



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΣΥΣΤΗΜΑΤΩΝ ΒΑΣΕΩΝ ΓΝΩΣΕΩΝ ΚΑΙ
ΔΕΔΟΜΕΝΩΝ**

Πολυπλεξία τροχιών κινούμενων αντικειμένων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Κυριάκου Χ. Τούμπα

Επιβλέπων: Τιμολέων Σελλής - Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2011



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΣΥΣΤΗΜΑΤΩΝ ΒΑΣΕΩΝ ΓΝΩΣΕΩΝ ΚΑΙ ΔΕΔΟΜΕΝΩΝ

Πολυπλεξία τροχιών κινούμενων αντικειμένων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Κυριάκου Χ. Τούμπα

Επιβλέπων: Τιμολέων Σελλής - Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 11η Ιουλίου 2011.

Τιμολέων Σελλής
Καθηγητής Ε.Μ.Π.

Νεκτάριος Κοζύρης
Αναπλ. Καθηγητής
Ε.Μ.Π.

Θοδωρής Δαλαμάγκας
Ερευνητής Β'
ΙΠΣΥΠ/Ε.Κ. "Αθηνά"

Αθήνα, Ιούλιος 2011.

.....
Κυριάκος Χ. Τούμπας

Διπλωματούχος Ηλεκτρολόγος Μηχανικός & Μηχανικός Ηλεκτρονικών
Υπολογιστών

© Τούμπας Κυριάκος, 2011 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 το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Πολυπλεξία τροχιών κινούμενων αντικειμένων

Περίληψη

Σκοπός της διπλωματικής εργασίας είναι η σχεδίαση και η υλοποίηση ενός αλγορίθμου που παρακολουθεί την κίνηση αντικειμένων στο χώρο σε πραγματικό χρόνο και εντοπίζει ανάμεσά τους ομάδες αντικειμένων με κοινή τροχιά. Για κάθε τέτοια ομάδα, ο αλγόριθμος επιστρέφει μία κατασκευασμένη τροχιά ως αποτέλεσμα της πολυπλεξίας των επιμέρους τροχιών των αντικειμένων.

Οι πληροφορίες για την κίνηση των αντικειμένων φτάνουν στο σύστημα με τη μορφή ρεύματος δεδομένων. Τα κινούμενα αντικείμενα θεωρούνται σημειακά και η θέση τους μεταβάλλεται δυναμικά. Κάθε αντικείμενο αποστέλλει ανά τακτά χρονικά διαστήματα την ανανεωμένη θέση του σε ένα κεντρικό επεξεργαστή ο οποίος εκτελεί τον αλγόριθμο πολυπλεξίας.

Ο αλγόριθμος δουλεύει επαναληπτικά. Κάθε κύκλος εκτέλεσης περιλαμβάνει πέντε διεργασίες. Αρχικά, υπολογίζει την κατεύθυνση που ακολούθησε κάθε αντικείμενο σε σχέση με την θέση του κατά τον προηγούμενο κύκλο (εξομάλυνση κίνησης). Στη συνέχεια, η κατεύθυνση αυτή κωδικοποιείται και συνδυάζεται με προηγούμενες κωδικοποιήσεις για την δημιουργία μιας συμβολικής αναπαράστασης της πρόσφατης κίνησης του κάθε αντικειμένου (προσέγγιση πορείας). Κάνοντας χρήση των συμβολικών αναπαραστάσεων, ο αλγόριθμος εντοπίζει αντικείμενα που είχαν παράλληλη κίνηση στο πρόσφατο παρελθόν (ομαδοποίηση). Στο επόμενο βήμα, εντοπίζονται ομάδες αντικειμένων που εκτός από παράλληλη κίνηση είχαν όντως κοινή τροχιά (εύρεση αντικειμένων με κοινές τροχιές). Τέλος, υπολογίζεται η πολυπλεγμένη τροχιά για κάθε ομάδα, ώστε να προσεγγίζει τις τροχιές που εμπίπτουν σε αυτήν με όσο το δυνατόν καλύτερο τρόπο (πολυπλεξία τροχιών).

Ο αλγόριθμος δοκιμάστηκε σε συνθετικά δεδομένα που προσομοίωσαν την κίνηση 10000 αυτοκινήτων στο οδικό δίκτυο της Αθήνας. Τα αποτελέσματα ήταν πολύ ενθαρρυντικά, αφού οι χρόνοι εκτέλεσης που επιτεύχθηκαν για τον εντοπισμό πολυπλεγμένων τροχιών ήταν της τάξεως μερικών δευτερολέπτων. Η πολυπλεγμένη τροχιά προσεγγίζει με ικανοποιητική πιστότητα τις επιμέρους κινήσεις των αντικειμένων και το σφάλμα δεν υπερβαίνει τα όρια ανοχής που τίθενται από τον χρήστη. Τα αποτελέσματα δείχνουν πώς ο αλγόριθμος αυτός θα μπορούσε να χρησιμοποιηθεί με επιτυχία για τον εντοπισμό πομπών αντικειμένων σε διάφορα πεδία εφαρμογής (οχήματα, πλοία, πανίδα κ.α), καθώς επίσης και για τον εντοπισμό αντικειμένων που κινούνται μαζικά προς την ίδια κατεύθυνση σε διάταξη μετώπου.

Λέξεις κλειδιά: Ρεύματα δεδομένων, Κινούμενα αντικείμενα, Τροχιά, Πολυπλεξία, Παράθυρο, Συμβολική αναπαράσταση.

Multiplexing Trajectories of Moving Objects

Abstract

The goal of this thesis is to design and implement a real-time algorithm that tracks the movement of spatial objects and identifies groups having a similar trajectory. For each such group, the algorithm returns a calculated trajectory as the result of multiplexing the individual trajectories of the objects that belong to that group.

Positional data from numerous moving objects arrive at the system as a data stream. Each object is abstracted as a point, but its position changes dynamically and relayed frequently to a main processing unit where the multiplexing algorithm is being executed.

The algorithm works iteratively in execution cycles. Each cycle consists of five processes. First, it calculates the direction of movement for each object by considering its position during the previous execution cycle (*movement smoothing*). Then, the direction is encoded and combined with previous encodings in order to create a symbolic representation of the recent movement of each object (*movement approximation*). Utilizing these symbolic representations, the algorithm identifies objects with parallel courses (*grouping*) in the recent past. Next, it identifies groups of objects with parallel movement which also share a common trajectory (*finding objects with common trajectories*). Finally, it calculates one multiplexed trajectory for each group, minimizing the error with respect to the original ones (*trajectory multiplexing*).

This approach was tested against a synthetic dataset simulating the movement of 10000 vehicles on the road network of Athens. Experimental results were very encouraging since detection of multiplexed trajectories did not require more than a few seconds per cycle. Besides, the multiplexed trajectory achieves a good approximation of the original ones, while the error does not exceed the tolerance imposed by the user. The results show that the algorithm can be successfully used for detecting convoys of various types of objects (vehicles, ships, animals, etc.), as well for discovering objects that move en masse towards the same direction forming a front.

Keywords: Data stream, Moving objects, Trajectory, Symbolic representation, Multiplexing, Window.

Πρόλογος

Παντού στο κόσμο υπάρχουν “αντικείμενα”. Ζωντανά ή άψυχα όπως ζώα, άνθρωποι, δέντρα ή αντικείμενα τα οποία δεν υπάρχουν στη φύση αλλά επινόησε ο άνθρωπος. Οι τρόποι με τους οποίους αλληλεπιδρούν είναι ποικίλοι και η μελέτη τους εκτείνεται από τη φυσική και τα μαθηματικά έως τη βιολογία και τα οικονομικά.

Μία ομάδα αντικειμένων που παρουσιάζει μεγάλο ενδιαφέρον είναι τα αντικείμενα τα οποία κινούνται. Σαφώς οτιδήποτε μπορεί να μετακινηθεί τότε μπορεί να τεθεί και σε κίνηση, αλλά η κίνηση αυτή είναι προβλέψιμη, αφού αυτός που την προκαλεί γνωρίζει από πριν τι πρόκειται να συμβεί. Όταν για παράδειγμα κάποιος πετάει μια μπάλα είναι σε θέση να υπολογίσει πού πρόκειται αυτή να προσγειωθεί.

Το ενδιαφέρον παρουσιάζεται στα αντικείμενα τα οποία μπορούν να τεθούν σε κίνηση από μόνα τους, όπως οι άνθρωποι τα ζώα και τα πουλιά. Η κίνησή τους μπορεί να είναι προδιαγεγραμμένη από τα ίδια, αλλά κοιτάζοντάς τα ομαδικά είναι αδύνατο να συμπεράνει κανείς την κίνηση αυτή με την πρώτη ματιά. Η μελέτη και η καταγραφή της μπορούν να δώσουν πολύ σημαντικές πληροφορίες οι οποίες μπορούν να αξιοποιηθούν σε πολλούς τομείς. Η μελέτη της κίνησης των πουλιών μπορεί να αποκαλύψει μεταναστευτικές και βιολογικές συνήθειες, όπως τόπους όπου βόσκουν ή πίνουν νερό. Μελετώντας την κίνηση των ανθρώπων σε μια πόλη μπορούν να εντοπιστούν πολυσύχναστες περιοχές. Η μελέτη της κίνησης των νεφών μπορεί να βοηθήσει στην πρόβλεψη βροχοπτώσεων κ.ο.κ.

Η εργασία αυτή ασχολείται με αυτά τα αντικείμενα και την μελέτη της κίνησής τους. Εξετάζει τρόπους καταγραφής της, αποθήκευσής της και τέλος προτείνει μία καινούρια μέθοδο για την ανίχνευση συλλογικής κίνησης πολλαπλών αντικειμένων.

Διατύπωση του προβλήματος

Σε πολλές σύγχρονες εφαρμογές συλλέγονται πληροφορίες για κινούμενα αντικείμενα (λ.χ. οχήματα, πλοία, άγρια ζώα κλπ) τα οποία μεταβάλλουν τη θέση τους δυναμικά. Με τη σύγχρονη τεχνολογία τα αντικείμενα είναι σε θέση να βρίσκουν διαρκώς το στίγμα τους χρησιμοποιώντας συσκευές GPS ή ετικέτ-

τες RFID και να την ενημερώνουν ασύρματα μέσω τηλεπικοινωνιακών δικτύων.

Τα αντικείμενα κινούνται αυθαίρετα στο χώρο. Κάποιες φορές τυγχάνει ομάδες αντικειμένων να εκτελούν παρόμοια κίνηση για κάποιο, συνήθως περιορισμένο, χρονικό διάστημα. Για ορισμένες εφαρμογές, όταν ο αριθμός των αντικειμένων ή το χρονικό διάστημα για το οποίο αυτό συμβαίνει είναι επαρκώς μεγάλα, η εύρεση των σχετικών αντικειμένων αλλά και η περιγραφή της κοινής τους κίνησης με κάποιο τρόπο κρίνεται απαραίτητη.

Στόχος της εργασίας είναι η δημιουργία ενός αλγορίθμου που θα εντοπίζει τέτοιες περιπτώσεις αντικειμένων και θα υπολογίζει μία συνδυασμένη τροχιά που να περιγράφει όσο το δυνατόν καλύτερα τη συσχετισμένη κίνησή τους. Ουσιαστικά πρόκειται για *πολυπλεξία τροχιών*, η οποία διενεργείται σε πραγματικό χρόνο και ανανεώνεται περιοδικά με τις νέες θέσεις των αντικειμένων.

Η συμβολή της εργασίας αυτής

Μελετώντας κανείς την βιβλιογραφία διαπιστώνει ένα κενό για εφαρμογές που εντοπίζουν κοινή κίνηση αντικειμένων σε πραγματικό χρόνο. Η εργασία αυτή προσπαθεί να μειώσει το κενό αυτό, προτείνοντας μία εξειδικευμένη μέθοδο.

Συγκεκριμένα, επινοήθηκε και αναπτύχθηκε ένας αλγόριθμος που μπορεί να εντοπίζει αντικείμενα τα οποία κινούνται συλλογικά και σε κοντινή απόσταση μεταξύ τους για ορισμένα χρονικά διαστήματα. Ο αλγόριθμος δέχεται ως μοναδική είσοδο θέσεις αντικειμένων στον χώρο. Καθώς αυτές μεταβάλλονται, εντοπίζονται όσα κινούνται “μαζί” και επιστρέφεται μία τροχιά η οποία περιγράφει προσεγγιστικά την κίνηση αυτή.

Διάρθρωση του τεύχους

Στο κεφάλαιο 1 γίνεται μία σύντομη επισκόπηση του μοντέλου ρευμάτων δεδομένων, πώς αυτό συνδυάζεται με τα κινούμενα αντικείμενα και ποια χαρακτηριστικά πρέπει να ληφθούν υπόψη κατά την επεξεργασία των στοιχείων.

Στο κεφάλαιο 2 μελετάται η έννοια της τροχιάς ενός αντικειμένου, τρόποι καταγραφής και αναπαράστασής της, όπως και βασικοί τρόποι επεξεργασίας της.

Στο κεφάλαιο 3 επεξηγείται ο προτεινόμενος αλγόριθμος πολυπλεξίας, ενώ στο κεφάλαιο 4 σχολιάζονται αποτελέσματα που προέκυψαν από πειραματικές δοκιμές του.

Τέλος, στο κεφάλαιο 5 γίνεται μία ανασκόπηση της εργασίας και προτείνονται επεκτάσεις της.

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω ειλικρινά όλους όσους με βοήθησαν και μου συμπαραστάθηκαν κατά τη διάρκεια της επίπονης αυτής προσπάθειας.

Καταρχήν, το πιο μεγάλο ευχαριστώ το οφείλω στον υποψήφιο διδάκτωρα κ. Κώστα Πατρούμπα, τόσο για την καθοδήγηση και καθοριστική συμβολή του στην εκπόνηση αυτής της διπλωματικής εργασίας, όσο και για τη μεγάλη επιμονή και υπομονή του καθ'όλη τη διάρκεια της συνεργασίας μας. Η έκβαση αυτής της εργασίας δεν θα ήταν η ίδια χωρίς τις δικές του παρεμβάσεις. Ελπίζω η τελική αυτή μορφή της να είναι αντάξια των προσδοκιών του και να δικαιολογεί τις ατελείωτες ώρες συζήτησης που είχαμε προκειμένου να υλοποιηθεί το έργο αυτό.

Θερμές ευχαριστίες οφείλω επίσης στον καθηγητή μου κ. Τίμο Σελλή για το ειλικρινές του ενδιαφέρον για την πρόοδο της εργασίας σε όλα τα στάδια εκπόνησής της.

Τέλος, θα ήθελα να εκφράσω τις ευχαριστίες μου προς τους γονείς μου, χωρίς τους οποίους δεν θα είχα καταφέρει τίποτα από όσα έχω κάνει μέχρι σήμερα, αλλά και στους φίλους μου που με ανέχτηκαν όλον αυτό τον καιρό. Μαρίνα και Αντρέα, ελπίζω τα νεύρα σας να κρατάνε ακόμα...

Περιεχόμενα

Πρόλογος	vii
1 Ρεύματα κινούμενων αντικειμένων	1
1.1 Επισκόπηση ρευμάτων δεδομένων	1
1.1.1 Μοντέλο ρεύματος δεδομένων	2
1.1.2 Ερωτήματα σε ρεύματα δεδομένων	5
1.1.3 Παράθυρα	7
1.1.4 Κυριότερα συστήματα διαχείρισης ρευμάτων δεδομένων	8
1.2 Χωρικά και Χωροχρονικά δεδομένα	9
1.2.1 Χωρικοί τύποι δεδομένων	10
1.2.2 Ερωτήματα σε χωρικά δεδομένα	11
1.2.3 Χωρικά ευρετήρια	13
1.2.4 Χωροχρονικά δεδομένα	14
1.3 Διαχείριση κινούμενων αντικειμένων	16
1.3.1 Χαρακτηριστικά της κίνησης	16
1.3.2 Ενημέρωση θέσης	16
1.3.3 Ρεύματα κινούμενων αντικειμένων	17
1.3.4 Συστήματα διαχείρισης κινούμενων αντικειμένων	18
2 Τροχιές κινούμενων αντικειμένων	21
2.1 Αναπαράσταση τροχιάς	22
2.1.1 Συνεχές μοντέλο	22
2.1.2 Διακριτό μοντέλο	23
2.2 Συμπύεση πληροφορίας κίνησης	26
2.2.1 Offline συμπύεση	27
2.2.2 Online συμπύεση	28
2.2.3 Προβλήματα	29
2.3 Ερωτήματα σε τροχιές	29
2.3.1 Τοπολογικά ερωτήματα	30
2.3.2 Ερωτήματα πλοήγησης	32
2.4 Συνδυασμός τροχιών	33
2.4.1 Σύξευξη τροχιών	33
2.4.2 Εύρεση σημαντικών μονοπατιών	34

2.4.3	Πομπές αντικειμένων	35
3	Πολυπλεξία τροχιών	37
3.1	Τροχιές με κοινές ιδιότητες	37
3.1.1	Χωρική εγγύτητα και Χρονική συνάφεια	37
3.1.2	Ταύτιση τροχιών	38
3.1.3	Προδιαγραφές προβλήματος	39
3.2	Περιγραφή και παράμετροι	40
3.2.1	Γενική ιδέα	40
3.2.2	Παράμετροι	41
3.3	Μεθοδολογία πολυπλεξίας τροχιών	46
3.3.1	Εξομάλυνση κίνησης	46
3.3.2	Προσέγγιση πορείας	49
3.3.3	Ομαδοποίηση αντικειμένων βάσει παράλληλων τροχιών	54
3.3.4	Εύρεση αντικειμένων με κοινή τροχιά	54
3.3.5	Εύρεση πολυπλεγμένης τροχιάς ομάδας αντικειμένων	56
3.3.6	Παθολογικές περιπτώσεις	57
4	Πειραματική μελέτη	61
4.1	Αρχιτεκτονικό διάγραμμα	61
4.2	Μετρήσεις	62
4.2.1	Πειραματικά δεδομένα	62
4.2.2	Αποτελέσματα επιδόσεων	63
4.2.3	Σφάλμα προσέγγισης πολυπλεγμένης τροχιάς	71
4.2.4	Οπτικοποιημένα αποτελέσματα	72
5	Συμπεράσματα - Προοπτικές	75
5.1	Συμπεράσματα	75
5.2	Πρακτικές εφαρμογές	75
5.2.1	Βελτίωση απόδοσης τοπολογικών ερωτημάτων	76
5.2.2	Εντοπισμός πομπής/φάλαγγας	76
5.2.3	Βελτιστοποίηση Vehicular Ad-Hoc δικτύων	76
5.2.4	Εντοπισμός μαζικής κίνησης/διαφυγής ζώων	77
5.3	Μελλοντική εργασία	77
5.3.1	Μείωση χρόνου εκτέλεσης	77
5.3.2	Ποιότητα αποτελεσμάτων	79
5.3.3	Επέκταση λειτουργικότητας	79
A'	Τεκμηρίωση κώδικα	81
A'.1	Δομικά στοιχεία	81
A'.1.1	LocationStamp	81
A'.1.2	MovingObject	81
A'.1.3	TrajectoryHistory	81
A'.1.4	Buffer	82

A.2 Coordinator	82
A.3 ObjectSet	82
A.4 Multiplexer	83
A.4.1 Hashing	83
A.4.2 DBScan	83
A.4.3 TrajectoryCalc	83
B' Αλγόριθμος DBScan	85
Γ' Ορολογία	89

Κεφάλαιο 1

Ρεύματα κινούμενων αντικειμένων

Στις σημερινή πραγματικότητα, η εξέλιξη των ασύρματων δικτύων και του διαδικτύου, έχουν δώσει την δυνατότητα ανάπτυξης εφαρμογών για παροχή υπηρεσιών πραγματικού χρόνου σε χρήστες που βρίσκονται σε διαρκή κίνηση. Τέτοιες εφαρμογές χαρακτηρίζονται από την ανάγκη να περιγράφουν πολύ δυναμική πληροφορία με μεγάλο συνήθως όγκο. Τα κλασικά μοντέλα δεδομένων και τα υπάρχοντα συστήματα διαχείρισής τους κρίνονται ανεπαρκή να ικανοποιήσουν τις απαιτήσεις αυτές. Ιδανικό μοντέλο κρίθηκε αυτό των ρευμάτων δεδομένων. Συστήματα που μπορούν να διαχειρίζονται τέτοιου είδους πληροφορία διαφοροποιούνται από τα κλασικά όσον αφορά τη δυνατότητα τους να διαχειρίζονται πολύ μεγάλο όγκο δεδομένων με χρόνους επεξεργασίας που να δίνουν απαντήσεις σε πραγματικό χρόνο.

Το κεφάλαιο αυτό επεξηγεί εν συντομία τα χαρακτηριστικά των ρευμάτων δεδομένων, όπως επίσης και τον τρόπο που τα κινούμενα αντικείμενα και η διαχείρισή τους από τις σύγχρονες εφαρμογές μπορεί να εκφραστεί από το μοντέλο αυτό.

1.1 Επισκόπηση ρευμάτων δεδομένων

Πολλές σύγχρονες εφαρμογές ασχολούνται με την παρακολούθηση φαινομένων ή διεργασιών προκειμένου να εξάγουν χρήσιμα συμπεράσματα και αποτελέσματα. Οι εφαρμογές αυτές, χρειάζεται να διαχειρίζονται πολύ μεγάλο όγκο πληροφορίας και να αποκρίνονται σε πραγματικό χρόνο. Τυπικά παραδείγματα τέτοιων εφαρμογών είναι:

- *Χρηματιστηριακές εφαρμογές:* Παρακολουθούν πολύ μεγάλο όγκο συναλλαγών και μεταβολών παραμέτρων όπως τιμές μετοχών και δεικτών, απαντήσεις σε πραγματικό χρόνο.

2 ΚΕΦΑΛΑΙΟ 1. ΠΕΥΜΑΤΑ ΚΙΝΟΥΜΕΝΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ

- *Δίκτυα αισθητήρων:* Εκατοντάδες ή χιλιάδες κόμβοι αποστέλλουν δεδομένα ασταμάτητα με σκοπό την εξαγωγή συμπερασμάτων που προκύπτουν από τις μετρήσεις (λ.χ μέση θερμοκρασία).
- *Διαχείριση δικτύων:* Παρακολούθηση και ανάλυση της κίνησης στο δίκτυο για σκοπούς ασφαλείας και βελτιστοποίησης απόδοσης. Αντιπροσωπευτικό παράδειγμα οι εξυπηρετητές που απαρτίζουν το διαδίκτυο.
- *Τηλεπικοινωνιακά κυψελωτά συστήματα:* Παρακολούθηση κινητικότητας χρηστών για ανακατανομή πόρων και αποφυγή συμφορήσεων.
- *Κινούμενα αντικείμενα:* Παρακολούθηση κίνησης αντικειμένων στον χώρο με σκοπό τον εντοπισμό εγγύτερων γειτόνων, κοινών τροχιών και άλλων χωροχρονικών ερωτημάτων.
- *Στρατιωτικές εφαρμογές:* Παρακολούθηση στρατευμάτων και δημιουργία στρατηγικών πλάνων.

Οι εφαρμογές αυτές παρακολουθούν συνεχώς τιμές που φτάνουν για να ενημερώσουν για την κατάσταση κάποιου φαινομένου, όπως είναι η θερμοκρασία σε ένα δωμάτιο ή η νέα τιμή που απέκτησε μία μετοχή. Η συνεχής αυτή έλευση των νέων τιμών περιγράφεται από το μοντέλο του ρεύματος δεδομένων.

1.1.1 Μοντέλο ρεύματος δεδομένων

Η συνεχής ροή πληροφορίας που παράγεται από μία ή περισσότερες πηγές με προορισμό κάποιο δέκτη μεταδιδόμενο μέσω ενός καναλιού επικοινωνίας ονομάζεται *ρεύμα δεδομένων* (data stream).

Η πληροφορία εξαρτάται άμεσα από την εκάστοτε εφαρμογή. Μπορεί να είναι η παρατήρηση ενός φυσικού φαινομένου (θερμοκρασία, ταχύτητα, ένταση φωτισμού κλπ.) ή κατασκευασμένη πληροφορία όπως πακέτα δεδομένων σε ένα δίκτυο υπολογιστών ή οι τιμές των μετοχών στο χρηματιστήριο. Η μεταφορά της γίνεται με τη χρήση διακριτών πακέτων πληροφορίας ώστε να μπορεί να αξιοποιηθεί από τα σύγχρονα ψηφιακά συστήματα.

Για παράδειγμα, ένα δίκτυο αισθητήρων καταγράφει την θερμοκρασία σε διάφορα σημεία ενός κτηρίου. Ένας κεντρικός υπολογιστής λαμβάνει τις τιμές των μετρήσεων και εκτελεί αναλόγως ενέργειες για την ρύθμιση της θερμοκρασίας. Οι αισθητήρες μπορούν να στέλνουν τιμές είτε σε τακτά χρονικά διαστήματα (λ.χ κάθε 10 δευτερόλεπτα) είτε όποτε αντιλαμβάνονται αλλαγή θερμοκρασίας περισσότερη από $0.5\text{ }^{\circ}\text{C}$. Όταν ο αισθητήρας αποστέλλει μια μέτρηση, στην ουσία στέλνει μια πλειάδα τιμών της μορφής <χρόνος καταγραφής, μέτρηση, κωδικός αισθητήρα>. Στην πλευρά του κεντρικού υπολογιστή, καταφθάνουν συνεχώς πλειάδες τέτοιας μορφής με μετρήσεις που προκύπτουν

από τους αισθητήρες. Στην ουσία ο υπολογιστής δέχεται ένα ρεύμα δεδομένων.

Βάσει του παραδείγματος γίνεται εμφανής ο τρόπος που λειτουργεί ένα ρεύμα δεδομένων. Το ρεύμα αποτελείται από πλειάδες τιμών που συνήθως εμπεριέχουν τον χρόνο καταγραφής της πλειάδας και ένα αναγνωριστικό του αποστολέα, αφού ενδέχεται να είναι περισσότεροι του ενός. Οι πλειάδες φτάνουν στην πλευρά του δέκτη σε άγνωστες χρονικές στιγμές και η σειρά με την οποία αποστέλλουν οι αποστολείς είναι ακανόνιστη.

Έτσι, το ρεύμα δεδομένων αποτελεί ακολουθία πλειάδων της μορφής:

$$\langle t_i, S_i, X \rangle$$

όπου

- t_i : η χρονική στιγμή που καταγράφηκε η πληροφορία (προαιρετικό),
- S_i : αναγνωριστικό του αποστολέα (όταν υπάρχουν περισσότεροι του ενός),
- X : η πληροφορία η οποία μπορεί και η ίδια να είναι μια πλειάδα από τιμές.

Γενικά, τα κυριότερα χαρακτηριστικά ενός ρεύματος δεδομένων είναι:

- άγνωστο μέγεθος,
- δυναμικά δεδομένα,
- άγνωστος και μεταβλητός ρυθμός άφιξης,
- ανάγκη για επεξεργασία πραγματικού χρόνου.

Επίσης ο τρόπος μετάδοσης των δεδομένων (ενσύρματος/ασύρματος), τα προβλήματα που προκαλούνται κατά τη μετάδοση αλλά και η ανακρίβεια των δεδομένων είναι παράγοντες που χαρακτηρίζουν ένα ρεύμα δεδομένων. Ακολουθεί πιο λεπτομερής περιγραφή των χαρακτηριστικών αυτών.

i) Άγνωστος χρόνος άφιξης δεδομένων

Οι πηγές στέλνουν πακέτα δεδομένων κατά δική τους βούληση. Ακόμη και στην περίπτωση που το σύστημα επιβάλλει ένα μέγιστο χρονικό διάστημα κατά το οποίο κάθε πηγή οφείλει να στείλει ένα πακέτο πληροφορίας, η ακριβής στιγμή που αυτό θα ληφθεί δεν μπορεί να είναι γνωστή από πριν. Αυτό οφείλεται τόσο στην αυτονομία των πηγών όσο και στο γεγονός ότι τα πακέτα πρέπει να μεταφέρονται με κάποιο τρόπο στο δίκτυο (συνήθως με ασύρματη μετάδοση). Ο φόρτος του δικτύου επηρεάζει σε μεγάλο βαθμό την δρομολόγηση της πληροφορίας με απρόβλεπτο τρόπο.

Μια εφαρμογή πρέπει να είναι σε έτοιμότητα να λάβει κάποιο πακέτο συνεχώς, αφού ο χρόνος άφιξής του είναι άγνωστος.

ii) Άγνωστος ρυθμός άφιξης δεδομένων

Δεδομένου ότι οι πηγές στέλνουν πακέτα σε αυθαίρετους χρόνους, η ροή της πληροφορίας στο δέκτη δεν είναι δυνατόν να ελεγχθεί. Κάποια στιγμή μπορεί να μην υπάρχει λήψη πακέτων από την πλευρά του δέκτη ενώ σε κάποια άλλη πολλές πηγές μπορεί να στέλνουν ταυτόχρονα πακέτα πληροφορίας. Η αυθαιρεσία αυτή μπορεί να προκαλέσει καταιγισμό πληροφορίας στο δέκτη σε κάποιες χρονικές στιγμές.

Η εφαρμογή διαχείρισης ενός ρεύματος δεδομένων πρέπει να είναι σε θέση να παραλάβει όσα πακέτα καταφθάνουν, άσχετα με το ρυθμό που αυτό συμβαίνει, ώστε να αποφευχθεί η απώλεια πληροφορίας λόγω υπερφόρτωσης του δέκτη.

iii) Μεγάλος όγκος δεδομένων

Επειδή πακέτα πληροφορίας στέλνονται συνεχώς από πολλαπλές πηγές, η ποσότητα πληροφορίας που καταφθάνει από ένα ρεύμα δεδομένων είναι ενδεχομένως τεράστια. Ενδεικτικά, αν θεωρήσουμε μια εφαρμογή που παρακολουθεί 100000 αισθητήρες, και κάθε αισθητήρας στέλνει μία μέτρηση κάθε δευτερόλεπτο, σε μια ώρα ο δέκτης θα έχει συγκεντρώσει αριθμό πακέτων της τάξεως των 360 000 000 μετρήσεων!

Είναι προφανές πως η γρήγορη επεξεργασία της πληροφορίας ενός ρεύματος δεδομένων δυσχεραίνει πολύ γρήγορα και πρέπει να βρεθούν τεχνικές προκειμένου αυτό να γίνει εφικτό. Ενδεικτικά, η ύπαρξη γρήγορων ευρετηρίων και η απόρριψη πληροφορίας μειώνοντας την ακρίβεια των απαντήσεων εφαρμόζονται ως επί το πλείστον [22, 6].

iv) Περιορισμοί εύρους ζώνης

Η μετάδοση της πληροφορίας από τις πηγές στο δέκτη γίνεται συνήθως με χρήση ασύρματων τηλεπικοινωνιακών δικτύων. Η μεγάλη ποσότητα της πληροφορίας επιφέρει και μεγάλη κατανάλωση πόρων προκειμένου να γίνει η μεταφορά της από τις πηγές στο δέκτη. Αυτό σημαίνει αυξημένο κόστος σε χρήμα αφού μέγαλο εύρος ζώνης πρέπει να χρησιμοποιηθεί.

Το σύστημα πρέπει να λαμβάνει υπόψη του την επιβάρυνση που δέχονται οι πηγές καθώς μεταδίδουν πληροφορία και να προσπαθεί να μειώνει τις μη αναγκαίες ενημερώσεις στο ελάχιστο. Το ρεύμα δεδομένων υπόκειται σε απόρριψη πληροφορίας που ενδεχομένως να μειώνει την ακρίβεια των παραγόμενων αποτελεσμάτων. Ένα ισοζύγιο αυτών των δύο χρειάζεται να επιτευχθεί προκειμένου να υπάρχει ικανοποιητική ακρίβεια σε συνδιασμό με περιορισμένη κατανάλωση πόρων του δικτύου.

v) Διακοπτόμενη λειτουργία

Όπως έχει προαναφερθεί, οι πηγές χρησιμοποιούν διαύλους επικοινωνίας για να μεταδώσουν τα πακέτα τους στο δέκτη του ρεύματος δεδομένων. Πολλές είναι οι περιπτώσεις όπου ένας δέκτης λόγω περιορισμένης κάλυψης του δικτύου, φυσικών εμποδίων (λ.χ σήραγγες) ή μειωμένης επάρκειας σε ενέργεια δεν είναι σε θέση να στείλει κάποιο πακέτο. Ακόμα, λόγω συγκρούσεων κατά τη μεταφορά, κάποια πακέτα πληροφορίας είναι πολύ πιθανόν να χαθούν.

Ο δέκτης του ρεύματος, πρέπει να γνωρίζει από πριν πως σε αρκετές περιπτώσεις κάποια πακέτα δεν πρόκειται ποτέ να φτάσουν τον προορισμό τους με αποτέλεσμα να υπάρχει ελλιπής πληροφορία. Κατ' επέκταση, πρέπει να είναι σε θέση να αντιμετωπίσει ή να προνοήσει για τέτοιες περιπτώσεις ώστε η σωστή εξαγωγή συμπερασμάτων να είναι εξασφαλισμένη.

vi) Αβεβαιότητα πληροφορίας

Κάθε πηγή στέλνει πακέτα πληροφορίας. Η πληροφορία που μεταφέρεται, ιδανικά θα ήταν η ίδια με την πραγματική (λ.χ αν αυτή ήταν μια μέτρηση ενός φυσικού φαινομένου). Αυτό όμως δεν συμβαίνει και στην πραγματικότητα λόγω:

1. σφαλμάτων μέτρησης,
2. θορύβου διαύλου επικοινωνίας.

Η χρήση τηλεπικοινωνιακών διαύλων για μεταφορά της πληροφορίας είναι επιρρεπής σε θόρυβο λόγω παρεμβολών του περιβάλλοντος. Ακόμα, κατά τη δημιουργία του πακέτου πληροφορίας η πηγή μπορεί να παράγει από μόνη της σφάλματα μετρήσεων. Ο δέκτης πρέπει να λαμβάνει υπόψη την αβεβαιότητα που χαρακτηρίζει την πληροφορία που λαμβάνει και όταν είναι αναγκαίο να μεταβάλλει τα συμπεράσματα του βάσει αυτής [14].

1.1.2 Ερωτήματα σε ρεύματα δεδομένων

Σε κάθε σύστημα το οποίο εισάγεται πληροφορία, σημασία έχουν τα συμπεράσματα που μπορεί να εξάγει. Ένα συμπέρασμα μπορεί να ταυτιστεί με ένα ερώτημα αφού θεωρώντας πως το σύνολο των δεδομένων συνιστά μια βάση δεδομένων, τα ερωτήματα σε αυτήν προκαλούν εξαγωγή συμπερασμάτων.

Ένα ρεύμα δεδομένων διαφοροποιείται από τις κλασικές βάσεις δεδομένων λόγω της δυναμικότητας των δεδομένων και της μη στατικότητας που τα χαρακτηρίζει. Έτσι εκτός από τα κλασικά ερωτήματα που χρησιμοποιούνται σε συνήθεις βάσεις δεδομένων ένα ρεύμα δεδομένων λόγω της συνεχούς ροής πληροφορίας απαιτεί και άλλου τύπου ερωτήματα.

Γενικά, σε ένα ρεύμα δεδομένων μπορούν να γίνουν δύο είδη ερωτημάτων:

- ερωτήματα στιγμιοτύπου (one-time queries),

- ερωτήματα διαρκείας (continuous queries).

Ερωτήματα στιγμιότυπου

Τα ερωτήματα στιγμιότυπου μοιάζουν στην ουσία με τα ερωτήματα που γίνονται σε κλασικές βάσεις δεδομένων. Ένα ερώτημα εκτελείται πάνω στα δεδομένα που έχουν ληφθεί μέχρι στιγμής (το στιγμιότυπο της βάσης) και εξάγονται τα απαραίτητα συμπεράσματα. Τότε το ερώτημα έχει τερματίσει. Αυτό συμβαίνει και στις κλασικές βάσεις δεδομένων όπου τα δεδομένα είναι στατικά, άρα και τα ερωτήματα που εκτελούνται δεν χρειάζεται παρά να ασχολούνται μόνο με κάποιο στιγμιότυπο της βάσης.

