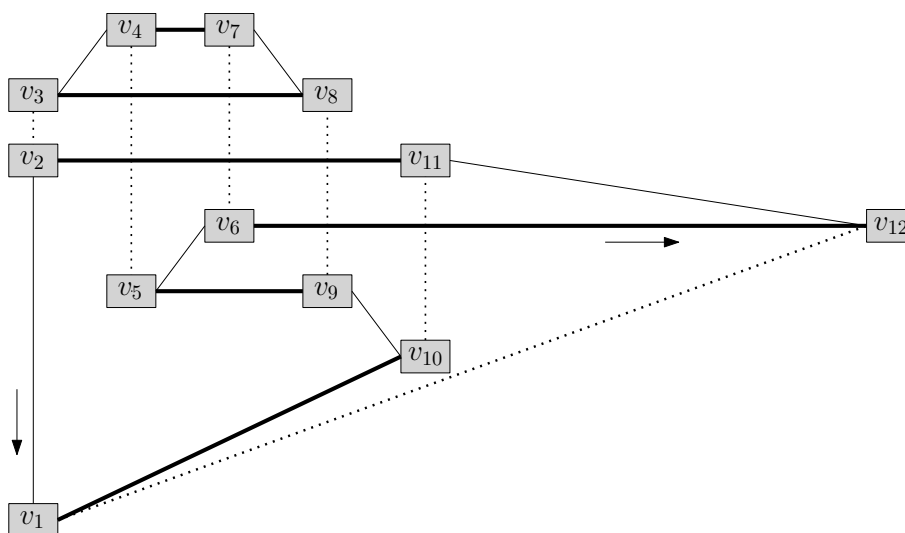
**Εικόνα 14.7:** Η απεικόνιση στην οποία το γράφημα  $\mathcal{M}$  ταιριάζει κάθε δεύτερη κορυφή του μονοπατιού.

**Θεώρημα 14.2.2.** Ένα μονοπάτι  $\mathcal{P}$  και ένα τέλει ταίριασμα  $\mathcal{M}$  στο ίδιο σύνολο κορυφών επιδέχονται πάντοτε μία GRacSim απεικόνιση σε ένα  $n \times n/2$  ακέραιο πλέγμα, όπου  $n$  είναι το μέγεθος του συνόλου κορυφών. Επιπλέον, η απεικόνιση μπορεί να υπολογιστεί σε γραμμικό χρόνο.

Επεκτείνουμε περαιτέρω τον αλγόριθμο που κατασκευάζει μία GRacSim απεικόνιση ενός μονοπατιού και ενός ταίριασματος, ώστε να καλύπτει, επίσης, και την περίπτωση ενός κύκλου  $\mathcal{C}$  και ενός ταίριασματος  $\mathcal{M}$ . Προφανώς, εάν διαγράψουμε μία ακμή από τον κύκλο εισόδου (κατά προτίμηση, μία ακμή που ανήκει στο σύνολο  $E(\mathcal{C}) - E(\mathcal{M})$ ), το γράφημα το οποίο προκύπτει είναι ένα μονοπάτι  $\mathcal{P}$  (βλ. Εικόνα 14.8). Ακολούθως, εφαρμόζοντας τον αλγόριθμο που έχουμε ήδη αναπτύξει, λαμβάνουμε μία GRacSim απεικόνιση του  $\mathcal{P}$  και του  $\mathcal{M}$ .

Παρακάτω περιγράφουμε την περίπτωση όπου  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ . Η περίπτωση κατά την οποία  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$  αντιμετωπίζεται με παρόμοιο τρόπο. Από το Θεώρημα 14.2.1, έπεται ότι η απεικόνιση χωράει σε ένα  $(n/2 + 1) \times n/2$  ακέραιο πλέγμα. Επιπλέον, η πρώτη

κορυφή του μονοπατιού  $\mathcal{P}$  απεικονίζεται στο κατώτατο επίπεδο της (συνεπώς, η ακμή του ταιριάσματος  $\mathcal{M}$  που πρόσκεινται σε αυτήν δεν εμπλέκεται σε διασταυρώσεις), ενώ η τελευταία κορυφή του μονοπατιού  $\mathcal{P}$  απεικονίζεται στη δεξιότερη θέση της απεικόνισης. Βασιζόμενοι σε αυτές τις δύο ιδιότητες, μπορούμε να επαναφέρουμε τη διαγραφείσα ακμή μεταξύ της πρώτης και της τελευταίας κορυφής του μονοπατιού  $\mathcal{P}$  χωρίς να εισάγουμε νέες διασταυρώσεις. Για να το επιτύχουμε αυτό, αρκεί να μετακινήσουμε προς τα κάτω την πρώτη κορυφή του μονοπατιού  $\mathcal{P}$  κατά  $n/2 + 2$  μονάδες μήκους (διατηρώντας τη  $x$ -συντεταγμένη του αμετάβλητη) και την τελευταία κορυφή του μονοπατιού  $\mathcal{P}$  κατά  $n/2 + 1$  μονάδες μήκους δεξιά (διατηρώντας την  $y$ -συντεταγμένη της αμετάβλητη). Με τον τρόπο αυτό, η επαναφορά στην απεικόνιση της ακμής που κλείνει τον κύκλο μπορεί να γίνει χωρίς να εισάγουμε νέες διασταυρώσεις. Το παρακάτω θεώρημα συνοψίζει το αποτέλεσμα μας.



Εικόνα 14.8: Μία GRacSim απεικόνιση ενός κύκλου και ενός ταιριάσματος.

**Θεώρημα 14.2.3.** Ένας κύκλος  $\mathcal{C}$  και ένα τέλει ταιρίασμα  $\mathcal{M}$  στο ίδιο σύνολο κορυφών και τέτοιο ώστε  $E(\mathcal{C}) \cap E(\mathcal{M}) = \emptyset$  επιδέχονται πάντοτε μία GRacSim απεικόνιση σε ένα  $(n+2) \times (n+2)$  ακέραιο πλέγμα, όπου  $n$  είναι το μέγεθος του συνόλου κορυφών. Επιπλέον, η απεικόνιση μπορεί να υπολογιστεί σε γραμμικό χρόνο.

Όπως έχει ήδη αναφερθεί, η περίπτωση όπου  $E(\mathcal{P}) \cap E(\mathcal{M}) \neq \emptyset$  αντιμετωπίζεται με παρόμοιο τρόπο. Η μόνη διαφορά έγκειται στο μέγεθος της τελικής απεικόνισης. Πιο συγκεκριμένα, σύμφωνα με το Θεώρημα 14.2.2, θα πρέπει να επανεισάγουμε τη διαγραφείσα ακμή σε μία απεικόνιση μεγέθους  $n \times n/2$ . Συνεπώς, η τελική απεικόνιση θα πρέπει να είναι μεγέθους  $3n/2 \times 3n/2$ , έτσι ώστε η επανεισαγωγή της διαγραφείσας ακμής να μην εισάγει νέες διασταυρώσεις. Το παρακάτω θεώρημα συνοψίζει το αποτέλεσμα μας.

**Θεώρημα 14.2.4.** Ένας κύκλος  $C$  και ένα τέλει ταίριασμα  $M$  στο ίδιο σύνολο κορυφών επιδέχονται πάντοτε μία GRacSim απεικόνιση σε ένα  $3n/2 \times 3n/2$  ακέραιο πλέγμα, όπου  $n$  είναι το μέγεθος του συνόλου κορυφών. Επιπλέον, η απεικόνιση μπορεί να υπολογιστεί σε γραμμικό χρόνο.

### 14.2.1 Αλγοριθμικές Επεκτάσεις

Σύμφωνα με τον τυπικό ορισμό του GRacSim προβλήματος, τα γραφήματα εισόδου ορίζονται στο ίδιο σύνολο κορυφών. Από αυτό άμεσα έπεται ότι το ταίριασμα  $M$  είναι τέλει και ότι το μέγεθος  $n$  του συνόλου κορυφών τους είναι άρτιο. Ωστόσο, ο αλγόριθμος μας μπορεί να τροποποιηθεί ώστε να υποστηρίξει και την περίπτωση που τα δύο γραφήματα εισόδου δεν ορίζονται απαραίτητα στο ίδιο σύνολο κορυφών.

Αρχικά εξετάζουμε την περίπτωση όπου η είσοδος του προβλήματος αποτελείται από ένα μονοπάτι  $\mathcal{P}$  και ένα ταίριασμα  $M$ , τα οποία πρέπει να απεικονιστούν RAC ταυτόχρονα και υποθέτουμε, χωρίς βλάβη της γενικότητας, ότι η ένωση των  $\mathcal{P}$  και  $M$  είναι ένα συνδεδεμένο γράφημα<sup>5</sup>. Έστω  $V$  η ένωση των  $V(\mathcal{P})$  και  $V(M)$  και  $n$  το μέγεθος του συνόλου  $V$ . Χωρίς βλάβη της γενικότητας, περαιτέρω υποθέτουμε ότι ο ακέραιος  $n$  είναι άρτιος<sup>6</sup>. Ακολούθως, επαυξάνουμε το μονοπάτι  $\mathcal{P}$ , έτσι ώστε να καλύπτει όλες τις κορυφές του συνόλου κορυφών  $V$ . Έστω  $\mathcal{P}_{aug}$  το επαυξημένο μονοπάτι. Έχοντας υποθέσει ότι ο ακέραιος  $n$  είναι περιττός, μπορούμε να επαυξήσουμε και το ταίριασμα  $M$ , έτσι ώστε να καλύπτει όλες τις κορυφές του συνόλου κορυφών  $V$ . Έστω  $M_{aug}$  το επαυξημένο ταίριασμα. Προφανώς, τόσο το επαυξημένο μονοπάτι  $\mathcal{P}_{aug}$  όσο και το επαυξημένο ταίριασμα  $M_{aug}$  ορίζονται στο ίδιο σύνολο κορυφών, το οποίο είναι άρτιου μεγέθους και επιπλέον το επαυξημένο ταίριασμα  $M_{aug}$  είναι τέλει. Συνεπώς, τα δύο γραφήματα  $\mathcal{P}_{aug}$  και  $M_{aug}$  μπορούν να απεικονιστούν RAC ταυτόχρονα με τον αλγόριθμο του Θεωρήματος 14.2.2. Για την απεικόνιση των αρχικών γραφημάτων  $\mathcal{P}$  και  $M$ , αρκεί να διαγραφούν από την GRacSim απεικόνιση των γραφημάτων  $\mathcal{P}_{aug}$  και  $M_{aug}$  οι επιπλέον ακμές που προστέθηκαν για την επαύξηση των  $\mathcal{P}$  και  $M$  σε  $\mathcal{P}_{aug}$  και  $M_{aug}$ , αντίστοιχα. Το παρακάτω θεώρημα συνοψίζει το αποτέλεσμα μας.

**Θεώρημα 14.2.5.** Ένα μονοπάτι  $\mathcal{P}$  και ένα ταίριασμα  $M$  επιδέχονται πάντοτε μία GRacSim απεικόνιση σε ένα  $n \times n/2$  ακέραιο πλέγμα, όπου  $n$  είναι το μέγεθος του συνόλου κορυφών  $V(\mathcal{P})$  and  $V(M)$ . Επιπλέον, η απεικόνιση μπορεί να υπολογιστεί σε γραμμικό χρόνο.

Στην περίπτωση που η είσοδος του προβλήματος αποτελείται από έναν κύκλο  $C$  και ένα ταίριασμα  $M$ , ακολουθούμε μία ελαφρώς διαφορετική προσέγγιση, καθώς ο κύκλος  $C$  δεν

<sup>5</sup>Εάν αυτό δεν ισχύει, τότε ο αλγόριθμος μας αντιμετωπίζει κάθε συνδεδεμένο συστατικό ξεχωριστά.

<sup>6</sup>Εάν ο ακέραιος  $n$  είναι περιττός, μπορούμε να προσθέσουμε μία απομονωμένη κορυφή στο σύνολο κορυφών  $V$  και να γίνει άρτιου μεγέθους.

μπορεί να επαυξηθεί. Υποθέτουμε, και πάλι χωρίς βλάβη της γενικότητας, ότι η ένωση των δύο γραφημάτων  $\mathcal{C}$  και  $\mathcal{M}$  είναι ένα συνδεδεμένο γράφημα. Αρχικά λοιπόν διαγράφουμε τις κορυφές που ανήκουν αποκλειστικά στο ταίριασμα  $\mathcal{M}$ . Ακολούθως, επαυξάνουμε το ταίριασμα  $\mathcal{M}$ , έτσι ώστε να καλύπτει όλες τις κορυφές του κύκλου  $\mathcal{C}$  (και με τον τρόπο αυτό δημιουργείται ένα νέο επαυξημένο ταίριασμα  $\mathcal{M}_{aug}$ , όπως στην προηγούμενη περίπτωση). Σημειώνεται, ωστόσο, ότι αυτό είναι εφικτό μόνο στην περίπτωση που η πληθικότητα του συνόλου κορυφών του κύκλου  $\mathcal{C}$  είναι άρτια. Εάν αυτό δεν ισχύει, τότε αναπόφευκτα θα υπάρχει μία κορυφή του κύκλου  $\mathcal{C}$  η οποία δεν καλύπτεται από μία ακμή του επαυξημένου ταίριασματος  $\mathcal{M}_{aug}$ . Υποθέτουμε, χωρίς βλάβη της γενικότητας, ότι έχουμε να αντιμετωπίσουμε αυτή την περίπτωση, καθώς είναι πιο γενική. Προκειμένου να συμμορφωθούμε με τις παραδοχές του αλγορίθμου του Θεωρήματος 14.2.4, διαγράφουμε προσωρινά την κορυφή αυτή από τον κύκλο  $\mathcal{C}$ , συνδέοντας τις δύο γειτονικές της κορυφές στον κύκλο  $\mathcal{C}$  με μία ακμή. Ο προκύπτων κύκλος, έστω  $\mathcal{C}_{dec}$  και το επαυξημένο ταίριασμα  $\mathcal{M}_{aug}$  μπορούν να απεικονιστούν RAC ταυτόχρονα χρησιμοποιώντας τον αλγόριθμο του Θεωρήματος 14.2.4, καθώς ορίζονται στο ίδιο σύνολο κορυφών και το επαυξημένο ταίριασμα  $\mathcal{M}_{aug}$  είναι τέλειο.

Για την απεικόνιση των αρχικών γραφημάτων  $\mathcal{C}$  και  $\mathcal{M}$ , αρχικά διαγράφουμε από την απεικόνιση των γραφημάτων  $\mathcal{C}_{dec}$  και  $\mathcal{M}_{aug}$  τις επιπλέον ακμές που χρησιμοποιήθηκαν για την επαύξηση του ταίριασματος  $\mathcal{M}$ . Ακολούθως, επαναφέρουμε στην απεικόνιση την κορυφή του κύκλου  $\mathcal{C}$ , η οποία είχε προσωρινά διαγραφεί κατά τον μετασχηματισμό του κύκλου  $\mathcal{C}$  στον κύκλο άρτιου-μήκους  $\mathcal{C}_{dec}$ . Για την επίτευξη αυτής της επαναφοράς, μία οριζόντια επέκταση κατά μία μονάδα μήκους (παρόμοια με αυτές που περιγράφηκαν στην απόδειξη του Θεωρήματος 14.2.2) αρκεί. Ωστόσο, με αυτή την προσέγγιση είτε το μήκος ή το ύψος της απεικόνισης αυξάνονται κατά μία μονάδα μήκους. Συνεπώς, ιδιαίτερη προσοχή θα πρέπει να δοθεί στην ακμή του κύκλου που προσπίπτει σε κορυφή του χαμηλότερου επιπέδου της απεικόνισης (δηλαδή στην ακμή του κύκλου που δεν απεικονίζεται ως οριζόντιο ή κάθετο ευθύγραμμο τμήμα, βλ. Εικόνα 14.8). Πιο συγκεκριμένα, πιθανές διασταυρώσεις εξαιτίας της ακμής αυτής μπορούν να εξαλειφθούν μετατοπίζοντας τη χαμηλότερη (δεξιότερη, αντίστοιχα) κορυφή της απεικόνισης κατά μία μονάδα μήκους προς τα κάτω (δεξιά, αντίστοιχα).

Για την ολοκλήρωση της περιγραφής της προσέγγισης μας, απομένει να περιγράψουμε τον τρόπο με τον οποίο θα επαναφέρουμε στην παραχθείσα απεικόνιση τις κορυφές που ανήκουν αποκλειστικά στο ταίριασμα  $\mathcal{M}$  και τις αντίστοιχες ακμές του ταίριασματος (οι οποίες είχαν διαγραφεί στην αρχή αυτής της διαδικασίας). Έχοντας υποθέσει ότι η ένωση των γραφημάτων  $\mathcal{C}$  και  $\mathcal{M}$  είναι ένα συνδεδεμένο γράφημα, καθεμία από αυτές τις κορυφές θα πρέπει να προσπίπτει σε μία κορυφή του κύκλου  $\mathcal{C}$  (μέσω μιας ακμής του ταίριασματος  $\mathcal{M}$ ). Έστω  $v \in V(\mathcal{C})$  μία τέτοια κορυφή του κύκλου  $\mathcal{C}$ . Προφανώς, στην απεικόνιση που έχει

κατασκευαστεί έως τώρα, η κορυφή  $v$  δεν προσπίπτει σε κάποια ακμή του ταιριάσματος και συνεπώς, είτε το αριστερό ή το δεξιό σημείο επαφής (port) της είναι ελεύθερο. Επομένως, η ακμή αυτή του ταιριάσματος μπορεί να ενσωματωθεί στην παραχθείσα απεικόνιση ως οριζόντιο ευθύγραμμο τμήμα μοναδιαίου μήκους στο ελεύθερο σημείο επαφής της κορυφής  $v$ , αφού προηγηθεί μία οριζόντια επέκταση κατά μία μονάδα μήκους ακριβώς δίπλα στο ελεύθερο σημείο επαφής της κορυφής  $v$ . Και σε αυτή την περίπτωση επέκτασης, πιθανές διασταυρώσεις, εξαιτίας της ακμής του κύκλου που προσπίπτει σε κορυφή του χαμηλότερου επίπεδου της απεικόνισης, μπορούν να εξαλειφθούν μετατοπίζοντας τη χαμηλότερη (δεξιότερη, αντίστοιχα) κορυφή της απεικόνισης κατά μία μονάδα μήκους προς τα κάτω (δεξιά, αντίστοιχα), κάθε φορά που μία νέα κορυφή που ανήκει αποκλειστικά στο ταίριασμα  $\mathcal{M}$  ενσωματώνεται στην απεικόνιση. Σημειώνεται ότι το εμβαδόν της απεικόνισης δεν επηρεάζεται από τις επεκτάσεις και παραμένει  $3n/2 \times 3n/2$ , όπου  $n$  είναι το μέγεθος της ένωσης των συνόλων  $V(\mathcal{C})$  και  $V(\mathcal{M})$ . Το παρακάτω θεώρημα συνοψίζει το αποτέλεσμα μας.

**Θεώρημα 14.2.6.** Ένας κύκλος  $\mathcal{C}$  και ένα ταίριασμα  $\mathcal{M}$  επιδέχονται πάντοτε μία GRacSim απεικόνιση σε ένα  $3n/2 \times 3n/2$  ακέραιο πλέγμα, όπου  $n$  είναι το μέγεθος του συνόλου  $V(\mathcal{C})$  και  $V(\mathcal{M})$ . Επιπλέον, η απεικόνιση μπορεί να υπολογιστεί σε γραμμικό χρόνο.

Από τα παραπάνω, έπεται ότι μπορούμε να ορίσουμε μία νέα κλάση RAC γραφημάτων.

**Πόρισμα 14.2.1.** Έστω  $G$  ένα απλό συνδεδεμένο γράφημα, το οποίο μπορεί να αναλυθεί σε ένα ταίριασμα και είτε ένα μονοπάτι ή ένα κύκλο. Το γράφημα  $G$  είναι ένα RAC-γράφημα.

### 14.3 Επίπεδο Γράφημα και Δυϊκό: Μία Ενδιαφέρουσα Παραλλαγή

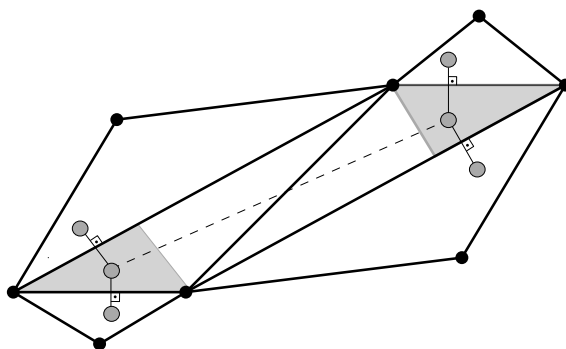
Στην ενότητα αυτή, εξετάζεται το GDual-GRacSim πρόβλημα απεικόνισης γραφημάτων. Το πρόβλημα αυτό μπορεί να θεωρηθεί ως μία παραλλαγή του GRacSim προβλήματος, όπου το πρώτο γράφημα εισόδου (δηλαδή το επίπεδο γράφημα) καθορίζει το δεύτερο (δηλαδή το δυϊκό) και ορίζει περιορισμούς στην απεικόνιση του. Υπενθυμίζεται ότι σύμφωνα με το GDual-GRacSim πρόβλημα, δοθέντος ενός επιπεδικού γραφήματος  $G$ , το ενδιαφέρον επικεντρώνεται στην εύρεση μιας γεωμετρικής απεικόνισης του γραφήματος  $G$  και του ασθενούς δυϊκού  $G^*$  του  $G$  έτσι ώστε: (i) Τα γραφήματα  $G$  και  $G^*$  απεικονίζονται επίπεδα, (ii) κάθε κορυφή του δυϊκού απεικονίζεται στο εσωτερικό της αντίστοιχης όψης του πρωτεύοντος γραφήματος  $G$  και (iii) οι διασταυρώσεις μεταξύ των ακμών του πρωτεύοντος και του δυϊκού γραφήματος σχηματίζουν ορθές γωνίες.

Όπως έχει ήδη αναφερθεί στην Ενότητα 12.4.2, οι Brightwell και Scheinermann [BS93] απέδειξαν ότι το GDual-GRacSim πρόβλημα επιδέχεται πάντοτε λύση, εάν το γράφημα εισόδου είναι ένα τρισυνδεδεμένο επίπεδο γράφημα. Για τη γενικότερη περίπτωση των επίπεδων γραφημάτων, δείχνουμε με τη χρήση ενός παραδείγματος ότι δεν είναι πάντοτε εφικτός



ο υπολογισμός μιας τέτοιας απεικόνισης και περιορίζουμε τη μελέτη μας στην περίπτωση των εξωεπίπεδων γραφημάτων.

Αρχικά, εξετάζουμε την περίπτωση σύμφωνα με την οποία μία επίπεδη απεικόνιση  $\Gamma(G)$  του γραφήματος  $G$  καθορίζεται ως μέρος της εισόδου του προβλήματος και απαιτείται η διατήρηση της στην έξοδο του προβλήματος. Δείχνουμε με ένα παράδειγμα ότι δεν είναι πάντοτε εφικτή η εμφύτευση του δυϊκού γραφήματος στην απεικόνιση  $\Gamma(G)$ , έτσι ώστε να παραχθεί μία GDual-GRacSim απεικόνιση των γραφημάτων  $G$  και  $G^*$ . Το παράδειγμα απεικονίζεται στην Εικόνα 14.9. Παρακάτω, αποδεικνύουμε ότι αν το γράφημα εισόδου είναι επιπεδικό, τότε πάλι το GDual-GRacSim πρόβλημα δεν επιδέχεται πάντοτε λύση.

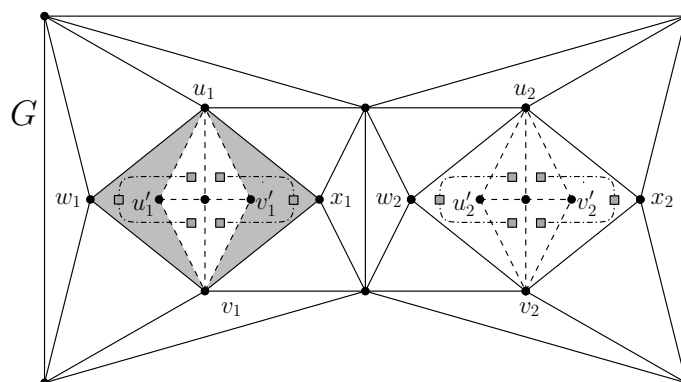


**Εικόνα 14.9:** Η επίπεδη απεικόνιση εισόδου του πρωτεύοντος γραφήματος  $G$  απεικονίζεται με μαύρες σκιασμένες κορυφές και έντονες (bold) ακμές και απαιτείται η διατήρηση της στην έξοδο του προβλήματος. Οι κορυφές του δυϊκού  $G^*$  είναι σκιασμένες με γκρι χρώμα. Τότε, η διακεκομμένη ακμή του δυϊκού αναπόφευκτα θα εισάγει μία τομή σε μη-ορθή γωνία.

**Θεώρημα 14.3.1.** Υπάρχει ένα επίπεδο γράφημα  $G$  με την εξής ιδιότητα: Δεν υπάρχει επίπεδη εμφύτευση του γραφήματος  $G$  για την οποία μία GDual-GRacSim απεικόνιση του  $G$  και ασθενούς δυϊκού  $G^*$  να είναι εφικτή.

*Απόδειξη.* Το γράφημα  $G$  που θα χρησιμοποιηθεί για την απόδειξη του θεωρήματος απεικονίζεται στην Εικόνα 14.10, στην οποία οι κορυφές που απεικονίζονται ως γκρι-σκιασμένα τετράγωνα αντιστοιχούν στις κορυφές του ασθενούς δυϊκού γραφήματος  $G^*$ . Εάν αντικαταστήσουμε τα δύο υπογραφήματα, τα οποία στην Εικόνα 14.10 απεικονίζονται με διακεκομμένες ακμές, με δύο ακμές τότε το παραχθέν γράφημα είναι ένα τρισυνδεδεμένο επίπεδο γράφημα. Συνεπώς, έχει μοναδική επίπεδη απεικόνιση ως προς την επιλογή της εξωτερικής όψης, μετατοπίσεων (translations), περιστροφών (rotations) και επεκτάσεων (stretchings). Συνεπώς, σε κάθε επίπεδη απεικόνιση του γραφήματος  $G$ , είτε οι κορυφές  $u_1w_1v_1x_1$  ή οι κορυφές  $u_2w_2v_2x_2$  θα ορίζουν μία εσωτερική όψη του. Έστω, χωρίς βλάβη της γενικότητας, ότι οι κορυφές  $u_1w_1v_1x_1$  ορίζουν μία εσωτερική όψη του. Στην περίπτωση αυτή, το δυϊκό

γράφημα θα πρέπει να έχει δύο κορυφές στο εσωτερικό των γκρι-σκιασμένων όψεων της Εικόνας 14.10. Καθεμία από αυτές τις δύο κορυφές είναι γειτονική με δύο κορυφές του δυϊκού γραφήματος που βρίσκονται στο εσωτερικό των τριγωνικών όψεων του υπογραφήματος, οι οποίες στην Εικόνα 14.10 απεικονίζονται με διακεκομμένες ακμές. Συνεπώς, σε κάθε RAC-απεικόνιση των  $G$  και  $G^*$ , τα τετράπλευρα  $u_1 w_1 v_1 u'_1$  και  $u_1 x_1 v_1 v'_1$  θα πρέπει να είναι κυρτά, κάτι το οποίο δεν μπορεί να συμβεί.



**Εικόνα 14.10:** Ένα παράδειγμα ενός επίπεδου γραφήματος  $G$ , για το οποίο το πρόβλημα GDual-GRacSim δεν επιδέχεται λύσεις. Οι “προβληματικές” όψεις του είναι σκιασμένες γκρι.

□

**Πόρισμα 14.3.1.** Υπάρχει μία άπειρη οικογένεια επίπεδων γραφημάτων  $\mathcal{F}$  με την εξής ιδιότητα: Για κάθε γράφημα  $G \in \mathcal{F}$  και για κάθε επίπεδη εμφύτευση  $\mathcal{E}(G)$  του γραφήματος  $G$ , μία GDual-GRacSim απεικόνιση του γραφήματος  $G$  και του ασθενούς δυϊκού  $G^*$  δεν είναι εφικτό να κατασκευαστεί.

Στο υπόλοιπο της ενότητας αυτής, η μελέτη εστιάζει στο GDual-GRacSim πρόβλημα στην περίπτωση που το γράφημα εισόδου είναι εξωεπίπεδο. Σημειώνεται ότι ένα εξωεπίπεδο γράφημα δεν είναι απαραίτητα τρισυνδεδεμένο, καθώς η διαγραφή κάθε ζεύγους γειτονικών κορυφών που πρόσκεινται σε κοινή εσωτερική όψη, αλλά δεν είναι διαδοχικές στην εξωτερική όψη, αποσυνδέει το γράφημα. Ωστόσο, κάθε εξωεπίπεδο γράφημα μπορεί να επαυξηθεί σε τρισυνδεδεμένο επίπεδο (αλλά όχι απαραίτητα εξωεπίπεδο) γράφημα με την προσθήκη μιας νέας κορυφής γειτονικής προς όλες τις κορυφές του εξωεπίπεδου γραφήματος. Συνεπώς, κάθε εξωεπιπεδικό γράφημα και ο ασθενής δυϊκός του επιδέχονται πάντοτε μία GDual-GRacSim απεικόνιση λόγω του αποτελέσματος των Brightwell και Scheinermann [BS93]. Ωστόσο, η κατασκευαστική προσέγγιση των Brightwell και Scheinermann απλά αποδεικνύει ότι μία τέτοια απεικόνιση υπάρχει και δεν μπορεί να χρησιμοποιηθεί για την εύρεση

της απεικόνισης, καθώς οι υπολογισμοί που χρησιμοποιούν περιλαμβάνουν ακολουθίες οι οποίες δε συγκλίνουν απαραίτητα σε πεπερασμένο αριθμό βημάτων.

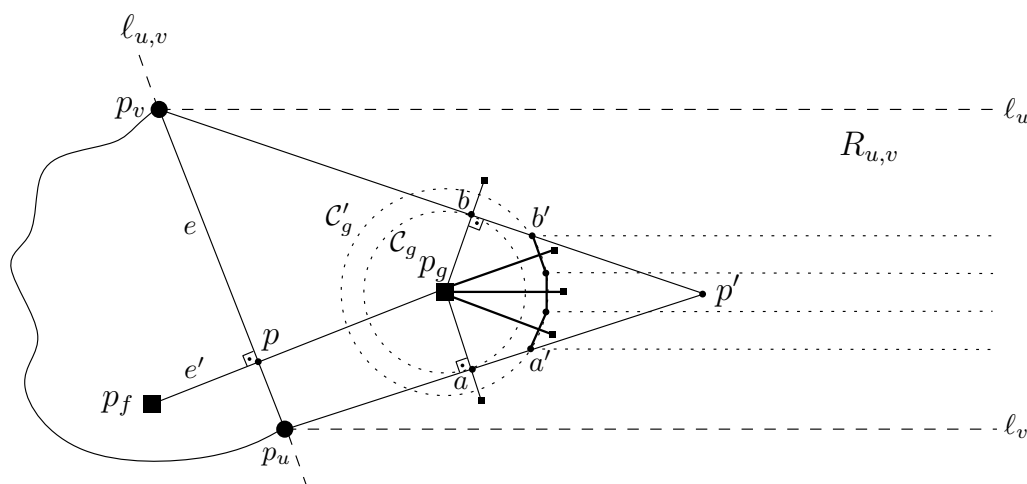
**Θεώρημα 14.3.2.** *Δοθείσας μιας εξωεπιπεδικής εμφύτευσης ενός εξωεπίπεδου γραφήματος  $G$ , είναι πάντοτε εφικτή η εύρεση μιας GDual-GRacSim απεικόνισης του γραφήματος  $G$  και του ασθενή δυϊκού  $G^*$ .*

*Απόδειξη.* Η απόδειξη δίνεται με μία αναδρομική γεωμετρική κατασκευή, η οποία υπολογίζει μία GDual-GRacSim απεικόνιση του γραφήματος  $G$  και του ασθενή δυϊκού  $G^*$ . Έστω  $(u, v)$  μία ακμή του εξωεπίπεδου γραφήματος, η οποία δεν ανήκει στην εξωτερική όψη του και έστω  $f$  και  $g$  οι όψεις στα αριστερά και δεξιά της ακμής αυτής, αντίστοιχα, όπως κινούμαστε κατά μήκος της ακμής  $(u, v)$  από την κορυφή  $u$  προς την κορυφή  $v$ . Προφανώς, η ακμή  $(f, g)$  είναι μία ακμή του ασθενούς δυϊκού γραφήματος  $G^*$ . Αφού το ασθενές δυϊκό γράφημα κάθε εξωεπίπεδου γραφήματος είναι ένα δένδρο, από τη διαγραφή της ακμής  $(f, g)$  προκύπτουν δύο δένδρα  $T_f$  και  $T_g$ , τα οποία μπορούμε να θεωρήσουμε ότι έχουν ρίζα τις κορυφές  $f$  και  $g$  του  $G^*$ , αντίστοιχα. Για το αναδρομικό βήμα του αλγορίθμου απεικόνισης μας, υποθέτουμε ότι έχουμε ήδη υπολογίσει μία GDual-GRacSim απεικόνιση του γραφήματος  $T_f$  και του αντίστοιχου υπογραφήματος του  $G$ , η οποία ικανοποιεί τις παρακάτω αναλλοίωτες ιδιότητες:

- I-P1: Η ακμή  $(u, v)$  απεικονίζεται στην εξωτερική όψη της GDual-GSimRAC απεικόνισης του  $T_f$ . Έστω ότι οι κορυφές  $u$  και  $v$  απεικονίζονται στα σημεία  $p_u$  και  $p_v$ , αντίστοιχα. Έστω επίσης  $\ell_{u,v}$  η ευθεία που ορίζεται από τα σημεία  $p_u$  και  $p_v$ .
- I-P2: Αν η κορυφή-όψη  $f$  απεικονίζεται στο σημείο  $p_f$ , τότε η ευθεία από το σημείο  $p_f$  που είναι κάθετη στην ευθεία  $\ell_{u,v}$  τέμνει το ευθύγραμμο τμήμα  $p_u p_v$ . Έστω  $p$  το σημείο τομής τους.
- I-P3: Υπάρχουν δύο παράλληλες ημιευθείες  $\ell_u$  και  $\ell_v$  που διέρχονται από τα σημεία  $p_u$  και  $p_v$ , αντίστοιχα, οι οποίες ορίζουν μία λωρίδα (semi-strip)  $R_{u,v}$  στα δεξιά του ευθύγραμμου τμήματος  $p_u p_v$  που δεν τέμνει την απεικόνιση που έχει έως τώρα κατασκευαστεί.

Στη συνέχεια, περιγράφουμε τον τρόπο με τον οποίο αναδρομικά μπορούμε να κατασκευάσουμε μία απεικόνιση του δένδρου  $T_g$  και του αντίστοιχου υπογραφήματος του  $G$ , έτσι ώστε η συνολική απεικόνιση να είναι μία GDual-GRacSim απεικόνιση του γραφήματος  $G$  και του δυϊκού  $G^*$  (βλ. Εικόνα 14.11). Έστω  $p_g$  ένα σημείο της λωρίδας  $R_{u,v}$  το οποίο ανήκει στην κάθετη προς το ευθύγραμμο τμήμα  $p_u p_v$  ευθεία που διέρχεται από το σημείο  $p$ . Το ευθύγραμμο τμήμα που αντιστοιχεί στην ακμή  $(f, g)$  του δυϊκού γραφήματος τέμνει υπό ορθή γωνία το αντίστοιχο ευθύγραμμο τμήμα που αντιστοιχεί στην ακμή  $(u, v)$  του πρωτεύοντος γραφήματος  $G$ , όπως απαιτείται από το GDual-GRacSim πρόβλημα. Εάν η κορυφή  $g$

είναι φύλλο στο δένδρο  $T_g$  (δηλαδή όλες οι ακμές της όψης  $g$  εξαιρουμένης της ακμής  $(u, v)$  πρόσκεινται στην εξωτερική όψη του  $G$ ), τότε μπορούμε να απεικονίσουμε όλες τις ακμές της όψης  $g$  ως μία πολυγραμμική αλυσίδα κατάλληλου αριθμού σημείων με άκρα τα σημεία  $p_u$  και  $p_v$  και η οποία περικλείει το σημείο  $p_g$ .



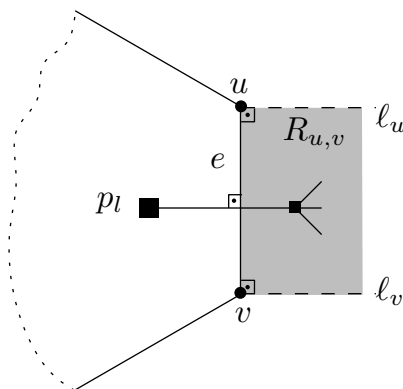
Εικόνα 14.11: Το αναδρομικό βήμα του αλγορίθμου μας.

Στη συνέχεια, θεωρούμε την πιο ενδιαφέρουσα περίπτωση όπου η κορυφή  $g$  δεν είναι φύλλο στο δένδρο  $T_g$ . Στην περίπτωση αυτή, θεωρούμε δύο ομόκεντρους κύκλους, έστω  $C_g$  και  $C'_g$ , με κοινό κέντρο το σημείο  $p_g$  και κατάλληλα επιλεγμένες ακτίνες, έτσι ώστε και οι δύο κύκλοι να βρίσκονται εξ'ολοκλήρου στο εσωτερικό της λωρίδας  $R_{u,v}$  (δηλαδή να μην τέμνουν ούτε την ευθεία  $\ell_u$  ούτε την ευθεία  $\ell_v$ ). Έστω ότι ο κύκλος  $C'_g$  είναι ο εξωτερικός των δύο ομόκεντρων κύκλων. Φέρουμε από το σημείο  $p_u$  την εφαπτομένη στον κύκλο  $C_g$  και έστω  $a$  το σημείο στο οποίο εφάπτεται του κύκλου  $C_g$  και  $a'$  το σημείο στα δεξιά του  $a$ , όπου η εφαπτομένη αυτή τέμνει τον κύκλο  $C'_g$  (βλ. Εικόνα 14.11). Ομοίως, ορίζουμε τα σημεία  $b$  και  $b'$  με βάση την εφαπτόμενη από το σημείο  $p_v$  προς τον κύκλο  $C_g$ .

Έστω  $k \geq 4$  ο αριθμός των κορυφών που ορίζουν την όψη  $g$ . Η περίπτωση όπου η όψη  $g$  ορίζεται από τρία σημεία (δηλαδή  $k = 3$ ) θα εξεταστεί αργότερα. Εισάγουμε  $k - 4$  σημεία κατά μήκος του τόξου  $(a', b')$ , που είναι πιο απομακρυσμένο από το ευθύγραμμο τμήμα  $p_u p_v$ . Τα σημεία αυτά, έστω  $\{p_i \mid 1 \leq i \leq k - 4\}$ , μαζί με τα σημεία  $p_u, p_v, a'$  και  $b'$  σχηματίζουν την όψη  $g$ . Σημειώνεται ότι από το σημείο  $p_g$ , μπορούμε να φέρουμε κάθετες ευθείες προς κάθε ακμή της όψης  $g$ , καθώς τα ευθύγραμμα τμήματα  $p_g a$  και  $p_g b$  είναι κάθετα στα ευθύγραμμα τμήματα  $p_u a'$  και  $p_v b'$ , αντίστοιχα. Οι υπόλοιπες ακμές της όψης  $g$  είναι χορδές του κύκλου  $C'_g$  και συνεπώς, μπορούμε να φέρουμε κάθετες στο μέσο τους από το κέντρο  $p_g$  του κύκλου  $C'_g$ . Αν από καθένα από τα νεοεισαχθέντα σημεία της όψης  $g$  φέρουμε μία ημιευθεία παράλληλη προς την ημιευθεία  $\ell_u$  και στο εσωτερικό της λωρίδας  $R_{u,v}$ , τότε δεν είναι δύ-

σκολο να παρατηρήσουμε ότι όλες οι αναλλοίωτες ιδιότητες που περιγράφηκαν παραπάνω ισχύουν για κάθε όψη-παιδί της όψης  $g$  στο υποδένδρο  $T_g$  του ασθενούς δυϊκού του  $G$ . Συνεπώς, ο αλγόριθμος μας μπορεί αναδρομικά να εφαρμοστεί. Η περίπτωση όπου η όψη  $g$  ορίζεται από  $k = 3$  σημεία αντιμετωπίζεται ως εξής. Χρησιμοποιούμε την τομή, έστω  $p'$ , των δύο εφαπτόμενων ως το τρίτο σημείο της τριγωνικής όψης  $g$ . Σημειώνεται ότι είναι πάντοτε εφικτό να επιλέξουμε το σημείο  $p_g$ , έτσι ώστε να είναι αρκετά κοντά στο ευθύγραμμο τμήμα  $p_u p_v$  και κατάλληλη μικρή ακτίνα για τον κύκλο  $C_g$ , έτσι ώστε το σημείο  $p'$  να βρίσκεται στο εσωτερικό της λωρίδας  $R_{u,v}$ .

Έχοντας περιγράψει το αναδρομικό βήμα του αλγορίθμου, απομένει να περιγράψουμε τον τρόπο με τον οποίο ξεκινάει η αναδρομή (βλ. Εικόνα 14.12). Πιο συγκεκριμένα, η αναδρομή ξεκινάει από μία οποιαδήποτε όψη, έστω  $l$ , του γραφήματος  $G$ , η οποία είναι φύλλο στο ασθενές δυϊκό δένδρο. Απεικονίζουμε την όψη  $l$  ως κανονικό πολύγωνο, με την κορυφή-όψη  $l$  στο κέντρο του, έστω  $p_l$ . Έστω  $e = (u, v)$  η μόνη ακμή της όψης  $l$  που είναι εσωτερική στην εξωεπιπεδική εμφύτευση του  $G$ . Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι η ακμή  $e$  απεικονίζεται ως κάθετο ευθύγραμμο τμήμα. Ακολουθώντας, φέρουμε οριζόντιες ημιευθείες, έστω  $\ell_u$  και  $\ell_v$ , από τα άκρα  $u$  και  $v$ , αντίστοιχα, της ακμής  $e$  για να ορίζουμε τη λωρίδα  $R_{u,v}$ . Ακολουθώντας, ο αλγόριθμος μας μπορεί αναδρομικά να εφαρμοστεί για την απεικόνιση των υπόλοιπων όψεων του γραφήματος  $G$  και του ασθενούς δυϊκού  $G^*$ .



Εικόνα 14.12: Η βάση του αναδρομικού αλγορίθμου.

□

Σημειώνεται ότι η παραχθείσα GDual-GRacSim απεικόνιση του γραφήματος  $G$  και του ασθενούς δυϊκού  $G^*$  αποδεικνύει απλά ότι η κατασκευή μιας τέτοιας απεικόνισης είναι πάντοτε εφικτή. Ωστόσο, η ίδια η απεικόνιση δεν είναι ιδιαίτερα ελκυστική, καθώς το ύψος των λωρίδων σύντομα γίνεται πολύ μικρό. Ωστόσο, αποτελεί ένα πρώτο βήμα προς την παραγωγή ποιοτικότερων απεικονίσεων. Επίσης, σημειώνεται ότι ο αλγόριθμος μας εκτελεί γραμμικού αριθμού “υπολογισμούς σημείων”, καθώς για κάθε κορυφή-όψη του ασθενούς δυ-

ϊκού δένδρου, οι απαιτούμενοι υπολογισμοί είναι ανάλογοι του βαθμού της κορυφής-όψης. Ωστόσο, οι συντεταγμένες κάποιων σημείων μπορεί να είναι άρρητοι αριθμοί.

## 15 Το Πρόβλημα της Μεγιστοποίησης της Συνολικής Γωνιακής Ευκρίνειας

Σε αυτό το κεφάλαιο, θα μελετηθεί το πρόβλημα της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας μιας απεικόνισης ενός γραφήματος, δηλαδή της μεγιστοποίησης του ελαχίστου της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών. Από όσο είμαστε σε θέση να γνωρίζουμε, είναι η πρώτη φορά που μελετάται το συγκεκριμένο πρόβλημα. Η βασική συμβολή αυτού του κεφαλαίου είναι απεικονίσεις βέλτιστης συνολικής γωνιακής ευκρίνειας για πλήρη γραφήματα (κυκλικές απεικονίσεις) και πλήρη διμερή γραφήματα (απεικονίσεις δύο επιπέδων). Επιπρόσθετα, παρουσιάζουμε και αξιολογούμε πειραματικά ένα force-directed αλγόριθμο για την κατασκευή απεικονίσεων υψηλής συνολικής γωνιακής ευκρίνειας.

### 15.1 Προκαταρκτικές Έννοιες και Ορολογία

Δοθείσας μιας απεικόνισης  $\Gamma(G)$  ενός μη-κατευθυνόμενου γραφήματος  $G = (V, E)$ , συμβολίζουμε με  $p_u = (x(u), y(u))$  τη θέση της κορυφής  $u \in V$  στο επίπεδο. Το διάνυσμα από το σημείο  $p_u$  προς το σημείο  $p_v$ , το οποίο έχει μοναδιαίο μήκος συμβολίζεται με  $\overrightarrow{p_u p_v}$ , όπου  $u, v \in V$ .

Δοθέντος ενός ζεύγους σημείων  $q_1, q_2 \in \mathbb{R}^2$ , με μία μικρή παραβίαση της σημειολογίας, συμβολίζουμε με  $\|q_1 - q_2\|$  την Ευκλείδεια απόσταση μεταξύ των σημείων  $q_1$  και  $q_2$ . Συμβολίζουμε, επίσης, με  $\overline{q_1 q_2}$  το ευθύγραμμο τμήμα που ορίζεται από τα σημεία  $q_1$  και  $q_2$ .

Έστω  $\vec{\alpha}$  και  $\vec{\gamma}$  δύο διανύσματα. Το διάνυσμα που διχοτομεί τη γωνία που σχηματίζεται ανάμεσα στα διανύσματα  $\vec{\alpha}$  και  $\vec{\gamma}$  είναι  $\frac{\vec{\alpha}}{\|\vec{\alpha}\|} + \frac{\vec{\gamma}}{\|\vec{\gamma}\|}$ . Συμβολίζουμε με  $\text{Bsc}(\vec{\alpha}, \vec{\gamma})$ , το αντίστοιχο διάνυσμα μοναδιαίου μήκους. Δοθέντος ενός διανύσματος  $\vec{\beta}$ , συμβολίζουμε με  $\text{Perp}(\vec{\beta})$  το μοναδιαίο μήκος διάνυσμα, το οποίο είναι κάθετο στο διάνυσμα  $\vec{\beta}$  και προηγείται κατά τη φορά των δεικτών του ρολογιού. Μερικές από τις αποδείξεις μας κάνουν χρήση των παρακάτω βασικών γεωμετρικών ιδιοτήτων:

$$\tan(\omega_1 \pm \omega_2) = \frac{\tan \omega_1 \pm \tan \omega_2}{1 \pm (-\tan \omega_1 \cdot \tan \omega_2)} \quad (15.1)$$

$$\tan(\omega/2) = \frac{\sin \omega}{1 + \cos \omega} \quad (15.2)$$

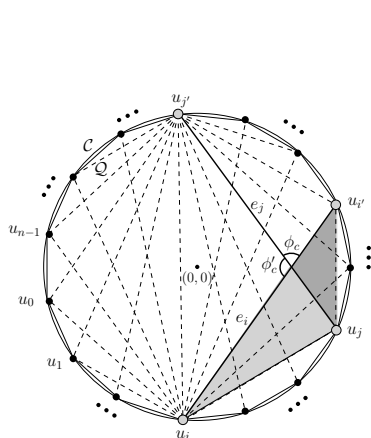
$$\omega \in (0, \frac{\pi}{2}) \Rightarrow \tan \omega > \omega \quad (15.3)$$

## 15.2 Απεικονίσεις με Βέλτιστη Συνολική Γωνιακή Ευκρίνεια για Πλήρη και Πλήρη Διμερή Γραφήματα

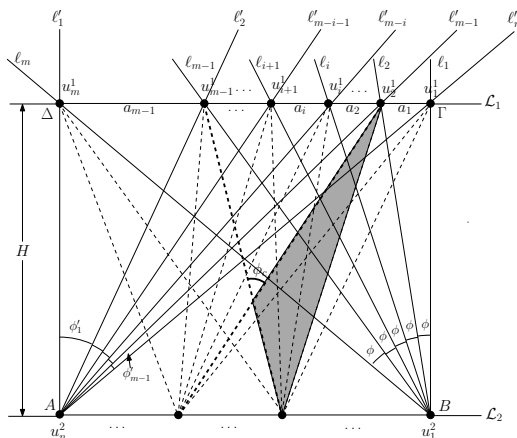
Σε αυτή την ενότητα, αρχικά ορίζουμε τυπικά τη συνολική γωνιακή ευκρίνεια μιας απεικόνισης. Προφανώς, το πρόβλημα του καθορισμού της μέγιστης συνολικής γωνιακής ευκρίνειας για την περίπτωση των επίπεδων γεωμετρικών απεικονίσεων των επίπεδων γραφημάτων είναι ισοδύναμο με το αντίστοιχο πρόβλημα καθορισμού της μέγιστης γωνιακής ευκρίνειας, το οποίο είναι ένα πρόβλημα που έχει μελετηθεί αρκετά στη βιβλιογραφία της Απεικόνισης Γραφημάτων. Επομένως, επικεντρώνουμε τη μελέτη μας σε μη-επίπεδες κλάσεις γραφημάτων. Ως ένα πρώτο βήμα προς την επίλυση του προβλήματος της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας σε τυχαία μη-επίπεδα γραφήματα, μελετάμε τις διαισθητικά πιο απλές περιπτώσεις των πλήρων και πλήρων διμερών γραφημάτων για τις οποίες αναζητούμε κυκλικές και 2-επιπέδων απεικονίσεις μέγιστης συνολικής γωνιακής ευκρίνειας, αντίστοιχα. Σημειώνεται ότι τέτοιες απεικονίσεις είναι συνήθεις για την οπτικοποίηση πλήρων και πλήρων διμερών γραφημάτων. Πιο συγκεκριμένα, παρουσιάζουμε δύο μεθόδους για την παραγωγή απεικονίσεων ασυμπτωτικά βέλτιστης συνολικής γωνιακής ευκρίνειας για πλήρη και πλήρη διμερή γραφήματα. Αποδεικνύουμε ότι (i) κάθε πλήρες γράφημα  $K_n$  επιδέχεται μία κυκλική απεικόνιση συνολικής γωνιακής ευκρίνειας  $O(1/n)$ , η οποία είναι ασυμπτωτικά βέλτιστη και (ii) κάθε πλήρες διμερές γράφημα  $K_{n,m}$  επιδέχεται μία πολυεπίπεδη απεικόνιση συνολικής γωνιακής ευκρίνειας  $O(1/\max\{n, m\})$ , η οποία είναι, επίσης, ασυμπτωτικά βέλτιστη.

Θεωρούμε αρχικά την περίπτωση των πλήρων γραφημάτων. Έστω  $K_n = (V, E)$  ένα πλήρες γράφημα, όπου  $V = \{u_0, u_1, \dots, u_{n-1}\}$  και  $E = V \times V$ . Ο στόχος μας είναι η κατασκευή μιας κυκλικής απεικόνισης του γραφήματος  $K_n$  μέγιστης συνολικής γωνιακής ευκρίνειας. Η προσέγγισή μας είναι κατασκευαστική και αρκετά συνήθης κατά την απεικόνιση πλήρων γραφημάτων. Μία παρόμοια κατασκευή παρουσιάστηκε από τους Formann et al. [FHH<sup>+</sup>93] για την κατασκευή βέλτιστων απεικονίσεων πλήρων γραφημάτων σε όρους γωνιακής ευκρίνειας. Θεωρούμε έναν κύκλο  $\mathcal{C}$ , ακτίνας  $r_c > 0$ , με κέντρο το σημείο  $(0, 0)$  και εγγράφουμε σε αυτόν ένα κανονικό  $n$ -γώνιο  $\mathcal{Q}$ . Στην κατασκευή μας, οι κορυφές του γραφήματος  $K_n$  ταυτίζονται με τις κορυφές του  $n$ -γώνου  $\mathcal{Q}$ . Χωρίς βλάβη της γενικότητας, υποθέτουμε περαιτέρω ότι οι κορυφές  $u_0, u_1, \dots, u_{n-1}$  εμφανίζονται με αυτή τη σειρά, κατά την αντίθετη φορά των δεικτών του ρολογιού γύρω από το σημείο  $(0, 0)$ , όπως φαίνεται στην Εικόνα 15.1i.





(i) Μία κυκλική απεικόνιση του γραφήματος  $K_n$ .



(ii) Μία 2-επιπέδων απεικόνιση του γραφήματος  $K_{m,n}$ .

Σχήμα 15.1: Κατασκευαστικές προσεγγίσεις

**Θεώρημα 15.2.1.** Ένα πλήρες γράφημα  $K_n$  επιδέχεται μία κυκλική απεικόνιση συνολικής γωνιακής ευκρίνειας  $O(\frac{1}{n})$ , η οποία είναι ασυμπτωτικά βέλτιστη.

*Απόδειξη.* Θα αποδειχθεί ότι η γωνιακή ευκρίνεια της απεικόνισης του γραφήματος  $K_n$  που παρουσιάστηκε παραπάνω είναι  $\frac{\pi}{n}$ , ενώ η γωνιακή ευκρίνεια τομών του είναι  $\frac{2\pi}{n}$ . Αρχικά, παρατηρούμε ότι το μήκος του τόξου του κύκλου  $C$  που ενώνει δύο διαδοχικές κορυφές  $u_i$  και  $u_{(i+1) \bmod n}$  ισούται με  $\frac{2\pi r_c}{n}$ , για κάθε  $i = 0, 1, \dots, n - 1$ . Επομένως, η γωνιακή ευκρίνεια της απεικόνισης είναι  $\frac{\pi}{n}$ , όπως επιθυμούσαμε. Έστω τώρα ότι  $e_i = (u_i, u_{i'})$  και  $e_j = (u_j, u_{j'})$  είναι δύο τεμνόμενες ακμές. Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι  $i < j < i' < j'$ , όπως στην Εικόνα 15.1i. Η τομή των ακμών  $e_i$  και  $e_j$  ορίζει δύο γωνίες  $\phi_c$  και  $\phi'_c$ , τέτοιες ώστε  $\phi_c + \phi'_c = \pi$ . Στην Εικόνα 15.1i, η γωνία  $\phi_c$  είναι εξωτερική στο τρίγωνο που σχηματίζεται από την τομή των ακμών  $e_i$  και  $e_j$  και τις κορυφές  $u_j$  και  $u_{i'}$  (βλ. τα γκρι-σκιασμένα τρίγωνα της Εικόνας 15.1i). Επομένως:  $\phi_c = (j' - i') \frac{\pi}{n} + (j - i) \frac{\pi}{n}$ . Όμοια,  $\phi'_c = (i' - j) \frac{\pi}{n} + (n - (j' - i)) \frac{\pi}{n}$ . Στην περίπτωση που  $j = (i + 1) \bmod n$  και  $j' = (i' + 1) \bmod n$ , δηλαδή εάν οι κορυφές  $u_i$  ( $u_{i'}$ , αντίστοιχα) και  $u_j$  ( $u_{j'}$ , αντίστοιχα) είναι διαδοχικές, η γωνία  $\phi_c$  παίρνει την ελάχιστη τιμή της, η οποία ισούται με  $\frac{2\pi}{n}$ . Όμοια, μπορούμε να αποδείξουμε ότι η ελάχιστη τιμή της γωνίας  $\phi'_c$  είναι, επίσης,  $\frac{2\pi}{n}$ . Συνεπώς, η γωνιακή ευκρίνεια τομών της απεικόνισης είναι  $\frac{2\pi}{n}$ . Σημειώνεται ότι το ίδιο άνω φράγμα για τη γωνιακή ευκρίνεια τομών του γραφήματος  $K_n$  έχει αποδειχθεί ανεξάρτητα από τους Di Giacomo et al. στην εργασία [DDE<sup>+</sup>12].  $\square$

Στη συνέχεια, θεωρούμε την κλάση των πλήρων διμερών γραφημάτων. Δεδομένου ότι ένα πλήρες διμερές γράφημα  $n$ -κορυφών αποτελεί υπογράφημα ενός πλήρους γραφήματος  $n$ -κορυφών, μία απεικόνιση συνολικής γωνιακής ευκρίνειας  $\Theta(\frac{1}{n})$  για το πλήρες διμερές

γράφημα παράγεται από τη βέλτιστη απεικόνιση του πλήρους γραφήματος. Ωστόσο, στην περίπτωση που οι κορυφές του γραφήματος πρέπει να έχουν ακέραιες συντεταγμένες (δηλαδή περιοριζόμαστε σε απεικονίσεις σε πλέγματα), λίγα αποτελέσματα είναι γνωστά για το απαιτούμενο εμβαδόν αυτής της απεικόνισης. Ένα άνω φράγμα της τάξης του  $O(n^3)$  έπεται από την εργασία [BT04]. Αυτό μας δίνει το κίνητρο να μελετήσουμε ξεχωριστά την κλάση των πλήρων διμερών γραφημάτων, καθώς μπορούμε να βελτιώσουμε δραστικά το παραπάνω φράγμα. Σημειώνεται ότι η σχέση μεταξύ γωνιακής ευκρίνειας ή γωνιακής ευκρίνειας τομών και εμβαδού έχει μελετηθεί στο παρελθόν. Οι Malitz και Parakostas [MP94] έδειξαν ότι υπάρχουν γραφήματα των οποίων οι γεωμετρικές εμφυτεύσεις με καλή γωνιακή ευκρίνεια, απαιτούν πάντα εκθετικό εμβαδόν. Ο ισχυρισμός παραμένει αληθής, ακόμα και αν οι ακμές απεικονιστούν με κυκλικά τόξα, αντί για ευθείες γραμμές [CDGK99]. Πιο πρόσφατα, οι Angelini et al. [ACD<sup>+</sup>09] έδειξαν κατασκευαστικά ότι υπάρχουν γραφήματα των οποίων οι γεωμετρικές ανερχόμενες RAC-απεικονίσεις απαιτούν εκθετικό εμβαδόν.

