



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ & ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ - ΜΗΧΑΝΙΚΩΝ ΓΕΩΠΛΗΡΟΦΟΡΙΚΗΣ  
ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ - ΕΡΓΑΣΤΗΡΙΟ ΧΑΡΤΟΓΡΑΦΙΑΣ

# Χαρτογραφική γενίκευση φυσικών γραμμικών χωρικών οντοτήτων με δεσμεύσεις ευαναγνωσιμότητας

Διδακτορική Διατριβή

Βασίλειος Μητρόπουλος

Διπλωματούχος Αγρονόμος και Τοπογράφος Μηχανικός, ΕΜΠ

ΑΘΗΝΑ - ΜΑΡΤΙΟΣ 2023





# Χαρτογραφική γενίκευση φυσικών γραμμικών χωρικών οντοτήτων με δεσμεύσεις ευαναγνωσιμότητας

Διδακτορική Διατριβή

Βασίλειος Μητρόπουλος

Διπλωματούχος Αγρονόμος και Τοπογράφος Μηχανικός, ΕΜΠ

ΤΡΙΜΕΛΗΣ ΣΥΜΒΟΥΛΕΥΤΙΚΗ  
ΕΠΙΤΡΟΠΗ:

Βύρωνας Νάκος,  
Καθηγητής (Επιβλέπων)

Βασιλική Φιλιππακοπούλου,  
Ομότιμη Καθηγήτρια ΣΑΤΜ-ΜΓ, ΕΜΠ

Βασίλειος Βεσκούκης  
Καθηγητής ΣΑΤΜ-ΜΓ, ΕΜΠ

ΕΠΤΑΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ  
ΕΠΙΤΡΟΠΗ:

Μαρίνος Κάβουρας  
Καθηγητής ΣΑΤΜ-ΜΓ, ΕΜΠ

Χρυσούλα Μπούτουρα  
Καθηγήτρια ΣΑΤΜ, ΑΠΘ

Ανδριανή Σκοπελίτη  
Επίκουρη Καθηγήτρια ΣΑΤΜ-ΜΓ, ΕΜΠ

Βασίλειος Κρασανάκης  
Επίκουρος Καθηγητής ΤΜΤ&Γ, ΠΑΔΑ

ΑΘΗΝΑ - ΜΑΡΤΙΟΣ 2023

“Η έγκριση της διδακτορικής διατριβής από τη Σχολή Αγρονόμων και Τοπογράφων Μηχανικών – Μηχανικών Γεωπληροφορικής του ΕΜΠ δεν υποδηλώνει αποδοχή των γνώμων του συγγραφέα” (Ν. 5343/1932, Άρθρο 202).

©2023 Βασίλειος Μητρόπουλος (Με επιφύλαξη παντός δικαιώματος).

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας διδακτορικής διατριβής, 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.

## Πρόλογος

Με τη μετάβαση από την αναλογική στην ψηφιακή χαρτογραφία, η γενίκευση ψηφιακών γραμμικών οντοτήτων αποτελεί ένα από τα κύρια ζητούμενα της χαρτογραφικής παραγωγής. Τα τελευταία χρόνια, μετά από ένα πλήθος θεωρητικών και πρακτικών μελετών, οι χαρτογράφοι προσανατολίζονται στη δημιουργία ‘καθολικών συστημάτων γενίκευσης’, τα οποία εκτιμούν ότι αποτελούν τη βέλτιστη λύση διαχείρισης και γενίκευσης των ψηφιακών γραμμών. Επιδίωξη είναι η προσομοίωση μέσω αυτών της αντίστοιχης χειροκίνητης διαδικασίας. Για το λόγο αυτό, οι χαρτογράφοι επιδιώκουν να καταγράψουν και να ‘αντιγράψουν’ τον τρόπο με τον οποίο ο άνθρωπος αντιλαμβάνεται και αναλύει μία γραμμική οντότητα και να τον αυτοματοποιήσουν μέσω ηλεκτρονικών υπολογιστών. Πάνω σε αυτό το πλαίσιο, οι χαρτογράφοι έχουν καταλήξει στο συμπέρασμα πως, κατά τη χειροκίνητη γενίκευση, ο άνθρωπος τείνει να επιλέγει θέσεις όπου παρατηρείται μεγάλη μεταβολή στην κλίση της γραμμής. Οι θέσεις αυτές ονομάζονται ‘κρίσιμα σημεία’ και παίζουν σημαντικό ρόλο στην ανάλυση και επεξεργασία των ψηφιακών γραμμών. Επιπρόσθετα, ο άνθρωπος αναλύει τη γραμμή, εντοπίζει τμήματά της με κοινά χαρακτηριστικά, τα οποία και μετασχηματίζει κατάλληλα. Έτσι, η φιλοσοφία και η λειτουργία των ‘καθολικών συστημάτων γενίκευσης’ στηρίζεται στην κατάτμηση της γραμμής σε ομοιογενή τμήματα, το χαρακτηρισμό τους μέσω ποσοτικών ή και ποιοτικών κριτηρίων και βάσει αυτών, τη γενίκευση τους μέσω κατάλληλων τελεστών.