Ερωτήματα διαρκείας

Τα ερωτήματα διαρκείας είναι αυτά που έχουν τη μεγαλύτερη σημασία στην επεξεργασία ενός ρεύματος δεδομένων. Η δυναμική φύση των δεδομένων και η συνεχής ροή τους (η ποσότητα της πληροφορίας δεν είναι σταθερή αλλά αυξάνεται συνεχώς) γεννά την ανάγκη για ύπαρξη ερωτημάτων που μεταβάλλουν τις απαντήσεις τους συμβαδίζοντας με την μεταβολή των δεδομένων.

Δεδομένου ότι ένα ρεύμα δεδομένων έχει άπειρο μέγεθος λόγω της συνεχούς άφιξης δεδομένων, αλλά και κυμαινόμενους ρυθμούς άφιξης των στοιχείων, είναι εμφανές πως η συσσώρευση της πληροφορίας προκαλεί προβλήματα. Κατά την επεξεργασία της, λαμβάνοντας υπόψη και το γεγονός ότι συνεχώς φτάνουν ανανεωμένες τιμές για μεγέθη που παρουσιάστηκαν και προηγουμένως, φαίνεται πως η επεξεργασία της πληροφορίας είναι σημαντικό να γίνεται άμεσα καθώς καταφθάνουν τα δεδομένα και ιδανικά σε χρόνο μικρότερο από τον ρυθμό με τα οποία αυτό συμβαίνει προκειμένου να υπάρξει απόκριση του συστήματος σε πραγματικό χρόνο.

Σε μία κλασική βάση δεδομένων, η εκτέλεση ενός ερωτήματος γίνεται με την παρουσίαση του ερωτήματος στη βάση, την αναζήτηση των εγγραφών που τηρούν τις συνθήκες και επιστροφή των τιμών που εντοπίστηκαν. Αυτή η προσέγγιση ταιριάζει με τη στατική φύση των δεδομένων. Η δυναμική φύση των δεδομένων του ρεύματος δεδομένων, κάνει την εκτέλεση συγκεκριμένων ερωτημάτων με την κλασική προσέγγιση ανεπαρκή. Ως παράδειγμα, μπορεί να ληφθεί υπόψη το ερώτημα:

Πόσες μετοχές είχαν αύξηση της τιμής τους μεγαλύτερη της τάξης του 5%;

Σε μία κλασική βάση δεδομένων, το ερώτημα θα εκτελείτο μία φορά και μία απάντηση θα επιστρεφόταν. Στο επόμενο δευτερόλεπτο όμως, (ενδεχομένως) ένα νέο πακέτο πληροφορίας θα κατέφθανε από το ρεύμα για κάποια νέα τιμή μετοχής. Αν η μετοχή είχε προηγουμένως αύξηση οριακά κάτω από 5%,

ενώ τώρα αυτή βρέθηκε να το έχει ξεπεράσει, το αποτέλεσμα που δόθηκε μόλις πριν ένα δευτερόλεπτο θα ήταν πλέον λανθασμένο! Η προφανής λύση θα ήταν να εκτελείται το ερώτημα στο σύστημα κάθε δευτερόλεπτο προκειμένου να επικαιροποιείται η απάντηση λαμβάνοντας υπόψη κάθε νέα τιμή μετοχής. Είναι προφανές πως το κόστος εκτέλεσης μιας τέτοιας διαδικασίας είναι απαγορευτικό, πόσο μάλλον αν τα ερωτήματα στη βάση είναι πολύ περισσότερα από ένα.

Ο δέκτης, προκειμένου να δίνει έγκυρα και έγκαιρα αποτελέσματα, εκτελεί ερωτήματα διαρκείας. Ένα ερώτημα διαρκείας υποβάλλεται από το χρήστη και μέχρι να ακυρωθεί η λειτουργία του το σύστημα δίνει σε πραγματικό χρόνο την απάντηση που ισχύει σε κάθε χρονική στιγμή. Η μέθοδος για να γίνει αυτό είναι να εξετάζεται κάθε νέο πακέτο πληροφορίας καθώς αυτό φτάνει στο δέκτη και όχι να γίνεται ομαδικός έλεγχος σε όλες τις εγγραφές για την ικανοποίηση κάποιου ερωτήματος. Όταν μία πηγή στείλει ένα νέο πακέτο πληροφορίας (λ.χ νέα τιμή μιας μετοχής), το σύστημα ελέγχει την πληροφορία αυτή. Αν η πληροφορία αυτή ανήκε ως απάντηση σε κάποιο ενεργό ερώτημα και είναι ακόμα έγκυρη, τότε δεν αλλάζει τίποτα. Αλλιώς, τροποποιείται η απάντηση του ερωτήματος που επηρεάζεται (προστίθεται η συγκεκριμένη μετοχή στο σύνολο). Στη συνέχεια, το πακέτο ελέγχεται αν ικανοποιεί κάποιο άλλο τρέχον ερώτημα και αν ναι, τότε επιστρέφεται και ως μέρος αυτής της απάντησης.

Μεγάλη έρευνα έχει γίνει για την διατύπωση και βελτιστοποίηση των ερωτημάτων διαρκείας. Όσο πολύπλοκο και αν είναι ένα ερώτημα, συνήθως ανάγεται σε κάποιο σύνολο βασικών τελεστών, όπως σύνδεση (*join*), συνάθροιση (*aggregation*), επιλογή (*selection*), απαλοιφή διπλοτύπων (*duplicate elimination*) κλπ.

1.1.3 Παράθυρα

Ένα ρεύμα δεδομένων δεν έχει προκαθορισμένο μέγεθος ή ρυθμό άφιξης των πακέτων του. Το πακέτα φτάνουν σε αυθαίρετες χρονικές στιγμές και ενδεχομένως επ'άπειρον ή αλλιώς για όσο χρόνο αυτά είναι απαραίτητα για την τροφοδότηση ενός συστήματος. Αυτό σημαίνει πως η αποθήκευση όλου του όγκου της πληροφορίας καθίσταται αδύνατη λόγω του μεγέθους της. Επίσης, οι αυθαίρετες χρονικές στιγμές άφιξης τους ενίοτε δημιουργούν καταιγισμό δεδομένων στο σύστημα, ενώ σε άλλες περιπτώσεις παρουσιάζεται απουσία πληροφορίας για άγνωστο χρονικό διάστημα.

Ένα σύστημα που λαμβάνει ένα ρεύμα δεδομένων και λειτουργεί σε πραγματικό χρόνο, συνήθως ενδιαφέρεται να δίνει απαντήσεις που αφορούν τις πρόσφατες τιμές που ελήφθησαν. Τα παράθυρα στηρίζονται σε αυτή τη λογική προκειμένου να μειώνουν την πληροφορία που πρέπει να ελεγχθεί για την υπολογισμό κάποιου ερωτήματος.

Ένα παράθυρο (*window*), απομονώνει ένα πεπερασμένο πλήθος στοιχείων από το απειράριθμο σύνολο που λαμβάνεται από ένα ρεύμα δεδομένων και

η επεξεργασία γίνεται μόνο σε αυτό. Συνήθως, επιλέγονται πιο πρόσφατα στοιχεία έναντι παλαιότερων, αφού οι εφαρμογές ενδιαφέρονται κυρίως για την πιο πρόσφατη πληροφορία που είναι διαθέσιμη.

Καθώς εισέρχεται το ρεύμα δεδομένων στον δέκτη, αυτός χρησιμοποιεί ένα παράθυρο για να λαμβάνει υπόψη του στα διάφορα ερωτήματα μόνο τα πακέτα που εμπίπτουν εντός παραθύρου. Όλα τα υπόλοιπα, είτε απορρίπτονται είτε τοποθετούνται σε δευτερεύουσα μνήμη για πιθανή μελλοντική αναφορά.

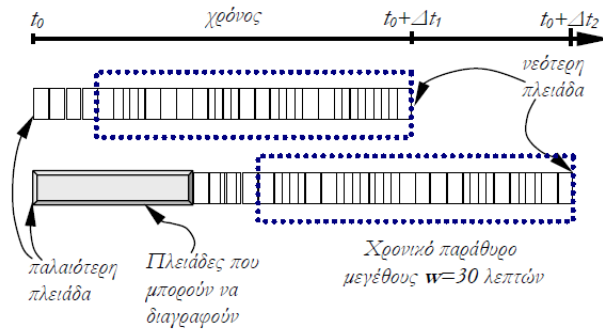
Υπάρχουν διάφορα είδη παραθύρων αλλά τα κυριότερα εξ' αυτών είναι τα:

- Παράθυρα σταθερής αρχής ή παράθυρα οροσήμου (landmark windows)

Τα παράθυρα οροσήμου, καθορίζουν ως αρχή του παραθύρου κάποια στιγμή του παρελθόντος αλλά το τέλος τους μεταβάλλεται καθώς εξελίσσεται ο χρόνος, με αποτέλεσμα το μέγεθος του παραθύρου να αυξάνεται συνεχώς.

- Κυλιόμενα παράθυρα (sliding windows)

Τα κυλιόμενα παράθυρα έχουν μεταβαλλόμενη αρχή και τέλος οι οποίες μετακινούνται με την χρονική εξέλιξη των στοιχείων. Το τέλος του παραθύρου βρίσκεται στην πιο πρόσφατη χρονική στιγμή και η αρχή στη χρονική στιγμή του παρελθόντος που απέχει από το παρόν όσο είναι το εύρος του παραθύρου.



Σχήμα 1.1: Λογική λειτουργίας παραθύρων σε ρεύματα δεδομένων. (Πηγή: [15])

1.1.4 Κυριότερα συστήματα διαχείρισης ρευμάτων δεδομένων

Η φύση των ρευμάτων δεδομένων απαιτεί ειδική μεταχείριση που δεν παρέχεται από τα κλασικά συστήματα διαχείρισης βάσεων δεδομένων. Για το σκοπό αυτό έχουν αναπτυχθεί πρωτότυπα συστήματα διαχείρισης ρευμάτων

δεδομένων ως αποτέλεσμα ερευνητικής εργασίας που γίνεται τα τελευταία χρόνια. Κυριότερα από αυτά είναι:

- STREAM
- AURORA
- TelegraphCQ
- Gigascope
- NiagaraCQ

Κάποια από αυτά αποτελούν συστήματα γενικού σκοπού (STREAM, Aurora), ενώ άλλα απευθύνονται περισσότερο σε δικτυακές εφαρμογές και δίκτυα αισθητήρων (TelegraphCQ, NiagaraCQ, Gigascope). Σε αυτά τα συστήματα, γίνεται χρήση ειδικών τελεστών που μπορούν να διαχειριστούν καλύτερα την δυναμική φύση των δεδομένων και προτείνονται επεκτάσεις στις γλώσσες ανάκτησης δεδομένων [6]. Παραδείγματα αποτελούν οι επεκτάσεις της SQL: GSQL απο το σύστημα Gigascope και CQL (Continuous Query Language) από το STREAM.

Το σύστημα STREAM (STanford stREam Data Management) ([20]) αποτελεί μία αξιοσημείωτη προσπάθεια του πανεπιστημίου του Stanford. Αντί να γίνει προσαρμογή κάποιου κλασικού συστήματος βάσεων δεδομένων όπως συνηθιζόταν, επιλέχθηκε αντ' αυτού να αναπτυχθεί ένα σύστημα με γνώμονα εξαρχής τα ρεύματα δεδομένων. Στα πλαίσια της ανάπτυξης υλοποιήθηκε και η γλώσσα CQL που αποτελεί υπερσύνολο της γλώσσας SQL προσθέτοντας εξειδικευμένες δομές για υποστήριξη κυλιόμενων παραθύρων και δειγματοληψίας.

Το σύστημα TelegraphCQ στηρίχθηκε στο γνωστό ΣΔΒΔ PostgreSQL κάνοντας απαραίτητες προσθήκες σε δομές παραθύρων, βελτιστοποίησης εκτέλεσης ερωτημάτων, προσαρμοστικότητα σε κυμαινόμενους ρυθμούς άφιξης δεδομένων κ.α. Το σύστημα NiagaraCQ στοχεύει σε δεδομένα μορφής XML για διαδικτυακή χρήση, ενώ το σύστημα Gigascope υλοποιήθηκε με σκοπό την διαχείριση τηλεπικοινωνιακών δικτύων και δικτύων υπολογιστών.

1.2 Χωρικά και Χωροχρονικά δεδομένα

Τα δεδομένα που δέχεται μια εφαρμογή μπορεί να είναι οποιουδήποτε τύπου: μουσικά αρχεία, τιμές προϊόντων, προσωπικά στοιχεία πολιτών μιας χώρας, ή ακόμα και το απλό πάτημα ενός πλήκτρου. Στο πλαίσιο της εργασίας αυτής τα δεδομένα που τίθενται προς επεξεργασία είναι χωρικά, γι' αυτό και παρουσιάζονται στην συνέχεια. Το ενδιαφέρον για το χρόνο ισχύος αυτών των δεδομένων πέρα από τα ίδια τα δεδομένα, τα καθιστά χωροχρονικά.

1.2.1 Χωρικοί τύποι δεδομένων

Χωρικά δεδομένα (spatial data) είναι αυτά που αφορούν θέσεις (σημεία) ή περιοχές στο χώρο. Ανάλογα με την εφαρμογή, τα δεδομένα μπορούν να περιγράφουν περιοχές στο δισδιάστατο και τρισδιάστατο Ευκλείδειο χώρο ή ακόμα και σε άλλους είδους χωρικές συντεταγμένες. Παραδείγματα τέτοιων δεδομένων μπορεί να είναι η σημειακή θέση ενός αντικειμένου στο χώρο, μια περιοχή ακανόνιστου σχήματος που να περιγράφει μια λίμνη ή μια ευθεία που να δείχνει τις θέσεις από τις οποίες πέρασε ένα αυτοκίνητο.

Τα χωρικά δεδομένα χρησιμοποιούνται για να περιγράψουν στοιχεία που αφορούν χωρικές διαστάσεις. Η περιγραφή των στοιχείων μπορεί να γίνει με τη χρήση:

- σημείων,
- γραμμών,
- πολύγωνων.

i) Σημεία

Τα σημεία χρησιμοποιούνται για να περιγράψουν σημειακές θέσεις στο χώρο. Γενικά, τα αντικείμενα στο χώρο καταλαμβάνουν κάποια έκταση και δεν είναι σημειακά.

Η θέση ενός αντικειμένου είναι στην ουσία το σύνολο των σημείων του χώρου που καταλαμβάνει. Για παράδειγμα η θέση ενός κτηρίου, θεωρώντας δισδιάστατο χώρο, ισοδυναμεί με το σχήμα που δημιουργεί στο επίπεδο η προβολή του κτηρίου αυτού. Αν το ενδιαφέρον μιας εφαρμογής στρέφεται στις θέσεις των κτηρίων σε επίπεδο τετραγώνου, τότε ο χώρος που καταλαμβάνουν τα κτήρια έχει σημασία. Εντούτοις, οι περισσότερες εφαρμογές ενδιαφέρονται για τη θέση αντικειμένων σε κλίμακα χάρτη. Σε τέτοιες διαστάσεις, ο χώρος που καταλαμβάνει ένα αντικείμενο είναι ασήμαντος και είναι αρκετό να υπολογίζεται ένα σημείο αντιπροσωπευτικό για όλο το αντικείμενο. Έτσι ο όγκος του αντικειμένου εκφυλίζεται σε μια σημειακή θέση.

ii) Γραμμές

Με τη χρήση γραμμών μπορούν να αναπαρασταθούν περιοχές με κοινά χαρακτηριστικά όπως μια ισοϋψής καμπύλη σε ένα χάρτη ή τα σύνορα μεταξύ χωρών.

Επίσης, για την αναπαράσταση της τροχιάς που ακολούθησε ένα αντικείμενο μπορεί να χρησιμοποιηθεί μια γραμμή. Οι γραμμές συνενώνουν μια ακολουθία από σημεία και καταδεικνύουν τη διαδρομή στην οποία κινήθηκε το αντικείμενο στο χώρο.

iii) Πολύγωνα

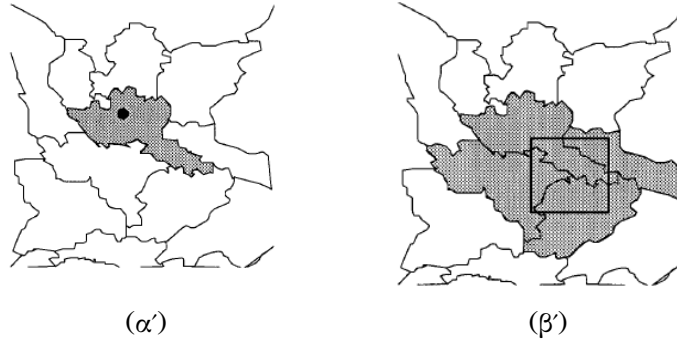
Για την αναπαράσταση περιοχών ή επιφανειών χρησιμοποιούνται πολύγωνα. Επίσης, όταν το μέγεθος ενός αντικειμένου έχει ιδιαίτερη σημασία τότε το αντικείμενο περιγράφεται από ένα πολύγωνο αντί ενός σημείου.

Για παράδειγμα, με τη χρήση πολυγώνων μπορεί να αναπαρασταθεί η επιφάνεια που καλύπτει μια λίμνη ή ένα μεγάλο πλοίο που βρίσκεται στη θάλασσα.

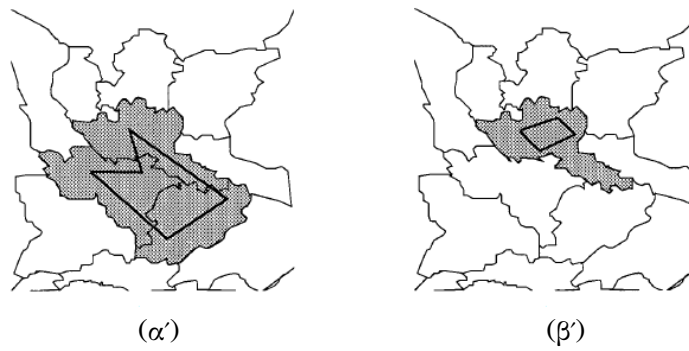
1.2.2 Ερωτήματα σε χωρικά δεδομένα

Παρακάτω παρουσιάζονται τα συνηθέστερα ερωτήματα που γίνονται σε στατικά χωρικά δεδομένα.

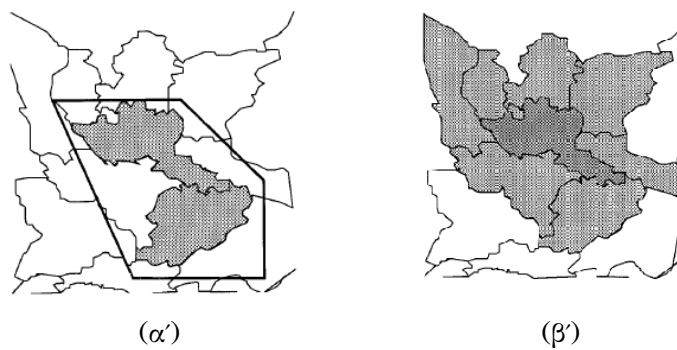
1. *Ακριβής αναζήτηση αντικειμένων (exact match query)*: Εύρεση των αντικειμένων που ταυτίζονται με το δοσμένο.
2. *Σημείου (point query)*: Εύρεση όλων των αντικειμένων που η χωρική τους έκταση περιλαμβάνει το δοσμένο σημείο (σχήμα 1.2α').
3. *Περιοχής (range query)*: Εύρεση των αντικειμένων που περιέχονται στη δοσμένη περιοχή, η οποία μπορεί να είναι ορθογώνιο, κύκλος ή οποιοδήποτε πολύγωνο (σχήμα 1.2β').
4. *Τομής (intersection query)*: Εύρεση των αντικειμένων που τέμνουν το δοσμένο (έχουν τουλάχιστον ένα κοινό σημείο) (σχήμα 1.3α').
5. *Περίφραξης (enclosure query)*: Εύρεση των αντικειμένων που περικλείουν το δοσμένο (σχήμα 1.3β').
6. *Περιεχομένου (containment query)*: Εύρεση των αντικειμένων που περιέχονται πλήρως στο δοσμένο (σχήμα 1.4α').
7. *Γειτνίασης (adjacency query)*: Εύρεση των αντικειμένων που συνορεύουν με το δοσμένο (σχήμα 1.4β').
8. *Εγγύτερου γείτονα (nearest-neighbor query)*: Εύρεση των αντικειμένων που βρίσκονται κοντινότερα στο δοσμένο (ή τα k κοντινότερά του).
9. *Χωρικής σύνδεσης (spatial join)*: για δύο δοσμένα σύνολα αντικειμένων, βρες τα ζεύγη αντικειμένων από το ένα και το άλλο που ικανοποιούν μια προκαθορισμένη συνθήκη (λ.χ. απόσταση μικρότερη από 100m). Η συνθήκη μπορεί να είναι ή να περιέχει άλλα ερωτήματα.



Σχήμα 1.2: α)ερώτημα σημείου, β)ερώτημα περιοχής. (Πηγή: [4])



Σχήμα 1.3: α)ερώτημα τομής, β)ερώτημα περίφραξης. (Πηγή: [4])



Σχήμα 1.4: α)ερώτημα περιεχομένου, β)ερώτημα γειτνίασης. (Πηγή: [4])

1.2.3 Χωρικά ευρετήρια

Οι επιδόσεις είναι ζήτημα μείζονος σημασίας σε κάθε σύγχρονη εφαρμογή. Προκειμένου να επιτευχθεί αύξηση της απόδοσης των εφαρμογών που διαχειρίζονται χωρικά δεδομένα, έχουν επινοηθεί ειδικά ευρετήρια για την αποθήκευσή τους [22]. Τα ευρετήρια αυτά αποσκοπούν στην γρήγορη απάντηση ερωτημάτων, ευνοώντας συνήθως περισσότερο ένα είδος ερωτήματος και επιβαρύνοντας κάποιο άλλο.

Διακρίνονται δύο βασικές ομάδες ευρετηρίων: όσα αφορούν σημειακά αντικείμενα (*μονοδιάστατα ευρετήρια*) και όσα στοχεύουν στην καταχώρηση ορθογωνίων/σφαιρικών/πολυεδρικών αντικειμένων (*πολυδιάστατα ευρετήρια*). Μερικά από τα πιο ευρέως χρησιμοποιούμενα ευρετήρια είναι τα εξής:

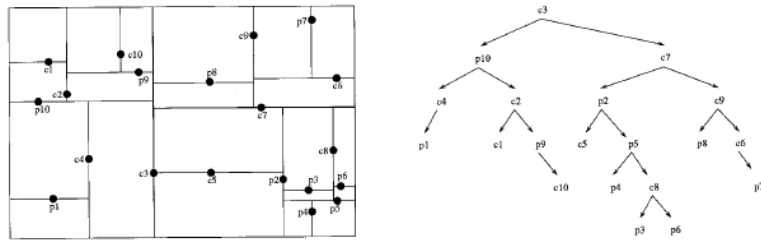
Μονοδιάστατα ευρετήρια

Με την τεχνική του *κατακερματισμού*, ο χώρος διαιρείται σε τετραγωνικές περιοχές όπου κάθε περιοχή εμπίπτει σε έναν κάδο της δομής. Κάθε σημείο, ανάλογα με τη θέση του στο χώρο, ανήκει σε κάποια περιοχή και άρα εντάσσεται στον ανάλογο κάδο. Σημεία εντός της ίδιας περιοχής ανατίθενται στον ίδιο κάδο. Υπάρχουν διάφορες παραλλαγές ευρετηρίων που στηρίζονται στον κατακερματισμό, όπως ευρετήρια με γραμμικό και επεκτεινόμενο κατακερματισμό.

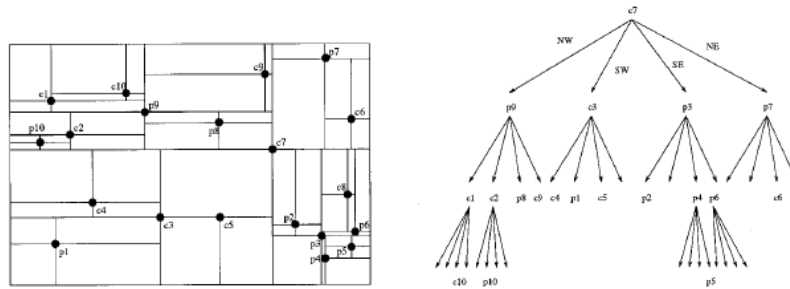
Το *B-δέντρο* είναι πιο αποδοτικό. Διαιρεί το χώρο ιεραρχικά χρησιμοποιώντας ισοζυγισμένη δενδρική δομή. Και πάλι, ο χώρος τεμαχίζεται σε διαστήματα, αλλά κάθε διάστημα ξανατεμαχίζεται σε μικρότερα και ούτω καθεξής μέχρι να προκύψει ένα ελάχιστο τετραγωνικό τμήμα. Ψηλότερα επίπεδα του δέντρου αντιστοιχούν σε μεγαλύτερα διαστήματα και χαμηλότερα επίπεδα σε μικρότερα τμήματα που εμπίπτουν στους γονείς τους. Στα φύλλα του δέντρου βρίσκονται οι σημειακές θέσεις των αντικειμένων. Τμήματα με κοινό γονέα είναι γειτονικά τμήματα.

Το *k-d-δέντρο* (σχήμα 1.5) είναι μετεξέλιξη του B-δέντρου και λειτουργεί με παρόμοιο τρόπο, μόνο που κάθε τμήμα δεν τεμαχίζεται ισομορφικά με τα υπόλοιπα. Αντίθετα, ο τεμαχισμός γίνεται ανάλογα με την τοπολογία των σημείων μέσα στο τμήμα. Οι τεμαχισμοί γίνονται σε σημεία έτσι ώστε όποτε μια περιοχή αδειάσει τότε τερματίζεται και ο περαιτέρω τεμαχισμός της. Το *k-d-δέντρο* υλοποιείται με δομή δυαδικού δέντρου.

Το *τετραδικό δέντρο* (*quad tree*) (σχήμα 1.6) αποτελεί στενό συγγενή του *k-d-δέντρου* μόνο που τώρα δεν χρησιμοποιείται δυαδικό δέντρο, αλλά κάθε κόμβος (τμήμα) μπορεί να έχει αριθμό υποπεριοχών ίσο με μια σταθερά *d*. Αν μια περιοχή περιέχει αντικείμενα, τότε διαιρείται ξανά σε *d* υποτμήματα μέχρι ένα ελάχιστο μέγεθος τμήματος. Το τετραδικό δέντρο τεμαχίζει περιοχές με αυξημένη συγκέντρωση σημείων αρκετά γρήγορα, ενώ ο τεμαχισμός άδειων περιοχών σταματά μόλις αυτές εντοπιστούν.



Σχήμα 1.5: k-d-Δέντρο (Πηγή: [4])



Σχήμα 1.6: Τετραδικό Δέντρο (Πηγή: [4])

Πολυδιάστατα ευρετήρια

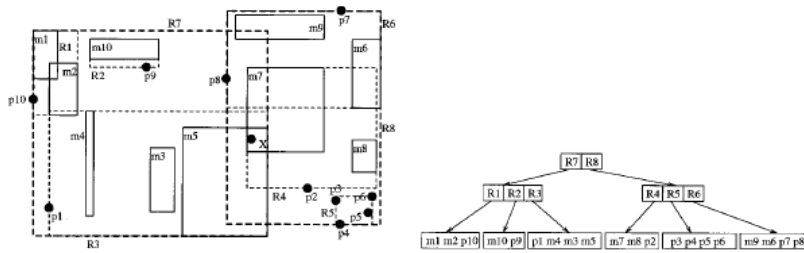
Τα πολυδιάστατα ευρετήρια γενικεύουν την ιδέα των μονοδιάστατων ώστε ο τεμαχισμός του χώρου να γίνεται με γνώμονα αντικείμενα που καταλαμβάνουν χώρο και όχι μονάχα σημεία.

Το *R*-δέντρο (σχήμα 1.7) τεμαχίζει το χώρο σε περιοχές που περιέχουν αντικείμενα με χωρική έκταση και τα οργανώνει σε συστάδες κοντινών μεταξύ τους αντικειμένων. Ο τεμαχισμός εξαρτάται από την συνάφεια των αντικειμένων καθώς επίσης και από τη σειρά εισαγωγής τους στο ευρετήριο. Η διαδικασία συνεχίζεται ώστε να δημιουργούνται υποτιμήματα που να μπορούν να συμπεριλάβουν όλο και μικρότερα αντικείμενα.

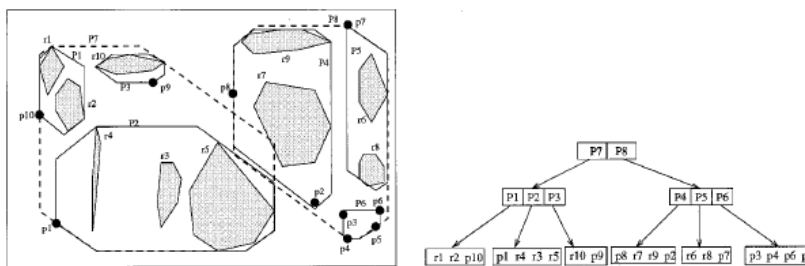
Το *P*-δέντρο (σχήμα 1.8) προσπαθεί να βελτιώσει την χρήση της μνήμης εκτελώντας τεμαχισμούς που δεν στηρίζονται σε τετραγωνικά τμήματα αλλά σε ακανόνιστα πολύεδρα. Στόχος είναι να συμπεριλαμβάνει κάθε φορά όσο περισσότερα χωρικά αντικείμενα είναι δυνατό, μειώνοντας τον ακάλυπτο χώρο σε κάθε τμήμα. Υπάρχουν αρκετές παραλλαγές των δομών αυτών που προσπαθούν να εντοπίσουν καλύτερους τρόπους τεμαχισμού. Παραδείγματα αποτελούν τα *R*⁺-δέντρα και *R*^{*}-δέντρα.

1.2.4 Χωροχρονικά δεδομένα

Χωροχρονικά δεδομένα είναι δεδομένα που συνδυάζουν χωρική πληροφορία με τη διάσταση του χρόνου. Όταν το ενδιαφέρον στρέφεται σε αντικείμενα



Σχήμα 1.7: R-Δέντρο (Πηγή: [4])



Σχήμα 1.8: P-Δέντρο (Πηγή: [4])

που δεν είναι στατικά, τότε κρίνεται απαραίτητη η χρήση χωροχρονικής πληροφορίας. Στατικά είναι αντικείμενα όπως ένα κτήριο ή μια λίμνη, ενώ κινούμενα είναι οντότητες όπως ένα αυτοκίνητο ή ένας πεζός.

Για παράδειγμα, η θέση ενός αντικείμενου στο χώρο αποτελεί χωρική πληροφορία, ενώ η γνώση της θέσης αυτής σε συνδυασμό με το πότε ίσχυε, δηλαδή ποια ακριβώς χρονική στιγμή το αντικείμενο βρισκόταν σε εκείνη τη θέση, αποτελεί χωροχρονική πληροφορία.

Προκειμένου να απαντηθούν πιο πολύπλοκα ερωτήματα που αφορούν την κίνηση αντικειμένων είναι απαραίτητο τα δεδομένα που διατίθενται να είναι χωροχρονικής φύσεως. Για παράδειγμα, γνωρίζοντας τη διαδρομή που ακολούθησαν δύο αντικείμενα στο χώρο αγνοώντας το χρόνο που αυτή έγινε, μπορεί να εξαχθεί το συμπέρασμα αν τα δύο αυτά αντικείμενα πέρασαν από ίδια σημεία. Εντούτοις, γνωρίζοντας και το χρόνο που εκτελέστηκε αυτή η κίνηση, είναι δυνατόν να εντοπιστεί όχι μόνο αν τα δύο αντικείμενα πέρασαν από ίδια σημεία αλλά και αν αυτά διασταυρώθηκαν κάποια στιγμή ή με πόση χρονική διαφορά προσπέρασε το ένα το άλλο.

Επίσης, χρησιμοποιώντας την επιπλέον πληροφορία του χρόνου, μπορούν να διαχωριστούν τα δεδομένα βάση της αρχαιότητάς τους, και να γίνει δυνατή η χρήση παραθύρων και η απάντηση ερωτημάτων διαρκείας όπως παρουσιάστηκαν προηγουμένως.

1.3 Διαχείριση κινούμενων αντικειμένων

Όταν το ενδιαφέρον μιας εφαρμογής που τροφοδοτείται με ένα ρεύμα δεδομένων στρέφεται σε αντικείμενα που κινούνται στο χώρο, τότε η συσσωρευμένη πληροφορία συνιστά ένα ρεύμα κινούμενων αντικειμένων.

Ένα ρεύμα κινούμενων αντικειμένων έχει γενικά όλα τα χαρακτηριστικά που ισχύουν για τα ρεύματα δεδομένων. Το γεγονός ότι η πληροφορία αφορά κίνηση αντικειμένων συνεπάγεται κάποιους επιπλέον περιορισμούς και χαρακτηριστικά που έχουν να κάνουν τόσο με τη φύση των αντικειμένων προς παρακολούθηση όσο και τη χωροχρονική φύση της πληροφορίας που περιγράφεται.

1.3.1 Χαρακτηριστικά της κίνησης

Η κίνηση που εκτελεί ένα σύνολο αντικειμένων εξαρτάται άμεσα από τη φύση των αντικειμένων αυτών. Κάποια αντικείμενα μπορεί να εκτελούν προκαθορισμένη κίνηση (δρομολόγια τρένων) ενώ κάποια άλλα εντελώς ακανόνιστη (ζώα στο δάσος).

Γενικά, μια εφαρμογή που παρακολουθεί αντικείμενα εν κινήσει συνήθως αγνοεί οποιαδήποτε πληροφορία για την κίνηση που πρόκειται να εκτελεστεί. Στις πλείστες εφαρμογές, το μόνο που είναι διαθέσιμο είναι η θέση των αντικειμένων σε διάφορα χρονικά διαστήματα και ενδεχομένως ένα διάνυσμα ταχύτητας. Πέρα από αυτό, η εφαρμογή αγνοεί τις προθέσεις που έχουν τα αντικείμενα ως προς την κίνηση που πρόκειται να εκτελέσουν.

Βάσει αυτού, η κίνηση των αντικειμένων προς παρακολούθηση θεωρείται *αυθαίρετη* και *άτακτη*. Τα αντικείμενα μπορούν να κινηθούν οπουδήποτε θέλουν εκτός και αν υπάρχει κάποιο φυσικό εμπόδιο. Στη γενικότερη περίπτωση τα εμπόδια αγνοούνται πλήρως από την εφαρμογή παρακολούθησης, ενώ σε πιο εξειδικευμένες εφαρμογές αυτά μπορούν να λαμβάνονται υπόψη (λ.χ η γνώση ενός οδικού δικτύου και ο περιορισμός της κίνησης των αντικειμένων σε αυτό). Η γνώση των φυσικών εμποδίων μπορεί να φανεί χρήσιμη για την διόρθωση σφαλμάτων κατά τη μέτρηση θέσεως και για την απλοποίηση υπολογισμών.

1.3.2 Ενημέρωση θέσης

Με τη σύγχρονη τεχνολογία, είναι εφικτό να καταγράφεται η θέση κινούμενων αντικειμένων με πολλούς τρόπους. Με τη χρήση συσκευών GPS, ετικετών RFIDS ή ακόμα και με την παρακολούθηση της θέσης κινητών συσκευών βάσει του σήματος που εκπέμπουν σε ένα τηλεπικοινωνιακό δίκτυο. Η μέθοδος που επιλέγεται εξαρτάται σαφώς από το είδος των αντικειμένων που παρακολουθείται και από τις απαιτήσεις κάθε εφαρμογής.

Όταν το κινούμενο αντικείμενο στέλνει μία ενημέρωση θέσης, η ενημέρωση αυτή μπορεί να είναι ένα ή και συνδυασμός των παρακάτω:

- Σημείο στο χώρο/επίπεδο

Η πιο ευρέως χρησιμοποιούμενη μέθοδος ενημέρωσης θέσης είναι να αποστέλεται ένα σημείο στο χώρο ή αλλιώς ένα ζευγάρι συντεταγμένων, που ισοδυναμούν με την θέση του αντικειμένου κατά την αποστολή της ενημέρωσης.