Έστω  $K_{m,n} = (V_1 \cup V_2, E)$  ένα πλήρες διμερές γράφημα, όπου  $V_1 = \{u_1^1, u_2^1, \dots, u_m^1\}$ ,  $V_2 = \{u_1^2, u_2^2, \dots, u_n^2\}$  και  $E = V_1 \times V_2$ . Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι  $m \geq n$ . Όπως ήδη αναφέραμε, αναζητούμε απεικονίσεις 2-επιπέδων στις οποίες οι κορυφές των συνόλων  $V_1$  και  $V_2$  βρίσκονται σε δύο παράλληλες ευθείες, έστω  $\mathcal{L}_1$  και  $\mathcal{L}_2$ , αντίστοιχα. Για την απλοποίηση της παρουσίασης, υποθέτουμε ότι και οι δύο ευθείες  $\mathcal{L}_1$  και  $\mathcal{L}_2$  είναι οριζόντιες. Αρχικά, παρουσιάζουμε μία γεωμετρική κατασκευή σύμφωνα με την οποία οι κορυφές δεν απαιτείται να βρίσκονται σε σημεία του πλέγματος και στη συνέχεια, θα δείξουμε τον τρόπο με τον οποίο θα τη μετατρέψουμε σε μία απεικόνιση σε πλέγμα.

Αρχικά, θεωρούμε ένα τετράγωνο  $\mathcal{R} = AB\Gamma\Delta$ , του οποίου οι πάνω και κάτω πλευρές συμπίπτουν με τις ευθείες  $\mathcal{L}_1$  και  $\mathcal{L}_2$ , αντίστοιχα (βλ. Εικόνα 15.1ii). Έστω  $H$  το ύψος (και πλάτος) του τετραγώνου  $\mathcal{R}$ . Σύμφωνα με την προσέγγισή μας, οι κορυφές του συνόλου  $V_1$  ( $V_2$ , αντίστοιχα) βρίσκονται κατά μήκος της πλευράς  $\Gamma\Delta$  ( $AB$ , αντίστοιχα) του τετραγώνου  $\mathcal{R}$ . Για να καθορίσουμε τις ακριβείς θέσεις των κορυφών  $u_1^1, u_2^1, \dots, u_m^1$  κατά μήκος της πλευράς  $\Gamma\Delta$ , κατασκευάζουμε πρώτα μία δέσμη (bundle) από  $m$  ημιευθείες, έστω  $\ell_1, \dots, \ell_m$ , καθεμία από τις οποίες ξεκινάει από την κορυφή  $B$  και τέμνει την πλευρά  $\Gamma\Delta$  του τετραγώνου  $\mathcal{R}$ , έτσι ώστε η γωνία που σχηματίζεται από την πλευρά  $B\Gamma$  και την ημιευθεία  $\ell_i$  να ισούται με  $\frac{(i-1) \cdot \widehat{\Delta B\Gamma}}{m-1}$ , για κάθε  $i = 1, \dots, m$ . Αυτές οι ημιευθείες διαμερίζουν τη γωνία  $\widehat{\Delta B\Gamma}$  σε  $m-1$  γωνίες, καθεμία από τις οποίες είναι ίση με  $\frac{\pi}{4 \cdot (m-1)}$ , καθώς  $\widehat{\Delta B\Gamma} = \pi/4$ . Έστω ότι  $\phi = \frac{\pi}{4 \cdot (m-1)}$ . Ακολουθώντας, τοποθετούμε την κορυφή  $u_i^1$  στο σημείο τομής της ημιευθείας  $\ell_i$  και της πλευράς  $\Gamma\Delta$ , για κάθε  $i = 1, \dots, m$  (βλ. Εικόνα 15.1ii). Συμμετρικά, μπορούμε να υπολογίσουμε τις θέσεις των κορυφών του συνόλου  $V_2$  κατά μήκος της πλευράς  $AB$  του τετραγώνου  $\mathcal{R}$ .

Συμβολίζουμε με  $a_i$  την οριζόντια απόσταση μεταξύ δύο διαδοχικών κορυφών  $u_i^1$  και

$u_{i+1}^1, i = 1, \dots, m-1$ . Ορίζουμε, επίσης, μία επιπλέον δέσμη  $m$  ημιευθειών, έστω  $\ell'_1, \dots, \ell'_m$ , οι οποίες ξεκινούν από την κορυφή  $A$ . Πιο συγκεκριμένα, η ημιευθεία  $\ell'_i$  ξεκινάει από την κορυφή  $A$  και διέρχεται από το σημείο τομής της ημιευθείας  $l_{m-i}$  και της πλευράς  $\Gamma\Delta$  (σ.σ. την κορυφή  $u_{m-i}^1$ ), για κάθε  $i = 1, \dots, m$  (βλ. Εικόνα 15.1ii). Έστω  $\phi'_i$  η γωνία που σχηματίζεται από δύο διαδοχικές ημιευθείες  $\ell'_i$  και  $\ell'_{i+1}$ , για κάθε  $i = 1, \dots, m-1$ . Μπορούμε τώρα να εξετάσουμε μερικές γεωμετρικές ιδιότητες της προτεινόμενης κατασκευής.

**Λήμμα 15.2.1.** Για κάθε  $i = 1, 2, \dots, m-1$ , ισχύει ότι  $a_{i-1} < a_i$ .

*Απόδειξη.* Με επαγωγή. Για την επαγωγική βάση, θα δείξουμε ότι  $a_1 < a_2$ . Αρχικά, παρατηρούμε ότι  $a_1 = H \tan \phi$  και  $a_1 + a_2 = H \tan 2\phi$ . Επομένως:

$$a_2 = H(\tan 2\phi - \tan \phi) \stackrel{(15.1)}{=} a_1 \cdot (1 + \tan 2\phi \cdot \tan \phi)$$

Ωστόσο, και οι δύο γωνίες  $\tan \phi$  και  $\tan 2\phi$  είναι μεγαλύτερες του μηδενός, το οποίο άμεσα συνεπάγεται ότι  $a_1 < a_2$ . Για την επαγωγική υπόθεση, υποθέτουμε ότι  $\forall k$ , με  $k < m-1$ , ισχύει ότι  $a_{k-1} < a_k$  και αποδεικνύουμε ότι  $a_k < a_{k+1}$ . Προφανώς,  $a_1 + \dots + a_k = H \tan k\phi$ . Λόγω της Εξίσωσης 15.1 και όμοια με την επαγωγική βάση ισχύουν τα παρακάτω:

- $a_{k+1} = H \tan \phi \cdot (1 + \tan(k+1)\phi \cdot \tan k\phi)$
- $a_k = H \tan \phi \cdot (1 + \tan(k-1)\phi \cdot \tan k\phi)$

Για να ολοκληρωθεί η απόδειξη, παρατηρούμε ότι  $(k+1)\phi > (k-1)\phi$ . □

**Λήμμα 15.2.2.** Για κάθε  $i = 2, \dots, m-1$ , ισχύει ότι  $\phi'_{i-1} > \phi'_i$ .

*Απόδειξη.* Με επαγωγή. Για την επαγωγική βάση, θα πρέπει να δείξουμε ότι  $\phi'_1 > \phi'_2$  ή ισοδύναμα ότι  $\tan \phi'_1 > \tan \phi'_2$ . Ισχύει ότι  $\tan \phi'_1 = a_{m-1}/H$  και  $\tan(\phi'_1 + \phi'_2) = (a_{m-1} + a_{m-2})/H$ . Συνδυάζοντας τις δύο αυτές σχέσεις με την Εξίσωση 15.1 έχουμε ότι  $\tan \phi'_2 = \frac{H a_{m-2}}{H^2 + a_{m-1}^2 + a_{m-1} a_{m-2}}$ . Επομένως:

$$\tan \phi'_1 > \tan \phi'_2 \Leftrightarrow$$

$$H^2(a_{m-1} - a_{m-2}) + a_{m-1}^3 + a_{m-1}^2 a_{m-2} > 0,$$

το οποίο ισχύει τετριμμένα λόγω του Λήμματος 15.2.1. Για την επαγωγική υπόθεση, υποθέτουμε ότι  $\forall k, k < m-1$ , ισχύει ότι  $\phi'_{k-1} > \phi'_k$  και θα δείξουμε ότι  $\phi'_k > \phi'_{k+1}$ . Παρατηρούμε ότι:

$$\begin{aligned} \tan \phi'_k &= \frac{\tan(\phi'_1 + \dots + \phi'_k) - \tan(\phi'_1 + \dots + \phi'_{k-1})}{1 + \tan(\phi'_1 + \dots + \phi'_k) \cdot \tan(\phi'_1 + \dots + \phi'_{k-1})} \\ &= \frac{H a_{m-k}}{H^2 + (a_{m-1} + \dots + a_{m-k})(a_{m-1} + \dots + a_{m-k+1})} \end{aligned}$$

$$\tan(\phi'_{k+1}) = \frac{H a_{m-(k+1)}}{H^2 + (a_{m-1} + \dots + a_{m-(k+1)})(a_{m-1} + \dots + a_{m-k})}$$

Από το Λήμμα 15.2.1 έχουμε ότι  $(a_{m-1} + \dots + a_{m-(k-1)}) < (a_{m-1} + \dots + a_{m-(k+1)})$  και  $H \cdot a_{m-k} > H \cdot a_{m-(k+1)}$ . Επομένως,  $\tan \phi'_k > \tan \phi'_{k+1}$ .  $\square$

**Λήμμα 15.2.3.** Ισχύει ότι  $\phi'_{m-1} \leq \phi$ .

Απόδειξη. Ισοδύναμα αποδεικνύουμε ότι  $\tan \phi'_{m-1} \leq \tan \phi$ .

$$\begin{aligned} \tan \phi'_{m-1} \leq \frac{a_1}{H} &\Leftrightarrow \frac{\frac{a_1}{H}}{1 + \frac{a_1 + \dots + a_{m-1}}{H} \cdot \frac{a_2 + \dots + a_{m-1}}{H}} \leq \frac{a_1}{H} \\ &\Leftrightarrow \frac{1}{1 + \frac{H - a_1}{H}} \leq 1 \\ &\Leftrightarrow a_1 \leq H \end{aligned}$$

το οποίο προφανώς ισχύει.  $\square$

**Λήμμα 15.2.4.** Η γωνία  $\phi'_{m-1}$  είναι η μικρότερη από όλες τις γωνίες που σχηματίζονται στην απεικόνιση.

Απόδειξη. Από το Λήμμα 15.2.2 έπεται ότι η γωνία  $\phi'_{m-1}$  είναι η μικρότερη από όλες τις γωνίες  $\phi'_i, i = 1, \dots, m-1$ . Επιπρόσθετα, κάθε γωνία  $\phi'_i$  μεγαλώνει, καθώς το άκρο της δέσμης (σ.σ. η κορυφή  $u_n^2$ ) μετακινείται κατά μήκος της πλευράς  $AB$ , με κατεύθυνση προς το σημείο τομής της πλευράς  $AB$  με τη μεσοκάθετο του ευθυγράμμου τμήματος  $\overline{u_{m-(i-1)}^1 u_{m-i}^1}$ , στο οποίο λαμβάνει τη μεγαλύτερη τιμή της. Από το σημείο αυτό και δεξιότερα, μειώνεται συνεχώς, έως ότου λάβει τη μικρότερη τιμή της, όταν ταυτιστεί με την κορυφή  $u_1^2$ . Στο σημείο αυτό, η γωνία  $\phi'_i$  είναι ίση με τη γωνία  $\phi$ , η οποία από το Λήμμα 15.2.3 είναι μεγαλύτερη από τη γωνία  $\phi'_{m-1}$ . Επομένως, η γωνία  $\phi'_{m-1}$  είναι η μικρότερη από όλες τις γωνίες που σχηματίζονται από ζεύγη διαδοχικών ακμών, προσκείμενα σε κάθε κορυφή του συνόλου  $V_2$ . Δεδομένου ότι  $m \geq n$ , το ίδιο ισχύει και για τις κορυφές του συνόλου  $V_1$ . Επομένως, η γωνία  $\phi'_{m-1}$  ορίζει τη γωνιακή ευκρίνεια της απεικόνισης.

Θα αποδείξουμε τώρα ότι η γωνιακή ευκρίνεια τομών της παραγόμενης απεικόνισης είναι πάντα μεγαλύτερη από τη γωνιακή της ευκρίνεια. Για να το δείξουμε αυτό, θεωρούμε δύο τεμνόμενες ακμές (βλ. τις έντονες διακεκομμένες τεμνόμενες ακμές της Εικόνας 15.1ii). Η τομή τους ορίζει (i) ένα ζεύγος γωνιών που είναι μικρότερες από  $90^\circ$  και (ii) ένα ακόμα ζεύγος γωνιών, οι οποίες είναι μεγαλύτερες από  $90^\circ$ . Προφανώς, μόνο οι οξείες γωνίες θα συμμετάσχουν στον υπολογισμό της γωνιακής ευκρίνειας τομών (βλ. τη γωνία  $\phi_c$  της Εικόνας 15.1ii). Ωστόσο, σε ένα πλήρες διμερές γράφημα, οι οξείες γωνίες που σχηματίζονται από δύο τεμνόμενες ακμές είναι εξωτερικές σε ένα τρίγωνο, το οποίο έχει δύο από τις κορυφές του στα σύνολα  $V_1$  και  $V_2$ , αντίστοιχα (βλ. το γκρι-σκιασμένο τρίγωνο της Εικόνας 15.1ii). Επομένως, η γωνιακή ευκρίνεια τομών είναι πάντα μεγαλύτερη από τη γωνιακή ευκρίνεια, όπως επιθυμούσαμε.  $\square$

**Λήμμα 15.2.5.** Ισχύει ότι  $\phi'_{m-1} \geq \frac{\phi}{2}$ .

*Απόδειξη.* Ισοδύναμα, αποδεικνύουμε ότι  $\tan \phi'_{m-1} > \tan \frac{\phi}{2}$ . Έστω ότι  $x$  είναι το μήκος της ακμής  $(u_1^2, u_2^1)$  στην παραγόμενη απεικόνιση. Τότε, από την Εξίσωση 15.2 και καθώς  $x > H$ , έχουμε τα παρακάτω:

$$\tan \frac{\phi}{2} = \frac{\frac{a_1}{x}}{1 + \frac{H}{x}} < \frac{a_1}{2H}$$

Από τις παραπάνω σχέσεις έπεται ότι για να ολοκληρωθεί η απόδειξη του λήμματος, αρκεί να δείξουμε ότι  $\tan \phi'_{m-1} > \frac{a_1}{2H}$ . Πιο συγκεκριμένα, έχουμε ότι:

$$\begin{aligned} \tan \phi'_{m-1} > \frac{a_1}{2H} &\Leftrightarrow \frac{\frac{a_1}{H}}{1 + \frac{a_1 + \dots + a_{m-1}}{H} \cdot \frac{a_2 + \dots + a_{m-1}}{H}} > \frac{a_1}{2H} \\ &\Leftrightarrow \frac{1}{1 + \frac{H-a_1}{H}} > \frac{1}{2} \\ &\Leftrightarrow a_1 > 0 \end{aligned}$$

το οποίο προφανώς ισχύει. □

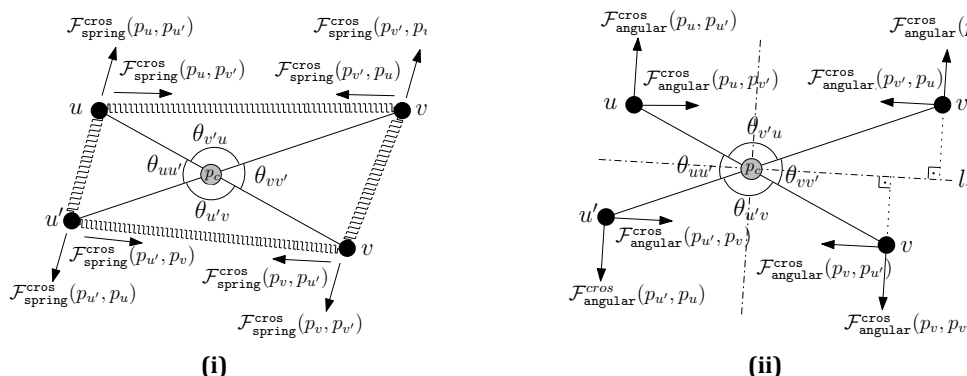
**Θεώρημα 15.2.2.** Ένα πλήρες διμερές γράφημα  $K_{m,n}$  επιδέχεται μία απεικόνιση 2-επιπέδων συνολικής γωνιακής ευκρίνειας  $O(\frac{1}{\max\{m,n\}})$ , η οποία είναι ασυμπτωτικά βέλτιστη.

*Απόδειξη.* Έπεται άμεσα από τα Λήμματα 15.2.4 και 15.2.5. □

Υποθέτουμε, χωρίς βλάβη της γενικότητας, ότι οι ευθείες  $\mathcal{L}_1$  και  $\mathcal{L}_2$  είναι δύο οριζόντιες ευθείες, η ευθεία  $\mathcal{L}_2$  συμπίπτει με τον άξονα των  $x$  και η απεικόνιση που παράγεται από τον αλγόριθμό μας έχει  $a_1 = 1$ . Τότε, μπορούμε να εκφράσουμε το ύψος της απεικόνισης  $\Gamma(K_{m,n})$  ως συνάρτηση της γωνίας  $\phi$  ως εξής:

$$a_1 = 1 \iff \tan \phi \cdot H = 1 \iff H = 1/\tan \phi$$

Με βάση τα παραπάνω, το εμβαδόν της απεικόνισης  $\Gamma(K_{m,n})$ , το οποίο ισούται με  $H^2$ , φράσσεται από την τιμή  $1/\tan^2 \phi$ . Από την Εξίσωση 15.3, το εμβαδόν φράσσεται περαιτέρω από την τιμή  $1/\phi^2$ . Από το Θεώρημα 15.2.2, ισχύει ότι  $\phi = O(1/\max\{m,n\})$ . Επομένως, το συνολικό εμβαδόν που καταλαμβάνει η απεικόνιση είναι της τάξης του  $O(\max\{m^2, n^2\})$ . Ωστόσο, οι κορυφές του γραφήματος  $K_{m,n}$  στην απεικόνιση  $\Gamma(K_{m,n})$  δεν συμπίπτουν απαραίτητα με σημεία του πλέγματος. Για να το πετύχουμε αυτό, μετακινούμε την οριζόντια ευθεία  $\mathcal{L}_1$  στην αμέσως παραπάνω οριζόντια ευθεία πλέγματος και κάθε κορυφή των συνόλων  $V_1$  και  $V_2$  στο δεξιότερο σημείο του πλέγματος στα αριστερά της. Με αυτό τον τρόπο, παράγεται μία νέα απεικόνιση  $\Gamma'(K_{m,n})$ , στην οποία κάθε κορυφή βρίσκεται σε ένα σημείο



**Σχήμα 15.2:** Οι δυνάμεις που ασκούνται στις κορυφές για τη μεγιστοποίηση της γωνιακής ευκρίνειας τομών. (i) Ελατήρια σε κορυφές που εμπλέκονται σε μία διασταύρωση. (ii) Απωθητικές ή ελκτικές δυνάμεις με βάση τις γωνίες.

του πλέγματος. Από το Λήμμα 15.2.1 έπεται ότι δεν υπάρχουν δύο κορυφές, οι οποίες μοιράζονται το ίδιο σημείο πλέγματος, καθώς το τμήμα  $a_1$  είναι ίσο με μία μονάδα πλέγματος. Δεδομένου ότι ούτε η οριζόντια ευθεία  $\mathcal{L}_1$  ούτε οι κορυφές του  $K_{m,n}$  μετακινούνται πάνω από μία μονάδα μήκους, η συνολική γωνιακή ευκρίνεια της απεικόνισης  $\Gamma'(K_{m,n})$  ασυμπτωτικά δεν επηρεάζεται και επιπλέον, το ύψος της απεικόνισης δεν είναι σημαντικά μεγαλύτερο (δηλαδή ασυμπτωτικά παραμένει το ίδιο). Το ακόλουθο θεώρημα συνοψίζει αυτό το αποτέλεσμα.

**Θεώρημα 15.2.3.** Ένα πλήρες διμερές γράφημα  $K_{m,n}$  επιδέχεται μία απεικόνιση 2-επιπέδων σε πλέγμα συνολικής γωνιακής ευκρίνειας  $\Theta(\frac{1}{\max\{m,n\}})$  και συνολικού εμβαδού  $O(\max\{m^2, n^2\})$ .

### 15.3 Ένας Force-Directed Αλγόριθμος

Στην ενότητα αυτή, παρουσιάζουμε ένα force-directed αλγόριθμο, ο οποίος ξεκινώντας από μία αρχική απεικόνιση, η οποία έχει παραχθεί από μία κλασική force-directed τεχνική, καταλήγει σε μία απεικόνιση βελτιωμένης συνολικής γωνιακής ευκρίνειας. Ο αλγόριθμος ενισχύει τον κλασικό force-directed αλγόριθμο του Eades [Ead84] με επιπλέον δυνάμεις που ασκούνται στις κορυφές του γραφήματος. Πιο συγκεκριμένα, αυτές οι επιπλέον δυνάμεις περιλαμβάνουν ελατήρια και πρόσθετες ελκτικές ή απωθητικές δυνάμεις σε κορυφές με βαθμό μεγαλύτερο του ένα, καθώς και στα άκρα των ακμών που εμπλέκονται σε μία τομή. Αυτό διασφαλίζει ότι οι γωνίες, οι οποίες σχηματίζονται μεταξύ ακμών προσκείμενων στην ίδια κορυφή, καθώς και αυτές που σχηματίζονται από ζεύγη τεμνόμενων ακμών θα είναι όσο το δυνατόν μεγαλύτερες σε μία κατάσταση ισορροπίας του μοντέλου.

Όπως ήδη αναφέρθηκε στο Κεφάλαιο 11, ο κλασικός force-directed αλγόριθμος του Eades

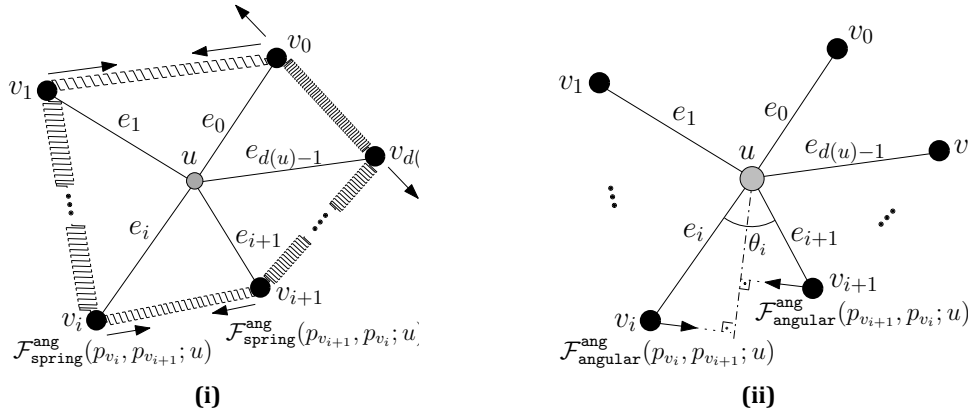
[Ead84] μοντελοποιεί τις κορυφές του γραφήματος σαν ηλεκτρικά φορτισμένα σωματίδια που απωθούνται μεταξύ τους και τις ακμές ως ελατήρια που έλκουν γειτονικές κορυφές. Βασιζόμενοι σε πειραματικά αποτελέσματα, εφαρμόζουμε τις ελκτικές δυνάμεις του κλασικού force-directed αλγορίθμου του Eades και παραλείπουμε τις απωθητικές. Επιπρόσθετα, έχουμε επιλέξει να χρησιμοποιήσουμε ελατήρια, τα οποία ακολουθούν το λογαριθμικό νόμο (αντί του αντίστοιχου νόμου του Hooke), ώστε να αποφύγουμε να ασκήσουμε ισχυρές δυνάμεις σε απομακρυσμένες κορυφές. Πιο συγκεκριμένα, οι ελκτικές δυνάμεις (συμβολίζονται με  $\mathcal{F}_{\text{spring}}$ ) υπολογίζονται με βάση τον τύπο:

$$\mathcal{F}_{\text{spring}}(p_u, p_v) = C_{\text{spring}} \cdot \log \frac{\|p_u - p_v\|}{\ell_{\text{spring}}} \cdot \overrightarrow{p_u p_v}, \quad (u, v) \in E$$

όπου οι σταθερές  $C_{\text{spring}}$  και  $\ell_{\text{spring}}$  αντιστοιχούν στη *σκληρότητα (stiffness)* και το *φυσικό μήκος (natural length)* των ελατηρίων, αντίστοιχα. Υπενθυμίζεται ότι με  $\overrightarrow{p_u p_v}$  συμβολίζεται το διάνυσμα μοναδιαίου μήκους από το σημείο  $p_u$  προς το σημείο  $p_v$ .

Αρχικά, περιγράφουμε την προσέγγισή μας για την περίπτωση που δύο ακμές, έστω  $e = (u, v)$  και  $e' = (u', v')$ , εμπλέκονται σε μία τομή. Έστω  $p_c$  το σημείο τομής τους (βλ. το γκρι-σκιασμένο σημείο της Εικόνας 15.2). Χωρίς βλάβη της γενικότητας, υποθέτουμε ότι η κορυφή  $u$  είναι στα αριστερά της κορυφής  $v$ , η κορυφή  $u'$  στα αριστερά της κορυφής  $v'$ ,  $y_u < y_v$  και  $y_{v'} < y_{u'}$ , όπως στην Εικόνα 15.2. Έστω  $\theta_{vv'}$  η γωνία που σχηματίζεται από τα ευθύγραμμα τμήματα  $\overline{p_c p_v}$  και  $\overline{p_c p_{v'}}$  κατά την αντίθετη φορά των δεικτών του ρολογιού γύρω από την κορυφή  $u$ , από το ευθύγραμμο τμήμα  $\overline{p_c p_v}$  προς το ευθύγραμμο τμήμα  $\overline{p_c p_{v'}}$ . Περαιτέρω, υποθέτουμε ότι  $\theta_{vv'} = \theta_{v'u'}$ , δηλαδή παραβιάζουμε τον υπολογισμό κατά την αντίθετη φορά των δεικτών του ρολογιού των γωνιών, ο οποίος θα οδηγούσε σε  $\theta_{vv'} = 2\pi - \theta_{v'u'}$ . Όμοια, ορίζουμε τις υπόλοιπες γωνίες της Εικόνας 15.2. Προφανώς,  $\theta_{vv'} + \theta_{v'u} = \pi$ . Ιδανικά, θα θέλαμε να ισχύει  $\theta_{vv'} = \theta_{v'u} = \frac{\pi}{2}$ , δηλαδή οι ακμές  $e$  και  $e'$  να σχηματίζουν μία τομή υπό ορθή γωνία. Όπως θα δούμε παρακάτω, το μέτρο των δυνάμεων που ασκούνται στις κορυφές  $u, u', v$  και  $v'$  εξαρτάται από (α) τις γωνίες  $\theta_{vv'}$  και  $\theta_{v'u}$  και (β) τα μήκη των ευθυγράμμων τμημάτων  $\overline{p_c p_u}, \overline{p_c p_{u'}}, \overline{p_c p_v}$  και  $\overline{p_c p_{v'}}$ .

Το φυσικό μοντέλο που περιγράφει την προσέγγισή μας παρουσιάζεται στην Εικόνα 15.2. Αρχικά, για κάθε ζεύγος τεμνόμενων ακμών στο σημείο  $p_c$  τοποθετούμε ελατήρια τα οποία συνδέουν διαδοχικές κορυφές κατά την αντίθετη φορά των δεικτών του ρολογιού γύρω από το σημείο  $p_c$ , όπως στην Εικόνα 15.2i. Το μέτρο των δυνάμεων λόγω αυτών των ελατηρίων θα πρέπει να ικανοποιεί την προτίμησή μας για ορθές γωνίες. Θεωρούμε τώρα τα ελατήρια που συνδέουν τις κορυφές  $v$  και  $v'$ . Οι εναπομείναντες κορυφές αντιμετωπίζονται συμμετρικά. Θέτουμε το φυσικό μήκος, έστω  $\ell_{\text{spring}}^{vv'}$ , του ελατηρίου που συνδέει τις κορυφές  $v$  και  $v'$  ίσο με  $\sqrt{\|p_c - p_v\|^2 + \|p_c - p_{v'}\|^2}$ . Η τιμή αυτή αντιστοιχεί στο μήκος του ευθύγραμμου τμήματος το οποίο συνδέει τις κορυφές  $v$  και  $v'$  στη βέλτιστη περίπτωση όπου  $\theta_{vv'} = \frac{\pi}{2}$ .



**Σχήμα 15.3:** Οι δυνάμεις που ασκούνται στις κορυφές για τη μεγιστοποίηση της γωνιακής ευκρίνειας. (i) Ελατήρια σε διαδοχικές ακμές γύρω από την κορυφή  $u$ . (ii) Απωθητικές ή ελκτικές δυνάμεις με βάση τις γωνίες.

Επομένως, σε μία κατάσταση ισορροπίας του μοντέλου σε ένα γράφημα που αποτελείται μόνο από τις ακμές  $e$  και  $e'$ , οι ακμές  $e$  και  $e'$  θα σχηματίζουν μία τομή υπό ορθή γωνία. Κλείνοντας, η δύναμη που ασκείται στην κορυφή  $v$  λόγω του ελατηρίου της κορυφής  $v'$  ορίζεται από τον παρακάτω τύπο:

$$\mathcal{F}_{\text{spring}}^{\text{cross}}(p_v, p_{v'}) = C_{\text{spring}}^{\text{cross}} \cdot \log \frac{\|p_v - p_{v'}\|}{l_{\text{spring}}^{vv'}} \cdot \overrightarrow{p_v p_{v'}}$$

Οι εναπομείναντες δυνάμεις της Εικόνας 15.2i ορίζονται όμοια. Σημειώνεται ότι στον παραπάνω τύπο, η σταθερά  $C_{\text{spring}}^{\text{cross}}$  χρησιμοποιείται για να ελέγξει τη σκληρότητα των ελατηρίων.

Έως τώρα, έχουμε εκφράσει την προτίμησή μας για τομές υπό ορθές γωνίες με βάση τα μήκη των ευθυγράμμων τμημάτων  $\overline{p_c p_u}$ ,  $\overline{p_c p_{u'}}$ ,  $\overline{p_c p_v}$  και  $\overline{p_c p_{v'}}$ . Το ίδιο μπορεί να επιτευχθεί χρησιμοποιώντας τις γωνίες  $\theta_{vv'}$  και  $\theta_{v'u}$  (βλ. Εικόνα 15.2ii). Θα περιορίσουμε και πάλι, την περιγραφή μας στη γωνία που σχηματίζεται από τα ευθύγραμμα τμήματα  $\overline{p_c p_v}$  και  $\overline{p_c p_{v'}}$ . Ιδανικά, θα θέλαμε να ασκούνται δυνάμεις στις κορυφές  $v$  και  $v'$  τέτοιες ώστε: (i) όταν  $\theta_{vv'} \rightarrow 0$ , το μέτρο της δύναμης να είναι πολύ μεγάλο (για να απωθούνται οι κορυφές  $v$  και  $v'$ ) και (ii) όταν  $\theta_{vv'} \rightarrow \frac{\pi}{2}$ , το μέτρο της δύναμης να είναι πολύ μικρό. Μία συνάρτηση, έστω  $f: \mathbb{R} \rightarrow \mathbb{R}$ , που ικανοποιεί αυτή την ιδιότητα είναι:  $f(\theta) = \frac{|\frac{\pi}{2} - \theta|}{\theta}$ . Έχοντας καθορίσει το μέτρο της δύναμης, απομένει να ορίσουμε την κατεύθυνσή της. Πιο συγκεκριμένα, θέτουμε την κατεύθυνση της δύναμης που ασκείται στην κορυφή  $v$  (εξαιτίας της κορυφής  $v'$ ) να είναι κάθετη στην ευθεία που διχοτομεί τη γωνία  $\theta_{vv'}$  (βλ. τη διακεκομμένη διάστικτη ευθεία  $l_{vv'}$  της Εικόνας 15.2ii), ή ισοδύναμα, παράλληλη στο διάνυσμα μοναδιαίου μήκους  $\text{Perp}(\text{Bsc}(\overrightarrow{p_c p_v}, \overrightarrow{p_c p_{v'}}))$ . Υπενθυμίζεται ότι, τα διανύσματα  $\text{Perp}$  και  $\text{Bsc}$  αντιστοιχούν στο κά-



θετο διάνυσμα και το διάνυσμα της διχοτόμου, αντίστοιχα (βλ. Ενότητα 15.1). Είναι εμφανές ότι, εάν  $\theta_{vv'} < \frac{\pi}{2}$ , οι δυνάμεις που ασκούνται στις κορυφές  $v$  και  $v'$  θα πρέπει να είναι απωθητικές (ώστε να μεγαλώνει η μεταξύ τους γωνία), διαφορετικά ελκτικές. Αυτό επιτυγχάνεται με τη χρήση της συνάρτησης προσήμου (*sign function*). Συνοψίζοντας τα παραπάνω, χρησιμοποιούμε τον ακόλουθο τύπο, ο οποίος εκφράζει τη δύναμη που ασκείται στην κορυφή  $v$  εξαιτίας της κορυφής  $v'$ :

$$\mathcal{F}_{\text{angle}}^{\text{cross}}(p_v, p_{v'}) = C_{\text{angle}}^{\text{cross}} \cdot \text{sign}(\theta_{vv'} - \frac{\pi}{2}) \cdot f(\theta_{vv'}) \cdot \text{Perp}(\text{Bsc}(\overrightarrow{p_v p_{v'}}, \overrightarrow{p_c p_{v'}}))$$

όπου η σταθερά  $C_{\text{angle}}^{\text{cross}}$  ελέγχει την ισχύ της δύναμης. Όμοια, ορίζουμε και τις εναπομείναντες δυνάμεις της Εικόνας 15.2ii.

Θεωρούμε τώρα μία κορυφή  $u \in V$  στην οποία πρόσκεινται  $d(u)$  ακμές, έστω  $e_0 = (u, v_0), e_1 = (u, v_1), \dots, e_{d(u)-1} = (u, v_{d(u)-1})$ , όπως στην Εικόνα 15.3i. Υποθέτουμε ότι οι ακμές  $e_0, e_1, \dots, e_{d(u)-1}$  είναι διαδοχικές κατά την αντίθετη φορά των δεικτών του ρολογιού γύρω από την κορυφή  $u$  στην απεικόνιση του γραφήματος. Όμοια με την περίπτωση των τεμνόμενων ακμών, θα ενώσουμε τα άκρα διαδοχικών ακμών γύρω από την κορυφή  $u$  με ελατήρια, όπως στην Εικόνα 15.3i. Σε αυτή την περίπτωση, το φυσικό μήκος κάθε ελατηρίου θα πρέπει να ικανοποιεί την προτίμηση μας για γωνίες ίσες με  $\frac{2\pi}{d(u)}$  (σ.σ. το προφανές άνω φράγμα). Για να το επιτύχουμε αυτό, ακολουθούμε την παρακάτω προσέγγιση: Για κάθε  $i = 0, 1, \dots, d(u) - 1$ , θέτουμε το φυσικό μήκος, έστω  $\ell_{\text{spring}}^i$ , του ελατηρίου που συνδέει τις κορυφές  $v_i$  και  $v_{(i+1) \bmod d(u)}$ , ίσο με:

$$\ell_{\text{spring}}^i = \sqrt{\|a_i\|^2 + \|b_i\|^2 - 2\|a_i\|\|b_i\| \cdot \cos(2\pi/d(u))}$$

όπου  $a_i = e_i$  και  $b_i = e_{(i+1) \bmod d(u)}$ . Η τιμή  $\ell_{\text{spring}}^i$  αντιστοιχεί στο μήκος του ευθυγράμμου τμήματος που συνδέει τις κορυφές  $v_i$  και  $v_{(i+1) \bmod d(u)}$  στη βέλτιστη περίπτωση όπου η γωνία που σχηματίζεται από τις ακμές  $e_i$  και  $e_{(i+1) \bmod d(u)}$  ισούται με  $\frac{2\pi}{d(u)}$ , για κάθε  $i = 0, 1, \dots, d(u) - 1$ . Επομένως, οι δυνάμεις των ελατηρίων μεταξύ διαδοχικών ακμών ακολουθούν τον παρακάτω τύπο:

$$\mathcal{F}_{\text{spring}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod d(u)}}; u) = C_{\text{spring}}^{\text{angular}} \cdot \log \frac{\|p_{v_i} - p_{v_{(i+1) \bmod d(u)}}\|}{\ell_{\text{spring}}^i} \cdot \overrightarrow{p_{v_i} p_{v_{(i+1) \bmod d(u)}}}$$

όπου η ποσότητα  $C_{\text{spring}}^{\text{angular}}$  είναι μία σταθερά που ελέγχει τη σκληρότητα του ελατηρίου.

Έστω τώρα ότι  $\theta_i$  είναι η γωνία που σχηματίζεται από τις ακμές  $e_i$  και  $e_{(i+1) \bmod d(u)}$ , υπολογισμένη κατά την αντίθετη φορά των δεικτών του ρολογιού από την ακμή  $e_i$  προς την ακμή  $e_{(i+1) \bmod d(u)}$ ,  $i = 0, 1, \dots, d(u) - 1$ . Όμοια με την περίπτωση των δύο τεμνόμενων ακμών, εφαρμόζουμε δυνάμεις στις κορυφές  $v_i$  και  $v_{(i+1) \bmod d(u)}$  κάθετες στη διχοτόμο της

γωνίας  $\theta_i$ , όπως φαίνεται στην Εικόνα 15.3ii. Ωστόσο, σε αυτή την περίπτωση, χρειαζόμαστε μία συνάρτηση για το μέτρο, τέτοια ώστε: (i) όταν  $\theta_i \rightarrow 0$ , το μέτρο της δύναμης να είναι πολύ μεγάλο (ώστε να απωθούνται οι κορυφές  $v_i$  και  $v_{(i+1) \bmod(d(u))}$ ) και (ii) όταν  $\theta_i \rightarrow \frac{2\pi}{d(u)}$ , το μέτρο της δύναμης να είναι πολύ μικρό. Μία τέτοια συνάρτηση, έστω  $g : \mathbb{R} \times V \rightarrow \mathbb{R}$ , είναι η εξής:  $g(\theta; u) = \frac{|\frac{2\pi}{d(u)} - \theta|}{\theta}$ . Έχοντας καθορίσει πλήρως το μέτρο των δυνάμεων που ασκούνται στα άκρα διαδοχικών ακμών και την κατεύθυνσή τους, θα παρουσιάσουμε τους ακριβείς τύπους τους οποίους ακολουθούν οι δυνάμεις:

$$\mathcal{F}_{\text{angle}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod(d(u))}}; u) = C_{\text{angle}}^{\text{angular}} \cdot \text{sign}(\theta_i - \frac{2\pi}{d(u)}) \cdot g(\theta_i; u) \cdot \text{Perp}(\text{Bsc}(\overrightarrow{p_u p_{v_i}}, \overrightarrow{p_u p_{v_{(i+1) \bmod(d(u))}}}))$$

όπου η σταθερά  $C_{\text{angle}}^{\text{angular}}$  ελέγχει την ισχύ της δύναμης. Σημειώνεται ότι η πειραματική αξιολόγηση έδειξε ότι οι δυνάμεις  $\mathcal{F}_{\text{cross}}^x$  και  $\mathcal{F}_{\text{angle}}^x$ , όπου  $x \in \{\text{cross}, \text{angular}\}$  έχουν συμπληρωματική ισχύ και παράγουν καλύτερες απεικονίσεις όταν εφαρμόζονται και οι δύο μαζί. Υποθέτουμε ότι αυτό οφείλεται στο γεγονός ότι η κίνηση των κορυφών καθορίζεται από τη συνισταμένη μεγαλύτερου αριθμού διανυσμάτων. Επιπλέον, θέτοντας μηδενικές τιμές είτε στις σταθερές  $C_{\text{spring}}^{\text{cross}}$  και  $C_{\text{angle}}^{\text{cross}}$  ή στις σταθερές  $C_{\text{spring}}^{\text{angular}}$  και  $C_{\text{angle}}^{\text{angular}}$ , ο αλγόριθμός μπορεί να διαμορφωθεί έτσι ώστε να μεγιστοποιεί μόνο τη γωνιακή ευκρίνεια ή μόνο τη γωνιακή ευκρίνεια τομών, αντίστοιχα.

### 15.3.1 Ανάλυση Χρονικής Πολυπλοκότητας

Η προσέγγισή μας συνοψίζεται στον Αλγόριθμο 15.1. Σε κάθε επανάληψη, ο αλγόριθμος υπολογίζει τρεις τύπους δυνάμεων. Οι δυνάμεις Τύπου-1 αντιστοιχούν στις ελκτικές δυνάμεις του κλασσικού force-directed μοντέλου μεταξύ ζευγών γειτονικών κορυφών του γραφήματος και ο υπολογισμός τους απαιτεί  $O(E)$  χρόνο ανά επανάληψη. Οι δυνάμεις Τύπου-2 του Αλγορίθμου 15.1 αντιστοιχούν στις δυνάμεις εξαιτίας των τομών μεταξύ των ακμών. Ο υπολογισμός των ακμών που εμπλέκονται σε τομές στη Γραμμή 9 του Αλγορίθμου 15.1 απαιτεί  $O(E^2)$  χρόνο. Έχοντας υπολογίσει όλα τα ζεύγη τεμνόμενων ακμών, μπορούμε να υπολογίσουμε τις συσχετιζόμενες δυνάμεις σε σταθερό χρόνο για κάθε ζεύγος τεμνόμενων ακμών. Επομένως, οι δυνάμεις Τύπου-2 μπορούν να υπολογιστούν σε  $O(E^2)$  χρόνο ανά επανάληψη. Οι δυνάμεις Τύπου-3 του Αλγορίθμου 15.1 αντιστοιχούν στις δυνάμεις εξαιτίας των γωνιών μεταξύ διαδοχικών ακμών. Στην περίπτωση αυτή, αρχικά ταξινομούμε τις προσκείμενες ακμές κάθε κορυφής του γραφήματος σε κυκλική διάταξη (βλ. Γραμμή 21 του Αλγορίθμου 15.1). Αυτό απαιτεί  $O(d(G) \log d(G))$  χρόνο, όπου με  $d(G)$  συμβολίζεται ο βαθμός του γραφήματος. Οι Γραμμές 21-23, όπου υπολογίζουμε τις δυνάμεις, απαιτούν  $O(d(G))$  επιπλέον χρόνο. Επομένως, οι δυνάμεις Τύπου-3 απαιτούν  $O(E + Vd(G) \log d(G))$  χρόνο ανά επανάληψη. Συνοψίζοντας τα παραπάνω, κάθε επανάληψη του Αλγορίθμου 15.1 απαιτεί

$O(E^2 + V(V + d(G) \log d(G)))$  χρόνο.

Η υπολογιστική πολυπλοκότητα του Αλγορίθμου 15.1 μπορεί να βελτιωθεί περαιτέρω με κλασσικές τεχνικές της υπολογιστικής γεωμετρίας [dBvKOS00, O'R98, PS85]. Πιο συγκεκριμένα, εάν  $K$  είναι ο αριθμός των τεμνόμενων-ανά-ζεύγη ακμών, τότε οι  $K$  τομές για τις δυνάμεις Τύπου-2 μπορούν να υπολογιστούν σε  $O(K + E \log^2 E / \log \log E)$  χρόνο [PS85, pp.277], το οποίο οδηγεί σε πολυπλοκότητα συνολικού χρόνου  $O(K + E \log^2 E / \log \log E + V(V + d(G) \log d(G)))$  ανά επανάληψη.

### 15.3.2 Ομοιότητες και Διαφορές με Προηγούμενες Τεχνικές

Στην εργασία των Lin και Yen [LY05], η τεχνική που εφαρμόζει μεγάλες απωθητικές δυνάμεις κάθετες στις διχοτόμους των γωνιών που σχηματίζονται από διαδοχικές ακμές σε μία κορυφή, όταν αυτές είναι μικρές, αναφέρεται ως *απόθεση ακμής-ακμής (edge-edge repulsion)*. Στην εργασία τους, η συνάρτηση, έστω  $g : \mathbb{R} \rightarrow \mathbb{R}$ , η οποία ελέγχει το μέτρο της δύναμης ισούται με:  $g(\theta) = \cot(\frac{\theta}{2})$ , όπου  $\theta$  είναι η γωνία που σχηματίζεται μεταξύ δύο διαδοχικών ακμών προσκείμενων στην ίδια κορυφή. Παρατηρούμε ότι η συνάρτηση  $g$  έχει την ίδια ιδιότητα με την αντίστοιχη που χρησιμοποιήσαμε στην προσέγγισή μας, δηλαδή η τιμή της  $g(\theta)$  γίνεται πολύ μεγάλη, όταν  $\theta \rightarrow 0$ . Προτιμήσαμε ωστόσο, να χρησιμοποιήσουμε μία διαφορετική συνάρτηση για να ελέγξουμε το μέτρο της δύναμης, ώστε να ακολουθήσουμε μία ομοιόμορφη προσέγγιση για την περίπτωση της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών.

Οι Huang et al. [HEHL10], οι οποίοι ανεξάρτητα δημοσίευσαν ένα παρόμοιο force-directed αλγόριθμο, χρησιμοποιούν μόνο "γωνιακές" δυνάμεις (σ.σ. οι οποίες εξαρτώνται μόνο από τις γωνίες που σχηματίζονται είτε από τεμνόμενες ακμές είτε από διαδοχικές ακμές προσκείμενες στην ίδια κορυφή) με διαφορετικές συναρτήσεις για τον έλεγχο του μέτρου και της κατεύθυνσης των δυνάμεων τους. Δε χρησιμοποιούν, επίσης, δυνάμεις ελατηρίων για να επηρεάζουν τη γωνιακή ευκρίνεια ή τη γωνιακή ευκρίνεια τομών (εκτός από αυτές του κλασσικού force-directed αλγόριθμου του Eades [Ead84]). Στην Ενότητα 15.3.3, παρουσιάζουμε μία πειραματική σύγκριση των παραπάνω τεχνικών.

Στις εργασίες [DLR10] και [DLR11], η γωνιακή ευκρίνεια τομών βελτιώνεται σε ένα μεταγενέστερο βήμα, στο οποίο κάθε ζεύγος τεμνόμενων ακμών συσχετίζεται με ένα δίσκο με κέντρο το σημείο τομής. Τα σημεία τομής του δίσκου με τις ακμές που εμπλέκονται στην τομή ορίζουν τέσσερις νέες κορυφές, οι οποίες σχηματίζουν έναν 4-κύκλο που ονομάζεται *θάλαμος (cage)*. Στη συνέχεια, εφαρμόζονται κατάλληλες δυνάμεις, τέτοιες ώστε οι θάλαμοι να απεικονίζονται όσο το δυνατόν περισσότερο ως τετράγωνα, το οποίο αναγκάζει τις διαγώνιες να τέμνονται υπό μεγάλες γωνίες. Δεδομένου ότι αυτές οι τεχνικές καταλήγουν σε απεικονίσεις με σημεία καμπής, δεν συμπεριελήφθησαν στην πειραματική σύγκριση.

**Αλγόριθμος 15.1:** FORCE-DIRECTED ΑΛΓΟΡΙΘΜΟΣ

**Input** : Ένα μη-κατευθυνόμενο γράφημα  $G$  και μία αρχική τοποθέτηση  $P = (p_v)_{v \in V}$ .

**Output** : Μία απεικόνιση του  $G$  με υψηλή γωνιακή ευκρίνεια και γωνιακή ευκρίνεια τομών.

**for** ( $t \leftarrow 1$  **to** ITERATIONS) **do**

{Τύπου-1: Δυνάμεις Ελατηρίων.}

**foreach**  $u \in V$  **do**

$$\mathcal{F}_u(t) \leftarrow \sum_{v:(u,v) \in E} \mathcal{F}_{\text{spring}}(p_v, p_u);$$

{Τύπου-2: Δυνάμεις που ασκούνται στις κορυφές για τη μεγιστοποίηση της γωνιακής ευκρίνειας τομών.}

**foreach** (ζεύγος τεμνόμενων ακμών  $e = (u, v)$  και  $e' = (u', v')$ ) **do**

//Οι σχετικές θέσεις των ακμών  $e$  και  $e'$  παρουσιάζονται στην Εικόνα 15.2.;

$$\mathcal{F}_v(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_v, p_{v'}) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_v, p_{u'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_v, p_{v'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_v, p_{u'});$$

$$\mathcal{F}_{v'}(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{v'}, p_u) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{v'}, p_v) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{v'}, p_u) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{v'}, p_v);$$

$$\mathcal{F}_u(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_u, p_{v'}) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_u, p_{u'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_u, p_{v'}) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_u, p_{u'});$$

$$\mathcal{F}_{u'}(t) += \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{u'}, p_v) + \mathcal{F}_{\text{spring}}^{\text{cros}}(p_{u'}, p_u) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{u'}, p_v) + \mathcal{F}_{\text{angle}}^{\text{cros}}(p_{u'}, p_u);$$

{Τύπου-3: Δυνάμεις που ασκούνται στις κορυφές για τη μεγιστοποίηση της γωνιακής ευκρίνειας.}

**foreach** ( $u \in V$  με  $d(u) > 1$ ) **do**

$e_0, \dots, e_{d(u)-1} \leftarrow$  οι προσκείμενες ακμές στην κορυφή  $u$  κατά την αντίθετη φορά των δεικτών του ρολογιού (βλ. Εικόνα 15.3);

Έστω  $e_i = (u, v_i)$ ,  $i = 0, \dots, d(u) - 1$ ;

**for** ( $i \leftarrow 0$  **to**  $d(u) - 1$ ) **do**

$$\mathcal{F}_u(t) += \mathcal{F}_{\text{spring}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod d(u)}}; u) + \mathcal{F}_{\text{angle}}^{\text{angular}}(p_{v_i}, p_{v_{(i+1) \bmod d(u)}}; u);$$

**foreach**  $u \in V$  **do**

$$p_u \leftarrow p_u + \delta \cdot \mathcal{F}_u(t);$$

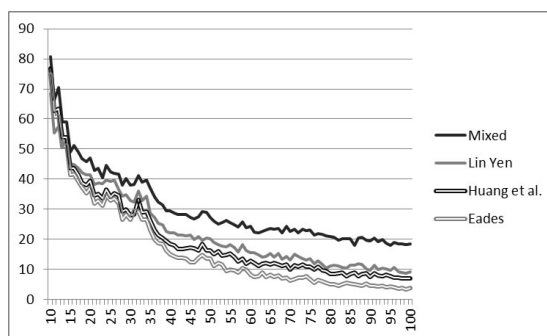
**15.3.3 Πειραματικά Αποτελέσματα**

Στην ενότητα αυτή, παρουσιάζουμε τα αποτελέσματα της πειραματικής αξιολόγησης του αλγορίθμου μας. Εκτός από τον αλγόριθμό μας, υλοποιήσαμε τον κλασικό force-directed αλγόριθμο του Eades [Ead84], τον αλγόριθμο των Lin και Yen [LY05] και τον αλγόριθμο των

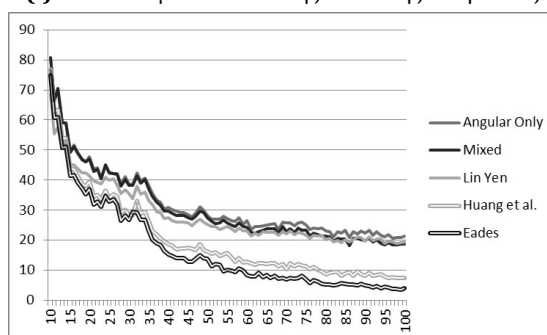
Huang et al. [HEHL10]. Οι υλοποιήσεις έγιναν σε Java χρησιμοποιώντας τη βιβλιοθήκη των yFiles (<http://www.yworks.com>). Το πείραμα πραγματοποιήθηκε σε ένα μηχάνημα Linux με 2.00 GHz CPU και 2GB RAM, χρησιμοποιώντας τα Rome γραφήματα (μία συλλογή περίπου 11.500 γραφημάτων) από την ιστοσελίδα [graphdrawing.org](http://graphdrawing.org). Η Εικόνα 15.8 παρουσιάζει μία απεικόνιση ενός Rome γραφήματος με 99 κορυφές και 135 ακμές που παράχθηκε από το force-directed αλγόριθμό μας. Παρόλο που ο αλγόριθμός μας εστιάζει στη γωνιακή ευκρίνεια, στη γωνιακή ευκρίνεια τομών και τη συνολική γωνιακή ευκρίνεια θα παρουσιάσουμε δεδομένα και από την απόδοση του αλγορίθμου σε σχέση με το συνολικό εμβαδόν, το μήκος των ακμών και τον αριθμό των τομών (βλ. Εικόνα 15.5).

Το πείραμα πραγματοποιήθηκε ως εξής: Αρχικά, για κάθε Rome γράφημα κατασκευάστηκε μία απεικόνιση χρησιμοποιώντας τον SmartOrganic layouter της βιβλιοθήκης yFiles. Η συγκεκριμένη απεικόνιση ήταν η είσοδος όλων των αλγορίθμων με σκοπό να επιταχύνουμε το πείραμα και να ξεπεράσουμε προβλήματα εγκλωβισμού των αλγορίθμων σε τοπικά ελάχιστα, ειδικότερα σε μεγάλα γραφήματα. Εάν η γωνιακή ευκρίνεια και η γωνιακή ευκρίνεια τομών δε βελτιώνονταν περισσότερο από 0.001 μοίρες μεταξύ δύο διαδοχικών επαναλήψεων, θεωρούσαμε ότι ο αλγόριθμος έχει πετύχει σύγκλιση και δεν προχωρούσαμε περαιτέρω. Σημειώνεται ότι η συνθήκη τερματισμού είναι αρκετά αυστηρή και απαιτεί ένα μεγάλο αριθμό επαναλήψεων. Ο μέγιστος αριθμός επαναλήψεων που μπορούσε να πραγματοποιήσει ο αλγόριθμος μέχρι να πετύχει τη σύγκλιση, είχε καθοριστεί στις 100.000. Ωστόσο, πολύ λίγα γραφήματα χρειάστηκε να προσεγγίσουν τον αριθμό αυτό μέχρι να επιτευχθεί η σύγκλιση. Μετά τη σύγκλιση του αλγορίθμου, υπολογίζαμε τη γωνιακή ευκρίνεια, τη γωνιακή ευκρίνεια τομών, το μέσο μήκος ακμών, τον αριθμό των τομών, το συνολικό χρόνο που απαιτήθηκε, καθώς και τον αριθμό των επαναλήψεων μέχρι τη σύγκλιση. Σημειώνεται ότι ο αλγόριθμός μας αξιολογήθηκε ως (α) Angular-Only (μόνο γωνιακή ευκρίνεια), (β) Crossing-Only (μόνο γωνιακή ευκρίνεια τομών) και (γ) Mixed (συνολική γωνιακή ευκρίνεια). Τα αποτελέσματα παρουσιάζονται στις Εικόνες 15.4, 15.5 και 15.6. Οι τιμές που εμφανίζονται στις Εικόνες 15.4, 15.5 και 15.6 είναι οι μέσες τιμές για όλα τα γραφήματα της βάσης δεδομένων για συγκεκριμένο μέγεθος (σ.σ., αριθμό κορυφών).

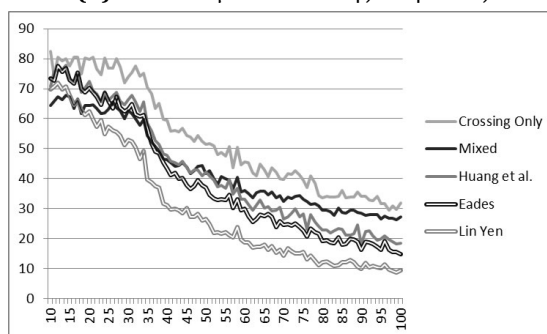
*Το Πρόβλημα της Μεγιστοποίησης της Συνολικής Γωνιακής Ευκρίνειας:* Το διάγραμμα της Εικόνας 15.4i δείχνει ότι ο Mixed αλγόριθμος υπερέχει εμφανώς έναντι των υπόλοιπων αλγορίθμων απεικόνισης του πειράματος. Πιο συγκεκριμένα, στο 80.46% (90.83%, αντίστοιχα) των γραφημάτων του πειράματος, ο Mixed αλγόριθμος οδηγεί σε καλύτερη λύση σε σύγκριση με τον αλγόριθμο των Lin και Yen (Huang et. al, αντίστοιχα) με μία μέση βελτίωση 10.21 (11.63, αντίστοιχα) μοιρών. Το διάγραμμα της Εικόνας 15.4i δείχνει περαιτέρω ότι ο Mixed αλγόριθμος παρήγαγε απεικονίσεις συνολικής ανάλυσης περίπου 20 μοιρών για γραφήματα με περισσότερες από 50 κορυφές της βάσης δεδομένων. Ένα παράδειγμα ενός τέτοιου γρα-



(i) Αποτελέσματα Συνολικής Γωνιακής Ευκρίνειας



(ii) Αποτελέσματα Γωνιακής Ευκρίνειας

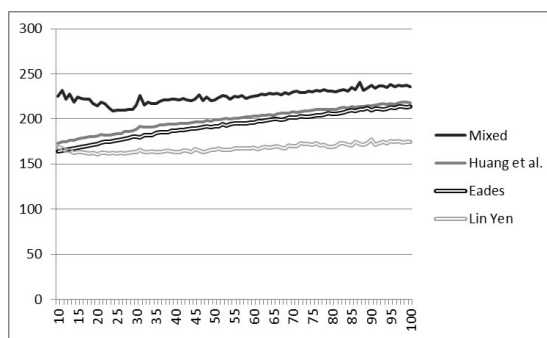


(iii) Αποτελέσματα Γωνιακής Ευκρίνειας Τομών

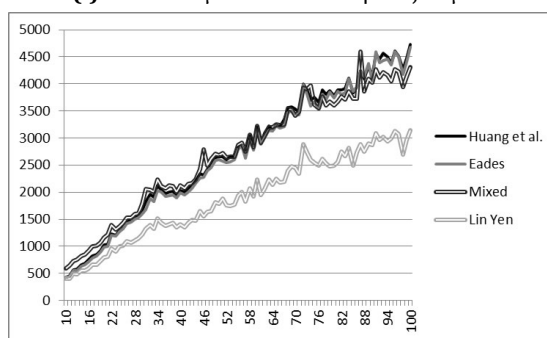
**Σχήμα 15.4:** Παρουσίαση των πειραματικών αποτελεσμάτων στη **Γωνιακή Ευκρίνεια**: Ο  $X$ -άξονας αντιστοιχεί στον αριθμό των κορυφών του γραφήματος και ο  $Y$ -άξονας στη γωνιακή ευκρίνεια υπολογισμένη σε μοίρες.

φήματος δίνεται στην Εικόνα 15.8.