Με την παρούσα διατριβή επιχειρείται η θεωρητική και πρακτική προσέγγιση των δύο παραπάνω ζητούμενων. Πιο συγκεκριμένα, δημιουργείται μία μέθοδος εντοπισμού κρίσιμων σημείων κατά μήκος χαρτογραφικών γραμμών, η οποία είναι βασισμένη σε γεωμετρικές αρχές. Επιπρόσθετα, αναπτύσσεται ένα μοντέλο γενίκευσης ψηφιακών χαρτογραφικών γραμμών, το οποίο στηρίζεται στο εννοιολογικό πλαίσιο ‘κατάτμηση-χαρακτηρισμός-γενίκευση’. Ειδικότερα, σχεδιάζεται μία μέθοδος κατά την οποία η γραμμή κατατέμενεται σε ομοιογενή τμήματα με κριτήριο την ευαναγνωσιμότητάς της, η οποία ορίζεται μέσω χαρτογραφικών κανόνων που στηρίζονται στην οπτική αντίληψη. Τα τμήματα που δημιουργούνται κατηγοριοποιούνται με βάση τη γεωμετρία τους και χαρακτηρίζονται μέσω ποσοτικών κριτηρίων. Η γενίκευση της γραμμής ολοκληρώνεται με την εφαρμογή κατάλληλων τελεστών γενίκευσης σε κάθε τμήμα, ανάλογα με την κατηγορία στην οποία ανήκει και τα ποσοτικά χαρακτηριστικά του.

Η παρούσα διδακτορική διατριβή εκπονήθηκε στο Εργαστήριο Χαρτογραφίας του Τομέα Τοπογραφίας της Σχολής Αγρονόμων και Τοπογράφων Μηχανικών - Μηχανικών Γεωπληροφορικής του Εθνικού Μετσόβιου Πολυτεχνείου (Ε.Μ.Π.).

Θα ήθελα να ευχαριστήσω θερμά τον καθηγητή κ. Βύρωνα Νάκο, για τη συνεχή καθοδήγηση, τις συμβουλές, την υποστήριξη και το ειλικρινές ενδιαφέρον του καθ’ όλη τη διάρκεια της εκπόνησης της διδακτορικής διατριβής. Εξίσου θερμές ευχαριστίες θα ήθελα να απευθύνω στην ομότιμη καθηγήτρια κα. Βασιλική Φιλιππακοπούλου για τις συμβουλές και τη βοήθεια που μου προσέφερε όλα αυτά τα χρόνια.

Ευχαριστώ τον καθηγητή κ. Βασίλειο Βεσκούκη για τη συμμετοχή του στην τριμελή συμβουλευτική επιτροπή και τις εποικοδομητικές παρατηρήσεις του. Επίσης, ένα μεγάλο ευχαριστώ σε όλα τα μέλη του Εργαστηρίου Χαρτογραφίας, με τα οποία μοιραστήκαμε για αρκετά χρόνια τον ίδιο χώρο και συντέλεσαν ώστε να έχω μία όμορφη και ευχάριστη καθημερινότητα.

Θα ήθελα να αφιερώσω την παρούσα διδακτορική διατριβή στην Άννα, το Χάρη και τη Λυγία. Τους ευχαριστώ για τη συμπαράσταση, την υπομονή και την κατανόησή τους όλη αυτή την περίοδο.