- Περιοχή

Όταν ο χώρος που καταλαμβάνει ένα αντικείμενο είναι μεταβλητός και απαραίτητος για την εφαρμογή, τότε η ενημέρωση πρέπει να είναι μια περιοχή (ένα πολύγωνο) με τις συντεταγμένες της στο χώρο.

Επίσης, σε εφαρμογές όπου πρέπει να λαμβάνεται υπόψη η αβεβαιότητα της θέσης, το αντικείμενο μπορεί αντί να στέλνει την ακριβή του σημειακή θέση, να στέλνει αντ' αυτού ένα σημείο σε συνδυασμό με κάποια ακτίνα που να δείχνει το ενδεχόμενο σφάλμα της θέσης του από το σημείο που απεστάλη.

- Διάνυσμα ταχύτητας

Σε αρκετές εφαρμογές η γνώση της ταχύτητας ενός αντικειμένου είναι αρκετά χρήσιμη για την εξαγωγή συμπερασμάτων αλλά και την βελτίωση της απόδοσης. Αυτή μπορεί να αποστέλλεται ως βαθμωτό μέγεθος αλλά και ως διάνυσμα.

Η χρήση του διανύσματος καθιστά δυνατή και την πρόβλεψη της μελλοντικής θέσης του αντικειμένου σε επόμενες χρονικές στιγμές. Αυτό μπορεί να γίνει υπολογίζοντας προσεγγιστικά πού πρόκειται να βρισκείται το αντικείμενο σε μελλοντικό χρόνο βάσει της τελευταίας γνωστής θέσης του και του χρονικού διαστήματος που πέρασε από τότε. Ο χρόνος αυτός σε συνδυασμό με το διάνυσμα της ταχύτητας μπορεί να δώσει την πιθανή θέση του αντικειμένου. Κάτι τέτοιο είναι χρήσιμο τόσο για τον περιορισμό των ενημερώσεων θέσης που εκτελεί ένα αντικείμενο όσο και για την παροχή δυνατότητας στο σύστημα επεξεργασίας να απαντάει ερωτήματα που αφορούν μελλοντικούς χρόνους. Φυσικά, τέτοιες απαντήσεις είναι καθαρά πιθανοτικές, αφού κανένας δεν εγγυάται πως το αντικείμενο θα διατηρήσει το ίδιο διάνυσμα ταχύτητας για όλο το μελλοντικό διάστημα που λαμβάνεται υπόψη.

1.3.3 Ρεύματα κινούμενων αντικειμένων

Σε ένα ρεύμα κινούμενων αντικειμένων, πηγές των πακέτων πληροφορίας συνιστούν τα ίδια τα κινούμενα αντικείμενα ενώ δέκτης είναι ένα κεντρικό σύστημα επεξεργασίας.

Κάθε πηγή (κινούμενο αντικείμενο) στέλνει ανά διαστήματα ενημερώσεις θέσεων στο δέκτη. Το διάστημα αυτό μπορεί να είναι αυθαίρετο για την πηγή ή να καθορίζεται από τον δέκτη. Στη δεύτερη περίπτωση συνήθως καθορίζεται ένα ελάχιστο διάστημα κατά το οποίο κάθε αντικείμενο οφείλει να ενημερώσει για την θέση του.

Επειδή οι πηγές είναι πολυάριθμες και τα μεταδιδόμενα πακέτα φτάνουν στο δέκτη σε αυθαίρετους χρόνους, κάθε αντικείμενο κατά την ενημέρωση της θέσης του αποστέλλει και ένα αναγνωριστικό της ταυτότητάς του.

Επίσης, επειδή ενδιαφέρει η κίνηση των αντικειμένων (το πρόβλημα είναι χωροχρονικό και όχι μόνο χωρικό), κάθε ενημέρωση θέσης αντικειμένου εμπεριέχει και ένα χρονόσημο (timestamp) που καθορίζει είτε την στιγμή της καταγραφής της θέσεως είτε τη στιγμή λήψης της ενημέρωσης από την πλευρά του δέκτη. Η παράμετρος αυτή κάνει εφικτή την χρήση χρονικών παραθύρων όπως επίσης και την απάντηση χωροχρονικών ερωτημάτων.

Βάσει των páραπάνω, ένα πακέτο πληροφορίας (ενημέρωση θέσης) που στέλνει ένα κινούμενο αντικείμενο είναι μία πλειάδα τιμών με την εξής γενική μορφή:

$$\langle id, t, x, y \rangle$$

όπου

id: αναγνωριστική ετικέτα για κάθε αντικείμενο,

t: η χρονική στιγμή της καταγραφής της θέσης του αντικειμένου *id*,

x: η τετμημένη του αντικειμένου *id* τη χρονική στιγμή *t*,

y: η τεταγμένη του αντικειμένου *id* τη χρονική στιγμή *t*.

Σε αυτή την πλειάδα θα μπορούσε να προστεθεί πληροφορία που να περιγράφει την ταχύτητα ή και την αβεβαιότητα της θέσης.

Όλα τα αντικείμενα ενημερώνουν συνεχώς τον δέκτη στέλνοντας πλειάδες τιμών της πιο πάνω μορφής, και η ακολουθία των πλειάδων αυτών που λαμβάνει ο δέκτης συνιστά ένα ρεύμα κινούμενων αντικειμένων.

1.3.4 Συστήματα διαχείρισης κινούμενων αντικειμένων

Κάθε εφαρμογή που επεξεργάζεται δεδομένα, χρησιμοποιεί κάποια βάση δεδομένων. Αυτή μπορεί να είναι απλά ένα αρχείο ή ένα πολύπλοκο σύστημα διαχείρισης βάσεων δεδομένων. Σε κλασικές βάσεις, τα δεδομένα αντιμετωπίζονται ως οντότητες στατικές, δηλαδή από τη στιγμή της δημιουργίας τους υφίστανται εως ότου τροποποιηθούν. Για παράδειγμα, αν σε κάποια βάση αποθηκευτούν οι μισθοί κάποιων υπαλλήλων, από τούδε και στο εξής ο μισθός κάποιου υπαλλήλου θα είναι αυτός που αποθηκεύθηκε εξαρχής εκτός και αν η τιμή αυτή ενημερωθεί σε κάποια φάση στο μέλλον.

Σε εφαρμογές επεξεργασίας ρευμάτων δεδομένων, όπως είναι οι εφαρμογές διαχείρισης κινούμενων αντικειμένων, ο χρόνος ζωής των δεδομένων εξαρτάται άμεσα από το είδος της εφαρμογής. Παραδείγματος χάρη, η μελέτη της θέσης αντικειμένων δεν νοείται αν τα δεδομένα εισόδου που υπάρχουν είναι στίγματα που προήλθαν την περασμένη ώρα, εκτός και αν μας ενδιαφέρει ρητά η θέση τους εκείνη τη χρονική περίοδο. Οι εφαρμογές πραγματικού χρόνου χρειάζονται πολύ πρόσφατη πληροφορία προκειμένου τα συμπεράσματα που εξάγουν να είναι επίκαιρα γιαυτό και απαιτούν συνεχείς ενημερώσεις από την πλευρά των αντικειμένων. Χρησιμοποιώντας μία κλασική βάση δεδομένων θα ήταν αναγκαίο να γίνονται συνεχώς ενημερώσεις οι οποίες εκ των πραγμάτων είναι πολύ χρονοβόρες. Όλα τα κλασικά συστήματα βάσεων αποθηκεύουν τα δεδομένα σε δευτερεύουσα μνήμη αφού οι ενημερώσεις τους είναι πολύ πιο σπάνιες από τις αναγνώσεις τους. Στα συστήματα πραγματικού χρόνου συμβαίνει ακριβώς το αντίθετο με αποτέλεσμα οι κλασικές βάσεις δεδομένων να κρίνονται ανεπαρκείς να ικανοποιήσουν τις ανάγκες των εφαρμογών.

Ακόμα, οι απαιτήσεις επεξεργασίας πραγματικού χρόνου σε συνδιασμό με τον μεγάλο όγκο των δεδομένων, εκφράζουν την ανάγκη σωστής διαχείρισης των δεδομένων από την πλευρά του συστήματος, ώστε η προσπέλασή τους να γίνεται όσο το δυνατόν γρηγορότερα [10].

Για την ικανοποίηση τέτοιων απαιτήσεων, τα συστήματα διαχείρισης ρευμάτων δεδομένων αναγκάζονται να αποθηκεύουν τα δεδομένα στην κύρια μνήμη προκειμένου να γίνεται πολύ πιο γρήγορη η επεξεργασία τους. Οι απαιτήσεις σε μνήμη γίνονται έτσι πιο σημαντικές, αφού το μέγεθος της κύριας μνήμης είναι σαφώς πολύ πιο μικρό από την αντίστοιχη δευτερεύουσα μνήμη. Ακόμα, ειδικές δομές δεδομένων και ευρετηρίων για κάθε εφαρμογή γίνονται απαραίτητες προκειμένου να επιτευχθεί γρηγορότερα η προσπέλαση των δεδομένων [23].

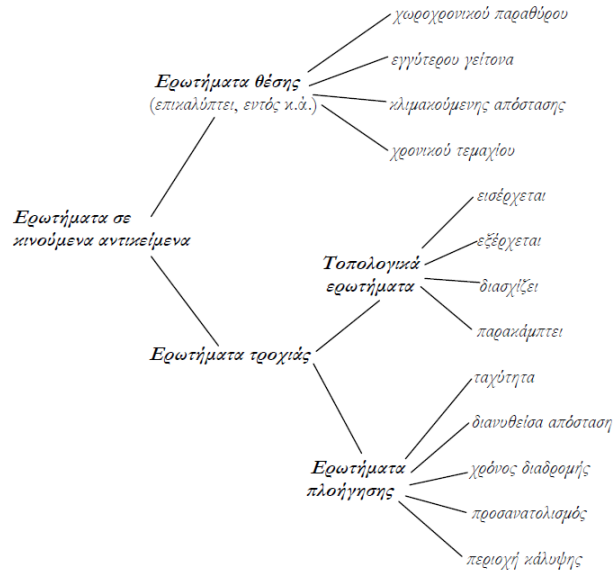
Για να είναι σε θέση ένα σύστημα επεξεργασίας κινούμενων αντικειμένων να δίνει απαντήσεις σε πραγματικό χρόνο, χρησιμοποιεί διαφορετικές προσεγγίσεις στην επεξεργασία των δεδομένων από ένα κλασικό σύστημα βάσης δεδομένων:

1. Επεξεργάζεται κάθε στίγμα την στιγμή που φτάνει, προκειμένου να αντιμετωπίσει την δυναμικότητα των δεδομένων
2. Δεδομένου πως το σύστημα ενδιαφέρεται να δίνει επικαιροποιημένες απαντήσεις, απορρίπτει πληροφορία που θεωρείται παρωχημένη με στόχο να μειώσει τον όγκο της προς επεξεργασία πληροφορίας.

Το πρώτο επιτυγχάνεται με την υλοποίηση ερωτημάτων διαρκείας ενώ το δεύτερο κυρίως με τη χρήση χρονικών παραθύρων. Επίσης κατανεμημένες μέθοδοι έχουν προταθεί στη βιβλιογραφία προκειμένου ένα σύστημα να αντεπεξέλθει στις απαιτήσεις λειτουργίας πραγματικού χρόνου [5, 8, 11].

Η διαχείριση χωροχρονικών δεδομένων δημιουργεί την ανάγκη για χρήση πιο εξειδικευμένων ερωτημάτων διαρκείας από αυτά που παρουσιάστηκαν σε

προηγούμενες ενότητες [13, 21, 24]. Μια λίστα με τα αντιπροσωπευτικότερα από αυτά παρουσιάζεται στο σχήμα 1.9.



Σχήμα 1.9: Κατηγορίες ερωτημάτων σε κινούμενα αντικείμενα. (Πηγή: [16])

Κεφάλαιο 2

Τροχιές κινούμενων αντικειμένων

Καθώς ένα αντικείμενο κινείται, χαρακτηρίζεται από μεγέθη όπως η ταχύτητά του, η πορεία που ακολουθεί και η θέση του στο χώρο σε σχέση με άλλα αντικείμενα.

Μεγάλης σημασίας πληροφορία μεταξύ αυτών, αποτελεί η ιστορία της κίνησής του. Από αυτήν μπορούν να εξαχθούν σημαντικές πληροφορίες ανάλογα με την φύση του αντικειμένου. Για παράδειγμα, γνωρίζοντας την ιστορία της κίνησης αυτοκινήτων σε μία πόλη, μπορούν να εξαχθούν στοιχεία που θα βελτιώσουν την κυκλοφοριακή κίνηση και θα εντοπίσουν δρόμους με αυξημένη συμφόρηση. Παρομοίως, έχοντας καταγράψει την κίνηση ζώων σε ένα βιότοπο, μπορούν να εξαχθούν πληροφορίες για τις βιολογικές τους συνήθειες (λ.χ. διαδρομές που ακολουθούν κατά το κυνήγι) ή να εντοπιστούν περιπτώσεις ζώων που ξεφεύγουν από το βιότοπο. Η τροχιά ενός αντικειμένου συνιστά ακριβώς αυτό, την ιστορία της κίνησής του. Ζητούμενο είναι η καταγραφή της θέσεώς του στο χώρο με την πάροδο του χρόνου.

Πολλά αντικείμενα στον κόσμο έχουν προδιαγεγραμμένη κίνηση η οποία είναι γνωστή ή μπορεί να προβλεφθεί εκ των προτέρων. Για παράδειγμα, οι πλανήτες και οι δορυφόροι, εκτελούν περιοδική κίνηση γύρω από άλλα ουράνια σώματα. Η εξακρίβωση της θέσης τους γίνεται εύκολα με την εφαρμογή φυσικών νόμων και μαθηματικών εξισώσεων. Το ίδιο μπορεί να λεχθεί και για τα δρομολόγια των τρένων, αν θεωρήσουμε πως η λειτουργία τους δεν πρόκειται να διακοπεί για κάποιο λόγο. Απεναντίας, υπάρχουν αντικείμενα τα οποία κινούνται χωρίς να υπάρχει κάποιος τρόπος να προϋπολογιστεί η κίνηση η οποία θα εκτελέσουν. Παραδείγματα τέτοια συνιστούν τα αυτοκίνητα ή οι πεζοί μιας πόλης, τα ζώα και προπάντων τα πουλιά. Η κίνηση των αντικειμένων αυτών παρουσιάζει επιπλέον ενδιαφέρον λόγω της δυσκολίας που δημιουργεί στην μελέτη της σε πραγματικό χρόνο.

Το πώς αναπαρίσταται αυτή η κίνηση, πώς μπορεί να αποθηκευτεί και

τί ερωτήματα μπορούν να απαντηθούν γνωρίζοντάς την, είναι ζητήματα που πραγματεύεται το κεφάλαιο αυτό [17].

2.1 Αναπαράσταση τροχιάς

Η τροχιά ενός αντικειμένου υφίσταται από την έναρξη της κίνησής του και συνεχίζεται επ' αόριστον. Επομένως ορίζεται μέχρι την τελευταία πραγματική θέση του αντικειμένου και όχι για μελλοντικές.

Έστω ένα αντικείμενο o που κινείται στο Ευκλείδειο επίπεδο. Η τροχιά του αντικειμένου $traj(o)$ ορίζεται ως η ακολουθία όλων των σημείων της θέσης του p_i στο χώρο για κάθε χρονική στιγμή t . Ο χώρος για τους σκοπούς της εργασίας αυτής περιορίζεται στο δισδιάστατο επίπεδο xy . Δηλαδή:

$$p_i \equiv (x_i, y_i) \mid x, y \in \mathcal{R}, i \in \mathcal{N}$$

$$traj(o) \equiv \{p_i(t) \mid \forall t \in (-\infty, Now]\}$$

Προκειμένου να χρησιμοποιηθεί η τροχιά ενός κινούμενου αντικειμένου, πρέπει να βρεθεί μία μέθοδος αναπαράστασής της. Πρέπει να σημειωθεί ότι κάθε αντικείμενο καταλαμβάνει μία πεπερασμένη περιοχή στο χώρο που εξαρτάται από το μέγεθος και το σχήμα του. Εντούτοις, αφού το ενδιαφέρον στρέφεται στην κίνηση του αντικειμένου, τα δύο αυτά μεγέθη θεωρούνται ασήμαντα και εκφυλίζονται σε ένα αδιάστατο μέγεθος, το σημείο.

Εγγενώς η τροχιά είναι φαινόμενο συνεχές, αφού ορίζεται για κάθε χρονική στιγμή και ο χρόνος είναι άπειρο σύνολο. Γι' αυτό, για την αναπαράστασή της απαιτείται ένα συνεχές μοντέλο. Εν τούτοις, επειδή το ενδιαφέρον της εργασίας στρέφεται σε αντικείμενα που η κίνησή τους είναι αυθαίρετη, προκύπτουν πρακτικά προβλήματα τα οποία οδηγούν στη χρήση ενός διακριτού μοντέλου. Παρακάτω αναπτύσσονται τα δύο αυτά μοντέλα αναπαράστασης της τροχιάς.

2.1.1 Συνεχές μοντέλο

Η τροχιά ορίστηκε ως η ακολουθία των σημείων της θέσης κάποιου αντικειμένου, για κάθε χρονική στιγμή. Η αναπαράσταση αυτής της ακολουθίας επιτυγχάνεται με τη χρήση μιας συνάρτησης από το πεδίο του χρόνου στο χωρικό επίπεδο.

Η συνάρτηση αυτή είναι μονότονη αφού σε κάθε χρονική στιγμή είναι αδύνατο να αντιστοιχούν περισσότερα από ένα σημεία θέσης του αντικειμένου αφού το αντικείμενο δεν μπορεί να βρίσκεται σε δύο μέρη ταυτόχρονα. Με αυτή την προϋπόθεση, ορίζεται η συνάρτηση $\mathcal{F}(t)$ όπου

$$\mathcal{F} : T \rightarrow \mathcal{R} \times \mathcal{R}$$

Η συνάρτηση $\mathcal{F}(t)$ περιγράφει την τροχιά ενός αντικειμένου και αντιστοιχεί χρονόσημα (δηλαδή χρονικές στιγμές) σε σημεία στο χώρο. Η συνάρτηση $\mathcal{F}(t)$

είναι ένα προς ένα και συνεχής, αφού η θέση οποιουδήποτε φυσικού αντικειμένου θεωρείται πάντοτε προσδιορισμένη.

2.1.2 Διακριτό μοντέλο

Ακόμα και αν είναι εξαρχής γνωστή η τροχιά που πρόκειται να ακολουθήσει ένα αντικείμενο, η αναπαράσταση μιας πολύπλοκης και εν γένει άτακτης καμπύλης με τη χρήση συνάρτησης αποτελεί από μόνη της ένα δύσκολο έργο. Λαμβάνοντας υπόψη και την απρόβλεπτη κίνηση των αντικειμένων, είναι προφανές ότι η περιγραφή της τροχιάς τους ως ένα συνεχές φαινόμενο καθώς αυτή εξελίσσεται είναι αδύνατη.

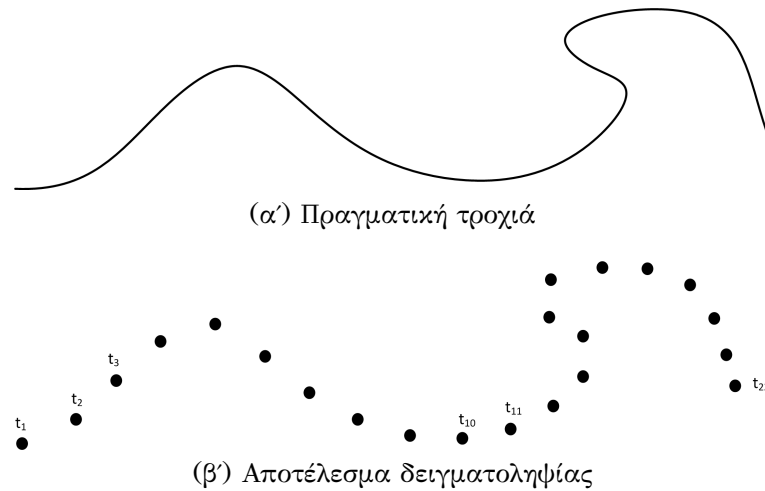
Ένα ακόμη πρόβλημα που έρχεται να προστεθεί είναι η χρήση ψηφιακών συστημάτων για την αποθήκευση και επεξεργασία των πληροφοριών αυτών. Εκ κατασκευής, τα ψηφιακά συστήματα μπορούν να διαχειριστούν μόνο διακριτή πληροφορία πεπερασμένου μεγέθους και η αναπαράσταση άπειρων συνόλων καθίσταται αδύνατη.

Λόγω τέτοιων προβλημάτων, κρίνεται αναγκαία η εύρεση ενός μοντέλου το οποίο θα περιγράφει την τροχιά ενός αντικειμένου με διακριτό τρόπο, ώστε να γίνεται εφικτή η αναπαράστασή της.

Αρχικά, κρίνεται σκόπιμο να οριστεί η έννοια του τρισδιάστατου μοντέλου του χωροχρόνου, όπου οι δύο διαστάσεις περιγράφουν ένα σημείο στο δισδιάστατο επίπεδο και η τρίτη την χρονική στιγμή κατά την οποία το αντικείμενο βρισκόταν στο σημείο αυτό. Η αναπαράσταση ενός σημείου γίνεται με μία πλειάδα $\langle x, y, t \rangle$ η οποία καθορίζει το σημείο και την στιγμή που αυτό καταγράφηκε. Αυτή η πλειάδα καθορίζει το τρέχον στίγμα της θέσης του αντικειμένου.

Το διακριτό μοντέλο της τροχιάς προκύπτει δειγματοληπτώντας την ανά τακτά χρονικά διαστήματα. Έτσι, αντί να υπάρχει ένα σημείο της τροχιάς για κάθε χρονική στιγμή, καταγράφεται μόνο ένα πεπερασμένο σύνολο στιγμάτων για συγκεκριμένα χρονόσημα. Το πλήθος τους καθορίζει και την ακρίβεια της αναπαράστασης. Στο σχήμα 2.1 παρουσιάζεται μία τροχιά και ένα πιθανό σύνολο δειγμάτων που μπορούν να αποθηκευτούν γι' αυτή. Τα δείγματα αποτελούν μία ακολουθία στιγμάτων και η τροχιά του αντικειμένου προσεγγίζεται από το σύνολο αυτό.

Απομένει να βρεθεί τρόπος να ανακτηθεί η θέση του αντικειμένου για στιγμές που δεν υπάρχουν καταγραφές της θέσης του. Λόγω της δειγματοληψίας, αυτή μπορεί να υπολογιστεί μόνο προσεγγιστικά. Για τη προσέγγιση της τροχιάς έχουν προταθεί τρεις ευρέως χρησιμοποιούμενες μέθοδοι οι οποίες αναφέρονται πιο κάτω.



Σχήμα 2.1: Τροχιά αντικειμένου και το αποτέλεσμα της δειγματοληψίας.

α) Τμηματική σύνθεση

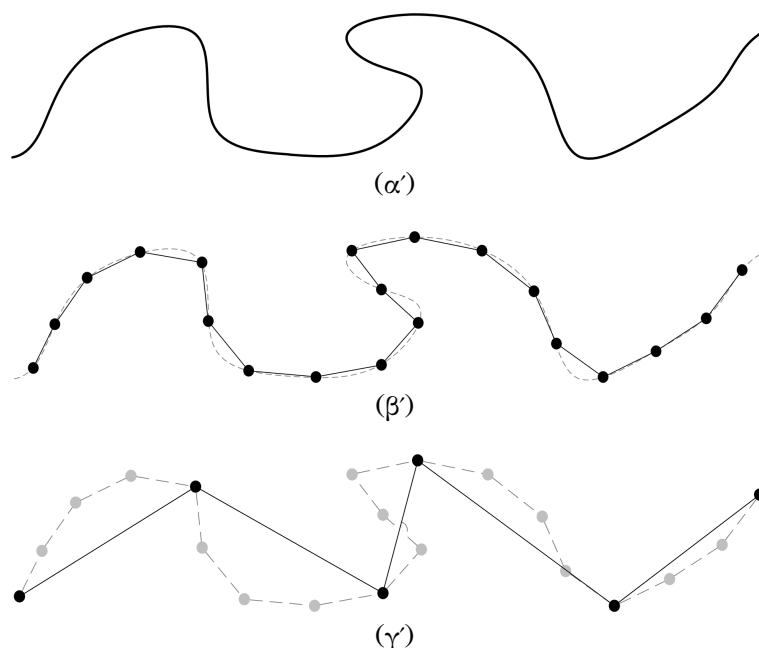
Η πιο απλή μέθοδος για να ανακατασκευασθεί μία τροχιά από τα δείγματα της είναι με τμηματική σύνθεσή τους. Κάθε σημείο της τροχιάς ενώνεται με το επόμενο του με ένα ευθύγραμμο τμήμα. Το αποτέλεσμα που προκύπτει είναι μία τεθλασμένη γραμμή η οποία προσεγγίζει την πραγματική τροχιά.

Στο σχήμα 2.2 παρουσιάζεται η πραγματική τροχιά και το αποτέλεσμα της τμηματικής σύνθεσης από τα στίγματα που προέκυψαν από τη δειγματοληψία. Η ποιότητα της προσέγγισης εξαρτάται ευθέως από τον αριθμό αλλά και την συγκεκριμένη επιλογή των δειγμάτων.

Πρόκειται για την απλούστερη μέθοδο προσέγγισης και γι' αυτό είναι επιρρεπής σε μεγάλα σφάλματα. Όπως φαίνεται και στο σχήμα 2.2, αν η κίνηση του αντικειμένου είναι πολύ άτακτη σε σύντομο χρονικό διάστημα ενώ η δειγματοληψία γίνεται σε αραιά διαστήματα, σημαντική πληροφορία για τη κίνηση του αντικειμένου μπορεί να χαθεί.

β) Πολυωνυμική παρεμβολή

Η πολυωνυμική παρεμβολή αποτελεί συνήθως την καλύτερη επιλογή αν το ζητούμενο είναι να ανακατασκευασθεί η τροχιά όσο το δυνατόν καλύτερα από τα δείγματά της και να επιτευχθεί αυτό χωρίς πολύ μεγάλη υπολογιστική ισχύ. Η μέθοδος προσπαθεί να βρει ένα πολυώνυμο που η καμπύλη του να περιλαμβάνει όσα περισσότερα δείγματα γίνεται και να μην αποκλίνει αρκετά από τα υπόλοιπα. Η εύρεση ενός τέτοιου πολυωνύμου στηρίζεται σε μαθηματικές μεθόδους και συνήθως προσεγγίζει αρκετά καλά την αρχική καμπύλη από την οποία πάρθηκαν τα δείγματα (τροχιά). Φυσικά, το τελικό αποτέλεσμα εξαρτάται άμεσα από την συχνότητα δειγματοληψίας κατά την διάρκεια της



Σχήμα 2.2: α) Πραγματική τροχιά, β) Τμηματική σύνθεση, γ) Τμηματική σύνθεση με αραιά δείγματα

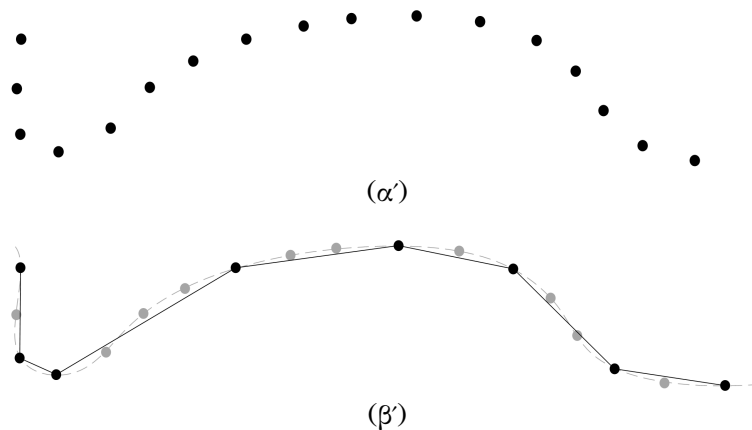
κίνησης.

Αν και δίνει πολύ ικανοποιητικά αποτελέσματα, η παρεμβολή δεν προτιμάται όπως θα περίμενε κανείς. Η πολυπλοκότητα της είναι αρκετά αυξημένη ώστε τα συστήματα χειρισμού κινούμενων αντικειμένων να προτιμούν να θυσιάζουν την ακρίβεια της προσέγγισης υπέρ του υπολογιστικού πλεονεκτήματος που πετυχαίνουν με τη χρήση της γραμμικής σύνθεσης.

γ) Πολυωνυμικές συναρτήσεις

Οι πολυωνυμικές συναρτήσεις (splines) μεταφέρουν την ιδέα της πολυωνυμικής παρεμβολής ένα βήμα παραπέρα προσεγγίζοντας την τροχιά με περισσότερα από ένα πολυώνυμα. Αυτό γίνεται χωρίζοντας την καμπύλη σε κομμάτια, όπου για κάθε κομμάτι η περιγραφή του επιτυγχάνεται με τη χρήση διαφορετικού πολυωνύμου. Με αυτό τον τρόπο επιτυγχάνεται καλύτερη ανακατασκευή.

Στη περίπτωση που εξετάζεται, η αναπαράσταση της τροχιάς με τη λεπτομέρεια που επιτυγχάνεται από την χρήση πολυωνυμικών συναρτήσεων θεωρείται άσκοπη, αφού τα δείγματα συνήθως είναι αρκετά για να μπορεί να προσεγγιστεί η τροχιά ικανοποιητικά χωρίς μεγάλο σφάλμα από τις δύο προηγούμενες προσεγγίσεις.



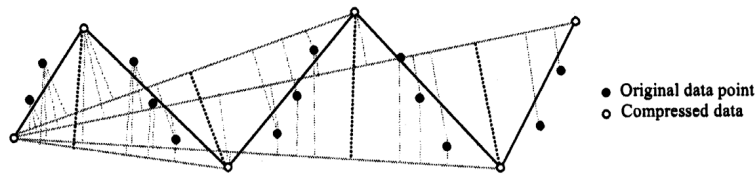
Σχήμα 2.3: α) Τροχιά που προκύπτει από την αρχική δειγματοληψία, β) Τροχιά μετά την απόρριψη περιττής πληροφορίας

2.2 Συμπύεση πληροφορίας κίνησης

Λόγω της αδυναμίας που παρουσιάζεται στην προσπάθεια καταγραφής της τροχιάς με συνεχή τρόπο, εφαρμόζεται αναγκαστικά η μέθοδος της δειγματοληψίας. Με αυτό το τρόπο, περιγράφεται ένα άπειρο σύνολο από στίγματα με ένα αντίστοιχο πεπερασμένο σύνολο που είναι εν γένει διαχειρίσιμο. Ακόμα και με την μέθοδο αυτή όμως, ο όγκος της πληροφορίας που προκύπτει για ένα και μόνο αντικείμενο είναι μεγάλος. Λαμβάνοντας υπόψη πως τα σημερινά συστήματα διαχείρισης κινούμενων αντικειμένων, πρέπει να καταγράφουν δεκάδες μέχρι εκατοντάδες χιλιάδες αντικειμένων, ο όγκος της συλλεγόμενης πληροφορίας γίνεται τεράστιος σε σημείο που να μην μπορεί να είναι εύκολα επεξεργάσιμος.

Μελετώντας το αποτέλεσμα που προκύπτει από τη δειγματοληψία μιας τροχιάς, μπορεί κάποιος να συμπεράνει πως αρκετά στίγματα θα μπορούσαν πολύ εύκολα να απορριφθούν χωρίς μεγάλη απώλεια πληροφορίας. Ένα τέτοιο παράδειγμα εμφανίζεται στην περίπτωση κίνησης ενός αντικειμένου σε μια μεγάλη ευθεία, λ.χ. ένα αυτοκίνητο στην εθνική οδό. Κατά την διάρκεια της ευθύγραμμης κίνησης του αντικειμένου, ένας μεγάλος αριθμός στιγμάτων θα μπορούσαν να απορριφθούν χωρίς να χαθεί οποιαδήποτε ουσιαστική πληροφορία για την κίνησή του. Ένα τέτοιο παράδειγμα παρουσιάζεται στο σχήμα 2.3.

Γενικεύοντας την ιδέα αυτή, γίνεται εμφανές ότι απορρίπτοντας όλο και μεγαλύτερο αριθμό στιγμάτων, μειώνεται μεν η ακρίβεια της αναπαράστασης, αλλά την ίδια στιγμή ελαττώνεται και ο όγκος των δεδομένων που πρέπει να αποθηκευτούν. Φυσικά, η συμπίεση της πληροφορίας εξαρτάται από το είδος της κίνησης αφού ένα αντικείμενο που εκτελεί πολύ άτακτη κίνηση θα είναι δύσκολο να περιγραφεί σωστά η κίνησή του αν απορριφθεί πολλή πληροφορία. Επιπλέον, ο βαθμός συμπίεσης μπορεί να ελεγχθεί σε συνάρτηση με το μέγεθος



Σχήμα 2.4: Αλγόριθμος Douglas Peucker. Η τροχιά είχε αρχικό μέγεθος δείγματος 19 σημεία. Επιλεχθέντα σημεία προέκυψαν τα 16, 12, 8, 4 (Πηγή: [12])

της ανοχής που μπορεί να δείξει μια συγκεκριμένη εφαρμογή στο σφάλμα, χωρίς να επηρεάζονται σε μεγάλο βαθμό τα αποτελέσματά της.

Προκειμένου να επιτευχθεί η συμπίεση της πληροφορίας, έχουν προταθεί αλγόριθμοι που είτε δέχονται ως είσοδο το σύνολο των στιγμάτων (offline προσέγγιση) και με επαναληπτικές μεθόδους απορρίπτουν στίγματα μέχρι να φτάσουν ένα ανώτατο κατώφλι σφάλματος, είτε εκτελούν την απόρριψη αυτή κατά την διάρκεια της λήψης των στιγμάτων (online προσέγγιση) καθώς η τροχιά επικαιροποιείται. Ενδεικτικά, παρουσιάζονται αλγόριθμοι συμπίεσης που εφαρμόζουν τις δύο μεθόδους προκειμένου να γίνει πιο κατανοητή η διαδικασία της συμπίεσης.

2.2.1 Offline συμπίεση

Αντιπροσωπευτικότερος αλγόριθμος που ακολουθεί offline προσέγγιση για την περιγραφή μιας τροχιάς είναι ο αλγόριθμος Douglas-Peucker [12]. Ο αλγόριθμος αυτός σχεδιάστηκε με σκοπό να απλοποιεί καμπύλες γραμμές που περιγράφονται από ακολουθία σημείων έχοντας ως στόχο μία ικανοποιητική περιγραφή της καμπύλης χρησιμοποιώντας ελάχιστο σύνολο σημείων.

Ο αλγόριθμος δουλεύει με τον εξής τρόπο. Όπως παρουσιάζεται στο σχήμα 2.4, ξεκινάει ενώνοντας με μία ευθεία το αρχικό και τελευταίο σημείο που περιγράφουν την καμπύλη. Για κάθε καταγεγραμμένο σημείο της καμπύλης, υπολογίζεται η Ευκλείδεια απόστασή του από το ίχνος του στην ευθεία αυτή. Στη συνέχεια επιλέγεται το σημείο με την μεγαλύτερη απόσταση. Αν η απόσταση αυτή είναι μικρότερη από ένα προκαθορισμένο κατώφλι, τότε η ευθεία που επιλέχθηκε θεωρείται ικανοποιητική για να περιγράψει την καμπύλη και ο αλγόριθμος σταματάει. Αν όχι, τότε η καμπύλη διαιρείται στο επιλεχθέν σημείο. Αυτό τώρα αντιπροσωπεύει το τελευταίο σημείο της πρώτης καμπύλης και το πρώτο της δεύτερης. Ο αλγόριθμος επιλογής σημείου εκτελείται επαναληπτικά για τα δύο κομμάτια της καμπύλης έως ότου προκύψει το προκαθορισμένο κατώφλι σφάλματος. Στη συνέχεια, τα επιλεχθέντα σημεία ενώνονται μεταξύ τους με ευθύγραμμα τμήματα για να δημιουργήσουν την απλοποιημένη περιγραφή της αρχικής τροχιάς.