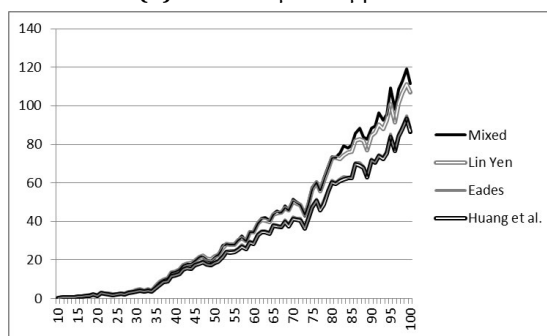
*Το Πρόβλημα της Μεγιστοποίησης της Γωνιακής Ευκρίνειας:* Το διάγραμμα της Εικόνας 15.4ii αναδεικνύει ότι ο Angular-Only αλγόριθμος παράγει απεικονίσεις με καλύτερη γωνιακή ανάλυση σε σχέση με τους υπόλοιπους αλγορίθμους απεικόνισης του πειράματος. Πιο συγκεκριμένα, στο 67.82% (92.12%, αντίστοιχα) των γραφημάτων του πειράματος, ο Angular-Only αλγόριθμος οδηγεί σε καλύτερη λύση σε σύγκριση με τον αλγόριθμο των Lin και Yen (Huang



(i) Αποτελέσματα Μέσου Μήκους Ακμών



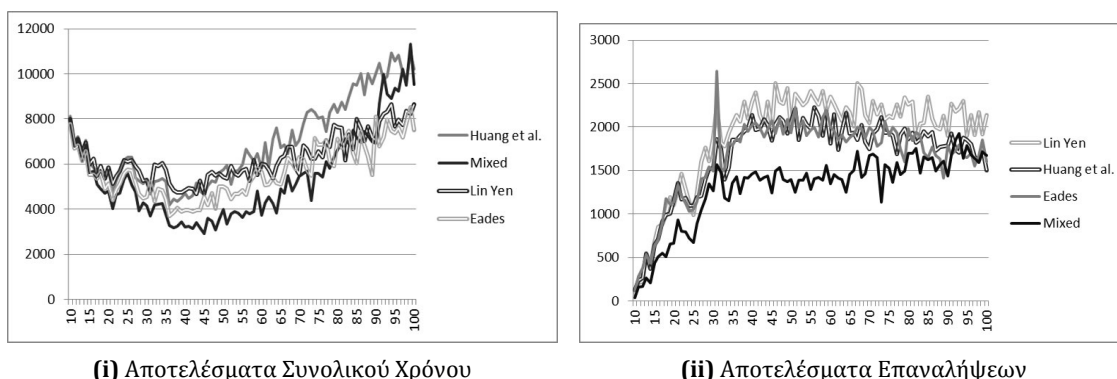
(ii) Αποτελέσματα Εμβαδού



(iii) Αποτελέσματα Αριθμού Τομών

**Σχήμα 15.5:** Παρουσίαση των πειραματικών αποτελεσμάτων σε **συνήθεις μετρικές απεικόνισης γραφημάτων**: Ο  $X$ -άξονας αντιστοιχεί στον αριθμό των κορυφών του γραφήματος. Στις Εικόνες (i) και (ii), ο  $Y$ -άξονας αντιστοιχεί στο εμβαδόν και στο μέσο μήκος ακμών της παραγόμενης απεικόνισης, αντίστοιχα. Στην Εικόνα (iii), ο  $Y$ -άξονας αντιστοιχεί στον αριθμό των τομών της παραγόμενης απεικόνισης.

et. al, αντίστοιχα) με μία μέση βελτίωση των 6.73 (12.75, αντίστοιχα) μοιρών. Η γωνιακή ανάλυση των απεικονίσεων που παράγονται από το Mixed αλγόριθμο είναι περίπου ίση με εκείνη των απεικονίσεων που παράγονται από τον αλγόριθμο των Lin και Yen. Σημειώνεται ότι ο αλγόριθμος των Lin και Yen, σε αντίθεση με τον αλγόριθμό μας, δεν τροποποιεί την εμ-



**Σχήμα 15.6:** Παρουσίαση των πειραματικών αποτελεσμάτων στην **αλγοριθμική απόδοση**: Ο *X*-άξονας αντιστοιχεί στον αριθμό των κορυφών του γραφήματος. Στις Εικόνες (i) και (ii) ο *Y*-άξονας αντιστοιχεί στο συνολικό χρόνο υπολογισμένο σε milliseconds και στις επαναλήψεις που απαιτήθηκαν έως τη σύγκλιση, αντίστοιχα.

φύτευση της αρχικής απεικόνισης (βλ. την εργασία [LY05] για περισσότερες λεπτομέρειες), δηλαδή απαιτεί μία σχεδόν-κοντά-στην-τελική αρχική απεικόνιση και τη βελτιώνει. Αυτό εξηγεί το γεγονός ότι ο Mixed αλγόριθμος πετυχαίνει την ίδια περίπου απόδοση σε όρους γωνιακής ανάλυσης με τον αντίστοιχο των Lin και Yen. Πιο συγκεκριμένα, στο 59.23% των γραφημάτων, ο Mixed αλγόριθμος οδηγεί σε καλύτερη λύση σε σύγκριση με τον αλγόριθμο των Lin και Yen με μία μέση βελτίωση των 6.49°.

*Το Πρόβλημα της Μεγιστοποίησης της Γωνιακής Ευκρίνειας Τομών:* Στην Εικόνα 15.4iii τα δεδομένα φιλτράρονται έτσι ώστε να απεικονίζουν μόνο τα αποτελέσματα που αφορούν σε μη-επίπεδες απεικονίσεις που παράγονται από τους αλγορίθμους, ώστε να αποφευχθούν άπειρες τιμές στην περίπτωση των επίπεδων απεικονίσεων. Ο Crossing-Only αλγόριθμος υπερέχει εμφανώς έναντι των υπόλοιπων αλγορίθμων απεικόνισης του πειράματος. Πιο συγκεκριμένα, στο 92.26% (78.47%, αντίστοιχα) των γραφημάτων του πειράματος, ο Crossing-Only αλγόριθμος οδηγεί σε καλύτερη λύση σε σύγκριση με τον αλγόριθμο των Lin και Yen (Huang et. al, αντίστοιχα). Η μέση βελτίωση που επιτυγχάνεται από τον Crossing-Only αλγόριθμο είναι 27.13 (17.08, αντίστοιχα) μοιρών σε σύγκριση με τον αλγόριθμο των Lin και Yen (Huang et. al, αντίστοιχα). Ο Mixed αλγόριθμος αποφέρει καλύτερη λύση στο 85.23% (54.62%, αντίστοιχα) των γραφημάτων του πειράματος σε σύγκριση με τον αλγόριθμο των Lin και Yen (Huang et. al, αντίστοιχα) με μία μέση βελτίωση των 18.75 (11.17, αντίστοιχα) μοιρών.

Από τα παραπάνω έπεται ότι ο αλγόριθμός μας καταλήγει σε απεικονίσεις με καλύτερη γωνιακή ευκρίνεια σε σύγκριση με απεικονίσεις που παράγονται από τους υπόλοιπους αλγορίθμους του πειράματος. Ωστόσο, η πειραματική αξιολόγηση έδειξε, επίσης, ότι το μέσο μή-



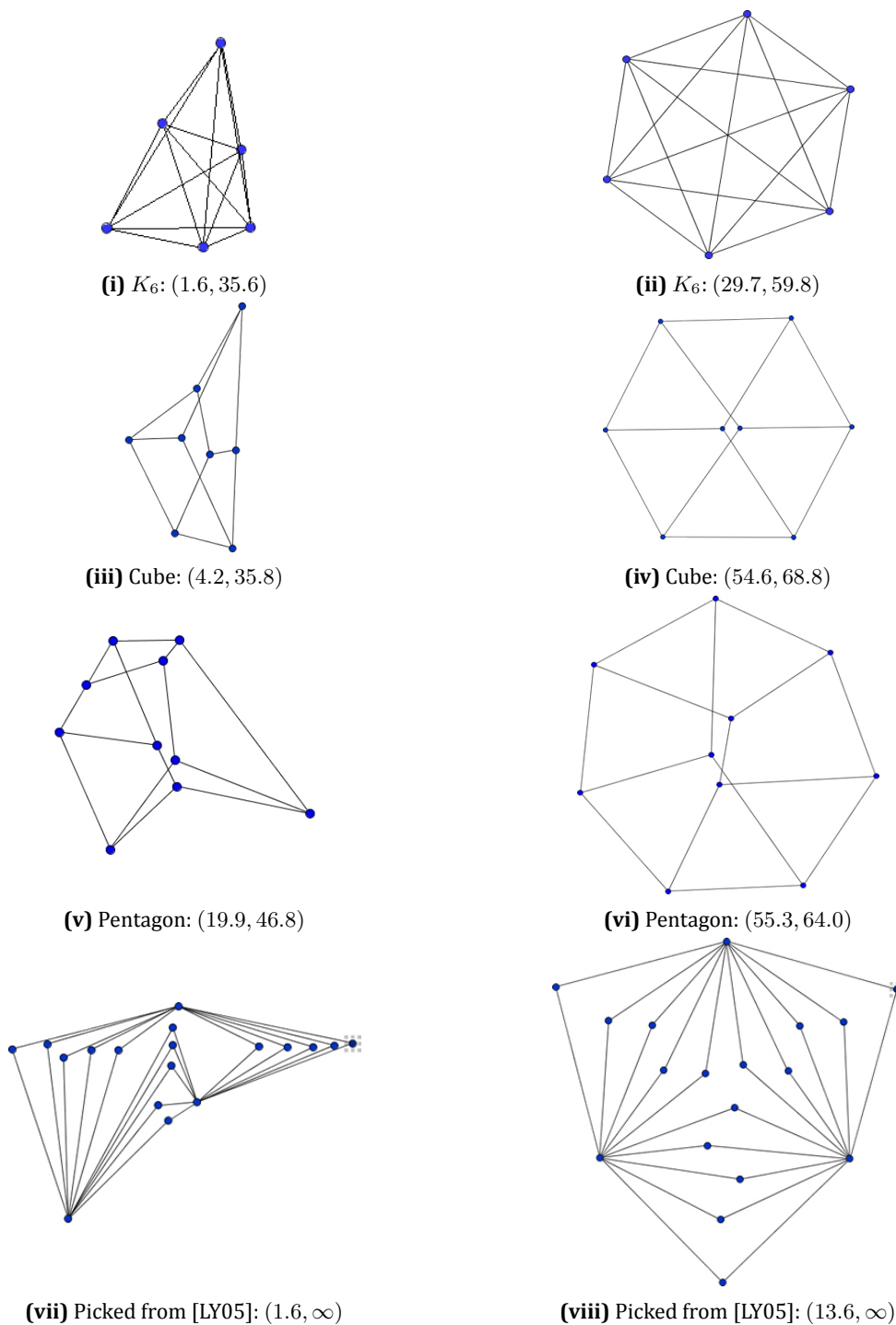
κος ακμών των απεικονίσεων που παράγονται από τον αλγόριθμό μας είναι μεγαλύτερο από το αντίστοιχο των απεικονίσεων που παράγονται από τους υπόλοιπους αλγορίθμους (βλ. Εικόνα 15.5i). Το μήκος των ακμών επηρεάζεται από το φυσικό μήκος των ελατηρίων που χρησιμοποιούνται από τους force-directed αλγορίθμους. Στο πείραμα, το φυσικό μήκος των ελατηρίων είχε καθοριστεί στις 150 μονάδες. Το μέσο μήκος ακμών των απεικονίσεων που παράγονται από το Mixed αλγόριθμο ήταν 225.11 μονάδες, ενώ τα αντίστοιχα των Eades, Lin και Yen και Huang et al. ήταν 191.61, 167.1 και 198.45 μονάδες, αντίστοιχα. Η πειραματική αξιολόγηση έδειξε, επίσης, ότι ο αλγόριθμός μας παράγει απεικονίσεις συγκρίσιμου εμβαδού με αυτό που απαιτούν οι αλγόριθμοι των Eades και Huang et al. Ωστόσο, στον τομέα αυτό υπερέχει ο αλγόριθμος των Lin και Yen, ο οποίος παράγει μικρότερες απεικονίσεις. Παρά το γεγονός ότι ο Mixed αλγόριθμος παράγει απεικονίσεις μεγαλύτερου μέσου μήκους ακμών σε σχέση με τις απεικονίσεις που παράγονται από τους αλγορίθμους των Eades και Huang et al., οι παραγόμενες απεικονίσεις είναι συγκρίσιμου εμβαδού. Αυτό μπορεί να αποδοθεί στο γεγονός ότι, όπως δείχνει η πειραματική ανάλυση, ο Mixed αλγόριθμος παράγει απεικονίσεις με μεγαλύτερο αριθμό τομών. Πιο συγκεκριμένα, η Εικόνα 15.5iii δείχνει ότι ο Mixed αλγόριθμος οδηγεί σε απεικονίσεις με περίπου τον ίδιο αριθμό τομών με αυτές των Lin and Yen και λίγες περισσότερες από αυτές των Eades και Huang et al.

Τέλος, οι Εικόνες 15.6i και 15.6ii συνοψίζουν τη χρονική απόδοση των αλγορίθμων και τις επαναλήψεις που απαιτούνται μέχρι τη σύγκλιση τους, αντίστοιχα. Ο Mixed αλγόριθμος απαιτεί, κατά μέσο όρο, 5191 milliseconds και 1298 επαναλήψεις μέχρι τη σύγκλιση, ενώ ο αλγόριθμος των Huang et al. (ο οποίος είναι της ίδιας χρονικής πολυπλοκότητας) 6626 και 1694, αντίστοιχα. Ο αλγόριθμος των Lin και Yen απαιτεί, κατά μέσο όρο, 6123 milliseconds και 1920 επαναλήψεις μέχρι τη σύγκλιση. Σημειώνεται ότι η χρονική πολυπλοκότητα του αλγορίθμου των Lin και Yen είναι καλύτερη από την αντίστοιχη του αλγορίθμου μας. Ωστόσο, η συνθήκη τερματισμού λαμβάνει υπόψη της και τη βελτίωση της γωνιακής ευκρίνειας τομών και επομένως, ο αλγόριθμος των Lin και Yen απαιτεί περισσότερες επαναλήψεις για να συγκλίνει, το οποίο εξηγεί αυτή την αντίφαση. Από τη σύγκριση, εξαιρέσαμε στον τομέα αυτό τον αλγόριθμο του Eades, καθώς δε λαμβάνει υπόψη του ούτε τη γωνιακή ευκρίνεια ούτε τη γωνιακή ευκρίνεια τομών και επομένως, απαιτεί πολλές επαναλήψεις για να συγκλίνει, όπως φαίνεται στην Εικόνα 15.6ii.

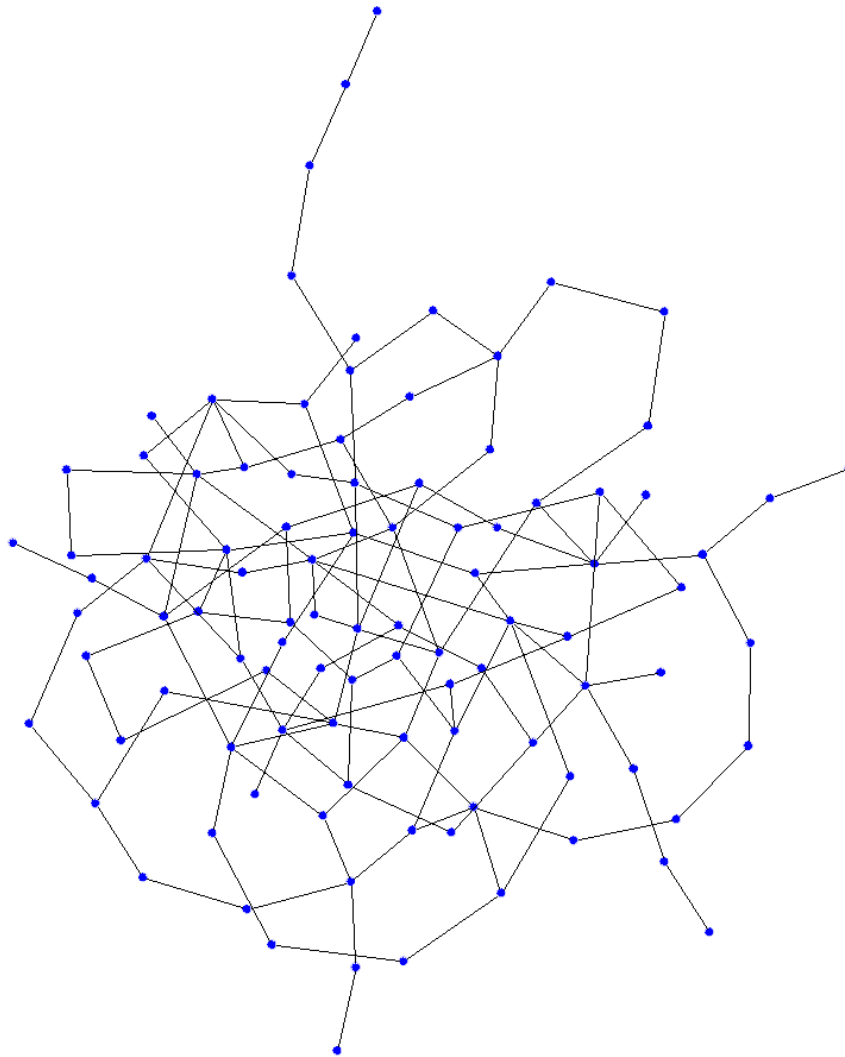
## 15.4 Παραδείγματα Απεικονίσεων

Στην ενότητα αυτή, παρουσιάζουμε μερικές ενδεικτικές απεικονίσεις που παράγονται από το force-directed αλγόριθμό μας. Η Εικόνα 15.7 παρουσιάζει (μερικές εύκολα παραγόμενες) απεικονίσεις γνωστών γραφημάτων. Απεικονίσεις αυτών των γραφημάτων έχουν,

επίσης, παρουσιάζεται είτε στην κλασσική εργασία των Fruchterman και Reingold [FR91] ή στην εργασία των Lin και Yen [LY05]. Στην Εικόνα 15.8, παρουσιάζουμε μία απεικόνιση ενός Rome γραφήματος με 99 κορυφές και 135 ακμές.



**Σχήμα 15.7:** Παραδείγματα απεικονίσεων παραγόμενων από το force-directed αλγόριθμο μας. Οι απεικονίσεις σε κάθε γραμμή αντιστοιχούν στο ίδιο γράφημα. Οι απεικονίσεις στα αριστερά αποτελούν την είσοδο, ενώ στα δεξιά την έξοδο του αλγορίθμου μας. Στη λεζάντα κάθε απεικόνισης το  $(a, b)$  μεταφράζεται ως εξής:  $a$  ( $b$ , αντίστοιχα) είναι η γωνιακή ευκρίνεια (γωνιακή ευκρίνεια τομών, αντίστοιχα) υπολογισμένη σε μοίρες. Απεικονίσεις αυτών των γραφημάτων έχουν, επίσης, παρουσιαστεί είτε στην κλασική εργασία των Fruchterman και Reingold [FR91] ή στην εργασία των Lin και Yen [LY05].



**Σχήμα 15.8:** Μία απεικόνιση του Rome γραφήματος `grafo10129.99`, το οποίο αποτελείται από 99 κορυφές και 135 ακμές με γωνιακή ευκρίνεια  $20.15^\circ$  και γωνιακή ευκρίνεια τομών  $26.12^\circ$ .

## 16 Μία Εισαγωγή στην Ανίχνευση Εταιρικής Απάτης

Στο κεφάλαιο αυτό, παρουσιάζουμε βασικές έννοιες από την περιοχή της ανίχνευσης εταιρικής απάτης. Εισάγουμε τους απαραίτητους ορισμούς και παρουσιάζουμε τα προβλήματα που θα μελετηθούν στα επόμενα κεφάλαια της παρούσας διδακτορικής διατριβής.

### 16.1 Τι είναι η Εταιρική Απάτη?

Όπως ήδη αναφέραμε στην Ενότητα 11.6.3, σε επιχειρηματικό επίπεδο, η *απάτη (fraud)* ορίζεται ως η “εσκεμμένα κακή χρήση ή κατάχρηση των περιουσιακών στοιχείων μιας εταιρείας” [AoCFE12]. Με βάση τη σχέση του ατόμου που διαπράττει την απάτη με την εταιρεία, η απάτη μπορεί να διαπράττεται *εσωτερικά* ή *εξωτερικά* της εταιρείας. Η *εσωτερική απάτη (internal fraud)* αναφέρεται σε περιπτώσεις απάτης που προέρχονται από το εσωτερικό ενός οργανισμού. Η *εξωτερική απάτη (external fraud)* αναφέρεται σε περιπτώσεις απάτης που προέρχονται από άτομα που δε σχετίζονται με άμεσο τρόπο τον οργανισμό.

Οι κυριότερες μορφές απάτης με βάση μία σχετική έκθεση της KPMG [KPM03] μπορούν να κατηγοριοποιηθούν ως εξής (βλ. Πίνακα 16.1): (i) εταιρική απάτη (occupational fraud), (ii) καταναλωτική απάτη (consumer fraud), (iii) απάτη οικονομικών καταστάσεων (financial reporting fraud), (iv) κακοδιοίκηση (misconduct), (v) απάτη προμηθευτών ή τρίτων (vendor or third-party fraud) και (vi) ιατρική/ασφαλιστική απάτη (medical/insurance fraud). Δεδομένου ότι η απάτη μπορεί να μελετηθεί από πολλές διαφορετικές πλευρές, οι οποίες είναι πέραν του σκοπού της παρούσας διδακτορικής διατριβής, θα εστιάσουμε αποκλειστικά στην εταιρική απάτη.

Έρευνες σχετικές με την εταιρική απάτη δείχνουν ότι το μεγαλύτερο μέρος των αναφερόμενων περιπτώσεων απάτης έχουν διαπραχθεί από έμπιστους συνεργάτες (trusted associates). Η Εικόνα 16.1 απεικονίζει τα αποτελέσματα μιας έρευνας σχετικά με τις κυριότερες μορφές απάτης που αναφέρθηκαν παραπάνω [KPM03]. Μεταξύ αυτών των κατηγοριών, η εταιρική απάτη εμφανίζεται σε περίπου διπλάσιο ποσοστό από την καταναλωτική απάτη, η οποία ήταν η δεύτερη σε σειρά εμφάνισης. Τυπικά, η *εταιρική απάτη (occupational or employee fraud)* μπορεί να οριστεί ως η “χρήση του επαγγέλματος ενός ατόμου για προσωπικό

<b>ΚΑΤΗΓΟΡΙΕΣ ΑΠΑΤΗΣ</b>	
<b><i>Καταναλωτική Απάτη</i></b>	<b><i>Εταιρική Απάτη</i></b>
<ul style="list-style-type: none"> <li>○ Κλοπή ΑΤΜ</li> <li>○ Απάτη επιταγών</li> <li>○ Απάτη πιστωτικών καρτών</li> <li>○ Δόλια δασμολογική κατάταξη εμπορευμάτων πελατών</li> <li>○ Κλοπή ταυτότητας</li> </ul>	<ul style="list-style-type: none"> <li>○ Κλοπή επιταγών</li> <li>○ Υπεξαίρεση λογαριασμών εξόδων</li> <li>○ Απάτη μισθοδοσίας</li> <li>○ Κλοπή σύνταξης</li> <li>○ Κλοπή ή υπεξαίρεση περιουσιακών στοιχείων</li> </ul>
<b><i>Ηλεκτρονικό Έγκλημα</i></b>	<b><i>Απάτη Οικονομικών Καταστάσεων</i></b>
<ul style="list-style-type: none"> <li>○ Hacking</li> <li>○ Ηλεκτρονικό “ψάρεμα” (phishing)</li> <li>○ Ανεπιθύμητη ηλεκτρονική αλληλογραφία</li> <li>○ Κλοπή πνευματικών δικαιωμάτων</li> <li>○ Κλοπή ταυτότητας</li> </ul>	<ul style="list-style-type: none"> <li>○ Ανακριβής δηλώσεις εισοδήματος</li> <li>○ Απόκρυψη αποθεματικών και εξόδων</li> <li>○ Ανάρμοστη αναγνώριση εισοδήματος</li> <li>○ Πλημμεληματικές παραλείψεις (inadequate omissions)</li> <li>○ Ανήθικες συμφωνίες (inappropriate disclosures)</li> </ul>
<b><i>Απάτη προμηθευτών ή τρίτων</i></b>	<b><i>Ιατρική/Ασφαλιστική Απάτη</i></b>
<ul style="list-style-type: none"> <li>○ Νοθεία διαγωνισμών (bid rigging) και προσυμφωνημένες τιμές (price fixing)</li> <li>○ Δωροδοκία</li> <li>○ Εκτροπή πωλήσεων (diversion of sales)</li> <li>○ Διπλές τιμολογήσεις</li> <li>○ Εκβιασμός</li> <li>○ Λανθασμένες τιμολογήσεις και ανύπαρκτοι πωλητές (phantom vendors)</li> <li>○ Κλοπές απογραφέντων στοιχείων (inventory theft)</li> <li>○ Κλοπή πνευματικών δικαιωμάτων</li> <li>○ Δανειακές απάτες</li> <li>○ Ανταποδοτικά οφέλη (kickbacks) και σύγκρουση συμφερόντων</li> </ul>	<ul style="list-style-type: none"> <li>○ Είσπραξη προμηθειών από άσκοπες συναλλαγές (policy churning)</li> <li>○ Απάτη ιατρικών/ασφαλιστικών απαιτήσεων</li> <li>○ Απάτη εργατικής αποζημίωσης</li> </ul>
	<b><i>Κακοδιοίκηση</i></b>
	<ul style="list-style-type: none"> <li>○ Αθέμιτη εκμετάλλευση εμπιστευτικών πληροφοριών (insider trading)</li> <li>○ Σύγκρουση συμφερόντων</li> <li>○ Απάτη μισθοδοσίας</li> <li>○ Κλοπή σύνταξης</li> <li>○ Κλοπή ή υπεξαίρεση περιουσιακών στοιχείων</li> </ul>

**Πίνακας 16.1:** Κύριες Κατηγορίες Απάτης [KPM03]

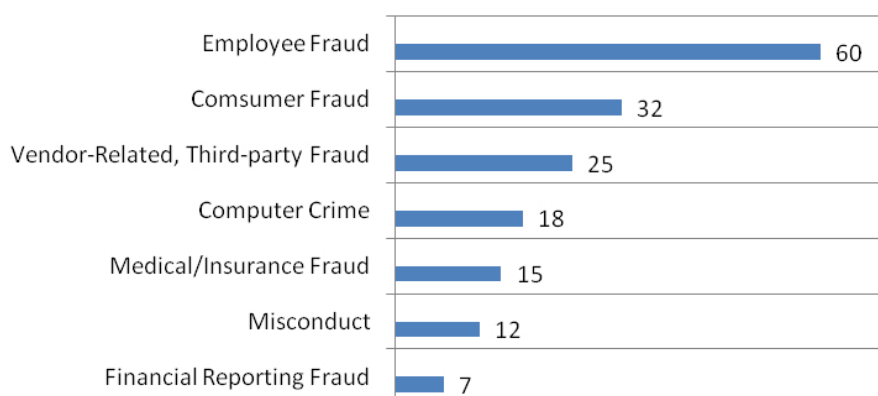
πλουτισμό μέσω της εσκεμμένης κακής χρήσης ή κατάχρησης των πόρων ή των περιουσιακών στοιχείων ενός οργανισμού” [AoCFE12]. Η Ένωση Πιστοποιημένων Ελεγκτών κατά της Απάτης (ACFE) παρουσιάζει ένα μοντέλο ταξινόμησης της απάτης, το οποίο αναφέρεται ως δέντρο απάτης (*fraud tree*), που παραθέτει τις διάφορες μορφές απάτης, ομαδοποιημένες σε κατηγορίες και υποκατηγορίες (βλ. Εικόνα 16.2). Οι τρεις κυριότερες μορφές απάτης στο δέντρο αυτό είναι οι ακόλουθες:

**Παραποίηση οικονομικών δηλώσεων (falsification of financial statements):** Ο συγκεκριμένος τύπος απάτης περιλαμβάνει περιπτώσεις στις οποίες ένας υπάλληλος παραποιεί εσκεμμένα λογαριασμούς ή αναφορές π.χ. διογκώνει στοιχεία, κέρδη ή υποτιμά αποθεματικά, έξοδα ή ζημιές. Τέτοια σχήματα προκαλούν σημαντική ζημία στην εταιρεία και συνήθως, διαπράττονται από ανώτερα διοικητικά στελέχη και διευθυντές.

**Υπεξαίρεση περιουσιακών στοιχείων (assets misappropriation):** Ο συγκεκριμένος τύπος απάτης περιλαμβάνει περιπτώσεις στις οποίες ένας υπάλληλος κλέβει ή καταχράται τους πόρους της εταιρείας, π.χ. κλοπή μετρητών, απάτες τιμολόγησης και ούτω καθεξής. Στις περισσότερες περιπτώσεις, διαπράττονται από προσωπικό χαμηλών βαθμίδων και λόγω του ότι προκαλούν αμελητέες απώλειες ανά περιστατικό, δεν μπορούν να εντοπιστούν εύκολα. Ωστόσο, τα συγκεκριμένα σχήματα είναι μακράν τα πιο συνήθη σχήματα απάτης που υφίσταται μία εταιρεία. Η *απάτη τιμολόγησης (billing fraud)* είναι η πιο συνηθισμένη σε αυτή την κατηγορία και μπορεί να προκαλέσει μία μέση απώλεια των 100.000\$ [AoCFE12]. Περιλαμβάνει κάθε σχήμα στο οποίο ένας υπάλληλος παραποιεί ή διαγράφει τιμολογήσεις, δημιουργεί τιμολογήσεις για πλασματικά αγαθά ή υπηρεσίες και ούτω καθεξής.

**Διαφθορά (corruption):** Όπως υποδηλώνει ο ίδιος ο όρος, η *διαφθορά* αναφέρεται στην παράνομη χρήση της επιρροής ενός υπαλλήλου σε μία εταιρεία παραβιάζοντας το καθήκον του με σκοπό το κέρδος. Τέτοια σχήματα περιλαμβάνουν δωροδοκίες, ανταποδοτικά οφέλη, σύγκρουση συμφερόντων και ούτω καθεξής.

Η απάτη προκαλεί απώλειες της τάξης του 5% του εισοδήματος των εταιρειών κάθε χρόνο, το οποίο στο εκτιμώμενο παγκόσμιο ακαθάριστο προϊόν του 2011, οδηγεί σε μία πιθανή συνολική απώλεια άνω των \$3.5 τρισεκατομμυρίων δολαρίων, ενώ η μέση απώλεια που προκαλείται από την εταιρική απάτη στην έρευνα ήταν 140.000\$ [AoCFE12]. Εκτός από τις οικονομικές απώλειες και το γεγονός ότι μόνο ένα μικρό ποσοστό των απωλειών μπορεί τελικά να ανακτηθεί, μία εταιρεία η οποία έπεσε θύμα απάτης έχει να αντιμετωπίσει πολλές ακόμα μη-αμελητέες συνέπειες. Μεταξύ αυτών, επηρεάζονται αρνητικά η επαγγελματική φήμη της απέναντι σε πελάτες, υπαλλήλους και άλλες οντότητες (χρηματοπιστωτικά ιδρύματα, προμηθευτές, κτλ), η εταιρική αξία, καθώς και η δημόσια και πελατειακή εμπιστοσύνη.



**Σχήμα 16.1:** Τύποι απάτης που υφίστανται επιχειρήσεις στη διάρκεια ενός έτους [KPM03].

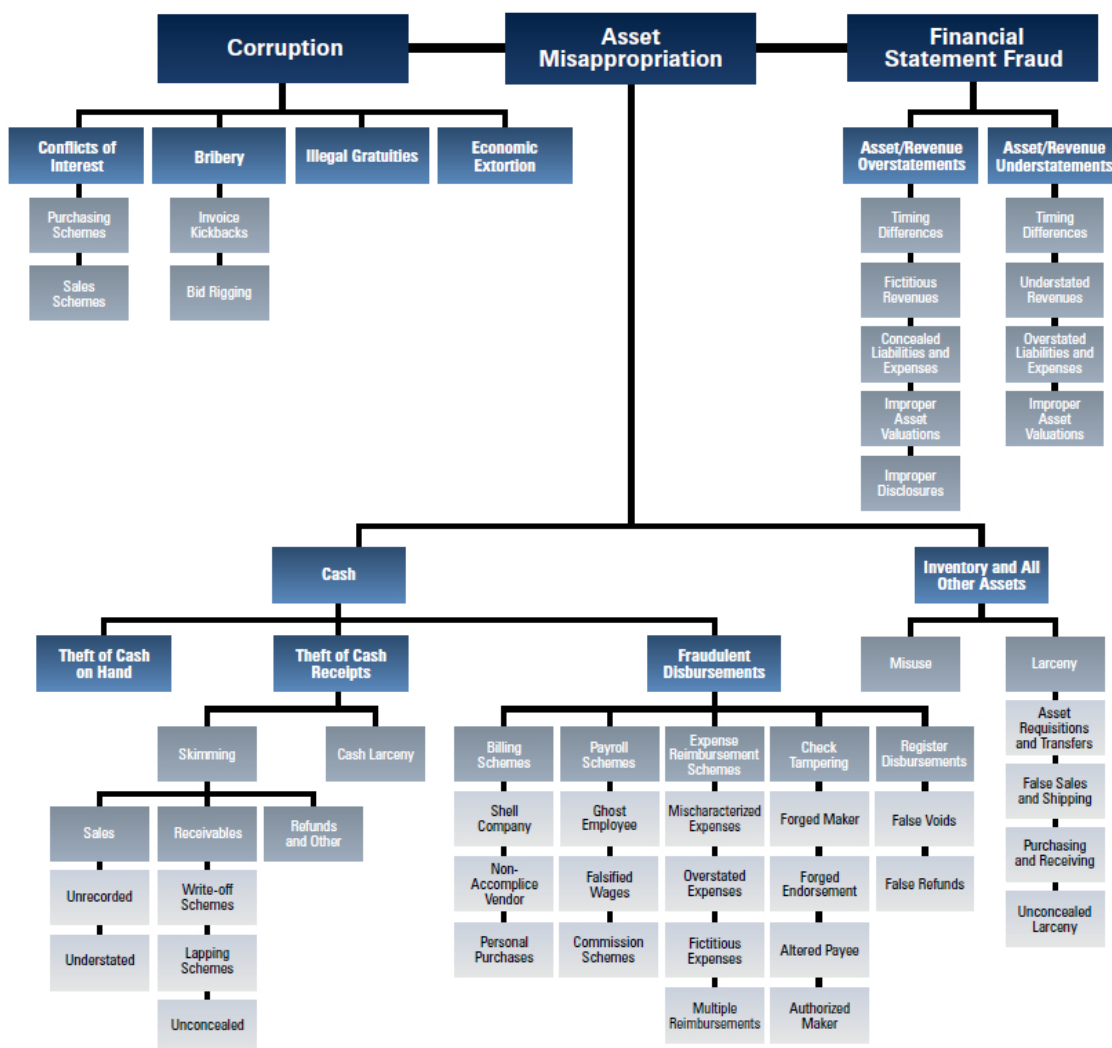
Επομένως, είναι θεμελιώδες για μία εταιρεία να αναπτύξει μηχανισμούς ενάντια στην εταιρική απάτη.

## 16.2 Τι είναι η Ανίχνευση Απάτης?

Παρόλο που οι περισσότεροι άνθρωποι μπορεί να πιστεύουν ότι η “απάτη δε θα μπορούσε ή δε θα συμβεί εδώ”, η αλήθεια είναι ότι κάθε εταιρεία, ανεξάρτητα από το μέγεθος της, το είδος ή τη χώρα που βρίσκεται, είναι ευάλωτη στην απάτη η οποία δεν περιορίζεται μόνο σε τράπεζες και χρηματοπιστωτικά ιδρύματα. Ειδικότερα, τα τελευταία χρόνια έχει καταγραφεί μία ανοδική τάση στα καταγεγραμμένα περιστατικά απάτης, με μία αύξηση της τάξης του 13% μόνο το 2011 [AoCFE12]. Ως απάντηση στον ολοένα και αυξανόμενο αριθμό περιστατικών απάτης, πολλές εταιρείες έχουν επικεντρωθεί στην *ανίχνευση της απάτης (fraud detection)*, η οποία περιλαμβάνει τις τεχνικές και τις μεθόδους που απαιτούνται για την αναγνώριση των πραγματικών ή δυνητικών περιστατικών απάτης στα πλαίσια ενός οργανισμού. Συνήθως, η στρατηγική καταπολέμησης της απάτης περιλαμβάνει τρία στάδια: (i) πρόληψη κατά της απάτης (*fraud prevention*), (ii) ανίχνευση της απάτης (*fraud detection*) και (iii) διερεύνηση των περιπτώσεων απάτης και απόκριση (*fraud investigation and response*).

Οι *τεχνικές πρόληψης* περιλαμβάνουν πολιτικές, ελέγχους και ενέργειες όπως η εκτίμηση κινδύνου (*risk assessment*), η κατάρτιση (*training*) και η ενημέρωση κατά της απάτης (*fraud awareness*), οι οποίες μπορεί να αποθαρρύνουν τους επίδοξους παραβάτες από τη διάπραξη απάτης. Περιλαμβάνει, επίσης, την οργάνωση/συγκρότηση της υπηρεσίας εσωτερικού ελέγχου και τη δημοσίευση των περιστατικών απάτης. Η *διερεύνηση* και η *απόκριση* αποτελούν το τελευταίο βήμα σε μία επιχειρηματική στρατηγική καταπολέμησης της απάτης. Κατά τη διάρκεια της έρευνας, οι ελεγκτές πρέπει να δώσουν απάντηση σε μία σειρά ερωτήσεων





Σχήμα 16.2: Το σύστημα κατηγοριοποίησης της εταιρικής απάτης όπως παρουσιάζεται στην αναφορά του έτους 2012 προς τα έθνη από τον ACFE [AoCFE12].

(ποιος, πώς, πότε και γιατί) μόλις εμφανιστούν οι πρώτες υποψίες απάτης. Στην περίπτωση που υπάρχουν αποδείξεις για τη διάπραξη απάτης, η απόκριση (δηλαδή οι διορθωτικές ενέργειες για την αποκατάσταση της προκληθείσας ζημιάς) θα πρέπει να είναι άμεση.

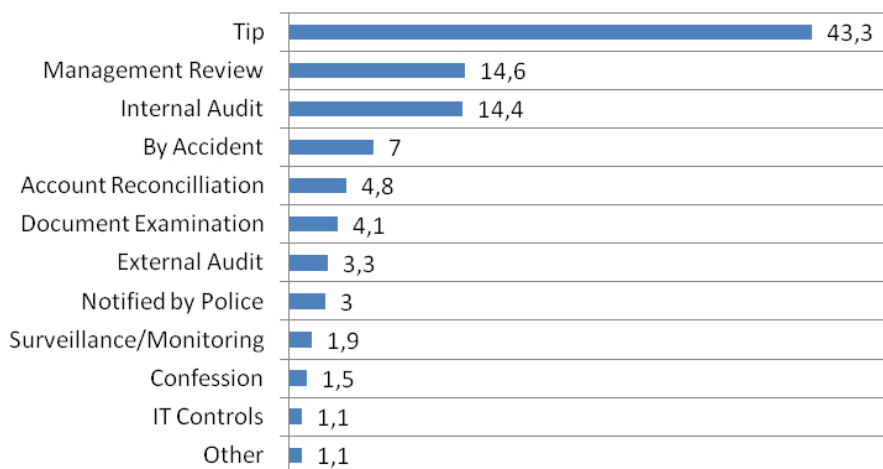
Ωστόσο, δεδομένου ότι είναι αδύνατο να αποτραπούν όλοι οι επίδοξοι παραβάτες, η απάτη θα πρέπει να εντοπιστεί το συντομότερο δυνατόν, προκειμένου να αποφευχθεί η οικονομική απώλεια και να προστατευτεί η εταιρεία, καθώς και το προσωπικό και οι πελάτες της. Οι μέθοδοι ανίχνευσης μπορεί να είναι είτε αντιδραστικές (*reactive*), είτε προδραστικές (*proactive*). Οι παραδοσιακές μέθοδοι ανίχνευσης απάτης θεωρείται ότι είναι αντιδραστικές, καθώς ξεκινούν από μία ένδειξη, υπόδειξη ή εμπιστευτική πληροφορία (*tip*), ασυνήθι-

στες οικονομικές δηλώσεις ή άλλους δείκτες που κάποιος παρατηρεί ή ακούει και μπορεί να περιλαμβάνουν, για παράδειγμα, έναν ανώνυμο μηχανισμό καταγγελιών [AA12]. Οι αντιδραστικές μέθοδοι ανίχνευσης απάτης στοχεύουν σε προκαθορισμένα σχήματα απάτης και περιλαμβάνουν διαδικαστικούς ελέγχους όπως η αναγνώριση/εύρεση των λαθών, συμπτωμάτων ή κινδύνων (red-flags).

Μία προδραστική στρατηγική ανίχνευσης απάτης περιλαμβάνει ανάλυση δεδομένων, αναγνώριση ανωμαλιών και τάσεων σε σύνολα των δεδομένων και μηχανισμούς καταγγελιών, οι οποίοι κοινοποιούν πληροφορίες σχετικές με δυνητικές περιπτώσεις απάτης. Η ανάλυση δεδομένων επιτρέπει στους ελεγκτές να αποκτήσουν μία εικόνα σχετικά με την αποτελεσματικότητα των εσωτερικών ελέγχων, την αναγνώριση παράνομων ενεργειών και τον κίνδυνο εμφάνισης κρουσμάτων απάτης. Η προδραστική ανίχνευση περιλαμβάνει, επίσης, ανάλυση συμπεριφοράς (behavioral analysis), η οποία έχει ως στόχο την αξιολόγηση του προφίλ των υπαλλήλων και την πιθανή αναγνώριση των προφίλ εκείνων που τυχόν ταιριάζουν με ένα ήδη γνωστό παραβατικό προφίλ. Σημειώνεται ότι οι μέθοδοι αυτές θα πρέπει να είναι ευέλικτες και συνεχώς εξελίξιμες με σκοπό να αμύνονται ενάντια στους επίδοξους παραβάτες και να προσαρμόζονται στις στρατηγικές τους [GTA09].

Επιπρόσθετα, η στρατηγική καταπολέμησης της απάτης κάθε οργανισμού θα πρέπει να προσαρμοστεί στο πλαίσιο στο οποίο λειτουργεί ο οργανισμός. Για παράδειγμα, ένας οργανισμός που δραστηριοποιείται σε μία χώρα με υψηλό επίπεδο διαφθοράς, θα πρέπει να λάβει συμπληρωματικά μέτρα για την πρόληψη της απάτης. Η συνεχής παρακολούθηση είναι επίσης αναγκαία, έτσι ώστε η αναγνώριση των περιστατικών απάτης να είναι άμεση και αποτελεσματική, δεδομένου ότι το μέγεθος της απάτης αυξάνεται με την πάροδο του χρόνου, καθώς ο δράστης αποκτάει την πεποίθηση ότι δεν μπορεί να εντοπιστεί. Ειδικότερα, η έκθεση της KPMG (Forensics' Fraud Risk Management) αναφέρει ότι *σε αντίθεση με αναδρομικές αναλύσεις, η συνεχής παρακολούθηση των συναλλαγών επιτρέπει σε έναν οργανισμό να αναγνωρίσει δυνητικά παράνομες συναλλαγές, για παράδειγμα, σε μία καθημερινή, εβδομαδιαία ή μηνιαία βάση. Οι οργανισμοί συχνά χρησιμοποιούν συνεχείς τεχνικές παρακολούθησης, οι οποίες επικεντρώνονται σε μικρότερου εύρους συναλλαγές ή "περιοχές" που παρουσιάζουν ιδιαίτερα σοβαρούς κινδύνους* [KPM06].

Όπως ήδη αναφέρθηκε, υπάρχει ένας μεγάλος αριθμός σχημάτων εταιρικής απάτης και δυστυχώς, κανένας μηχανισμός άμυνας δεν είναι σε θέση να αποτρέψει ή να εντοπίσει όλες τις περιπτώσεις απάτης. Για το λόγο αυτό, τα σχήματα εταιρικής απάτης μπορεί να συνεχιστούν για αρκετά μεγάλο χρονικό διάστημα μέχρι τελικά η απάτη να επιβεβαιωθεί (με ένα μέσο όρο 18 μηνών αφού προηγουμένως γίνει αντιληπτή) και η απάτη είναι πιο πιθανό να ανιχνευθεί από μία υπόδειξη ή εμπιστευτική πληροφορία (tip), παρά από οποιαδήποτε άλλη μέθοδο ανίχνευσης (βλ. Εικόνα 16.3).



**Σχήμα 16.3:** Ανίχνευση της εταιρικής απάτης όπως παρουσιάστηκε στην αναφορά του έτους 2012 προς τα έθνη από τον ACFE [AoCFE12].

Παρόλο που προηγμένες τεχνολογίες πληροφοριών έχουν ήδη ενσωματωθεί στους οργανισμούς προκειμένου να μειωθεί ο κίνδυνος της εσωτερικής απάτης, η παρακολούθηση των συστημάτων καταπολέμησης απάτης εξακολουθεί να αποτελεί ένα πρόβλημα αξιοσημείωτης δυσκολίας και απαιτεί σοβαρό εσωτερικό έλεγχο. Υπάρχουν διάφοροι παράγοντες που δυσχεραίνουν τον άμεσο εντοπισμό της απάτης, όπως για παράδειγμα, η έλλειψη εμπειρίας (κυρίως στα συμπτώματα της απάτης) και η έλλειψη ειδικών εργαλείων καταπολέμησης απάτης, η μεγάλη χρήση της δειγματοληψίας, ο περιορισμένος χρόνος και ούτω καθεξής. Ο τελευταίος παράγοντας, σε συνδυασμό με το γεγονός ότι τα δεδομένα που παράγονται από τα συστήματα καταπολέμησης απάτης είναι σε μορφή κειμένου και ο όγκος τους δεν είναι εύκολα διαχειρίσιμος σε καθημερινή βάση (αλλά ούτε και σε εβδομαδιαία βάση), μεγεθύνει το πρόβλημα της άμεσης ανίχνευσης της απάτης. Η οπτικοποίηση πληροφοριών μπορεί να είναι πολλά υποσχόμενη σε ένα τέτοιο πλαίσιο, δεδομένου ότι διευκολύνει τη γρήγορη αναγνώριση της “ύποπτης” δραστηριότητας.

### 16.3 Κίνητρο και Ορισμός του Προβλήματος

Το κίνητρό μας να εργαστούμε σε προβλήματα ανίχνευσης απάτης βασίζεται σε πραγματικά προβλήματα, τα οποία μας τέθηκαν από τους εσωτερικούς ελεγκτές ενός μεγάλου οργανισμού τηλεπικοινωνιών. Οι εσωτερικοί ελεγκτές μας κοινοποίησαν τα προβλήματα που αντιμετωπίζουν κατά τη διάρκεια της διαδικασίας ανίχνευσης απάτης καθώς και τα πρότυπα απάτης που εξετάζουν. Το κυριότερο πρόβλημα που κληθήκαμε να αντιμετωπίσουμε είναι το ακόλουθο:

**Πρόβλημα 16.3.1.** Αναζητούμε ένα συνδυασμό μεθόδων και οπτικοποιήσεων με σκοπό την ανίχνευση σχημάτων απάτης σε εταιρικά συστήματα στα οποία εμπλέκονται ζεύγη οντοτήτων (υπάλληλος και πελάτης), όπως για παράδειγμα τα συστήματα τιμολόγησης/χρέωσης, ανανέωσης συνδρομών και ούτω καθεξής.

Η εξέταση σχημάτων εταιρικής απάτης σε τέτοια εταιρικά συστήματα αποκαλύπτει ότι γεγονότα που συμβαίνουν ανά τακτά χρονικά διαστήματα μπορεί να αποτελούν ένδειξη απάτης. Για παράδειγμα, σε ένα σύστημα τιμολόγησης μιας εταιρείας, αν ένας συγκεκριμένος υπάλληλος φαίνεται να έχει μία μηνιαία δραστηριότητα (π.χ. διαγράφει/προσθέτει τιμολογήσεις, κλπ) προς το λογαριασμό ενός συγκεκριμένου πελάτη, αυτό μπορεί να θεωρηθεί ως μία “ύποπτη” περιοδική σειρά γεγονότων που πρέπει να εξεταστούν περαιτέρω. Τα γεγονότα αυτά αποκτούν μεγαλύτερη σημασία στην περίπτωση που εμφανίζονται κατά τη διάρκεια του κύκλου χρέωσης του λογαριασμού του πελάτη ή εκτός των ωρών εργασίας του υπαλλήλου. Ως εκ τούτου, η αναζήτηση περιοδικής δραστηριότητας σε σύνολα δεδομένων παίζει σημαντικό ρόλο στην αποκάλυψη σχημάτων εταιρικής απάτης.

**Πρόβλημα 16.3.2.** Αναζητούμε ένα συνδυασμό μεθόδων και οπτικοποιήσεων με σκοπό τον εντοπισμό περιοδικών γεγονότων, τα οποία μπορεί να υποκρύπτουν σχήματα εταιρικής απάτης σε σειριακά δεδομένα που παράγονται από εταιρικά συστήματα στα οποία εμπλέκονται ζεύγη οντοτήτων (υπάλληλος και πελάτης).

Ένα σχετικό πρόβλημα με το πρόβλημα της ανίχνευσης απάτης είναι η παρακολούθηση της δραστηριότητας των υπαλλήλων στο δίκτυο ενός οργανισμού. Επικεντρώνουμε κυρίως στην ανίχνευση ασυνήθιστης δραστηριότητας των υπαλλήλων όπως μη-εξουσιοδοτημένη πρόσβαση σε υπολογιστές, εταιρικά συστήματα, λογαριασμούς υπαλλήλων και πελατών και ούτω καθεξής.

**Πρόβλημα 16.3.3.** Αναζητούμε οπτικοποιήσεις οι οποίες παρακολουθούν τη δραστηριότητα των υπαλλήλων ενός οργανισμού και ανιχνεύουν πρότυπα ασυνήθιστης εταιρικής δραστηριότητας.

## 16.4 Βιβλιογραφική Επισκόπηση

Η ανίχνευση απάτης έχει μελετηθεί αρκετά στη βιβλιογραφία. Μέχρι σήμερα, υπάρχει ένας σημαντικός αριθμός εργασιών οι οποίες υιοθετούν διαφορετικές μεθόδους όπως τεχνικές εξόρυξης δεδομένων, στατιστική [CFPS99, BC00, KSM07], τεχνητή νοημοσύνη [FCS95, GC98, Kos04, LHB03] ή γενετικούς αλγόριθμους [WRW98, HKLS07, DO11] αποσκοπώντας στην ανίχνευση απάτης. Ωστόσο, στην παρουσίασή μας θα επικεντρώσουμε μόνο σε ερευνητικές εργασίες που υποστηρίζονται από οπτικοποιήσεις. Για μία πιο πλήρη επισκόπηση του

πεδίου της ανίχνευσης απάτης, ο αναγνώστης παραπέμπεται στις εργασίες [BH02, KLSH04, ZZ04, YWW<sup>+</sup>07, HRRV08, Wan10].

Το σύστημα ADS (NASD Regulation Detection System) [KSH<sup>+</sup>98, Sen00, SGS<sup>+</sup>02] παρακολουθεί τις τάσεις και τη διαμόρφωση των τιμών στη χρηματιστηριακή αγορά του Nasdaq και προσπαθεί να ανιχνεύσει πρότυπα και πρακτικές παράνομης δραστηριότητας. Το σύστημα είναι εφοδιασμένο με μηχανισμούς ανίχνευσης και αναγνώρισης, οι οποίοι χρησιμοποιούν τεχνικές αναγνώρισης προτύπων και τεχνικές εξόρυξης δεδομένων, αντίστοιχα. Τα αποτελέσματα οπτικοποιούνται εφαρμόζοντας διάφορες τεχνικές. Το σύστημα FAIS (Financial Crimes Enforcement Network AI System) [GS95, SGW<sup>+</sup>95] ανιχνεύει τυχόν νομιμοποίηση εσόδων από παράνομες δραστηριότητες (money laundering) σε δεδομένα συναλλαγών με μετρητά. Το σύστημα υποστηρίζει μεταξύ άλλων δύο βασικές λειτουργίες. Η πρώτη επιτρέπει τη χρήση ειδικών επερωτημάτων (queries) από το χρήστη, ενώ η δεύτερη οπτικοποιεί ύποπτες περιπτώσεις και λογαριασμούς με βάση προκαθορισμένους κανόνες.

Οι Chang et al. [CLG<sup>+</sup>08] παρουσίασαν το σύστημα WireVis, το οποίο χρησιμοποιεί διαδραστικές οπτικοποιήσεις τραπεζικών εμβασμάτων (financial wire transactions) και στοχεύει στην ανίχνευση χρηματοοικονομικής απάτης. Η προσέγγισή τους υποστηρίζει διαδραστικές οπτικοποιήσεις προβολής του δικτύου, θερμικούς χάρτες, εργαλεία αναζήτησης καθώς και ένα μηχανισμό οπτικοποίησης αποσκοπώντας στην αποκάλυψη των σχέσεων ανάμεσα σε λογαριασμούς, λέξεις-κλειδιά, χρόνο και πρότυπα δραστηριότητας. Οι Huang et al. [HLN09] παρουσίασαν μία μεθοδολογία οπτικής ανάλυσης (framework of visual analytics) για την ασφάλεια των χρηματιστηριακών αγορών. Πιο συγκεκριμένα, χρησιμοποίησαν τρισδιάστατες δένδροειδείς απεικονίσεις (3D-treemaps) για την οπτικοποίηση της τρέχουσας τιμής των μετοχών και τον εντοπισμό μετοχών που παράγουν ένα ασυνήθιστο πρότυπο συναλλαγών. Οι Eberle and Holder [EH09] εισήγαγαν ένα μοντέλο ανίχνευσης ανωμαλιών βασισμένο σε γραφήματα (graph-based anomaly detection model), το οποίο χρησιμοποιεί προσεγγίσεις από τη θεωρία πληροφοριών, τις πιθανότητες και τη μέγιστη μερική υποδομή (maximum partial substructure), για την ανίχνευση δομικών ανωμαλιών σε εταιρικές συναλλαγές και διαδικασίες που προκαλούνται από υπαλλήλους εταιριών. Ο Luell [Lue10] συνδυάζει τεχνικές εξόρυξης δεδομένων και οπτικοποιήσεις για την ανίχνευση συμβουλευτικής απάτης σε πελάτες (client advisor fraud) ενός χρηματοπιστωτικού ιδρύματος.

Οι Di Giacomo et al. [DDL10] παρουσίασαν ένα σύστημα βασισμένο σε τεχνικές οπτικοποίησης δικτύων με σκοπό τον εντοπισμό οικονομικών εγκλημάτων (financial crimes). Ειδικότερα, το σύστημα επιτρέπει τον εντοπισμό των ήδη γνωστών προτύπων σε δεδομένα που φαίνεται να προέρχονται από ύποπτες συναλλαγές (STRs). Οι Didimo et al. [DLMP11] ανέπτυξαν ένα σύστημα, το οποίο παρείχε στον αναλυτή αποτελεσματικά εργαλεία για την ανίχνευση οικονομικών εγκλημάτων, όπως η νομιμοποίηση εσόδων από παράνομες δραστη-

ριότητες και απάτες. Το SynerScope [Syn11] αποτελεί ένα εργαλείο οπτικοποίησης για την ανάλυση “Μεγάλων Δεδομένων (Big Data)”, ικανό να εντοπίσει σχήματα χρηματοοικονομικής απάτης. Οι Didimo et al. [DLM12] παρουσίασαν το σύστημα VIS4AUI, το οποίο συλλέγει χρηματοοικονομικές πληροφορίες σχετικές με τραπεζικές σχέσεις και υψηλής αξίας συναλλαγές και προσπαθεί να ανιχνεύσει νομιμοποίηση εσόδων από παράνομες δραστηριότητες και οικονομικά εγκλήματα.

Το Link Analysis Workbench (LAW) [WBH<sup>+</sup>03] αναπτύχθηκε για τον εντοπισμό περιπτώσεων τρομοκρατίας και άλλης εγκληματικής δραστηριότητας σε ελλιπή δεδομένα ή σε δεδομένα με “θόρυβο”. Το σύστημα εφαρμόζει τεχνικές αναγνώρισης προτύπων και αναγνώρισης προτύπων με γραφήματα (graph-pattern matching) και βασίζεται στο ταίριασμα με ήδη γνωστά πρότυπα ενδιαφέροντος και την αναγνώριση νέων προτύπων. Οι Xu και Chen [XC05] παρουσίασαν μία μεθοδολογία για την αυτοματοποιημένη ανάλυση εγκληματολογικών δικτύων, η οποία υποστηρίζει οπτικοποιήσεις, με στόχο την εξόρυξη πολύτιμης γνώσης για το εγκληματολογικό δίκτυο από ένα μεγάλο όγκο δεδομένων. Οι Stasko et al. [SGL08] ανέπτυξαν ένα σύστημα οπτικοποίησης, το οποίο διευκολύνει την πιο αποτελεσματική εξέταση αναφορών καθώς και εγγράφων, με στόχο τον εντοπισμό δυνητικών απειλών που κρύβονται πίσω από τις γραμμές τους. Άλλα εμπορικά εργαλεία που εφαρμόζουν τεχνικές οπτικοποίησης δικτύων στα πλαίσια της εγκληματολογικής ανάλυσης είναι τα ακόλουθα: I2 Analyst’s Notebook [IAN], Netmap [Net], XANALYS Link Explorer [XLE].