Βασίλης Μητρόπουλος

Αθήνα, Μάρτιος 2023

## Πίνακας Περιεχομένων

<b>Πίνακας Περιεχομένων Σχημάτων.....</b>	<b>5</b>
<b>Πίνακας Περιεχομένων Πινάκων.....</b>	<b>9</b>
<b>Περίληψη .....</b>	<b>11</b>
<b>Extended Abstract.....</b>	<b>14</b>
<b>ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ.....</b>	<b>16</b>
1.1 Βιβλιογραφική Ανασκόπηση - Θεωρητική αναφορά.....	16
1.2 Στόχος της διατριβής.....	24
2.1 Κατηγορίες $\epsilon$ -τμημάτων.....	28
2.2 Ορισμός διαμέτρου εφαρμογής $\epsilon$ .....	29
2.3 Υλοποίηση της τεχνικής του κυλιόμενου δίσκου σε ηλεκτρονικό υπολογιστή .....	30
2.3.1 Βασική αρχή και περιβάλλον υλοποίησης.....	30
2.3.2 Εντοπισμός, φιλτράρισμα και συγχώνευση $\epsilon$ -μη-κυρτών τμημάτων και εντοπισμός $\epsilon$ -κυρτών τμημάτων ..	32
2.3.3 Διαχωρισμός $\epsilon$ -μη-κυρτών τμημάτων σε κατηγορίες.....	33
2.3.4 Νέα συγχώνευση $\epsilon$ -μη-κυρτών τμημάτων και δημιουργία τελικών $\epsilon$ -τμημάτων .....	36
<b>ΚΕΦΑΛΑΙΟ 3: ΜΕΘΟΔΟΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΧΑΡΤΟΓΡΑΦΙΚΩΝ ΓΡΑΜΜΩΝ .....</b>	<b>39</b>
3.1 Μέθοδος εντοπισμού κρίσιμων σημείων κατά μήκος χαρτογραφικών γραμμών μέσω του δείκτη αναλογίας μήκους LR.....	39
3.1.1 Ορισμός δείκτη αναλογίας μήκους LR .....	41
3.1.2 Ορισμός μεγέθους ακτίνας κύκλου R .....	44
3.1.3 Ταξινόμηση τιμών του δείκτη LR και ορισμός ορίου επιλογής .....	46
3.1.4 Σύγκριση δείκτη LR με ‘μέτρο ημιτονικότητας’ του Dutton .....	47
3.1.5 Ορισμός δεικτών LLR και ALR.....	48
3.1.6 Έλεγχος λειτουργικότητας μεθόδου .....	49
3.1.6.1 Σύγκριση με σχετικές μελέτες .....	49
3.1.6.2 Σύγκριση με μέθοδο οφθαλμικών καταγραφών (Eye-Tracking).....	55
3.2 Εξομάλυνση $\epsilon$ -τμημάτων .....	64
3.2.1 Θεωρητική αναφορά .....	64
3.2.2 Τελεστής εξομάλυνσης Gauss – Προσαρμογή και ενσωμάτωση στο μοντέλο.....	64
3.3 Μέθοδος εντοπισμού καμπών κατά μήκος των γραμμών .....	68
3.4 Ομοπαράλληλος μετασχηματισμός.....	70
3.5 Αλγόριθμος Balloon .....	71
3.6 Αλγόριθμος Bendsimplify.....	73
<b>ΚΕΦΑΛΑΙΟ 4: ΓΕΝΙΚΕΥΣΗ <math>\epsilon</math>-ΤΜΗΜΑΤΩΝ ΧΑΡΤΟΓΡΑΦΙΚΩΝ ΓΡΑΜΜΩΝ .....</b>	<b>74</b>
4.1 Γενίκευση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων (Κατηγορία Α) .....	74
4.1.1 Διεύρυνση μονόπλευρων $\epsilon$ -μη κυρτών τμημάτων .....	74
4.1.2 Επεξεργασία διευρυμένων μονόπλευρων $\epsilon$ -μη κυρτών τμημάτων .....	77
4.1.2.1 Εξομάλυνση και υπολογισμός γεωμετρικών χαρακτηριστικών των μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων.....	78
4.1.2.2 Διόγκωση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων.....	80
4.1.2.2.1 Διόγκωση με δεσμεύσεις – Αλγόριθμος ‘Δεσμευμένης Διόγκωσης’ .....	82
4.1.2.2.2 Διόγκωση χωρίς δεσμεύσεις – Αλγόριθμος ‘Ανεξάρτητης Διόγκωσης’ .....	89
4.2 Γενίκευση αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων (Κατηγορία Β) .....	93
4.2.1 Εντοπισμός καμπών και απλοποίηση των $\epsilon$ -μη-κυρτών τμημάτων .....	94
4.2.1.1 Αρχή λειτουργίας.....	94
4.2.1.2 Επιλογή προς διατήρηση/αφαίρεση καμπών .....	96
4.2.2 Διόγκωση καμπών.....	99