Ο αλγόριθμος αυτός, όπως και κάθε άλλος του είδους του, απαιτεί την ύπαρξη όλων των δειγμάτων της τροχιάς προκειμένου να εκτελέσει την απλο-

ποίηση. Ωστόσο παραμένει το πρόβλημα της αποθήκευσης τεράστιου όγκου δεδομένων μέχρις ότου ενεργοποιηθεί η διαδικασία απλοποίησης τροχιών στο σύστημα.

2.2.2 Online συμπίεση

Η offline προσέγγιση για συμπίεση των τροχιών είναι αποτελεσματική στις περιπτώσεις που χρειάζονται τα δεδομένα για μελλοντική χρήση και απαιτείται να μειωθεί ο όγκος τους ώστε να είναι εφικτό να αποθηκευτούν. Όταν η επεξεργασία των τροχιών χρειάζεται να γίνεται σε πραγματικό χρόνο, η offline προσέγγιση αδυνατεί να βοηθήσει αφού οι τροχιές βρίσκονται ακόμα υπό εξέλιξη, ενώ η ύπαρξη όλης της τροχιάς είναι απαραίτητη για την λειτουργία των αλγορίθμων.

Για την αντιμετώπιση του προβλήματος, προτάθηκαν μέθοδοι που εκτελούν συμπίεση της αναπαράστασης της τροχιάς καθώς αυτή εξελίσσεται. Το αποτέλεσμα επιτυγχάνεται με την χρήση της έννοιας του *κυλιόμενου παραθύρου* (*sliding window*). Οι αλγόριθμοι θεωρούν ένα χρονικό διάστημα δεδομένου μεγέθους (μπορεί να είναι μεταβλητό κάτω υπό συνθήκες) και προσθέτουν χρονοστίγματα στο παράθυρο μέχρι αυτό να γεμίσει. Όταν αυτό συμβεί επιλέγουν το σύνολο των στιγμάτων που προσεγγίζουν καλύτερα το κομμάτι της τροχιάς που περιγράφεται από τα σημεία του παραθύρου και απορρίπτουν τα υπόλοιπα. Το παράθυρο ‘κυλίεται’ προκειμένου να ξαναγεμίσει με τα επόμενα σημεία της τροχιάς που θα καταφθάσουν στο σύστημα. Αν το παράθυρο είναι σταθερού μεγέθους, τότε ο αλγόριθμος ονομάζεται *κυλιόμενου παραθύρου*, ενώ αν το μέγεθος του παραθύρου εξαρτάται από κάποια συνθήκη τότε ονομάζεται *ανοικτού παραθύρου*.

Για να γίνει περισσότερο κατανοητή η διαδικασία, παρουσιάζεται ένα παράδειγμα αλγορίθμου “ανοικτού παραθύρου”. Θεωρούμε αρχικά πως η δειγματοληψία της τροχιάς μόλις έχει ξεκινήσει. Ο αλγόριθμος θεωρεί το πρώτο σημείο ως αρχικό και με κάθε νέα άφιξη σημείου υπολογίζει ένα εθύγραμμο τμήμα που να ενώνει το αρχικό με το τελευταίο ληφθέν σημείο. Στην συνέχεια υπολογίζει τις κάθετες αποστάσεις των ενδιάμεσων σημείων από το ευθύγραμμο τμήμα. Αν όλες είναι μικρότερες από ένα κατώφλι, τότε ο αλγόριθμος προσπαθεί να συμπεριλάβει και το επόμενο σημείο που πρόκειται να φτάσει στα σημεία εντός παραθύρου, μεγαλώνοντας έτσι και το παράθυρο. Αυτό γίνεται με κάθε άφιξη στίγματος έως ότου ξεπεραστεί το κατώφλι. Τότε μπορούν να εφαρμοστούν δύο στρατηγικές:

- είτε επιλέγεται το σημείο που προκάλεσε το σφάλμα, ως το τελικό σημείο αυτού του τμήματος και ως αρχικό του επόμενου που πρόκειται να υπολογιστεί
- είτε επιλέγεται για την ίδιο σκοπό το προηγούμενο σημείο από το τελευταίο που εξέτασε ο αλγόριθμος (και η εξέτασή του προκάλεσε σφάλμα).

Το παράθυρο τότε μικραίνει (ουσιαστικά μετακινείται) ώστε να συμπεριλαμβάνει το σημείο που επιλέχθηκε και όλα τα επόμενά του και ο αλγόριθμος συνεχίζει την λειτουργία του.

2.2.3 Προβλήματα

Με τις μεθόδους που παρουσιάστηκαν η πληροφορία συμπιέζεται δημιουργώντας μία γραμμική προσέγγιση της τροχιάς χρησιμοποιώντας ένα υποσύνολο των σημείων που την περιγράφουν. Η προσέγγιση αυτή όμως παρουσιάζει ένα εγγενές πρόβλημα που προκύπτει από την αντιμετώπιση της τροχιάς ως χωρικό και όχι ως χωροχρονικό φαινόμενο.

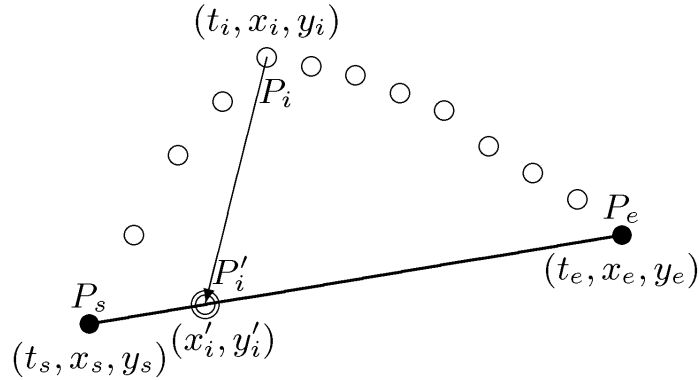
Σκοπός της συμπίεσης είναι να μειωθεί ο όγκος της πληροφορίας και ταυτόχρονα να είναι εφικτό να υπολογιστεί προσεγγιστικά η θέση του εκάστοτε αντικειμένου για χρονικές στιγμές στις οποίες δεν έχει ληφθεί κάποιο στίγμα. Τα αντιπροσωπευτικά σημεία για τη συμπίεση επιλέγονται ώστε η απόσταση των ενδιάμεσων από το γραμμικό τμήμα να είναι μικρότερη από ένα κατώφλι. Η απόσταση αυτή συμβολίζει τη διαφορά του πραγματικού σημείου από το αντίστοιχο προσεγγιστικό του. Η δήλωση αυτή θα ήταν σωστή, αν η τροχιά δεν ήταν χωροχρονικό φαινόμενο αλλά μονάχα χωρικό. Όπως φαίνεται και στο σχήμα 2.5, λαμβάνοντας υπόψη ότι το αντικείμενο κινείται με μεταβλητή ταχύτητα, η απόσταση ενός σημείου με αυτό που του αντιστοιχεί στη προσεγγιστική τροχιά δεν είναι κατ' ανάγκη η κάθετη. Είναι αυτή που προκύπτει αν ληφθεί υπόψη ο χρόνος που έφτασε το στίγμα σε σχέση με το χρονικό διάστημα που διαφέρουν τα δύο σημεία στα άκρα. Η απόσταση αυτή είναι το πραγματικό σφάλμα που υπεισέρχεται από την προσέγγιση και αυτό είναι πιθανό να ξεπερνάει το κατώφλι που έθεσε ο αλγόριθμος ως προϋπόθεση. Έτσι πολύ πιθανόν ο αλγόριθμος συμπίεσης ενδεχομένως καταλήγει να δίνει λάθος συμπεράσματα.

Η αντιμετώπιση του προβλήματος αυτού απαιτεί την χρήση πιο σύνθετων αλγόριθμων ώστε η προσεγγιστική τροχιά να είναι συγχρονισμένη με την πραγματική προκειμένου να υπολογίζεται σωστότερα το σφάλμα της προσέγγισης. Το θέμα εξετάζεται ενδελεχώς στην εργασία των Meratnia and By [12].

2.3 Ερωτήματα σε τροχιές

Η παρακολούθηση της κίνησης των αντικειμένων και η καταγραφή της τροχιάς τους έχει ως απώτερο στόχο την εξαγωγή παρατηρήσεων και συμπερασμάτων που αφορούν την αλληλεπίδραση των κινούμενων αντικειμένων μεταξύ τους και με το χώρο γύρω τους.

Η καταγραφή της τροχιάς αυτής καθεαυτής, δίνει τη δυνατότητα στον χρήστη να υποβάλλει 'ερωτήματα' προς το σύστημα για τα οποία δεν θα μπορούσε να πάρει απαντήσεις αν η καταγραφή περιοριζόταν μονάχα σε μεμονωμένες



Σχήμα 2.5: Συγχρονισμένη Ευκλείδεια απόσταση. (Πηγή: [12])

θέσεις του αντικειμένου. Τέτοια ερωτήματα έχουν να κάνουν με την κίνηση των αντικειμένων σε σχέση με μια περιοχή ή μεγέθη που χαρακτηρίζουν την ίδια την κίνησή τους όπως η ταχύτητα και ο προσανατολισμός τους.

Τα ερωτήματα για τροχιές θα μπορούσαν να διακριθούν σε δύο κατηγορίες [16]:

- στα τοπολογικά ερωτήματα που ασχολούνται με την ίδια την τροχιά,
- στα ερωτήματα πλοήγησης που ασχολούνται με εξαγόμενα συμπεράσματα που μπορούν να προκύψουν από την τροχιά.

Ακολουθεί μια περιγραφή των ερωτημάτων αυτών με σχετικά παραδείγματα.

2.3.1 Τοπολογικά ερωτήματα

Τα τοπολογικά ερωτήματα ασχολούνται με την ‘τοπολογία’ της τροχιάς, δηλαδή με την μορφή της και τα συμπεράσματα που μπορούν να προκύψουν απ’ αυτήν.

Η τοπολογία της τροχιάς είναι σε θέση να απαντήσει πολλά ερωτήματα τα οποία ανάγονται όλα σε τρία βασικά. Την

- είσοδο/έξοδο (*enter/exit*)
- διάβαση (*cross*)
- προσπέρασμα (*bypass*)

μιας περιοχής κάποια στιγμή/περίοδο από ένα μεμονωμένο αντικείμενο.

Έστω ότι το σύστημα ασχολείται με αυτοκίνητα που κινούνται στο δίκτυο μιας πόλης. Ένα ερώτημα εισόδου θα μπορούσε να ήταν “Ποια αυτοκίνητα

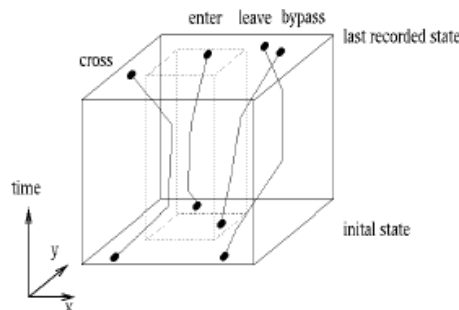
εισήλθαν κάποια στιγμή σήμερα στον κεντρικό δακτύλιο;”. Αντίστοιχα, ένα ερώτημα διάβασης θα μπορούσε να ήταν “Πέρασε κάποιο αυτοκίνητο τα όρια της περιοχής Ζωγράφου κινούμενο προς την περιοχή Αμπελοκήπων την περίοδο 8:00 με 10:00 το πρωί;”.

Στηριζόμενοι στα αποτελέσματα που μπορούν να προκύψουν από τα βασικά αυτά ερωτήματα, είναι δυνατόν να εκτελεστούν πιο σύνθετα ερωτήματα για να δώσουν απαντήσεις όπως:

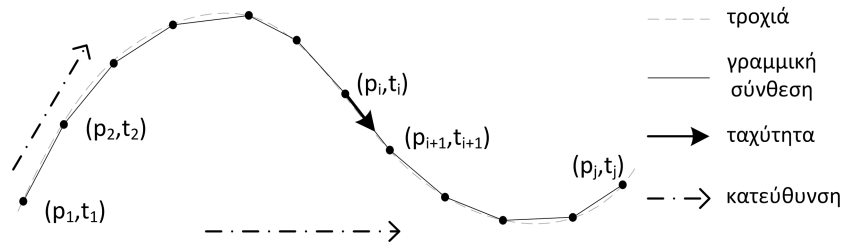
- αν δύο αυτοκίνητα συναντήθηκαν κάποια στιγμή,
- αν η κίνηση ενός αυτοκινήτου έγινε σε μια περιορισμένη περιοχή,
- ποια αυτοκίνητα πέρασαν από τον ίδιο χώρο και με ποια χρονική απόκλιση,
- κατά πόσο μια ορισμένη διαδρομή καλύφθηκε από κάποιο όχημα, κλπ.

Είναι βασικό να τονισθεί ότι για την απάντηση των περισσότερων ερωτημάτων, είναι απαραίτητο να ληφθεί υπόψη και ο χρόνος που εκτελέστηκε κάποια κίνηση και όχι μόνο η χωρική διάστασή της. Για παράδειγμα, ένα ερώτημα εισόδου σε μια περιοχή δεν θα μπορούσε να απαντηθεί αν δεν ήταν γνωστό ότι το κομμάτι της καμπύλης που βρίσκεται έξω από την περιοχή προηγήθηκε χρονικά εκείνου εντός περιοχής. Άρα το αντικείμενο εισήλθε στην περιοχή και όχι το αντίθετο. Αυτό έρχεται να επιβεβαιώσει την χωροχρονική φύση της τροχιάς, ως ενός τρισδιάστατου φαινομένου που εξαρτάται από την κίνηση στο χώρο και την χρονική στιγμή που εκτελείται η κίνηση αυτή.

Ένα σημαντικό τοπολογικό ερώτημα είναι η εύρεση αντικειμένων που είχαν κοινές πορείες (join query). Λόγω της μεγάλης του συσχέτισης με το αντικείμενο της εργασίας αυτής κρίθηκε σκόπιμο να γίνει λεπτομερής ανάλυσή του σε επόμενη ενότητα (βλεπε ενότητα 2.4).



Σχήμα 2.6: Τοπολογικά ερωτήματα (Πηγή: [16])



Σχήμα 2.7: Ερωτήματα πλοήγησης.

2.3.2 Ερωτήματα πλοήγησης

Τα ερωτήματα πλοήγησης έρχονται να δώσουν απαντήσεις σε δυναμικά ερωτήματα, υπό την έννοια ότι οι απαντήσεις δεν προκύπτουν άμεσα από τα διαθέσιμα δεδομένα αλλά εξάγονται ως δευτερογενής πληροφορία.

Τέτοια ερωτήματα είναι μεταξύ άλλων η εύρεση της

- ταχύτητας
- κατεύθυνσης
- διανυθείσας απόστασης κ.α.

ενός ή περισσότερων αντικειμένων.

Τα ερωτήματα πλοήγησης στηρίζονται κυρίως στη χρονική διάσταση της τροχιάς. Για παράδειγμα, η ταχύτητα ενός αντικειμένου για κάποια περίοδο προκύπτει από τη διανυθείσα απόσταση διά τον χρόνο που χρειάστηκε για να καλύψει την απόσταση αυτή. Το χρονικό διάστημα που λαμβάνεται υπόψη για οποιοδήποτε υπολογισμό είναι επίσης καθοριστικής σημασίας, αφού για παράδειγμα η κατεύθυνση ενός αντικειμένου για μερικά λεπτά μπορεί να είναι βόρεια, αλλά αν λάβουμε υπόψη μας ένα μεγαλύτερο διάστημα μιας ώρας μπορεί να προκύψει πως το αντικείμενο είχε κατεύθυνση προς τα βορειοδυτικά.

Τα ερωτήματα πλοήγησης είναι αρκετά σημαντικά και συνήθως είναι χρήσιμα σε εφαρμογές πραγματικού χρόνου. Στην εργασία αυτή, παίζουν αρκετά σημαντικό ρόλο, αφού η κατεύθυνση των αντικειμένων είναι μέγιστης σημασίας για τον αλγόριθμο που προτείνεται.

Στο σχήμα 2.7 βλέπουμε ένα παράδειγμα τροχιάς με τα στίγματα που έχουν ληφθεί ως δείγματα. Από τα δεδομένα μπορούν να προκύψουν οι εξής πληροφορίες:

- μέση ταχύτητα: $\bar{v}_i = \frac{1}{j} \cdot \sum_{i=2}^j \frac{\text{dist}(p_i, p_{i-1})}{(t_i - t_{i-1})}$
- μέγιστη ταχύτητα: $v_{max} = \max(\bar{v}_i)$
- προσανατολισμός: κατεύθυνση $(t_1, t_3) = \text{βορειοανατολικά}$
κατεύθυνση $(t_1, t_j) = \text{ανατολικά}$

- διανυθείσα απόσταση: $\sum_{i=2}^j dist(p_i, p_{i-1})$

2.4 Συνδυασμός τροχιών

Ο συνδυασμός τροχιών ανήκει στα τοπολογικά ερωτήματα και γενικά αναφέρεται ως ερώτημα σύζευξης τροχιών (*trajectory join*). Αυτό δεν είναι το μοναδικό ερώτημα που αφορά συνδυασμό τροχιών, αφού υπάρχουν κι άλλες σχετικές κατηγορίες ερωτημάτων όπως η συσταδοποίηση τροχιών (*trajectory clustering*) και η εύρεση σημαντικών μονοπατιών (*hot motion path discovery*). Λόγω της μεγάλης του συσχέτισης με την εργασία αυτή κρίθηκε σκόπιμο να αναπτυχθεί με μεγαλύτερη λεπτομέρεια σε αυτή την ενότητα. Άλλωστε, η πολυπλοξία είναι μια παραλλαγή του ερωτήματος σύζευξης με τη διαφορά ότι εισάγει επιπλέον περιορισμούς ως προς το χρόνο επεξεργασίας αφού ο υπολογισμός πρέπει να γίνεται σε πραγματικό χρόνο.

Το ερώτημα σύζευξης παίρνει ως είσοδο ένα σύνολο τροχιών και γενικά καλείται να βρει τροχιές που ήταν ίδιες/παραπλήσιες, αλλά όχι κατ' ανάγκη την ίδια χρονική στιγμή. Ένα τέτοιο ερώτημα στηρίζεται κυρίως στη χωρική μορφή της τροχιάς. Τα εκάστοτε αποτελέσματα του ερωτήματος εξαρτώνται κυρίως από το είδος της ζητούμενης εφαρμογής και έχουν αναπτυχθεί διάφοροι αλγόριθμοι για να εντοπίζουν κάθε φορά απαντήσεις με συγκεκριμένες προδιαγραφές.

Οι κυριότερες προσεγγίσεις κινούνται στα πλαίσια της εξόρυξης δεδομένων (*data mining*). Η μεθοδολογία της εξόρυξης δεδομένων υπαγορεύει την συλλογή δεδομένων από μία πολύ μεγάλη βάση δεδομένων προκειμένου να εξαχθούν κάποια συμπεράσματα. Η συλλογή αυτή συνήθως απαιτεί εκτεταμένη χρήση εξειδικευμένων δομών δεδομένων, αλγορίθμων και τεχνικών προκειμένου τα συμπεράσματα που ζητούνται να μπορούν να υπολογιστούν σε αποδεκτά χρονικά πλαίσια. Η εξόρυξη δεδομένων είναι κυρίως *offline* διαδικασία λόγω του μεγάλου χρόνου που απαιτεί και της προϋπόθεσης για ύπαρξη όλης της ποσότητας της απαιτούμενης πληροφορίας, κάτι που είναι συνήθως μη διαθέσιμο σε *online* διαδικασίες. Παρακάτω παρουσιάζονται προσεγγίσεις που έγιναν για αντιμετώπιση ερωτημάτων σύζευξης και σχετικών.

2.4.1 Σύζευξη τροχιών

Η αφελής προσέγγιση στον υπολογισμό ενός ερωτήματος σύζευξης (*trajectory join*) θα ήταν να συγκριθεί κάθε τροχιά με όλες τις υπόλοιπες και να εντοπιστούν αυτές που ικανοποιούν το ερώτημα (λ.χ να βρίσκονται σε κοντινή απόσταση μεταξύ τους). Κάτι τέτοιο είναι τρομερά χρονοβόρο και σε περιπτώσεις πολύ μεγάλων βάσεων δεδομένων γίνεται απαγορευτικό.

Αντ' αυτού, οι Bakalov et al. [1] προτείνουν μία μέθοδο για αποτελεσματικότερη εκτέλεση του ερωτήματος. Προτού εκτελέσουν τις συγκρίσεις, χρη-

σιμοποιούν ένα αλγόριθμο προκειμένου να περιορίσουν τις ομάδες των τροχιών που ενδεχομένως έχουν κοινά τμήματα. Συμπιέζουν αρχικά τις τροχιές, ώστε να προκύψει μια απολούστερη αναπαράσταση και μετά διακριτοποιούν τη χωρική διάσταση σε στάθμες ώστε να πάρουν μια κωδικοποίηση της συμπιεσμένης τροχιάς. Κάθε σημείο της εκάστοτε τροχιάς κωδικοποιείται σε μια στάθμη. Δημιουργούν έτσι συμβολικές αναπαραστάσεις για κάθε τροχιά και συγκρίνουν αυτές ώστε να βρουν ποιες τροχιές πιθανόν να βρίσκονται κοντά. Με αυτό το τρόπο πετυχαίνουν να μειώσουν το σύνολο των συγκρίσεων για εύρεση τροχιών με κοινά μέρη σε μεγάλο βαθμό από την αφελή προσέγγιση. Επίσης, ο αλγόριθμος μπορεί να εντοπίσει κοινά τμήματα και δεν περιορίζεται στον εντοπισμό τροχιών που είναι κοινές σε όλο τους το μήκος.

Με το ίδιο πρόβλημα ασχολήθηκαν οι Lee et al. [9]. Στην εργασία τους προτείνουν μία μέθοδο για εντοπισμό κοινών τμημάτων τροχιών και όχι κατ' ανάγκη ολόκληρων. Το αποτέλεσμα του αλγορίθμου είναι μία συστάδα από προσεγγίσεις τμημάτων των πραγματικών τροχιών και μία αντιπροσωπευτική τροχιά που αναδεικνύει την τάση της κίνησης. Για να το πετύχουν αυτό, εκτελούν συμπίεση των τροχιών αλλά αντί να βρίσκουν μέσο όρο των σημείων της τροχιάς χρησιμοποιώντας ένα κυλιόμενο παράθυρο (όπως [1]), εντοπίζουν τα σημεία όπου η τροχιά αποκλίνει απότομα και δημιουργούν γραμμικά τμήματα που ξεκινούν και τελειώνουν σε τέτοια σημεία. Στη συνέχεια βρίσκουν τμήματα από τις τροχιές που έχουν κοντινή απόσταση και τέλος βλέποντας την κατεύθυνση αυτών υπολογίζουν ένα γενικό διάνυσμα κίνησης που περιγράφει τα αντίστοιχα τμήματα.

2.4.2 Εύρεση σημαντικών μονοπατιών

Ένα παρεμφερές ερώτημα με αυτό της σύζευξης, είναι αυτό της εύρεσης μονοπατιών/διαδρομών αυξημένης σημασίας (Hot route discovery). Μια διαδρομή θεωρείται σημαντική όταν ένας μεγάλος αριθμός αντικειμένων την διατρέχει σε αντίθεση με άλλες δυνατές διαδρομές. Οι διαδρομές μπορούν να είναι προκαθορισμένες ή να κατασκευάζονται από τμήματα διαδρομών που όταν συνενωθούν δημιουργούν μία νέα διαδρομή.

Με το πρόβλημα αυτό ασχολήθηκαν οι Chen et al. [2] οι οποίοι υλοποίησαν έναν αλγόριθμο προκειμένου να εντοπίζουν δημοφιλείς διαδρομές μεταξύ δύο δεδομένων τοποθεσιών, λαμβάνοντας υπόψη τροχιές οι οποίες εκτελέστηκαν από κινούμενα αντικείμενα στο παρελθόν. Ο αλγόριθμος βρίσκει κοινούς κόμβους μεταξύ τροχιών και στη συνέχεια δημιουργεί ένα δίκτυο μεταβάσεων (transfer network) μεταξύ τους. Για κάθε συνδυασμό κόμβων υπολογίζει την πιθανότητα μετάβασης από τον ένα κόμβο σε άλλο βάσει των κινήσεων που είχαν εκτελέσει κινούμενα αντικείμενα στο παρελθόν. Τέλος, αξιοποιώντας την πληροφορία αυτή, είναι σε θέση να βρει με δεδομένα έναν αρχικό και ένα τελικό κόμβο, την πιο δημοφιλή διαδρομή από τον πρώτο στο δεύτερο.

Με το ίδιο θέμα ασχολήθηκαν και οι Sacharidis et al. [19] με στόχο την εύ-

ρηση δημοφιλών μονοπατιών αλλά σε πραγματικό χρόνο. Από τον αλγόριθμό τους προκύπτει μία λίστα από μονοπάτια (ευθύγραμμα τμήματα) που παρουσιάζονται διαβαθμισμένα ανάλογα με τον αριθμό των αντικειμένων που τα ακολούθησαν. Η προσέγγισή τους υλοποιεί μια κατανεμημένη τεχνική για τη δημιουργία περίληψης της τροχιάς κάθε αντικειμένου στην πλευρά του χρήστη και στη συνέχεια αποθήκευσής της σε κεντρικό υπολογιστή (συντονιστής). Ο συντονιστής στη συνέχεια αναλαμβάνει την ευθύνη τήρησης των μονοπατιών και της καταγραφής του αριθμού των αντικειμένων που διάβηκαν κάθε μονοπάτι αν αυτό προκύψει κοινό για πολλαπλά αντικείμενα. Ο αλγόριθμος επιστρέφει το μονοπάτι με τη μεγαλύτερη σημασία, όπως αυτή προκύπτει ως συνάρτηση του αριθμού των αντικειμένων που διάβηκαν το μονοπάτι και του μήκους του.

2.4.3 Πομπές αντικειμένων

Το πρόβλημα της εύρεσης πομπών αντικειμένων (*convoy discovery*) είναι στην ουσία ένα αυστηρότερο ερώτημα σύζευξης τροχιών, με την έννοια ότι ζητείται να βρεθούν αντικείμενα που είχαν ίδια τροχιά, δηλαδή κινούνταν μαζί ταυτόχρονα για κάποιο χρονικό διάστημα και σε πολύ μικρή απόσταση μεταξύ τους. Η πομπή μπορεί να χαρακτηρίζεται από τον αριθμό των αντικειμένων που λαμβάνουν μέρος όπως και από το χρόνο που αυτή πραγματοποιείται. Το πρόβλημα αντιμετώπισαν οι Jeung et al. [25], οι οποίοι προτείνουν τεχνικές στην εργασία τους για την εύρεση τέτοιων ομάδων αντικειμένων.

Μια απλή μέθοδος που θα μπορούσε να εφαρμοστεί για την εύρεση πομπών στηρίζεται στην τεχνική της συσταδοποίησης σημείων βάση πυκνότητας (*density-based clustering*). Η συσταδοποίηση των θέσεων των αντικειμένων γίνεται σε κάθε χρονική στιγμή (με το βήμα ως παράμετρο) βρίσκοντας αντικείμενα κοντινά μεταξύ τους. Τα κοινά αντικείμενα μεταξύ των ομάδων που βρέθηκαν σε διαδοχικές στιγμές θεωρητικά περιγράφουν μια πομπή. Η συσταδοποίηση σε κάθε χρονική στιγμή απαιτεί την ύπαρξη ενός στίγματος για κάθε αντικείμενο· αν αυτά απουσιάζουν, απαιτείται ο υπολογισμός τους με τη μέθοδο της παρεμβολής. Ακόμα, το κόστος της συσταδοποίησης είναι μεγάλο. Λαμβάνοντας υπόψη πως γίνεται για όλα τα αντικείμενα σε κάθε χρονική στιγμή, το κόστος εκτέλεσης της μεθόδου αυτής είναι πολύ μεγάλο.

Ως δεύτερη προσέγγιση προτείνεται η απλοποίηση των τροχιών των αντικειμένων ώστε να είναι ευκολότερη η εύρεση αυτών που έχουν κοινή τροχιά [?]. Η απλοποίηση επιτυγχάνεται με έναν αλγόριθμο απλοποίησης όπως αυτούς που περιγράφηκαν (λ.χ. Douglas-Peucker). Στη συνέχεια εκτελείται ένα ερώτημα συσταδοποίησης τροχιών χρησιμοποιώντας τις απλοποιημένες εκδοχές τους για εντοπισμό ομάδων τροχιών πολύ κοντινών και σύγχρονων μεταξύ τους. Λόγω των απλοποιημένων περιγραφών τους, η συσταδοποίηση επιτυγχάνεται πολύ πιο εύκολα από την αντίστοιχη με τις πραγματικές τροχιές. Από τις συστάδες που θα προκύψουν γίνεται συσταδοποίηση για άλλη μια φορά με

χρήση των πραγματικών τροχιών των αντικειμένων και οι επιθυμητές πομπές εντοπίζονται ως το αποτέλεσμα της συσταδοποίησης αυτής. Λόγω του ενδιάμεσου σταδίου συσταδοποίησης που εκτελεί ο αλγόριθμος επιτυγχάνει πολύ καλές επιδόσεις, αφού η πράξη της συσταδοποίησης στις λεπτομερείς τροχιές γίνεται μόνο για ένα πολύ μικρό υποσύνολο.

Κεφάλαιο 3

Πολυπλεξία τροχιών

Πολυπλεξία τροχιών είναι η διαδικασία ανάλυσης τροχιών δύο ή περισσότερων αντικειμένων με σκοπό την εξαγωγή μίας τροχιάς η οποία θα τις περιγράφει. Διάφορες μελέτες έχουν γίνει που να εντοπίζουν κοινά τμήματα τροχιών ή διαδρομές που διατρέχονται από πολλά αντικείμενα. Οι μέθοδοι που υλοποιήθηκαν στηρίζονται κυρίως σε τεχνικές εξόρυξης δεδομένων που απαιτεί μεγάλο κόστος επεξεργασίας με αποτέλεσμα να αποκλείεται η εκτέλεση σε πραγματικό (online) χρόνο. ...μελέτη βιβλιογραφίας. Σε αυτή τη διπλωματική εργασία παρουσιάζεται μία νέα προσέγγιση όπου οι τροχιές πολυπλέκονται on-line και σκοπός είναι να βρεθούν τα αντικείμενα που κινούνται τώρα στο χώρο, έχουν κοινή πορεία κάτω υπό καθορισμένες συνθήκες για κάποιο χρονικό διάστημα. Στη συνέχεια του κεφαλαίου παρουσιάζονται οι προδιαγραφές του προβλήματος που τίθεται και οι παραμέτροι από τις οποίες εξαρτάται. Έπεται η περιγραφή του αλγορίθμου πολυπλεξίας τροχιών που προτείνεται από αυτή την εργασία.

3.1 Τροχιές με κοινές ιδιότητες

Στο κεφάλαιο παρουσιάζουμε την έννοια της πολυπλεγμένης τροχιάς που προκύπτει από τον συνδυασμό δύο ή περισσότερων τροχιών. Ο συνδυασμός αυτός για να έχει νόημα πρέπει να γίνεται κάτω από συγκεκριμένες προϋποθέσεις και να τηρεί προδιαγραφές οι οποίες ορίζονται παρακάτω.

3.1.1 Χωρική εγγύτητα και Χρονική συνάφεια

Καταρχήν πρέπει να διευκρινιστεί ότι η πολυπλεξία τροχιών όπως αναλύεται σε αυτή την εργασία, δεν αφορά μία διαδικασία “άθροισης” τροχιών. Πρόκειται για μια διαδικασία που σκοπό έχει να εξάγει τα κοινά χαρακτηριστικά που μοιράζονται οι τροχιές που πολυπλέκονται σε πραγματικό χρόνο και να δημιουργήσει μία νέα τροχιά η οποία να τις αντιπροσωπεύει. Με αυτή

την παρατήρηση γίνεται εμφανές ότι δύο τροχιές που δεν μοιράζονται κανένα κοινό χωρικά σημείο (δηλαδή οι καμπύλες που τις περιγράφουν δεν τέμνονται) ή κανένα κοινό χρονικά σημείο (δηλαδή οι καμπύλες που τις περιγράφουν δεν τέμνονται σε κάποιο σημείο την ίδια περίπου χρονική στιγμή), δεν μπορούν να πολυπλεχθούν. Φυσικά, οι απαιτήσεις αυτές είναι πολύ αυστηρές. Προκειμένου να εντοπιστούν τροχιές οι οποίες όντως πολυπλέκονται θα πρέπει να χαλαρωθούν αυτές οι απαιτήσεις. Έτσι, τα σημεία των τροχιών είναι αρκετό να χαρακτηρίζονται από:

1. **Χωρική εγγύτητα:** Κάθε τροχιά, δημιουργείται από μία ακολουθία σημείων στον χώρο. Χωρική εγγύτητα είναι η ιδιότητα που χαρακτηρίζει δύο σημεία που βρίσκονται σε κοντινή απόσταση μεταξύ τους στο χώρο. Το μέγεθος της απόστασης αυτής δεν είναι σταθερό, αλλά αποτελεί παράμετρο ανάλογα με το μέγεθος της ακρίβειας που θέλουμε να επιτύχουμε κατά την προσέγγιση των τροχιών τους από μία τρίτη, η οποία είναι το αποτέλεσμα της πολυπλεξίας.
2. **Χρονική συνάφεια:** Δύο αντικείμενα τα οποία πέρασαν από το ίδιο σημείο σε διαφορετικές χρονικές στιγμές έχουν κάποιο κοινό σημείο στις κινήσεις τους. Χρονική συνάφεια μεταξύ των δύο τροχιών στο σημείο εκείνο υπάρχει όταν τα αντικείμενα πέρασαν από το σημείο αυτό σε χρονικές στιγμές που απέχουν μικρό χρονικό διάστημα μεταξύ τους. Το μέγεθος της χρονικής αυτής διαφοράς εξαρτάται από το πόσο μεγάλη χρονική ταύτιση επιδιώκεται να επιτευχθεί. Είναι προφανές ότι δύο αντικείμενα δεν μπορούν να περάσουν από το ίδιο σημείο την ίδια ακριβώς χρονική στιγμή και πως για να υπάρχει χρονική εγγύτητα σε ένα σημείο, πρέπει να υπάρχει υποχρεωτικά και χωρική εγγύτητα.

Οι πιο πάνω ορισμοί συνοψίζουν τις προδιαγραφές που πρέπει να τηρεί η διαδικασία της πολυπλεξίας, για κάθε σημείο κάθε τροχιάς που πρόκειται να ληφθεί υπόψη. Η δήλωση αυτή δεν είναι ξεκάθαρη αφού κάθε αντικείμενο που κινείται στο χώρο, ανανεώνει την θέση του σε ακαθόριστες χρονικές στιγμές και με αυθαίρετο αριθμό ανανεώσεων στην μονάδα του χρόνου. Για να επιτευχθεί πολυπλεξία, χρειάζεται να υπάρχει ταύτιση τροχιών μεταξύ των αντικειμένων.

3.1.2 Ταύτιση τροχιών

Γενικά μία τροχιά αποτελείται από μία αλληλουχία σημείων του χώρου. Θεωρούμε ένα σύνολο N αντικειμένων

$$O = \{o_1, o_2, \dots, o_N\}$$

Τα αντικείμενα στέλνουν ενημερώσεις για τις θέσεις τους στον χώρο σε αυθαίρετες χρονικές στιγμές και η ακολουθία των στιγμάτων αυτών αποκαλείται ως η προσεγγιστική τροχιά τους. Στην ιδανική περίπτωση, τα αντικείμενα θα

έστελναν τις ενημερώσεις των θέσεων τους την ίδια χρονική στιγμή. Όσων οι θέσεις, χαρακτηρίζονται από χωρική εγγύτητα (η χρονική συνάφεια είναι εγγυημένη από την υπόθεση που έγινε) για περισσότερες από μία ενημερώσεις, θα είναι και αυτά που θα έχουν κοινές τροχιές - για όσο χρόνο θα ισχύει η συνθήκη αυτή.