Για την αναγνώριση περιοδικών προτύπων σε σειριακά δεδομένα, οι Carlis και Konstan [CK98] πρότειναν μία προσέγγιση, στην οποία τα σειριακά χαρακτηριστικά των δεδομένων απεικονίζονται κατά μήκος ενός σπειροειδή άξονα, ενώ τα περιοδικά κατά μήκος της ακτίνας του. Οι Weber et al. [WAM01] παρουσίασαν μία προσέγγιση, η οποία χρησιμοποιεί μία σπειροειδή απεικόνιση για την οπτικοποίηση μεγάλου όγκου δεδομένων χρονοσειρών με σκοπό την αποκάλυψη περιοδικών δομών. Στην προσέγγισή τους, ο άξονας του χρόνου απεικονίζεται ως σπειροειδής καμπύλη, ενώ οι υπόλοιπες “ιδιότητες” των δεδομένων απεικονίζονται με σημεία, χρώματα, ράβδους ή γραμμές. Οι Bertini et al. [BHL07] παρουσίασαν το εργαλείο SpiralView, το οποίο οπτικοποιεί σε μία σπειροειδή απεικόνιση τη διασπορά των ειδοποιήσεων του δικτύου στο χρόνο και βοηθάει στην αποκάλυψη πιθανών περιοδικών μοτίβων. Οι Suntinger et al. [SOSG08] ανέπτυξαν ένα σύστημα οπτικοποίησης, το οποίο αναπαριστά γεγονότα σε ένα κυλινδρικό κανάλι (cylindrical tunnel). Η οπτικοποίηση ανιχνεύει περιστατικά όπως συγκεκριμένα μοτίβα και ανωμαλίες, οι οποίες μπορούν να επηρεάζουν την απόδοση μιας εταιρείας. Σχετικά με την οπτικοποίηση χρονοσειρών και τις διαθέσιμες τεχνικές, μία επισκόπηση μπορεί να αναζητηθεί στις εργασίες [ABM<sup>+</sup>07, MS03, SC00]

## 16.5 Μεθοδολογία και Ορολογία

Ένα από τα κυριότερα προβλήματα που κληθήκαμε να αντιμετωπίσουμε κατά τη διάρκεια της ερευνητικής μας προσπάθειας στην περιοχή της ανίχνευσης επιθέσεων ήταν η μη ομοιόμορφη μορφή των αρχείων καταγραφής που παράγονται από τα διάφορα εταιρικά συστήματα (π.χ. διαφορετικά σχήματα βάσεων δεδομένων, διαφορετικά αρχεία καταγραφής, κτλ). Για να αντιμετωπίσουμε αυτό το πρόβλημα, αναπτύχθηκαν αναλυτές κειμένων (parsers), οι οποίοι μετατρέπουν σε ομοιόμορφη μορφή τα αρχεία καταγραφής που παράγονται από την πλειοψηφία των συστημάτων της εταιρείας. Στη συνέχεια, σχεδιάστηκε μία βάση δεδομένων, στην οποία αποθηκεύονται οι μετασχηματισμένες εγγραφές.

Ένα δεύτερο πρόβλημα που είχαμε να αντιμετωπίσουμε ήταν ο τεράστιος όγκος δεδομένων τα οποία έπρεπε να οπτικοποιηθούν. Δεδομένου ότι τα πιθανά σχήματα εταιρικής απάτης βρίσκονται καλά κρυμμένα στον όγκο των δεδομένων της εταιρείας, αναζητούσαμε μία προσέγγιση η οποία θα παρουσίαζε στον ελεγκτή όλα τα καταγεγραμμένα γεγονότα ανάλογα με τη σοβαρότητά τους. Δεδομένου ότι η ταυτόχρονη οπτικοποίηση τεράστιων συνόλων δεδομένων είναι αναποτελεσματική και προκαλεί σύγχυση, στα εργαλεία που αναπτύξαμε, χρησιμοποιήσαμε ένα συνδυασμό μηχανισμών ανίχνευσης απάτης (με βάση πρότυπα απάτης που συνέστησαν οι ελεγκτές) με οπτικοποιήσεις που υποστηρίζουν κινούμενες εικόνες (animations).

Όπως έχει ήδη αναφερθεί, τα γεγονότα στα οποία εστιάζουμε εμπλέκουν δύο οντότητες (υπάλληλος και πελάτης). Κατά συνέπεια, μία εγγραφή στο σύστημά μας αποτελείται (μεταξύ άλλων δευτερευόντων πεδίων) από ένα πεδίο που υποδηλώνει την ημερομηνία και την ώρα (time-stamp), το ID του υπαλλήλου, το ID του πελάτη και την ενέργεια που πραγματοποιήθηκε. Κάθε εγγραφή μπορεί να έχει παραχθεί από μία τηλεφωνική επικοινωνία μεταξύ του υπαλλήλου και του πελάτη, μία συναλλαγή που εμπλέκει τις δύο οντότητες και ούτω καθεξής.

Ένα γεγονός (*event*)  $e$  εμπλέκει ένα ζεύγος υπαλλήλου-πελάτη και ορίζεται ως μία τετράδα  $(t, u, c, a)$ , τέτοια ώστε:

- $t$  είναι η ημερομηνία και η ώρα που συνέβη το γεγονός,
- $u$  είναι το ID του υπαλλήλου,
- $c$  είναι το ID του πελάτη,
- $a$  είναι η ενέργεια που πραγματοποιήθηκε από τον υπάλληλο.

Για μία συγκεκριμένη τετράδα  $(t, u, c, a)$ , λέμε ότι ο πελάτης  $c$  *σχετίζεται* με το γεγονός  $e$  και, επίσης, *σχετίζεται* με τον υπάλληλο  $u$ .





## 17 Ανίχνευση Περιοδικότητας σε Σειριακά Δεδομένα Μέσω Οπτικοποίησης

Στο κεφάλαιο αυτό, παρουσιάζουμε τεχνικές απεικόνισης που αποσκοπούν στον εντοπισμό περιοδικών μοτίβων σε σειριακά δεδομένα, τα οποία προέρχονται από συστήματα που εμπλέκουν ζεύγη οντοτήτων (υπάλληλος και πελάτης). Οι τεχνικές που παρουσιάζουμε αποσκοπούν στην ανίχνευση εταιρικής απάτης. Προς αυτή την κατεύθυνση, μετράμε την περιοδικότητα εισάγοντας μία νέα μετρική κατάλληλα ορισμένη για τον υπολογισμό της συχνότητας με την οποία εμφανίζονται γεγονότα που σχετίζονται με ένα συγκεκριμένο ζεύγος υπαλλήλου-πελάτη. Η κύρια μέθοδος οπτικοποίησης καταλήγει σε απεικονίσεις που αποτελούνται από ομόκεντρους κύκλους, των οποίων οι ακτίνες αντιστοιχούν στην περιοδικότητα της δραστηριότητας κάθε ζεύγους. Γεγονότα που θεωρούνται “ύποπτα” διακρίνονται εύκολα, καθώς μετακινούνται προς το κέντρο του συστήματος των κύκλων. Επίσης, παρουσιάζουμε μία force-directed προσέγγιση προκειμένου να επιτύχουμε μία καλύτερη τοποθέτηση των γεγονότων στην οπτικοποίηση.

### 17.1 Υπολογισμός Περιοδικότητας

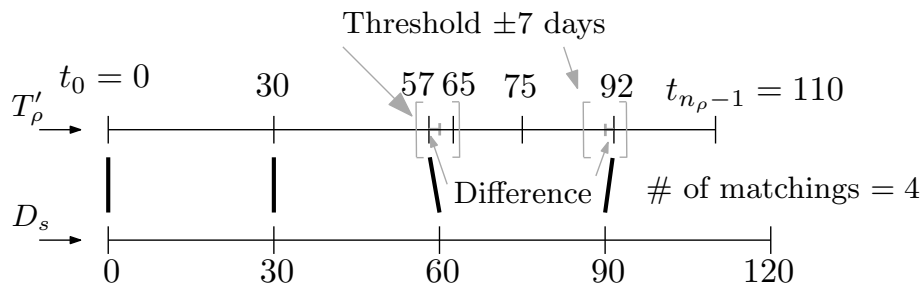
Έστω  $e = (t, u, c, a)$  ένα γεγονός που εμπλέκει έναν υπάλληλο  $u$  και έναν πελάτη  $c$  (βλ. Ενότητα 16.5). Για ένα ζεύγος υπαλλήλου-πελάτη  $(u, c)$ , μία σειρά γεγονότων (event-series)  $T_{u,c} = \{e_{(u,c)}^1, e_{(u,c)}^2, \dots\}$  είναι μία ακολουθία γεγονότων  $e_{(u,c)}^i = (t_i, u, c, a_i)$ , τα οποία σχετίζονται με τον πελάτη  $c$  και τον υπάλληλο  $u$  (υποθέτουμε ότι η σειρά είναι ταξινομημένη με βάση την ημερομηνία των γεγονότων που την αποτελούν). Πριν παραχθεί η απεικόνιση, το σύστημα επεξεργάζεται αυτές τις σειρές γεγονότων και υπολογίζει την περίοδο της δραστηριότητας για κάθε ζεύγος οντοτήτων.

Πιο συγκεκριμένα, για κάθε ζεύγος οντοτήτων, εισάγουμε μία μετρική, η οποία υπολογίζει την περιοδικότητα με κάποιο βαθμό εμπιστοσύνης. Για λόγους απλοποίησης της περιγραφής, έστω  $\rho = (u, c)$  ένα ζεύγος οντοτήτων και  $n_\rho$  είναι ο αριθμός των γεγονότων που σχετίζονται με το ζεύγος  $\rho$ . Υποθέτουμε ότι τα γεγονότα  $e_\rho^i, e_\rho^{i+1}, i = 1, \dots, n_\rho - 1$  είναι διαδοχικά και έστω  $d_\rho^{i,i+1}$  η χρονική διαφορά τους (έστω υπολογισμένη σε μέρες). Μία σειρά γεγονότων  $T'_\rho = (t_\rho^1, \dots, t_\rho^{n_\rho-1})$  παράγεται αναθέτοντας σε κάθε γεγονός  $e_\rho^i$  μία τιμή  $t_\rho^i$  με

βάση τον παρακάτω τύπο:

$$t_i^\rho = \begin{cases} \sum_{j=1}^{i-1} d_\rho^{j,j+1} & \text{εάν } 1 \leq i < n_\rho \\ 0 & \text{εάν } i = 0 \end{cases}$$

Για δοθείσα τιμή περιόδου  $s$ , η ιδανική σειρά γεγονότων  $D_s = (0, s, 2s, \dots)$  ορίζεται από τις ημερομηνίες που προκύπτουν στην περίπτωση που τα γεγονότα μεταξύ των οντοτήτων του ζεύγους  $\rho$  εμφανίζονται ανά χρονικά διαστήματα ίσα με  $s$  (βλ. Εικόνα 17.1). Για παράδειγμα, στην περίπτωση περιόδου 30 ημερών, η ιδανική σειρά γεγονότων είναι  $D_{30} = (0, 30, 60, \dots)$ .



**Σχήμα 17.1:** Η ευθεία  $T'_\rho$  αντιστοιχεί στη σειρά γεγονότων για το ζεύγος υπαλλήλου-πελάτη, ενώ η ευθεία  $D_s$  στην ιδανική σειρά γεγονότων για περίοδο 30 ημερών.

Έστω  $t \in T'_\rho$  και  $\lambda \in D_s$ . Λέμε ότι οι χρονικές στιγμές  $t$  και  $\lambda$  *ταιριάζουν μεταξύ τους* με βάση μία οριακή τιμή (*threshold*)  $\tau \in [1, s/2)$ , εάν ισχύει ότι  $t \in [\lambda - \tau, \lambda + \tau]$ . Έστω  $N_\rho^\tau$  το σύνολο των χρονικών στιγμών της ιδανικής σειράς γεγονότων  $D_s$ , οι οποίες μπορούν να ταιριαστούν με χρονικές στιγμές της σειράς γεγονότων  $T'_\rho$ , δηλαδή  $N_\rho^\tau = \{\lambda \in D_s : \exists t \in T'_\rho \tau.\omega, t \text{ και } \lambda \text{ ταιριάζουν}\}$ . Παραβιάζοντας ελαφρώς τη σημειολογία, ορίζουμε την πληθικότητα του συνόλου  $N_\rho^\tau$  ως τον *αριθμό των ταιριασμάτων* του ζεύγους  $\rho$ . Έστω, επίσης, η συνάρτηση  $diff_\rho^\tau : N_\rho^\tau \rightarrow \mathbb{R}$  με:

$$diff_\rho^\tau(\lambda) = \min\{|t - \lambda| : t \in T'_\rho, \lambda \in N_\rho^\tau \text{ και } (t, \lambda) \text{ ταιριάζουν}\}$$

Ο βαθμός εμπιστοσύνης του ζεύγους οντοτήτων  $\rho$ , για δοθείσα τιμή περιόδου  $s$  και οριακή τιμή ταιριασμάτων  $\tau$  δίνεται από τον ακόλουθο τύπο:

$$confidence(\rho, s, \tau) = \frac{\sum_{\lambda \in N_\rho^\tau} 1 - \frac{diff_\rho^\tau(\lambda)}{\tau}}{|N_\rho^\tau|}$$

Παρατηρείστε ότι ο βαθμός εμπιστοσύνης ανήκει στο διάστημα  $[0, 1]$ . Προφανώς, εάν η σειρά γεγονότων  $T'_\rho$  ταυτίζεται με την ιδανική σειρά  $D_s$ , τότε  $confidence(\rho, s, \tau) = 1, \forall \tau \in [1, s/2)$ . Για να επιτύχουμε μεγαλύτερη ακρίβεια στον υπολογισμό του βαθμού εμπιστοσύνης του ζεύγους  $\rho$  για δοθείσα τιμή περιόδου  $s$  και οριακή τιμή  $\tau$ , εναλλακτικά υπολογίζουμε

το βαθμό εμπιστοσύνης του ζεύγους  $\rho$  για κάθε ιδανική σειρά γεγονότων  $D_s^i = (i, s+i, \dots)$ ,  $i = 0, \dots, s/2$  και λαμβάνουμε υπόψη μας εκείνη που μεγιστοποιεί το βαθμό εμπιστοσύνης.

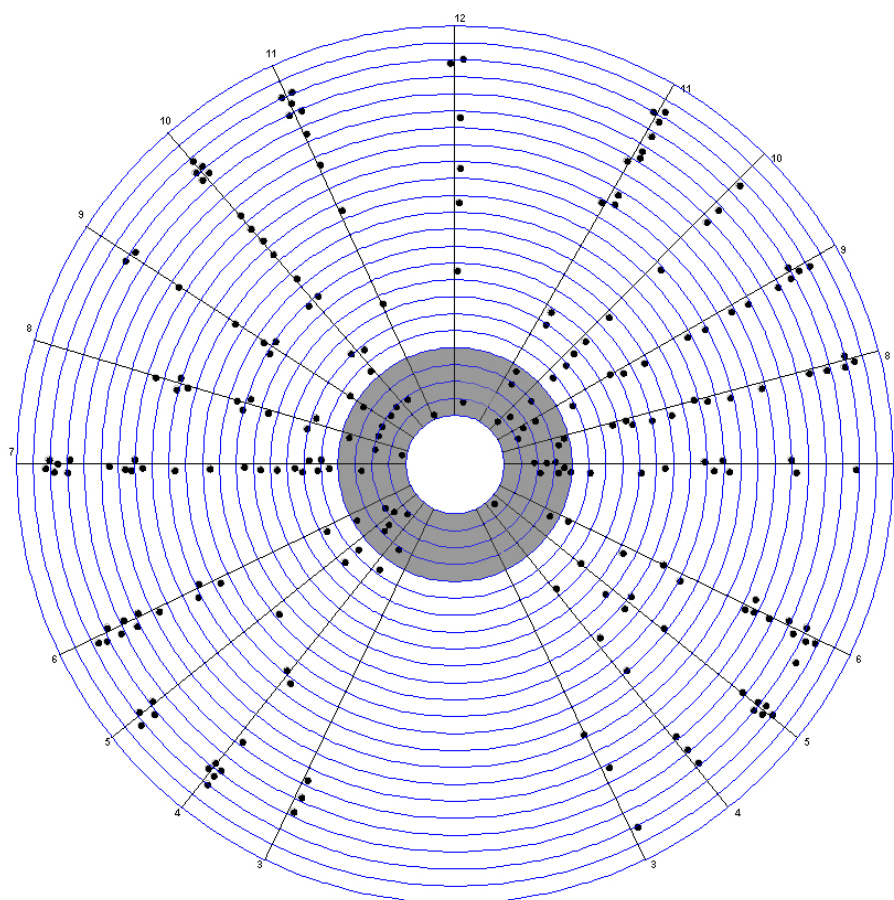
Για δοθέν ζεύγος οντοτήτων  $\rho$  και προκαθορισμένη οριακή τιμή  $\tau$ , υπολογίζουμε το βαθμό εμπιστοσύνης για κάθε τιμή περιόδου  $s \in [1, s_{max}]$ , όπου η τιμή  $s_{max}$  αντιστοιχεί στη μέγιστη τιμή περιοδικότητας, η οποία ορίζεται από τον ελεγκτή. Έχοντας υπολογίσει όλες τις τιμές εμπιστοσύνης για το ζεύγος  $\rho$ , η *περιοδικότητα* του ζεύγους  $\rho$  (ως προς την προκαθορισμένη οριακή τιμή  $\tau$ ) ισούται με την τιμή περιόδου που μεγιστοποιεί το βαθμό εμπιστοσύνης της.

Σημειώνεται ότι για κάθε ζεύγος οντοτήτων, παράγουμε μία οπτικοποίηση όμοια με αυτή της Εικόνας 17.1, με σκοπό να παρουσιάσουμε τις σειρές γεγονότων που σχετίζονται με το συγκεκριμένο ζεύγος και τα ταιριάσματα που πραγματοποιήθηκαν με την ιδανική σειρά γεγονότων.

## 17.2 Οπτικοποίηση Περιοδικότητας

Η κύρια μέθοδος οπτικοποίησης παρουσιάζεται στην Εικόνα 17.2 στην οποία αναζητούμε μηνιαία περιοδική δραστηριότητα, δηλαδή η περίοδος ισούται με 30 ημέρες. Η οπτικοποίηση αποτελείται από ένα σύστημα ομόκεντρων κύκλων των οποίων η ακτίνα αντιστοιχεί σε διαφορετικές τιμές περιοδικότητας. Οι κόμβοι της οπτικοποίησης αντιστοιχούν σε ζεύγη οντοτήτων. Ο εξωτερικός κύκλος αντιστοιχεί σε περίοδο 8 ημερών, ενώ ο πιο εσωτερικός σε περίοδο 31 ημερών. Σημειώνεται ότι υπολογίζουμε μόνο τιμές περιοδικότητας που είναι μεγαλύτερες από την οριακή τιμή, η οποία για την οπτικοποίηση της Εικόνας 17.2 έχει καθοριστεί στις 7 ημέρες. Ωστόσο, η τιμή αυτή καθορίζεται από τον ελεγκτή. Με αυτή τη διάταξη, οι κόμβοι με περιοδικότητα 30 και 31 ημερών μετακινούνται προς το κέντρο του συστήματος.

Το σύστημα είναι, επίσης, χωρισμένο σε κυκλικούς τομείς, οι οποίοι αντιστοιχούν σε διαφορετικό αριθμό ταιριασμάτων με την ιδανική σειρά γεγονότων για κάθε περίοδο, όπως περιγράφηκε παραπάνω. Για μία πιο ομοιόμορφη κατανομή των κόμβων στο σύστημα των κύκλων, οι κόμβοι των οποίων ο αριθμός των ταιριασμάτων είναι μεγαλύτερος από τη διάμεσο βρίσκονται στο πάνω ημικύκλιο, ενώ οι υπόλοιποι στο κάτω ημικύκλιο. Ο μέγιστος αριθμός ταιριασμάτων αντιστοιχεί στο μέσο του πάνω ημικυκλίου. Στην οπτικοποίηση, αγνοούμε τους κόμβους των οποίων οι σειρές των γεγονότων είχαν το πολύ δύο ταιριάσματα με την ιδανική σειρά γεγονότων, καθώς σε αντίθετη περίπτωση θα είχαμε πάντα ένα τέλειο ταιριασμα με τιμή δύο. Οι γκρι-σκιασμένες περιοχές της Εικόνας 17.2 περιέχουν κόμβους που πιθανόν να κρύβουν ύποπτη δραστηριότητα (εξαιτίας των τιμών περιοδικότητάς τους) και χρειάζεται να εξεταστούν περαιτέρω. Έχουμε, επίσης, επιλέξει να απεικονίσουμε ολόκληρη

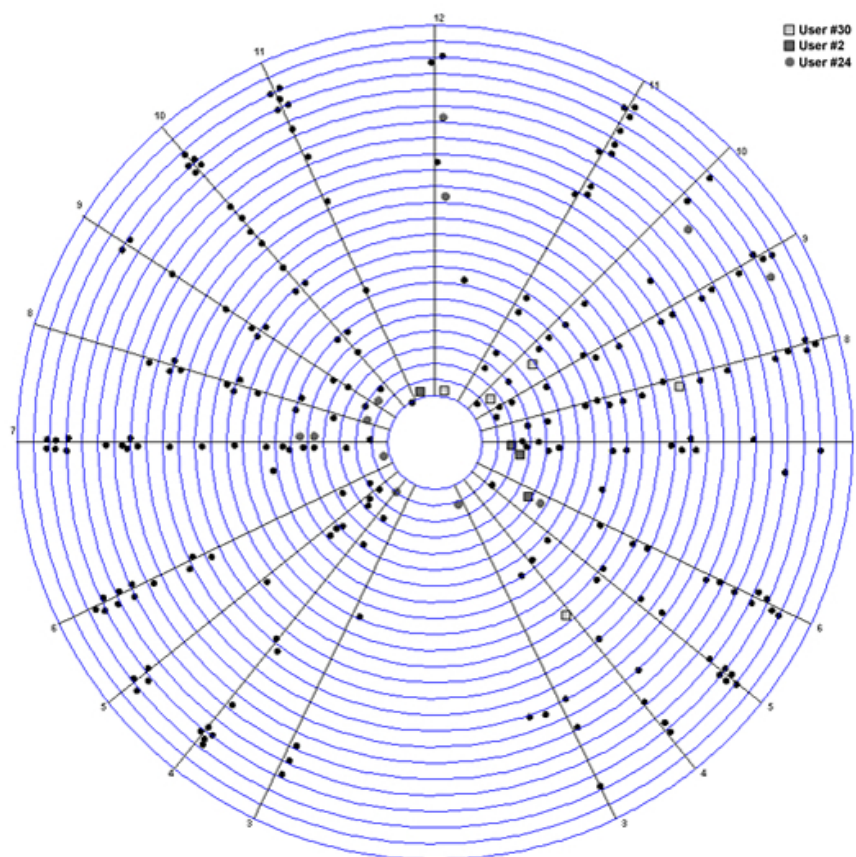


**Σχήμα 17.2:** Ένα σύστημα ομόκεντρων κύκλων στο οποίο κάθε ακτίνα αντιστοιχεί σε μία τιμή περιοδικότητας. Οι γκρι-σκιασμένες περιοχές είναι εκείνες που πρέπει να εξεταστούν πρώτα για την ύπαρξη ύποπτης δραστηριότητας.

ρους τους δακτυλίους που αντιστοιχούν σε περιόδους μεγαλύτερες των 27 ημερών, ακόμα και σε περιπτώσεις με μικρό αριθμό ταιριασμάτων, καθώς ο μικρός αριθμός ταιριασμάτων μπορεί να σημαίνει ότι πιθανή ύποπτη δραστηριότητα έχει μόλις ξεκινήσει.

Στην περίπτωση που ο ελεγκτής επιλέξει έναν κόμβο, όλοι οι κόμβοι του γραφήματος που αντιστοιχούν στον ίδιο υπάλληλο απεικονίζονται με ίδιο χρώμα ή σχήμα. Με τον τρόπο αυτό, ο ελεγκτής μπορεί να διακρίνει εάν ο ίδιος υπάλληλος εμφανίζει και άλλα περιστατικά “ύποπτης” δραστηριότητας. Επίσης, οι κόμβοι που αντιστοιχούν στα τρία “πιο επικίνδυνα” ζεύγη οντοτήτων απεικονίζονται με διαφορετικά χρώματα και σχήματα, έτσι ώστε να διακρίνονται εύκολα από τους υπόλοιπους κόμβους της απεικόνισης, όπως φαίνεται στην Εικόνα 17.3.

Για τη βελτίωση της αναγνωσιμότητας των παραγόμενων απεικονίσεων, εφαρμόσαμε



**Σχήμα 17.3:** Οι κόμβοι που αντιστοιχούν στα τρία πιο “επικίνδυνα” ζεύγη οντοτήτων απεικονίζονται με διαφορετικά χρώματα και σχήματα.

μία παραλλαγή του κλασικού force-directed αλγόριθμου του Eades [Ead84] σε συνδυασμό με την τεχνική της οπτικοποίησης μας. Όπως αναφέρθηκε στην Ενότητα 11.5.2, ένας force-directed αλγόριθμος μοντελοποιεί τους κόμβους του γραφήματος σαν ηλεκτρικά φορτισμένα σωματίδια που απωθούνται μεταξύ τους και τις ακμές σαν ελατήρια που έλκουν γειτονικούς κόμβους. Ωστόσο, πριν προχωρήσουμε με την αναλυτική περιγραφή του αλγορίθμου μας, θα εισάγουμε την απαραίτητη ορολογία. Δοθείσας μιας απεικόνισης  $\Gamma(G)$  ενός μη κατευθυνόμενου γραφήματος  $G = (V, E)$ , συμβολίζουμε με  $p_u = (x(u), y(u))$  τη θέση του κόμβου  $u \in V$  στο επίπεδο. Όπως και στην Ενότητα 15.1, το διάνυσμα από το σημείο  $p_u$  προς το σημείο  $p_v$ , το οποίο έχει μοναδιαίο μήκος συμβολίζεται με  $\overrightarrow{p_u p_v}$ , όπου  $u, v \in V$ .

Στην προσέγγισή μας, προσθέτουμε εικονικούς κόμβους σε κάθε κύκλο και κατά μήκος των ευθειών που χωρίζουν τους κύκλους σε κυκλικούς τομείς. Κάθε εικονικός κόμβος αντιστοιχεί στον αριθμό των ταιριασμάτων για δοθείσα τιμή περιοδικότητας. Στη συνέχεια, χρησιμοποιούμε ελατήρια για να ενώσουμε κάθε κόμβο με τον εικονικό κόμβο που αντιστοιχεί

στον αριθμό των ταιριασμάτων του στον αντίστοιχο κύκλο περιόδου. Τα ελατήρια ακολουθούν το λογαριθμικό νόμο (αντί του αντίστοιχου νόμου του Hooke), ώστε να αποφύγουμε να ασκήσουμε ισχυρές δυνάμεις σε απομακρυσμένους κόμβους. Οι ελκτικές δυνάμεις υπολογίζονται από τον παρακάτω τύπο:

$$\mathcal{F}_{spring}(p_u, p_v) = C \cdot \log \frac{\|p_u - p_v\|}{\ell} \cdot \overrightarrow{p_u p_v}, (u, v) \in E$$

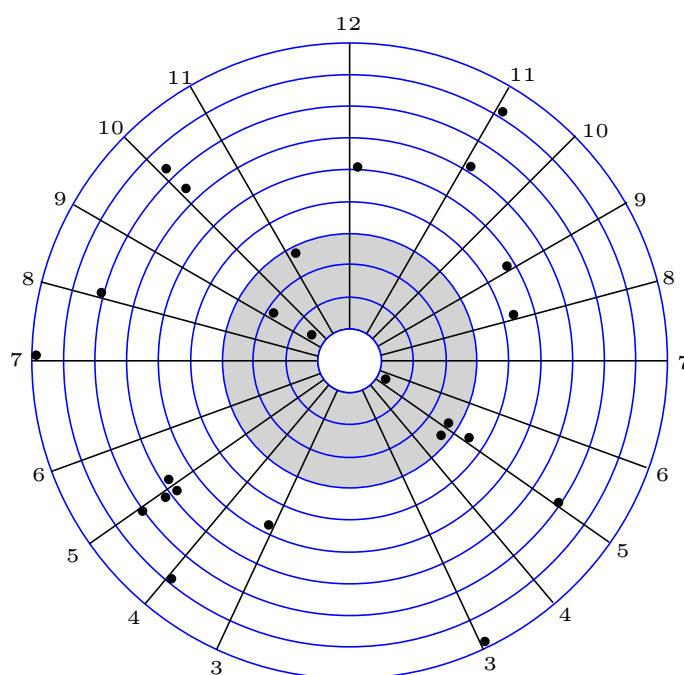
όπου οι σταθερές  $C$  και  $\ell$  αντιστοιχούν στη *σκληρότητα (stiffness)* και το *φυσικό μήκος (natural length)* των ελατηρίων, αντίστοιχα. Χρησιμοποιούμε, επίσης, απωθητικές δυνάμεις μεταξύ των κόμβων της οπτικοποίησης για να αποφύγουμε επικαλύψεις κόμβων. Οι απωθητικές δυνάμεις υπολογίζονται από τον παρακάτω τύπο:

$$\mathcal{F}_{rep}(p_u, p_v) = \frac{C_p}{\|p_u - p_v\|^2} \cdot \overrightarrow{p_u p_v}, u, v \in V$$

όπου  $C_p$  είναι η σταθερά *απόθησης (repulsion)*. Το σύνολο των δυνάμεων που περιγράφηκαν εξασφαλίζει ότι σε μία κατάσταση ισορροπίας του μοντέλου, οι κόμβοι θα απεικονίζονται κοντά στους αντίστοιχους κύκλους τους και ειδικότερα, κοντά στους εικονικούς κόμβους που “περιγράφουν” τον αριθμό των ταιριασμάτων τους. Σημειώνεται ότι δεν εφαρμόζουμε δυνάμεις στις εικονικούς κόμβους. Επομένως, η θέση τους παραμένει αμετάβλητη.

### 17.3 Οπτικοποίηση Μοναδικής Περιόδου

Για να αποκτήσουμε μία πιο πλήρη εικόνα των κόμβων που ανήκουν σε ένα συγκεκριμένο δακτύλιο (π.χ. όταν εξετάζουμε δραστηριότητα σε περίοδο 30 ημερών, προτείνουμε μία οπτικοποίηση βασισμένη και πάλι σε ομόκεντρους κύκλους (παρόμοια με αυτή που περιγράφηκε παραπάνω, βλ. Εικόνα 17.4), η οποία περιέχει τους κόμβους που συσχετίζονται με τη συγκεκριμένη τιμή περιόδου. Στην περίπτωση αυτή, οι ακτίνες των ομόκεντρων κύκλων αντιστοιχούν στους διαφορετικούς βαθμούς εμπιστοσύνης. Ο εξωτερικός κύκλος αντιστοιχεί σε βαθμό εμπιστοσύνης 0.1, ενώ ο τελευταίος εσωτερικά σε βαθμό εμπιστοσύνης 1. Επομένως, κόμβοι των οποίων ο βαθμός εμπιστοσύνης τείνει στο 1 μετακινούνται προς το κέντρο του συστήματος. Όπως και παραπάνω, η οπτικοποίηση χωρίζεται σε κυκλικούς τομείς με βάση τον υπολογισμένο αριθμό ταιριασμάτων και ταυτόχρονα υποστηρίζει όλες τις λειτουργίες της οπτικοποίησης που περιγράφηκαν στην Ενότητα 17.2. Και πάλι, η τελική απεικόνιση υπολογίζεται από ένα *force-directed* αλγόριθμο, που αποτελεί παραλλαγή εκείνου που περιγράφηκε στην Ενότητα 17.2.



**Σχήμα 17.4:** Μία οπτικοποίηση ομόκεντρων κύκλων για περίοδο 30 ημερών. Οι ακτίνες κάθε κύκλου αντιστοιχούν σε διαφορετικούς βαθμούς εμπιστοσύνης. Κόμβοι με βαθμό εμπιστοσύνης 1 μετακινούνται προς το κέντρο του συστήματος





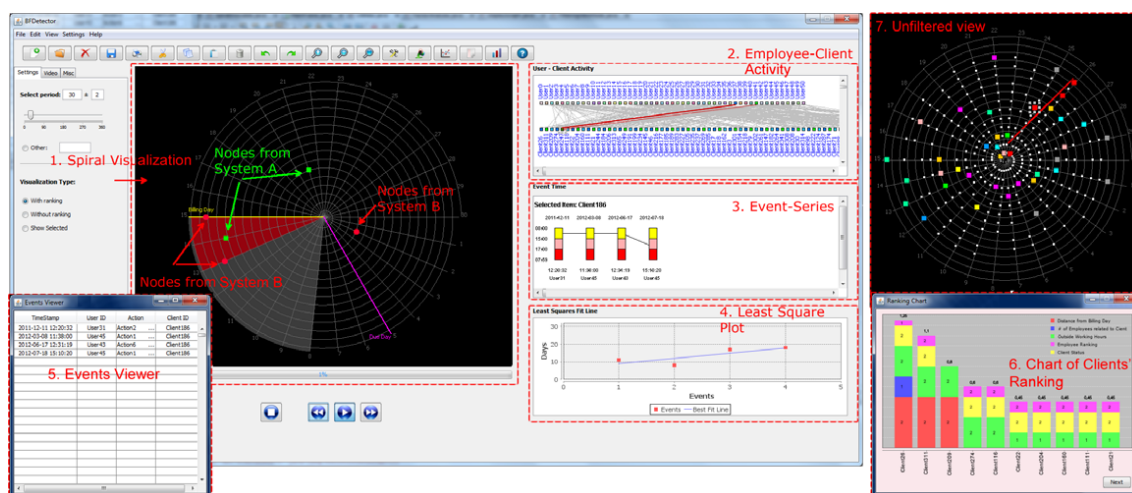
## 18 Ένα Σύστημα Ανίχνευσης Απάτης Βασισμένο σε Σπειροειδείς Οπτικοποιήσεις

Στο κεφάλαιο αυτό, εκμεταλλευόμαστε τα πλεονεκτήματα της οπτικοποίησης πληροφοριών και παρουσιάζουμε ένα σύστημα που έχει ως στόχο (α) την ανίχνευση απάτης σε συστήματα που εμπλέκουν ζεύγη οντοτήτων και (β) την ανίχνευση περιοδικής δραστηριότητας. Η κύρια μέθοδος οπτικοποίησης βασίζεται σε ένα σπειροειδές σύστημα στο οποίο τα γεγονότα απεικονίζονται με βάση τη χρονική στιγμή που συνέβησαν. “Υποπτα” θεωρούνται τα γεγονότα τα οποία εμφανίζονται κατά μήκος της ίδιας ακτίνας ή σε κοντινές ακτίνες του σπειροειδούς συστήματος. Για την παραγωγή της απεικόνισης, το σύστημα κατατάσσει τις δύο οντότητες με βάση κανόνες που ορίζουν οι εσωτερικοί ελεγκτές και παράγει ένα βίντεο της δραστηριότητας, στο οποίο γεγονότα με υψηλές ενδείξεις απάτης εμφανίζονται στα πρώτα καρέ του. Το σύστημα είναι, επίσης, εφοδιασμένο με αρκετές διαφορετικές οπτικοποιήσεις με στόχο να ανταποκρίνεται στις απαιτήσεις ενός συστήματος ανίχνευσης εσωτερικής απάτης.

### 18.1 Κύρια Χαρακτηριστικά του Συστήματος

Η Εικόνα 18.1 παρουσιάζει ένα στιγμιότυπο του συστήματός μας. Ο πυρήνας του συστήματος είναι μία σπειροειδής οπτικοποίηση στην οποία αναπαρίσταται κάθε γεγονός ανάλογα με τη χρονική στιγμή κατά την οποία συνέβη. Το κύριο πλεονέκτημα των σπειροειδών οπτικοποιήσεων είναι ότι πιθανά περιοδικά μοτίβα μπορούν να εντοπιστούν άμεσα, καθώς εμφανίζονται κατά μήκος της ίδιας ακτίνας ή σε κοντινές ακτίνες της σπειροειδούς οπτικοποίησης. Το σύστημα αποτελείται από διαφορετικές συντονισμένες οπτικοποιήσεις, καθεμία αφιερωμένη σε μία συγκεκριμένη πτυχή των δεδομένων. Το Πλαίσιο #2 της Εικόνας 18.1 παρουσιάζει τη συνολική δραστηριότητα των υπαλλήλων και των πελατών. Το Πλαίσιο #3 της Εικόνας 18.1 απεικονίζει την ώρα και την ημερομηνία κάθε γεγονότος που σχετίζεται με έναν πελάτη χρησιμοποιώντας διαφορετικά χρώματα για τον εντοπισμό προσβάσεων σε συγκεκριμένα χρονικά διαστήματα κατά τη διάρκεια μιας ημέρας. Η συγκεκριμένη οπτικοποίηση συνεισφέρει στην άμεση αναγνώριση γεγονότων που συμβαίνουν εκτός των ωρών εργασίας ή σε αργίες, τα οποία μπορεί να αποτελούν ενδείξεις απάτης. Το Πλαίσιο #4 της Ει-

κόντας 18.1 είναι ένα διάγραμμα ελαχίστων τετραγώνων, στο οποίο ο  $y$ -άξονας αντιστοιχεί στις ημέρες του μήνα και στοχεύει στην ανίχνευση περιοδικότητας. Στην περίπτωση που η ευθεία που προκύπτει είναι “σχεδόν” παράλληλη στον άξονα των  $x$  και τα σημεία είναι “κοινά” στην ευθεία, έπεται ότι υπάρχει ένα περιοδικό μοτίβο σχετικό με τη συγκεκριμένη μέρα του μήνα. Υπάρχει επίσης, ένα παράθυρο προβολής γεγονότων (βλ. το Πλαίσιο #5 της Εικόνας 18.1), το οποίο παρουσιάζει τα δεδομένα εισόδου υπό μορφή κειμένου και αλληλεπιδρά με την οπτικοποίηση.



Σχήμα 18.1: Ένα στιγμιότυπο του περιβάλλοντος του συστήματος.

Όπως αναφέρθηκε στην Ενότητα 16.5, το προτεινόμενο σύστημα έχει αναπτυχθεί σε συνεργασία με τους εσωτερικούς ελεγκτές ενός μεγάλου οργανισμού τηλεπικοινωνιών. Το βασικό πρόβλημα που αναφέρθηκε κατά την ανίχνευση εταιρικής απάτης είναι ο όγκος των δεδομένων, τα οποία παράγονται από περισσότερα του ενός εταιρικά συστήματα και συνήθως διερευνώνται γράφοντας και εκτελώντας κώδικα (scripting). Για το λόγο αυτό, προσπαθήσαμε να αναπτύξουμε ένα σύστημα, το οποίο άμεσα ανιχνεύει περιοδικά μοτίβα και ενσωματώνει αρκετά από τα πιο συνήθη πρότυπα απάτης που εξετάζονται από τους εσωτερικούς ελεγκτές. Επιπρόσθετα, καθώς η εταιρική απάτη είναι ένα “ευαίσθητο” θέμα, οι ελεγκτές τόνισαν την ανάγκη ανάπτυξης ενός εργαλείου, το οποίο θα μπορεί να επιβεβαιώνει ή να απορρίπτει γρήγορα υπόνοιες απάτης για κάποιο συγκεκριμένο υπάλληλο. Το προτεινόμενο σύστημα μπορεί να χρησιμοποιηθεί και για το σκοπό αυτό.

Το σύστημα είναι προσανατολισμένο προς το χρήστη (user-oriented) και η οπτικοποίηση μπορεί να προσαρμοστεί κατάλληλα έτσι ώστε να απεικονίζει τα πρότυπα που εξετάζονται από τον ελεγκτή. Η καινοτομία του εργαλείου είναι ότι συγκεντρώνει τη συνολική δραστηριότητα κάθε υπαλλήλου και πελάτη και κατατάσσει και τις δύο οντότητες με βάση κανόνες που ορίζονται από τους ελεγκτές. Με βάση αυτή την κατάταξη, το σύστημα παράγει

ένα αρχείο βίντεο του οποίου τα καρέ είναι ταξινομημένα έτσι ώστε εκείνα που περιέχουν υψηλές ενδείξεις απάτης να εμφανίζονται πρώτα.

Το σύστημα είναι, επίσης, εφοδιασμένο με επιπλέον οπτικοποιήσεις που παρέχουν συμπληρωματικές πληροφορίες σχετικά με τη δραστηριότητα των υπαλλήλων και των πελατών. Προκειμένου να ανταποκριθεί στις απαιτήσεις ενός εσωτερικού ελεγκτή, το σύστημα υποστηρίζει συμπληρωματικές λειτουργίες, όπως φιλτράρισμα, εξαγωγή αρχείων καταγραφής, αποθήκευση και μεταφόρτωση για μετέπειτα επεξεργασία των δεδομένων. Παρέχει, επίσης, προηγμένες γραφικές λειτουργίες, συμπεριλαμβανομένων αναδυόμενων μενού, δυνατότητες εκτύπωσης, επιλογής, μετακίνησης και αλλαγής μεγέθους των αντικειμένων, προσαρμοσμένου zoom και λειτουργιών fit-in window.

Η κύρια απαίτηση των ελεγκτών ήταν ο σχεδιασμός ενός συστήματος, το οποίο θα αποκαλύπτει επαναλαμβανόμενη δραστηριότητα μεταξύ ζευγών υπαλλήλων-πελατών. Για το λόγο αυτό, υιοθετήσαμε τη σπειροειδή απεικόνιση που επιτρέπει την οπτικοποίηση σειριακών δεδομένων. Ωστόσο, σε αντίθεση με τους Bertini et al. [BHL07] που χρησιμοποίησαν σπειροειδείς οπτικοποιήσεις για την απεικόνιση των γεγονότων τα οποία συμβαίνουν κατά τη διάρκεια μιας μέρας, τα σύνολα δεδομένων που επεξεργαζόμαστε καλύπτουν ένα πολύ μεγαλύτερο χρονικό διάστημα και θα πρέπει να οπτικοποιηθούν ταυτόχρονα. Το γεγονός αυτό, σε συνδυασμό με τον τεράστιο όγκο των δεδομένων έπρεπε να αντιμετωπιστεί αποτελεσματικά ώστε να μην επιβαρύνεται η οπτικοποίηση. Επιπρόσθετα, σε αντίθεση με τα σύνολα δεδομένων που χρησιμοποιούνται για την ανίχνευση χρηματοοικονομικού εγκλήματος, όπου υπάρχει συνήθως μία “υπόδειξη” σχετικά με “ύποπτες” περιπτώσεις ή συναλλαγές, στα σύνολα δεδομένων μας, τις περισσότερες περιπτώσεις, είναι εξαιρετικά απίθανο να υπάρχει πρότερη γνώση σχετικά με “ύποπτη” δραστηριότητα. Ως εκ τούτου, είναι δύσκολο και ίσως “επικίνδυνο” να προσπαθήσει κάποιος να φιλτράρει τα δεδομένα πριν την παραγωγή της οπτικοποίησης. Αυτό μας έδωσε το κίνητρο να σχεδιάσουμε τη διαδικασία κατάταξης των οντοτήτων, προκειμένου να διακρίνονται “ύποπτες” περιπτώσεις, οι οποίες θα πρέπει να διερευνηθούν περαιτέρω και να προσπαθήσουμε να την προσαρμόσουμε στις απαιτήσεις των ελεγκτών.

Σε σύγκριση με άλλα υπάρχοντα συστήματα για την ανίχνευση χρηματοοικονομικού εγκλήματος ή εταιρικής απάτης που αναφέρθηκαν στην Ενότητα 16.4, το σύστημα μας (i) παράγει ένα αρχείο βίντεο που περιέχει τη δραστηριότητα υπαλλήλων-πελατών σε ένα χρονικό διάστημα που καθορίζεται από τους ελεγκτές, (ii) διακρίνει “ύποπτα” γεγονότα με βάση τους παράγοντες κατάταξης που επιλέγονται από τους ελεγκτές και τα διανέμει στα καρέ του βίντεο έτσι ώστε να είναι να εντοπίζονται άμεσα και (iii) υποστηρίζει πολλαπλές συντομισμένες οπτικοποιήσεις που διευκολύνουν τη διερεύνηση και αποκάλυψη τυχόν περιοδικής δραστηριότητας.

## 18.2 Επισκόπηση της Διαδικασίας Ανίχνευσης

Στην ενότητα αυτή, παρουσιάζουμε μία επισκόπηση της διαδικασίας ανίχνευσης απάτης, όπως αυτή υποστηρίζεται από το σύστημά μας. Όπως ήδη αναφέρθηκε στην Ενότητα 16.5, το σύστημα λαμβάνει ως είσοδο δεδομένα που αποτελούνται από αρχεία καταγραφής ή σύνολα εγγραφών από βάσεις δεδομένων που παράγονται από εταιρικά συστήματα που εμπλέκουν ζεύγη οντοτήτων. Το σύστημα κατατάσσει τους υπαλλήλους και τους πελάτες με βάση κανόνες που ορίζει ο ελεγκτής και δημιουργεί ένα αρχείο βίντεο με τη συνολική δραστηριότητα των πελατών. Αρχικά, η κατάταξη υπολογίζεται στο σύνολο των δεδομένων, εκτός εάν ο ελεγκτής ορίσει ένα διαφορετικό χρονικό διάστημα. Στην οπτικοποίηση, όλα τα δυνητικά “ύποπτα” περιστατικά ανιχνεύονται και μπορεί να διερευνηθούν περαιτέρω από τον ελεγκτή, ο οποίος χρησιμοποιεί τις συνοδευτικές οπτικοποιήσεις του συστήματος. Πριν προχωρήσουμε στη λεπτομερή περιγραφή του συστήματος, εισάγουμε την απαραίτητη ορολογία για την περιγραφή της διαδικασίας κατάταξης. Έστω  $e = (t, u, c, a)$  ένα γεγονός που εμπλέκει τον υπάλληλο  $u$  και τον πελάτη  $c$  (βλ. Ενότητα 16.5). Για έναν πελάτη  $c$ , μία σειρά γεγονότων  $T_c = \{e_1^c, e_2^c, \dots\}$  είναι μία ακολουθία από γεγονότα  $e_i^c = (t_i, u_i, c, a_i)$  που σχετίζονται με τον πελάτη  $c$ . Σημειώνεται ότι μία τέτοια σειρά γεγονότων μπορεί να εμπλέκει περισσότερους από έναν υπαλλήλους.

## 18.3 Περιγραφή της Διαδικασίας Κατάταξης

Σε γενικές γραμμές, η κατάταξη ενός πελάτη βασίζεται σε διάφορους παράγοντες, όπως ο αριθμός των γεγονότων κοντά στην ημερομηνία χρέωσης του λογαριασμού (billing date), οι ενέργειες που πραγματοποιούνται από τους υπαλλήλους και ούτω καθεξής, οι οποίοι περιγράφονται λεπτομερώς στην Ενότητα 18.3.1. Η κατάταξη ενός υπαλλήλου βασίζεται στην κατάταξη των πελατών που σχετίζονται με το συγκεκριμένο υπάλληλο. Σημειώνεται ότι, εφόσον μία εταιρεία μπορεί να έχει χιλιάδες πελάτες, στη διαδικασία κατάταξης λαμβάνονται υπόψη μόνο αυτοί για τους οποίους υπάρχει ένα γεγονός που έχει παραχθεί από ένα μηχανισμό καταγραφής (δηλαδή όχι όλοι οι εγγεγραμμένοι πελάτες των συστημάτων της εταιρείας).

### 18.3.1 Συνάρτηση Κατάταξης Πελατών

Για την κατάταξη των πελατών, το σύστημα αναλύει τη σειρά γεγονότων που αντιστοιχεί σε κάθε πελάτη με βάση τους παράγοντες που ορίζονται από τον ελεγκτή. Στο σύστημα, έχουν ενσωματωθεί πολλά από τα επερωτήματα που χρησιμοποιούνται συνήθως από τους εσωτερικούς ελεγκτές όταν αναζητούν σχήματα εταιρικής απάτης.

Έστω  $N$  ο αριθμός των διακριτών παραγόντων που λαμβάνονται υπόψη κατά τον υπολογισμό της κατάταξης. Έστω, επίσης,  $a_f^c$  η “απόδοση” του πελάτη  $c$  στον παράγοντα  $f$ . Ανάλογα με τη σοβαρότητα (χαμηλή, μεσαία, υψηλή) της αντίστοιχης σειράς γεγονότων, η απόδοση  $a_f^c$  ισούται με μηδέν, ένα ή δύο, αντίστοιχα. Η κατάταξη  $R_c$  του υπαλλήλου  $c$  ορίζεται ως εξής:

$$R_c = \sum_{f=1}^N a_f^c \cdot w_f,$$

όπου η τιμή  $w_f$  είναι το βάρος που αντιστοιχεί στη σημαντικότητα του παράγοντα  $f$ .

Τα βάρη των παραγόντων  $w_f, f = 1, \dots, N$  καθορίζονται από τον ελεγκτή, ο οποίος επίσης ορίζει και μία διάταξη μεταξύ των παραγόντων που ταιριάζει καλύτερα σε αυτό που αναζητά. Για παράδειγμα, εάν ο ελεγκτής ενδιαφέρεται για γεγονότα που συμβαίνουν εκτός των ωρών εργασίας, ο αντίστοιχος παράγοντας θα πρέπει να είναι πρώτος στη διάταξη, και ως εκ τούτου το βάρος του θα είναι μεγαλύτερο των υπολοίπων που υποστηρίζονται από το σύστημα. Δοθείσας μιας διάταξης των παραγόντων, τα βάρη  $w_f, f = 1, \dots, N$  των παραγόντων υπολογίζονται με βάση τον τύπο που παρουσίασαν οι Stillwell et al. [SSE81], ως εξής:

$$w_f = \frac{N - r_f + 1}{\sum_{j=1}^N (N - r_j + 1)},$$

όπου με  $r_f$  συμβολίζεται η θέση του παράγοντα  $f$  στη διάταξη των παραγόντων.

Ακολούθως, περιγράφουμε τους παράγοντες που υποστηρίζονται από το σύστημα. Για καθένα από αυτούς, ορίζουμε τρεις κλάσεις πελατών ανάλογα με τη σοβαρότητα (χαμηλή, μεσαία, υψηλή) της σειράς γεγονότων  $T_c$  που αντιστοιχεί στον πελάτη  $c$ . Τότε, η απόδοση  $a_f^c$  του πελάτη  $c$  στον παράγοντα  $f$  ορίζεται από τον παρακάτω τύπο:

$$a_f^c = \begin{cases} 2, & T_c \in \text{Κλάση Υψηλής Σοβαρότητας για τον παράγοντα } f \\ 1, & T_c \in \text{Κλάση Μεσαίας Σοβαρότητας για τον παράγοντα } f \\ 0, & T_c \in \text{Κλάση Χαμηλής Σοβαρότητας για τον παράγοντα } f \end{cases}$$

### 18.3.1.1 Απόσταση από την Ημερομηνία Χρέωσης του Λογαριασμού

Η εμπειρία από την εξέταση των σχημάτων εταιρικής απάτης έχει δείξει ότι τα γεγονότα που σχετίζονται με το ίδιο ζεύγος υπαλλήλου-πελάτη και εμφανίζονται σε μηνιαία βάση και πριν από την ημερομηνία χρέωσης του λογαριασμού του πελάτη μπορεί να αποτελούν ενδείξεις απάτης. Δοθείσας μιας σειράς γεγονότων  $T_c$  που αντιστοιχεί στον πελάτη  $c$ , το σύστημα ανιχνεύει τα γεγονότα των οποίων οι χρονικές στιγμές είναι κοντά στην ημερομηνία

χρέωσης του λογαριασμού του πελάτη  $c$ . Με βάση τον αριθμό αυτών των γεγονότων, το σύστημα υπολογίζει τη σοβαρότητα της σειράς γεγονότων  $T_c$ . Σε περίπτωση που η χρονική στιγμή του συμβάντος έπεται της ημερομηνίας χρέωσης του λογαριασμού του συγκεκριμένου μήνα, θεωρούμε το γεγονός αυτό ως ένα περιστατικό που αφορά στη δραστηριότητα του επόμενου μήνα.

Έστω  $e$  ένα τυχαίο γεγονός και  $t_e$  η χρονική στιγμή που συνέβη. Έστω, επίσης  $t'_e$  η ημερομηνία χρέωσης του λογαριασμού που έπεται της χρονικής στιγμής  $t_e$ . Τότε, η απόσταση του γεγονότος από την ημερομηνία χρέωσης του λογαριασμού, η οποία συμβολίζεται με  $d_e$ , ορίζεται από τον αριθμό των ημερών που μεσολαβούν μεταξύ των χρονικών στιγμών  $t_e$  και  $t'_e$ .

Έστω  $D_c^0$  το σύνολο των γεγονότων, τα οποία συνέβησαν σε απόσταση τριών ημερών από την ημερομηνία χρέωσης του λογαριασμού, δηλαδή  $D_c^0 = \{e^c \in T_c : d_e \leq 3\}$  και  $|D_c^0|$  η πληθικότητα του. Όμοια, ορίζουμε το σύνολο των γεγονότων τα οποία συνέβησαν σε απόσταση μεγαλύτερη των τριών και μικρότερη των επτά ημερών από την ημερομηνία χρέωσης του λογαριασμού, δηλαδή  $D_c^1 = \{e^c \in T_c : 3 < d_e \leq 7\}$  και το σύνολο των γεγονότων με απόσταση μεγαλύτερη των επτά ημερών από την ημερομηνία χρέωσης του λογαριασμού, δηλαδή  $D_c^2 = \{e^c \in T_c : d_e > 7\}$ . Τα άκρα των παραπάνω διαστημάτων διερεύνησης μπορούν να τροποποιηθούν κατάλληλα, αν χρειάζεται, από τον ελεγκτή. Η αξιολόγηση της σημαντικότητας του συγκεκριμένου παράγοντα βασίζεται στην κατάταξη της σειράς γεγονότων  $T_c$  σε μία από τις ακόλουθες κλάσεις ανάλογα με τον αριθμό των γεγονότων που συνέβησαν κοντά στην ημερομηνία χρέωσης του λογαριασμού. Πιο συγκεκριμένα, η σειρά γεγονότων  $T_c$  που σχετίζεται με τον πελάτη  $c$ , ανήκει στην κλάση αυτή εάν:

*Κλάση Υψηλής Σοβαρότητας:* Η κλάση αυτή περιλαμβάνει σειρές γεγονότων με ισχυρές ενδείξεις απάτης για τις οποίες ισχύει ένα από τα ακόλουθα:

- $|D_c^0| \geq 2$ : Στην περίπτωση αυτή υπάρχουν τουλάχιστον δύο γεγονότα που σχετίζονται με τον πελάτη  $c$ , τα οποία συμβαίνουν πολύ κοντά στην ημερομηνία χρέωσης του λογαριασμού. Για την ελαχιστοποίηση των ψευδώς θετικών καταστάσεων (false-positives), περιπτώσεις στις οποίες υπάρχει μόνο ένα γεγονός κοντά στην ημερομηνία χρέωσης του λογαριασμού δεν κατατάσσονται από το σύστημα στην κλάση υψηλής σοβαρότητας, καθώς μπορεί να συνέβησαν τυχαία. Ωστόσο, κατατάσσονται στην κλάση μεσαίας σοβαρότητας, προκειμένου να διερευνηθούν περαιτέρω.
- $|D_c^1| \geq 3$ : Αυτή η περίπτωση συμπεριελήφθη στην κλάση υψηλής σοβαρότητας, καθώς η συγκεκριμένη σειρά γεγονότων περιέχει έναν "ασυνήθιστο" αριθμό γεγονότων που συνέβησαν σε απόσταση μιας εβδομάδας από την ημερομηνία χρέωσης του λογαριασμού. Και πάλι, το σύστημα προσπαθεί να ελαχιστοποιήσει τις ψευδώς θετικές καταστάσεις, λαμβάνοντας υπόψη μόνο γεγονότα που συνέβησαν τουλάχιστον τρεις

φορές σε απόσταση από την ημερομηνία χρέωσης του λογαριασμού που ανήκει στο διάστημα  $(3, 7]$ .

- $|D_c^0| + |D_c^1| \geq 2$ : Σε αυτή την περίπτωση, το σύστημα λαμβάνει υπόψη το συνολικό αριθμό των γεγονότων τα οποία συνέβησαν σε απόσταση μιας εβδομάδας από την ημερομηνία χρέωσης του λογαριασμού.

*Κλάση Μεσαίας Σοβαρότητας*: Στην κλάση αυτή, κατατάσσουμε σειρές γεγονότων, για τις οποίες ισχύει ένα από τα ακόλουθα:

- $|D_c^0| = 1$  ή  $|D_c^1| = 1$  ή  $|D_c^1| = 2$ : Οι περιπτώσεις αυτές εξαιρούνται από την κλάση υψηλής σοβαρότητας με σκοπό την ελαχιστοποίηση των ψευδώς θετικών καταστάσεων. Ωστόσο, θα πρέπει να διερευνηθούν, καθώς μπορεί να υποκρύπτουν κακόβουλη δραστηριότητα, η οποία μόλις ξεκίνησε.
- $|D_c^2| > thres$ , όπου η σταθερά  $thres$  είναι μία οριακή τιμή που ορίζεται από τον ελεγκτή (ως προεπιλογή, το σύστημα θεωρεί την τιμή 5). Στην περίπτωση αυτή υπάρχει μία επαναλαμβανόμενη δραστηριότητα που αφορά στον πελάτη  $c$ , η οποία θα πρέπει να διερευνηθεί περαιτέρω.

*Κλάση Χαμηλής Σοβαρότητας*: Όλες οι υπόλοιπες περιπτώσεις.

Όμοια ορίζεται ο παράγοντας που αφορά στην ημερομηνία λήξης του λογαριασμού ενός πελάτη. Και σε αυτή την περίπτωση ενδιαφερόμαστε για γεγονότα τα οποία συμβαίνουν πριν την ημερομηνία λήξης του λογαριασμού. Ωστόσο, είναι προτιμότερο να μη λαμβάνονται ταυτόχρονα υπόψη οι δύο παράγοντες “απόσταση από την ημερομηνία χρέωσης του λογαριασμού” και “απόσταση από την ημερομηνία λήξης του λογαριασμού”, καθώς μπορεί να προκύψουν επικαλύψεις μεταξύ των υπό εξέταση διαστημάτων, οι οποίες μπορεί να δημιουργήσουν ψευδώς θετικές καταστάσεις.

### 18.3.1.2 Περιοδικότητα Σειράς Γεγονότων

Δοθείσας μιας σειράς γεγονότων  $T_c$  που σχετίζεται με τον πελάτη  $c$ , υπολογίζουμε την περίοδο της δραστηριότητας με βάση τον αλγόριθμο που παρουσιάστηκε στην Ενότητα 17.1 (θεωρώντας ότι έχουμε ταξινομήσει τα γεγονότα με βάση την ημερομηνία που συνέβησαν). Δεδομένου ότι οι εσωτερικοί ελεγκτές ενδιαφέρονται για γεγονότα που συμβαίνουν σε μηνιαία βάση, το σύστημα αξιολογεί τη σοβαρότητα της σειράς γεγονότων ως εξής:

*Κλάση Υψηλής Σοβαρότητας*: Περιλαμβάνει σειρές γεγονότων με περιοδικότητα  $p$  τέτοια ώστε  $27 \leq p \leq 30$  ή 31 ημέρες.

*Κλάση Μεσαίας Σοβαρότητας*: Περιλαμβάνει σειρές γεγονότων με περιοδικότητα  $p$  τέτοια ώστε  $20 \leq p < 27$  ημέρες.