Το εγγενές πρόβλημα που υπάρχει με την δυναμική ενημέρωση των θέσεων των αντικειμένων είναι ότι κάθε αντικείμενο στέλνει αυθαίρετο αριθμό ενημερώσεων και σε αυθαίρετο χρόνο. Δηλαδή, κάποιο αντικείμενο μπορεί να στείλει τρία διαδοχικά στίγματα για μια χρονική περίοδο και ένα άλλο να στείλει μόνο ένα στίγμα στην ίδια περίοδο. Αναγκαστικά η ποικιλομορφία στα δεδομένα εισόδου πρέπει να λαμβάνεται υπόψη από τον αλγόριθμο πολυπλεξίας. Δύο αντικείμενα o_i, o_j που έχουν χωρική εγγύτητα για κάθε χρονική στιγμή ασχέτως του ρυθμού των ενημερώσεων θέσεων που κάνουν, έχουν ταύτιση των τροχιών τους και για την μεταξύ τους απόσταση ισχύει

$$d(o_i(t), o_j(t)) < \epsilon, \forall t \in (\text{Now} - \omega, \text{Now}]$$

Βάσει αυτών που έχουν ειπωθεί, ο αλγόριθμος πολυπλεξίας πρέπει να συνδιάζει τροχιές για αντικείμενα τα οποία δεν απέχουν μεταξύ τους περισσότερο από απόσταση ϵ για κάθε χρονική στιγμή. Το αποτέλεσμα της πολυπλεξίας θα πρέπει να είναι :

1. ομάδες αντικειμένων g_1, g_2, \dots που είχαν κοινή πορεία για καθορισμένο χρονικό διάστημα ω
2. μία ενδεικτική τροχιά για κάθε ομάδα g_k , της οποίας κάθε σημείο $\bar{o}(t)$ θα απέχει απόσταση το πολύ ϵ από κάθε αντίστοιχο συγχρονισμένο σημείο τροχιάς για όλα τα αντικείμενα της ομάδας, $d(o_i(t), \bar{o}(t)) < \epsilon, \forall t \in (\text{Now} - \omega, \text{Now}], \forall o_i \in g_k$

όπου $d(p_i, p_j)$ η ευκλείδεια απόσταση των σημείων p_i, p_j και ω το μέγεθος του κυλιόμενου παραθύρου (σε χρονόσημα).

3.1.3 Προδιαγραφές προβλήματος

Ο αλγόριθμος σχεδιάστηκε λαμβάνοντας υπόψη το εξής σενάριο λειτουργίας. Υπάρχει μία ομάδα N αντικειμένων στον δισδιάστατο χώρο. Τα αντικείμενα αυτά κινούνται αυθαίρετα και ενημερώνουν συνεχώς το σύστημα για την θέση τους. Δεν γίνεται καμμία υπόθεση για την φύση των αντικειμένων, άρα ούτε οι τροχιές τους αναμένεται να ακολουθούν κάποιο μοτίβο (λ.χ περιοδικότητα). Τα αντικείμενα μπορεί να είναι

- αυτοκίνητα σε οδικό δίκτυο
- πλοία στην ανοικτή θάλασσα

- ζώα σε ένα δάσος κλπ.

Τα αντικείμενα είναι γνωστά στο σύστημα και κάθε ένα από αυτά έχει μια προκαθορισμένη ταυτότητα *id* με την οποία το σύστημα το αναγνωρίζει. Οι ενημερώσεις φτάνουν στο σύστημα κατά μη περιοδικό τρόπο και με άγνωστο ρυθμό. Ωστόσο, υποτίθεται ότι κάθε αντικείμενο μπορεί να κάνει αυθαίρετο αριθμό ενημερώσεων στην μονάδα του χρόνου, αρκεί να έχει κάνει μια τουλάχιστον ενημέρωση κάθε β χρονικές στιγμές (βλέπε ενότητα 3.2.2).

3.2 Περιγραφή και παράμετροι

3.2.1 Γενική ιδέα

Καταρχήν, για την ορθή λειτουργία του αλγορίθμου, γίνεται η παραδοχή ότι κάθε αντικείμενο κάνει τουλάχιστον μία ενημέρωση θέσης πριν από κάθε εκτέλεση του αλγορίθμου. Μοναδικά δεδομένα εισόδου, είναι ένα ρεύμα από στίγματα θέσεων των κινούμενων αντικειμένων (timestamped locations). Ο αλγόριθμος, ύστερα από κάθε εκτέλεση, δίνει ως έξοδο ομάδες αντικειμένων που είχαν κοινή τροχιά για χρονικό παράθυρο ω (βλέπε 3.2.2) και μία τροχιά για κάθε ομάδα. Η τροχιά αυτή συνιστά την πολυπλεγμένη τροχιά που υπολογίστηκε από τις τροχιές των αντικειμένων για το χρονικό παράθυρο ω .

Ο αλγόριθμος είναι επαναληπτικός και σχεδιάστηκε για να τρέχει κάθε β χρονικές στιγμές (βλέπε 3.2.2). Δηλαδή, ο αλγόριθμος πολυπλεξίας έχει κύκλο εκτέλεσης β χρονόσημα.

Κύκλος εκτέλεσης μιας επαναληπτικής διαδικασίας, είναι το χρονικό διάστημα το οποίο τίθεται προκειμένου να εκτελεστεί και να τερματίσει μία επανάληψη αυτής της διαδικασίας.

Η διαδικασία μπορεί να ξανακληθεί μόνο μετά το πέρας αυτού του διαστήματος, ασχέτως αν η προηγούμενη εκτέλεση της κατάφερε να τερματίσει σε συντομότερο από τον διαθέσιμο χρόνο. Κατά τον κύκλο εκτέλεσης, παρέχονται κάποια δεδομένα ως δεδομένα εισόδου στην διαδικασία πριν την έναρξη του κύκλου, και χρησιμοποιούνται αποκλειστικά για όλη τη διάρκεια εκτέλεσης του κύκλου. Όταν ο κύκλος τερματίσει, η διαδικασία επιστρέφει τα αποτελέσματα που προέκυψαν κατά την εκτέλεση και είναι έτοιμη να λάβει νέα δεδομένα εισόδου για την εκτέλεση του επόμενου κύκλου.

Σε κάθε εκτέλεση, εξετάζει τις ενημερώσεις θέσεων που ελήφθησαν από την προηγούμενη εκτέλεση μέχρι τότε και υπολογίζει ένα διάνυσμα κίνησης που ‘συνοψίζει’ την κίνηση του αντικειμένου για τις τελευταίες β χρονικές στιγμές, χρησιμοποιώντας έναν αλγόριθμο εξομάλυνσης. Αυτό γίνεται για κάθε αντικείμενο. Ακολούθως, κάθε διάνυσμα κίνησης γενικεύεται σε μία προσεγγιστική κατεύθυνση σε σχέση με την τελευταία γνωστή θέση του αντικειμένου και τον ορίζοντα. Η ποιότητα της γενίκευσης αυτής αποτελεί παράμετρο

του αλγορίθμου (βλέπε 3.3.2). Η γενικευμένη αυτή κατεύθυνση στη συνέχεια κωδικοποιείται, προκειμένου να αποθηκευτεί χρησιμοποιώντας πιο αποδοτική συμβολική αναπαράσταση.

Για κάθε αντικείμενο, τηρείται μία αλληλουχία από γενικευμένες κατευθύνσεις που υπολογίστηκαν στο χρονικό παράθυρο ω . Η ιστορία αυτή περιγράφει την κίνησή του κατά το πιο πρόσφατο χρονικό διάστημα εύρους ω χρονοσήμων. Πρόκειται λοιπόν για μια σχηματική προσέγγιση της τροχιάς του, αφού δεν λαμβάνονται υπόψη οι συγκεκριμένες θέσεις του στον χώρο για την αναπαράσταση αυτή. Σε κάθε κύκλο εκτέλεσης, ενημερώνεται η ‘ιστορία’ κάθε αντικειμένου. Μετέπειτα, η σύνοψη αυτή χρησιμοποιείται για την ομαδοποίηση των αντικειμένων. Αυτό είναι ένα σημαντικό βήμα στην εκτέλεση του αλγορίθμου, αφού το αποτέλεσμα αυτής της λειτουργίας είναι να εντοπιστούν ομάδες αντικειμένων που είχαν ‘παράλληλες’ τροχιές για το χρονικό παράθυρο ω . Οι ομάδες αυτές δεν αντικατοπτρίζουν αντικείμενα που είχαν κοινές τροχιές, αλλά αντικείμενα που είχαν –περίπου– ανάλογες πορείες, δηλαδή τα ίχνη των τροχιών τους είχαν –περίπου– αντίστοιχο σχήμα.

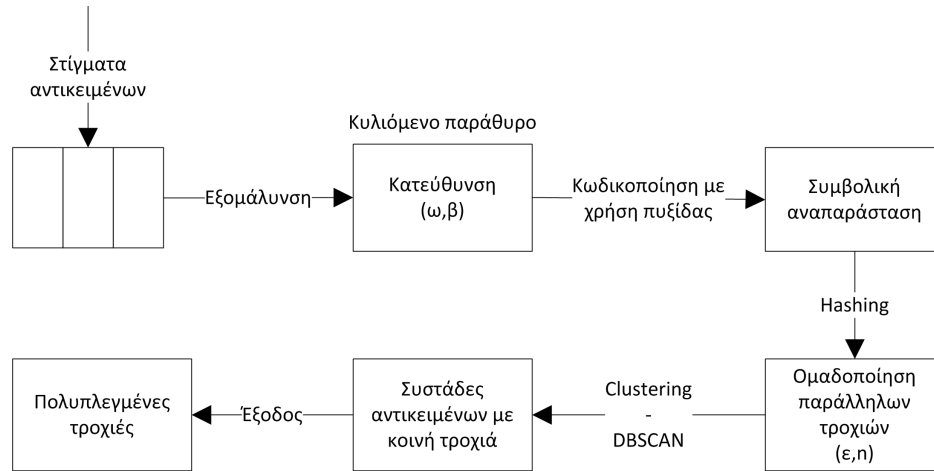
Για τον εντοπισμό των αντικειμένων που είχαν όχι μόνο ίδιο ίχνος τροχιάς αλλά και ίδια τροχιά, χρησιμοποιείται η τεχνική της συσταδοποίησης (clustering). Διαισθητικά, δεδομένου ότι δύο καμπύλες που είναι παράλληλες δεν μπορούν να ξεκινούν από διαφορετικά σημεία και να καταλήγουν στο ίδιο σημείο, χρησιμοποιούνται οι τελευταίες θέσεις των αντικειμένων για να βρεθούν εκείνα που βρίσκονται σε κοντινή απόσταση. Αντικείμενα τα οποία είχαν ίδιο ίχνος πορείας και βρίσκονται τώρα σε κοντινή απόσταση, συνιστούν αντικείμενα που χαρακτηρίζονται από παραπλήσιες τροχιές.

Τέλος, για κάθε ομάδα αντικειμένων που βρέθηκε να έχουν παραπλήσιες τροχιές, υπολογίζεται ένα σημείο ανά χρονικό διάστημα β ως το “κέντρο βάρους” των θέσεών τους σε αυτό το διάστημα και ως πολυπλεγμένη τροχιά της ομάδας επιστρέφεται η αλληλουχία των υπολογισμένων αυτών σημείων για το χρονικό παράθυρο ω .

Στο σχήμα 3.1 παρουσιάζεται το συνολικό διάγραμμα ροής δεδομένων της προτεινόμενης λύσης. Σε επόμενες ενότητες, αναλύεται το κάθε βήμα εκτέλεσης του αλγορίθμου λεπτομερώς.

3.2.2 Παράμετροι

Στην ενότητα αυτή συνοψίζονται οι παράμετροι του προβλήματος, όπως επίσης και οι παράμετροι που επηρεάζουν την λειτουργία του αλγορίθμου. Για κάθε παράμετρο, γίνεται ανάλυση της σημασίας της και εξηγούνται οι αναμενόμενες μεταβολές στην έξοδο του αλγορίθμου σε αναλογία με τη μεταβολή της τιμής της.



Σχήμα 3.1: Διάγραμμα ροής δεδομένων

Algorithm 1 Trajectory Multiplexer

- 1: **Procedure** *Coordinator* (sliding window ω , step β , accuracy α , minimum objects n , minimum distance ϵ)
 - 2: **Input:** Timestamped locations $\langle id, x, y, t \rangle$ for N moving objects.
 - 3: **Output:** A set of multiplexed trajectories for a time window ω .
 - 4: **for** each timestamp τ **do**
 - 5: Buffer incoming locations at time τ ;
 (Next calls are processed asynchronously)
 - 6: **if** τ is multiple of $\lceil \omega/\beta \rceil$ **then**
 - 7: Call *Smoothing*;
 - 8: Call *Encoding*(α);
 - 9: Call *Hash*;
 - 10: Call *DBSCAN*(n, ϵ);
 - 11: Call *Multiplex*;
 - 12: **end if**
 - 13: **end for**
 - 14: **End Procedure**
-

Χρονικό παράθυρο ω

Τα αντικείμενα κινούνται συνεχώς στο χώρο. Ο αλγόριθμος πολυπλεξίας λαμβάνει υπόψη την κίνηση των αντικειμένων μόνο για χρονικό διάστημα ω όπου η μονάδα μέτρησης του διαστήματος αυτού καθορίζεται από τη μονάδα μέτρησης των στιγμάτων που στέλνουν τα αντικείμενα (τάξη μεγέθους δευτερολέπτων, λεπτών κλπ).

Τα δεδομένα που τηρεί ο αλγόριθμος προκειμένου να εξάγει αποτελέσματα αναφέρονται μόνο στα πιο πρόσφατα ω χρονόσημα με σημείο αναφοράς την χρονική στιγμή εκτέλεσης του αλγορίθμου.

Ο αλγόριθμος εντοπίζει αντικείμενα τα οποία είχαν κοινή τροχιά, όπως επίσης και την τροχιά αυτή, μόνο για το χρονικό αυτό παράθυρο. Αν δύο αντικείμενα κινούνταν μαζί για χρόνο ίσο ή μεγαλύτερο από ω χρονικές στιγμές, τότε ο αλγόριθμος τα εντοπίζει και τα επιστρέφει. Αν η συνδυασμένη τους κίνηση εξελίσσεται για λιγότερο από ω στιγμές, τότε η κίνησή τους αυτή δεν θεωρείται σημασίας τέτοιας που να επιστραφεί ως έξοδος. Με άλλα λόγια, το παράθυρο ω καθορίζει για πόσο χρόνο πρέπει τουλάχιστον να κινούνται δύο αντικείμενα μαζί προκειμένου να θεωρείται ότι έχουν κοινή τροχιά.

Αυξάνοντας το παράθυρο ω , αυξάνεται και το ίχνος της τροχιάς των αντικειμένων το οποίο αποθηκεύεται, με αποτέλεσμα να γίνεται σπανιότερος ο εντοπισμός αντικειμένων με κοινή τροχιά, αφού τα αντικείμενα πρέπει να κινούνται συνδυασμένα για περισσότερο χρόνο προκειμένου να πληρούν τα κριτήρια πολυπλεξίας.

Βήμα κύλισης β

Ο αλγόριθμος λαμβάνει υπόψη συμβάντα που συμβαίνουν στο πρόσφατο χρονικό παράθυρο ω . Το υλοποιημένο σύστημα, δεν καλεί τον αλγόριθμο πολυπλεξίας σε κάθε χρονική στιγμή, αλλά τον εκτελεί κάθε β χρονόσημα. Ενδιάμεσα, το σύστημα λαμβάνει τις όποιες ενημερώσεις θέσεων των αντικειμένων και τις αποθηκεύει σε προσωρινό χώρο μνήμης ώστε να χρησιμοποιηθούν κατά τον επόμενο κύκλο εκτέλεσης. Ο αλγόριθμος λαμβάνει υπόψη συμβάντα που συμβαίνουν στο πρόσφατο χρονικό παράθυρο ω . Το υλοποιημένο σύστημα, δεν καλεί τον αλγόριθμο πολυπλεξίας σε κάθε χρονική στιγμή, αλλά τον εκτελεί κάθε β χρονόσημα. Ενδιάμεσα, το σύστημα λαμβάνει τις όποιες ενημερώσεις θέσεων των αντικειμένων και τις αποθηκεύει σε προσωρινό χώρο μνήμης ώστε να χρησιμοποιηθούν κατά τον επόμενο κύκλο εκτέλεσης.

Για την παράμετρο β , πρέπει να ισχύει η σχέση $\beta \leq \omega$. Σε κάθε κύκλο εκτέλεσης, υπολογίζονται οι κατευθύνσεις των αντικειμένων και ενημερώνεται η ιστορία της πορείας τους, με μία νέα κατεύθυνση. Βάσει αυτού, προκύπτει πως με δεδομένες τιμές για το ω και β , ο αλγόριθμος αποθηκεύει $\lceil \omega/\beta \rceil$ διακριτές κατευθύνσεις για κάθε αντικείμενο. Αν δεν τηρηθεί η συνθήκη, τότε ο αλγόριθμος πολυπλεξίας δεν επιστρέφει αποτέλεσμα, αφού δεν θα έχει ποτέ αποθηκευμένη αρκετή ιστορία για να εξάγει συμπεράσματα.

Όπως έχει προαναφερθεί, ανάμεσα στους κύκλους εκτέλεσης του αλγορίθμου, το σύστημα αποθηκεύει τις ενημερώσεις στιγμάτων. Κατά την εκτέλεση, ο αλγόριθμος λαμβάνει υπόψη όλα τα στίγματα ενός αντικειμένου για να εξάγει μία κατεύθυνση γι' αυτό χρησιμοποιώντας μία τεχνική εξομάλυνσης που περιγράφεται στην ενότητα 3.3.1. Κατά συνέπεια, μικρές τιμές του β οδηγούν σε πιο ακριβή περιγραφή της κίνησης των αντικειμένων, ενώ μεγάλες τιμές του β περιγράφουν πιο χονδρικά την τάση που έχει ένα αντικείμενο να κινείται προς μία κατεύθυνση για κάποιο χρονικό διάστημα.

Φυσικά, πιο λεπτομερής περιγραφή της κίνησης είναι περισσότερο επιρρεπής σε σφάλματα μετρήσεων της θέσης (λόγω υλικού GPS) αφού μια εσφαλμένη ανανέωση μπορεί να οδηγήσει σε προσωρινά λανθασμένη περιγραφή της κίνησης του αντικειμένου, η οποία εν συνεχεία οδηγεί σε ελλιπή αποτελέσματα της πολυπλεξίας. Αντίθετα, πάρα πολύ μεγάλες τιμές του β μπορεί να οδηγήσουν σε σημαντική απώλεια πληροφορίας, αφού λόγω της εξομάλυνσης θα απορροφούνται σύντομες μολονότι σημαντικές αλλαγές πορείας που μπορεί να είχε το αντικείμενο κατά την κίνησή του.

Η κατάλληλη τιμή του β εξαρτάται από την φύση των κινούμενων αντικειμένων που παρακολουθούνται, όπως επίσης και από το περιβάλλον στο οποίο κινούνται. Αν παρακολουθούνται αυτοκίνητα στο οδικό δίκτυο μιας πόλης όπου οι εναλλαγές πορείας είναι συχνές, μία μικρή τιμή του β θα ήταν θεμιτή ενώ για κινήσεις αυτοκινήτων σε επίπεδο νομού ή χώρας, μια μεγάλη τιμή του β θα ήταν καταλληλότερη ώστε να αγνοεί τις ασήμαντες αλλαγές πορείας των αντικειμένων.

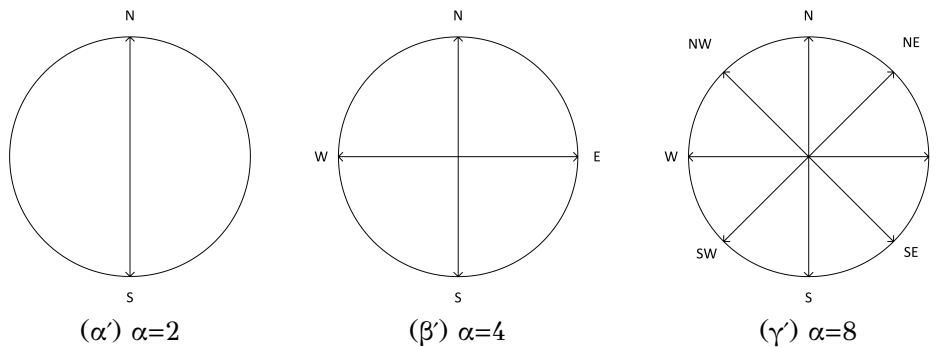
Ελάχιστο πλήθος αντικειμένων n

Κατά την εκτέλεση της πολυπλεξίας, δημιουργούνται ομάδες αντικειμένων που έχουν κοινή τροχιά. Είναι αρκετό να βρεθεί ομάδα δύο αντικειμένων που να έχουν κοινή τροχιά για να πολυπλεχθούν οι τροχιές τους. Όμως η ομάδα αυτή ίσως να μην είναι αρκετά σημαντική λόγω του μικρού αριθμού των αντικειμένων που περιγράφει. Η παράμετρος n δίνει την δυνατότητα να τεθεί ένα κάτω κατώφλι που να υπαγορεύει στον αλγόριθμο να επιστρέφει έξοδο πολυπλεγμένες τροχιές που να περιγράφουν τουλάχιστον n και περισσότερα αντικείμενα η κάθε μία.

Αν σε κάποια εκτέλεση δεν βρεθεί ομάδα με περισσότερα από n αντικείμενα, τότε ο αλγόριθμος δεν εκτελεί καμία διαδικασία πολυπλεξίας και επιστρέφει.

Ακρίβεια προσέγγισης ίχνους α

Μεταξύ των εκτελέσεων του αλγορίθμου πολυπλεξίας, στίγματα θέσης καταφθάνουν στο σύστημα από τα αντικείμενα. Αυτά, προτού ληφθούν υπόψη από τον αλγόριθμο, περνούν από ένα στάδιο εξομάλυνσης ώστε να προκύψει



Σχήμα 3.2: α) Κατευθύνσεις: βόρεια, νότια β) Κατευθύνσεις: βόρεια, δυτικά, νότια, ανατολικά γ) Κατευθύνσεις: βόρεια, βορειοδυτικά, δυτικά, νοτιοδυτικά, νότια, νοτιοανατολικά, ανατολικά, βορειοανατολικά

ένα διάνυσμα κατεύθυνσης που να περιγράφει την διεύθυνση του αντικειμένου βάσει των στιγμάτων αυτών.

Στη συνέχεια, αυτή η διεύθυνση, ανάγεται σε μία γενικευμένη διεύθυνση που προκύπτει από ένα σύνολο βασικών κατευθύνσεων (βλέπε ενότητα 3.3.1). Η παράμετρος α καθορίζει το μέγεθος του συνόλου αυτού. Μεγαλύτερες τιμές του α οδηγούν σε περισσότερες βασικές τροχιές και κατά συνέπεια σε καλύτερη προσέγγιση της κίνησης του αντικειμένου. Φυσικά, περιορίζοντας την τιμή του α γίνεται χειρότερη προσέγγιση στις κινήσεις, αλλά επιτρέπει την μελέτη τους σε ευρύτερη κλίμακα αφαιρώντας αχρείαστη λεπτομέρεια.

Η τιμή του α βολεύει να είναι δύναμη του δύο (2), ώστε να μπορούν να ισοκαταμεληθούν οι βασικές κατευθύνσεις στα σημεία του ορίζοντα. Στο σχήμα 3.2 παρουσιάζονται οι βασικές κατευθύνσεις που προκύπτουν για διάφορες τιμές της παραμέτρου.

Μέγιστη απόσταση αντικειμένων ϵ

Για να θεωρηθεί ότι δύο αντικείμενα έχουν κοινή τροχιά, πρέπει καθώς κινούνται να διατηρούν μεταξύ τους χωρική εγγύτητα. Το μέγεθος της εγγύτητας αυτής δεν είναι προκαθορισμένο αλλά μπορεί να οριστεί ανάλογα με τις ανάγκες του προβλήματος. Η παράμετρος ϵ περιγράφει την εγγύτητα αυτή, δηλαδή καθορίζει την απόσταση που πρέπει να έχουν δύο αντικείμενα ως προϋπόθεση για να θεωρείται πως κινούνται μαζί.

Για παράδειγμα, σε ένα οδικό δίκτυο, αν ενδιαφέρει να βρεθούν αυτοκίνητα που κινούνται μαζί στον αυτοκινητόδρομο, τότε η εγγύτητα τους πρέπει να είναι μεγάλη, δηλαδή να είναι αρκετά κοντά. Αυτό σημαίνει μικρή τιμή του ϵ , της τάξεως των μερικών δεκάδων μέτρων. Αν πάλι ενδιαφέρει η κοινή ομαδική κίνηση αυτοκινήτων σε μεγαλύτερο επίπεδο, για παράδειγμα η εύρεση αυτοκινήτων που κινούνται μακριά από το κέντρο της πόλης (αλλά το κά-

νουν αυτό ταυτόχρονα και προς την ίδια κατεύθυνση) τότε απαιτείται χρήση μεγαλύτερης τιμής για το ϵ .

3.3 Μεθοδολογία πολυπλεξίας τροχιών

Σε αυτήν την ενότητα παρουσιάζονται λεπτομερώς τα σημαντικά βήματα του προτεινόμενου αλγορίθμου για πολυπλεξία τροχιών. Στην ενότητα *εξομάλυνση κίνησης* εξηγείται ο αλγόριθμος προσέγγισης της κατεύθυνσης των αντικειμένων. Στην ενότητα *προσέγγιση πορείας*, αναλύεται ο τρόπος που προσεγγίζεται η πορεία κάθε αντικειμένου προκειμένου να χρησιμοποιηθεί ως ένδειξη για τον εντοπισμό αντικειμένων με κοινή τροχιά. Στην ενότητα *ομαδοποίηση αντικειμένων βάσει παράλληλων τροχιών*, περιγράφεται το πρώτο βήμα της διαδικασίας εντοπισμού αντικειμένων με κοινή τροχιά που ολοκληρώνεται στην ενότητα *εύρεση αντικειμένων με κοινή τροχιά*. Τέλος, στην ενότητα *εύρεση πολυπλεγμένης τροχιάς ομάδας αντικειμένων* εξηγείται πώς υπολογίζονται οι πολυπλεγμένες τροχιές των αντικειμένων που βρέθηκαν να έχουν κοινή πορεία.

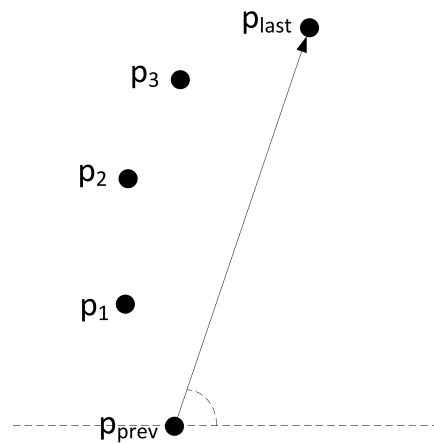
3.3.1 Εξομάλυνση κίνησης

Ο αλγόριθμος πολυπλεξίας εκτελείται κάθε β χρονόσημα. Κατά την στιγμή της εκτέλεσής του, λαμβάνονται υπόψη όσα στίγματα υπάρχουν διαθέσιμα από κάθε κινούμενο αντικείμενο. Κάθε νέα ενημέρωση θέσης που φτάνει κατά τη διάρκεια ενός κύκλου εκτέλεσης, αποθηκεύεται ώστε να ληφθεί υπόψη κατά τον επόμενο κύκλο. Έτσι, σε διάστημα β χρονικών στιγμών, κάθε αντικείμενο o_i έχει στείλει n_i ενημερώσεις θέσης, με την απαίτηση $n_i \geq 1 \forall_i$. Σκοπός της διαδικασίας εξομάλυνσης είναι να λάβει ως είσοδο τα n_i στίγματα κάθε αντικειμένου και να υπολογίσει ένα διάνυσμα κίνησης που να συνοψίζει όσο το δυνατόν καλύτερα την κίνησή του τις τελευταίες β χρονικές στιγμές.

Τα στίγματα που φτάνουν από κάθε αντικείμενο αποθηκεύονται σε μία προσωρινή μνήμη (buffer). Με αυτό το τρόπο, μπορεί να αποθηκευτεί αυθαίρετος αριθμός στιγμάτων και να υλοποιηθούν διάφοροι αλγόριθμοι εξομάλυνσης, ανάλογα με το πεδίο εφαρμογής. Ο αλγόριθμος πολυπλεξίας σχεδιάστηκε ώστε να μπορεί να υποστηρίξει ποικιλία αλγορίθμων εξομάλυνσης, των οποίων η χρήση θα καθορίζεται παραμετρικά και παρέχει την δυνατότητα ανάπτυξης νέων αλγορίθμων και ενσωμάτωσής τους με μηδαμινό κόστος.

Απλός αλγόριθμος εξομάλυνσης: Ο απλούστερος μηχανισμός εξομάλυνσης λαμβάνει υπόψη μόνο την πιο πρόσφατη θέση του αντικειμένου, και σε συνδυασμό με την τελευταία θέση που είχε κατά τον αμέσως προηγούμενο κύκλο εκτέλεσης, υπολογίζει το διάνυσμα κίνησης του αντικειμένου. Η πιο πάνω διαδικασία παρουσιάζεται στο σχήμα 3.3.

Το αποτέλεσμα της εξομάλυνσης είναι ένα διάνυσμα που περιγράφει την

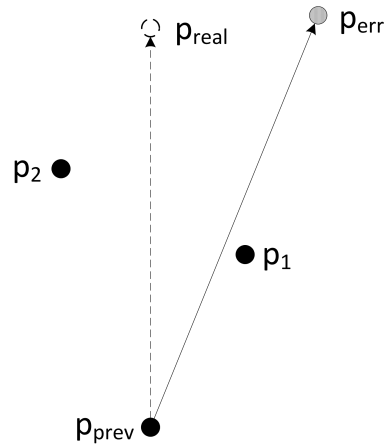


Σχήμα 3.3: Απλός μηχανισμός εξομάλυνσης

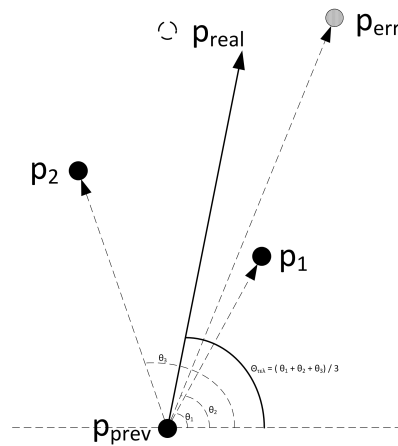
κατεύθυνση του αντικειμένου σε σχέση με την τελευταία του ενημέρωση κατά τον προηγούμενο κύκλο εκτέλεσης. Η σημαντική πληροφορία στο διάνυσμα αυτό, είναι η γωνία που σχηματίζει το διάνυσμα με τον άξονα x θεωρώντας καρτεσιανό σύστημα συντεταγμένων για τα στίγματα κάθε αντικειμένου. Η γωνία αυτή θα χρησιμοποιηθεί σε επόμενο βήμα για την προσέγγιση του ίχνους της τροχιάς (βλέπε 3.3.2). Μεγάλο πλεονέκτημα του αλγορίθμου είναι οι μειωμένες απαιτήσεις σε μνήμη, αφού από τα n_i σημεία θέσης του αντικειμένου αρκεί να τηρείται μόνο το τελευταίο, οδηγώντας σε ταχύτατη επεξεργασία. Εμπεριέχει όμως το μεγάλο μειονέκτημα ότι απορρίπτει πληροφορία, αφού τα ενδιάμεσα στίγματα δεν λαμβάνονται υπόψη, πράγμα που οδηγεί σε αποτελέσματα μειωμένης ακρίβειας, ακόμη και λανθασμένα, όπως μπορεί να φανεί στο σχήμα 3.4.

Ο μηχανισμός εξομάλυνσης, λαμβάνει υπόψη μόνο το πιο πρόσφατο σημείο και το τελευταίο σημείο κατά την προηγούμενη εκτέλεση. Αν το σημείο χαρακτηρίζεται από μεγάλο σφάλμα λόγω ανακρίβειών $grps$, το διάνυσμα κατεύθυνσης που προκύπτει δεν είναι αντιπροσωπευτικό για την πορεία του αντικειμένου

Αλγόριθμος εξομάλυνσης μέσης διεύθυνσης: Επιδιώκοντας περισσότερη ακρίβεια στην προσέγγιση της κατεύθυνσης των αντικειμένων, προτείνεται ένας αλγόριθμος εξομάλυνσης, ο οποίος λαμβάνει υπόψη όλες τις ενημερώσεις θέσης που έκανε το αντικείμενο τα τελευταία β χρονόσημα. Συγκεκριμένα, ο αλγόριθμος υπολογίζει τα διανύσματα κατεύθυνσης που προκύπτουν από διαδοχικά σημεία θέσης του αντικειμένου και τα αθροίζει υπολογίζοντας έτσι ένα διάνυσμα που εκφράζει την συνισταμένη των επιμέρους διανυσμάτων. Εκτελώντας αυτή την διαδικασία, ο αλγόριθμος αφομοιώνει καλύτερα τα σφάλματα που προκύπτουν από την ανακρίβεια των στιγμάτων θέσης και επιτυγχάνει να δώσει μία ορθότερη κατεύθυνση που προσεγγίζει την πραγματική κατεύθυνση του αντικειμένου, όπως φαίνεται και στο σχήμα 3.5.



Σχήμα 3.4



Σχήμα 3.5: Ο μηχανισμός εξομάλυνσης μέσης διεύθυνσης, λαμβάνει υπόψη όλα τα σημεία θέσης που έστειλε το αντικείμενο από την προηγούμενη εκτέλεση της πολυπλεξίας. Το αποτέλεσμα που προκύπτει είναι μία μέση κατεύθυνση που προκύπτει από τις επιμέρους κατευθύνσεις του αντικειμένου.

3.3.2 Προσέγγιση πορείας

Κατά την εκτέλεση του αλγορίθμου πολυπλεξίας, μετά την διαδικασία εξομάλυνσης τροχιάς για κάθε αντικείμενο, προκύπτει ένα διάνυσμα κατεύθυνσης για το καθένα, που προσεγγίζει την πιο πρόσφατη τροχιά του, εξάγοντας πληροφορία από τις τελευταίες ενημερώσεις θέσεως που έχει κάνει. Σκοπός του αλγορίθμου πολυπλεξίας είναι να εντοπίσει αντικείμενα με παραπλήσιες τροχιές όπως αυτές έχουν οριστεί στην ενότητα 3.1. Για να επιτευχθεί αυτό σε πραγματικό χρόνο, έπρεπε να βρεθεί μια μέθοδος η οποία θα έκανε αποδοτικά τη σύγκριση των τροχιών σε σχέση με τον χρόνο διάρκειας β του κύκλου εκτέλεσης.