*Κλάση Χαμηλής Σοβαρότητας:* Όλες οι υπόλοιπες περιπτώσεις.

Όπως και προηγουμένως, ο ελεγκτής μπορεί να προσαρμόσει τις τιμές που καθορίζουν τα διαστήματα καθεμίας εκ των παραπάνω κλάσεων.

### 18.3.1.3 Γεγονότα Εκτός Ωρών Εργασίας

Δοθείσας μιας σειράς γεγονότων που σχετίζεται με ένα συγκεκριμένο πελάτη, το σύστημα προσπαθεί να ανιχνεύσει “ύποπτες” περιπτώσεις που συνέβησαν κατά τη διάρκεια της ημέρας. Γεγονότα υψηλής σοβαρότητας θεωρούνται εκείνα που συμβαίνουν εκτός των ωρών εργασίας των υπαλλήλων, τα Σαββατοκύριακα ή τις αργίες. Επιπρόσθετα, γεγονότα μεσαίας σοβαρότητας θεωρούνται εκείνα που συμβαίνουν στο τέλος της βάρδιας. Έχουμε υποθέσει ότι η περίπτωση αυτή αντιστοιχεί στις δύο τελευταίες ώρες της βάρδιας των υπαλλήλων. Ωστόσο, η τιμή αυτή μπορεί να ρυθμιστεί κατάλληλα από τον ελεγκτή. Το σύστημα δέχεται ως είσοδο τις ώρες εργασίας κάθε υπαλλήλου, λαμβάνοντας υπόψη τα Σαββατοκύριακα και τις αργίες. Η κατάταξη ενός πελάτη με βάση τις χρονικές στιγμές των σχετιζόμενων γεγονότων πραγματοποιείται ως εξής:

*Κλάση Υψηλής Σοβαρότητας:* Υπάρχει τουλάχιστον ένα γεγονός που συνέβη εκτός των ωρών εργασίας, το Σαββατοκύριακο ή σε ημέρα αργίας.

*Κλάση Μεσαίας Σοβαρότητας:* Υπάρχουν τουλάχιστον δύο γεγονότα που συνέβησαν στο τέλος της βάρδιας του υπαλλήλου. Όπως και προηγουμένως, το σύστημα απαιτεί την ύπαρξη τουλάχιστον δύο τέτοιων γεγονότων για να ελαχιστοποιήσει τις ψευδώς θετικές καταστάσεις.

*Κλάση Χαμηλής Σοβαρότητας:* Όλες οι υπόλοιπες περιπτώσεις.

Όπως και προηγουμένως, ο ελεγκτής μπορεί να προσαρμόσει τις τιμές που καθορίζουν τα διαστήματα καθεμίας εκ των παραπάνω κλάσεων.

### 18.3.1.4 Αριθμός των Υπαλλήλων που Σχετίζονται με έναν Πελάτη

Ο παράγοντας αυτός αφορά στον αριθμό των υπαλλήλων που σχετίζονται με ένα συγκεκριμένο πελάτη. Φυσιολογικά, αναμένεται ότι διακριτά γεγονότα σχετίζονται με διαφορετικούς υπαλλήλους. Λόγω του γεγονότος ότι συνήθως ένας τυχαία επιλεγμένος υπάλληλος εξυπηρετεί ένα αίτημα ενός πελάτη, η πολλαπλή εξυπηρέτηση αιτήσεων του ίδιου πελάτη από τον ίδιο υπάλληλο μπορεί να αποτελεί ένδειξη της απάτης. Η κατάταξη της σειράς γεγονότων που σχετίζεται με τον πελάτη  $c$  με βάση αυτόν τον παράγοντα, είναι η ακόλουθη:

*Κλάση Υψηλής Σοβαρότητας:* Ένας υπάλληλος εξυπηρετεί περισσότερα από το 50% των γεγονότων που σχετίζονται με τον πελάτη  $c$ .

*Κλάση Μεσαίας Σοβαρότητας:* Δύο ή περισσότεροι υπάλληλοι εξυπηρετούν περισσότερα από το 50% των γεγονότων που σχετίζονται με τον πελάτη  $c$ .



*Κλάση Χαμηλής Σοβαρότητας:* Όλες οι υπόλοιπες περιπτώσεις.

Ο ελεγκτής μπορεί και πάλι να προσαρμόσει τα ποσοστά που ορίζουν τις παραπάνω κλάσεις. Ωστόσο, υπάρχουν και περιπτώσεις στις οποίες, για παράδειγμα, ο πελάτης επικοινωνεί με την εταιρεία για ένα θέμα που αφορά στον λογαριασμό του και ζητάει να εξυπηρετηθεί πάντα από τον ίδιο υπάλληλο. Αυτό προφανώς, δεν αποτελεί απάτη και το σύστημα θα δημιουργούσε ένα ψευδώς-θετικό αποτέλεσμα στην περίπτωση που μόνο αυτός ο παράγοντας λαμβανόταν υπόψη κατά τη διαδικασία κατάταξης των πελατών. Επομένως, είναι προτιμότερο ο παράγοντας αυτός να εφαρμόζεται σε συνδυασμό με άλλους παράγοντες.

### 18.3.1.5 Ενέργεια

Κάθε εταιρεία έχει τους δικούς της κανόνες σχετικά με τους υπαλλήλους που χρησιμοποιούν τα εταιρικά συστήματα και υποστηρίζει διαφορετικά δικαιώματα σε διαφορετικούς υπαλλήλους. Ως εκ τούτου, υπάρχει μία λίστα με ενέργειες που απαγορεύεται να εκτελεστούν από το σύνολο ή από μη εξουσιοδοτημένους υπαλλήλους. Επιπρόσθετα, υπάρχουν ενέργειες οι οποίες είναι “ύποπτες”, ακόμα κι αν ένας υπάλληλος είναι εξουσιοδοτημένος να τις εκτελεί. Επιπλέον, υπάρχουν ενέργειες που μπορεί να συσχετίζονται μεταξύ τους (π.χ., open-close ενέργεια) και είναι ασυνήθιστο, αν μία από αυτές δεν εμφανίζεται στη σειρά γεγονότων. Επομένως, δεδομένου ότι υπάρχουν διάφοροι κανόνες με σκοπό την ανίχνευση της απάτης σε διαφορετικά εταιρικά συστήματα, ο ελεγκτής θα πρέπει να προσαρμόσει τους κανόνες που ορίζουν κάθε τάξη σοβαρότητας στην αντίστοιχη οθόνη του συστήματος. Μία επισκόπηση της ταξινόμησης με βάση αυτόν τον παράγοντα είναι η ακόλουθη:

*Κλάση Υψηλής Σοβαρότητας:* Ενέργειες που απαγορεύονται για μη εξουσιοδοτημένους υπαλλήλους.

*Κλάση Μεσαίας Σοβαρότητας:* Ενέργειες που θεωρούνται “ύποπτες”, όπως περιγράφηκε παραπάνω.

*Κλάση Χαμηλής Σοβαρότητας:* Όλες οι υπόλοιπες περιπτώσεις.

### 18.3.1.6 Κατάσταση Πελάτη

Κατά την κατάταξη ενός πελάτη, είναι σημαντικό να ληφθεί υπόψη το αντίστοιχο ιστορικό του. Αυτό συνεπάγεται ότι ένας πελάτης για τον οποίο υπήρχαν ενδείξεις απάτης κατά το παρελθόν θα πρέπει να κατατάσσεται υψηλότερα. Ο ελεγκτής είναι σε θέση: (i) να τοποθετήσει έναν πελάτη στη “μαύρη λίστα”, αν μία προηγούμενη έρευνα οδήγησε στην απόδειξη απάτης, (ii) να χαρακτηρίσει έναν πελάτη ως “ύποπτο απάτης”, αν μία προηγούμενη έρευνα είναι σε εξέλιξη ή σε εκκρεμότητα ή (iii) να χαρακτηρίσει έναν πελάτη ως “καθαρό”, εάν υπόνοιες απάτης είτε δεν υπάρχουν στο σύστημα ή δεν επιβεβαιώθηκαν από προηγούμενη έρευνα. Σύμφωνα με τον παραπάνω χαρακτηρισμό, ορίζουμε την ακόλουθη ταξινόμηση:

*Κλάση Υψηλής Σοβαρότητας:* Ο πελάτης είναι στη “μαύρη λίστα”.

*Κλάση Μεσαίας Σοβαρότητας:* Ο πελάτης είναι “ύποπτος απάτης”.

*Κλάση Χαμηλής Σοβαρότητας:* Ο πελάτης είναι “καθαρός”.

### 18.3.2 Κατάταξη Υπαλλήλων

Όπως αναφέρθηκε στην Ενότητα 18.2, το σύστημα κατατάσσει τους υπαλλήλους με βάση διάφορες πτυχές της σχέσης με τους σχετιζόμενους πελάτες τους. Έστω ένας υπάλληλος  $u$  και  $S_u$  το σύνολο των πελατών που εξυπηρετούνται από τον υπάλληλο  $u$ , δηλαδή  $S_u = \{c : \exists \text{event } e = (t, u, c, a)\}$ . Έστω, επίσης,  $R_u = \{R_c : c \in S_u\}$  το σύνολο που περιέχει την κατάταξη αυτών των πελατών. Το σύστημα αναθέτει στον υπάλληλο την τιμή που αντιστοιχεί στη μέγιστη τιμή κατάταξης του συνόλου  $R_u$ . Αυτό συνεπάγεται ότι εάν ένας υπάλληλος σχετίζεται με έναν “ύποπτο” πελάτη, τότε και αυτός θα θεωρείται επίσης “ύποπτος”. Μία εναλλακτική προσέγγιση θα ήταν να θεωρήσουμε τους πελάτες των οποίων η κατάταξη ξεπερνά μία οριακή τιμή που καθορίζεται από τον ελεγκτή.

## 18.4 Περιγραφή του Συστήματος

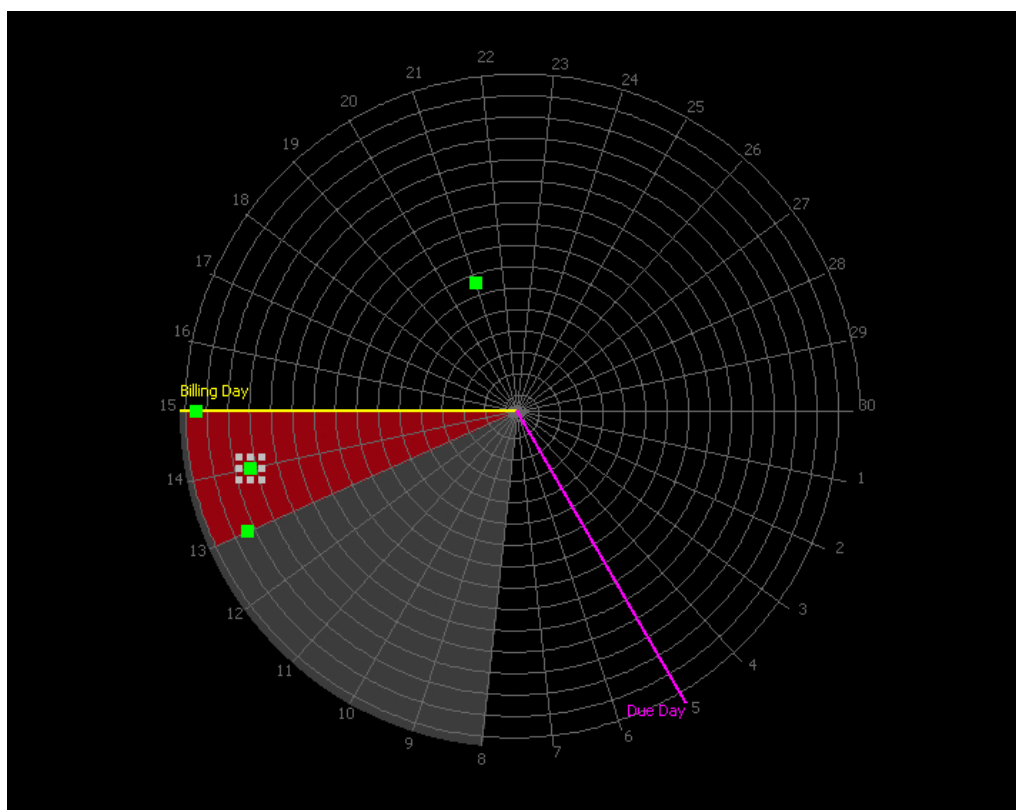
Στην ενότητα αυτή, περιγράφουμε αναλυτικά το σύστημα ανίχνευσης απάτης που υλοποιήσαμε. Το σύστημα αποτελείται από δύο βασικές λειτουργίες, την *off-line* και τη *semi-online* λειτουργία. Συνοπτικά, στην *off-line* λειτουργία, το σύστημα αναλύει στατικά δεδομένα για μία συγκεκριμένη χρονική περίοδο (π.χ., τα δεδομένα μιας εβδομάδας) και παρέχει τις αντίστοιχες οπτικοποιήσεις. Η *semi-online* λειτουργία μπορεί να χρησιμοποιηθεί σε καθημερινή βάση, προκειμένου να οπτικοποιήσει την καθημερινή δραστηριότητα των υπαλλήλων και των πελατών. Σε αμφότερες τις λειτουργίες, το σύστημα παρέχει διαδραστικές οπτικοποιήσεις που βοηθούν στην ανίχνευση “ύποπτης” δραστηριότητας. Οι οπτικοποιήσεις μεγάλου όγκου δεδομένων μπορεί να μην είναι χρήσιμες σε ορισμένες περιπτώσεις. Για να αντιμετωπιστεί το συγκεκριμένο πρόβλημα, ο ελεγκτής δύναται να προσδιορίσει ένα χρονικό παράθυρο και στη συνέχεια, το σύστημα οπτικοποιεί τα γεγονότα εκείνα των οποίων οι χρονικές στιγμές ανήκουν στο παράθυρο του επερωτήματος. Ωστόσο, η κατάταξη μπορεί να υπολογιστεί είτε με βάση ολόκληρο το σύνολο των δεδομένων (που μπορεί να περιλαμβάνει δεδομένα από ένα πολύ μεγαλύτερο χρονικό διάστημα) ή με βάση δεδομένα που συνέβησαν τη συγκεκριμένη χρονική περίοδο, ανάλογα με τις απαιτήσεις του ελεγκτή.

### 18.4.1 Off-line Λειτουργία

Όπως αναφέρθηκε παραπάνω, το σύστημα αποτελείται από πολλαπλές συντονισμένες οπτικοποιήσεις, καθεμία από τις οποίες οπτικοποιεί μία διαφορετική πτυχή των δεδομένων.

### 18.4.1.1 Η Σπειροειδής Οπτικοποίηση

Ένα στιγμιότυπο του συστήματος κατά την off-line λειτουργία απεικονίζεται στην Εικόνα 18.2. Στη σπειροειδή οπτικοποίηση, κάθε περιέλιξη (spiral branch) οπτικοποιεί μία περίοδο ενός μήνα, ενώ ο αριθμός των περιελίξεων σχετίζεται με την πρώτη και τελευταία χρονική στιγμή των δεδομένων εισόδου (εάν δεν έχει τροποποιηθεί διαφορετικά τον ελεγκτή), ξεκινώντας από τον πρώτο μήνα που συμπίπτει με την εσωτερική περιέλιξη της σπείρας. Κάθε περιέλιξη διαμερίζεται από έναν αριθμό ευθειών ανάλογα με την τιμή της περιόδου που εξετάζεται (δηλαδή 7 ημέρες, 15 ημέρες, 30 ημέρες, κλπ) και κάθε ευθεία αντιστοιχεί σε μία ημέρα ενός μήνα. Η προεπιλεγμένη τιμή είναι καθορισμένη στις 30 ημέρες, το οποίο συνεπάγεται ότι ο ελεγκτής αναζητά μηνιαία “ύποπτη” δραστηριότητα (βλ. Εικόνα 18.2).



**Σχήμα 18.2:** Η βασική οπτικοποίηση του συστήματος κατά την off-line λειτουργία.

Οι κόμβοι της σπειροειδούς οπτικοποίησης αναπαριστούν γεγονότα που σχετίζονται με έναν πελάτη και η θέση τους καθορίζεται με βάση την αντίστοιχη χρονική στιγμή που συνέβησαν. Κόμβοι με διαφορετικά χρώματα αναπαριστούν γεγονότα που σχετίζονται με διαφορετικούς πελάτες. Για να παραχθεί η σπειροειδής οπτικοποίηση, το σύστημα αγνοεί πολλαπλές εμφανίσεις γεγονότων που αντιστοιχούν στο ίδιο ζεύγος υπαλλήλου-πελάτη και

συμβαίνουν κατά την ίδια ημερομηνία. Σύμφωνα με τη σπειροειδή δομή, τα γεγονότα που σχετίζονται με τον ίδιο πελάτη και εμφανίζονται κατά μήκος μιας ακτίνας της σπείρας θεωρούνται “ύποπτα” και θα πρέπει να εξεταστούν περαιτέρω. Ωστόσο, η εξέταση σχημάτων εταιρικής απάτης αποκάλυψε ότι “ύποπτα” συμβάντα ενδέχεται να μην εμφανίζονται πάντα την ίδια ημερομηνία από μήνα σε μήνα και ως εκ τούτου, “ύποπτα” συμβάντα μπορεί να εμφανιστούν σε κοντινές ακτίνες της σπείρας. Αυτά θα πρέπει, επίσης, να διερευνηθούν.

Οι εργαζόμενοι και οι πελάτες κατατάσσονται σύμφωνα με τους κανόνες που ορίζουν οι ελεγκτές και τη διαδικασία κατάταξης που περιγράφηκε στην Ενότητα 18.3. Με βάση την κατάταξη αυτή, το σύστημα δημιουργεί ένα αρχείο βίντεο στο οποίο κάθε καρτέ απεικονίζει τη δραστηριότητα ενός πελάτη εντός του καθορισμένου χρονικού διαστήματος, δίνοντας προτεραιότητα σε αυτούς με την υψηλότερη κατάταξη. Η διάταξη των καρτέ εξασφαλίζει ότι οι πελάτες που θεωρούνται “ύποπτοι” δε θα παρακαμφθούν κατά την επεξεργασία (ακόμη και σε περιπτώσεις μεγάλων συνόλων δεδομένων θα διακρίνονται άμεσα).

Ο ελεγκτής δύναται να παύσει το βίντεο, προκειμένου να διερευνήσει περαιτέρω τη δραστηριότητα του πελάτη. Σε κάθε καρτέ του βίντεο απεικονίζονται τόσο η ημερομηνία χρέωσης, όσο και η ημερομηνία λήξης του λογαριασμού του αντίστοιχου πελάτη (βλ. Εικόνα 18.2). Οι περιοχές που είναι σκιασμένες με ανοιχτό-γκρι στην Εικόνα 18.2 αντιστοιχούν στο “επικίνδυνο” χρονικό διάστημα της μιας εβδομάδας πριν από την ημερομηνία χρέωσης του λογαριασμού. Η κόκκινη-σκιασμένη περιοχή της Εικόνας 18.2 φανερώνει ότι υπάρχουν γεγονότα από μήνα σε μήνα που διαφέρουν κατά λιγότερο από 3 ημέρες. Εάν είναι απαραίτητο, ο ελεγκτής δύναται να απεικονίσει τους κόμβους που αφορούν στον ίδιο υπαλληλο με το ίδιο χρώμα. Με αυτές τις λειτουργίες, ο ελεγκτής μπορεί άμεσα να εντοπίσει γεγονότα τα οποία συμβαίνουν κοντά στην ημερομηνία χρέωσης/λήξης του λογαριασμού και πιθανά περιοδικά μοτίβα που αφορούν σε συγκεκριμένο πελάτη.

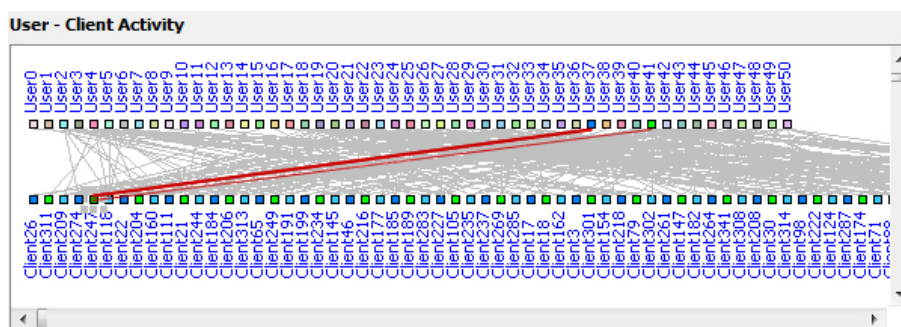
Το σύστημα υποστηρίζει, επίσης, τεχνικές φιλτραρίσματος. Οι παράγοντες κατάταξης που περιγράφονται στην Ενότητα 18.3 μπορούν, επίσης, να χρησιμοποιηθούν ως φίλτρα, ενώ τα αποτελέσματα μπορούν να εξαχθούν σε ξεχωριστά αρχεία καταγραφής. Για παράδειγμα, από ένα καρτέ του βίντεο, ο ελεγκτής μπορεί να επιλέξει μόνο τους κόμβους εκείνους που αναπαριστούν γεγονότα, τα οποία συνέβησαν εκτός των ωρών εργασίας των υπαλλήλων. Ο ελεγκτής μπορεί, επίσης, να εκτελέσει προσαρμοσμένα επερωτήματα στη βάση δεδομένων, κάτι το οποίο αποτελεί θεμελιώδη λειτουργία κατά τη διαδικασία ανίχνευσης απάτης. Προαιρετικά, το σύστημα μπορεί να αποθηκεύσει τις παραγόμενες οπτικοποιήσεις σε αρχεία για την περίπτωση που απαιτηθεί μετέπειτα επεξεργασία. Διατηρεί, επίσης, εγγραφές για τη δραστηριότητα των υπαλλήλων και των πελατών και την κατάταξή τους.

Όπως έχει ήδη αναφερθεί, η διάταξη των καρτέ πραγματοποιείται με βάση την κατάταξη των πελατών. Ωστόσο, οι κόμβοι της απεικόνισης μπορεί να διανέμονται στα καρτέ σύμφωνα

με μία εναλλακτική διάταξη που μπορεί να καθοριστεί από τον ελεγκτή και μπορεί να βασίζεται για παράδειγμα, σε πρότερη γνώση σχετικά με έναν πελάτη ή με μία λίστα πελατών που έχουν ήδη επισημανθεί από τον ελεγκτή από μία προηγούμενη έρευνα.

### 18.4.1.2 Συμπληρωματικές Οπτικοποιήσεις

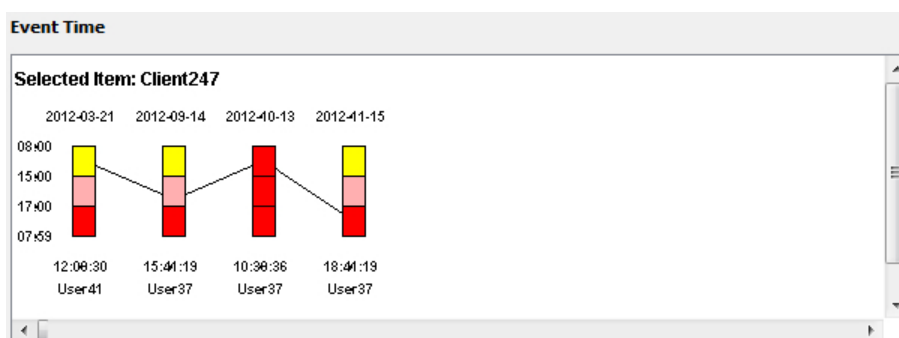
Θα προχωρήσουμε τώρα στην περιγραφή των συμπληρωματικών οπτικοποιήσεων του συστήματος. Η Εικόνα 18.3 παρουσιάζει μία οπτικοποίηση 2-επιπέδων που αναπαριστά τη συνολική δραστηριότητα των υπαλλήλων και των πελατών των δεδομένων εισόδου. Το πάνω επίπεδο της οπτικοποίησης αντιστοιχεί σε διακριτούς υπαλλήλους, ενώ το κάτω σε διακριτούς πελάτες. Η διάταξη των πελατών στο κάτω επίπεδο βασίζεται στην κατάταξή τους. Τα χρώματα των κόμβων ακολουθούν εκείνα που χρησιμοποιούνται για κάθε πελάτη στη σπειροειδή οπτικοποίηση. Η συγκεκριμένη οπτικοποίηση συμβάλλει στον άμεσο εντοπισμό ζευγών υπαλλήλων-πελατών που φαίνεται να εμπλέκονται σε πολλά γεγονότα και ταυτόχρονα παρέχει μία επισκόπηση της συνολικής δραστηριότητας των οντοτήτων. Επιπρόσθετα, αναδεικνύει υπαλλήλους που εμπλέκονται με περισσότερους του ενός “ύποπτους” πελάτες. Η οπτικοποίηση αλληλεπιδρά με εκείνη της σπείρας, έτσι ώστε όταν ένα ζεύγος υπαλλήλου-πελάτη επιλεγεί από τη σπειροειδή οπτικοποίηση, να επισημαίνεται και στην οπτικοποίηση των 2-επιπέδων και αντίστροφα. Στην περίπτωση που υπάρχουν περισσότερα του ενός γεγονότα που σχετίζονται με το ίδιο ζεύγος υπαλλήλου-πελάτη, το πάχος της αντίστοιχης ακμής αυξάνεται. Προαιρετικά, η οπτικοποίηση μπορεί να φιλτραρισθεί έτσι ώστε να απεικονίζεται μόνο ο πελάτης που εμφανίζεται στο βίντεο μαζί με τους σχετιζόμενους του υπαλλήλους.



**Σχήμα 18.3:** Μία οπτικοποίηση 2-επιπέδων που αναπαριστά τη συνολική δραστηριότητα των υπαλλήλων και των πελατών.

Στην οπτικοποίηση που παρουσιάζεται στην Εικόνα 18.4, η σειρά γεγονότων που σχετίζεται με ένα συγκεκριμένο πελάτη αναπαρίστανται ως ένα “μονοπάτι”. Κάθε κόμβος της οπτικοποίησης αντιστοιχεί στην ημέρα που συνέβη κάθε γεγονός της σειράς γεγονότων.

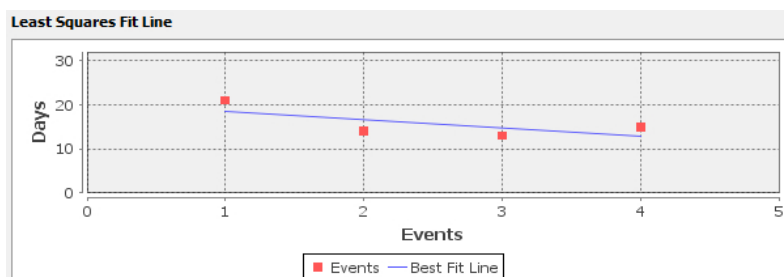
Κάθε τέτοιος κόμβος διαμερίζεται σε χρονικά διαστήματα που αντιστοιχούν στις ώρες της ημέρας. Το μεσαίο τμήμα (βλ. τις ροζ-σκιασμένες περιοχές των κόμβων της Εικόνας 18.4) αντιστοιχεί στο τέλος της βάρδιας των υπαλλήλων (δηλαδή τις δύο τελευταίες ώρες της βάρδιας τους). Το άνω μέρος (βλ. τις κίτρινες-σκιασμένες περιοχές των κόμβων της Εικόνας 18.4) αντιστοιχεί στη βάρδια κάθε υπαλλήλου έχοντας εξαιρέσει τις τελευταίες δύο ώρες, ενώ το κάτω μέρος (βλ. τις κόκκινες-σκιασμένες περιοχές των κόμβων της Εικόνας 18.4) αντιστοιχεί σε μη εργάσιμες ώρες της ημέρας. Τα άκρα κάθε ακμής ενώνονται με τα τμήματα που αντιστοιχούν στις χρονικές στιγμές που συνέβησαν τα γεγονότα. Σημειώνεται ότι για τη σπειροειδή οπτικοποίηση, το σύστημα αγνοεί πολλαπλά γεγονότα που αντιστοιχούν στο ίδιο ζεύγος υπαλλήλου-πελάτη. Για την οπτικοποίηση της Εικόνας 18.4, στην περίπτωση που υπάρχουν τέτοια πολλαπλά γεγονότα, θα δημιουργηθούν πολλαπλοί κόμβοι, οι οποίοι όμως θα οριοθετούνται από ένα ορθογώνιο, προκειμένου να διακρίνονται από τους υπόλοιπους. Το σύστημα λαμβάνει προαιρετικά ως είσοδο τις βάρδιες κάθε υπαλλήλου και κάνει τις κατάλληλες προσαρμογές στην οπτικοποίηση. Επιπρόσθετα, δεδομένου ότι τα Σαββατοκύριακα και οι αργίες μπορούν να λαμβάνονται υπόψη από το σύστημα, εφόσον εμφανιστούν τέτοιες περιπτώσεις, οι αντίστοιχοι κόμβοι θα είναι σκιασμένοι πλήρως με κόκκινο χρώμα (βλ. τον κόκκινο-σκιασμένο κόμβο της Εικόνας 18.4).



**Σχήμα 18.4:** Μία οπτικοποίηση που αναπαριστά τη σειρά γεγονότων που σχετίζεται με έναν πελάτη έτσι ώστε να διακρίνεται η ώρα της ημέρας κατά την οποία συνέβη ένα γεγονός.

Δοθείσας μιας σειράς γεγονότων που σχετίζεται με έναν πελάτη, το σύστημα παράγει ένα διάγραμμα στο οποίο κάθε σημείο  $(x, y)$  αναπαριστά την ημέρα  $y$  κατά την οποία συνέβη το γεγονός  $x$  (βλ. Εικόνα 18.5). Στη συνέχεια, χρησιμοποιώντας τη μέθοδο των ελαχίστων τετραγώνων [DS81], υπολογίζεται και σχεδιάζεται η ευθεία που προσεγγίζει περισσότερο αυτά τα σημεία. Περιπτώσεις στις οποίες η κλίση της ευθείας τείνει στο μηδέν (δηλαδή η ευθεία είναι παράλληλη στον άξονα των  $x$ ) και τα σημεία βρίσκονται “κοντά” στην ευθεία (το μοντέλο προσαρμόζεται καλά στο σύνολο των σημείων) φανερώνουν ότι τα περισσότερα από τα γεγονότα εμφανίζονται κοντά στην ίδια ημέρα του μήνα. Αυτό συνεπάγεται ότι

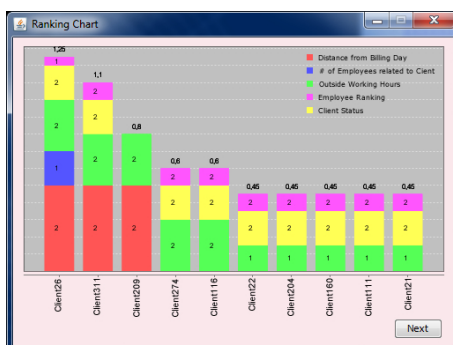
υπάρχει μία “ύποπτη” περιοδική δραστηριότητα (π.χ. κοντά στη 15η ημέρα κάθε μήνα). Μελετώντας το διάγραμμα των ελαχίστων τετραγώνων, θα πρέπει να λάβουμε υπόψη μας και περιπτώσεις στις οποίες η υπολογισμένη ευθεία είναι “σχεδόν” παράλληλη στον άξονα τον  $x$ , αλλά τα σημεία δεν απεικονίζονται κοντά στην ευθεία. Σε αυτή την περίπτωση, η εντύπωση της περιοδικότητας είναι πλασματική, καθώς η ευθεία δεν προσεγγίζει καλά το σύνολο των σημείων. Ωστόσο, αυτές οι περιπτώσεις μπορούν να αναγνωριστούν εύκολα είτε οπτικά είτε λαμβάνοντας υπόψη το σφάλμα του μοντέλου των ελαχίστων τετραγώνων.



**Σχήμα 18.5:** Ένα διάγραμμα ελαχίστων τετραγώνων που φανερώνει την ύπαρξη περιοδικών γεγονότων σε ένα χρονικό διάστημα.

Σημειώνεται ότι όλες οι παραπάνω οπτικοποιήσεις ενημερώνονται, καθώς εναλλάσσονται τα καρέ του βίντεο. Αυτό εξασφαλίζει ότι ο ελεγκτής έχει μία πλήρη εικόνα της δραστηριότητας κάθε πελάτη, χωρίς να απαιτούνται εναλλαγές μεταξύ παραθύρων. Ένα συμπληρωματικό παράθυρο, όπως εκείνο στο αριστερό τμήμα της Εικόνας 18.1, παρουσιάζει τις εγγραφές της βάσης δεδομένων που αφορούν στον πελάτη υπό τη μορφή κειμένου, ώστε ο ελεγκτής να μη χρειάζεται να ανατρέχει στα αρχεία της βάσης δεδομένων, προκειμένου να αποκτήσει εικόνα της αρχικής εισόδου. Το παράθυρο αυτό, επίσης, αλληλεπιδρά με τη σπειροειδή οπτικοποίηση, έτσι ώστε με την επιλογή ενός κόμβου από τη σείρα, να επιλέγεται και η αντίστοιχη γραμμή κειμένου στο παράθυρο που αντιστοιχεί στον κόμβο και αντίστροφα.

Ένα ακόμα χαρακτηριστικό του συστήματος είναι ότι παρέχει ένα διάγραμμα σωρευμένων ράβδων που αναδεικνύει την κατάταξη των πελατών, όπως απεικονίζεται στην Εικόνα 18.6. Κάθε ράβδος αντιστοιχεί σε έναν πελάτη και είναι χωρισμένη σε περιοχές που αναπαριστούν καθέναν από τους παράγοντες που λήφθηκαν υπόψη κατά τη διαδικασία κατάταξης, ενώ τα ύψη τους είναι ανάλογα με την απόδοση του πελάτη στο συγκεκριμένο παράγοντα. Επίσης, σε καθεμία από αυτές τις περιοχές καταγράφεται και η απόδοση του πελάτη στο συγκεκριμένο παράγοντα (σ.σ. 0, 1, ή 2). Το διάγραμμα της Εικόνας 18.6, δείχνει τους δέκα πρώτους πελάτες στην κατάταξη με βάση πέντε παράγοντες. Η πρώτη ράβδος δηλώνει ότι ο πελάτης “Client-26” κατατάσσεται στην κλάση υψηλής σοβαρότητας σε τρεις



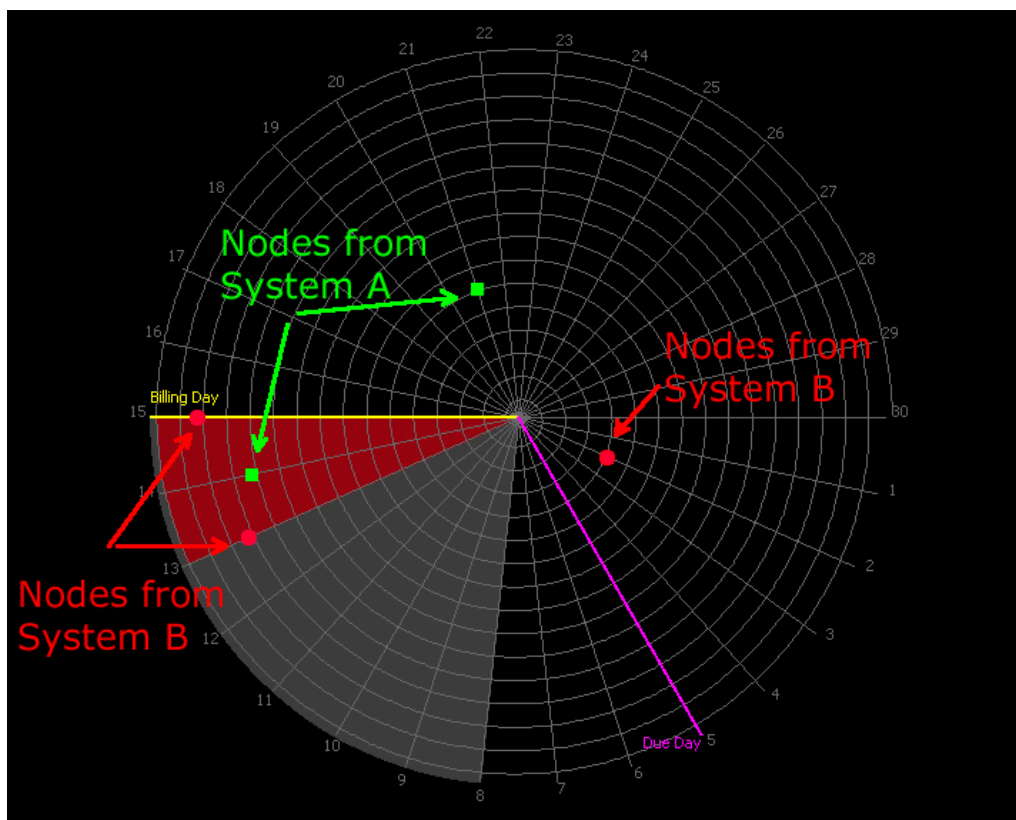
**Σχήμα 18.6:** Κατάταξη πελατών με βάση 5 παράγοντες. Εμφανίζονται οι δέκα πρώτοι πελάτες στην κατάταξη.

από τους πέντε παράγοντες, καθώς η απόδοσή του ήταν ίση με 2 για τους παράγοντες 1, 3 και 4.

Οι συστάσεις των εσωτερικών ελεγκτών πρόσθεσαν στο σύστημά μας μία ακόμα σημαντική λειτουργία. Μερικές φορές, προκειμένου να ανιχνευτούν ενδείξεις απάτης είναι αναγκαίος ο εντοπισμός “ύποπτων” γεγονότων, τα οποία συμβαίνουν σε περισσότερα του ενός εταιρικά συστήματα. Για το λόγο αυτό, ο ελεγκτής δύναται να επιλέξει έναν πελάτη και να οπτικοποιήσει τη δραστηριότητά του από διαφορετικά εταιρικά συστήματα. Στην περίπτωση αυτή, οι κόμβοι της οπτικοποίησης αναπαρίστανται με διαφορετικά σχήματα και χρώματα. Στην οπτικοποίηση της Εικόνας 18.7, οι πράσινοι-σκιασμένοι ορθογώνιοι κόμβοι αντιστοιχούν στο εταιρικό σύστημα Α και φαίνεται να μην έχουν καμία περιοδική δραστηριότητα. Ωστόσο, εάν απεικονιστούν ταυτόχρονα και οι κόκκινοι-σκιασμένοι κυκλικοί κόμβοι που προέρχονται από ένα διαφορετικό εταιρικό σύστημα Β γίνεται εμφανές ότι προκύπτει περιοδικότητα στη σειρά γεγονότων, η οποία μπορεί να υποκρύπτει απάτη.

Το σύστημα παρέχει, επίσης, μία αφιτράριστη προβολή των υπό επεξεργασία δεδομένων, όπως φαίνεται στην Εικόνα 18.8. Οι χρωματισμένοι κόμβοι αντιστοιχούν στους δέκα πρώτους υπαλλήλους στην κατάταξη. Οι κόμβοι που αντιστοιχούν στον ίδιο πελάτη αναπαρίστανται με το ίδιο χρώμα. Ωστόσο, δεδομένου ότι είναι δύσκολο να αναγνωριστεί “ύποπτη” συμπεριφορά σε αυτές τις οπτικοποιήσεις, το σύστημα προαιρετικά σχεδιάζει μία ευθεία που προσεγγίζει καλύτερα τη δραστηριότητα του επιλεγμένου πελάτη. Οι τεχνικές φίλτραρίσματα που αναφέρθηκαν παραπάνω, υποστηρίζονται και από αυτή την οπτικοποίηση. Αρχικά ωστόσο, η οπτικοποίηση φίλτράρεται έτσι ώστε οι πελάτες που σχετίζονται με ένα μόνο γεγονός να εξαιρούνται από την οπτικοποίηση.



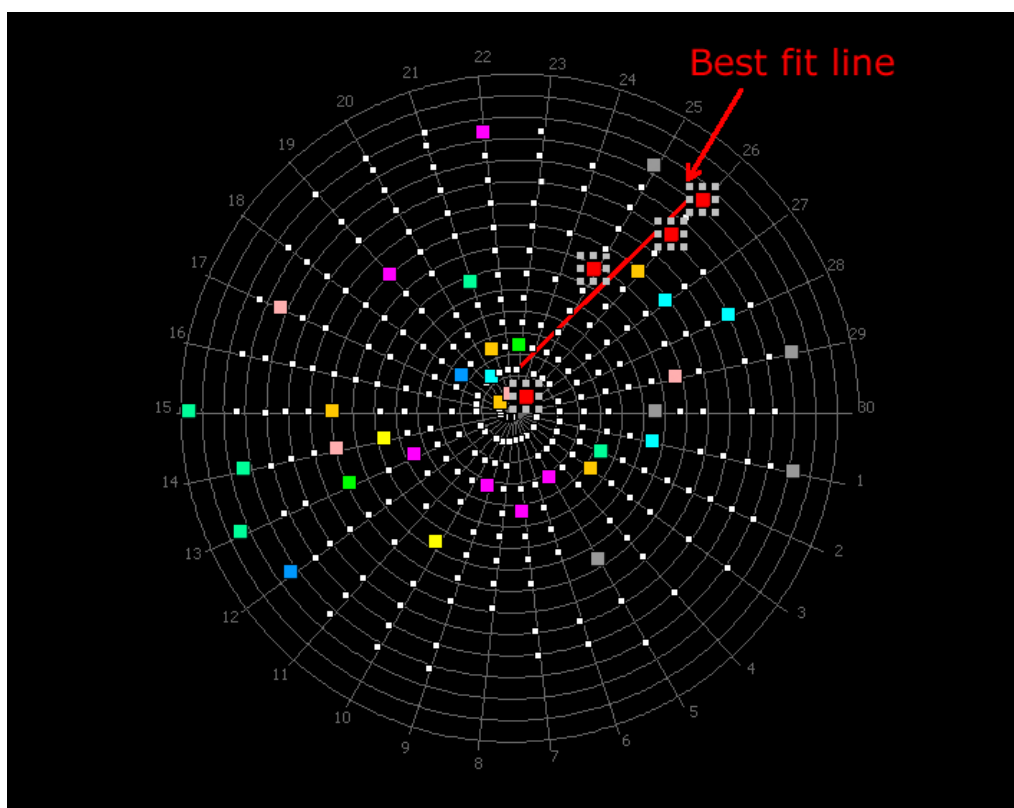


**Σχήμα 18.7:** Τα διαφορετικά χρώματα και σχήματα αντιστοιχούν σε γεγονότα που προέρχονται από διαφορετικά εταιρικά συστήματα. Ο συνδυασμός τους στην οπτικοποίηση αναδεικνύει την περιοδική δραστηριότητα, η οποία μπορεί να υποκρύπτει απάτη.

### 18.4.2 Semi-online Λειτουργία

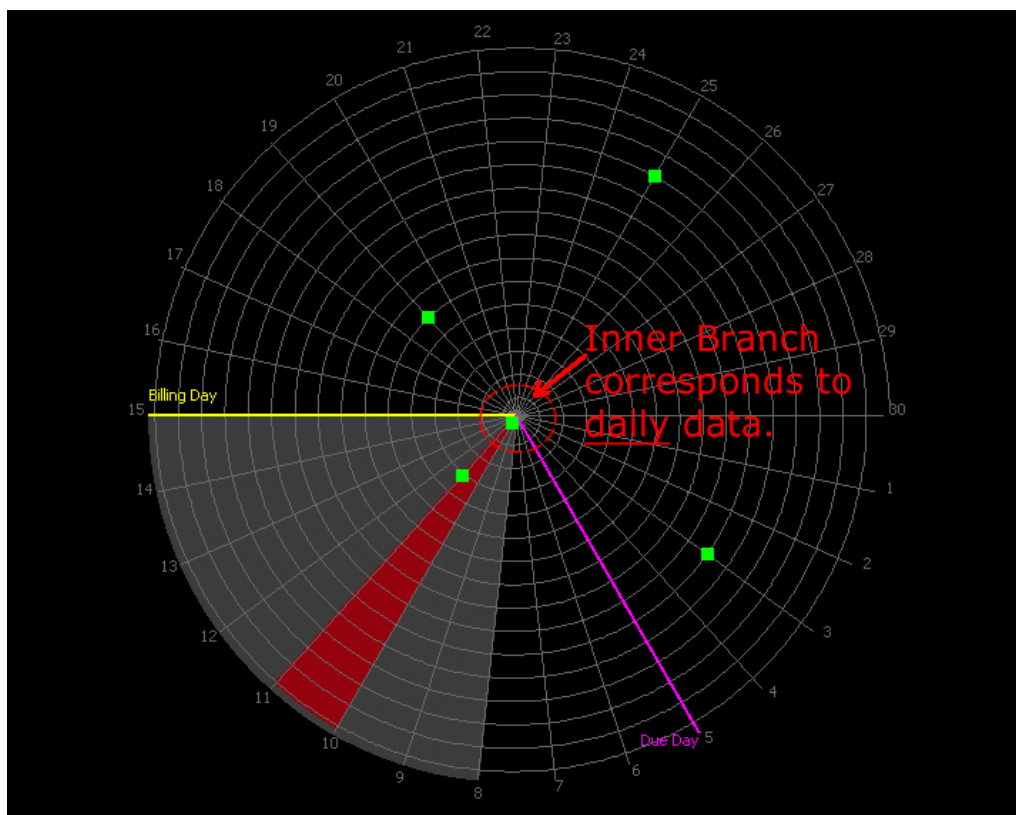
Κατά την ανάπτυξη του συστήματος, προσπαθήσαμε να ενσωματώσουμε λειτουργίες για την επεξεργασία δυναμικών δεδομένων. Ωστόσο, θα ήταν αδύνατο σε πραγματικές συνθήκες, ένας ελεγκτής να παρακολουθεί σε πραγματικό χρόνο τη δραστηριότητα των εργαζομένων. Οι συστάσεις των εσωτερικών ελεγκτών μας αποθαρρύνονται περισσότερο, καθώς η ανίχνευση απάτης πραγματοποιείται σε συστήματα διαφορετικά από εκείνα που δημιουργούν και συλλέγουν τα αρχεία καταγραφής και κατά συνέπεια, ο συγχρονισμός τους μπορεί να είναι εξαιρετικά δύσκολος. Για αυτούς τους λόγους, έχουμε αναπτύξει λειτουργίες για την επεξεργασία των δεδομένων σε ημερήσια βάση, αμέσως μόλις αυτά δημιουργούνται.

Η κύρια οπτικοποίηση της semi-online λειτουργίας είναι και πάλι, σπειροειδής και όλα τα χαρακτηριστικά της είναι παρόμοια με τα αντίστοιχα της off-line λειτουργίας (βλ. Εικόνα 18.9). Σε αντίθεση με την off-line λειτουργία, στη semi-online, η εσωτερική περιέλιξη της σπείρας αντιστοιχεί στα γεγονότα της τρέχουσας ημέρας. Οι υπόλοιπες περιελίξεις συ-



**Σχήμα 18.8:** Σπειροειδείς οπτικοποιήσεις όταν τα δεδομένα παρουσιάζονται χωρίς φιλτράρισμα.

μπίπτουν με τον πρώτο και τον τελευταίο μήνα των χρονικών στιγμών που εμφανίζονται στα δεδομένα εισόδου. Η σπείρα διαμερίζεται και πάλι από ευθείες που αναπαριστούν τις ημέρες ενός μήνα. Προκειμένου να παραχθεί η απεικόνιση, το σύστημα κατατάσσει εκ νέου τόσο τους πελάτες όσο και τους υπαλλήλους με βάση ολόκληρο το σύνολο των δεδομένων (δηλαδή συμπεριλαμβανομένων τόσο των προηγούμενων όσο και των νέων εγγραφών) και ορίζει τη διάταξή τους στα καρέ του βίντεο. Στη semi-online λειτουργία, κάθε γεγονός που σχετίζεται με ένα συγκεκριμένο πελάτη απεικονίζεται στην κατάλληλη θέση στην εσωτερική περιέλιξη της σπείρας με βάση τη χρονική στιγμή που συνέβη. Απεικονίζονται, επίσης, και όλα τα προηγούμενα γεγονότα της βάσης που σχετίζονται με το συγκεκριμένο πελάτη (στις κατάλληλες περιελίξεις της σπειροειδούς απεικόνισης). Και πάλι, ο ελεγκτής αναζητά γεγονότα που εμφανίζονται κατά μήκος μιας ακτίνας ή σε κοντινές ακτίνες της σπείρας. Η συγκεκριμένη οπτικοποίηση διευκολύνει την άμεση ανίχνευση ενός περιοδικού μοτίβου που μπορεί να έχει μόλις ξεκινήσει.



Σχήμα 18.9: Η σπειροειδής οπτικοποίηση όταν το σύστημα βρίσκεται στη semi-online λειτουργία.

## 18.5 Μελέτη Περίπτωσης

Σε αυτή την ενότητα, παρουσιάζουμε μία μελέτη περίπτωσης σε πραγματικά δεδομένα. Για λόγους εμπιστευτικότητας, τα στοιχεία που παρουσιάζονται σε αυτή τη μελέτη περίπτωσης έχουν υποστεί επεξεργασία ώστε να είναι ανώνυμα. Το σύνολο δεδομένων που έχουμε επεξεργαστεί, αντιστοιχεί σε ένα χρονικό διάστημα έξι μηνών και αποτελείται από περίπου 35.000 καταχωρήσεις που αφορούν 7200 διακριτούς πελάτες και 14 υπαλλήλους και προέρχονται από ένα εταιρικό σύστημα διαχείρισης απάτης. Σημειώνεται ότι τα δεδομένα είναι ευαίσθητα και δεν επιτρεπόταν να τα επεξεργαστούμε για τους σκοπούς της εργασίας μας. Η μελέτη περίπτωσης πραγματοποιήθηκε κατά την off-line λειτουργία του συστήματος και χωρίς να λαμβάνεται υπόψη τυχόν προηγούμενη κατάταξη των πελατών και των υπαλλήλων.

Δεδομένου ότι η μελέτη περίπτωσης διεξήχθη σε συνεργασία με τους εσωτερικούς ελεγκτές της εταιρείας, το κύριο ζήτημα που ετέθη ήταν ο εντοπισμός ζευγών υπάλληλων και πελατών που φαίνεται να συσχετίζονται με πάνω από δέκα γεγονότα κατά τη διάρκεια των

τελευταίων έξι μηνών. Προκειμένου να εξοικειωθούμε με το σύνολο των δεδομένων, κατάταξε αρχικά τους πελάτες με βάση τον αριθμό των γεγονότων στα οποία είχαν εμπλακεί τους. Το σύστημα εντόπισε 430 πελάτες των οποίων ο αριθμός των γεγονότων ήταν μεγαλύτερος από δέκα (περίπου 6% του αριθμού των πελατών στο αρχικό σύνολο δεδομένων), οι οποίοι διατάχθηκαν κατάλληλα στα καρέ του βίντεο. Ακόμα κι αν ο αριθμός αυτός είναι πολύ μικρότερος σε σχέση με το συνολικό αριθμό των πελατών στο σύνολο των δεδομένων, η διερεύνηση ήταν ακόμα αρκετά επίπονη για τον ελεγκτή (λόγω του αριθμού των πελατών που θα έπρεπε να εξεταστεί). Στο επόμενο βήμα, εκτελέσαμε ξανά τη διαδικασία κατάταξης (όπως περιγράφεται στην Ενότητα 18.3) με βάση τους εξής παράγοντες: (i) τον αριθμό των γεγονότων που σχετίζονται με έναν πελάτη, (ii) τον αριθμό των ενεργειών που είναι εξαιρετικά απίθανο να εκτελεστούν και αποτελούν ενδείξεις για πιθανή παράνομη δραστηριότητα και (iii) τον αριθμό των διακριτών υπαλλήλων που εξυπηρετούν κάθε πελάτη. Ακριβείς λεπτομέρειες σχετικά με τις τιμές που ορίζουν κάθε παράγοντα παραλείπονται για εμπιστευτικούς λόγους. Επιπρόσθετα, δεδομένου ότι δεν μας κοινοποιήθηκαν πληροφορίες σχετικά με την ημερομηνία χρέωσης του λογαριασμού του κάθε πελάτη και τις βάρδιες των εργαζομένων αγνοήσαμε τους συγκεκριμένους παράγοντες.

Η διαδικασία κατάταξης αναγνώρισε 52 από τους 430 πελάτες (περίπου 0.7% του αριθμού των πελατών του αρχικού συνόλου δεδομένων) που βρίσκονταν στην κλάση υψηλής σοβαρότητας με βάση τους παραπάνω παράγοντες και τους παρουσίασε στα πρώτα καρέ του βίντεο. Στα επόμενα καρέ, το σύστημα παρουσίασε 62 πελάτες (περίπου 0.9% του αριθμού των πελατών του αρχικού συνόλου δεδομένων) που βρίσκονταν στην κλάση μεσαίας σοβαρότητας στους παραπάνω παράγοντες. Τα αποτελέσματα διερευνήθηκαν περαιτέρω από τους ελεγκτές, οι οποίοι προσπάθησαν να ανιχνεύσουν περιοδικά μοτίβα (σε ημερήσια ή μηνιαία βάση) στα συγκεκριμένα καρέ. Οι ελεγκτές πρότειναν, επίσης, την εφαρμογή του παράγοντα που αφορά στην περιοδικότητα που περιγράφεται στην Ενότητα 18.3.1.2, προκειμένου να εντοπιστούν γεγονότα που εμφανίζονται εντός χρονικού διαστήματος (i) 28 ημερών και (ii) 5 ημερών. Για την πρώτη περίπτωση, το σύστημα εντόπισε 3 από τους 52 πελάτες, οι οποίοι βρέθηκαν στην κλάση υψηλής σοβαρότητας και 6 από τους 62 πελάτες, οι οποίοι βρέθηκαν στην κλάση μεσαίας σοβαρότητας. Για τη δεύτερη περίπτωση, 11 από τους 52 βρέθηκαν στην κλάση υψηλής σοβαρότητας και 12 από 62 στην κλάση μεσαίας σοβαρότητας.

Για το τελικό στάδιο της διερεύνησης, οι ελεγκτές χρησιμοποίησαν τις συμπληρωματικές οπτικοποιήσεις, την εμπειρία τους, καθώς και συμπληρωματικά στοιχεία τα οποία δεν μας κοινοποιήθηκαν, προκειμένου να αξιολογήσουν τη σοβαρότητα αυτών των γεγονότων. Θα πρέπει να τονιστεί ότι κατά τη μη-αυτόματη διερεύνηση των δεδομένων από τους ελεγκτές (δηλαδή κατά τη μη-αυτοματοποιημένη επεξεργασία των αρχείων καταγραφής) δεν

είχε εντοπίσει η παραπάνω επαναλαμβανόμενη δραστηριότητα. Τα αποτελέσματα ελήφθησαν υπόψη από τους ελεγκτές για περαιτέρω εσωτερική έρευνα. Ευτυχώς, δεν επιβεβαιώθηκε η ύπαρξη απάτης σε καμία από τις παραπάνω περιπτώσεις.



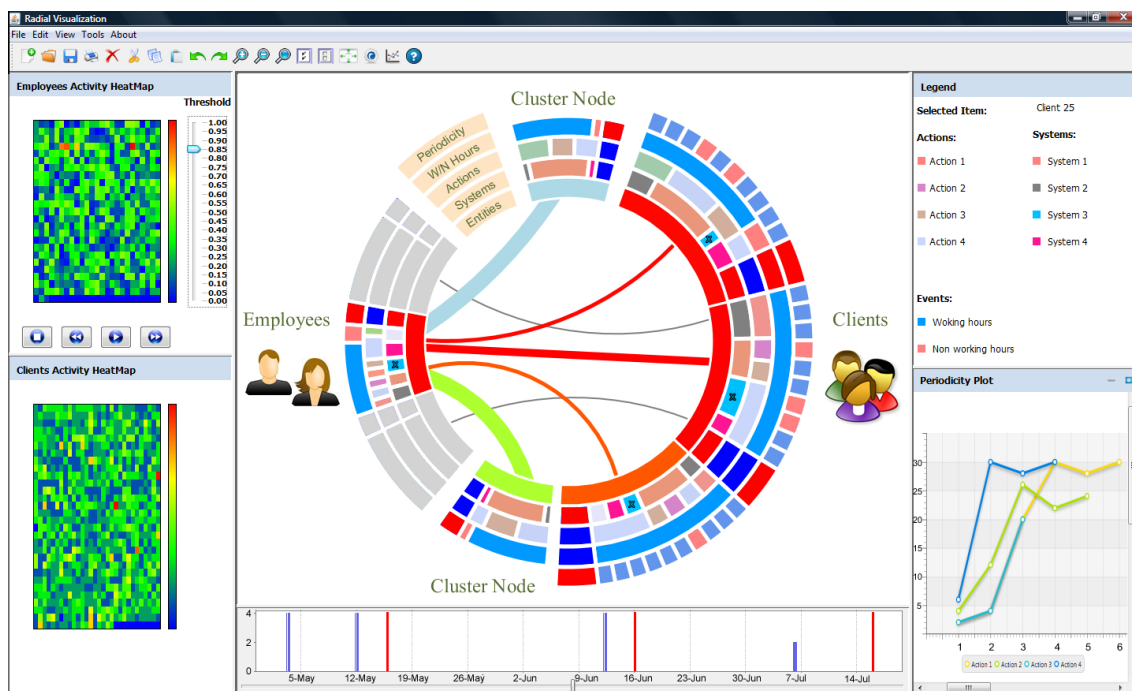
## 19 Ένα Σύστημα Οπτικοποίησης με Κυκλικές Απεικονίσεις και Θερμικούς Χάρτες

Σε αυτό το κεφάλαιο, εξελίσσουμε περαιτέρω το σύστημα που παρουσιάστηκε στο Κεφάλαιο 18 ώστε η οπτικοποίηση να μην περιορίζεται μόνο στον εντοπισμό περιοδικών γεγονότων που εμπλέκουν ζεύγη οντοτήτων, αλλά να παρέχει μία ολοκληρωμένη εικόνα όλων των υπό εξέταση προτύπων απάτης, καθώς και των αποτελεσμάτων της εξέτασης σε κάθε πρότυπο. Το σύστημα υποστηρίζει, επίσης, την παρακολούθηση της δραστηριότητας των υπαλλήλων στο δίκτυο της εταιρείας αποσκοπώντας στην ανίχνευση ασυνήθιστης δραστηριότητας, όπως μη εξουσιοδοτημένη πρόσβαση σε εταιρικά συστήματα και λογαριασμούς υπαλλήλων ή πελατών.