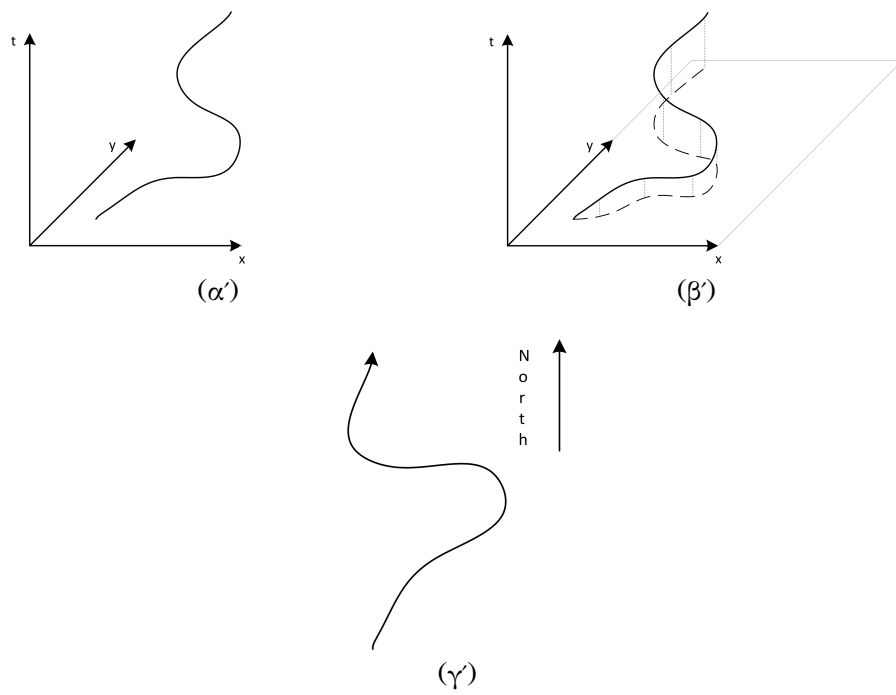
Η προφανής μέθοδος εντοπισμού παραπλήσιων τροχιών, θα έπαιρνε ως είσοδο τα επί μέρους διανύσματα κατεύθυνσης των αντικειμένων για όλο το παράθυρο ω , θα έβρισκε την τροχιά κάθε αντικειμένου ενώνοντάς τα και στη συνέχεια, με αλληπάλληλες συγκρίσεις μεταξύ των καμπυλών όλων των αντικειμένων, θα προσπαθούσε να εντοπίσει αντικείμενα με παραπλήσια τροχιά. Η μέθοδος αυτή είναι εξαντλητική όσον αφορά την διαθέσιμη πληροφορία, και όσο το δυνατόν ακριβέστερη λαμβάνοντας υπόψη πως η τροχιά που υπολογίζεται για κάθε αντικείμενο είναι προσεγγιστική (ενότητα 2.1). Εντούτοις, ο χρόνος εκτέλεσης μιάς τέτοιας διαδικασίας είναι απαγορευτικός όταν πρέπει να ληφθούν υπόψη τροχιές αντικειμένων της τάξεως των εκατοντάδων χιλιάδων, πόσο μάλλον όταν οι υπολογισμοί αυτοί πρέπει να γίνονται σε πραγματικό χρόνο.

Για να μειωθεί το κόστος εκτέλεσης της σύγκρισης, απαιτείται γενίκευση της πληροφορίας που να μειώνει τις αναγκαίες συγκρίσεις μεταξύ των τροχιών. Για το λόγο αυτό, αποφασίστηκε πρώτα να χρησιμοποιηθεί η πορεία των αντικειμένων και όχι η ίδια η χωροχρονική τροχιά τους.

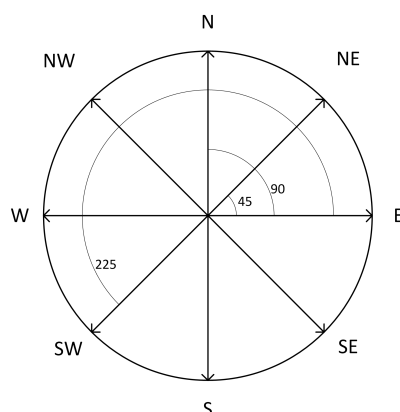
Πορεία ενός αντικειμένου ισοδυναμεί με την καμπύλη που προκύπτει από την προβολή της τροχιάς του στο επίπεδο του χώρου xy , είναι δηλαδή η καμπύλη που περιγράφει την κίνηση του αντικειμένου χωρίς να συνδέεται με κάποιο συγκεκριμένο σημείο του χώρου και χωρίς να αποθηκεύει χρονική πληροφορία.

Η έννοια της πορείας παρουσιάζεται στο σχήμα 3.6. Μετέπειτα, προκλήθηκε μία αναπαράσταση της πορείας αυτής που θα περιέγραφε τις διάφορες κατευθύνσεις με διακριτό τρόπο ώστε να είναι ευκολότερη η επεξεργασία τους.

Κύρια παράμετρος της αναπαράστασης, είναι η γωνία που σχηματίζει κάποιο διάνυσμα κατεύθυνσης με τον άξονα x , θεωρώντας πως τα αντικείμενα κινούνται σε καρτεσιανό σύστημα συντεταγμένων. Ορίζεται εξ αρχής ένα σύνολο γωνιών το οποίο ισοκατανέμεται στον τριγωνομετρικό κύκλο. Το σύνολο αυτό μπορεί να αποτελείται από δύο, τέσσερις, οκτώ, δεκαέξι ή και περισσότερες γωνίες ανάλογα με την ακρίβεια που θέλουμε να πετύχουμε κατά την περιγραφή των πραγματικών κατευθύνσεων. Κάθε γωνία περιγράφει ένα



Σχήμα 3.6: α)η τροχιά ενός αντικειμένου, β)η προβολή της τροχιάς στο επίπεδο, γ)η πορεία του αντικειμένου.



Σχήμα 3.7: Σύνολο οκτώ γωνιών. Από αυτό προκύπτει πώς θα οριστούν 8 σημεία στον ορίζοντα. Ο βορράς βρίσκεται στις 90° ενώ τα νοτιοδυτικά βρίσκονται στις 225° .

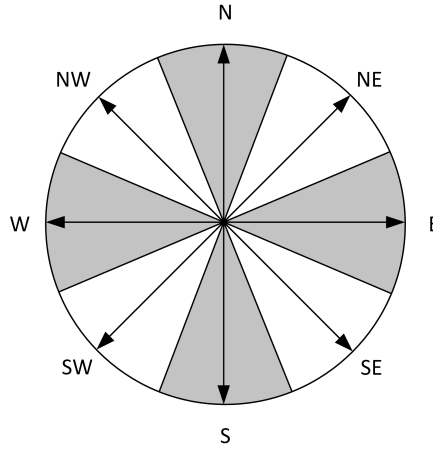
σημείο του ορίζοντα. Για παράδειγμα, στις 90° βρίσκεται ο βορράς. Παραδείγματα τέτοιων γωνιών φαίνονται στο σχήμα 3.7. Έχοντας ορίσει το πλήθος των βασικών κατευθύνσεων (ο αριθμός των γωνιών), ο τριγωνομετρικός κύκλος χωρίζεται σε ισομερείς τομείς των οποίων η διχοτόμος συμβολίζει μία από τις βασικές κατευθύνσεις (σχήμα 3.8). Η όλη προσέγγιση δουλεύει ως μια εικονική ‘πυξίδα’ που περιγράφει τα σημεία του ορίζοντα. Βάσει αυτού του διαχωρισμού, κάθε κατεύθυνση αντικειμένου που υπολογίζεται από το βήμα εξομάλυνσης κίνησης, ανάγεται σε μία εκ των βασικών κατευθύνσεων ανάλογα με τον τομέα του κύκλου εντός του οποίου εμπίπτει.

Η αναγωγή αυτή γίνεται σε κάθε επανάληψη του αλγορίθμου πολυπλεξίας και αντί να αποθηκεύεται κάθε φορά η κατεύθυνση του αντικειμένου που υπολογίζεται στο βήμα εξομάλυνσης, αποθηκεύεται η αντίστοιχη βασική κατεύθυνση. Η αποθήκευση των βασικών αυτών κατευθύνσεων δημιουργεί χονδρικά σε βάθος χρόνου την ιστορία της πρόσφατης κίνησης - πορείας κάθε αντικειμένου.

Συμβολική αναπαράσταση κίνησης θα αποκαλείται η κωδικοποιημένη ιστορία της πρόσφατης κίνησης του αντικειμένου.

Η απόκλιση που προκύπτει σε σχέση με την πραγματική κίνηση, εξαρτάται μεν από την ακρίβεια της αναγωγής (ο αριθμός των επιλεγμένων βασικών κατευθύνσεων), είναι δε ασήμαντη για τους σκοπούς του αλγορίθμου αφού πρέπει να προσεγγιστεί η πορεία κάθε αντικειμένου ώστε να είναι εφικτό μετέπειτα να εντοπιστούν αντικείμενα με κοινή τροχιά. Δημιουργώντας μία προσέγγιση της πορείας, ο εντοπισμός αντικειμένων με κοινή τροχιά ανάγεται πρώτα στον εντοπισμό αντικειμένων που είχαν παρόμοια πορεία και μετέπειτα στον εντοπισμό αυτών που όντως είχαν κοινή τροχιά.

Με τη μέθοδο αυτή, αν ένα αντικείμενο κινείται συνεχώς βόρεια κατά το χρονικό παράθυρο ω , ενώ κάποιο άλλο κινείται συνεχώς νότια, δεν θα υπάρξει



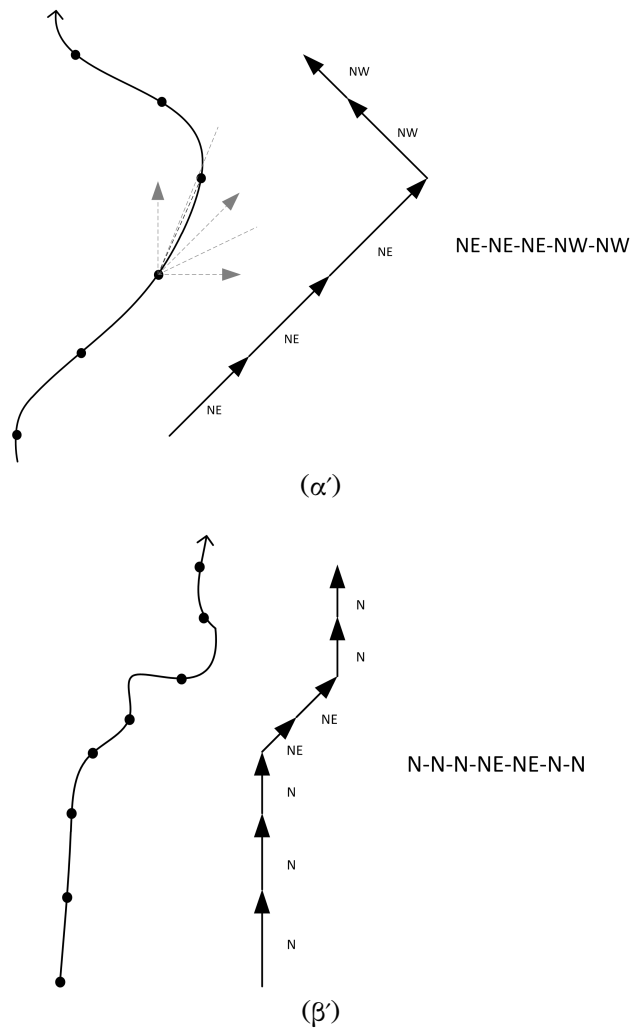
Σχήμα 3.8: Εικονική “πυξίδα” οκτώ κατευθύνσεων. Ο τριγωνομετρικός κύκλος χωρίζεται σε ισομερείς τομείς, των οποίων η διχοτόμος αντιπροσωπεύει μία εκ των βασικών κατευθύνσεων.

ποτέ ανάγκη σύγκρισης των τροχιών αυτών των αντικειμένων αφού η συμβολική αναπαράσταση της κίνησής τους δεν συμβαδίζει, οπότε είναι σίγουρο πως ούτε η λεπτομερής πορεία τους πρόκειται άρα ούτε και η τροχιά τους.

Στο σχήμα 3.9, παρουσιάζονται αντικείμενα με την πραγματική πορεία τους και αντίστοιχα την συμβολική αναπαράσταση της κίνησης που επιτυγχάνεται εφαρμόζοντας την τεχνική αυτού του βήματος.

Algorithm 2 Encoding algorithm

- 1: **Procedure** *Encoding*
 - 2: **Input:** An angle θ_i for each object o_i .
 - 3: **Output:** For each object o_i , a symbolic representation composed of its $\lceil \omega/\beta \rceil$ most recent basic directions.
 - 4: $SR \leftarrow \{SR_i : \text{sequence of its } \lceil \omega/\beta \rceil \text{ most recent basic directions } \tilde{\theta}_i, \forall o_i\}$;
 - 5: **for** each angle θ_i **do**
 - 6: $\tilde{\theta}_i \leftarrow$ Map input angle into *compass* C ;
 - 7: $SR_i \leftarrow SR_i \cup \{\tilde{\theta}_i\}$;
 - 8: $\tilde{\theta}_i^{old} \leftarrow$ the oldest item in SR_i ;
 - 9: $SR_i \leftarrow SR_i - \{\tilde{\theta}_i^{old}\}$; //Sliding Window
 - 10: **end for**
 - 11: **End Procedure**
-



Σχήμα 3.9: Παραδείγματα συμβολικών αναπαράστασεων

3.3.3 Ομαδοποίηση αντικειμένων βάσει παράλληλων τροχιών

Στην προηγούμενη ενότητα παρουσιάστηκε η μέθοδος προσέγγισης της πορείας των κινούμενων αντικειμένων. Χρησιμοποιώντας την προσέγγιση αυτή, δηλαδή τη συμβολική αναπαράσταση κίνησης, που περιγράφει αδρά την κίνηση των αντικειμένων σε βάθος χρόνου, επιδιώκεται να εντοπιστούν αντικείμενα που είχαν κοινή πορεία στο χρονικό παράθυρο ω .

Για να εντοπιστούν αντικείμενα που πιθανόν να έχουν κοινή τροχιά, θα πρέπει να συγκριθούν όλα τα αντικείμενα μεταξύ τους και να βρεθούν αυτά που έχουν ίδια συμβολική αναπαράσταση κίνησης (προσεγγιστική πορεία). Μεταξύ αυτών θα βρίσκονταν τα αντικείμενα που πέραν της κοινής πορείας είχαν και κοινή τροχιά. Για να γίνει δυνατή η επίτευξη της σύγκρισης αυτής, έπρεπε να επινοηθεί μία μέθοδος που θα έκανε τη διαδικασία αυτή γρήγορη, εξαφανίζοντας την ανάγκη σύγκρισης κάθε αντικειμένου με τα υπόλοιπα.

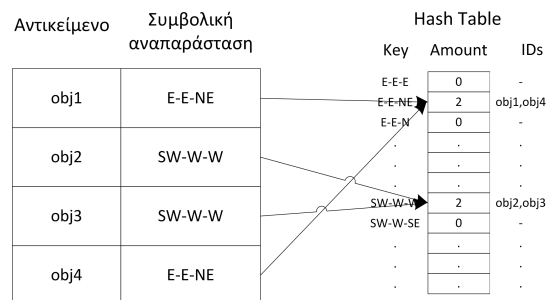
Το ζητούμενο αποτέλεσμα επιτεύχθηκε υλοποιώντας μία δομή πίνακα κατακερματισμού (hash table) όπου ως κλειδί θα χρησιμοποιούνται οι συμβολικές αναπαραστάσεις κίνησης των αντικειμένων. *Parapompi se biblio [????]* Σε κάθε κύκλο εκτέλεσης του αλγορίθμου, μετά την ενημέρωση της ιστορίας κίνησης κάθε αντικειμένου στο βήμα 3.3.2, γίνεται η ανάθεση των αντικειμένων στον πίνακα κατακερματισμού. Κάθε αντικείμενο με διαφορετική συμβολική αναπαράσταση κίνησης (για βάθος χρόνου ω), ανατίθεται σε διαφορετικό κάδο (bucket) του πίνακα. Όσα αντικείμενα έχουν ίδια συμβολική αναπαράσταση, θα ανατεθούν στον ίδιο κάδο. Έχοντας χρόνο εκτέλεσης $O()$ (όπου o αριθμός των αντικειμένων) επιτυγχάνεται ομαδοποίηση των αντικειμένων βάσει της προσεγγιστικής πορείας τους.

Κάθε κάδος συλλέγει μία ομάδα αντικειμένων που είχαν ίδια συμβολική αναπαράσταση, όχι όμως και ίδια τροχιά (βλέπε ενότητα 3.3.4). Στη συνέχεια, όσες ομάδες περιέχουν αριθμό αντικειμένων μεγαλύτερο από το ελάχιστο πλήθος αντικειμένων n , ελέγχονται για εύρεση αντικειμένων με κοινή τροχιά.

Στο σχήμα 3.10 παρουσιάζεται η τεχνική της ομαδοποίησης με την χρήση πίνακα κατακερματισμού με την διαδοχική ανάθεση τεσσάρων αντικειμένων

3.3.4 Εύρεση αντικειμένων με κοινή τροχιά

Κατά την ομαδοποίηση των αντικειμένων χρησιμοποιώντας τις συμβολικές αναπαραστάσεις κίνησής τους, προκύπτουν ομάδες οι οποίες χαρακτηρίζονται από ίδια πορεία. Αυτό δεν είναι αρκετό για να υποδείξει πως τα αντικείμενα μιας συγκεκριμένης ομάδας έχουν όντως κοινή τροχιά. Για παράδειγμα, ένα αντικείμενο που κινείται συνεχώς βόρεια στο κέντρο της πόλης και ένα άλλο αντικείμενο που κινείται το ίδιο αλλά σε έναν επαρχιακό δρόμο, θα ανήκουν στην ίδια ομάδα βάσει του κατακερματισμού αλλά φυσικά δεν έχουν κοινή τροχιά.



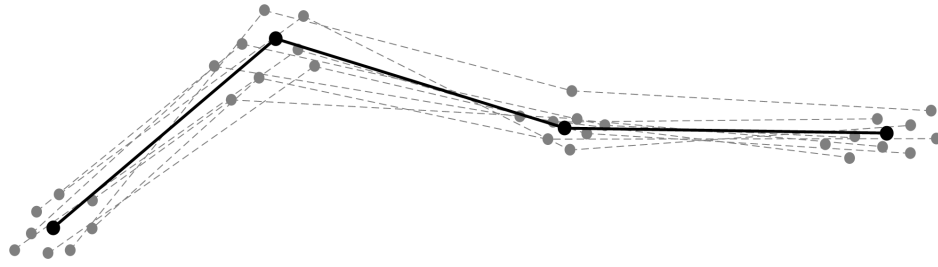
Σχήμα 3.10: Ομαδοποίηση αντικειμένων με ίδια συμβολική αναπαράσταση με χρήση hash table.

Προκειμένου ανάμεσα στα αντικείμενα της ομάδας να βρεθούν αυτά που είχαν όντως κοινή τροχιά, χρησιμοποιήθηκε η μέθοδος της συσταδοποίησης. Η βάση αυτής της προσέγγισης στηρίζεται στο γεγονός ότι δύο καμπύλες που ξεκινούν από το ίδιο ή παραπλήσιο σημείο, θα καταλήγουν στο ίδιο ή σε παραπλήσιο σημείο αν αυτές είναι παράλληλες. Άρα, δύο αντικείμενα που χαρακτηρίζονται τώρα από χωρική εγγύτητα, και για το χρονικό παράθυρο ω είχαν αντίστοιχες συμβολικές αναπαραστάσεις κίνησης, άρα και προσεγγιστικά παράλληλες τροχιές, συμπεραίνεται ότι για όλο το χρονικό παράθυρο χαρακτηρίζονταν από χωρική εγγύτητα. Οπότε τα δύο αντικείμενα είχαν κοινή τροχιά για το χρονικό παράθυρο ω . Το μέγεθος της εγγύτητας που θα χαρακτηρίζει τα αντικείμενα αυτά σε διάφορες στιγμές του παραθύρου, εξαρτάται άμεσα από την ακρίβεια της προσέγγισης που χρησιμοποιήθηκε κατά τη δημιουργία της συμβολικής αναπαράστασης της κίνησής τους. Επομένως, αρκεί να εξετάζεται η εγγύτητα του πιο πρόσφατου στίγματος κάθε αντικειμένου της ομάδας.

Ο αλγόριθμος συσταδοποίησης δέχεται ως είσοδο ένα σύνολο από σημεία στο χώρο και επιστρέφει συστάδες σημείων που ικανοποιούν χωρικές συνθήκες. Κάθε συστάδα εγγυάται πως περιλαμβάνει σημεία που δεν απέχουν από τα κοντινά τους απόσταση μεγαλύτερη από ϵ . Για τη συσταδοποίηση στον αλγόριθμο πολυπλεξίας επιλέχθηκε να χρησιμοποιηθεί ο αλγόριθμος DBSCAN που προτάθηκε στο [3], λόγω των ξεχωριστών επιδόσεών του.

Ο αλγόριθμος DBSCAN, γενικεύει την ιδέα της ελάχιστης απόστασης, εντοπίζοντας συστάδες αντικειμένων που το καθένα έχει τουλάχιστον ένα άλλο αντικείμενο σε απόσταση μικρότερη από το ϵ και όχι όλα τα αντικείμενα της συστάδας. Η ιδέα αυτή διαφέρει από την εύρεση συστάδων όπως παρουσιάστηκε προηγουμένως. Η χρήση αυτού του αλγορίθμου στη περίπτωση που εξετάζεται είναι επαρκής αφού αντικείμενα που απέχουν πάρα πολύ μεγάλη απόσταση μεταξύ τους εν τέλει θα έχουν και διαφορετικές συμβολικές αναπαραστάσεις, άρα δεν θα προκύπτουν μαζί ως είσοδος στον αλγόριθμο (βλέπε παράρτημα Β').

Για την εύρεση αντικειμένων με κοινή τροχιά, παρέχεται ως είσοδος στον αλγόριθμο συσταδοποίησης η τελευταία γνωστή θέση κάθε αντικειμένου από



Σχήμα 3.11: Οι συστάδες σημείων ορίζονται για τις χρονικές στιγμές εκτέλεσης του αλγορίθμου. Ανάμεσα σε κάθε συστάδα παρατηρείται ένα έντονο σημείο που είναι το κέντρο βάρους της συστάδας. Η καμπύλη που ενώνει τα κέντρα βάρους είναι η πολυπλεγμένη τροχιά των αντικειμένων για το χρονικό παράθυρο ω .

μία ομάδα που είχαν ίδια συμβολική αναπαράσταση κίνησης μεταξύ τους και επιστρέφονται όσες συστάδες αντικειμένων εντοπιστούν. Κάθε συστάδα συνιστά μια ομάδα αντικειμένων που είχαν κοινή τροχιά.

3.3.5 Εύρεση πολυπλεγμένης τροχιάς ομάδας αντικειμένων

Αφού εντοπιστούν τα αντικείμενα τα οποία είχαν κοινή τροχιά για το χρονικό παράθυρο ω απομένει να βρεθεί η ίδια η αντιπροσωπευτική τους τροχιά. Όπως έχει οριστεί στην ενότητα 3.1, η πολυπλεγμένη τροχιά πρέπει να απέχει γεωμετρικά όσο το δυνατό λιγότερο από τις τροχιές που περιγράφει (που είναι εν γένει ανομοιογενείς). Για τον υπολογισμό της, χρησιμοποιήθηκε μια διαδικασία εξαγωγής των διακριτών σημείων της από τα αντίστοιχα στίγματα των αντικειμένων που περιγράφει.

Για κάθε αντικείμενο, υπάρχει αποθηκευμένη η πρόσφατη ιστορία των κινήσεών του. Συγκεκριμένα, σε κάθε κύκλο εκτέλεσης και κατά την εξομάλυνση τροχιάς, υπολογίζεται ένα αντιπροσωπευτικό σημείο για κάθε αντικείμενο. Αν το αντικείμενο είχε στείλει μονάχα μία ενημέρωση θέσης, τότε αυτό το σημείο τηρείται αυτόματα ως αντιπροσωπευτικό.

Για τον υπολογισμό της πολυπλεγμένης τροχιάς τους, υπολογίζεται επαναληπτικά το κέντρο βάρους των στιγμάτων των αντικειμένων με κοινή τροχιά για κάθε διακριτή χρονική στιγμή t , από την τωρινή στιγμή μέχρι και ω χρονόσημα πριν με βήμα $[\omega/\beta]$. Δηλαδή, υπολογίζεται το κέντρο βάρους με είσοδο τα σημεία που αποθηκεύτηκαν στις τελευταίες $[\omega/\beta]$ εκτελέσεις του αλγορίθμου. Το αποτέλεσμα του υπολογισμού είναι μια αλληλουχία στιγμάτων που περιγράφει την πολυπλεγμένη τροχιά των αντικειμένων.

Η πιο πάνω διαδικασία συνοψίζεται στο σχήμα 3.11.

Algorithm 3 Multiplexing algorithm

```

1: Procedure Multiplex
2: Input: A set of objects  $O_g$  in the same cluster  $g$ .
3: Output: A sequence of  $\lceil \omega/\beta \rceil$  timestamped locations  $\langle x, y, t \rangle$  that represent
   the approximated trajectory of the object set  $O$ .
4:  $AL \leftarrow \{AL_i : \text{sequence of its } \lceil \omega/\beta \rceil \text{ most recent locations } \langle \tilde{x}, \tilde{y}, t \rangle, \forall o_i\}$ ;
5:  $MT \leftarrow \emptyset$ ;
6: for each timestep  $t_j$  in  $(t_{now-\lceil \omega/\beta \rceil}, t_{now})$  do
7:    $(X_j^{centr}, Y_j^{centr}) \leftarrow$  Aggregated location of objects  $o_i \in O$ ;
8:    $MT \leftarrow MT \cup \{\langle X_j^{centr}, Y_j^{centr}, t_j \rangle\}$ ;
9: end for
10: Return  $MT // MT \leftarrow$  sequence of timestamped locations  $\langle x_k^{centr}, y_k^{centr}, t_k \rangle, k \in$ 
     $[1, \lceil \omega/\beta \rceil]$ ;
11: End Procedure

```

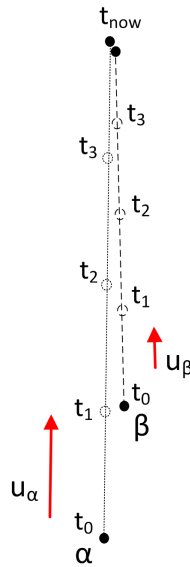
3.3.6 Παθολογικές περιπτώσεις

Ο αλγόριθμος, όντας απωλεστικός (lossy), δεν εντοπίζει με αριθμητικό τρόπο τα αντικείμενα που πολυπλέκονται αλλά χρησιμοποιεί προσεγγιστικές μεθόδους για την επίτευξη του αποτελέσματος. Λόγω της προσέγγισης πορείας που γίνεται, ενδέχεται σε κάποιες περιπτώσεις να προκύψουν σφάλματα στην έξοδο. Τα σφάλματα αυτά εντοπίζονται σε δύο ατέλειες της προσέγγισης. Ο αλγόριθμος δεν λαμβάνει υπόψη την διαφορά ταχυτήτων μεταξύ των αντικειμένων και ούτε ελέγχει τις αρχαιότερες θέσεις των αντικειμένων κατά τη στιγμή της πολυπλεξίας για χωρική εγγύτητα.

Διαφορά ταχυτήτων

Η πρώτη παθολογική περίπτωση εμφανίζεται στο σχήμα 3.12. Τα δύο αντικείμενα βρίσκονται αρχικά μακριά το ένα από το άλλο αλλά η οριζόντια απόσταση μεταξύ τους είναι μικρή (η τετμημένη των αντικειμένων είναι περίπου ίδια). Αν το αντικείμενο α κινείται με μεγαλύτερη ταχύτητα από το β τότε κάποια στιγμή στο μέλλον τα δύο αντικείμενα πρόκειται να βρεθούν σε κοντινή απόσταση μεταξύ τους. Ο αλγόριθμος πολυπλεξίας λαμβάνει υπόψη του μόνο τις αλλαγές πορείας των αντικειμένων και την τελευταία γνωστή τους θέση προκειμένου να εντοπίσει αντικείμενα με κοινή τροχιά. Στην προκειμένη περίπτωση, ο αλγόριθμος τη χρονική στιγμή t_{now} πρόκειται να εντοπίσει πως τα δύο αντικείμενα είχαν κοινή πορεία για το χρονικό παράθυρο ω (έστω $\omega = 4$) πράγμα που δεν ισχύει αφού τα αντικείμενα την χρονική στιγμή t_0 βρίσκονταν σε μεγάλη απόσταση μεταξύ τους.

Παθολογικές περιπτώσεις σαν αυτή δεν συνιστούν σοβαρό πρόβλημα αφού το σφάλμα εμφανίζεται στιγμιαία. Αν τα αντικείμενα συνεχίσουν να έχουν διαφορά στην ταχύτητά τους, τότε στις επόμενες χρονικές στιγμές πρόκειται



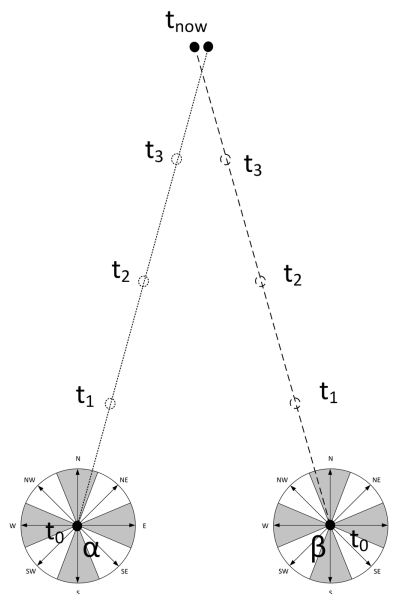
Σχήμα 3.12: Αντικείμενα με ίδια κατεύθυνση αλλά διαφορετική ταχύτητα.

πάλι να απομακρυνθούν μεταξύ τους. Αφού ο αλγόριθμος λαμβάνει υπόψη του την τελευταία θέση των αντικειμένων, κατά την εκτέλεση της πολυπλεξίας σε επόμενες χρονικές στιγμές αυτά δεν πρόκειται να εντοπιστούν ως αντικείμενα με κοινή πορεία.

Σφάλμα πυξίδας

Το δεύτερο παθολογικό σενάριο εντοπίζεται σε περιπτώσεις που δύο αντικείμενα βρίσκονται σε μεγάλη απόσταση μεταξύ τους, αλλά η κίνηση τους ανάγεται στην ίδια κατεύθυνση κατά την εξομάλυνση κίνησης. Το σφάλμα οφείλεται στην προσέγγιση της κίνησης που γίνεται με βάση την πυξίδα. Τα δύο αντικείμενα βρίσκονται σε μεγάλη απόσταση μεταξύ τους. Λόγω της κίνησής τους τείνουν να συγκλίνουν κάποια στιγμή στο μέλλον. Επειδή η κατεύθυνση των δύο αντικειμένων βρίσκεται στα όρια ενός τμήματος της πυξίδας, τότε αυτά εντοπίζονται να έχουν ίδια συμβολική αναπαράσταση για τις χρονικές στιγμές που παρουσιάζονται στο σχήμα. Όταν αυτά συναντηθούν, ο αλγόριθμος θα εντοπίσει πως τα δύο αντικείμενα είχαν κοινή τροχιά και θα πολυπλέξει τις τροχιές τους.

Το σφάλμα αυτό εξαρτάται άμεσα από την ακρίβεια που τίθεται για την πυξίδα κατά την εκτέλεση του αλγορίθμου. Μεγαλύτερες τιμές της ακρίβειας μειώνουν το εύρος των ισομερών τομών της πυξίδας μειώνοντας έτσι και την μέγιστη απόσταση που μπορούν να έχουν δύο αντικείμενα προκειμένου η κίνησή τους να ανάγεται στην ίδια γενικευμένη κατεύθυνση.

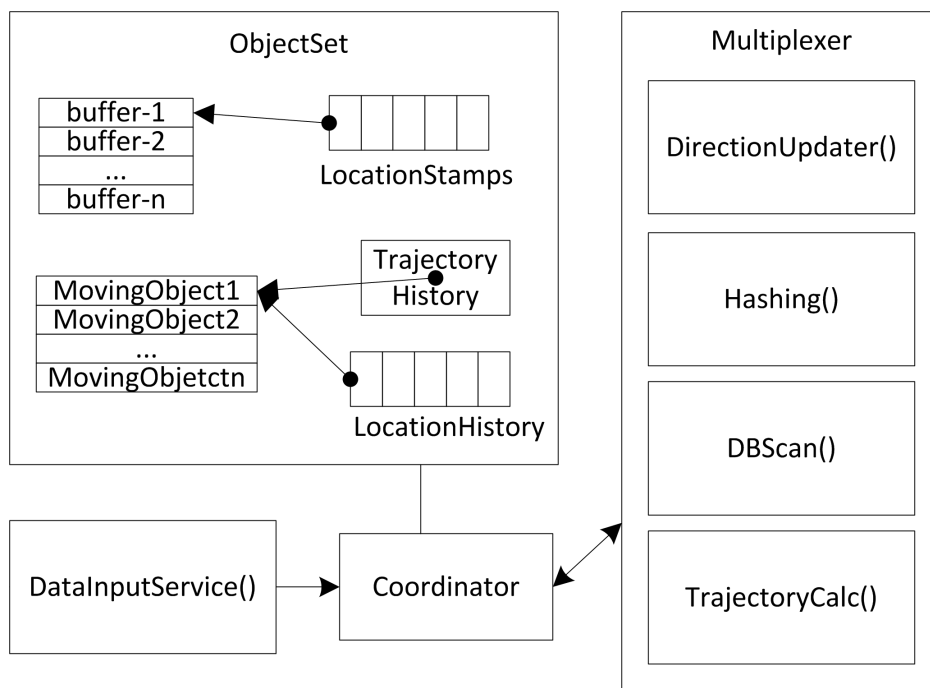


Σχήμα 3.13: Αντικείμενα με διαφορετική αρχική θέση. Η προσέγγιση της πυξίδας εντοπίζει τα δύο να έχουν κοινή τροχιά γεγονός που δεν ισχύει.

Κεφάλαιο 4

Πειραματική μελέτη

4.1 Αρχιτεκτονικό διάγραμμα



Σχήμα 4.1: Αρχιτεκτονικό διάγραμμα εφαρμογής

Ο αλγόριθμος της πολυπλεξίας υλοποιήθηκε στα πλαίσια της εργασίας προκειμένου να δοκιμαστεί τόσο σε επιδόσεις όσο και σε ποιότητα αποτελεσμάτων. Η υλοποίηση έγινε σε γλώσσα προγραμματισμού C++ με χρήση της Standard Library (STL) και σε κάποιες περιπτώσεις της βιβλιοθήκης Boost. Η

ανάπτυξη της εφαρμογής έγινε σε περιβάλλον Microsoft Windows7 με χρήση του Visual Studio 2010.

Στο σχήμα 4.1 παρουσιάζεται σε απλοποιημένη μορφή το αρχιτεκτονικό διάγραμμα της εφαρμογής περιλαμβάνοντας τα κυριότερα συστατικά που απαρτίζουν την εφαρμογή. Όλη η εφαρμογή ελέγχεται από τον έναν συντονιστή που αναλαμβάνει την μεταβίβαση των δεδομένων μεταξύ των συστατικών, καθ' ένα από τα οποία είναι υπεύθυνο για την εκτέλεση μιας συγκεκριμένης λειτουργίας. Τα συστατικά είτε συνιστούν κλάσεις αντικειμένων είτε συναρτήσεις.

4.2 Μετρήσεις

4.2.1 Πειραματικά δεδομένα

Για την εκτέλεση των πειραμάτων χρησιμοποιήθηκαν συνθετικά δεδομένα που παρήχθησαν με χρήση εξειδικευμένου λογισμικού. Τα δεδομένα αποθηκεύτηκαν σε ψηφιακή μορφή όπου και εισήχθησαν στον αλγόριθμο με τη χρήση ενός αρχείου εισόδου.

Τα πειραματικά δεδομένα των κινούμενων αντικειμένων παρήχθησαν βάσει ενός ψηφιακού χάρτη του οδικού δικτύου του πολεοδομικού συγκροτήματος Αθηνών. Το συγκεκριμένο ψηφιακό υπόβαθρο αφορά το βασικό οδικό δίκτυο της πρωτεύουσας, προέρχεται από χάρτες κλίμακας 1:5000 και τηρείται σε διανυσματική (vector) μορφή, ενώ καλύπτει έκταση περίπου 300 τ.χλμ. Οι οδικοί άξονες διακρίνονται σε κατηγορίες (λεωφόροι ταχείας κυκλοφορίας, κύριες και δευτερεύουσες αρτηρίες, βοηθητικοί δρόμοι), και χαρακτηρίζονται από την μέση ταχύτητα κίνησης των οχημάτων στην διάρκεια της ημέρας, όπως έχει προκύψει από επιτόπιες μετρήσεις. Αυτό ακριβώς το στοιχείο μπορεί να αξιοποιηθεί για τον υπολογισμό του μέσου χρόνου διαδρομής ενός οχήματος κατά μήκος των συνδέσμων του δικτύου, καθιστώντας αυτήν την γεωγραφική βάση δεδομένων κατάλληλη για υπολογισμό της βέλτιστης διαδρομής (shortest path) μέσα στην πόλη.