### 19.1 Κύρια Χαρακτηριστικά του Συστήματος

Το σύστημα παράγει μία πολυεπίπεδη κυκλική οπτικοποίηση (βλ. Εικόνα 19.1) που αναπαριστά τη δραστηριότητα των υπαλλήλων και των πελατών, μαζί με άλλες σημαντικές πληροφορίες που επιτρέπουν τον εντοπισμό πιθανών σχημάτων απάτης. Προκειμένου να παραχθεί η οπτικοποίηση, το σύστημα υπολογίζει την ομοιότητα της δραστηριότητας των υπαλλήλων με πρότυπα ανίχνευσης απάτης (που προτείνονται από τους ελεγκτές με βάση την εμπειρία τους και τη στρατηγική της επιχείρησης για τη μείωση του κινδύνου εσωτερικής απάτης) και υπολογίζει κατάλληλους θερμικούς χάρτες (heat-maps) που παράγονται και ενσωματώνονται στο σύστημα. Οι παραγόμενες οπτικοποιήσεις παρουσιάζονται με τη μορφή κινούμενων εικόνων (animation). Το σύστημα υποστηρίζει συμπληρωματικές λειτουργίες, όπως παράθυρο προβολής γεγονότων (database log viewer), φιλτράρισμα, εξαγωγή αρχείων καταγραφής, αποθήκευση και μεταφόρτωση για μετέπειτα επεξεργασία των δεδομένων, διαγράμματα και γραφήματα (βλ. Εικόνα 19.2). Παρέχει, επίσης, προηγμένες γραφικές λειτουργίες, συμπεριλαμβανομένων αναδυόμενων μενού, δυνατότητες εκτύπωσης, επιλογής, μετακίνησης και αλλαγής μεγέθους των αντικειμένων, προσαρμοσμένο zoom και λειτουργίες fit-in window.

Το σύστημα επεκτείνει εκείνο που παρουσιάστηκε στο Κεφάλαιο 18, το οποίο ανιχνεύει περιοδικά μοτίβα που μπορεί να υποκρύπτουν εταιρική απάτη, με αρκετές τρόπους: (i) Η



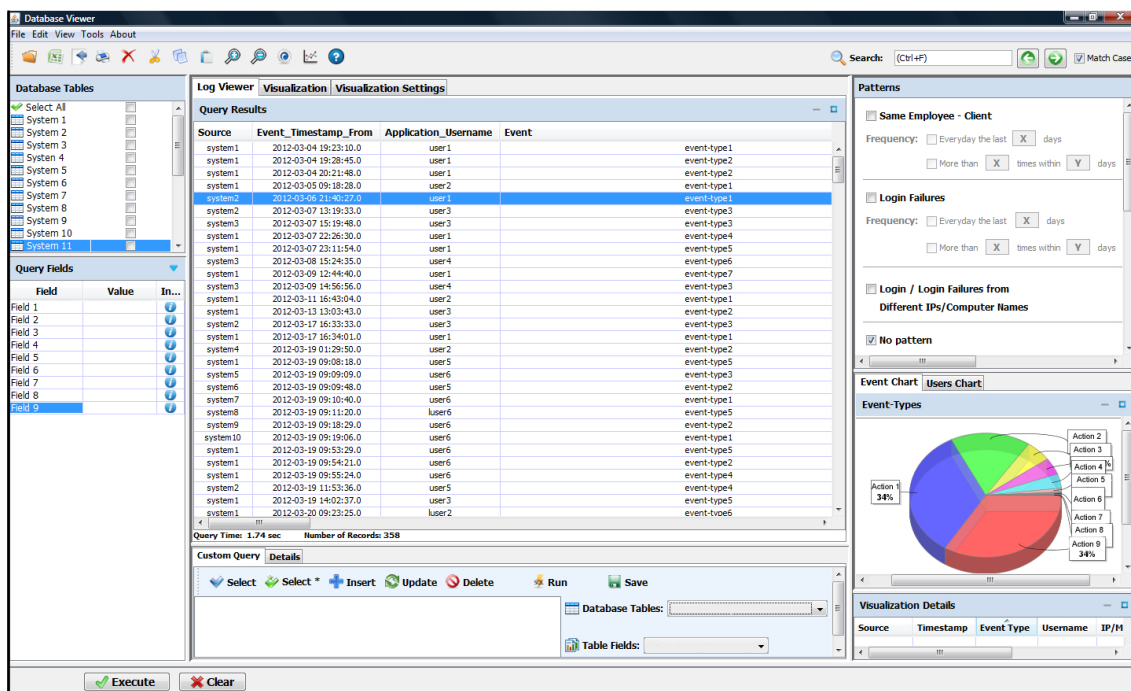
Σχήμα 19.1: Ένα στιγμιότυπο του περιβάλλοντος του συστήματος.

τρέχουσα οπτικοποίηση παρέχει μία ολοκληρωμένη εικόνα όλων των υπό εξέταση προτύπων και των αποτελεσμάτων της εξέτασης σε κάθε πρότυπο (στο Κεφάλαιο 18, η διαδικασία ανίχνευσης ήταν ένα “μαύρο κουτί” που καθόριζε τη σειρά παρουσίασης των πελατών σε ένα βίντεο που αναπαριστούσε τη δραστηριότητά τους, στο οποίο “ύποπτοι” πελάτες εμφανίζονταν στα πρώτα καρέ, ενώ αποσπασματικά αποτελέσματα της διαδικασίας ανίχνευσης παρουσιάζονταν σε διαφορετικά διαγράμματα και γραφήματα, γεγονός το οποίο επιβράδυνε τη διαδικασία διερεύνησης), (ii) ο μηχανισμός ανίχνευσης βασίζεται σε ένα δέντρο απόφασης, ενώ ταυτόχρονα έχουμε ενσωματώσει τους περισσότερους από τους παράγοντες που παρουσιάστηκαν στο Κεφάλαιο 18, (iii) για την ανίχνευση περιοδικότητας, εφαρμόζουμε μία παραλλαγή του αλγορίθμου της Μέγιστης Κοινής Υπακολουθίας (Longest Common Subsequence) [VHGK03] που αντιμετωπίζει δεδομένα με “θόρυβο”, (iv) ασυνήθιστη συμπεριφορά υπαλλήλων (μη εξουσιοδοτημένη πρόσβαση σε υπολογιστές, εταιρικά συστήματα, κλπ) μπορεί, επίσης, να ανιχνευθεί με τη χρήση ενός διαγράμματος παράλληλων συντεταγμένων και (v) το σύστημα παρέχει ένα παράθυρο προβολής γεγονότων προκειμένου να διευκολυνθεί η διαδικασία διερεύνησης.

Η κύρια οπτικοποίηση του συστήματος είναι παρόμοια με εκείνη του Synerscope [Syn11], ενός εμπορικού εργαλείου που αναπαριστά τις σχέσεις χρέωσης (billing links) και τις συσχετίσεις μεταξύ μιας εταιρείας και άλλων συναφών φορέων με σκοπό την ανίχνευση απάτης.



Ωστόσο, το σύστημα μας επικεντρώνει αποκλειστικά στην ανίχνευση εταιρικής απάτης με βάση πρότυπα που προτείνονται από τους ελεγκτές και για το λόγο αυτό, είναι εφοδιασμένο με ένα μηχανισμό ανίχνευσης που προ-επεξεργάζεται τα δεδομένα, ενώ η οπτικοποίηση προσαρμόζεται, επίσης, σε αυτά τα πρότυπα. Και πάλι, χρησιμοποιούμε κινούμενες εικόνες για την ανίχνευση σχημάτων απάτης για να αποφευχθεί η επιβάρυνση της οπτικοποίησης.

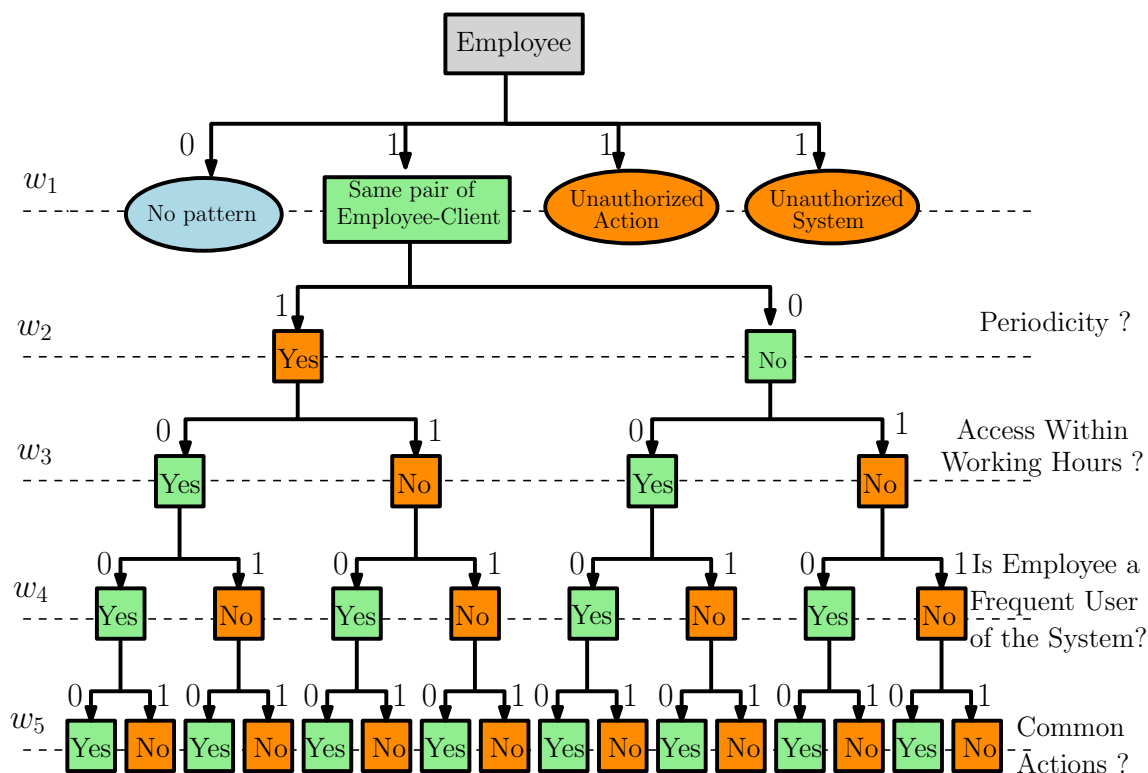


Σχήμα 19.2: Η αρχική οθόνη του συστήματος. Προσαρμοσμένα ερωτήματα μπορούν να εκτελεστούν και τα αποτελέσματα παρουσιάζονται στο παράθυρο προβολής γεγονότων. Τα δεδομένα είναι ανώνυμα για λόγους ασφάλειας και ιδιωτικότητας.

## 19.2 Επισκόπηση της Διαδικασίας Ανίχνευσης

Το σύστημα δέχεται ως είσοδο αρχεία καταγραφής από διαφορετικά εταιρικά συστήματα που εμπλέκουν υπαλλήλους και πελάτες. Έστω ότι  $e = (t, u, c, a)$  είναι ένα γεγονός που εμπλέκει τον υπάλληλο  $u$  και τον πελάτη  $c$  (βλ. Ενότητα 16.5). Όπως έχει ήδη αναφερθεί, λέμε ότι ο πελάτης  $c$  σχετίζεται με το γεγονός  $e$  και σχετίζεται, επίσης, με τον υπάλληλο  $u$ . Για ένα ζεύγος υπαλλήλου-πελάτη  $(u, c)$ , μία σειρά γεγονότων  $T_{u,c} = \{e_{(u,c)}^1, e_{(u,c)}^2, \dots\}$  είναι μία ακολουθία γεγονότων  $e_{(u,c)}^i = (t_i, u, c, a_i)$ , τα οποία σχετίζονται με τον πελάτη  $c$  και τον υπάλληλο  $u$ .

Η οπτικοποίηση μπορεί να παραχθεί είτε με βάση ολόκληρο το σύνολο δεδομένων της



**Σχήμα 19.3:** Το δέντρο απόφασης με βάση το οποίο στη σειρά γεγονότων των υπαλλήλων ανατίθεται μία τιμή σοβαρότητας.

βάσης δεδομένων ή πάνω σε επερωτήματα που εκτελούνται από τον ελεγκτή στην αρχική οθόνη του συστήματος (βλ. Εικόνα 19.2). Όπως αναφέρθηκε παραπάνω, τα δεδομένα πρέπει να προ-επεξεργαστούν πριν την παραγωγή της οπτικοποίησης, έτσι ώστε υπάλληλοι με ισχυρές ενδείξεις απάτης να διακρίνονται στην οπτικοποίηση. Για το λόγο αυτό, για δοθέν υπάλληλο  $u$ , η σειρά γεγονότων που σχετίζεται με τον υπάλληλο  $u$  θα πρέπει να αξιολογηθεί με βάση πιθανά πρότυπα απάτης και μία τιμή που αναδεικνύει τη σοβαρότητα των σχετιζόμενων γεγονότων (εντός του διαστήματος  $[0, 1]$ ) θα ανατεθεί πρώτα στη σειρά γεγονότων και στη συνέχεια στον υπάλληλο. Αν εντοπιστούν περισσότερα του ενός πρότυπα απάτης, στον υπάλληλο  $u$  ανατίθεται η μέγιστη τιμή σοβαρότητας των ήδη υπολογισμένων σειρών γεγονότων που σχετίζονται με τον υπάλληλο  $u$ . Η αξιολόγηση πραγματοποιείται με βάση ένα δέντρο απόφασης που κατασκευάζεται από τα ακόλουθα πρότυπα που προτείνονται από τους ελεγκτές (βλ. Εικόνα 19.3): (i) Υπάρχουν περισσότερα από  $X$  γεγονότα που σχετίζονται με έναν υπάλληλο  $u$  και ένα πελάτη  $c$  σε χρονικό διάστημα  $Y$  ημερών/μηνών, όπου οι τιμές  $X, Y$  καθορίζονται από τον ελεγκτή (βλ. τον πράσινο-σκιασμένο ορθογώνιο κόμβο του πρώτου επιπέδου κάτω από τη ρίζα του δέντρου απόφασης στην Εικόνα 19.3), (ii) ο

υπάλληλος έχει πραγματοποιήσει μη εξουσιοδοτημένες ενέργειες (με βάση έναν κατάλογο ενεργειών που παρέχεται από τους ελεγκτές) και (iii) ο υπάλληλος λειτουργεί σε συστήματα στα οποία δεν είναι εξουσιοδοτημένος χρήστης.

Στην περίπτωση που εντοπιστούν τα πρότυπα των περιπτώσεων (ii) ή (iii), στη σειρά γεγονότων του υπαλλήλου ανατίθεται η μέγιστη τιμή (δηλαδή η τιμή 1), έτσι ώστε ο υπάλληλος να διακριθεί σίγουρα στην οπτικοποίηση. Ωστόσο, στην περίπτωση κατά την οποία εντοπιστεί το πρότυπο της περίπτωσης (i), η διερεύνηση θα πρέπει να προχωρήσει περαιτέρω. Τα πρότυπα που λαμβάνονται υπόψη σε αυτή την περίπτωση περιλαμβάνουν τα ακόλουθα:

- Περιοδικότητα σειράς γεγονότων:** Ένα κοινό μοτίβο κατά την εξέταση τέτοιων σχημάτων απάτης είναι η εμφάνιση γεγονότων ανά τακτά χρονικά διαστήματα. Για παράδειγμα, ένας υπάλληλος τροποποιεί σκόπιμα το λογαριασμό ενός πελάτη εντός του κύκλου χρέωσης του λογαριασμού και πιο συγκεκριμένα, πριν από την ημερομηνία χρέωσης του. Υποθέτοντας ότι η σειρά γεγονότων  $T_{u,c}$ , η οποία σχετίζεται με υπάλληλο  $u$  και τον πελάτη  $c$  είναι ταξινομημένη με βάση τις χρονικές στιγμές που συνέβησαν τα γεγονότα, το σύστημα έχει στόχο να ανιχνεύσει ομοιότητες μεταξύ χρονοσειρών-προτύπων, με βάση μία παραλλαγή του αλγορίθμου εύρεσης της Μέγιστης Κοινής Υπακολουθίας (Longest Common Subsequence) για χρονοσειρές [VHGK03], ο οποίος είναι ανθεκτικός (robust) υπό συνθήκες “θορύβου”. Οι χρονοσειρές-πρότυπα περιλαμβάνουν την ιδανική σειρά γεγονότων στην περίπτωση που τα γεγονότα μεταξύ των οντοτήτων εμφανίζονται ανά χρονικά διαστήματα ακριβώς ίσα με 1, 7, 15, 30 ημέρες και άλλες χρονοσειρές που έχουν εντοπιστεί στο παρελθόν σε περιστατικά απάτης. Στην περίπτωση όπου ανιχνευτεί ομοιότητα με οποιαδήποτε από τις παραπάνω χρονοσειρές, θεωρούμε ότι η σειρά γεγονότων του υπαλλήλου έχει περιοδικότητα.
- Γεγονότα που συμβαίνουν εκτός ωρών εργασίας:** Οι παράνομες δραστηριότητες συνήθως συμβαίνουν εκτός των ωρών εργασίας, τα Σαββατοκύριακα, σε αργίες ή στο τέλος της βάρδιας των υπαλλήλων. Για το λόγο αυτό, εάν υπάρχουν τέτοια γεγονότα, θα πρέπει να ληφθούν υπόψη.
- Η συχνότητα του υπαλλήλου στο καταγεγραμμένο σύστημα:** Κάθε επάλληλος ανάλογα με τα καθήκοντά του χειρίζεται συγκεκριμένα εταιρικά συστήματα. Εάν δε συμβαίνει αυτό, ο υπάλληλος θα πρέπει να δικαιολογήσει το καταγεγραμμένο γεγονός. Επιπλέον, σε αρκετά εταιρικά συστήματα, όπως αυτά της διαχείρισης απάτης, είναι αναμενόμενο ότι ένας υπάλληλος θα παρακολουθεί έναν “ύποπτο” πελάτη. Επομένως, στα γεγονότα που παράγονται από αυτά τα συστήματα θα πρέπει να δίνεται μικρότερο βάρος.

- **Οι ενέργειες που πραγματοποιούνται από τους υπαλλήλους:** Όμοια με την προηγούμενη περίπτωση, υπάρχουν εξουσιοδοτημένες ενέργειες τις οποίες, όμως, ένας υπάλληλος είναι απίθανο να εκτελέσει, καθώς δε συνάδουν με τα καθήκοντα του στην εταιρεία.

Η ιδέα πίσω από το δέντρο απόφασης είναι η αντιστοίχιση κάθε επιπέδου του δέντρου με ένα πρότυπο απάτης και η κατασκευή ενός μονοπατιού σύμφωνα με το αποτέλεσμα της εξέτασης σε κάθε πρότυπο. Θεωρούμε τη σημαντικότητα των παραγόντων με βάση το επίπεδο στο οποίο αντιστοιχούν στο δέντρο απόφασης, έτσι ώστε τα υψηλότερα επίπεδα (δηλαδή τα πιο κοντινά στη ρίζα) να είναι πιο σημαντικά. Έστω  $x = [x_1, x_2, x_3, x_4, x_5]$  το διάνυσμα-πρότυπο που εξετάζεται (π.χ. το διάνυσμα  $x = [0, 0, 0, 0, 0]$  αντιστοιχεί σε μηδόνια δραστηριότητα) και έστω  $y = [y_1, \dots, y_5]$  το διάνυσμα που προκύπτει από τη διαπέραση του δέντρου απόφασης από τη ρίζα προς τα φύλλα, ανάλογα με την αξιολόγηση των γεγονότων του ζεύγους  $(u, c)$  σε κάθε πρότυπο. Εάν η εξέταση των γεγονότων οδηγήσει σε “Μη εξουσιοδοτημένη ενέργεια” ή “Μη εξουσιοδοτημένο σύστημα”, στη σειρά γεγονότων ανατίθεται η τιμή 1. Διαφορετικά, σε κάθε επίπεδο του δέντρου  $i$  ανατίθεται ένα βάρος, έστω  $w_i, i = 1, \dots, 5$ , με βάση τον τύπο  $w_i = (N - i + 1) / (\sum_{j=1}^N (N - i + 1))$ , όπου  $N$  είναι ο συνολικός αριθμός των επιπέδων του δέντρου [SSE81], έτσι ώστε ανομοιότητες που προκύπτουν μεταξύ των δύο διανυσμάτων στα υψηλότερα επίπεδα του δέντρου να θεωρούνται πιο σημαντικές. Για το λόγο αυτό, η απόσταση μεταξύ των διανυσμάτων  $x$  και  $y$ , έστω  $d(x, y)$ , η οποία αντιστοιχεί στην ανομοιότητα από το διάνυσμα-πρότυπο, υπολογίζεται με βάση τον τύπο της κανονικοποιημένης Ευκλείδειας με βάρη μετρικής:  $d(x, y) = \sqrt{\sum_{i=1}^5 (x_i - y_i)^2 * w_i} / \sqrt{\sum_{i=1}^5 w_i}$ . Η τιμή αυτή (ή η μέγιστη εκ των ήδη υπολογισμένων τιμών, εάν ανιχνευτούν περισσότερα από ένα πρότυπα απάτης) αντιστοιχεί στην τιμή σοβαρότητας της σειράς γεγονότων του υπαλλήλου  $u$ , η οποία τελικά ανατίθεται στον υπάλληλο.

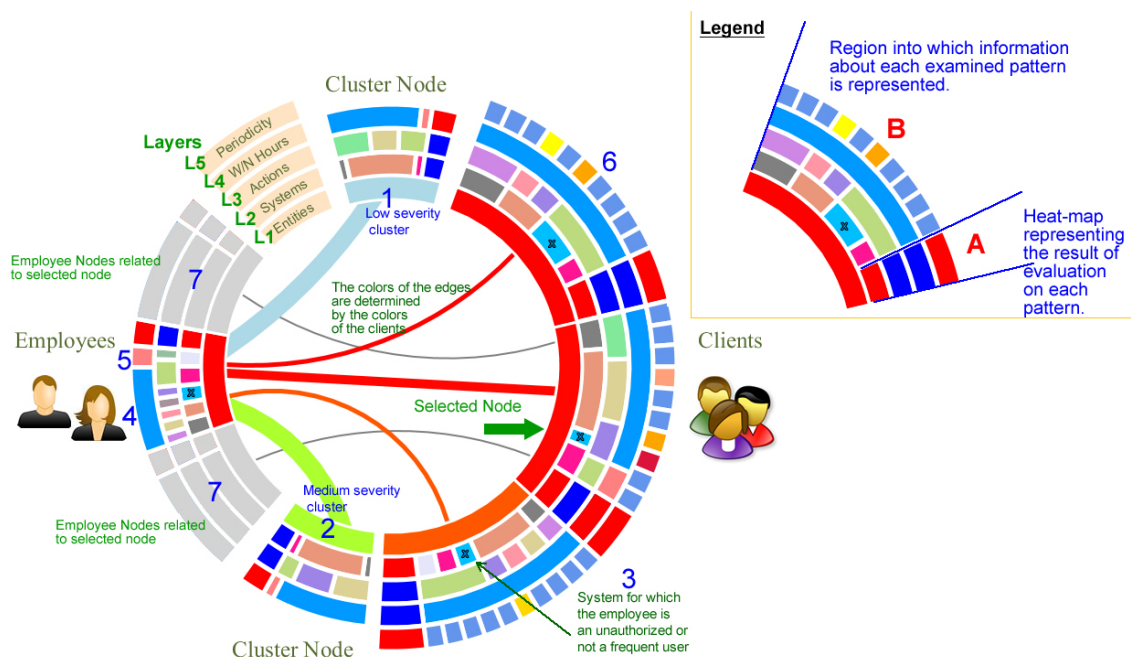
Με βάση τις τιμές αυτές, το σύστημα παράγει ένα θερμικό χάρτη, ο οποίος αναπαριστά όλους τους υπαλλήλους με τετράγωνους κόμβους και διαβαθμίσεις χρωμάτων από το μπλε προς το κόκκινο (βλ. τον πάνω-αριστερά θερμικό χάρτη της Εικόνας 19.1), έτσι ώστε κόμβοι σκιασμένοι με “κοντά-στο-κόκκινο” χρώμα αναπαριστούν υπαλλήλους με ισχυρές υποψίες απάτης, ενώ οι μπλε-σκιασμένοι κόμβοι αναπαριστούν υπαλλήλους χωρίς υποψία απάτης. Όμοια με τον υπολογισμό της σοβαρότητας των σειρών γεγονότων των υπαλλήλων, το σύστημα αναθέτει, επίσης, μία τιμή σοβαρότητας και στους πελάτες με βάση τα παραπάνω πρότυπα. Η μόνη διαφορά είναι ότι για δοθέν πελάτη  $c$ , η τιμή σοβαρότητας υπολογίζεται πάνω σε όλα τα γεγονότα που σχετίζονται με τον πελάτη  $c$  (όχι μόνο αυτά που αφορούν σε συγκεκριμένο υπάλληλο). Με αυτό τον τρόπο, μπορεί να ανιχνευτεί ένας πελάτης που εμπλέκεται σε “ύποπτη” δραστηριότητα με περισσότερους από έναν υπαλλήλους.

### 19.3 Περιγραφή του Συστήματος

Το παράθυρο οπτικοποίησης αποτελείται από δύο θερμικούς χάρτες που απεικονίζουν τη σοβαρότητα της δραστηριότητας των υπαλλήλων και των πελατών (βλ. τα πάνω αριστερά και κάτω αριστερά παράθυρα της Εικόνας 19.1). Παρά το γεγονός ότι αρκετές ερευνητικές εργασίες έχουν επικεντρωθεί στα μειονεκτήματα των έγχρωμων χαρτών με χρώματα της ίριδας (rainbow color maps) [BT107], αυτοί εξακολουθούν να χρησιμοποιούνται ευρέως στις οπτικοποιήσεις, καθώς η αποτελεσματικότητά τους εξαρτάται άμεσα από τη φύση των δεδομένων. Στα σύνολα δεδομένων μας, ήταν αναγκαία η χρήση τριών βασικών χρωμάτων (κόκκινο, πράσινο, μπλε), τα οποία θα αναπαριστούν τη σοβαρότητα (υψηλή, μέση, χαμηλή, αντίστοιχα) των υπό εξέταση οντοτήτων με ευκρίνεια και ως εκ τούτου, επιλέξαμε το συγκεκριμένο χρωματισμό για το θερμικό χάρτη. Ωστόσο, εξετάσαμε διάφορα είδη χρωματικών χαρτών έως ότου καταλήξουμε σε αυτή την επιλογή, αλλά το αποτέλεσμα ήταν αρκετά παραπλανητικό κατά τη διερεύνηση. Η συγκεκριμένη επιλογή των χρωμάτων δεν προκαλούσε σύγχυση ούτε στους ελεγκτές, καθώς στην πραγματικότητα, περιπτώσεις απάτης εμφανίζονται σπάνια σε μία εταιρεία, το οποίο συνεπάγεται ότι τελικά ένας μικρός αριθμός κόκκινων-σκιασμένων κόμβων θα εμφανιστούν τελικά στο θερμικό χάρτη.

Κατά την εκκίνηση του συστήματος παράγεται μόνο ο θερμικός χάρτης που αναπαριστά τη δραστηριότητα των υπαλλήλων. Ο ελεγκτής επιλέγει μία οριακή τιμή που καθορίζει τους υπαλλήλους που θα παρουσιαστούν με βάση τις τιμές σοβαρότητάς τους. Η οπτικοποίηση αποτελείται από κινούμενες εικόνες, στις οποίες εμφανίζονται, ένας προς έναν, οι υπάλληλοι με τους σχετιζόμενους πελάτες τους. Πριν την οπτικοποίηση ενός υπαλλήλου, παράγεται και ο θερμικός χάρτης που αναπαριστά τη δραστηριότητα των σχετιζόμενων πελατών του.

Η κύρια οπτικοποίηση του συστήματος είναι μία πολυεπίπεδη κυκλική απεικόνιση, στην οποία κάθε επίπεδο  $L_1, \dots, L_5$  (βλ. Εικόνα 19.4) αναπαριστά μία διαφορετική πτυχή των δεδομένων ελέγχου (συστήματα, ενέργειες, εντός/εκτός ωρών εργασίας, περιοδικότητα) και κάθε κυκλικός τομέας αναπαριστά μία οντότητα (υπάλληλος, πελάτης ή ομάδα). Ακολουθώντας, παράγεται ένα γράφημα, του οποίου οι κόμβοι αντιστοιχούν σε καθεμία από τις παραπάνω οντότητες, ενώ οι ακμές αναπαριστούν τις μεταξύ τους σχέσεις. Το τελευταίο εσωτερικά επίπεδο της οπτικοποίησης (το επίπεδο  $L_1$  της Εικόνας 19.4) περιέχει τους κόμβους του γραφήματος (που απεικονίζονται ως τμήματα ενός δακτυλίου). Οι κόμβοι που αντιστοιχούν στους υπαλλήλους απεικονίζονται στο αριστερό τμήμα της οπτικοποίησης, ενώ εκείνοι που αντιστοιχούν στους σχετιζόμενους πελάτες στο δεξί τμήμα της οπτικοποίησης. Για την αποφυγή επιβάρυνσης της οπτικοποίησης με κόμβους που αντιστοιχούν σε πελάτες χωρίς υποψία απάτης, ο ελεγκτής δύναται να προσδιορίσει οριακές τιμές, οι οποίες διαμερίζουν τους πελάτες σε μία ή δύο ομάδες (clusters) ανάλογα με την τιμή σοβαρότητάς τους (ομάδα χαμηλής σοβαρότητας και/ή ομάδα μεσαίας σοβαρότητας). Οι κόμβοι αυτοί τοποθετούνται



Σχήμα 19.4: Περιγραφή της κύριας οπτικοποίησης του συστήματος.

στο πάνω και στο κάτω τμήμα της οπτικοποίησης. Το χρώμα κάθε κόμβου (εκτός από εκείνων που αντιστοιχούν στις ομάδες, καθώς και των γκρι-σκιασμένων κόμβων στους οποίους θα αναφερθούμε παρακάτω) ακολουθούν το χρώμα της αντίστοιχης οντότητας στους θερμικούς χάρτες. Ο κόμβος που αντιστοιχεί σε μία ομάδα και είναι σκιασμένος με ανοιχτό μπλε (πράσινο, αντίστοιχα) αντιστοιχεί στην ομάδα χαμηλής (μεσαίας, αντίστοιχα) σοβαρότητας (βλ. τα σημεία αναφοράς 1 και 2 της Εικόνας 19.4, αντίστοιχα). Όσον αφορά τις ακμές της απεικόνισης, το σύστημα υποστηρίζει είτε ακμές με ευθείες γραμμές (straight-line edges) είτε κυκλικά τόξα (circular arcs). Το πάχος κάθε ακμής είναι ανάλογο του αριθμού των συνδέσεων μεταξύ του υπαλλήλου και του πελάτη, ενώ το χρώμα της καθορίζεται από το χρώμα του πελάτη. Στα πλαίσια της ανίχνευσης απάτης, “μεγάλου-πάχους” ακμές (εξαιρουμένων των ακμών που συνδέουν κόμβους που αντιστοιχούν σε υπαλλήλους με κόμβους που αντιστοιχούν σε ομάδες) και πιο συγκεκριμένα, οι κόκκινες-σκιασμένες ακμές αποτελούν ενδείξεις απάτης που πρέπει να εξεταστούν περαιτέρω.

Τα επόμενα επίπεδα (δηλαδή τα επίπεδα  $L_2 - L_5$ ) αντιστοιχούν στα πρότυπα που περιγράφονται στην Ενότητα 19.2 και χωρίζονται σε δύο περιοχές A και B (βλ. Εικόνα 19.4). Η περιοχή A αναπαριστά ένα θερμικό χάρτη που περιγράφει το αποτέλεσμα της εξέτασης της οντότητας στο συγκεκριμένο πρότυπο. Το κόκκινο χρώμα υποδηλώνει τον εντοπισμό ύποπτου μοτίβου. Η περιοχή B αναπαριστά πληροφορίες σχετικές με το αντίστοιχο εξεταζόμενο πρότυπο. Στο επίπεδο  $L_2$ , αναπαρίστανται τα διάφορα εταιρικά συστήματα που σχετίζονται

με καθεμία από τις οντότητες. Κάθε τέτοιο σύστημα χαρακτηρίζεται από ένα συγκεκριμένο χρώμα και καταλαμβάνει χώρο ανάλογο με το αντίστοιχο συνολικό ποσοστό της χρήσης από την οντότητα. Για κάθε υπάλληλο, το ποσοστό αυτό υπολογίζεται με βάση το συνολικό ποσοστό χρήσης πάνω σε όλους τους πελάτες της παρούσας οπτικοποίησης, ενώ για κάθε πελάτη με βάση το ποσοστό χρήσης από τον υπάλληλο που εμφανίζεται στην οπτικοποίηση (εκτός εάν περισσότεροι του ενός υπάλληλοι που σχετίζονται με τον πελάτη οπτικοποιούνται ταυτόχρονα). Όμοια, για τους κόμβους που αντιστοιχούν σε ομάδες, το συνολικό ποσοστό χρήσης υπολογίζεται με βάση σε όλους τους πελάτες που ανήκουν στην ομάδα. Τα συστήματα για τα οποία ο υπάλληλος δεν είναι εξουσιοδοτημένος χρήστης ή δε χρησιμοποιεί τακτικά, επισημαίνονται με  $X$  (βλ. το σημείο αναφοράς 3 της Εικόνας 19.4).

Το επίπεδο  $L_3$  αναπαριστά τις ενέργειες που καταγράφηκαν για κάθε οντότητα και απεικονίζονται με τρόπο παρόμοιο με εκείνο του επιπέδου  $L_2$ . Και πάλι, μη εξουσιοδοτημένες ή “ύποπτες” ενέργειες επισημαίνονται με  $X$ . Το επίπεδο  $L_4$  αναπαριστά το ποσοστό των γεγονότων που συμβαίνουν εντός ή εκτός των ωρών εργασίας. Τα τμήματα που είναι σκιασμένα με ανοιχτό-μπλε αναπαριστούν γεγονότα που συνέβησαν εντός των ωρών εργασίας, ενώ τα κόκκινα-σκιασμένα τμήματα υποδεικνύουν την ύπαρξη γεγονότων που συνέβησαν εκτός των ωρών εργασίας (π.χ., βλ. τα σημεία αναφοράς 4 και 5 της Εικόνας 19.4, αντίστοιχα). Για κάθε κόμβο που αντιστοιχεί σε ένα πελάτη υπάρχει ένα επιπλέον επίπεδο (βλ. το επίπεδο  $L_5$  της Εικόνας 19.4, π.χ. το σημείο αναφοράς 6), το οποίο δείχνει εάν η σειρά γεγονότων του πελάτη είναι περιοδική. Η σειρά γεγονότων συγκρίνεται με τις χρονοσειρές-πρότυπα, οι οποίες είναι αποθηκευμένες στο σύστημα και ένας θερμικός χάρτης παράγεται, παρουσιάζοντας το βαθμό ομοιότητας της με κάθε χρονοσειρά-πρότυπο. Και πάλι, το κόκκινο χρώμα φανερώνει ύποπτες περιπτώσεις.

Όσον αφορά στη διαδικασία διερεύνησης, όπως έχει ήδη αναφερθεί, ο ελεγκτής καθορίζει μία οριακή τιμή και οι υπάλληλοι με τιμή σοβαρότητας μεγαλύτερη της οριακής παρουσιάζονται στην οπτικοποίηση ένας προς έναν, από κοινού με τους σχετιζόμενους πελάτες τους. Ο ελεγκτής δύναται να ξεκινήσει, να παύσει ή να διακόψει το βίντεο και να επεξεργαστεί την οπτικοποίηση. Στην περίπτωση που ένας κόμβος που αντιστοιχεί σε ένα πελάτη έχει επιλεγεί, οι υπόλοιποι υπάλληλοι που σχετίζονται με τον πελάτη μπορούν να προστεθούν στην οπτικοποίηση, το οποίο διευκολύνει τον πιθανό εντοπισμό δύο ή περισσότερων υπαλλήλων που μπορεί να συνεργάζονται για τη διάπραξη απάτης (η περίπτωση που ένας κόμβος που αντιστοιχεί σε υπάλληλο επιλεγεί, αντιμετωπίζεται όμοια). Στην Εικόνα 19.4, οι γκρι-σκιασμένοι κόμβοι (βλ. σημείο αναφοράς 7) αναπαριστούν υπαλλήλους που έχουν προστεθεί κατά τη διάρκεια της διερεύνησης, κατά την επιλογή ενός κόμβου που αντιστοιχεί σε ένα πελάτη (βλ. τον κόμβο που δείχνει το πράσινο βέλος της Εικόνας 19.4). Στην περίπτωση που περισσότεροι του ενός κόμβοι, οι οποίοι αναπαριστούν υπαλλήλους, υπάρχουν ταυτό-

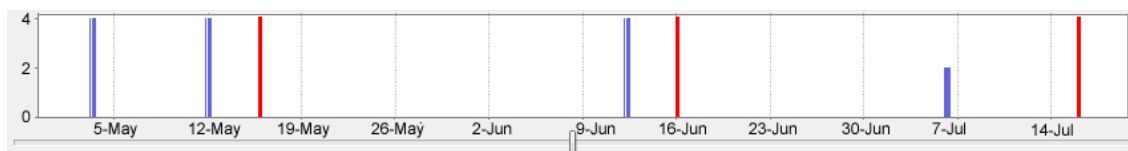
χρονα στην οπτικοποίηση, ο ελεγκτής δύναται να επιλέξει έναν από αυτούς και να προσθέσει τους σχετιζόμενους πελάτες του στην οπτικοποίηση. Το γκρι-χρώμα χρησιμοποιείται για μη επιλεγμένους κόμβους που αντιστοιχούν σε υπαλλήλους, τις ακμές τους και τους σχετιζόμενους πελάτες τους (εφόσον δεν σχετίζονται, επίσης, και με τον επιλεγμένο υπάλληλο), ώστε να μην αποσπάται η προσοχή του ελεγκτή.

Δεδομένου ότι είναι δυνατή η εναλλαγή μεταξύ υπαλλήλων κατά τη διάρκεια της διερεύνησης, στην περίπτωση που η μετέπειτα επεξεργασία μιας υπόθεσης ολοκληρωθεί, το βίντεο συνεχίζει από τον τελευταίο υπάλληλο που οπτικοποιούνταν πριν την παύση του βίντεο. Στην περίπτωση που επιλεγεί ένας κόμβος που αναπαριστά μία ομάδα, τα αντίστοιχα ορθογώνια του θερμικού χάρτη, ο οποίος αναπαριστά τους πελάτες, επισημαίνονται με  $X$  έτσι ώστε οι αντίστοιχοι πελάτες να μπορούν να προστεθούν (αν είναι αναγκαίο) στην οπτικοποίηση. Ωστόσο, το σύστημα επιτρέπει ένα συγκεκριμένο αριθμό προσθέσεων κόμβων που αντιστοιχούν σε υπαλλήλους ή πελάτες. Σε περίπτωση υπέρβασης αυτού του αριθμού, το σύστημα δύναται προαιρετικά να παράγει μία οπτικοποίηση, στην οποία απεικονίζεται μόνο το εσωτερικό επίπεδο της οπτικοποίησης (δηλαδή εκείνο που αντιστοιχεί στους πελάτες και τους υπαλλήλους) και οι μεταξύ τους σχέσεις. Και πάλι, το πλάτος των ακμών είναι ανάλογο του αριθμού των γεγονότων που συσχετίζουν τις δύο οντότητες. Στην περίπτωση όπου απαιτείται περαιτέρω διερεύνηση, ο ελεγκτής επιλέγει τον επιθυμητό κόμβο (ο οποίος μεγαλώνει) και παρουσιάζονται τα υπόλοιπα επίπεδα της οπτικοποίησης (δηλαδή τα επίπεδα  $L_2 - L_5$ ), τα οποία αντιστοιχούν στο συγκεκριμένο κόμβο. Με τον τρόπο αυτό, το σύστημα είναι σε θέση να οπτικοποιήσει ταυτόχρονα ένα μεγαλύτερο σύνολο οντοτήτων και να αποκαλύψει τις σχέσεις μεταξύ τους. Ωστόσο, αυτή η επιλογή μπορεί να επιβραδύνει τη διαδικασία διερεύνησης και για το λόγο αυτό, υιοθετήσαμε την προσέγγιση των κινούμενων εικόνων και την ταυτόχρονη οπτικοποίηση όλων των επιπέδων για κάθε κόμβο.

Η διάταξη των κόμβων που αναπαριστούν τους υπαλλήλους πραγματοποιείται ανάλογα με την τιμή σοβαρότητάς τους (οι πιο ύποπτοι κόμβοι παρουσιάζονται πρώτοι στις κινούμενες εικόνες). Οι πελάτες εμφανίζονται σε αυθαίρετη σειρά, δεδομένου ότι τομές μεταξύ ακμών που συνδέουν ένα συγκεκριμένο υπάλληλο με τους σχετιζόμενους πελάτες του δεν μπορεί να υπάρχουν. Οι μόνες τομές που μπορεί να προκύψουν, προκαλούνται από τους γκρι-σκιασμένους κόμβους των υπαλλήλων, οι οποίοι σχετίζονται με πελάτες που έχουν ήδη απεικονιστεί και δεδομένου ότι αυτές οι ακμές είναι, επίσης, γκρι-χρώματος, δεν προκαλούν σύγχυση στον ελεγκτή (βλ. Εικόνα 19.4). Οι γκρι-σκιασμένοι υπάλληλοι που προστίθενται στην οπτικοποίηση τοποθετούνται είτε πάνω είτε κάτω από τους ήδη τοποθετημένους κόμβους των υπαλλήλων, ώστε να διατηρηθούν οι σχετικές θέσεις των ήδη τοποθετημένων κόμβων. Έχουμε, επίσης, επιλέξει να μην εφαρμόσουμε ευρετικούς αλγόριθμους για την ελαχιστοποίηση του αριθμού των τομών μεταξύ ακμών, δεδομένου ότι η προσθήκη νέων κόμβων στην



οπτικοποίηση μπορεί να συνεπάγεται μία αναδιάταξη των θέσεων των ήδη τοποθετημένων κόμβων, το οποίο διαταράσσει τον υποκείμενο νοητό χάρτη.

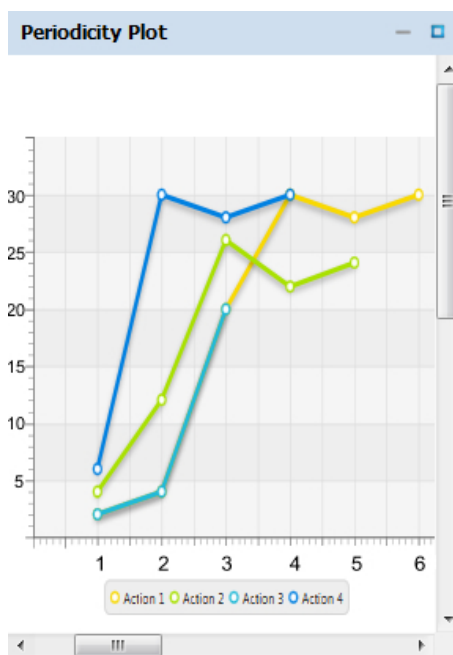


**Σχήμα 19.5:** Ένα διάγραμμα χρόνου, το οποίο αναπαριστά τη σειρά γεγονότων για συγκεκριμένο ζεύγος οντοτήτων.

Η Εικόνα 19.5 παρουσιάζει ένα διάγραμμα χρόνου, το οποίο αναπαριστά τη σειρά γεγονότων για ένα συγκεκριμένο ζεύγος οντοτήτων (βλ. τη μπλε σειρά), όπου ο  $x$ -άξονας αντιστοιχεί στην ημερομηνία της εμφάνισης κάθε γεγονότος και ο  $y$ -άξονας στον αριθμό των γεγονότων που συνέβησαν κατά τη συγκεκριμένη ημερομηνία. Οι κόκκινες ράβδοι αντιστοιχούν στις ημερομηνίες χρέωσης του λογαριασμού του συγκεκριμένου πελάτη. Το διάγραμμα αυτό διευκολύνει την αναγνώριση πιθανής περιοδικής δραστηριότητας, ειδικότερα κοντά στην ημερομηνία χρέωσης του λογαριασμού του πελάτη.

Σε σχήματα εταιρικής απάτης, είναι επίσης πιθανό, η σειρά γεγονότων που σχετίζεται με ένα ζεύγος οντοτήτων να είναι περιοδική μόνο ως προς μία συγκεκριμένη ενέργεια. Για το λόγο αυτό, έχουμε ενσωματώσει στο σύστημα ένα δεύτερο διάγραμμα (βλ. Εικόνα 19.6), το οποίο παρουσιάζει όλες τις ενέργειες που καταγράφηκαν χρησιμοποιώντας διακριτές σειρές (series). Σε αυτή τη γραφική παράσταση, ο  $y$ -άξονας αντιστοιχεί στην ημέρα του μήνα κατά την οποία συνέβη κάθε γεγονός που σχετίζεται με μία συγκεκριμένη ενέργεια (π.χ. το πρώτο γεγονός που σχετίζεται με μία ενέργεια και συνέβη την 15η ημέρα του μήνα θα τοποθετηθεί στο σημείο (1, 15)). Για την περίπτωση αυτή, αγνοούμε πολλαπλές εμφανίσεις γεγονότων που σχετίζονται με την ίδια ενέργεια και συνέβησαν την ίδια ημέρα του μήνα. Στην περίπτωση που εμφανιστεί περιοδικότητα ως προς μία συγκεκριμένη ενέργεια, η αντίστοιχη σειρά θα έχει ένα μέρος (ή το σύνολο της) σχεδόν παράλληλο με τον  $x$ -άξονα.

Το σύστημα υποστηρίζει μηχανισμούς για τον εντοπισμό προτύπων ασυνήθιστης δραστηριότητας υπαλλήλων, όπως μη εξουσιοδοτημένη πρόσβαση σε υπολογιστές, εταιρικά συστήματα και λογαριασμούς υπαλλήλων ή πελατών, με την παραγωγή ενός διαγράμματος παράλληλων συντεταγμένων (βλ. Εικόνα 19.7). Κάθε εγγραφή αποτελείται από έναν υπάλληλο, τη χρονική στιγμή που συνέβη το γεγονός (time-stamp), την IP διεύθυνση του υπολογιστή του υπαλλήλου, το όνομα του υπολογιστή (computer name) και μία ενέργεια (π.χ. login, login failure, κλπ). Το μέγεθος των κόμβων και των ακμών είναι ανάλογο του αριθμού των εμφανίσεων τους στη βάση δεδομένων. Οι κόμβοι σε κάθε επίπεδο ταξινομούνται με βάση τον αριθμό των εμφανίσεων τους στο σύνολο των δεδομένων.

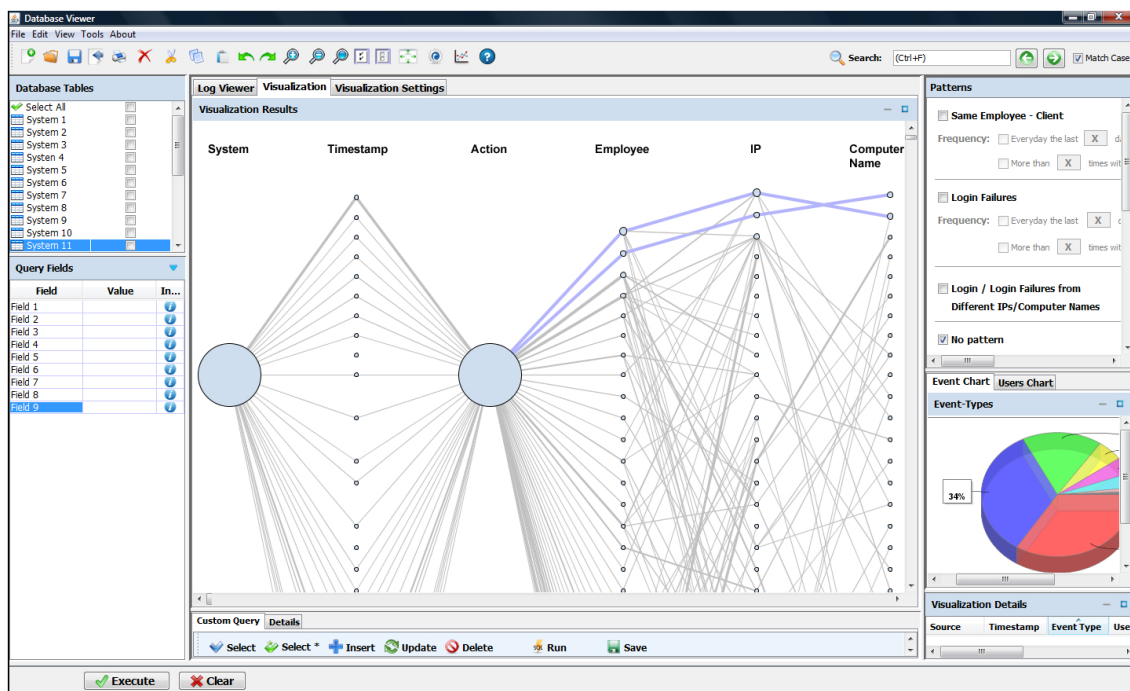


**Σχήμα 19.6:** Ένα διάγραμμα περιοδικότητας που παρουσιάζει όλες τις καταγεγραμμένες ενέργειες με διακριτές σειρές για το συγκεκριμένο ζεύγος υπαλλήλου-πελάτη.

Τα πρότυπα που εξετάζονται σε αυτό το σενάριο περιλαμβάνουν τα ακόλουθα: (i) περισσότερες από  $X$  αποτυχημένες προσπάθειες σύνδεσης μέσα σε ένα χρονικό διάστημα  $Y$  ημερών/μηνών, όπου οι τιμές  $X$  και  $Y$  καθορίζονται από τον ελεγκτή και (ii) προσπάθειες σύνδεσης ή αποτυχημένες προσπάθειες σύνδεσης από διαφορετικές IP διευθύνσεις ή/και ονόματα υπολογιστών. Η οπτικοποίηση της Εικόνας 19.4 μπορεί, επίσης, να προσαρμοστεί και σε αυτό το σενάριο -είτε με είτε χωρίς το στάδιο της προ-επεξεργασίας (δεδομένου ότι ο αριθμός των υπαλλήλων και των ενεργειών είναι διαχειρίσιμος σε μία κυκλική οπτικοποίηση)- αντικαθιστώντας τους κόμβους που αντιστοιχούν στους πελάτες από τους κόμβους που αντιστοιχούν στις ενέργειες.

## 19.4 Μελέτη Περίπτωσης

Σε αυτή την ενότητα, παρουσιάζουμε τα αποτελέσματα της αξιολόγησης του συστήματος σε ένα πραγματικό σύνολο δεδομένων που καταγράφηκαν από δύο εταιρικά συστήματα. Όλα τα δεδομένα ήταν ανώνυμα για λόγους ασφάλειας και ιδιωτικότητας. Το σύνολο δεδομένων αποτελείται από περίπου 180.637 καταχωρήσεις που καλύπτουν ένα χρονικό διάστημα έξι μηνών. Αποτελείται, επίσης, από 710 διακριτούς υπαλλήλους και 83.030 διακριτούς πελάτες. Στο σύνολο δεδομένων, το 66.2% των πελατών είχε μόνο μία εμφάνιση στο σύνολο



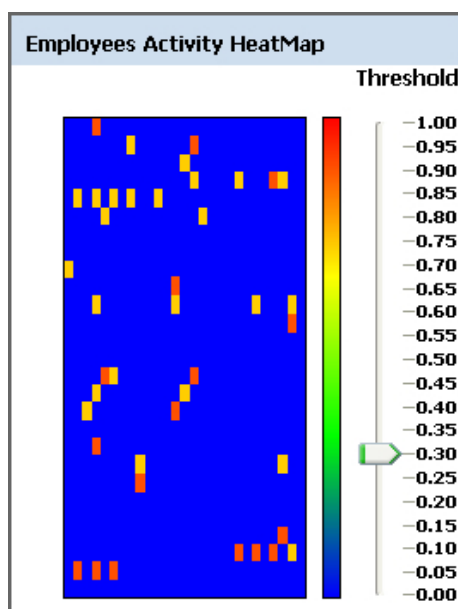
**Σχήμα 19.7:** Ένα διάγραμμα παράλληλων συντεταγμένων για την παρακολούθηση των αποτυχημένων προσπαθειών σύνδεσης σε ένα συγκεκριμένο σύστημα.

των δεδομένων, το 31.6% των πελατών μεταξύ 2 και 5, το 1.4% των πελατών μεταξύ 5 και 10, ενώ οι υπόλοιποι πελάτες (δηλαδή το 0.8% των πελατών) είχαν πάνω από 10 καταγεγραμμένα γεγονότα. Οι ελεγκτές συμπεριέλαβαν, επίσης, μία σειρά από καταχωρήσεις που αντιστοιχούν σε ένα “φανταστικό” σενάριο απάτης, στο οποίο ένας υπάλληλος τροποποιεί το λογαριασμό ενός πελάτη. Ωστόσο, δεν μας κοινοποιήθηκαν πληροφορίες σχετικά με την ημερομηνία χρέωσης των λογαριασμών των πελατών.

Δεδομένου ότι ένα από τα σύνολα δεδομένων προέρχεται από ένα σύστημα διαχείρισης απάτης, είναι αναμενόμενο να προκύπτει επαναλαμβανόμενη δραστηριότητα μεταξύ του ίδιου ζεύγους υπαλλήλου και πελάτη (ένας “ύποπτος” πελάτης που αναγνωρίστηκε από ένα σύστημα διαχείρισης απάτης αναμένεται να εποπτεύεται από έναν υπάλληλο). Για το λόγο αυτό, επικεντρώσαμε τη μελέτη μας στον εντοπισμό ζευγών υπαλλήλων και πελατών που σχετίζονται μεταξύ τους με περισσότερα από 10 γεγονότα. Το σύστημα εντόπισε 41 υπαλλήλους (5.8% του συνολικού αριθμού των υπαλλήλων) που σχετίζονται με τον ίδιο πελάτη, με περισσότερα από 10 γεγονότα (βλ. τα πορτοκαλί και κόκκινα σκιασμένα ορθογώνια του θερμικού χάρτη της Εικόνας 19.8).

Για καθέναν από τους παραπάνω υπαλλήλους, έπρεπε να υπολογιστεί η ομοιότητα της σειράς γεγονότων με τις χρονοσειρές-πρότυπα (βλ. Ενότητα 19.2) με σκοπό την ανίχνευση

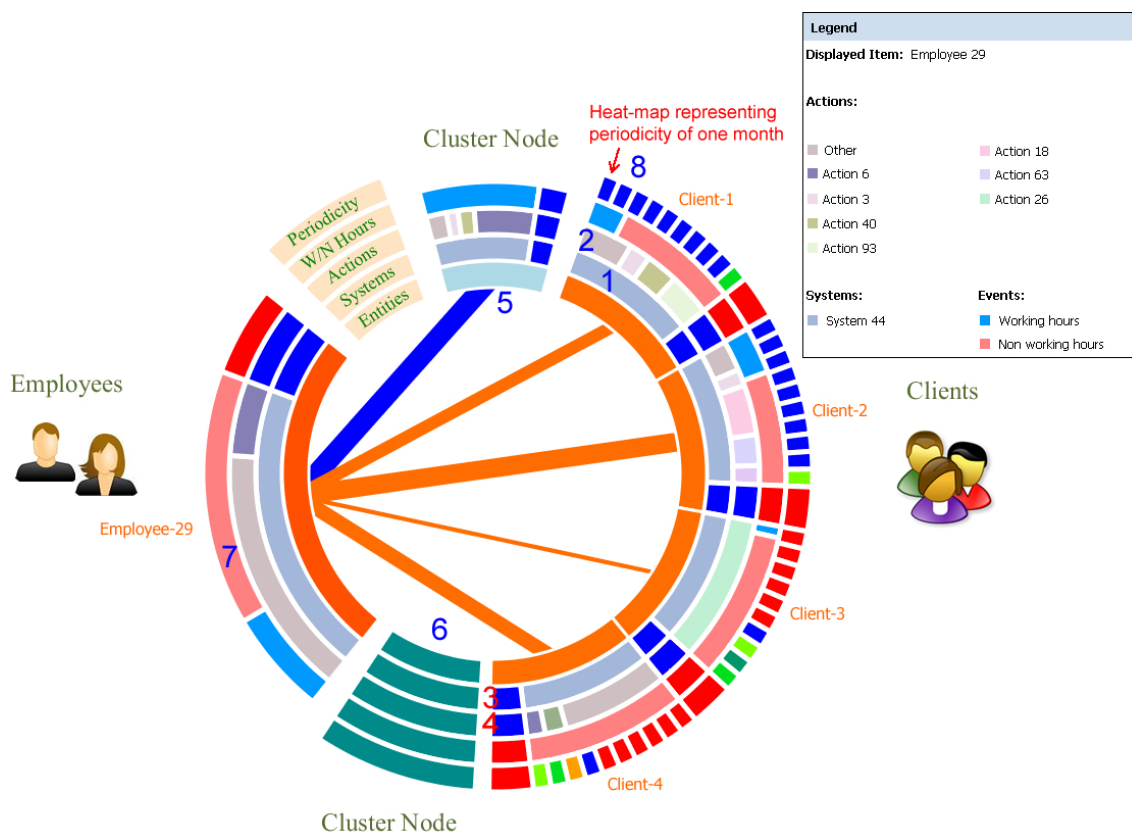
περιοδικότητας. Ειδικότερα, οι ελεγκτές ενδιαφέρονταν για περιοδικά γεγονότα που συμβαίνουν ανά μήνα (δηλαδή περιοδικότητα από 28 έως 30 ημέρες). Επομένως, θα μπορούσαμε να θεωρήσουμε μόνο τη χρονοσειρά-πρότυπο, η οποία αντιστοιχεί στη μηνιαία δραστηριότητα. Ωστόσο, αποφασίσαμε να υπολογίσουμε τις τιμές σοβαρότητας με βάση όλες τις χρονοσειρές-πρότυπα, προκειμένου να διακρίνουμε οποιαδήποτε περιοδική δραστηριότητα και να αποφασίσουμε στη συνέχεια, από την οπτικοποίηση, εάν υπάρχει πράγματι “ύποπτη” δραστηριότητα. Παρόλο που η συγκεκριμένη προσέγγιση δημιουργεί περισσότερες ψευδώς θετικές καταστάσεις, οι οποίες θα πρέπει μετέπειτα να διερευνηθούν, μας εξασφαλίζει ότι τυχόν άλλα περιοδικά γεγονότα δε θα παραληφθούν. Μεταξύ αυτών των 41 υπαλλήλων που σχετίζονταν με τον ίδιο πελάτη με περισσότερα από 10 γεγονότα, οι 17 από αυτούς (το 2.4% του συνολικού αριθμού των υπαλλήλων) εμφάνιζαν περιοδική δραστηριότητα και γεγονότα που συνέβησαν εκτός των ωρών εργασίας (βλ. τα κόκκινα σκιασμένα ορθογώνια της Εικόνας 19.8), ενώ οι υπόλοιποι (24 υπάλληλοι ή το 3.4% του συνολικού αριθμού των υπαλλήλων) είχαν γεγονότα που συνέβησαν εκτός των ωρών εργασίας (βλ. τα πορτοκαλί σκιασμένα ορθογώνια της Εικόνας 19.8). Επιπρόσθετα, κανένας υπάλληλος δεν είχε πραγματοποιήσει μη εξουσιοδοτημένη πρόσβαση σε συστήματα ή μη εξουσιοδοτημένες ενέργειες, ενώ μόνο ένας υπάλληλος είχε χρησιμοποιήσει μη συνηθισμένες ενέργειες. Τα αποτελέσματα κοινοποιήθηκαν στους ελεγκτές για τη διερεύνηση του κατά πόσον υπάρχουν πραγματικές ενδείξεις απάτης.



**Σχήμα 19.8:** Ένας θερμικός χάρτης που αναδεικνύει τις τιμές σοβαρότητας για τους υπαλλήλους του συνόλου δεδομένων που χρησιμοποιήθηκε στην παρούσα μελέτη περίπτωσης.

Στη συνέχεια, παρουσιάζονται μερικές από τις πρώτες κινούμενες εικόνες του συστήματος, προκειμένου να περιγραφεί η διαδικασία διερεύνησης. Η Εικόνα 19.9 παρουσιάζει έναν υπάλληλο με υψηλή τιμή σοβαρότητας (σ.σ. "Employee-29"). Ο υπάλληλος "Employee-29" σχετίζεται με τέσσερις πελάτες (σ.σ. "Client-1", "Client-2", "Client-3" και "Client-4") με περισσότερα από 10 γεγονότα, χρησιμοποιώντας ένα μοναδικό σύστημα, στο οποίο ο υπάλληλος ήταν "τακτικός" χρήστης και εκτελώντας συνηθισμένες ενέργειες (βλ. τα σημεία αναφοράς 1 και 2; τα τμήματα που αντιστοιχούν στα συστήματα και στις ενέργειες δεν επισημαίνονται με  $X$ , το οποίο συνεπάγεται ότι ο χρήστης χρησιμοποιεί συχνά το συγκεκριμένο σύστημα και τις συγκεκριμένες ενέργειες, ενώ και οι αντίστοιχοι θερμικοί χάρτες είναι σκιασμένοι με μπλε; βλ. τα σημεία αναφοράς 3 και 4 της Εικόνας 19.9). Όλοι οι υπόλοιποι πελάτες που σχετίζονται με τον υπάλληλο "Employee-29", τοποθετήθηκαν στην ομάδα χαμηλής σοβαρότητας, ενώ κανένας πελάτης δεν τοποθετήθηκε στην ομάδα μεσαίας σοβαρότητας (βλ. τα σημεία αναφοράς 5 και 6 της Εικόνας 19.9, αντίστοιχα). Τα περισσότερα από τα γεγονότα που σχετίζονται με τον υπάλληλο "Employee-29" συνέβησαν εκτός των ωρών εργασίας (βλ. το σημείο αναφοράς 7 της Εικόνας 19.9). Το επίπεδο της οπτικοποίησης που αναπαριστά τις ενέργειες που πραγματοποίησε ο υπάλληλος "Employee-29", έχει διαμεριστεί σε δύο τμήματα που αναπαριστούν την ενέργεια "Action 6" και "Other". Στην ενέργεια "Other", έχουμε ομαδοποιήσει ενέργειες, των οποίων τα ποσοστά χρήσης ήταν πολύ μικρά ( $< 0.1\%$ ) για να οπτικοποιηθούν. Το συγκεκριμένο πρόβλημα προκαλείται εξαιτίας του μεγάλου αριθμού των πιθανών ενεργειών στο συγκεκριμένο σύστημα, ο οποίος δεν επιτρέπει την ταυτόχρονη οπτικοποίησή τους. Ωστόσο, εάν μία από τις ενέργειες που έχουν ομαδοποιηθεί δεν είναι συνηθισμένη για ένα συγκεκριμένο υπάλληλο, το κόκκινο χρώμα στο θερμικό χάρτη του αντίστοιχου επιπέδου και η επισήμανση με  $X$  στο τμήμα που αναπαριστά την ενέργεια "Other" θα αναδείξει το πρόβλημα στον ελεγκτή.

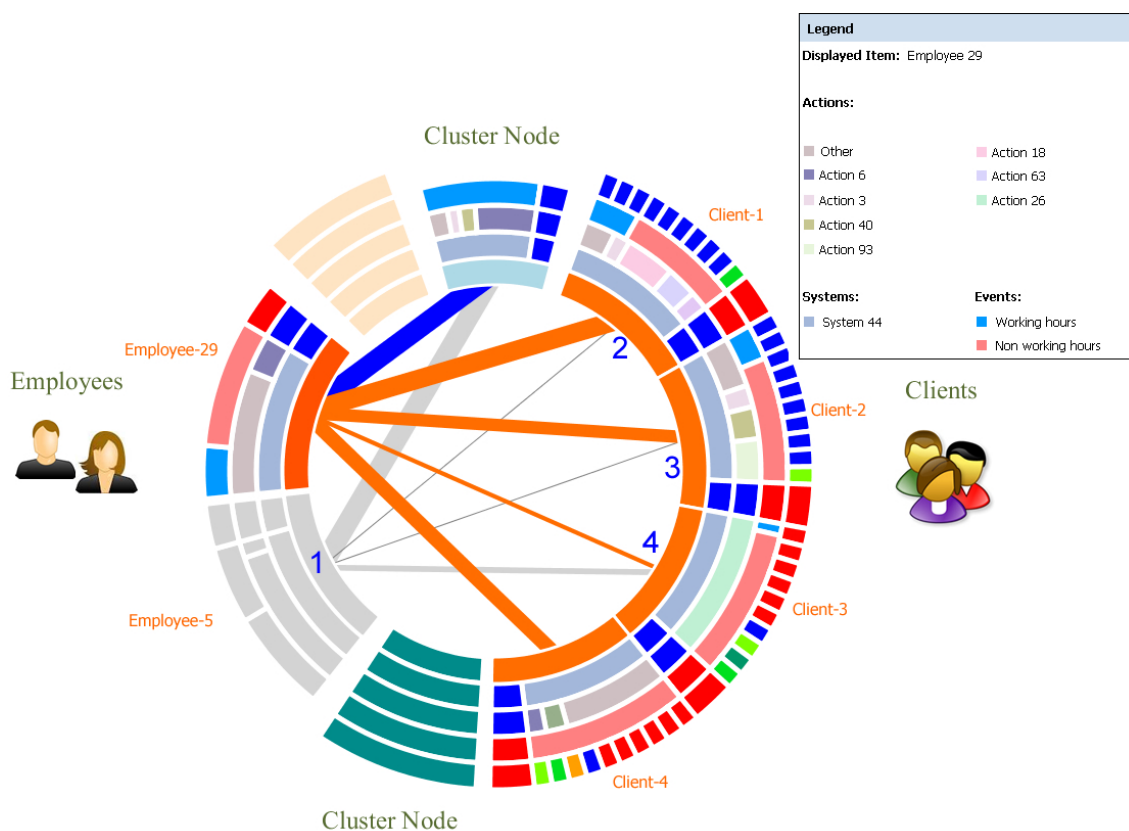
Θα προχωρήσουμε τώρα στην περιγραφή της διαδικασίας διερεύνησης για καθέναν από τους τέσσερις πελάτες. Οι πελάτες "Client-1" και "Client-2" έχουν την πλειοψηφία των γεγονότων που σχετίζονται με τον υπάλληλο "Employee-26" εκτός των ωρών εργασίας (βλ. το αντίστοιχο επίπεδο των πελατών "Client-1" και "Client-2"). Ωστόσο, φαίνεται να έχουν μία μικρή ομοιότητα (λιγότερο από 50%) με μία μόνο χρονοσειρά-πρότυπο. Σημειώνεται ότι, η χρονοσειρά-πρότυπο που αντιστοιχεί στην περιοδικότητα ενός μήνα αντιστοιχεί στο πρώτο θερμικό χάρτη του κάθε επιπέδου που αντιστοιχεί στην περιοδικότητα (βλ. π.χ. το σημείο αναφοράς 8 της Εικόνας 19.9). Δεδομένου ότι οι ελεγκτές ενδιαφέρονται για γεγονότα που εμφανίζουν μηνιαία περιοδικότητα, οι πελάτες αυτοί δεν θεωρήθηκαν "ύποπτοι" απάτης. Όσον αφορά στους πελάτες "Client-3" και "Client-4", έχουν επίσης την πλειοψηφία των γεγονότων που τους σχετίζουν με τον υπάλληλο "Employee-26" εκτός των ωρών εργασίας, αλλά φαίνεται να έχουν ισχυρή περιοδική δραστηριότητα. Συγκεκριμένα, υπάρχουν έξι



**Σχήμα 19.9:** Μία οπτικοποίηση που παρουσιάζει τη δραστηριότητα του υπαλλήλου “Employee-26” με υψηλή τιμή σοβαρότητας.

θερμικοί χάρτες που αναδεικνύουν ισχυρές ομοιότητες με τις χρονοσειρές-πρότυπα, συμπεριλαμβανομένης και αυτής που αντιστοιχεί σε μηνιαία δραστηριότητα. Για το λόγο αυτό, οι συγκεκριμένοι πελάτες θα πρέπει να υποστούν περαιτέρω διερεύνηση.

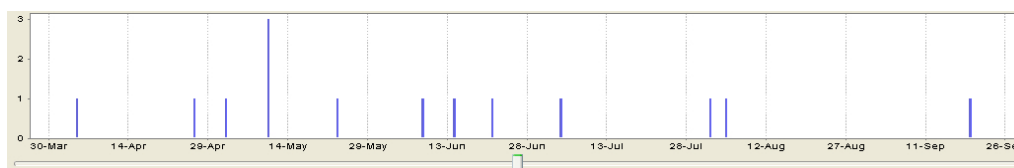
Ο πελάτης “Client-4” δε σχετίζεται με κανένα άλλον υπάλληλο (επιλέγοντας τον κόμβο που αντιστοιχεί στον πελάτη “Client-4”, κανέναν άλλος υπάλληλος δεν προστέθηκε στην οπτικοποίηση). Ωστόσο, επιλέγοντας τον πελάτη “Client-3”, ένας νέος υπάλληλος (“Employee-5”) προστέθηκε στην οπτικοποίηση (βλ. το γκρι-σκιασμένο κόμβο της οπτικοποίησης; βλ. το σημείο αναφοράς 1 της Εικόνας 19.10). Το ενδιαφέρον ήταν, ότι ο υπάλληλος “Employee-5” σχετίζονταν σχεδόν με όλους τους πελάτες του υπαλλήλου “Employee-29” (εκτός από τον πελάτη “Client-4”, βλ. τα σημεία αναφοράς 2, 3 και 4 της Εικόνας 19.10), οι οποίοι είχαν υψηλή τιμή σοβαρότητας. Υπάρχουν δύο πιθανές εξηγήσεις για αυτό το σενάριο. Είτε οι δύο υπάλληλοι είναι υπεύθυνοι για την παρακολούθηση της δραστηριότητας των συγκεκριμένων πελατών ή είναι συνεργοί σε απάτη. Επομένως, η διερεύνηση θα πρέπει να προχωρήσει περαιτέρω. Το διάγραμμα χρόνου της Εικόνας 19.11i αναπαριστά τις χρονικές στιγμές των



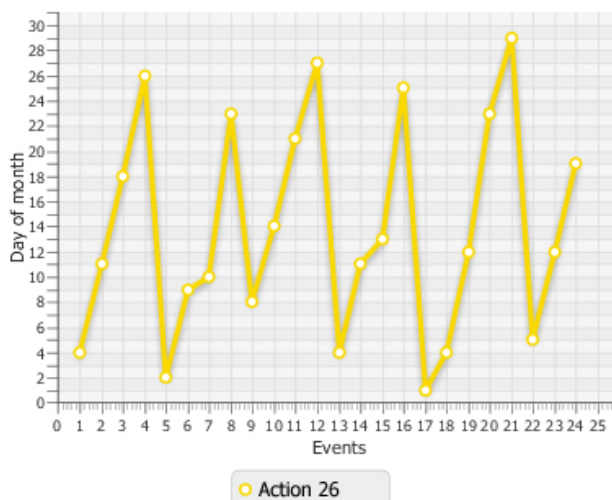
**Σχήμα 19.10:** Όλοι οι υπάλληλοι που σχετίζονται με τον πελάτη “Client-3” έχουν προστεθεί στην οπτικοποίηση. Οι γκρι-σκιασμένοι κόμβοι αντιστοιχούν στο νέο υπάλληλο (“Employee-5”) που προστέθηκε, ο οποίος απροσδόκητα σχετίζεται με σχεδόν όλους τους πελάτες του υπαλλήλου “Employee-29”, οι οποίοι έχουν υψηλή τιμή σοβαρότητας.

γεγονότων που αφορούν στον υπάλληλο “Employee-29” και τον πελάτη “Client-3”. Προφανώς, υπάρχει μία συνεχής δραστηριότητα μεταξύ των δύο οντοτήτων. Ωστόσο, η παρατηρούμενη δραστηριότητα, σύμφωνα με τους ελεγκτές, ταιριάζει περισσότερο σε μία δραστηριότητα παρακολούθησης παρά σε ένα πρότυπο απάτης. Η υπόθεση αυτή ενισχύεται και από την Εικόνα 19.11ii, η οποία αναπαριστά όλες τις καταγεγραμμένες ενέργειες για το επιλεγμένο ζεύγος οντοτήτων. Ειδικότερα, μόνο μία ενέργεια πραγματοποιείται από τον υπάλληλο “Employee-29” προς τον πελάτη “Client-3” και σύμφωνα με τους ελεγκτές, η συγκεκριμένη ενέργεια αποτελεί μέρος της διαδικασίας παρακολούθησης. Μελετώντας, με παρόμοιο τρόπο, τα διαγράμματα χρόνου και περιοδικότητας για τον πελάτη “Client-4”, οι ελεγκτές αποφάνθηκαν ότι, επίσης, δεν υπάρχουν ενδείξεις απάτης (οι καταγεγραμμένες ενέργειες ήταν, επίσης, μέρος της διαδικασίας παρακολούθησης).

Η Εικόνα 19.12 απεικονίζει έναν ακόμα υπάλληλο (“Employee-26”) με υψηλή τιμή σοβα-



(i)



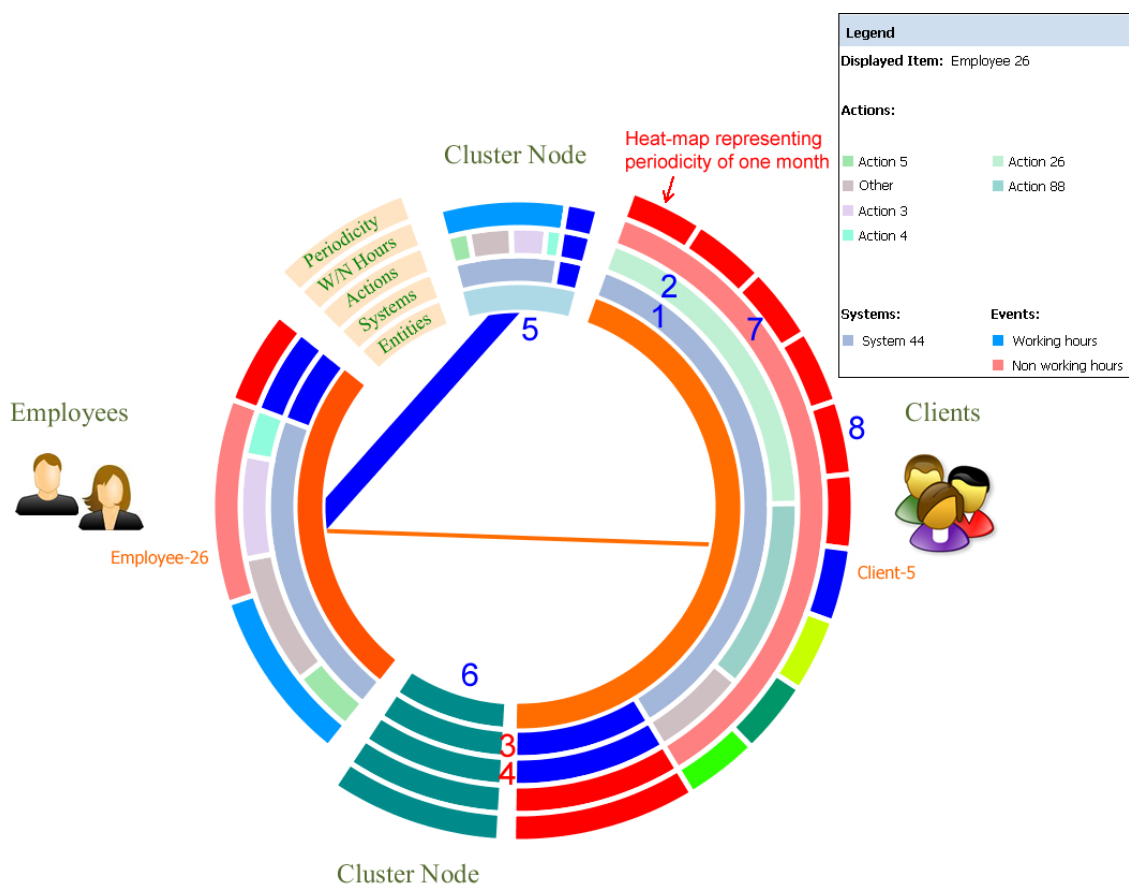
(ii)

**Σχήμα 19.11:** (i) Τα διαγράμματα χρόνου για τον υπάλληλο “Employee-40” και τον πελάτη “Client-6” αναδεικνύουν εμφανή μηνιαία δραστηριότητα. (ii) Ένα διάγραμμα που παρουσιάζει την περιοδικότητα των καταγεγραμμένων ενεργειών για τον υπάλληλο “Employee-40” και τον πελάτη “Client-6”.

ρότητας. Ο υπάλληλος “Employee-26” σχετιζόταν μόνο με τον πελάτη “Client-5”, με περισσότερα από 10 γεγονότα, χρησιμοποιώντας ένα μοναδικό σύστημα, στο οποίο ο υπάλληλος ήταν “τακτικός” χρήστης και εκτελώντας συνηθισμένες ενέργειες (βλ. τα σημεία αναφοράς 1, 2 3 και 4 της Εικόνας 19.12). Και πάλι, όλοι οι υπόλοιποι πελάτες που σχετίζονται με τον υπάλληλο “Employee-26”, τοποθετούνται στην ομάδα χαμηλής σοβαρότητας, ενώ κανένας πελάτης δεν έχει τοποθετηθεί στην ομάδα μεσαίας σοβαρότητας (βλ. τα σημεία αναφοράς 5 και 6 της Εικόνας 19.12, αντίστοιχα). Επίσης, ο πελάτης “Client-5” σχετιζόταν μόνο με τον υπάλληλο “Employee-26” (επιλέγοντας τον κόμβο που αντιστοιχεί στον πελάτη “Client-5”, κανένας άλλος υπάλληλος δεν προστέθηκε στην οπτικοποίηση). Ωστόσο, σχεδόν όλα τα γεγονότα που αφορούν στις δύο οντότητες έχουν καταγραφεί εκτός των ωρών εργασίας (βλ. το σημείο αναφοράς 7 της Εικόνας 19.12). Η οπτικοποίηση της Εικόνας 19.12 φανερώνει μία ισχυρή περιοδική δραστηριότητα (βλ. το σημείο αναφοράς 8 που αντιστοιχεί στους θερμοκούς χάρτες στο τελευταίο επίπεδο του τμήματος της οπτικοποίησης που αναπαριστά τον

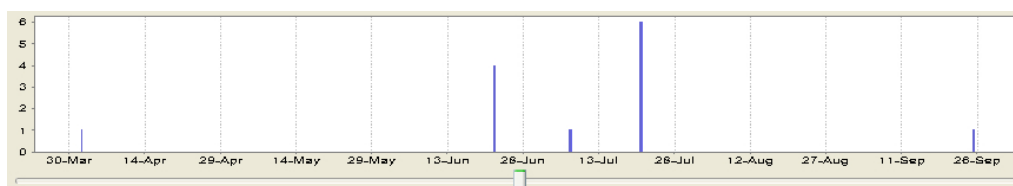


πελάτη “Client-5”). Στο επόμενο βήμα, εξετάζεται το διάγραμμα χρόνου για τις δύο οντότητες (βλ. Εικόνα 19.13). Κάποιος μπορεί να διακρίνει μηνιαία περιοδική δραστηριότητα μεταξύ των μηνών Ιουνίου και Σεπτεμβρίου (σ.σ. οι ακριβείς ημερομηνίες είναι 22/6, 21/7, 25/9), ακόμη και με μικρά κενά (δεν υπάρχουν καταχωρήσεις το μήνα Αύγουστο) και κάποια δεδομένα θορύβου (σ.σ. 7/7). Οι ελεγκτές που διερεύνησαν τη συγκεκριμένη περίπτωση διαπίστωσαν ότι δεν υπήρχαν ενδείξεις απάτης, κυρίως λόγω των ενεργειών που πραγματοποιήθηκαν από τον υπάλληλο, οι οποίες αποτελούσαν και πάλι συνηθισμένες ενέργειες παρακολούθησης. Ο πελάτης “Client-5” είχε αναγνωριστεί από ένα σύστημα διαχείρισης απάτης και επομένως, ο υπάλληλος “Employee-26” παρακολουθούσε τη δραστηριότητά του ανά τακτά χρονικά διαστήματα.



**Σχήμα 19.12:** Μία οπτικοποίηση που παρουσιάζει τη δραστηριότητα του υπαλλήλου “Employee-26”.

Η Εικόνα 19.14 παρουσιάζει τη “φανταστική” περίπτωση απάτης που προστέθηκε από τους ελεγκτές. Σε αυτό το σενάριο, ο υπάλληλος “Employee-40” σχετίζεται με τον πελάτη “Client-6”, χρησιμοποιώντας δύο εταιρικά συστήματα, στα οποία είναι “τακτικός” χρήστης

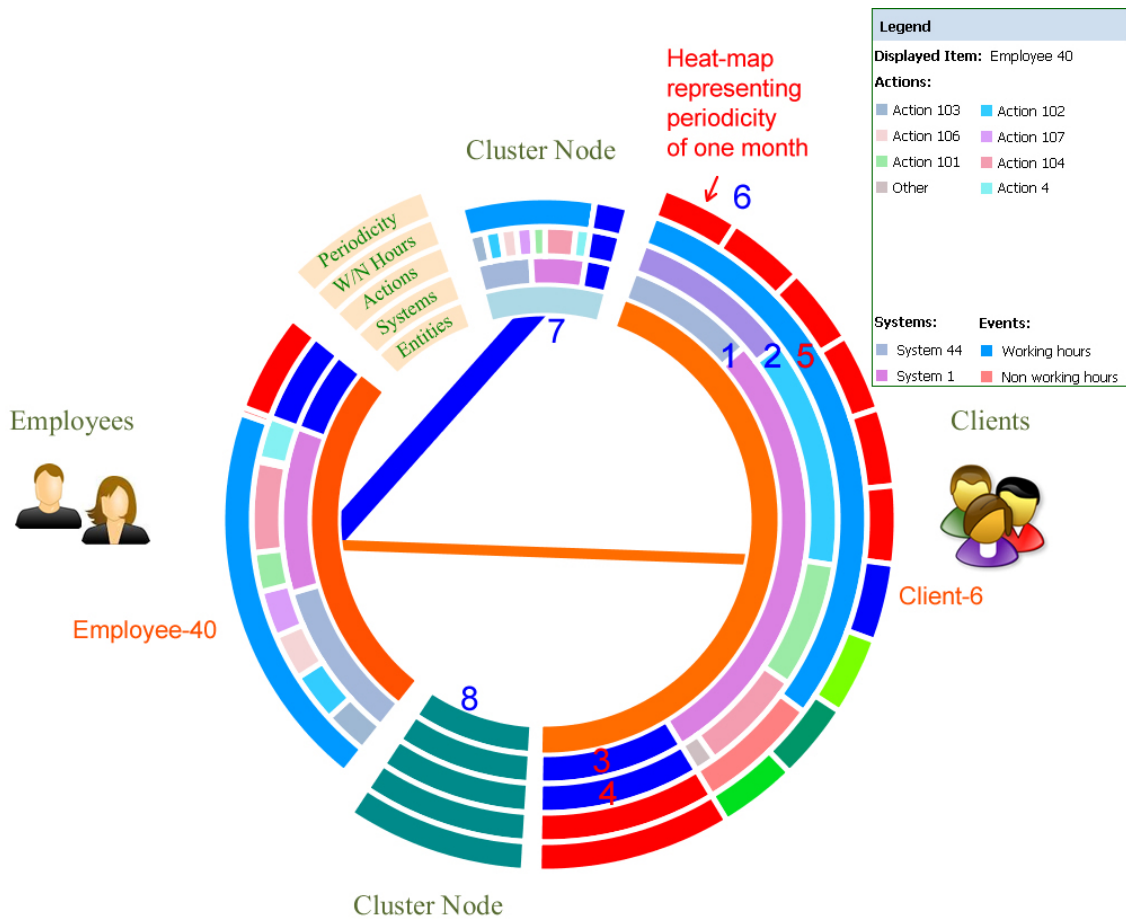


**Σχήμα 19.13:** Το αντίστοιχο διάγραμμα χρόνου για τον υπάλληλο “Employee-26” και τον πελάτη “Client-5”.

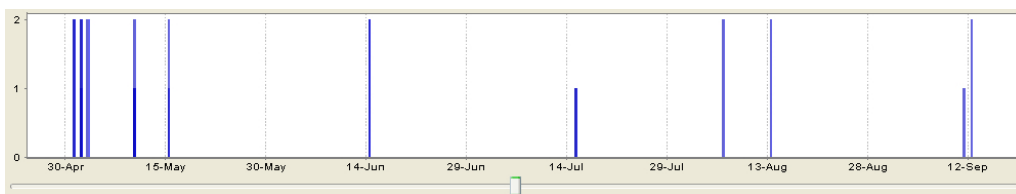
και εκτελώντας συνηθισμένες ενέργειες (βλ. τα σημεία αναφοράς 1, 2, 3, 4 της Εικόνας 19.14). Όλα τα γεγονότα συνέβησαν εντός του ωραρίου εργασίας (βλ. το σημείο αναφοράς 5 της Εικόνας 19.14). Ωστόσο, η σειρά γεγονότων του υπαλλήλου εμφανίζει μεγάλη ομοιότητα με έξι χρονοσειρές-πρότυπα (βλ. το σημείο αναφοράς 6 της Εικόνας 19.14). Επίσης, ο πελάτης “Client-6” δε σχετίζεται με κανένα άλλο υπάλληλο της εταιρείας (όταν επιλέχθηκε κανένας νέος πελάτης δεν προστέθηκε στην οπτικοποίηση). Και πάλι, όλοι οι υπόλοιποι πελάτες που σχετίζονται με τον υπάλληλο “Employee-40”, τοποθετούνται στην ομάδα χαμηλής σοβαρότητας, ενώ κανένας πελάτης δεν τοποθετήθηκε στην ομάδα μεσαίας σοβαρότητας (βλ. τα σημεία αναφοράς 7 και 8 της Εικόνας 19.14, αντίστοιχα). Παρόλο που η συγκεκριμένη οπτικοποίηση μοιάζει αρκετά με εκείνη του υπαλλήλου “Employee-26” (βλ. Εικόνα 19.12), τα διαγράμματα χρόνου (βλ. Εικόνα 19.15) και περιοδικότητας (βλ. Εικόνα 19.16) εξηγούν το λόγο για τον οποίο η περίπτωση αυτή θεωρείται ως ύποπτη απάτης. Οι πρώτες υποψίες, σύμφωνα με τους ελεγκτές, προκαλούνται από το γεγονός ότι υπάρχει δραστηριότητα μεταξύ των δύο οντοτήτων, η οποία συναντάται σε δύο εταιρικά συστήματα (οι ελεγκτές λαμβάνουν, επίσης, υπόψη τον τύπο αυτών των συστημάτων, μία πληροφορία που όμως δεν μας κοινοποιήθηκε με πολλές λεπτομέρειες).

Οι ελεγκτές μας εξήγησαν ότι το διάγραμμα χρόνου (βλ. Εικόνα 19.15) ταιριάζει αρκετά σε ένα σενάριο απάτης, σύμφωνα με το οποίο υπάρχει δραστηριότητα στην αρχή (από το μήνα Απρίλιο έως το μήνα Μάιο) χωρίς συγκεκριμένη περιοδικότητα και στη συνέχεια, εμφανίζεται περιοδική δραστηριότητα (από το μήνα Μάιο έως το μήνα Σεπτέμβριο). Στο πρώτο χρονικό διάστημα, ο επίδοξος παραβάτης προσπαθεί να προγραμματίσει και να οργανώσει την απάτη, εκτελώντας μία σειρά ενεργειών. Όταν η απάτη “οργανωθεί”, είναι αναγκαίες μόνο περιοδικές ενέργειες. Ένα άλλο ύποπτο γεγονός, σε αυτή την περίπτωση, είναι ότι τα καταγεγραμμένα γεγονότα από το Μάιο έως το Σεπτέμβριο συνέβησαν κοντά στην ίδια ημερομηνία κάθε μήνα (από τη 10η έως τη 15η ημέρα). Προφανώς, υπάρχουν κάποια δεδομένα που προκαλούν “θόρυβο” και πρέπει να εξαιρεθούν, προκειμένου να αποκαλυφθεί το μοτίβο της απάτης. Αυτά μπορεί να έχουν προκληθεί είτε με σκοπό την κάλυψη της απάτης ή αποτελούσαν μέρος των καθηκόντων του υπαλλήλου.

Η παραπάνω υπόθεση ενισχύεται από το διάγραμμα της Εικόνας 19.16 που αποκαλύ-



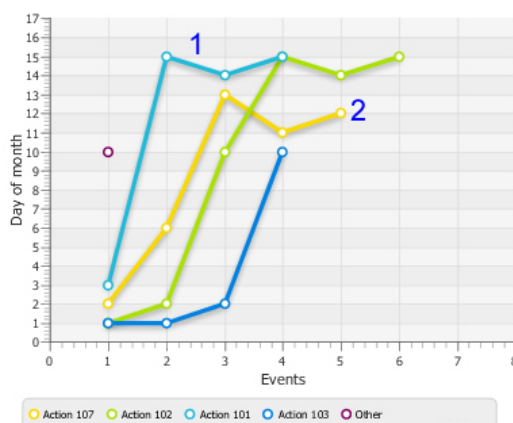
**Σχήμα 19.14:** Το αντίστοιχο διάγραμμα χρόνου για τον υπάλληλο “Employee-40”. Η περίπτωση αυτή αντιστοιχεί σε ένα “φανταστικό” σενάριο απάτης.



**Σχήμα 19.15:** Το διάγραμμα χρόνου για τον υπάλληλο Employee-40 και τον πελάτη Client-6 αναδεικνύει προφανή μηνιαία δραστηριότητα.

ππει την περιοδική εμφάνιση κάθε ενέργειας που εκτελέστηκε. Για παράδειγμα, η ενέργεια “Action 101” (βλ. το σημείο αναφοράς 1) εμφανίζει περιοδικότητα γύρω στη 15η ημέρα του μήνα, από τη δεύτερη φορά εμφάνισής της και αργότερα. Επίσης, η ενέργεια “Action 107” (βλ. το σημείο αναφοράς 2) φαίνεται να έχει περιοδικότητα γύρω στη 11η έως τη 13η ημέρα του

μήνα, από την τρίτη φορά εμφάνισης της και αργότερα. Ειδικότερα, η συντριπτική πλειοψηφία των γεγονότων καταγράφονται μεταξύ της 10ης και 15ης ημέρας του μήνα. Θα μπορούσαμε να είμαστε πιο σίγουροι για το αν η συγκεκριμένη υπόθεση αποτελεί όντως απάτη, αν γνωρίζαμε ακριβώς τον κύκλο χρέωσης του λογαριασμού του πελάτη.



**Σχήμα 19.16:** Ένα διάγραμμα που αποκαλύπτει την περιοδική εμφάνιση κάθε ενέργειας που εκτελέστηκε από τον υπάλληλο “Employee-40” προς τον πελάτη “Client-6”.

Κατά παρόμοιο τρόπο, διερευνήθηκαν και τα υπόλοιπα καρέ που οπτικοποιούν υπαλλήλους με υψηλή τιμή σοβαρότητας. Συγκεκριμένα, δόθηκε περισσότερη προσοχή πρώτα στα 17 καρέ που περιείχαν περιοδική δραστηριότητα. Δεδομένου ότι τα σύνολα δεδομένων που εξετάσαμε σε αυτή την ανάλυση περίπτωσης ήταν ευαίσθητα, δεν μας κοινοποιήθηκαν πολλές λεπτομέρειες σχετικά με τα τελικά αποτελέσματα της έρευνας. Ευτυχώς για την εταιρεία, η μόνη πραγματική περίπτωση απάτης σε αυτό το σύνολο δεδομένων, ήταν εκείνη που προστέθηκε από τους ελεγκτές. Ωστόσο, οι ελεγκτές δεν είχαν εντοπίσει όλες αυτές τις περιπτώσεις κατά τη μη αυτοματοποιημένη εξέταση των δεδομένων.

## 20 Συμπεράσματα

Στην παρούσα διδακτορική διατριβή, εστίασαμε σε δύο διαφορετικές περιοχές της απεικόνισης γραφημάτων, δηλαδή στο πρόβλημα της μεγιστοποίησης της γωνιακής ευκρίνειας τομών μιας απεικόνισης ενός γραφήματος και σε εφαρμογές της απεικόνισης γραφημάτων στην ανίχνευση απάτης. Για το πρώτο πρόβλημα, μελετήσαμε κυρίως προβλήματα σχετικά με τις απεικονίσεις ορθογώνιων τομών, δηλαδή απεικονίσεις στις οποίες κάθε ζεύγος τεμνόμενων ακμών σχηματίζει ορθή γωνία. Εισάγαμε, επίσης, ένα σχετικό πρόβλημα, το πρόβλημα της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας μιας απεικόνισης ενός γραφήματος, σύμφωνα με το οποίο μεγιστοποιείται το ελάχιστο της γωνιακής ευκρίνειας και της γωνιακής ευκρίνειας τομών. Για το πρόβλημα της ανίχνευσης απάτης, παρουσιάσαμε ένα συνδυασμό μεθόδων με οπτικοποιήσεις που αποσκοπούν στην ανίχνευση εταιρικής απάτης σε εταιρικά συστήματα, στα οποία εμπλέκονται ζεύγη οντοτήτων (υπάλληλος και πελάτης), όπως για παράδειγμα συστήματα χρέωσης, ανανέωσης συνδρομών και ούτω καθεξής. Παρουσιάσαμε δύο προσεγγίσεις που έχουν ως στόχο την ανίχνευση περιοδικών γεγονότων, τα οποία μπορεί να υποκρύπτουν εταιρική απάτη και ένα ολοκληρωμένο σύστημα ανίχνευσης εταιρικής απάτης με οπτικοποιήσεις με βάση πρότυπα απάτης που προτάθηκαν από τους εσωτερικούς ελεγκτές ενός οργανισμού τηλεπικοινωνιών.

Στο Κεφάλαιο 13, αποδείξαμε ότι είναι  $\mathcal{NP}$ -δύσκολο να καθορίσουμε εάν ένα γράφημα επιδέχεται μία γεωμετρική RAC απεικόνιση. Οι Didimo et al. [DEL11] απέδειξαν ότι είναι πάντοτε εφικτό να κατασκευαστεί μία RAC απεικόνιση ενός δοθέντος γραφήματος με το πολύ τρία σημεία καμπής ανά ακμή. Εάν επιτρέψουμε ένα ή δύο σημεία καμπής ανά ακμή, το πρόβλημα παραμένει  $\mathcal{NP}$ -δύσκολο; Η ίδια ερώτηση μπορεί να τεθεί και στην περίπτωση που το γράφημα εισόδου έχει  $n$  κορυφές και ακριβώς  $4n - 10$  ακμές, δηλαδή εάν το πρόβλημα της αναγνώρισης γραφημάτων που ανήκουν στην κλάση των μεγιστικώς πυκνών RAC γραφημάτων είναι, επίσης,  $\mathcal{NP}$ -δύσκολο.

Στο Κεφάλαιο 14, εισάγαμε και εξετάσαμε γεωμετρικές RAC ταυτόχρονες απεικονίσεις. Ιδιαίτερο ενδιαφέρον παρουσιάζει το πρόβλημα της αναγνώρισης και άλλων μη τετριμμένων κλάσεων γραφημάτων, εκτός από ένα ταίριασμα και είτε ένα μονοπάτι είτε έναν κύκλο, οι οποίες επιδέχονται GRacSim απεικονίσεις. Επίσης, θεωρήσαμε μόνο γεωμετρικές RAC

ταυτόχρονες απεικονίσεις. Για τις κλάσεις όπου οι GRacSim απεικονίσεις δεν είναι εφικτές, θα ήταν ενδιαφέρον να εξετάσουμε απεικονίσεις με σημεία καμπής ή να χαλαρώσουμε την απαίτηση για βέλτιστη γωνιακή ευκρίνεια τομών στις παραγόμενες απεικονίσεις. Αρκετά συναφές με το πρόβλημα της GRacSim απεικόνισης είναι το πρόβλημα της απεικόνισης δύο (ή περισσότερων) γραφημάτων του ίδιου συνόλου κορυφών, έτσι ώστε κάθε γράφημα να απεικονίζεται RAC (σ.σ. μόνο ακμές διαφορετικών γραφημάτων μπορεί να εισάγουν τομές υπό μη-ορθές γωνίες). Σημειώνεται ότι η κλάση των γραφημάτων που επιδέχονται τέτοιες απεικονίσεις περιέχει την κλάση των γραφημάτων για τα οποία μία ταυτόχρονη απεικόνιση είναι εφικτή. Τέλος, ενδιαφέρον παρουσιάζει το πρόβλημα της παραγωγής μιας πιο ελκυστικής GDual-GRacSim απεικόνισης ενός εξωεπίπεδου γραφήματος και του δυϊκού του. Σε αυτή την περίπτωση, θα μπορούσαμε, επίσης, να μελετήσουμε το απαιτούμενο εμβαδόν και να παρέχουμε ένα χαρακτηρισμό των κλάσεων, εκτός από τα εξωεπίπεδα γραφήματα, οι οποίες επιδέχονται GDual-GRacSim απεικονίσεις.

Στο Κεφάλαιο 15, μελετήσαμε το πρόβλημα της μεγιστοποίησης της συνολικής γωνιακής ευκρίνειας μιας απεικόνισης. Θα ήταν ενδιαφέρον να προσπαθήσουμε να αναγνωρίσουμε και άλλες κλάσεις γραφημάτων που επιδέχονται βέλτιστες απεικονίσεις. Ακόμα και η περίπτωση των επίπεδων γραφημάτων παρουσιάζει ενδιαφέρον, καθώς επιτρέποντας μερικές τομές ακμών (έστω υπό μεγάλες γωνίες) μπορεί να βελτιωθεί η γωνιακή ευκρίνεια και επομένως, και η συνολική γωνιακή ευκρίνεια μιας απεικόνισης. Στην εργασία [vK10], δίνεται μία κλάση γραφημάτων που ικανοποιεί αυτή την ιδιότητα.

Στο Κεφάλαιο 17, παρουσιάσαμε μία μετρική και τεχνικές οπτικοποίησης για την ανίχνευση περιοδικών γεγονότων σε σειριακά δεδομένα. Θα ήταν ενδιαφέρον να μελετηθούν εναλλακτικές μετρικές για τον υπολογισμό του βαθμού εμπιστοσύνης που χρησιμοποιείται για τον ακριβέστερο υπολογισμό της περιοδικότητας. Αυτό μπορεί να επηρεάσει τόσο την ποιότητα όσο και τον τύπο των παραγόμενων οπτικοποιήσεων. Η αναγνώριση ομάδων (αντί υπαλλήλων) που εμφανίζουν “ύποπτη” συμπεριφορά είναι, επίσης, ενδιαφέρουσα. Κλασικές τεχνικές ομαδοποίησης από την Απεικόνιση Γραφημάτων μπορεί να είναι, επίσης, χρήσιμες για την παραγωγή τέτοιων οπτικοποιήσεων.

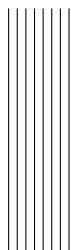
Στο Κεφάλαιο 18, παρουσιάσαμε ένα σύστημα για την ανίχνευση περιοδικής δραστηριότητας και εταιρικής απάτης σε εταιρικά συστήματα. Θα ήταν ενδιαφέρον να ενσωματωθούν περισσότεροι παράγοντες με σκοπό τον υπολογισμό ακριβέστερων τιμών κατά τη διαδικασία κατάταξης των υπαλλήλων και των πελατών, καθώς και τον ορισμό κατάλληλων συναρτήσεων από τον ελεγκτή μέσω του γραφικού περιβάλλοντος του συστήματος. Και πάλι, θα ήταν ενδιαφέρουσα η αναζήτηση ομάδων συνεργαζόμενων υπαλλήλων και πελατών.

Στο Κεφάλαιο 19, παρουσιάσαμε ένα ολοκληρωμένο σύστημα οπτικοποίησης, με στόχο την ανίχνευση μοτίβων που μπορεί να υποκρύπτουν εταιρική απάτη, μέσω ενός συνδυα-

σμού αναγνώρισης προτύπων και οπτικοποιήσεων. Η εργασία μας αφήνει αρκετές πτυχές για μελλοντική εργασία όπως την ενσωμάτωση περισσότερων προτύπων ανίχνευσης απάτης, τη χρήση στατιστικών μεθόδων και την επέκταση του συστήματος, έτσι ώστε να μπορεί να εντοπίσει πιο πολύπλοκα σχήματα απάτης (απάτη από την πλευρά των πελατών/συνδρομητών, κτλ) σε περισσότερα εταιρικά συστήματα.







## List of Publications

### Journal Publications

1. Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, and Antonios Symvonis. Geometric RAC simultaneous drawings of graphs. *Journal of Graph Algorithms and Applications*, 17(1):11–34, 2013.
2. Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. Maximizing the total resolution of graphs. *The Computer Journal*, 56(7):887–900, 2013.
3. Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. The straight-line RAC drawing problem is NP-hard. *Journal of Graph Algorithms and Applications*, 16(2):569–597, 2012

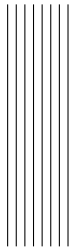
### International Conference Publications (refereed)

1. Evmorfia N. Argyriou, Antonios Symvonis, and Vassilis Vassiliou. A fraud detection visualization system utilizing radial drawings and heat-maps. In *Proceedings of the fifth International Conference on Information Visualization Theory and Applications (IVAPP '14)*, to appear, 2014.
2. Evmorfia N. Argyriou, Aikaterini Sotiraki, and Antonios Symvonis. Occupational fraud detection through visualization. In *Proceeding of the IEEE Intelligence and Security Informatics (ISI '13)*, pages 4–7, 2013.
3. Evmorfia N. Argyriou and Antonios Symvonis. Detecting periodicity in serial data through visualization. In *Proceedings of the ninth International Symposium on Visual Computing (ISVC '12)*, volume 7432 of LNCS, pages 295–304, 2012.
4. Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, and Antonios Symvonis. Geometric RAC simultaneous drawings of graphs. In Joachim Gudmundsson, Julián Mestre, Taso Viglas, editors, *Proceedings of the 18th Annual International Computing and Combinatorics Conference (COCOON '12)*, volume 7434 of LNCS, pages 287–298, 2012.

5. Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, and Antonios Symvonis. Combining problems on RAC drawings and simultaneous graph drawings. In Marc J. van Kreveld and Bettina Speckmann, editors, *Proceedings of the 19th International Symposium on Graph Drawing (GD '11)*, volume 7034 of LNCS, pages 433–434, 2011.
6. Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. The straight-line RAC drawing problem is NP-hard. In Ivana Cerná, Tibor Gyimóthy, Juraj Hromkovic, Keith G. Jeffery, Rastislav Královic, Marko Vukolic and Stefan Wolf, editors, *Proceedings of the 37th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM '11)*, volume 6543 of LNCS, pages 74–85, 2011.
7. Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. Maximizing the total resolution of graphs. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceedings 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of LNCS, pages 62–67, 2010.

#### **Other Publications (omitted in this thesis)**

1. Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, Antonios Symvonis: On Metro-Line Crossing Minimization. *Journal of Graph Algorithms and Applications (JGAA): Special Issue on Selected Papers from the 16th International Symposium on Graph Drawing*, 14(1):75-96, 2010.
2. Theologos Athanaselis, Stelios Bakamidis, Ioannis Dologlou, Evmorfia N. Argyriou, Antonios Symvonis: Making assistive reading tools user friendly: a new platform for Greek dyslexic students empowered by automatic speech recognition, *Multimedia Tools and Applications*, pages 1-19, Springer US, 2012.
3. Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, Antonios Symvonis: Two Polynomial Time Algorithms for the Metro-Line Crossing Minimization Problem. In Ioannis Tollis and Maurizio Patrignani, editors, *Proceedings of the 16th International Symposium on Graph Drawing (GD '08)*, volume 5417 of LNCS, pages 336-347, 2008.
4. Theologos Athanaselis, Stelios Bakamidis, Ioannis Dologlou, Evmorfia N. Argyriou, Antonios Symvonis: Incorporating Speech Recognition Engine into an Intelligent Assistive Reading System for Dyslexic Students, In *Proceedings of the 12th Annual Conference of the International Speech Communication Association (INTERSPEECH '11)*, pages 2993-2996, 2011.



## Bibliography

- [AA12] Conan C. Albrecht and W. Steve Albrecht. Strategic fraud detection: A technology-based model. SeminarPapers, Brigham Young University, Provo, UT USA, 2012.
- [ABD<sup>+</sup>12] Patrizio Angelini, Giuseppe Battista, Walter Didimo, Fabrizio Frati, Seok-Hee Hong, Michael Kaufmann, Giuseppe Liotta, and Anna Lubiw. Large angle crossing drawings of planar graphs in subquadratic area. *Computational Geometry*, 7579:200–209, 2012.
- [ABKS11] Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, and Antonios Symvonis. Combining problems on RAC drawings and simultaneous graph drawings. In Marc J. van Kreveld and Bettina Speckmann, editors, *Proceedings of the 19th International Symposium on Graph Drawing (GD '11)*, volume 7034 of LNCS, pages 433–434, Berlin, Heidelberg, 2011. Springer-Verlag.
- [ABKS12] Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, and Antonios Symvonis. Geometric RAC simultaneous drawings of graphs. In Joachim Gudmundsson, Julián Mestre, and Taso Viglas, editors, *Proceedings of the 18th Annual International Computing and Combinatorics Conference (COCOON '12)*, volume 7434 of LNCS, pages 287–298, 2012.
- [ABKS13] Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, and Antonios Symvonis. Geometric RAC simultaneous drawings of graphs. *Journal of Graph Algorithms and Applications*, 17(1):11–34, 2013.
- [ABM<sup>+</sup>07] Wolfgang Aigner, Alessio Bertone, Silvia Miksch, Christian Tominski, and Hei-drun Schumann. Towards a conceptual framework for visual analytics of time and time-oriented data. In *Proceedings of the 39th Conference on Winter Simulation: 40 years! The best is yet to come (WSC '07)*, pages 721–729, 2007.
- [ABS10] Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. Maximizing the total resolution of graphs. In Ulrik Brandes and Sabine Cornelsen, editors,

- Proceedings 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of *LNCS*, pages 62–67, Berlin, Heidelberg, 2010. Springer-Verlag.
- [ABS11] Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. The straight-line RAC drawing problem is NP-hard. In Ivana Cerná, Tibor Gyimóthy, Juraj Hromkovic, Keith G. Jeffery, Rastislav Královic, Marko Vukolic, and Stefan Wolf, editors, *Proceedings of the 37th International Conference on Current Trends in Theory and Practice of Computer Science, (SOFSEM '11)*, volume 6543 of *LNCS*, pages 74–85, 2011.
- [ABS12] Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. The straight-line RAC drawing problem is NP-hard. *Journal of Graph Algorithms and Applications*, 16(2):569–597, 2012.
- [ABS13] Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. Maximizing the total resolution of graphs. *The Computer Journal*, 56(7):887–900, 2013.
- [ACD<sup>+</sup>09] Patrizio Angelini, Luca Cittadini, Giuseppe Di Battista, Walter Didimo, Fabrizio Frati, Michael Kaufmann, and Antonios Symvonis. On the perspectives opened by right angle crossing drawings. In David Eppstein and Emden R. Gansner, editors, *Proceedings of the 17th International Symposium on Graph Drawing (GD '09)*, volume 5849 of *LNCS*, pages 21–32, Berlin, Heidelberg, 2009. Springer-Verlag.
- [AF12] Mohd Anwar and Philip W. L. Fong. A visualization tool for evaluating access control policies in facebook-style social network systems. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing (SAC '12)*, pages 1443–1450, New York, NY, USA, 2012. ACM.
- [AFK<sup>+</sup>10] Karin Arikushi, Radoslav Fulek, Balaazs Keszegh, Filip Moric, and Csaba Tóth. Graphs that admit right angle crossing drawings. In Dimitrios M. Thilikos, editor, *Proceedings 36th International Workshop on Graph Theoretic Concepts in Computer Science (WG '10)*, volume 6410 of *LNCS*, pages 135–146, 2010.
- [AFT10] Eyal Ackerman, Radoslav Fulek, and Csaba D. Tóth. On the size of graphs that admit polyline drawings with few bends and crossing angles. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceedings of the 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of *LNCS*, pages 1–12, Berlin, Heidelberg, 2010. Springer-Verlag.

- [AGKN10] Patrizio Angelini, Markus Geyer, Michael Kaufmann, and Daniel Neuwirth. On a tree and a path with no geometric simultaneous embedding. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceedings of the 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of *LNCS*, pages 38–49, Berlin, Heidelberg, 2010. Springer-Verlag.
- [ALC<sup>+</sup>05] Kulsoom Abdullah, Chris Lee, Gregory Conti, John A. Copeland, and John Stasko. Ids rainstorm: Visualizing ids alarms. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, pages 1–10. IEEE Computer Society, 2005.
- [AoCFE12] Association of Certified Fraud Examiners. *2012 Report to the Nation on Occupational Fraud and Abuse*, 2012.
- [AP61] Louis Auslander and S. V. Parter. On embedding graphs in the sphere. *Journal of Mathematics and Mechanics*, 10(4):517–523, 1961.
- [AS12] Evmorfia N. Argyriou and Antonios Symvonis. Detecting periodicity in serial data through visualization. In *Proceedings of the ninth International Symposium on Visual Computing (ISVC '12)*, volume 7432, pages 295–304, 2012.
- [ASS13] Evmorfia N. Argyriou, Aikaterini Sotiraki, and Antonios Symvonis. Occupational fraud detection through visualization. In *Proceedings of the IEEE Intelligence and Security Informatics (ISI '13)*, pages 4–7, 2013.
- [ASV14] Evmorfia N. Argyriou, Antonios Symvonis, and Vassilis Vassiliou. A fraud detection visualization system utilizing radial drawings and heat-maps. In *Proceedings of the fifth International Conference on Information Visualization Theory and Applications (IVAPP '14)*, 2014. to appear.
- [Bac07] Christian Bachmaier. A radial adaptation of the sugiyama framework for visualizing hierarchical information. *IEEE Transactions on Visualization and Computer Graphics*, 13(3):583–594, 2007.
- [Bad64] W. Bader. Das topologische problem der gedruckten schaltung und seine losung. *Electrical Engineering (Archiv fur Elektrotechnik)*, 49(1):2–12, 1964.
- [BC00] Timothy B. Bell and Joseph V. Carcello. A decision aid for assessing the likelihood of fraudulent financial reporting. *Auditing: A Journal of Practice & Theory*, 19:169–184, 2000.

- [BCD<sup>+</sup>07] Peter Brass, Eowyn Cenek, Cristian A. Duncan, Alon Efrat, Cesim Erten, Dan P. Ismailescu, Stephen G. Kobourov, Anna Lubiw, and Joseph S.B. Mitchell. On simultaneous planar graph embeddings. *Computational Geometry: Theory and Applications*, 36(2):117–130, 2007.
- [BDF<sup>+</sup>08] Ryan Blue, Cody Dunne, Adam Fuchs, Kyle King, and Aaron Schulman. Visualizing real-time network resource usage. In *Proceedings of the fifth International Workshop on Visualization for Computer Security (VizSec '08)*, pages 119–135, Berlin, Heidelberg, 2008. Springer-Verlag.
- [Ber81] Margaret A. Bernard. On the automated drawing of graphs. In *Proceedings of the Third Caribbean Conference on Combinatorics and Computing*, page 43 – 55, 1981.
- [Ber00] François Bertault. A force-directed algorithm that preserves edge-crossing properties. *Information Processing Letters*, 74(1-2):7–13, 2000.
- [BFN85] Carlo Batini, L. Furlani, and Enrico Nardelli. What is a good diagram? A pragmatic approach. In *Proceedings of the fourth International Conference on Entity-Relationship Approach*, pages 312–319, Washington, DC, USA, 1985. IEEE Computer Society.
- [BGW03] Ulrik Brandes, Marco Gaertler, and Dorothea Wagner. Experiments on graph clustering algorithms. In Giuseppe Battista and Uri Zwick, editors, *Proceedings of the 11th Annual European Symposium on Algorithms (ESA '03)*, volume 2832 of *Lecture Notes in Computer Science*, pages 568–579. Springer Berlin Heidelberg, 2003.
- [BH02] Richard J. Bolton and David J. Hand. Statistical fraud detection: A review. *Statistical Science*, 17:2002, 2002.
- [BHL07] Enrico Bertini, Patrick Hertzog, and Denis Lalanne. Spiralview: Towards security policies assessment through visual correlation of network resources with evolution of alarms. In *Proceedings of the 2007 IEEE Symposium on Visual Analytics Science and Technology*, pages 139–146, 2007.
- [BHR96] Franz Brandenburg, Michael Himsolt, and Christoph Rohrer. An experimental comparison of force-directed and randomized graph drawing algorithms. In Franz-Josef Brandenburg, editor, *Proceedings of the third International Symposium on Graph Drawing (GD '95)*, volume 1027 of *LNCS*, pages 76–87, 1996.

- [BHvW99] Mark Bruls, Kees Huizing, and Jarke van Wijk. Squarified treemaps. In *Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization*, pages 33–42. Press, 1999.
- [BJL02] Christoph Buchheim, Michael Jünger, and Sebastian Leipert. Improving walker’s algorithm to run in linear time. In Michael T. Goodrich and Stephen G. Kobourov, editors, *Proceedings of the 10th International Symposium on Graph Drawing (GD ’02)*, volume 2528 of *LNCS*, pages 344–353, London, UK, 2002. Springer Berlin Heidelberg.
- [BK94] Therese C. Biedl and Goos Kant. A better heuristic for orthogonal graph drawings. In *Proceedings of the second European Symposium on Algorithms (ESA ’94)*, volume 855 of *LNCS*, pages 24–35, 1994.
- [BK01] Ulrik Brandes and Boris Köpf. Fast and simple horizontal coordinate assignment. In Petra Mutzel, Michael Jünger, and Sebastian Leipert, editors, *Proceedings of the ninth International Symposium on Graph Drawing (GD ’01)*, volume 2265 of *LNCS*, pages 31–44. Springer Berlin Heidelberg, 2001.
- [BKR07] Thomas Bläsius, Stephen G. Kobourov, and Ignaz Rutter. Simultaneous embedding of planar graphs. In Roberto Tamassia, editor, *Handbook of Graph Drawing and Visualization (Discrete Mathematics and Its Applications)*, pages 285–316. Chapman & Hall/CRC, 2007.
- [BKR12] Thomas Bläsius, Stephen G. Kobourov, and Ignaz Rutter. Simultaneous embedding of planar graphs. *CoRR*, abs/1204.5853, 2012.
- [BL76] Kellogg S. Booth and George S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms. *Journal of Computer and System Sciences*, 13(3):335 – 379, 1976.
- [BLME02] Jürgen Branke, Stefan Leppert, Martin Middendorf, and Peter Eades. Width-restricted layering of acyclic digraphs with consideration of dummy nodes. *Information Processing Letters*, 81(2):59 – 63, 2002.
- [Blo93] Anthony Bloesch. Aesthetic layout of generalized trees. *Software—Practice & Experience*, 23(8):817–827, 1993.
- [BM03] M. Bernard and S. Mohammed. Labeled radial drawing of data structures. In *Proceedings of the Seventh International Conference on Information Visualization (IV ’03)*, page 479, Washington, DC, USA, 2003. IEEE Computer Society.

- [BMK96] Jim Blythe, Cathleen McGrath, and David Krackhardt. The effect of graph layout on inference from social network data. In *Proceedings of the 3th International Symposium on Graph Drawing (GD '95)*, pages 40–51, London, UK, UK, 1996. Springer-Verlag.
- [BP90] Karl-Friedrich Böhringer and Frances Newbery Paulisch. Using constraints to achieve stability in automatic graph layout algorithms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '90)*, pages 43–51, New York, NY, USA, 1990. ACM.
- [Bra97] Franz-Josef Brandenburg. Graph clustering i: Circles of cliques. In Giuseppe Di Battista, editor, *Proceedings of the fifth International Symposium on Graph Drawing (GD '97)*, volume 1353 of *LNCS*, pages 158–168. Springer Berlin Heidelberg, 1997.
- [Bra01] Ulrik Brandes. Drawing on physical analogies. In Michael Kaufmann and Dorothea Wagner, editors, *Drawing Graphs*, volume 2025 of *LNCS*, pages 71–86, 2001.
- [Bra09] Ulrik Brandes. The left-right planarity test. *Manuscript submitted for publication*, 2009.
- [BS90] Bonnie Berger and Peter W. Shor. Approximation algorithms for the maximum acyclic subgraph problem. In *Proceedings of the first Annual ACM-SIAM Symposium on Discrete algorithms (SODA '90)*, pages 236–243, Philadelphia, PA, USA, 1990. Society for Industrial and Applied Mathematics.
- [BS93] Graham Brightwell and Edward R. Scheinerman. Representations of planar graphs. *SIAM Journal Discrete Mathematics*, 6(2):214–229, 1993.
- [BS06] Ulrik Brandes and Barbara Schlieper. Angle and distance constraints on tree drawings. In Michael Kaufmann and Dorothea Wagner, editors, *Proceedings of the 14th International Symposium on Graph Drawing (GD '06)*, volume 4372 of *LNCS*, pages 54–65, Berlin, Heidelberg, 2006. Springer-Verlag.
- [BST00] Ulrik Brandes, Galina Shubina, and Roberto Tamassia. Improving angular resolution in visualizations of geographic networks. In Willem Cornelis Leeuw and Robert Liere, editors, *Proceedings of the second Eurographics/IEEE TCVG Symposium on Visualization (VisSym '00)*, Eurographics, pages 23–32. Springer Vienna, 2000.