Με χρήση του λογισμικού ArcView GIS 3.2 και της επέκτασής του Network Analyst 1.0b, δημιουργήθηκαν συνολικά 10000 τροχιές ισάριθμων αντικειμένων, θέτοντας ως προέλευση και προορισμό κάθε διαδρομής τυχαία επιλεγμένα ζεύγη κόμβων του δικτύου. Οι κινήσεις διεξάγονται ως επί το πλείστον ακτινικά, θεωρώντας ότι τα περισσότερα αντικείμενα ξεκινούν από την περιφέρεια, διέρχονται από το κέντρο της πόλης και κατευθύνονται προς κάποιο προάστειο. Κατόπιν, από κάθε τροχιά έγινε δειγματοληψία σημειακών θέσεων, λαμβάνοντας συνολικά 200 στίγματα ανά τροχιά με ίση χρονική απόσταση μεταξύ τους ώστε να αντιπροσωπεύουν τακτικές ενημερώσεις της θέσης των αντικειμένων. Τελικά, προέκυψε ένα αρχείο τροχιών με εγγραφές που φέρουν την ταυτότητα (ID) του αντικειμένου, τις συντεταγμένες (x , y) και το χρονόσημο (t).



Σχήμα 4.2: Πειραματικά δεδομένα αποτυπωμένα στο χάρτη.

4.2.2 Αποτελέσματα επιδόσεων

Όπως παρουσιάστηκε στην ενότητα 3.2.2, ο αλγόριθμος εξαρτάται από τις παραμέτρους ω , β , ϵ , n , α . Πειραματικές μετρήσεις έγιναν για τους πιο κάτω συνδυασμούς τιμών:

Παράμετρος	Τιμή		
Κυλιόμενο παράθυρο ω	10	20	50
Βήμα εκτέλεσης β	2	2, 4, 5, 10	5,10
Ελάχιστη απόσταση ϵ	100, 200		
Ελάχιστο πλήθος αντικειμένων n	10, 20 , 50, 100		
Ακρίβεια πυξίδας α	8, 16 , 32		

Από τις μετρήσεις, προέκυψαν αποτελέσματα που αφορούν:

- Χρόνο εκτέλεσης ανά κύκλο εκτέλεσης,
- Επιμέρους χρόνους εκτέλεσης αλγορίθμου,
- Αριθμό συστάδων που εντοπίστηκαν,
- Ποσοστό πολυπλεγμένων αντικειμένων (βαθμός πολυπλεξίας).

Οι μετρήσεις έγιναν σε υπολογιστή με τις εξής προδιαγραφές:

- CPU: Intel Core 2 Duo P8600 @ 2.40GHz(x2)

- Μνήμη: 4,00 GB

Ακολουθούν διαγράμματα που επιδεικνύουν με ποιό τρόπο επηρεάζουν οι παράμετροι τις επιδόσεις του αλγορίθμου.

Συνολικός χρόνος εκτέλεσης

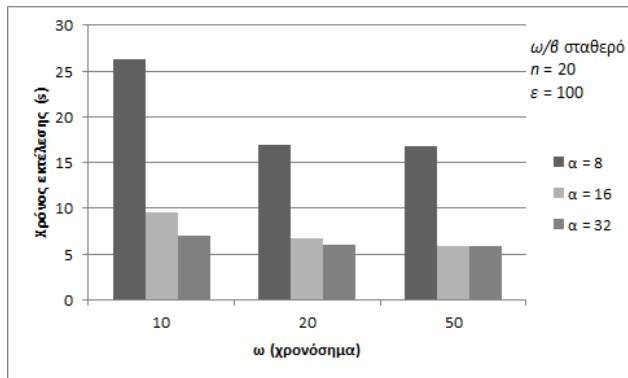
Στο σχήμα 4.3 παρατηρείται με ποιό τρόπο επηρεάζει το παράθυρο ω τον χρόνο εκτέλεσης ενός κύκλου σε συνδυασμό με την ακρίβεια α της πυξίδας και το ελάχιστο πλήθος αντικειμένων n .

Στα δύο διαγράμματα, η σχέση ω/β κρατείται σταθερή. Αυτό ισοδυναμεί με ίδιο μέγεθος συμβολικών αναπαραστάσεων για τα αντικείμενα ανεξαρτήτως του μεγέθους του παραθύρου ω . Όσο το παράθυρο αυξάνεται, ο χρόνος εκτέλεσης γενικά μειώνεται. Χρησιμοποιώντας μεγαλύτερο παράθυρο, ο αλγόριθμος “κοιτάζει” σε μεγαλύτερο κομμάτι της ιστορίας των αντικειμένων προκειμένου να εντοπίσει τροχιές που μπορούν να πολυπλεχθούν. Αυτό ισοδυναμεί με την εύρεση αντικειμένων που κινούνται μαζί για περισσότερο χρονικό διάστημα πράγμα που είναι σπανιότερο όσο αυξάνει το διάστημα αυτό. Με λιγότερα αντικείμενα που μπορούν να πολυπλεχθούν, μειώνεται και ο συνολικός χρόνος εκτέλεσης.

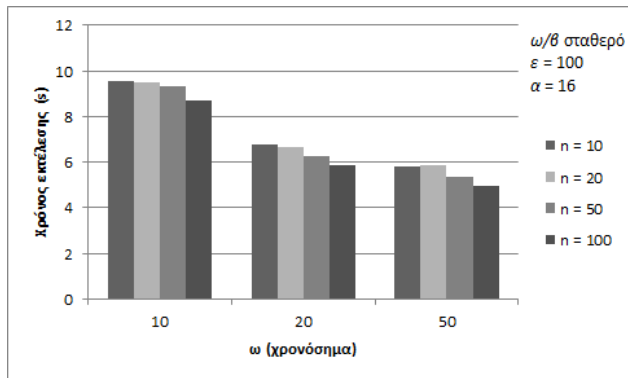
Το γεγονός αυτό αντικατοπτρίζεται και στην περίπτωση της μεταβολής της ακρίβειας α . Όπως φαίνεται στο σχήμα 4.3 για ακρίβεια $\alpha = 8$ οι χρόνοι εκτέλεσης είναι μεγαλύτεροι για κάθε τιμή του παραθύρου. Με μικρότερη ακρίβεια αναπαράστασης, προκύπτουν περισσότερα αντικείμενα που μπορούν να πολυπλεχθούν άρα και αυξημένος χρόνος επεξεργασίας.

Η παράμετρος n (ελάχιστο πλήθος αντικειμένων) φαίνεται να επηρεάζει τους χρόνους εκτέλεσης αλλά σε πολύ μικρό βαθμό. Αυτό οφείλεται στο γεγονός ότι τα αντικείμενα πρώτα τοποθετούνται σε συστάδες και μετέπειτα απορρίπτονται αυτές που δεν τηρούν το ελάχιστο πλήθος αντικειμένων. Άρα η παράμετρος n δεν περιορίζει τον αριθμό των συστάδων που πρόκειται να υπολογιστούν. Εντούτοις ελαττώνει τον αριθμό των ομάδων με κοινή συμβολική αναπαράσταση που πρόκειται να οδηγηθούν για συσταδοποίηση, μειώνοντας έτσι ελάχιστα τον συνολικό χρόνο εκτέλεσης.

Στο διάγραμμα 4.4α' φαίνεται πώς μεταβάλλεται ο χρόνος εκτέλεσης με την αύξηση του βήματος κύλισης β για σταθερό παράθυρο ω . Η αύξηση του β οδηγεί σε πολύ μικρή αύξηση του χρόνου εκτέλεσης. Αυξάνοντας το βήμα κύλισης, ο αλγόριθμος ανάγει την κατεύθυνση κάθε αντικειμένου σε πιο γενικά επίπεδα κάθε φορά, αφού υπολογίζονται λιγότερα σύμβολα αναπαράστασης της κίνησης για ίδιο χρονικό παράθυρο. Αυτό συνεπάγεται ελαφρώς περισσότερα αντικείμενα με κοινή συμβολική αναπαράσταση άρα και ελαφρώς αυξημένο χρόνο εκτέλεσης. Το φαινόμενο αυτό γίνεται πολύ εμφανές για τιμή του βήματος κύλισης ίση με 10. Εκεί, για παράθυρο $\omega = 20$, λαμβάνονται υπόψη μόνο δύο σύμβολα για την κίνηση κάθε αντικειμένου, άρα προκύπτουν πάρα πολλά αντικείμενα που έχουν “κοινή” κατεύθυνση.

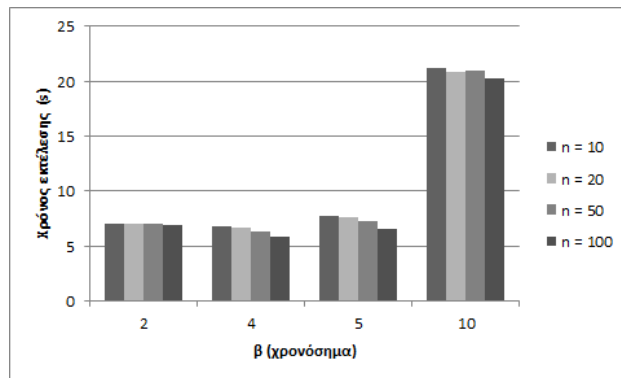


(α')

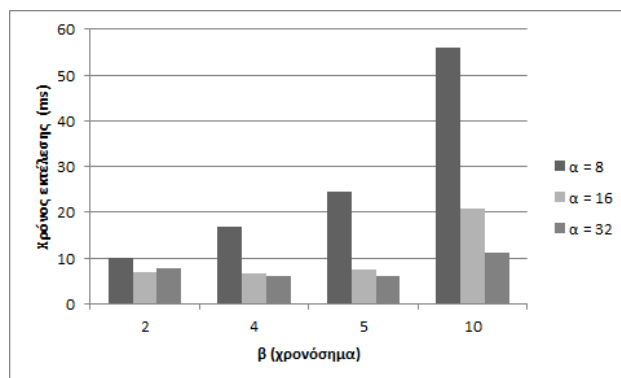


(β')

Σχήμα 4.3: Επίδραση μεγέθους παραθύρου



(α'')

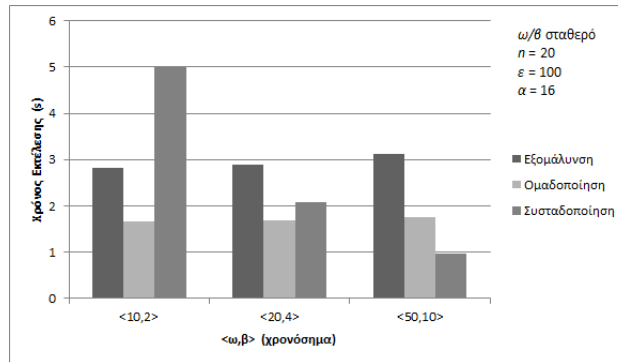


(β'')

Σχήμα 4.4: Επίδραση βήματος κύλισης

Το ίδιο θα περίμενε κανείς να συμβαίνει και με την μείωση της ακρίβειας α της πυξίδας. Πράγματι, όπως φαίνεται στο σχήμα 4.4β', για 8 ή 16 κατευθύνσεις στην πυξίδα αυτό είναι εμφανές. Εντούτοις, για 32 κατευθύνσεις και μικρό β , προκύπτει μεγαλύτερος χρόνος εκτέλεσης από μεγαλύτερες τιμές του βήματος κύλισης. Ο λόγος είναι ότι η μεγάλη ακρίβεια της πυξίδας, περιγράφει πολύ καλά την κίνηση των αντικειμένων, οπότε απαιτείται μικρό βήμα κύλισης προκειμένου να συλλαμβάνονται οι αλλαγές αυτές από τον αλγόριθμο και να προκύπτουν έτσι τα αντικείμενα σε ίδιες ομάδες του πίνακα κατακερματισμού.

Χρόνοι εκτέλεσης φάσεων



Σχήμα 4.5: Κόστος εκτέλεσης (ανά φάση)

Στο διάγραμμα 4.5 φαίνονται οι επί μέρους χρόνοι εκτέλεσης για κάθε ρουτίνα για διάφορα χαρακτηριστικά παράθυρα.

Από το διάγραμμα είναι εμφανές ότι ο χρόνος για τη διαδικασία της εξομάλυνσης επιβαρύνεται ελάχιστα καθώς αυξάνεται το παράθυρο. Αυτό οφείλεται στο γεγονός ότι με μεγαλύτερο παράθυρο, ο αλγόριθμος πρέπει να διαχειριστεί περισσότερα στίγματα ανά αντικείμενο ανά κύκλο εκτέλεσης.

Η διαδικασία της ομαδοποίησης (hashing) καταναλώνει ίδιο χρόνο άσχετα με το μέγεθος του παραθύρου. Ο χρόνος που καταναλώνει εξαρτάται από τον συνολικό αριθμό των αντικειμένων.

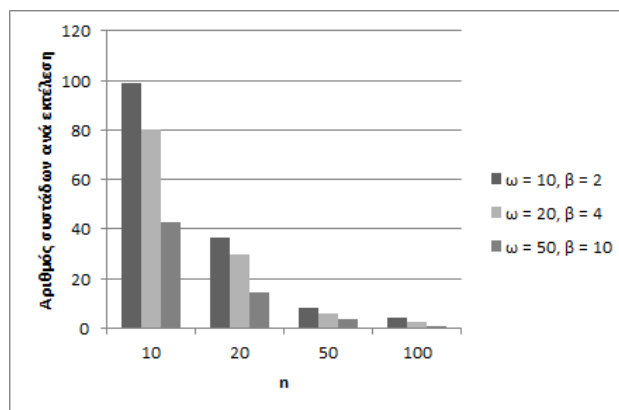
Το μεγαλύτερο κόστος για μικρή τιμή του παραθύρου φαίνεται να απαιτεί η διαδικασία της συσταδοποίησης. Αυτό πέφτει απότομα με την πρώτη αύξηση του παραθύρου. Με πολύ μικρό παράθυρο, πάρα πολλά αντικείμενα προκύπτουν να έχουν ίδια συμβολική αναπαράσταση (αφού εξετάζει πολύ πρόσφατη κίνηση των αντικειμένων) με αποτέλεσμα να χρειάζεται πολύ μεγάλος χρόνος εκτέλεσης. Η συσταδοποίηση είναι εν γένει χρονοβόρα διαδικασία και η αύξηση των αντικειμένων που πρόκειται να συσταδοποιηθούν αυξάνει εκθετικά τον χρόνο ανίχνευσης συστάδων, αφού απαιτεί υπολογισμό αποστάσεων ανάμεσα σε όλα τα μέλη κάθε υποψήφιας ομάδας.

Πλήθος πολυπλεγμένων τροχιών

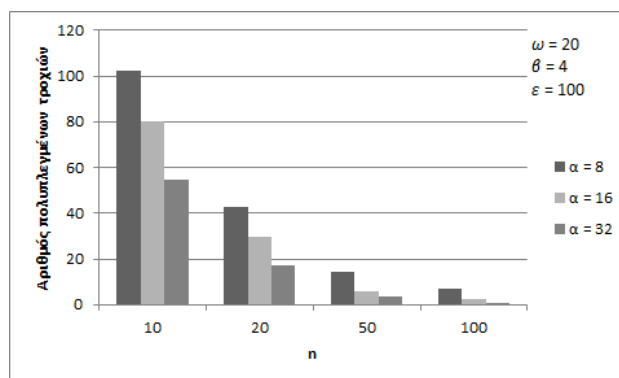
Στο διάγραμμα 4.6α' φαίνεται πώς το μέγεθος τους παραθύρου και το ελάχιστο πλήθος αντικειμένων επηρεάζει το σύνολο των συστάδων που εντοπίζονται.

Η αύξηση του ελάχιστου πλήθους αντικειμένων n προκειμένου μία συστάδα να θεωρείται σημαντική για να συμπεριληφθεί στην έξοδο του αλγορίθμου επηρεάζει όπως ήταν αναμενόμενο τον συνολικό αριθμό συστάδων. Η πτώση του αριθμού των συστάδων είναι εκθετική με την αύξηση του n .

Το ίδιο συμβαίνει και με την αύξηση μεγέθους του παραθύρου ω . Μεγαλύτερο παράθυρο οδηγεί σε λιγότερες συστάδες αντικειμένων. Για πολύ μεγάλες τιμές του παραθύρου σε συνδυασμό με μεγάλες τιμές του n , το σύνολο των συστάδων μειώνεται υπερβολικά. Το αποτέλεσμα είναι αναμενόμενο αφού το φαινόμενο της ομαδικής κίνησης πάρα πολλών αντικειμένων για πολύ μεγάλο χρονικό διάστημα μαζί σπανίζει, εκτός από τις περιπτώσεις κίνησης πομπών αντικειμένων (convoys).



(α')

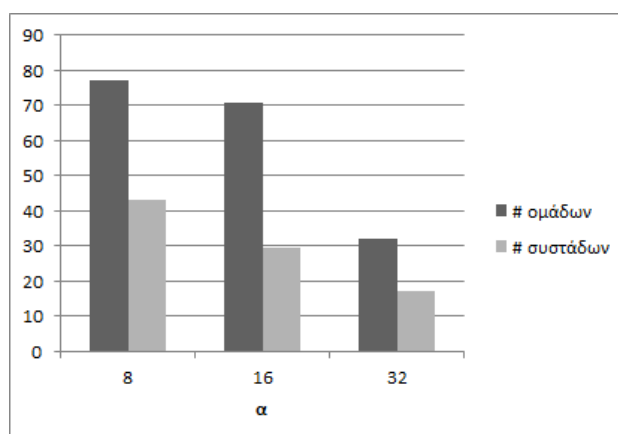


(β')

Σχήμα 4.6: Πλήθος πολυπλεγμένων τροχιών

Με τον ίδιο τρόπο επηρεάζει και η ακρίβεια της πυξίδας α (Σχήμα 4.6β'). Όπως φαίνεται, ακριβέστερη αναπαράσταση της κίνησης των αντικειμένων επιφέρει ανίχνευση λιγότερων συστάδων. Η λεπτομερέστερη περιγραφή της κίνησης οδηγεί αντικείμενα που προηγουμένως ανήκαν σε μία συστάδα, να διαχωρίζονται τώρα ενδεχομένως σε πολλές. Αν η αρχική συστάδα δεν συμπεριελάμβανε αρκετά μεγάλο πλήθος αντικειμένων, τότε οι καινούριες δεν θα τηρούν τον περιορισμό του ελάχιστου πλήθους αντικειμένων και δεν θα παρουσιάζονται στην έξοδο του αλγορίθμου μειώνοντας έτσι τις “πολυπλεγμένες” τροχιές.

Η επιρροή της ακρίβειας α παρουσιάζεται καλύτερα στο διάγραμμα 4.7. Η πρώτη στήλη συμβολίζει τον αριθμό των ομάδων αντικειμένων που είχαν ίδιες συμβολικές αναπαραστάσεις και αριθμούσαν περισσότερα αντικείμενα από n αντικείμενα. Η δεύτερη στήλη δείχνει πόσες συστάδες προέκυψαν (μέσος όρος) ανά εκτέλεση, δηλαδή το πλήθος των πολυπλεγμένων τροχιών που υπολογίστηκαν. Η αύξηση της ακρίβειας μειώνει το πλήθος των πολυπλεγμένων τροχιών που ανακαλύπτονται.

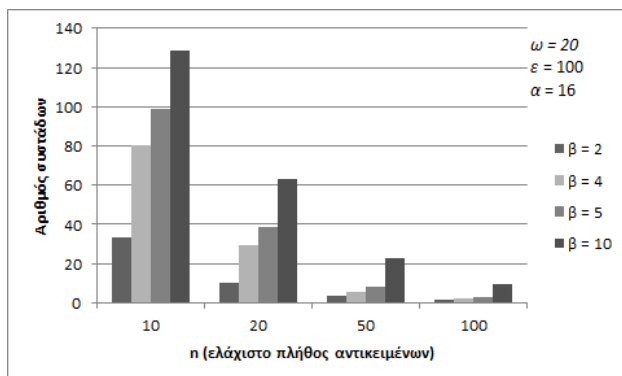


Σχήμα 4.7: Επίδραση ακρίβειας σε αριθμό πολυπλεγμένων τροχιών

Δραστικά επηρεάζει και η τιμή του βήματος κύλισης β σε σχέση με το παράθυρο που χρησιμοποιείται (Σχήμα 4.8). Μεγαλώνοντας την τιμή της παραμέτρου β αυξάνεται και ο αριθμός των συστάδων που εντοπίστηκαν. Αυτό οφείλεται στην γενικότερη περιγραφή της κίνησης των αντικειμένων που οδηγεί σε περισσότερες ομάδες αντικειμένων με ίδια συμβολική αναπαράσταση άρα και στον εντοπισμό περισσότερων συστάδων.

Αξίζει να παρατηρηθεί ότι δύο αντικείμενα με χωρική εγγύτητα είναι πιθανότερο να προκύψουν σε κάποια συστάδα χρησιμοποιώντας αδρύτερη περιγραφή της κίνησής τους παρά ακριβέστερη (α μεγάλο). Η παραμικρή διαφορά στην ιστορία κίνησής τους (που επηρεάζεται άμεσα από την διαδικασία εξομάλυνσης) θα τα αποτρέψει καταλυτικά από το να τοποθετηθούν σε κοινή συστάδα, ασχέτως αν βρίσκονταν συνεχώς κοντά το ένα με το άλλο κατά την

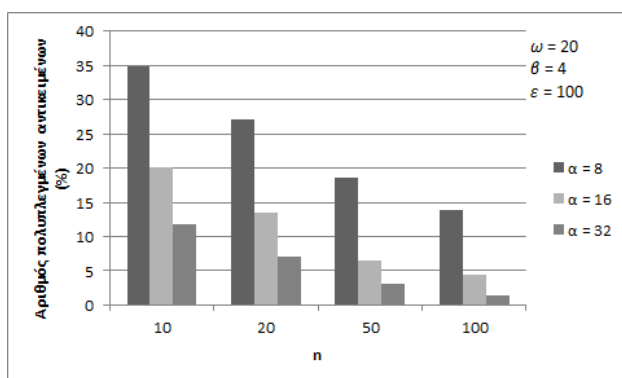
διάρκεια της κίνησής τους.



Σχήμα 4.8: Μέγεθος πολυπλεγμένων τροχιών

Βαθμός πολυπλεξίας

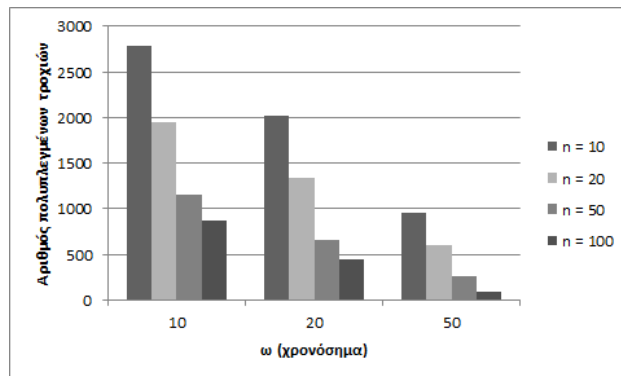
Ο αριθμός των πολυπλεγμένων αντικειμένων εξαρτάται καταρχήν από το ελάχιστο πλήθος αντικειμένων. Όσες συστάδες βρέθηκαν να έχουν μικρότερο πλήθος από αυτό της παραμέτρου n απορρίπτονται, με αποτέλεσμα να μην πολυπλέκονται οι τροχιές των αντικειμένων τα οποία περιέχουν. Αυτομάτως μειώνεται το ποσοστό των αντικειμένων που πολυπλέχθηκαν (Σχήμα 4.9). Είναι φανερό ότι ανάλογη πτώση στον αριθμό των πολυπλεγμένων αντικειμένων προκύπτει και από την αύξηση της ακρίβειας αναπαράστασης της κίνησης. Περισσότερη ποικιλομορφία στην κίνηση συνεπάγεται λιγότερους συνδυασμούς αντικειμένων με ίδια -ακριβώς- πορεία άρα και μικρότερο βαθμό πολυπλεξίας.



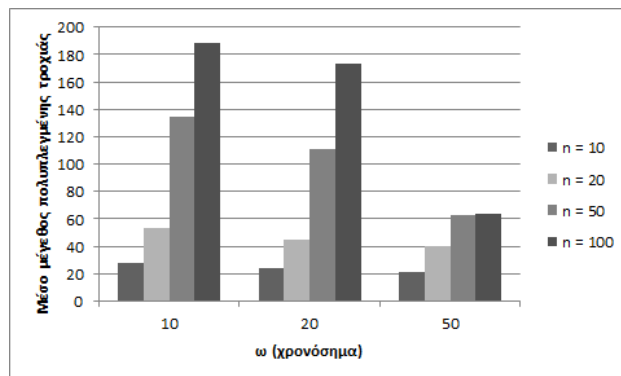
Σχήμα 4.9: Βαθμός πολυπλεξίας

Ενδιαφέρον παρουσιάζει η επίδραση του χρονικού παραθύρου στον συνολικό αριθμό των τροχιών που πολυπλέκονται. Όπως φαίνεται στο σχήμα 4.10 καθώς αυξάνεται το παράθυρο μειώνεται ο αριθμός των αντικειμένων που πολυπλέκονται. Αυτό επιβεβαιώνει την λογική διαίσθηση ότι όσο αυξάνεται το

χρονικό εύρος όπου απαιτείται για χωρική εγγύτητα των αντικειμένων τόσο λιγότερα βρίσκονται να κινούνται μαζί και συνεπώς να πολυπλέκονται.

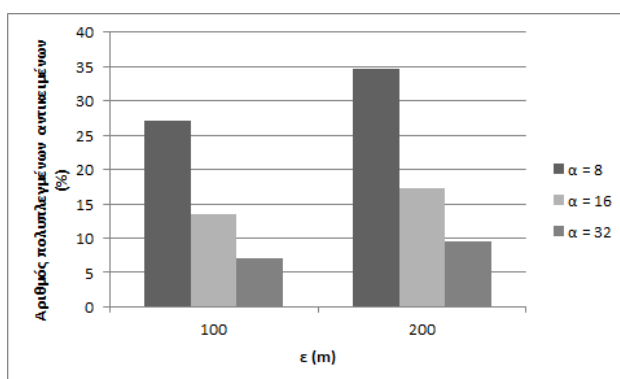


Σχήμα 4.10: Επίδραση παραθύρου στον βαθμό πολυπλεξίας



Σχήμα 4.11: comment out

Στο διάγραμμα 4.12 φαίνεται πώς αλλάζει το ποσοστό των αντικειμένων που πολυπλέκονται μεταβάλλοντας την ελάχιστη απόσταση ϵ μεταξύ των αντικειμένων. Όπως ήταν αναμενόμενο, χαλαρώνοντας το κριτήριο της απόστασης επιτρέπεται σε περισσότερα αντικείμενα να πολυπλεχθούν. Αυτό οδηγεί σε πιο γενικευμένες εκτιμήσεις για την κίνηση των αντικειμένων.



Σχήμα 4.12

4.2.3 Σφάλμα προσέγγισης πολυπλεγμένης τροχιάς

Ο αλγόριθμος σε κάθε κύκλο εκτέλεσης υπολογίζει μία πολυπλεγμένη τροχιά για όσα αντικείμενα εντοπίσει ότι είχαν κοινή πορεία στο χρονικό παράθυρο ω . Η πολυπλεγμένη τροχιά πρέπει να προσεγγίζει τις πραγματικές με τον καλύτερο δυνατό τρόπο. Παρακάτω παρουσιάζονται πίνακες τιμών που εμφανίζουν την μέση απόκλιση των πραγματικών τροχιών από την αντίστοιχη πολυπλεγμένη τους. Η απόκλιση μετρήθηκε στα σημεία κατασκευής της πολυπλεγμένης τροχιάς και τις αντίστοιχες θέσεις των αντικειμένων.

Στον πρώτο πίνακα παρουσιάζονται οι αποκλίσεις για εκτέλεση του αλγορίθμου θέτοντας την ελάχιστη απόσταση αντικειμένων ίση με $\epsilon = 100m$. Για μικρή ακρίβεια της πυξίδας η απόσταση αυτή δεν φαίνεται να τηρείται. Ο λόγος εμφανίζεται στην μικρή ακρίβεια κατά την προσέγγιση της κίνησης όπου όσο μεγαλύτερη είναι η κίνηση αυτή (μέγεθος παραθύρου) τόσο περισσότερο η πολυπλεγμένη τροχιά αποκλίνει από τις πραγματικές. Αυξάνοντας την ακρίβεια της πυξίδας η προσέγγιση γίνεται όλο και καλύτερη με αποτέλεσμα η απόκλιση να μειώνεται αρκετά. Εντούτοις όλες οι αποκλίσεις ξεπερνούν την ελάχιστη απόσταση των 100m και ο λόγος εντοπίζεται στη χρήση του αλγορίθμου DBScan για την εκτέλεση της συσταδοποίησης. Ο DBScan εντοπίζει αντικείμενα που έχουν ανά δύο ελάχιστη απόσταση ϵ και όχι κατ' ανάγκη όλα μεταξύ τους. Έτσι οι συστάδες που εντοπίζονται ενδεχομένως δεν συμβαδίζουν με την ελάχιστη απόσταση που τίθεται από τον χρήστη.

Στο δεύτερο πίνακα όπου η ελάχιστη απόσταση τίθεται ίση με $\epsilon = 200m$ τα αποτελέσματα είναι πιο ακριβή. Και πάλι η απόκλιση για μικρές τιμές της ακρίβειας της πυξίδας είναι μεγάλη αλλά στη γενική περίπτωση αυτή εμπίπτει στο όριο των 200m που καθορίστηκε από την παράμετρο ϵ . Ο αλγόριθμος καταφέρνει επιτυχώς να εντοπίσει αντικείμενα που έχουν κοινή πορεία και η πολυπλεγμένη τροχιά που υπολογίζει προσεγγίζει τις αντίστοιχες πραγματικές με σφάλμα μικρότερο των διακοσίων μέτρων όπως καθορίστηκε κατά την εκτέλεσή του.

Ακρίβεια	$\omega = 10$	$\omega = 20$	$\omega = 50$
8	131m	168m	305m
16	104m	122m	192m
32	100m	116m	153m

(α') $\epsilon = 100m$

Ακρίβεια	$\omega = 10$	$\omega = 20$	$\omega = 50$
8	166m	194m	339m
16	124m	139m	215m
32	117m	131m	170m

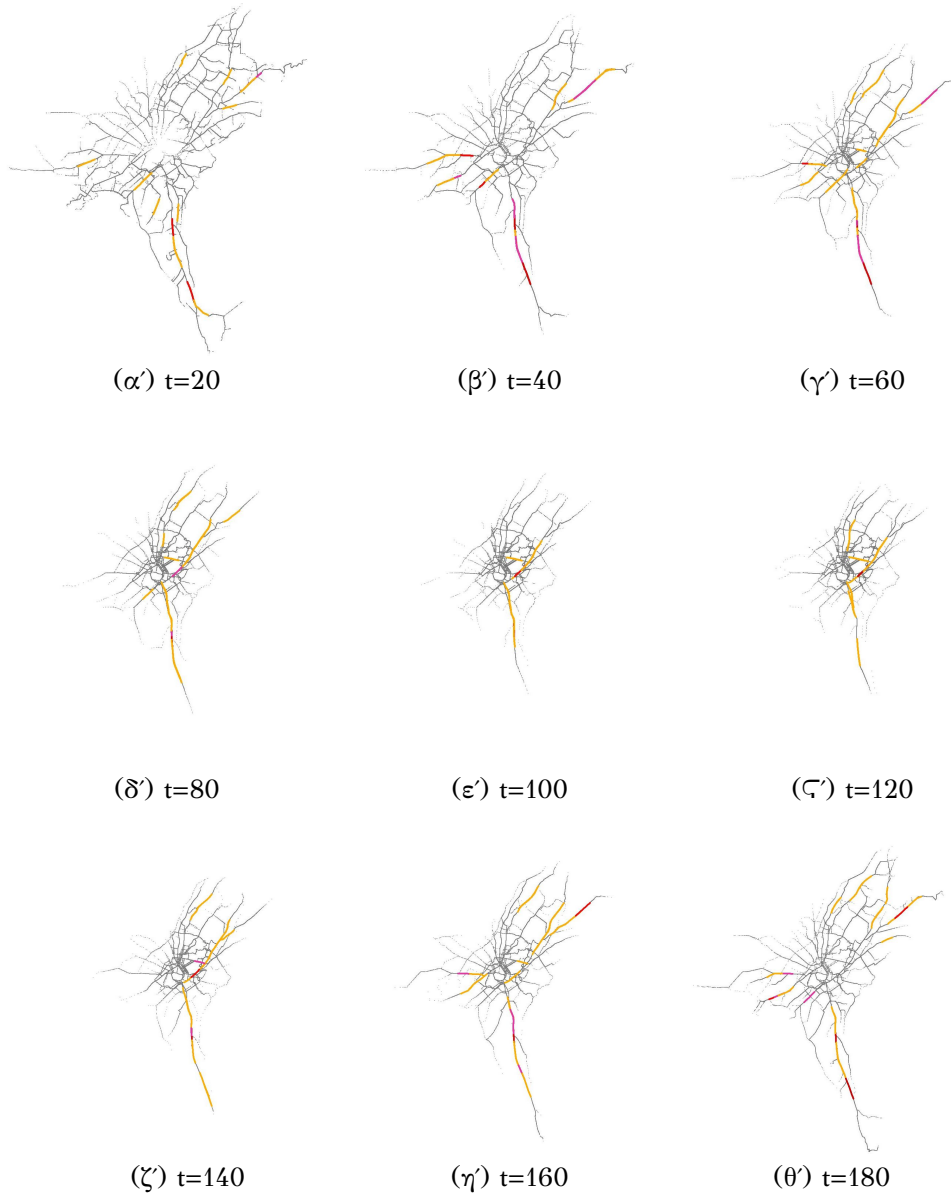
(β') $\epsilon = 200m$

Σχήμα 4.13: Μέση απόκλιση πραγματικών τροχιών από την αντίστοιχη πολυπλεγμένη τους.

4.2.4 Οπτικοποιημένα αποτελέσματα

Παρακάτω παρουσιάζονται οπτικοποιημένα αποτελέσματα από μία εκτέλεση του αλγορίθμου με τα προαναφερθέντα πειραματικά δεδομένα, με χρήση παραθύρου $\omega = 20$, βήματος $\beta = 4$ και ακρίβεια $\alpha = 16$. Τα αποτελέσματα παρουσιάζονται σε 9 στιγμιότυπα ανά 20 χρονικές στιγμές το καθένα. Οι γκριζες κουκκίδες αντιστοιχούν σε σημειακές θέσεις αντικειμένων για τη δεδομένη χρονική στιγμή. Οι χρωματιστές γραμμές δείχνουν υπολογισμένες πολυπλεγμένες τροχιές ενώ η ένταση του χρώματος αντικατοπτρίζει την ποσότητα των αντικειμένων που περιγράφει η συγκεκριμένη τροχιά (εντονότερο χρώμα σηματοδοτεί περισσότερα αντικείμενα με κοινή τροχιά). Να σημειωθεί πως στις περιπτώσεις που παρουσιάζονται πολύ μακρόσυρτες τροχιές, αυτές στην ουσία αποτελούν τμήματα τροχιών με ίδιο χρώμα που λόγω επικάλυψης εμφανίζονται ως μία.

Είναι εμφανές ότι η εφαρμογή καταφέρει να συλλάβει την ακτινική κίνηση των αντικειμένων προς το κέντρο της πόλης εντοπίζοντας ομάδες αντικειμένων με κοινή πορεία κυρίως στους μεγάλους αυτοκινητόδρομους όπως ήταν αναμενόμενο.