- [BSW02] Benjamin B. Bederson, Ben Shneiderman, and Martin Wattenberg. Ordered and quantum treemaps: Making effective use of 2d space to display hierarchies. *ACM Transactions on Graphics (TOG)*, 21(4):833–854, 2002.
- [BT96] Giuseppe Di Battista and Roberto Tamassia. On-line planarity testing. *SIAM Journal on Computing*, 25(5):956–997, 1996.
- [BT98] Stina Bridgeman and Roberto Tamassia. Difference metrics for interactive orthogonal graph drawing algorithms. In Sue H. Whitesides, editor, *Proceedings of the sixth International Symposium on Graph Drawing (GD '98)*, volume 1547 of LNCS. Springer Berlin Heidelberg, 1998.
- [BT04] Imre Bárány and Norihide Tokushige. The minimum area of convex lattice n-gons. *Combinatorica*, 24(2), 2004.
- [BTI07] David Borland and Russell M. Taylor II. Rainbow color map (still) considered harmful. *IEEE Computer Graphics and Applications*, 27(2):14–17, 2007.
- [BTT84] Carlo Batini, Maurizio Talamo, and Roberto Tamassia. Computer aided layout of entity relationship diagrams. *Journal of Systems and Software*, 4(2–3):163 – 173, 1984.
- [BW98] Ulrik Brandes and Dorothea Wagner. Using graph layout to visualize train interconnection data. In *Proceedings of the sixth International Symposium on Graph Drawing (GD '98)*, pages 44–56, London, UK, UK, 1998. Springer-Verlag.
- [Cay57] Arthur Cayley. XXVIII On the theory of the analytical forms called trees. *Philosophical Magazine Series 4*, 13(85):172–176, 1857.
- [CCG<sup>+</sup>12] Roman Chernobelskiy, Kathryn I. Cunningham, Michael T. Goodrich, Stephen G. Kobourov, and Lowell Trott. Force-directed lombardi-style graph drawing. In Marc J. van Kreveld and Bettina Speckmann, editors, *Proceedings of the 19th International Symposium on Graph Drawing (GD '11)*, LNCS, pages 320–331, Berlin, Heidelberg, 2012. Springer-Verlag.
- [CDGK99] C. C. Cheng, Christian A. Duncan, Michael T. Goodrich, and Stephen G. Kobourov. Drawing planar graphs with circular arcs. *Discrete and Computational Geometry*, 25:117–126, 1999.
- [CEBFK09] Justin Cappos, Alejandro Estrella-Balderrama, Joseph Fowler, and Stephen G. Kobourov. Simultaneous graph embedding with bends and circular arcs. *Computational Geometry: Theory and Applications*, 42(2):173–182, 2009.

- [CF07] Amanda Cox and Hannah Fairfield. The health of the car, van, suv, and truck market (february 25, 2007). *The New York Times*, 11, 2007.
- [CFPS99] Philip K. Chan, Wei Fan, Andreas L. Prodromidis, and Salvatore J. Stolfo. Distributed data mining in credit card fraud detection. *IEEE Intelligent Systems*, 14(6):67–74, 1999.
- [CIL<sup>+</sup>10] Matthew Chu, Kyle Ingols, Richard Lippmann, Seth Webster, and Stephen Boyer. Visualizing attack graphs, reachability, and trust relationships with navigator. In *Proceedings of the seventh International Symposium on Visualization for Cyber Security (VizSec '10)*, pages 22–33, New York, NY, USA, 2010. ACM.
- [CK98] John V. Carlis and Joseph A. Konstan. Interactive visualization of serial periodic data. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, pages 29–38, 1998.
- [CK12] Sabine Cornelsen and Andreas Karrenbauer. Accelerated bend minimization. *Journal of Graph Algorithms Applications*, 16(3):635–650, 2012.
- [CLG<sup>+</sup>08] Remco Chang, Alvin Lee, Mohammad Ghoniem, Robert Kosara, and William Ribarsky. Scalable and interactive visual analysis of financial wire transactions for fraud detection. *Information Visualization*, 7(1):63–76, 2008.
- [CNA085] Norishige Chiba, Takao Nishizeki, Shigenobu Abe, and Takao Ozawa. A linear algorithm for embedding planar graphs using pq-trees. *Journal of Computer and System Sciences*, 30(1):54 – 76, 1985.
- [Con07] Greg Conti. *Security Data Visualization*. No Starch Press, San Francisco, CA, USA, 2007.
- [CPM<sup>+</sup>98] Ed H. Chi, James Pitkow, Jock Mackinlay, Peter Pirolli, Rich Gossweiler, and Stuart K. Card. Visualizing the evolution of web ecologies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '98)*, pages 400–407. ACM Press/Addison-Wesley Publishing Co., 1998.
- [CPP00] Ed H. Chi, Peter Pirolli, and James Pitkow. The scent of a site: a system for analyzing and predicting information scent, usage, and usability of a web site. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems (CHI '00)*, pages 161–168, New York, NY, USA, 2000. ACM.

- [CvKL<sup>+</sup>11] Sergio Cabello, Marc J. van Kreveld, Giuseppe Liotta, Henk Meijer, Bettina Speckmann, and Kevin Verbeek. Geometric simultaneous embeddings of a graph and a matching. *Journal of Graph Algorithms and Applications*, 15(1):79–96, 2011.
- [DBMPP04] Giuseppe Di Battista, Federico Mariani, Maurizio Patrignani, and Maurizio Pizzonia. Bgplay: A system for visualizing the interdomain routing evolution. In Giuseppe Liotta, editor, *Proceedings of the 12th International Symposium on Graph Drawing (GD '04)*, volume 2912 of LNCS, pages 295–306. Springer Berlin Heidelberg, 2004.
- [dBvKOS00] Mark de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, second edition, 2000.
- [DC98] Edmund Dengler and William Cowan. Human perception of laid-out graphs. In Sue H. Whitesides, editor, *Proceedings of the 6th International Symposium on Graph Drawing (GD '98)*, volume 1547 of LNCS, pages 441–443. Springer Berlin Heidelberg, 1998.
- [DDE<sup>+</sup>12] Emilio Di Giacomo, Walter Didimo, Peter Eades, Seok-Hee Hong, and Giuseppe Liotta. Bounds on the crossing resolution of complete geometric graphs. *Discrete Applied Mathematics*, 160(1-2):132–139, 2012.
- [DDL09] Emilio Di Giacomo, Walter Didimo, Giuseppe Liotta, and Henk Meijer. Area, curve complexity, and crossing resolution of non-planar graph drawings. In David Eppstein and Emden R. Gansner, editors, *Proceedings of the 17th International Symposium on Graph Drawing (GD '09)*, volume 5849 of LNCS, pages 15–20, Berlin, Heidelberg, 2009. Springer-Verlag.
- [DDL10] Emilio Di Giacomo, Walter Didimo, Giuseppe Liotta, and Pietro Palladino. Visual analysis of financial crimes. In Giuseppe Santucci, editor, *Proceedings of the Working Conference on Advanced Visual Interface (AVI '10)*, pages 393–394. ACM Press, 2010.
- [DE12] Hooman Reisi Dehkordi and Peter Eades. Every outer-1-plane graph has a right angle crossing drawing. *International Journal of Computational Geometry & Applications*, 22(6):543–557, 2012.
- [DEG<sup>+</sup>10a] Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Drawing trees with perfect angular resolution and polynomial area. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceedings of the*

- 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of LNCS, pages 183–194, Berlin, Heidelberg, 2010. Springer-Verlag.
- [DEG<sup>+</sup>10b] Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Lombardi drawings of graphs. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceeding of the 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of LNCS, pages 195–207, Berlin, Heidelberg, 2010. Springer-Verlag.
- [DEG<sup>+</sup>11] Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Maarten Löffler. Planar and poly-arc lombardi drawings. In Marc J. van Kreveld and Bettina Speckmann, editors, *Proceeding of the 19th International Symposium on Graph Drawing (GD '11)*, LNCS, pages 308–319, Berlin, Heidelberg, 2011. Springer-Verlag.
- [DEG<sup>+</sup>12] Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Lombardi drawings of graphs. *Journal of Graph Algorithms and Applications*, 16(1):85–108, 2012.
- [DEK04] Christian A. Duncan, David Eppstein, and Stephen G. Kobourov. The geometric thickness of low degree graphs. In *Proceedings of the 20th Annual Symposium on Computational Geometry (SCG '04)*, pages 340–346, New York, NY, USA, 2004. ACM.
- [DEL09] Walter Didimo, Peter Eades, and Giuseppe Liotta. Drawing graphs with right angle crossings. In *Proceedings of the Algorithm and Data Structure Symposium (WADS '09)*, pages 206–217, 2009.
- [DEL10] Walter Didimo, Peter Eades, and Giuseppe Liotta. A characterization of complete bipartite graphs. *Information Processing Letters*, 110(16):687–691, 2010.
- [DEL11] Walter Didimo, Peter Eades, and Giuseppe Liotta. Drawing graphs with right angle crossings. *Theoretical Computer Science*, 412(39):5156–5166, 2011.
- [DETT99] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.
- [dFdM96] Hubert de Fraysseix and Patrice Ossona de Mendez. Planarity and edge poset dimension. *European Journal of Combinatorics*, 17(8):731 – 740, 1996.
- [dFdM12] Hubert de Fraysseix and Patrice Ossona de Mendez. Trémaux trees and planarity. *European Journal of Combinatorics*, 33(3):279 – 293, 2012.

- [DGMW10] Vida Dujmović, Joachim Gudmundsson, Pat Morin, and Thomas Wolle. Notes on large angle crossing graphs. In T. Viglas and A. Potanin, editors, *Proceedings 16th Symposium on Computing: the Australasian Theory*, volume 109 of *CATS '10*, pages 19–24, 2010.
- [DH96] Ron Davidson and David Harel. Drawing graphs nicely using simulated annealing. *ACM Transactions on Graphics*, 15:301–331, 1996.
- [DK03] Christian Duncan and Stephen Kobourov. Polar coordinate drawing of planar graphs with good angular resolution. *Journal of Graph Algorithms and Applications*, 7(4):311–333, 2003.
- [DKL<sup>+</sup>12] Walter Didimo, Michael Kaufmann, Giuseppe Liotta, Yoshio Okamoto, and Andreas Spillner. Vertex angle and crossing angle resolution of leveled tree drawings. *Information Processing Letters*, 112(16):630–635, 2012.
- [DL13] Walter Didimo and Giuseppe Liotta. The crossing-angle resolution in graph drawing. *Thirty Essays on Geometric Graph Theory*, pages 167–184, 2013.
- [DLM12] Walter Didimo, Giuseppe Liotta, and Fabrizio Montecchiani. Vis4aui: Visual analysis of banking activity networks. In Paul Richard, Martin Kraus, Robert S. Laramée, and José Braz, editors, *Proceedings of the International Conference on Computer Graphics Theory and Applications and International Conference on Information Visualization Theory and Applications (GRAPP/IVAPP '12)*, pages 799–802. SciTePress, 2012.
- [DLMP11] Walter Didimo, Giuseppe Liotta, Fabrizio Montecchiani, and Pietro Palladino. An advanced network visualization system for financial crime detection. In Giuseppe Di Battista, Jean-Daniel Fekete, and Huamin Qu, editors, *Proceedings of the fourth IEEE Pacific Visualization Symposium (PacificVis '11)*, pages 203–210. IEEE, 2011.
- [DLR10] Walter Didimo, Giuseppe Liotta, and Salvatore Romeo. Topology-driven force-directed algorithms. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceeding of the 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of *LNCS*, pages 165–176, Berlin, Heidelberg, 2010. Springer-Verlag.
- [DLR11] Walter Didimo, Giuseppe Liotta, and Salvatore Romeo. A graph drawing application to web site traffic analysis. *Journal of Graph Algorithms and Applications*, 15(2):229–251, 2011.

- [DMM96] Ugur Dogrusöz, Brendan Madden, and Patrick Madden. Circular layout in the graph layout toolkit. In *Proceedings of the fifth International Symposium on Graph Drawing (GD '96)*, volume 1190 of *LNCS*, pages 92–100. Springer-Verlag, 1996.
- [DMP64] G. Demoucron, Y. Malgrange, and R. Pertuiset. Graphes planaires: reconnaissance et construction de représentations planaires topologiques. *Rev. Francaise Recherche Operationnelle*, 8:33–47, 1964.
- [DO11] Ekrem Duman and M. Hamdi Ozcelik. Detecting credit card fraud by genetic algorithm and scatter search. *Expert Systems with Applications*, 38(10):13057–13063, 2011.
- [DS81] N.R. Draper and H. Smith. *Applied regression analysis*. Wiley series in probability and mathematical statistics. 1981.
- [DSN12] Giuseppe Di Battista, Claudio Squarcella, and Wolfgang Nagele. How to visualize the k-root name server. *Journal of Graph Algorithms and Applications*, 16(3):675–699, 2012.
- [DV93] Giuseppe Di Battista and Luca Vismara. Angles of planar triangular graphs. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing (STOC '93)*, pages 431–437, New York, NY, USA, 1993. ACM.
- [DW01] Hervé Debar and Andreas Wespi. Aggregation and correlation of intrusion-detection alerts. In *Proceedings of the fourth International Symposium on Recent Advances in Intrusion Detection (RAID '00)*, pages 85–103, London, UK, 2001. Springer-Verlag.
- [Ead84] Peter Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.
- [Ead88] Peter Eades. Symmetry finding algorithms. In G. T. Toussaint, editor, *Computational Morphology*, pages 41–51. North-Holland Publishing Company, North-Holland, Amsterdam, Netherlands, 1988.
- [Ead92] Peter Eades. Drawing free trees. *Bulletin of the Institute for Combinatorics and its Applications*, 5:10–36, 1992.
- [EBG]<sup>+</sup>07] Alejandro Estrella-Balderrama, Elisabeth Gassner, Michael Jünger, Merijam Percan, Marcus Schaefer, and Michael Schulz. Simultaneous geometric graph em-

- beddings. In *Proceedings of the 15th International Symposium on Graph Drawing (GD' 07)*, LNCS, pages 280–290, 2007.
- [ECS05] Robert F. Erbacher, Kim Christensen, and Amanda Sundberg. Designing visualization capabilities for ids challenges. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, pages 121–127. IEEE Computer Society, 2005.
- [EFK01] Markus Eiglsperger, Sandor P. Fekete, and Gunnar W. Klau. Orthogonal graph drawing. In Michael Kaufmann and Dorothea Wagner, editors, *Drawing Graphs*, volume 2025 of LNCS, pages 121–171. Springer Berlin Heidelberg, 2001.
- [EFLN99] Peter Eades, Qingwen Feng, Xuemin Lin, and Hiroshi Nagamochi. Straight-line drawing algorithms for hierarchical graphs and clustered graphs. *Algorithmica*, pages 113–128, 1999.
- [EH09] William Eberle and Lawrence B. Holder. Mining for insider threats in business transactions and processes. In *IEEE Symposium on Computational Intelligence and Data Mining (CIDM '09)*, pages 163–170, 2009.
- [EK05a] Cesim Erten and Stephen G. Kobourov. Simultaneous embedding of a planar graph and its dual on the grid. *Theory Computing Systems*, 38(3):313–327, 2005.
- [EK05b] Cesim Erten and Stephen G. Kobourov. Simultaneous embedding of planar graphs with few bends. *Journal of Graph Algorithms and Applications*, 9(3):347–364, 2005.
- [EL11] Peter Eades and Giuseppe Liotta. Right angle crossing graphs and 1-planarity. In Marc J. van Kreveld and Bettina Speckmann, editors, *Proceedings 19th International Symposium on Graph Drawing (GD '11)*, volume 7034 of LNCS, pages 148–153, Berlin, Heidelberg, 2011. Springer-Verlag.
- [EL13] Peter Eades and Giuseppe Liotta. Right angle crossing graphs and 1-planarity. *Discrete Applied Mathematics*, 161(7–8):961 – 969, 2013.
- [ELMN13] David Eppstein, Maarten Löffler, Elena Mumford, and Martin Nöllenburg. Optimal 3d angular resolution for low-degree graphs. *Journal of Graph Algorithms and Applications*, 17(3):173–200, 2013.
- [ELMS91] Peter Eades, Wet Lai, Kazuo Misue, and Kozo Sugiyama. Preserving the mental map of a diagram. In *Proceedings of Compugraphics '91*, pages 24–33, 1991.

- [ELS93] Peter Eades, Xuemin Lin, and William F. Smyth. A fast effective heuristic for the feedback arc set problem. *Information Processing Letters*, 47:319–323, 1993.
- [Eul36] Leonhard Euler. Solutio problematis ad geometriam situs pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, 8:128–140, 1736.
- [EW11] David Eppstein and Kevin A. Wortman. Optimal angular resolution for face-symmetric drawings. *Journal of Graph Algorithms and Applications*, 15(4):551–564, 2011.
- [FCE95] Qing-Wen Feng, Robert F. Cohen, and Peter Eades. How to draw a planar clustered graph. In Ding-Zhu Du and Ming Li, editors, *Computing and Combinatorics*, volume 959 of *LNCS*, pages 21–30. Springer Berlin Heidelberg, 1995.
- [FCS95] Kurt Fanning, Kenneth O. Cogger, and Rajendra P. Srivastava. Detection of management fraud: a neural network approach. In *Proceedings of the 11th Conference on Artificial Intelligence for Applications (AIA '95)*, pages 220–223, 1995.
- [FHH<sup>+</sup>93] Michael Formann, Torben Hagerup, James Haralambides, Michael Kaufmann, Frank Thomson Leighton, Antonios Symvonis, Emo Welzl, and Gerhard J. Woeginger. Drawing graphs in the plane with high resolution. *SIAM Journal on Computing*, 22(5):1035–1052, 1993.
- [FHM<sup>+</sup>11] Martin Fink, Jan-Henrik Haunert, Tamara Mchedlidze, Joachim Spoerhase, and Alexander Wolff. Drawing graphs with vertices at specified positions and crossings at large angles. In Marc J. van Kreveld and Bettina Speckmann, editors, *Proceedings of the 19th International Symposium on Graph Drawing (GD '11)*, *LNCS*, pages 441–442, Berlin, Heidelberg, 2011. Springer-Verlag.
- [FK95] Ulrich Fößmeier and Michael Kaufmann. Drawing high degree graphs with low bend numbers. In Franz J. Brandenburg, editor, *Proceedings of the third Symposium on Graph Drawing (GD '95)*, volume 1027 of *LNCS*. Springer Berlin Heidelberg, 1995.
- [FKK09] Fabrizio Frati, Michael Kaufmann, and Stephen G. Kobourov. Constrained simultaneous and near-simultaneous embeddings. *Journal of Graph Algorithms and Applications*, 13(3):447–465, 2009.
- [FLM95] Arne Frick, Andreas Ludwig, and Heiko Mehltau. A fast adaptive layout algorithm for undirected graphs. In *Proceedings of the DIMACS International Workshop on Graph Drawing (GD '94)*, pages 388–403, London, UK, 1995. Springer-Verlag.



- [FN05] Glenn A. Fink and Chris North. Root polar layout of internet address data for security administration. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, pages 55–64. IEEE Computer Society, 2005.
- [Föß97] Ulrich Fößmeier. Interactive orthogonal graph drawing: Algorithms and bounds. In Giuseppe Di Battista, editor, *Proceedings of the fifth International Symposium on Graph Drawing (GD '97)*, volume 1353 of *LNCS*, pages 111–123. Springer Berlin Heidelberg, 1997.
- [FR91] Thomas Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Software-Practice and Experience*, 21(11):1129–1164, 1991.
- [FT04] Benjamin Finkel and Roberto Tamassia. Curvilinear graph drawing using the force-directed method. In János Pach, editor, *Proceedings of the 12th International Symposium on Graph Drawing (GD '04)*, volume 3383 of *LNCS*, pages 448–453, Berlin, Heidelberg, 2004. Springer-Verlag.
- [FTM<sup>+</sup>98] Deborah Frincke, Don Tobin, Jesse Mcconnell, Jamie Marconi, and Dean Polla. A framework for cooperative intrusion detection. In *Proceedings of the 21st NIST-NCSC National Information Systems Security Conference*, pages 361–373, 1998.
- [GADM06] Sébastien Grivet, David Auber, Jean-Philippe Domenger, and Guy Melancon. Bubble tree drawing algorithm. In K. Wojciechowski, B. Smolka, H. Palus, R.S. Kozera, W. Skarbek, and L. Noakes, editors, *Computer Vision and Graphics*, volume 32 of *Computational Imaging and Vision*, pages 633–641. Springer Netherlands, 2006.
- [GB98] Luc Girardin and Dominique Brodbeck. A visual approach for monitoring logs. In *Proceedings of the 12th USENIX Conference on System Administration (LISA '98)*, pages 299–308, Berkeley, CA, USA, 1998. USENIX Association.
- [GC98] Brian Green and Patrick Choi. Assessing the risk of management fraud through neural network technology. *Auditing - A Journal of Practice & Theory*, 16:14–28, 1998.
- [GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [GJ83] Michael R. Garey and David S. Johnson. Crossing number is NP-complete. *SIAM Journal on Algebraic and Discrete Methods*, 4:312–316, 1983.

- [GJR85] Martin Grötschel, Michael Jünger, and Gerhard Reinelt. On the acyclic subgraph polytope. *Mathematical Programming*, 33(1):28–42, 1985.
- [GKNV93] Emden Gansner, Eleftherios Koutsoufios, Stephen North, and Kiem-Phong Vo. A technique for drawing directed graphs. *IEEE Transactions Software Engineering*, 19(3):214–230, 1993.
- [GKV09] Markus Geyer, Michael Kaufmann, and Imrich Vrto. Two trees which are self-intersecting when drawn simultaneously. *Discrete Mathematics*, 309(7):1909–1916, 2009.
- [GLRK05] John R. Goodall, Wayne G. Lutters, Penny Rheingans, and Anita Komlodi. Preserving the big picture: Visual network traffic analysis with tn. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, page 6, 2005.
- [GM99] Carsten Gutwenger and Petra Mutzel. Planar polyline drawings with good angular resolution. In Sue Whitesides, editor, *Proceedings 6th International Symposium on Graph Drawing (GD '98)*, volume 1547 of LNCS, pages 167–182, 1999.
- [GMW05] Carsten Gutwenger, Petra Mutzel, and René Weiskircher. Inserting an edge into a planar graph. *Algorithmica*, 41(4):289–308, 2005.
- [Gol63] A. J. Goldstein. An efficient and constructive algorithm for testing whether a graph can be embedded in the plane. In Jr. John R. Edmonds, editor, *Graphs and Combinatorics Conference*, Technical Report, page 2 unnn. pp., Princeton University, 1963. JAI Press, Reading.
- [GP13] Michael T. Goodrich and Pawel Pszona. Achieving good angular resolution in 3d arc diagrams. In *Proceeding of the 21th International Symposium on Graph Drawing (GD '13)*, page To appear, 2013.
- [GR95] Barbara Guttman and Edward A. Roback. *An Introduction to Computer Security: The Nist Handbook*. Diane Publishing Company, 1995.
- [GS95] Henry G. Goldberg and Ted E. Senator. Restructuring databases for knowledge discovery by consolidation and link formation. In Usama M. Fayyad and Ramasamy Uthurusamy, editors, *Proceedings of the first International Conference on Knowledge Discovery and Data Mining (KDD '95)*, pages 136–141. AAAI Press, 1995.

- [GT94] Ashim Garg and Roberto Tamassia. Planar drawings and angular resolution: Algorithms and bounds. In Jan van Leeuwen, editor, *Proceedings of the Second Annual European Symposium (ESA '94)*, volume 855 of *LNCS*, pages 12–23, 1994.
- [GT01] Ashim Garg and Roberto Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM Journal on Computing*, 31(2):601–625, 2001.
- [GT11] Michael Goodrich and Roberto Tamassia. *Introduction to Computer Security*. Addison-Wesley Publishing Company, USA, 2011.
- [GTA09] Global technology audit guide: Fraud prevention and detection in an automated world. The Institute of Internal Auditors, 2009.
- [GTV96] Ashim Garg, Roberto Tamassia, and Paola Vocca. Drawing with colors. In Josep Diaz and Maria Serna, editors, *Proceedings of the fourth Annual European Symposium on Algorithms (ESA '96)*, volume 1136 of *LNCS*, pages 12–26. Springer Berlin Heidelberg, 1996.
- [GW00] Michael T. Goodrich and Christopher G. Wagner. A framework for drawing planar graphs with curves and polylines. *Journal of Algorithms*, 37(2):399–421, 2000.
- [HEHL10] Weidong Huang, Peter Eades, Seok-Hee Hong, and Chun-Cheng Lin. Improving force-directed graph drawings by making compromises between aesthetics. In Christopher D. Hundhausen, Emmanuel Pietriga, Paloma Díaz, and Mary Beth Rosson, editors, *Proceedings of the IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2010)*, *LNCS*, pages 176–183, 2010.
- [Hei03] Stephen Heisig. Treemaps for workload visualization. *IEEE Computer Graphics and Applications*, 23(2):60–67, 2003.
- [Hen09] Parker Henry. *Ancient Ceylon: An Account of the Aborigines and of Part of the Early Civilisation*. Luzac and Co., London, 1909.
- [HH10] Weidong Huang and Maolin Huang. Exploring the relative importance of crossing number and crossing angle. In *Proceedings of the third International Symposium on Visual Information Communication (VINCI '10)*, pages 10:1–10:8, New York, NY, USA, 2010. ACM.

- [HHE08] Weidong Huang, Seok-Hee Hong, and Peter Eades. Effects of crossing angles. In *Proceedings of the VGTC Pacific Visualization Symposium (PacificVis '08)*, pages 41–46, 2008.
- [HKLS07] Bethany Hoogs, Thomas Kiehl, Christina Lacombe, and Deniz Senturk. A genetic algorithm approach to detecting temporal patterns indicative of financial statement fraud. *Intelligent Systems in Accounting, Finance & Management*, 15(1-2):41–56, 2007.
- [HLN09] Mao Lin Huang, Jie Liang, and Quang Vinh Nguyen. A visualization approach for frauds detection in financial market. In *Proceedings of the 13th International Conference on Information Visualisation (IV '09)*, pages 197–202, 2009.
- [HPPT08] Alexander Heitzmann, Bernardo Palazzi, Charalampos Papamanthou, and Roberto Tamassia. Effective visualization of file system access-control. In *Proceedings of the fifth International Workshop on Visualization for Computer Security (VizSec '08)*, pages 18–25, Berlin, Heidelberg, 2008. Springer-Verlag.
- [HRRV08] Chris E. Hogan, Zabihollah Rezaee, Richard A. Riley, and Uma K. Velury. Financial statement fraud: Insights from the academic literature. *Auditing - A Journal of Practice & Theory*, 27:231–252, 2008.
- [HSMM00] Ivan Herman, Member IEEE CS Society, Guy Melançon, and M. Scott Marshall. Graph visualization and navigation in information visualization: a survey. *IEEE Transactions on Visualization and Computer Graphics*, 6:24–43, 2000.
- [HT74] John Hopcroft and Robert Tarjan. Efficient planarity testing. *Journal of Association Computing Machinery*, 21(4):549–568, 1974.
- [Hua07] Weidong Huang. Using eye tracking to investigate graph layout effects. In Seok-Hee Hong and Kwan-Liu Ma, editors, *Proceedings of the sixth International Asia-Pacific Symposium on Visualization (APVIS '07)*, pages 97–100, 2007.
- [Hua08] Weidong Huang. An eye tracking study into the effects of graph layout. *CoRR*, abs/0810.4431, 2008.
- [Hua13] Weidong Huang. Establishing aesthetics based on human graph reading behavior: two eye tracking studies. *Personal and Ubiquitous Computing*, 17(1):93–105, 2013.

- [HVOM08] John Homer, Ashok Varikuti, Xinming Ou, and Miles A. Mcqueen. Improving attack graph visualization through data reduction and attack grouping. In *Proceedings of the fifth international workshop on Visualization for Computer Security (VizSec '08)*, pages 68–79, Berlin, Heidelberg, 2008. Springer-Verlag.
- [IAN] I2 Analyst's Notebook. <http://www-03.ibm.com/software/products/us/en/analysts-notebook/>.
- [JD88] Anil K. Jain and Richard C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.
- [JG72] Edward G. Coffman Jr. and Ronald L. Graham. Optimal scheduling for two-processor systems. *Acta Informatica*, 1:200–213, 1972.
- [JMF99] Anil K. Jain, M. Narasimha Murty, and Patrick J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999.
- [JN10] Sushil Jajodia and Steven Noel. Topological vulnerability analysis. In Sushil Jajodia, Peng Liu, Vipin Swarup, and Cliff Wang, editors, *Cyber Situational Awareness*, volume 46 of *Advances in Information Security*, pages 139–154. Springer US, 2010.
- [Kan92] Goos Kant. Drawing planar graphs using the lmc-ordering (extended abstract). In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science (FOCS '92)*, pages 101–110, 1992.
- [Kan96] Goos Kant. Drawing planar graphs using the canonical ordering. *Algorithmica*, 16(1):4–32, 1996.
- [Kar72] Richard M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, page 85–103. Plenum, 1972.
- [KK89] Tomihisa Kamada and Satoru Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7–15, 1989.
- [KLSH04] Yufeng Kou, Chang-Tien Lu, S. Sirwongwattana, and Yo-Ping Huang. Survey of fraud detection techniques. In *Proceeding of the 2004 IEEE International Conference on Networking, Sensing and Control*, volume 2, pages 749–754, 2004.
- [KMBW02] Eriola Kruja, Joe Marks, Ann Blair, and Richard Waters. A short note on the history of graph drawing. In Petra Mutzel, Michael Jünger, and Sebastian Leipert,

- editors, *Proceedings of the 10th International Symposium on Graph Drawing (GD '02)*, volume 2265 of LNCS, pages 272–286. Springer Berlin Heidelberg, 2002.
- [KMG88] Gautam Kar, Brendan Madden, and Richard S. Gilbert. Heuristic layout algorithms for network management presentation services. *Network, IEEE*, 2(6):29–36, 1988.
- [Kob12] Stephen G. Kobourov. Spring embedders and force directed graph drawing algorithms. *CoRR*, abs/1201.3011, 2012.
- [Kos04] Eija Koskivaara. Artificial neural networks in analytical review procedures. *Managerial Auditing Journal*, 19(1):191–223, 2004.
- [KPM03] KPMG forensic: Fraud survey. Project page: <http://www.kpmg.com/>, 2003.
- [KPM06] KPMG forensic: Fraud risk management developping a strategy for prevention, detection and response. Project page: <http://www.kpmg.com/>, 2006.
- [Kre96] Valdis Krebs. Visualizing human networks. *Release 1.0: Esther Dyson's Monthly Report*, page 1–25, 1996.
- [KSH<sup>+</sup>98] J. Dale Kirkland, Ted E. Senator, James J. Hayden, Tom Dybala, Henry G. Goldberg, and Ping Shyr. The nasd regulation advanced detection system (ads). In *Proceedings of the fifteenth national/tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence (AAAI '98/IAAI '98)*, AAAI '98/IAAI '98, pages 1055–1062, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence.
- [kSIH92] Wei kuan Shih and Wen lian Hsu. A simple test for planar graphs. In *Proceedings of the Sixth Workshop on Discrete Mathematics and Theory of Computation*, pages 35–42, 1992.
- [KSM07] Efstathios Kirkos, Charalambos Spathis, and Yannis Manolopoulos. Data mining techniques for the detection of fraudulent financial statements. *Expert Systems with Applications* 3, 32(4):995–1003, 2007.
- [KW01] Michael Kaufmann and Dorothea Wagner, editors. *Drawing Graphs: Methods and Models*, volume 2025 of LNCS. Springer-Verlag, 2001.
- [KW02] Michael Kaufmann and Roland Wiese. Maintaining the mental map for circular drawings. In Petra Mutzel, Michael Jünger, and Sebastian Leipert, editors, *Proceedings of the 10th International Symposium on Graph Drawing (GD '02)*, volume 2265 of LNCS, pages 12–22, London, UK, UK, 2002. Springer-Verlag.

- [KZ91] Christiane Klapisch-Zuber. The genesis of the family tree. *I Tatti Studies: Essays in the Renaissance*, 4, 1991.
- [LEC67] Abraham Lempel, Shimon Even, and Israel Cederbaum. An algorithm for planarity testing of graphs. In Jr. John R. Edmonds, editor, *International Symposium Rome 1966, Theory of Graphs*, pages 215–232, Princeton University, 1967. Gordon and Breach.
- [Lei80] Charles E. Leiserson. Area-efficient graph layouts (for VLSI). In *Proceedings of IEEE Symposium on Foundations of Computer Science*, volume 1547 of *IEEE*, 1980.
- [IH03a] Wen lian Hsu. An efficient implementation of the pc-trees algorithm of Shih and Hsu’s planarity test. Technical report, Institute of Information Science, Academia Sinica, 2003.
- [LH03b] Mark Lombardi and Robert Hobbs. *Mark Lombardi: Global Networks*. Independent Curators, 2003.
- [LHB03] Jerry W. Lin, Mark I. Hwang, and Jack D. Becker. A fuzzy neural network for assessing the risk of fraudulent financial reporting. *Managerial Auditing Journal*, 18:657 – 665, 2003.
- [Lis47] Johann Benedict Listing. Vorstudien zur topologie. *Gottinger Studien*, 1:811--875, 1847.
- [LMR98] Kelly A. Lyons, Henk Meijer, and David Rappaport. Algorithms for cluster busting in anchored graph drawing. In *Journal of Graph Algorithms and Applications*, pages 7–17, 1998.
- [LMS98] Yanpei Liu, Aurora Morgana, and Bruno Simeone. A linear algorithm for 2-bend embeddings of planar graphs in the two-dimensional grid. *Discrete Applied Mathematics*, 81(1-3):69–91, 1998.
- [Lue10] Jonas Luell. *Employee fraud detection under real world conditions*. PhD thesis, 2010.
- [LY80] John M. Lewis and Mihalis Yannakakis. The node-deletion problem for hereditary properties is np-complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980.

- [LY05] Chun-Cheng Lin and Hsu-Chun Yen. A new force-directed graph drawing method based on edge-edge repulsion. In *Proceedings of the ninth International Conference on Information Visualisation (IV '05)*, pages 329–334, 2005.
- [LY07] Chun-Cheng Lin and Hsu-Chun Yen. On balloon drawings of rooted trees. *Journal of Graph Algorithms and Applications*, 11(2):431–452, 2007.
- [LYL04] Kiran Lakkaraju, William Yurcik, and Adam J. Lee. Nvisionip: netflow visualizations of system state for security situational awareness. In *Proceedings of the 2004 ACM Workshop on Visualization and data mining for computer security (VizSEC/DMSEC '04)*, pages 65–72. ACM, 2004.
- [MAC<sup>+</sup>95] Joseph Manning, Mikhail Atallah, Kingsley Cudjoe, Joe Lozito, and Ron Pacheco. A system for drawing graphs with geometric symmetry. In Roberto Tammasia and Ioannis G. Tollis, editors, *Proceedings of Graph Drawing DIMACS International Workshop (GD' 94)*, volume 894 of LNCS, pages pp. 262–265, 1995.
- [Mar08] Raffael Marty. *Applied Security Visualization*. Addison-Wesley Professional, 1 edition, 2008.
- [Mat10] Nicolaos Matsakis. Transforming a random graph drawing into a lombardi drawing. *CoRR*, abs/1012.2202, 2010.
- [MBK97] Cathleen McGrath, Jim Blythe, and David Krackhardt. The effect of spatial arrangement on judgments and errors in interpreting graphs. *Social Networks*, 19(3):223 – 242, 1997.
- [MELS95] Kazuo Misue, Peter Eades, Wei Lai, and Kozo Sugiyama. Layout adjustment and the mental map. *Journal of Visual Languages & Computing*, 6(2):183 – 210, 1995.
- [MFG<sup>+</sup>06] Jaime Montemayor, Andrew Freeman, John Gersh, Thomas Llanso, and Dennis Patrone. Information visualization for rule-based resource access control. In *Proceedings of the second International Symposium on Usable Privacy and Security (SOUPS '06)*, LNCS, 2006.
- [MGB<sup>+</sup>98] Petra Mutzel, Carsten Gutwenger, Ralf Brockenauer, Sergej Fialko, Gunnar Klau, Michael Krüger, Thomas Ziegler, Stefan Näher, David Alberts, Dirk Ambras, Gunter Koch, Michael Jünger, Christoph Buchheim, and Sebastian Leipert. A library of algorithms for graph drawing. In Sue H. Whitesides, editor, *Proceedings of the sixth International Symposium on Graph Drawing (GD '98)*, volume 1547 of LNCS, pages 456–457. Springer Berlin Heidelberg, 1998.



- [MH98] Guy Melancon and Ivan Herman. Circular drawings of rooted trees. Technical report, Amsterdam, The Netherlands, The Netherlands, 1998.
- [MHT93] Kanth Miriyala, Scot W. Hornick, and Roberto Tamassia. An incremental approach to aesthetic graph layout. In *Proceeding of the sixth International Workshop on Computer-Aided Software Engineering (CASE '93)*, pages 297–308, 1993.
- [MJL02] Patrick McConnell, Kimberly Johnson, and Simon Lin. Applications of tree-maps to hierarchical biological data. *Bioinformatics*, 18(9):1278–1279, 2002.
- [MLA12] Alessandra Mazzia, Kristen LeFevre, and Eytan Adar. The pviz comprehension tool for social network privacy settings. In *Proceedings of the eighth Symposium on Usable Privacy and Security (SOUPS '12)*, pages 13:1–13:12, New York, NY, USA, 2012. ACM.
- [MMB05] Chris Muelder, Kwan-Liu Ma, and Tony Bartoletti. A visualization methodology for characterization of network scans. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, page 4. IEEE Computer Society, 2005.
- [MMK08] Florian Mansman, Lorenz Meier, and Daniel A. Keim. Visualization of host behavior for network security. In *Proceedings of the fourth International Workshop on Visualization for Cyber Security (VizSEC '07)*, pages 187–202. Springer Berlin Heidelberg, 2008.
- [MNKF90] Sumio Masuda, Kazuo Nakajima, Toshinobu Kashiwabara, and Toshio Fujisawa. Crossing minimization in linear embeddings of graphs. *IEEE Transactions on Computers*, 39(1):124–127, 1990.
- [MP94] Seth M. Malitz and Achilleas Papakostas. On the angular resolution of planar graphs. *SIAM Journal on Discrete Mathematics*, 7(2):172–183, 1994.
- [MRK84] Jan van Leeuwen Mark R. Kramer. The complexity of wire routing and finding minimum area layouts for arbitrary VLSI circuits. In F. P. Preparata, editor, *Advances in Computing Research*, volume 2 of *VLSI Theory*, pages 129–146, MA, 1984. JAI Press, Reading.
- [MS03] Wolfgang Müller and Heidrun Schumann. Visualization for modeling and simulation: visualization methods for time-dependent data - an overview. In *Proceedings of the 35th Conference on Winter Simulation: driving innovation (WSC '03)*, pages 737–745, 2003.

- [Mur84] John E. Murdoch. *Album of Science - Antiquity and the Middle Ages*. Charles Scribner's Sons, New York, 1984.
- [NEHH10] Quan Nguyen, Peter Eades, Seok-Hee Hong, and Weidong Huang. Large crossing angles in circular layouts. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceedings of the 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of *LNCS*, pages 397–399, Berlin, Heidelberg, 2010. Springer-Verlag.
- [Net] Netmap. <http://info.iet.unipi.it/luigi/netmap/>.
- [NJ04] Steven Noel and Sushil Jajodia. Managing attack graph complexity through visual hierarchical aggregation. In *Proceedings of the 2004 ACM workshop on Visualization and Data Mining for Computer Security (VizSEC/DMSEC '04)*, pages 109–118, New York, NY, USA, 2004. ACM.
- [NJ07] Steven Noel and Sushil Jajodia. Attack graphs for sensor placement, alert prioritization, and attack response. In *Cyberspace Research Workshop, Air Force Cyberspace Symposium*, Shreveport, Louisiana, 2007.
- [NJK05] Steven Noel, Michael Jacobs, Pramod Kalapa, and Sushil Jajodia. Multiple coordinated views for network attack graphs. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, page 12, Washington, DC, USA, 2005. IEEE Computer Society.
- [Nor93] Stephen C. North. Drawing ranked digraphs with recursive clusters. In *Proceedings of the ALCOM Workshop on Graph Drawing (GD '93)*, pages 447–458, 1993.
- [Nor96] Stephen C. North. Incremental layout in dynadag. In Franz J. Brandenburg, editor, *Proceedings of the fourth International Symposium on Graph Drawing (GD '96)*, volume 1027 of *LNCS*, pages 409–418. Springer Berlin Heidelberg, 1996.
- [NR04] Takao Nishizeki and Md. Saidur Rahman. *Planar Graph Drawing*. World Scientific, 2004.
- [oCaL51] Alfonso X King of Castile and Leon. *Libros del ajedrez, dados y tablas*. 1251.
- [OCSZ05] Thian-Huat Ong, Hsinchun Chen, Wai-ki Sung, and Bin Zhu. Newsmap: a knowledge map for online news. *Decision Support Systems*, 39(4):583–597, 2005.
- [OKB06] Jon Oberheide, Manish Karir, and Dionysus Blazakis. Vast: Visualizing autonomous system topology. In *Proceedings of the third International Workshop*

- on *Visualization for Computer Security (VizSEC '06)*, pages 71–80, New York, NY, USA, 2006. ACM.
- [ONP08] Scott O'Hare, Steven Noel, and Kenneth Prole. A graph-theoretic visualization approach to network risk analysis. In JohnR. Goodall, Gregory Conti, and Kwan-Liu Ma, editors, *Proceedings of the fifth International Workshop on Visualization for Computer Security (VizSec '08)*, LNCS, pages 60–67, Berlin, Heidelberg, 2008. Springer-Verlag.
- [OPP07] Olga Ohrimenko, Charalampos Papamanthou, and Bernardo Palazzi. Computer security. In Roberto Tamassia, editor, *Handbook of Graph Drawing and Visualization (Discrete Mathematics and Its Applications)*, pages 285–316. Chapman & Hall/CRC, 2007.
- [O'R98] Joseph O'Rourke. *Computational geometry in C (2nd ed.)*. Cambridge University Press, 1998.
- [OR05] Adam Oline and Dirk Reiners. Exploring three-dimensional visualization for intrusion detection. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, pages 113–120. IEEE Computer Society, 2005.
- [PCA02] Helen C. Purchase, David A. Carrington, and Jo-Anne Allder. Empirical evaluation of aesthetics-based graph layout. *Empirical Software Engineering*, 7(3):233–255, 2002.
- [PCJ97] Helen C. Purchase, Robert F. Cohen, and Murray I. James. An experimental study of the basis for graph drawing algorithms. *Journal of Experimental Algorithmics*, 2, 1997.
- [PLSMG10] Clifton Phua, Vincent C. S. Lee, Kate Smith-Miles, and Ross W. Gayler. A comprehensive survey of data mining-based fraud detection research. *CoRR*, abs/1009.6119, 2010.
- [PS85] Franco P. Preparata and Michael I. Shamos. *Computational geometry: an introduction*. Springer-Verlag, 1985.
- [PS98] Cynthia Phillips and Laura Painton Swiler. A graph-based system for network-vulnerability analysis. In *Proceedings of the 1998 Workshop on New Security Paradigms (NSPW '98)*, pages 71–79, New York, NY, USA, 1998. ACM.
- [PT97] Achilleas Papakostas and Ioannis G. Tollis. A pairing technique for area-efficient orthogonal drawings (extended abstract). In *Proceedings of fourth International*

- Symposium on Graph Drawing (GD '97)*, volume 1190 of LNCS, pages 355–370. Springer Berlin Heidelberg, 1997.
- [PT98] Achilleas Papakostas and Ioannis G. Tollis. Interactive orthogonal graph drawing. *IEEE Transactions on Computers*, 47(11):1297–1309, 1998.
- [Pur97] Helen C. Purchase. Which aesthetic has the greatest effect on human understanding? In *Proceedings of the fifth International Symposium on Graph Drawing (GD '97)*, pages 248–261, London, UK, 1997. Springer-Verlag.
- [Pur00] Helen C. Purchase. Effective information visualisation: a study of graph drawing aesthetics and algorithms. *Interacting with Computers*, 13(2):147–162, 2000.
- [Pur02] Helen C. Purchase. Metrics for graph drawing aesthetics. *Journal of Visual Languages and Computing*, 13(5):501 – 516, 2002.
- [PX12] Li Pan and Qian Xu. Visualization analysis of multi-domain access control policy integration based on tree-maps and semantic substrates. *Intelligent Information Management*, 4:188–193, 2012.
- [RA00] Ronald W. Ritchey and Paul Ammann. Using model checking to analyze network vulnerabilities. In *Proceedings of the 2000 IEEE Symposium on Security and Privacy (SP '00)*, page 156, Washington, DC, USA, 2000. IEEE Computer Society.
- [RDNEH13] Hooman Reisi Dehkordi, Quan Nguyen, Peter Eades, and Seok-Hee Hong. Circular graph drawings with large crossing angles. In SubirKumar Ghosh and Takeshi Tokuyama, editors, *Proceedings of the seventh International Workshop on Algorithms and Computation (WALCOM '13)*, volume 7748 of LNVS, pages 298–309. Springer Berlin Heidelberg, 2013.
- [Rei85] Gerhard Reinelt. The linear ordering problem: Algorithms and applications. *Research and Exposition in Mathematics*, 8, 1985.
- [RT81] Edward M. Reingold and John S. Tilford. Tidier drawings of trees. *IEEE Transactions on Software Engineering*, 7(2):223–228, 1981.
- [Rus07] Adrian Rusu. Tree drawing algorithms. In Roberto Tamassia, editor, *Handbook of Graph Drawing and Visualization (Discrete Mathematics and Its Applications)*, pages 285–316. Chapman & Hall/CRC, 2007.
- [San96] Georg Sander Sander. A fast heuristic for hierarchical manhattan layout. In Franz J. Brandenburg, editor, *Proceedings of the third International Symposium*

- on *Graph Drawing (GD '96)*, volume 1027 of *LNCS*, pages 447–458. Springer Berlin Heidelberg, 1996.
- [SC00] Sônia Fernandes Silva and Tiziana Catarci. Visualization of linear time-oriented data: A survey. In *Proceedings of the first International Conference on Web Information Systems Engineering (WISE'00)*, volume 1 of *WISE '00*, pages 310–319, 2000.
- [Sch05] Satu Elisa Schaeffer. Stochastic local clustering for massive graphs. In TuBao Ho, David Cheung, and Huan Liu, editors, *Advances in Knowledge Discovery and Data Mining*, volume 3518 of *LNCS*, pages 354–360. Springer Berlin Heidelberg, 2005.
- [Sch07] Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27 – 64, 2007.
- [Sen00] Ted E. Senator. Ongoing management and application of discovered knowledge in a large regulatory organization: a case study of the use and impact of nasd regulation's advanced detection system (rads). In *Proceedings of the sixth ACM International Conference on Knowledge Discovery and Data Mining (KDD '00)*, pages 44–53, New York, NY, USA, 2000. ACM.
- [SGL08] John T. Stasko, Carsten Görg, and Zhicheng Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7(2):118–132, 2008.
- [SGS<sup>+</sup>02] Ted E. Senator, Henry G. Goldberg, Ping Shyr, Scott Bennett, Steve Donoho, and Craig Lovell. Information visualization in data mining and knowledge discovery. pages 363–371, 2002.
- [SGW<sup>+</sup>95] Ted E. Senator, Henry G. Goldberg, Jerry Wooton, Matthew A. Cottini, A. F. Umar Khan, Christina D. Klinger, Winston M. Llamas, Michael P. Marrone, and Raphael W. H. Wong. The financial crimes enforcement network ai system (fais) identifying potential money laundering from reports of large cash transactions. *AI Magazine*, 16(4):21–39, 1995.
- [SHJ<sup>+</sup>02] Oleg Sheyner, Joshua W. Haines, Somesh Jha, Richard Lippmann, and Jeanette M. Wing. Automated generation and analysis of attack graphs. In *IEEE Symposium on Security and Privacy*, pages 273–284, 2002.

- [Shn92] Ben Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics (TOG)*, 11(1):92–99, 1992.
- [Shn06] Ben Shneiderman. Discovering business intelligence using treemap visualizations. *B-EYE-Network-Boulder, CO, USA, April, 11, 2006*.
- [SKSE06] Asaf Shabtai, Denis Klimov, Yuval Shahr, and Yuval Elovici. An intelligent, interactive tool for exploration and visualization of time-oriented security data. In *Proceedings of the third International Workshop on Visualization for computer security (VizSEC '06)*, pages 15–22. ACM, 2006.
- [SM91] Kozo Sugiyama and Kazuo Misue Misue. Visualization of structural information: automatic drawing of compound digraphs. *IEEE Transactions on Systems, Man and Cybernetics*, 21(4):876–892, 1991.
- [SM93] Kozo Sugiyama and Kazuo Misue Misue. *An Overview of Diagram Based Idea Organizer: D-ABDUCTOR*. IAS-RR-93-3E. International Institute for Advanced Study of Social Information Science, Fujitsu Limited, 1993.
- [SOSG08] M. Suntinger, H. Obweger, J. Schiefer, and M. E. Groller. Event tunnel: Exploring event-driven business processes. *Computer Graphics and Applications, IEEE*, 28(5):46–55, 2008.
- [SPEC01] Laura P. Swiler, Cynthia Phillips, David Ellis, and Stefan Chakerian. Computer-attack graph generation tool. *DARPA Information Survivability Conference and Exposition*, 2:1307, 2001.
- [SSE81] William G. Stillwell, David A. Seaver, and Ward Edwards. A comparison of weight approximation techniques in multiattribute utility decision making. *Organizational Behavior and Human Performance*, 28(1):62 – 77, 1981.
- [ST07] Janet M. Six and Ioannis G. Tollis. Circular drawing algorithms. In Roberto Tamassia, editor, *Handbook of Graph Drawing and Visualization (Discrete Mathematics and Its Applications)*, pages 285–316. Chapman & Hall/CRC, 2007.
- [Sta99] William Stallings. *Network Security Essentials: Applications and Standards*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.
- [Sta00] John Stasko. An evaluation of space-filling information visualizations for depicting hierarchical structures. *International Journal of Human-Computer Studies - Empirical evaluation of information visualizations*, 53(5):663–694, 2000.

- [STT81] Kozo Sugiyama, Shojiro Tagawa, and Mitsuhiro Toda. Methods for visual understanding of hierarchical system structures. *IEEE Transactions on Systems, Man and Cybernetics*, 11(2):109–125, 1981.
- [Syn11] SynerScope. 2011. <http://www.synerscope.com/>.
- [Tam87] Roberto Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM Journal on Computing*, 16(3):421–444, 1987.
- [TJ92] David Turo and Brian Johnson. Improving the visualization of hierarchies with treemaps: design issues and experimentation. In *Proceedings of the third conference on Visualization (VIS '92)*, pages 124–131, Los Alamitos, CA, USA, 1992. IEEE Computer Society Press.
- [TN05] Jens Toelle and Oliver Niggemann. Supporting intrusion detection by graph clustering and graph drawing. In *Proceedings of third International Workshop on Recent Advances in Intrusion Detection (RAID '00)*. Springer, 2005.
- [TNB04] Alexandre Tarassov, Nikola S. Nikolov, and Jürgen Branke. A heuristic for minimum-width graph layering with consideration of dummy nodes. In Celso C. Ribeiro and Simone L. Martins, editors, *Experimental and Efficient Algorithms*, volume 3059 of *LNCS*, pages 570–583. Springer Berlin Heidelberg, 2004.
- [Tol00] Juan Toledo. Etherape a live graphical network monitor tool. Project page: <http://etherape.sourceforge.net/>, 2000.
- [TRNC06] Soon Tee Teoh, Supranamaya Ranjan, Antonio Nucci, and Chen-Nee Chuah. Bgp eye: a new visualization tool for real-time detection and analysis of bgp anomalies. In *Proceedings of the third International Workshop on Visualization for Computer Security (VizSEC '06)*, pages 81–90, New York, NY, USA, 2006. ACM.
- [TT89] Roberto Tamassia and Ioannis G. Tollis. Planar grid embedding in linear time. *IEEE Transactions on Circuits and Systems*, 36(9):1230–1234, 1989.
- [Tut63] William Thomas Tutte. How to draw a graph. *Proceedings of the London Mathematical Society*, s3-13(1):743–767, 1963.
- [TX94] Ioannis G. Tollis and Chunliang Xia. Drawing telecommunication networks. In *Proceedings of the DIMACS International Workshop on Graph Drawing (GD '94)*, pages 206–217, London, UK, 1994. Springer-Verlag.

- [Val81] Leslie G. Valiant. Universality considerations in VLSI circuits. *IEEE Transaction on Computers*, 30(2):135–140, 1981.
- [Van08] Grant Vandenberghe. Network traffic exploration application: A tool to assess, visualize, and analyze network security events. In *Proceedings of the fifth International Workshop on Visualization for Computer Security (VizSec '08)*, pages 181–196. Springer-Verlag, 2008.
- [VHGK03] Michail Vlachos, Marios Hadjieleftheriou, Dimitrios Gunopulos, and Eamonn Keogh. Indexing multi-dimensional time-series with support for multiple distance measures. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '03)*, pages 216–225, 2003.
- [vK10] Marc van Kreveld. The quality ratio of RAC drawings and planar drawings of planar graphs. In Ulrik Brandes and Sabine Cornelsen, editors, *Proceedings of the 18th International Symposium on Graph Drawing (GD '10)*, volume 6502 of *LNCS*, pages 371–376, Berlin, Heidelberg, 2010. Springer-Verlag.
- [VMF98] G. Vert, J. McConnell, and D. Frincke. Towards a mathematical model for intrusion. In *Proceedings of the 21st National Information Systems Security Conference*, pages 329–337, 1998.
- [VWvdW99] Jarke J. Van Wijk and Huub van de Wetering. Cushion treemaps: Visualization of hierarchical information. In *Proceedings of the 1999 IEEE Symposium on Information Visualization (INFOVIS '99)*, pages 73–78, Washington, DC, USA, 1999. IEEE Computer Society.
- [Wal90] John Q. Walker, II. A node-positioning algorithm for general trees. *Software—Practice & Experience*, 20(7):685–705, 1990.
- [WAM01] Marc Weber, Marc Alexa, and Wolfgang Müller. Visualizing time-series on spirals. In *Proceedings of the IEEE Symposium on Information Visualization (INFOVIS '01)*, pages 7–14, 2001.
- [Wan10] Shiguo Wang. A comprehensive survey of data mining-based accounting-fraud detection research. In *Proceedings of the second International Conference on Intelligent Computation Technology and Automation (ICICTA '10)*, volume 1, pages 50–53, 2010.
- [WBH<sup>+</sup>03] Michael Wolverton, Pauline Berry, Ian W. Harrison, John D. Lowrance, David N. Morley, Andres C. Rodriguez, Enrique H. Ruspini, and Jérôme Thoméré. Law: A



- workbench for approximate pattern matching in relational data. In John Riedl and Randall W. Hill Jr., editors, *Proceedings of the 15th Innovative Applications of Artificial Intelligence Conference (IAAI '03)*, pages 143–150. AAAI, 2003.
- [Wei01] René Weiskircher. Drawing planar graphs. In Michael Kaufmann and Dorothea Wagner, editors, *Drawing Graphs*, volume 2025 of *LNCS*, pages 23–46. Springer Berlin Heidelberg, 2001.
- [Wes04] Marcos Weskamp. Newsmap. <http://marumushi.com/projects/newsmap>, 2004.
- [WKWL99] Shih Wei-Kuan and Hsu Wen-Lian. A new planarity test. *Theoretical Computer Science*, 223(1–2):179 – 191, 1999.
- [WRW98] Orion J. Welch, Thomas E. Reeves, and Sandra T. Welch. Using a genetic algorithm-based classifier system for modeling auditor decision behavior in a fraud setting. *Intelligent Systems in Accounting, Finance and Management*, 7(3):173–186, 1998.
- [XC05] Jennifer J. Xu and Hsinchun Chen. Crimenet explorer: a framework for criminal network knowledge discovery. *ACM Transactions on Information Systems (TOIS)*, 23(2):201–226, 2005.
- [XLE] Xanalys link explorer. <http://www.xanalys.com/>.
- [XMB<sup>+</sup>06] Ioannis Xydias, Georgios Miaoulis, Pierre-François Bonnefoi, Dimitri Plemenos, and Djamchid Ghazanfarpour. 3D graph visualisation of web normal and malicious traffic. In *Proceedings of the 10th International Conference on Information Visualisation (IV '06)*, pages 621–629, 2006.
- [Yee06] George Yee. Visualization for privacy compliance. In *Proceedings of the third International Workshop on Visualization for Computer Security (VizSEC '06)*, pages 117–122, New York, NY, USA, 2006. ACM.
- [YSTW05] Danfeng Yao, Michael Shin, Roberto Tamassia, and William H. Winsborough. Visualization of automated trust negotiation. In *Proceedings of the IEEE Workshops on Visualization for Computer Security (VIZSEC '05)*, page 8, Washington, DC, USA, 2005. IEEE Computer Society.
- [yWo] yWorks. Project page: <http://www.yworks.com/en/index.html>.

- [YWW<sup>+</sup>07] Dianmin Yue, Xiaodan Wu, Yunfeng Wang, Yue Li, and Chao-Hsien Chu. A review of data mining-based financial fraud detection research. In *Proceedings of the 2007 International Conference on Wireless Communications, Networking and Mobile Computing (WiCom '07)*, pages 5519–5522, 2007.
- [YYT<sup>+</sup>04] Xiaoxin Yin, William Yurcik, Michael Treaster, Yifan Li, and Kiran Lakkaraju. Vis-flowconnect: Netflow visualizations of link relationships for security situational awareness. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security (VizSEC/DMSEC '04)*, pages 26–34, New York, NY, USA, 2004. ACM.
- [ZZ04] Dongsong Zhang and Lina Zhou. Discovering golden nuggets: data mining in financial application. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 34(4):513–522, 2004.