Σχήμα 4.14: Στιγμιότυπα πολυπλεγμένων τροχιών

Κεφάλαιο 5

Συμπεράσματα - Προοπτικές

5.1 Συμπεράσματα

Σκοπός της διπλωματικής ήταν η υλοποίηση ενός online αλγορίθμου για την εύρεση αντικειμένων με κοινές τροχιές και η κατασκευή μιας συνθετικής τροχιάς, ανυπροσωπευτικής για όσα αντικείμενα πολυπλέχθηκαν. Με βάση την λύση που προτάθηκε και τις πειραματικές μετρήσεις που έγιναν, προκύπτουν τα εξής:

- Ο συνολικός χρόνος ενός κύκλου εκτέλεσης του αλγορίθμου είναι της τάξεως των μερικών δευτερολέπτων, γεγονός που τον καθιστά ικανό να χρησιμοποιηθεί σε εφαρμογές πραγματικού χρόνου με επιτυχία.
- Η επιλογή του DBSCAN για την συσταδοποίηση επέξέτεινε την έννοια της χωρικής εγγύτητας εντοπίζοντας αλυσίδες αντικειμένων με αποτέλεσμα ο αλγόριθμος να εντοπίζει επιτυχώς πομπές αντικειμένων με τη χρήση κατάλληλων παραμέτρων.

5.2 Πρακτικές εφαρμογές

Ο αλγόριθμος της online πολυπλεξίας δίνει τη δυνατότητα σε κάποιο σύστημα να εντοπίζει και να καταγράφει σε ζωντανό χρόνο, μαζικές κινήσεις αντικειμένων είτε με την ίδια τροχιά είτε που κινούνται προς την ίδια κατεύθυνση. Παραδείγματος χάρη, θέτοντας κατάλληλες τιμές στις παραμέτρους μπορούν να εντοπιστούν αντικείμενα που κινούνται βόρεια την τελευταία ώρα (Μεγάλο παράθυρο ω , μικρό ελάχιστο πλήθος αντικειμένων n , μεγάλο βήμα κύλισης β). Μεταβάλλοντας τις παραμέτρους κατάλληλα, μπορούν να εντοπιστούν μεγάλες ομάδες αντικειμένων που κινούνται συλλογικά ανεξαρτήτως της κατεύθυνσης τους για το πρόσφατο παρελθόν (Μικρό παράθυρο ω , μεγάλο ελάχιστο πλήθος αντικειμένων n).

Η επιλογή των παραμέτρων κάθε φορά εξαρτάται από τα επιθυμητά αποτελέσματα. Η δυνατότητα που παρέχει ο αλγόριθμος για συνδιασμό μεγάλου εύρους τιμών των παραμέτρων τον καθιστά χρήσιμο σε διάφορες εφαρμογές. Ενδεικτικά παρουσιάζονται μερικές πιο κάτω.

5.2.1 Βελτίωση απόδοσης τοπολογικών ερωτημάτων

Η δυνατότητα πολυπλεξίας των τροχιών που παρέχει ο αλγόριθμος αυτή καθ' αυτή, τον καθιστά χρήσιμο σε συστήματα απάντησης ερωτημάτων αναφορικά με τροχιές. Θεωρώντας πως το σφάλμα που υπεισέρχεται από την πολυπλεγμένη τροχιά είναι ανεκτό, τα αποτελέσματα της πολυπλεξίας μπορούν να χρησιμοποιηθούν ενδεχομένως για να μειώσουν τους χρόνους απάντησης τοπολογικών ερωτημάτων. Παραδείγματος χάρη, σε ένα ερώτημα εισόδου σε μια περιοχή, η ύπαρξη μιας ήδη υπολογισμένης πολυπλεγμένης τροχιάς δίνει ως απάντηση όλα τα αντικείμενα τα οποία περιγράφει, χωρίς να υπάρχει η ανάγκη να ελεγχθεί η τροχιά κάθε αντικειμένου εξ αυτών ξεχωριστά. Παρομοίως μπορούν να επιταχυνθούν ερωτήματα περιοχής, μέσης ταχύτητας και διανυόμενης απόστασης αν κάποια από τα αντικείμενα που πρόκειται να συμπεριληφθούν έχουν ήδη πολυπλεχθεί.

5.2.2 Εντοπισμός πομπής/φάλαγγας

Ο αλγόριθμος, λόγω της κατασκευής του, εμφανίζει την ικανότητα να εντοπίζει πομπές αντικειμένων. Λόγω της χρήσης του DBSCAN, γίνεται εφικτός ο εντοπισμός αλυσίδων αντικειμένων που είχαν ίδια πορεία για συγκεκριμένο χρονικό παράθυρο εντοπίζοντας έτσι αποτελεσματικά πομπές που μπορεί να δημιουργούν διάφορα αντικείμενα. Η δυνατότητα δεν περιορίζεται μόνο σε μακρόσυρτες αλυσίδες (φάλαγγες) αλλά επεκτείνεται και σε μέτωπα αντικειμένων που κινούνται επαρκώς κοντά και προς την ίδια κατεύθυνση. Παραδείγματος χάρη, ομάδα πλοίων που κινείται βρισκόμενα το ένα δίπλα στο άλλο είναι εφικτό να εντοπιστεί με τη χρήση του προτεινόμενου αλγόριθμου.

5.2.3 Βελτιστοποίηση Vehicular Ad-Hoc δικτύων

Στο επιστημονικό πεδίο των ασύρματων δικτύων γίνεται έρευνα τα τελευταία χρόνια για τα Ad-Hoc δίκτυα. Στα δίκτυα αυτά όπου κόμβοι θεωρούνται κινούμενα αντικείμενα (λ.χ αυτοκίνητα) δημιουργούνται συνεχώς συνδέσεις μεταξύ γειτονικών κόμβων προκειμένου να δημιουργηθεί δυναμικά ένα δίκτυο. Έστω ένας κόμβος αναφοράς k . Όταν ο κόμβος (αυτοκίνητο) στο οποίο βρίσκεται συνδεδεμένος ο κόμβος αναφοράς απομακρυνθεί, ο κόμβος k αναγκάζεται να φάξει για άλλους κόμβους στην εμβέλειά του και να ενωθεί σε κάποιον από αυτούς. Σε ένα δίκτυο αποτελούμενο από κινούμενα αυτοκίνητα, τέτοιες

αλλαγές γίνονται πολύ συχνά λόγω της αυθαίρετης και συνεχούς κίνησης των αντικειμένων.

Χρησιμοποιώντας τα αποτελέσματα της πολυπλεξίας, είναι δυνατό να μειωθούν οι εναλλαγές κόμβων μεταξύ των κινούμενων αντικειμένων μειώνοντας την πιθανότητα απομάκρυνσης του κόμβου στον οποίο είναι ενωμένος κάποιος κόμβος. Εντοπίζοντας αντικείμενα τα οποία κινούνται μαζί του ή προς τα η ίδια κατεύθυνση για κάποιο χρονικό διάστημα, ο κόμβος k μπορεί να προτιμήσει κάποιο από αυτά (έναντι άλλων) ως κόμβο σύζευξης αφού η πιθανότητα να συνεχίσουν να κινούνται δίπλα-δίπλα είναι αυξημένη σε σχέση με κάποιον άλλο αυθαίρετο κόμβο. Ακολουθώντας αυτή τη στρατηγική όλα τα αντικείμενα του δικτύου, μειώνονται οι εναλλαγές κόμβων με αποτέλεσμα την αύξηση της ενεργειακής απόδοσης του υλικού (λ.χ. κινητά τηλέφωνα) και την μείωση του τηλεπικοινωνιακού φόρτου.

5.2.4 Εντοπισμός μαζικής κίνησης/διαφυγής ζώων

Η χρήση συστημάτων παρακολούθησης ζώων δίνει την δυνατότητα μελέτης των βιολογικών τους συνηθειών όπως τρόπους συγκέντρωσης και μεταναστευτικές συνήθειες. Η παρακολούθηση των ζώων μιας περιοχής σε ζωντανό χρόνο, σε συνδιασμό με τον αλγόριθμο πολυπλεξίας, θα μπορούσε να προσφέρει άμεση ενημέρωση για ομαδική κίνηση των ζώων προς κάποια κατεύθυνση επισημαίνοντας έτσι πιθανούς κινδύνους όπως μια πυρκαγιά. Επιπλέον, σε περίπτωση μιας καταστροφής σαν αυτή, ο εντοπισμός της κατεύθυνσης των ζώων θα μπορούσε να βοηθήσει στην έγκαιρη διάσωσή τους αν από τα αποτελέσματα προκύψει πως η πορεία τους πρόκειται να τα οδηγήσει σε παγίδευση.

5.3 Μελλοντική εργασία

Ο αλγόριθμος που έχει υλοποιηθεί εξάγει ικανοποιητικά αποτελέσματα και σε ανεκτούς χρόνους για να χρησιμοποιηθεί σε online εφαρμογές. Εντούτοις, βελτιώσεις μπορούν να πραγματοποιηθούν προκειμένου να μειωθεί ακόμα περισσότερο ο χρόνος εκτέλεσης του αλγορίθμου και να ελαττωθούν οι περιπτώσεις πολυπλεξίας με μεγάλη απόκλιση. Επιπλέον, με περαιτέρω αύξηση της λειτουργικότητας ο αλγόριθμος θα μπορούσε να χρησιμοποιηθεί πιο στοχευμένα σε συγκεκριμένες εφαρμογές.

5.3.1 Μείωση χρόνου εκτέλεσης

Για την μείωση του χρόνου εκτέλεσης προκύπτει από τη μελέτη των πειραματικών δεδομένων πως βελτιώσεις μπορούν να γίνουν στο βήμα της ομαδοποίησης και της συσταδοποίησης. Συγκεκριμένα μπορεί να γίνει προσπάθεια για:

- Αξιοποίηση ενδιάμεσων αποτελεσμάτων λόγω παραθύρου για επιτάχυνση της διαδικασίας ομαδοποίησης.
- Βελτιστοποίηση κατά την ανίχνευση συστάδων με χρήση αποδοτικότερου αλγορίθμου ή δομής.

Αξιοποίηση ενδιάμεσων συμβολικών αναπαραστάσεων

Μέχρι στιγμής, ο αλγόριθμος σε κάθε κύκλο εκτέλεσης ομαδοποιεί όλα τα αντικείμενα από την αρχή χρησιμοποιώντας ολόκληρη την συμβολική τους αναπαράσταση ως κλειδί. Η αναπαράσταση αυτή, δεδομένου ότι συλλαμβάνει την ιστορία κίνησης του κάθε αντικειμένου, μετά από κάθε ενημέρωσή της είναι σε πολύ μεγάλο βαθμό ίδια με την προηγούμενη (ανάλογα με το μέγεθος ω/β). Με αυτή την παρατήρηση γίνεται εμφανές ότι μια αποδοτικότερη δομή θα μπορούσε να χρησιμοποιηθεί όπου θα αξιοποιεί τα αποτελέσματα της ομαδοποίησης σε προηγούμενο κύκλο εκτέλεσης προκειμένου να εκτελέσει την τρέχον ομαδοποίηση σε πολύ λιγότερο χρόνο από το να την εκτελούσε όλη από την αρχή.

Βελτιστοποίηση συσταδοποίησης

Ως πρώτο μέσο βελτίωσης του χρόνου που απαιτείται για την εκτέλεση της διαδικασίας συσταδοποίησης προτείνεται η χρήση αλγορίθμου παράλληλης επεξεργασίας για αύξηση της απόδοσης. Δεδομένου ότι ο αλγόριθμος συσταδοποίησης εκτελεί στην ουσία μεγάλο όγκο απλών πράξεων για τον υπολογισμό των αποστάσεων μεταξύ των σημείων, η χρήση τεχνολογίας παραλληλοποίησης με χρήση αρχιτεκτονικών πολυνηματικών GPU (λ.χ. τεχνολογία CUDA) κρίνεται πως θα μειώσει τους χρόνους εκτέλεσης σε πολύ μεγάλο βαθμό.

Μια δεύτερη πρόταση για βελτίωση του χρόνου εκτέλεσης προτείνεται η χρήση δομής πλέγματος (grid) προκειμένου να εντοπίζονται συστάδες αντικειμένων. Τα στίγματα των αντικειμένων, καθώς καταφθάνουν στο σύστημα τοποθετούνται σε μία δομή πλέγματος όπου ανάλογα με τις συντεταγμένες του σημείου αυτό τοποθετείται στο κατάλληλο κελί. Στο τέλος, η εύρεση συστάδων με ικανοποιητικό αριθμό αντικειμένων (δηλαδή στιγμάτων) ισοδυναμεί με την καταγραφή των κελιών που έχουν περιεχόμενο περισσότερο του ελάχιστου πλήθους που απαιτείται. Η απόδοση μίας τέτοιας δομής είναι πολύ καλύτερη από την αντίστοιχη της συσταδοποίησης δεδομένου ότι δεν γίνονται καθόλου συγκρίσεις μεταξύ των αντικειμένων. Εντούτοις, θέματα όπως ο έλεγχος για συστάδες στα σύνορα κελιών και η αποδοτική δεικτιοδότηση του πλέγματος παραμένουν και θα πρέπει να διερευνηθούν σε περίπτωση υλοποίησης μίας τέτοιας δομής.

5.3.2 Ποιότητα αποτελεσμάτων

Η ακρίβεια των αποτελεσμάτων μπορεί να βελτιωθεί περαιτέρω, μελετώντας τη συνάρτηση εξομάλυνσης και τη συνάρτηση υπολογισμού της πολυπλεγμένης τροχιάς. Επίσης, το βήμα συσταδοποίησης, θα μπορούσε να ενισχυθεί προκειμένου να μειώσει τις περιπτώσεις ομάδων που δεν βρίσκονται επαρκώς κοντά σε παρελθοντικούς χρόνους και γιαυτό δίνουν αποτελέσματα με μεγάλη ανακρίβεια. Δεδομένου ότι βελτιώνεται ο χρόνος συσταδοποίησης, μία πρόταση είναι να γίνεται συσταδοποίηση όχι μόνο χρησιμοποιώντας την πιο πρόσφατη θέση των αντικειμένων, αλλά να γίνεται και μια δεύτερη συσταδοποίηση χρησιμοποιώντας το αρχαιότερο στίγμα τους σε σχέση με το χρονικό παράθυρο λειτουργίας ω . Από την τομή των δύο συσταδοποιήσεων, τα αντικείμενα που προκύπτουν θα έχουν κοινή τροχιά. Με αυτό τον τρόπο, θα απορρίπτονται αντικείμενα που σε παρελθοντικό χρόνο είχαν μεγάλη απόκλιση από την ομάδα αλλά προέκυψε να βρίσκονται μαζί με τα υπόλοιπα αντικείμενα την τρέχον χρονική στιγμή.

5.3.3 Επέκταση λειτουργικότητας

Προκειμένου να ενισχυθεί η χρησιμότητα του αλγορίθμου θα μπορούσε να προστεθεί η λειτουργία της στοχευμένης μελέτης κίνησης αντικειμένων σε συγκεκριμένη κατεύθυνση/τομέα. Ο χρήστης, θα μπορεί να ορίζει μια συγκεκριμένη κατεύθυνση ή τομέα προς τον οποίο τον ενδιαφέρει η ομαδική κίνηση αντικειμένων. Ο αλγόριθμος, θα ψάχνει για αντικείμενα που κατευθύνονται μόνο προς αυτή την κατεύθυνση ή σύνολο κατευθύνσεων και οποιαδήποτε αντικείμενα με διαφορετική πορεία θα τα απορρίπτει. Ο χρόνος εκτέλεσης θα μειωθεί ενώ ταυτόχρονα ο χρήστης θα φιλτράρει τα αποτελέσματα βάση των φαινομένων κίνησης που θέλει να μελετήσει.

Παράρτημα Α΄

Τεκμηρίωση κώδικα

Α΄.1 Δομικά στοιχεία

Σε αυτή την παράγραφο παρουσιάζονται περιληπτικά τα δομικά στοιχεία της εφαρμογής για να είναι γίνει πιο εύκολα αντιληπτός ο τρόπος λειτουργίας της.

Α΄.1.1 LocationStamp

Το *LocationStamp* είναι μια δομή που ενθυλακώνει ένα στίγμα ενός αντικειμένου μαζί με τη στιγμή που αυτό καταγράφηκε (timestamped location). Αποθηκεύει το χρονόσημο, το *id* του αντικειμένου και τις συντεταγμένες του x, y για το χρονόσημο αυτό.

Α΄.1.2 MovingObject

Η κλάση αυτή περιγράφει ένα κινούμενο αντικείμενο και ό,τι πληροφορία εμπεριέχεται σε αυτό. Συμπεριλαμβάνει:

- το *id* του αντικειμένου
- τα στίγματα του για τα τελευταία $[\omega/\beta]$ χρονόσημα ως μια λίστα από *LocationStamp*
- την συμβολική αναπαράσταση της κίνησης του (δομή *TrajectoryHistory*)

Α΄.1.3 TrajectoryHistory

Η δομή *TrajectoryHistory* ενθυλακώνει την έννοια της συμβολικής αναπαράστασης. Αποθηκεύει κωδικοποιημένα των ιστορία της κίνησης του αντικειμένου στο οποίο αναφέρεται, βάση της πυξίδας κατευθύνσεων που παρουν-

σιάστηκε σε προηγούμενες παραγράφους. Περιέχει συναρτήσεις για κωδικοποίηση/αποδικοποίηση, ενημέρωση και τεμαχισμό της ιστορίας κίνησης βάση συγκεκριμένου εύρους χρονοσήμων.

A'.1.4 Buffer

Η δομή *Buffer* αποτελεί στην ουσία μία ουρά (queue) και χρησιμοποιείται για την είσοδο δεδομένων από τη μία πλευρά της και εξαγωγή από την άλλη. Τα δεδομένα παραμένουν εκεί μετά την εισαγωγή τους έως ότου εξαχθούν ένα ένα ή εκτελεστεί εντολή αδειάσματος της ουράς.

A'.2 Coordinator

Ο *Coordinator* αποτελεί το κύριο συστατικό της εφαρμογής. Είναι υπεύθυνος για την διαμεταγωγή των δεδομένων ανάμεσα στα διάφορα συστατικά της εφαρμογής. Ως είσοδο παίρνει μία ακολουθία στιγμάτων (*LocationStamp*) η οποία θεωρείται ότι καταφθάνει στην εφαρμογή διαμέσου ενός συστήματος διαχείρισης κινούμενων αντικειμένων. Το σύστημα αυτό τοποθετεί τα στιγμάτα σε ένα *buffer* και στη συνέχεια ο *coordinator* τα εξάγει ασύγχρονα προκειμένου να τροφοδοτήσει την υπόλοιπη εφαρμογή. Για τους σκοπούς των πειραματικών μετρήσεων, το σύστημα αυτό προσομοιώθηκε με τη χρήση ενός αρχείου το οποίο περιείχε τα στιγμάτα των αντικειμένων. Την ευθύνη για την ανάγνωση του αρχείου και για μετατροπή του σε ακολουθία στιγμάτων είχε το συστατικό *DataInputService*.

Ο *Coordinator* εμπεριέχει μία δομή τύπου *ObjectSet* στην οποία διατηρεί όλη τη πληροφορία που διαχειρίζεται. Για την εκτέλεση της πολυπλεξίας καλεί τη συνάρτηση *Multiplexer* που χρησιμοποιώντας ως είσοδο το *ObjectSet* βρίσκει και υπολογίζει πολυπλεγμένες τροχιές.

A'.3 ObjectSet

Η δομή *ObjectSet* ενθυλακώνει ένα σύνολο από αντικείμενα (*MovingObject's*) και ένα σύνολο από *Buffers*. Το σύνολο των αντικειμένων είναι αυτό που 'παρακολουθεί' ο αλγόριθμος και καθ'ένα έρχεται με την συμβολική του αναπαράσταση και την ιστορία των πρόσφατων θέσεων του όπως προαναφέρθηκε (*LocationHistory*).

Τα *Buffers*, χρησιμοποιούνται για την αποθήκευση των στιγμάτων που φθάνουν κατά τη διάρκεια εκτέλεσης ενός κύκλου, ώστε να ληφθούν υπόψη κατά την επόμενη εκτέλεση. Σε κάθε νέο κύκλο, ο αλγόριθμος υπολογίζει μία κατεύθυνση για κάθε αντικείμενο βάση των στιγμάτων που βρίσκονται στο *buffer* (διαδικασία εξομάλυνσης) και μετέπειτα το αδειάζει.

A'.4 Multiplexer

Η ρουτίνα αυτή αντιπροσωπεύει τον ίδιο τον αλγόριθμο. Ο *Coordinator* είναι υπεύθυνος για το έναυσμα κάθε κύκλου εκτέλεσης. Ένας κύκλος εκτέλεσης εκκινά με την κλήση της ρουτίνας αυτής.

DirectionUpdater

Η ρουτίνα *DirectionUpdater* υλοποιεί την διαδικασία της εξομάλυνσης κατεύθυνσης. Εξάγει για κάθε αντικείμενο τα αποθηκευμένα στίγματα της θέσης του από το αντίστοιχο buffer, υπολογίζει την κατεύθυνσή του και στη συνέχεια την κωδικοποιεί. Τέλος, ενημερώνει την ιστορία κίνησης του κάθε αντικειμένου (*TrajectoryHistory*) με την κωδικοποίηση που υπολόγισε.

A'.4.1 Hashing

Η ρουτίνα αυτή επιτελεί τη διαδικασία της ομαδοποίησης. Κατά την έναρξή της, ορίζει ένα νέο hash table. Στη συνέχεια, τοποθετεί το id κάθε αντικειμένου στο αντίστοιχο κάδο που ορίζει η συμβολική του αναπαράσταση. Όταν γίνει αυτό για όλα τα αντικείμενα, ελέγχεται κάθε κάδος αν τηρεί το ελάχιστο πλήθος αντικειμένων. Για κάθε ομάδα (κάδος αντικειμένων) που έχει περισσότερα από το ελάχιστο πλήθος αντικειμένων, εκτελείται η διαδικασία συσταδοποίησης DBScan.

A'.4.2 DBScan

Εδώ εκτελείται η διαδικασία της συσταδοποίησης. Υλοποιήθηκε μια απλοποιημένη έκδοση του αλγόριθμου DBScan με χρήση *vector's* της STL. Ως αποτέλεσμα επιστρέφονται συστάδες που ικανοποιούν το κατώφλι του ελάχιστου πλήθους αντικειμένων.

A'.4.3 TrajectoryCalc

Τελευταίο βήμα του αλγορίθμου είναι ο υπολογισμός μίας πολυπλεγμένης τροχιάς για κάθε συστάδα που υπολογίστηκε κατά την εκτέλεση του DBScan. Η ρουτίνα διαβάζει τα στίγματα των αντικειμένων της εκάστοτε συστάδας τα οποία βρίσκονται αποθηκευμένα ως ιστορία των πρόσφατων θέσεων (*LocationHistory*). Από αυτά, υπολογίζει το κέντρο βάρους τους για κάθε χρονόσημο, από $[\omega/\beta]$ χρονόσημα πριν έως την τρέχων στιγμή. Η ακολουθία των υπολογισμένων σημείων επιστρέφεται ως η πολυπλεγμένη τροχιά.

Παράρτημα Β΄

Αλγόριθμος DBScan

Ο *DBScan* ([3, 18]) είναι ένας αλγόριθμος συσταδοποίησης χωρικών δεδομένων. Αναπτύχθηκε το 1996 από τους Martin Ester, Hans-Peter Kriegel, Jörg Sander και Xiaowei Xu στο πανεπιστήμιο του Μόναχο. Ο *DBScan* είναι από τους καλύτερους αλγορίθμους συσταδοποίησης που έχουν επινοηθεί μέχρι σήμερα. Έχει αυξημένες επιδόσεις σε σχέση με άλλους αλγόριθμους του είδους τόσο σε χρόνο λειτουργίας όσο και στον ορθότερο εντοπισμό συστάδων αντικειμένων.

Η συσταδοποίηση είναι όρος που προέρχεται από τον ερευνητικό τομέα της εξόρυξης δεδομένων (data mining). Θεωρώντας ότι έχουμε μια βάση δεδομένων (ενδεχομένως πολύ μεγάλου μεγέθους) με κάποιο είδος στοιχείων (ανευ σημασίας), η διαδικασία της συσταδοποίησης στοχεύει στον εντοπισμό ομάδων στοιχείων από τη βάση που να συσχετίζονται με κάποιο συγκεκριμένο τρόπο, δηλαδή να μοιάζουν μεταξύ τους. Για παράδειγμα, αν στην βάση βρίσκονται αποθηκευμένα στοιχεία μισθοδοσίας πολιτών μιας χώρας, προκειμένου να βρεθούν πολίτες που έχουν περίπου ίδιο μισθό (για στατιστικούς σκοπούς) θα μπορούσε να γίνει συσταδοποίηση στα δεδομένα με βάση των μισθό των πολιτών. Η διαδικασία θα επέστρεφε ομάδες από πολίτες για τους οποίους ο μισθός τους είναι παρόμοιος.

Η συσταδοποίηση σε χωρικά δεδομένα [7] γίνεται στηριζόμενη στην απόσταση. Εκτελώντας συσταδοποίηση σε μία βάση δεδομένων με θέσεις αντικειμένων, θα μπορούσαν να εντοπιστούν ομάδες αντικειμένων που βρίσκονται κοντά μεταξύ τους. Αν τα δεδομένα αφορούσαν τροχιές αντικειμένων, τότε η συσταδοποίηση θα μπορούσε να επιστρέψει τμήματα τροχιών που βρίσκονται κοντά το ένα με το άλλο. Η απόσταση μεταξύ των αντικειμένων αλλά και το πόσα αντικείμενα πρέπει να έχει μια συστάδα για να επιστραφεί είναι συνήθως παράμετροι του υπόψη αλγόριθμου.

Ο *DBScan* συσταδοποιεί χωρικά αντικείμενα βάση της χωρικής τους πυκνότητας. Ο αλγόριθμος, προσπαθεί να εντοπίσει συστάδες βρίσκοντας περιοχές στο χώρο που έχουν μεγάλη πυκνότητα σε αντικείμενα. Δέχεται ως είσοδο δύο

παράμετρους:

1. ελάχιστη απόσταση μεταξύ αντικειμένων Eps
2. ελάχιστη πυκνότητα αντικειμένων $MinPts$

Για τον εντοπισμό συστάδων, ο αλγόριθμος ξεκινάει επιλέγοντας τυχαία κάποιο αντικείμενο. Αν το αντικείμενο έχει γύρω του, σε απόσταση μικρότερη της Eps , αριθμό αντικειμένων μεγαλύτερο του $MinPts$ τότε το αντικείμενο καταγράφεται ως πρωτεύον αντικείμενο. Αυτό θα αποτελέσει τη βάση για τον εντοπισμό μιας συστάδας. Για να βρεθεί η έκταση της συστάδας ο αλγόριθμος επισκέπτεται επαναληπτικά κάθε αντικείμενο που γειτονεύει με το πρωτεύον και εντοπίζει άλλα αντικείμενα που γειτονεύουν με αυτό. Κάθε αντικείμενο που εντοπίζεται καταχωρείται ως μέλος της τρέχουσας συστάδας. Όταν δεν μπορούν να βρεθούν άλλα αντικείμενα τότε η συστάδα έχει ολοκληρωθεί. Ο αλγόριθμος επιλέγει τυχαία κάποιο άλλο αντικείμενο από τη βάση (που δεν βρίσκεται σε κάποια συστάδα) και με τον ίδιο τρόπο προσπαθεί να εντοπίσει άλλες συστάδες. Στο σχήμα Β.1 παρουσιάζεται ο ψευδοκώδικας του αλγόριθμου.

```

DBSCAN (SetOfPoints, Eps, MinPts)
//SetOfPoints is UNCLASSIFIED
ClusterId := nextId(NOISE);
FOR i FROM 1 TO SetOfPoints.size DO
  Point := SetOfPoints.get(i);
  IF Point.CIId = UNCLASSIFIED THEN
    IF ExpandCluster(SetOfPoints, Point, ClusterId, Eps, MinPts) THEN
      ClusterId := nextId(ClusterId)
    END IF
  END IF
END FOR
END; // DBSCAN

```

```

ExpandCluster(SetOfPoints, Point, CIId, Eps, MinPts) : Boolean;
seeds:=SetOfPoints.regionQuery(Point,Eps);
IF seeds.size<MinPts THEN // no core point
  SetOfPoint.changeCIId(Point,NOISE);
  RETURN False;
ELSE // all points in seeds are density-reachable from Point
  SetOfPoints.changeCIIds(seeds,CIId);
  seeds.delete(Point);
  WHILE seeds <> Empty DO
    currentP := seeds.first();
    result := SetOfPoints.regionQuery(currentP, Eps);
    IF result.size >= MinPts THEN
      FOR i FROM 1 TO result.size DO
        resultP := result.get(i);
        IF resultP.CIId IN UNCLASSIFIED, NOISE THEN
          IF resultP.CIId = UNCLASSIFIED THEN
            seeds.append(resultP);
          END IF;
          SetOfPoints.changeCIId(resultP,CIId);
        END IF; // UNCLASSIFIED or NOISE
      END FOR;
    END IF; // result.size >= MinPts
    seeds.delete(currentP);
  END WHILE; // seeds <> Empty
  RETURN True;
END IF
END; // ExpandCluster

```


Παράρτημα Γ'

Ορολογία

<i>bucket</i>	κάδος
<i>clustering</i>	συσταδοποίηση
<i>continuous query</i>	ερώτημα διαρκείας
<i>convooy</i>	πομπή/φάλαγγα
<i>data stream</i>	ρεύμα δεδομένων
<i>error</i>	σφάλμα
<i>hashing</i>	κατακερματισμός
<i>hot motion path</i>	μονοπάτι αυξημένης σημασίας
<i>index</i>	ευρετήριο
<i>moving object</i>	κινούμενο αντικείμενο
<i>multiplexing</i>	πολυπλεξία
<i>online</i>	πραγματικού χρόνου
<i>sliding window</i>	κυλιόμενο παράθυρο
<i>smoothing</i>	εξομάλυνση
<i>spatial</i>	χωρικός
<i>symbolic representation</i>	συμβολική αναπαράσταση
<i>timestamp</i>	χρονόσημο
<i>timestamped location</i>	στίγμα
<i>trajectory</i>	τροχιά
<i>trajectory join</i>	σύζευξη τροχιών
<i>trajectory trace</i>	ίχνος τροχιάς

Βιβλιογραφία

- [1] P. Bakalov, M. Hadjieleftheriou, E. Keogh, V.J. Tsotras. Efficient Trajectory Joins using Symbolic Representations. In Proceedings of the 6th International Conference on Mobile Data Management (MDM 2005), Ayia Napa, Cyprus, May 2005.
- [2] Z. Chen, H. T. Shen, and X. Zhou. Discovering Popular Routes from Trajectories. In Proceedings of the IEEE International Conference on Data Engineering (ICDE), Hannover, Germany, April 2011.
- [3] M. Ester, H. Kriegel, J. Sander, and X. Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96), pp. 226-231, Portland, Oregon, USA, August 1996.
- [4] V. Gaede and O. Gunther. Multidimensional Access Methods. ACM Computing Surveys, Vol. 30, No. 2, June 1998
- [5] B. Gedik and L. Liu. MobiEyes: Distributed Processing of Continuously Moving Queries on Moving Objects in a Mobile System. In Proceedings of the 9th International Conference on Extending Database Technology (EDBT'04), Heraklion (Crete), Greece, March 2004.
- [6] L. Golab, and M. Tamer Ozsu. Issues in Data Stream Management. ACM SIGMOD Record, 32(2):5-14, June 2003.
- [7] C. S. Jensen, D. Lin, B. Chin Ooi, and R. Zhang. Effective Density Queries on Continuously Moving Objects. In Proceedings of the 22nd International Conference on Data Engineering (ICDE'06), pp. 71-81, Atlanta, Georgia, USA, April 2006.
- [8] R. Lange, T. Farrell, F. Durr, and K. Rothermel. Remote real-time trajectory simplification. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PERCOM), pp. 1-10, Galveston, Texas, USA, March 2009.

- [9] J. Lee, J. Han, and K. Whang. Trajectory Clustering: a Partition-and-Group Framework. In Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 593-604, Beijing, China, June 2007.
- [10] M.F. Mokbel and W.G. Aref. GPAC: Generic and Progressive Processing of Mobile Queries over Mobile Data. In Proceedings of the 6th international conference on Mobile data management (MDM'05), pp. 155-163, Ayia Napa, Cyprus, May 2005.
- [11] M. F. Mokbel, W. G. Aref, S. E. Hambrusch, and S. Prabhakar. Towards Scalable Location-aware Services: Requirements and Research Issues. In Proceedings of the 11th ACM International Symposium on Advances in Geographic Information Systems (GIS'03), pp. 110-117, New Orleans, Louisiana, USA, November 2003.
- [12] N. Meratnia, and R. de By. Spatiotemporal Compression Techniques for Moving Point Objects. In Proceedings of the 2004 International Conference on Extending Database Technology (EDBT 2004), pp. 765-782, Heraklion (Crete), Greece, March 2004.
- [13] K. Mouratidis, M. Hadjieleftheriou, and D. Papadias. Conceptual Partitioning: An Efficient Method for Continuous Nearest Neighbor Monitoring. In Proceedings of the 24th ACM SIGMOD International Conference on Management of Data, pp. 634-645, Baltimore, Maryland, USA, June 2005.
- [14] J. Ni, C. Ravishankar, and B. Bhanu. Probabilistic Spatial Database Operations. In Proceedings of the 8th International Symposium on Advances in Spatial and Temporal Databases (SSTD 2003), pp. 140-158, Santorini Island, Greece, July 2003.
- [15] K. Patroumpas. Data stream systems for moving objects. M.Sc. Thesis in Geoinformatics, National Technical University of Athens, 2003 (in Greek).
- [16] D. Pfoser, C. S. Jensen and Y. Theodoridis. Novel Approaches to the Indexing of Moving Object Trajectories. In Proceedings of the 26th International Conference on Very Large Databases, Cairo, Egypt, 2000.
- [17] K. Patroumpas and T. Sellis. Managing Trajectories of Moving Objects as Data Streams. In Proceedings of the 2nd Workshop on Spatio-Temporal Database Management (STDBM'04), Toronto, Canada, August 2004.
- [18] J. Sander, M. Ester, H. Kriegel, and X. Xu. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. Data Mining and Knowledge Discovery, 2:169-194, 1998.

- [19] D. Sacharidis, K. Patroumpas, M. Terrovitis, V. Kantere, M. Potamias, K. Mouratidis, and T. Sellis. On-Line Discovery of Hot Motion Paths. In Proceedings of the 11th International Conference on Extending Database Technology (EDBT'08), pp. 392-403, Nantes, France, March 2008.
- [20] A. Arasu, B. Babcock, S. Babu, J. Cieslewicz, M. Datar, K. Ito, R. Motwani, U. Srivastava, and J. Widom. STREAM: The Stanford Data Stream Management System. Technical report, Department of Computer Science, Stanford University, 2004.
- [21] Y. Theodoridis. Ten Benchmark Database Queries for Location-based Services. *Computer Journal*, 46(6): 713-725, 2003.
- [22] Y. Theodoridis, T. Sellis, A.N. Papadopoulos and Y.Manolopoulos. Specifications for efficient indexing in spatiotemporal databases. In Proceedings of 10th International Conference on Scientific and Statistical Database Management (SSDB 1998), pp. 123-132, Capri, Italy, 1998.
- [23] O. Wolfson, B. Xu, S. Chamberlain, L. Jiang. Moving Objects Databases: Issues and Solutions. In Proceedings of 10th International Conference on Scientific and Statistical Database Management (SSDB 1998), pp. 111-122, Capri, Italy, July 1998.
- [24] X. Yu, K. Q. Pu, and N. Koudas. Monitoring k-Nearest Neighbor Queries Over Moving Objects. In Proceedings of the 21st International Conference on Data Engineering (ICDE'05), pp. 631-642, Tokyo, Japan, April 2005.
- [25] H. Jeungy, M.L. Yiu, X. Zhou, C.S. Jensen, and H. T. Shen. Discovery of Convoys in Trajectory Databases. In Proceedings of the VLDB Endowment, 1(1):1068-1080, Auckland, New Zealand, August 2008.