4.3 Γενίκευση τμημάτων σύγκλισης (Κατηγορία Γ) .....	101
4.3.1 Ιδιότητες περιοχής σύγκλισης .....	101
4.3.2 Επεξεργασία περιοχής σύγκλισης και διεύρυνση τμημάτων σύγκλισης.....	103
4.3.3 Συμπύεση τμήματος σύγκλισης - Αλγόριθμος 'Depress' .....	109
4.4 Γενίκευση $\epsilon$ -κυρτών τμημάτων (Κατηγορία Δ) .....	111
4.5 Συνένωση γενικευμένων $\epsilon$ -τμημάτων - Δημιουργία γενικευμένης γραμμής.....	112
4.6 Έλεγχος ευαναγνωσιμότητας γενικευμένης γραμμής στην παράγωγη κλίμακα. ....	113
<b>ΚΕΦΑΛΑΙΟ 5: ΠΙΛΟΤΙΚΗ ΕΦΑΡΜΟΓΗ.....</b>	<b>115</b>
5.1. Το αρχείο αναφοράς.....	115
5.2. Πιλοτική εφαρμογή του δείκτη αναλογίας μήκους LR.....	115
5.2.1. Εντοπισμός κρίσιμων σημείων μέσω του δείκτη ALR .....	115
5.2.2 Σύγκριση δύο αλγορίθμων απλοποίησης γραμμών με μέτρο τα κρίσιμα σημεία που έχουν εντοπισθεί μέσω του δείκτη ALR .....	118
5.3 Πιλοτική εφαρμογή μοντέλου κατάτμησης – γενίκευσης γραμμών .....	124
5.3.1 Σενάριο γενίκευσης 1:1.000.000.....	124
5.3.1.1 Κατάτμηση γραμμής.....	124
5.3.1.2 Γενίκευση γραμμής.....	126
5.3.1.2.1 Γενίκευση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων .....	126
5.3.1.2.2 Γενίκευση αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων .....	131
5.3.1.2.3 Γενίκευση τμημάτων σύγκλισης.....	140
5.3.1.2.4 Γενίκευση $\epsilon$ -κυρτών τμημάτων .....	142
5.3.1.2.5 Συνένωση γενικευμένων $\epsilon$ -τμημάτων - Εξομάλυνση γενικευμένης γραμμής - Δημιουργία γενικευμένης γραμμής.....	143
5.3.2 Σενάριο γενίκευσης 1:500.000.....	144
5.3.2.1 Κατάτμηση γραμμής.....	144
5.3.2.2 Γενίκευση γραμμής.....	146
5.3.2.2.1 Γενίκευση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων .....	146
5.3.2.2.2 Γενίκευση αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων .....	149
5.3.2.2.3 Γενίκευση $\epsilon$ -κυρτών τμημάτων .....	154
5.3.2.2.4 Συνένωση γενικευμένων $\epsilon$ -τμημάτων - Εξομάλυνση γενικευμένης γραμμής - Δημιουργία γενικευμένης γραμμής.....	155
5.3.3 Αξιολόγηση αποτελέσματος .....	156
<b>ΚΕΦΑΛΑΙΟ 6: ΣΥΜΠΕΡΑΣΜΑΤΑ.....</b>	<b>164</b>
<b>Βιβλιογραφία .....</b>	<b>170</b>
<b>Παράρτημα.....</b>	<b>175</b>
Π3.1 Πρόγραμμα εντοπισμού κρίσιμων σημείων μέσω του δείκτη αναλογίας μήκους LR .....	175
Π3.2 Πρόγραμμα τελεστή εξομάλυνσης Gauss (ανοιχτές γραμμές) .....	178
Π3.2Κ Πρόγραμμα τελεστή εξομάλυνσης Gauss (κλειστές γραμμές).....	182
Π3.3 Πρόγραμμα εντοπισμού καμπών .....	185
Π3.4 Πρόγραμμα Ομοπαράλληλου Μετασχηματισμού .....	201
Π4.1.1 Διεύρυνση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων .....	203
Π4.1.2 Πρόγραμμα γενίκευσης μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων .....	208
Π4.3 Πρόγραμμα γενίκευσης τμημάτων σύγκλισης .....	284



## Πίνακας Περιεχομένων Σχημάτων

<b>Σχήμα 2.1.</b> Παράδειγμα $\epsilon$ -κυρτού τμήματος γραμμής.	<b>27</b>
<b>Σχήμα 2.2.</b> Παράδειγμα $\epsilon$ -μη-κυρτού τμήματος γραμμής.	<b>27</b>
<b>Σχήμα 2.3.</b> Τυπικά παραδείγματα κατηγοριών $\epsilon$ -τμημάτων που αποκόπτονται με την προτεινόμενη τεχνική.	<b>29</b>
<b>Σχήμα 2.4.</b> Επιλογή διαμέτρου εφαρμογής $\epsilon$ του δίσκου του Perkal.	<b>30</b>
<b>Σχήμα 2.5.</b> Παράδειγμα εφαρμογής της τεχνικής του Perkal σε ψηφιακό περιβάλλον.	<b>31</b>
<b>Σχήμα 2.6.</b> Μοντέλο εντοπισμού, φιλτραρίσματος, συγχώνευσης $\epsilon$ -μη-κυρτών τμημάτων και εντοπισμού $\epsilon$ -κυρτών τμημάτων.	<b>32</b>
<b>Σχήμα 2.7.</b> Μοντέλο διαχωρισμού $\epsilon$ -μη-κυρτών τμημάτων σε κατηγορίες.	<b>33</b>
<b>Σχήμα 2.8.</b> Παράδειγμα απομονωμένου (αριστερά) και επικαλυπτόμενου (δεξιά) τμήματος.	<b>34</b>
<b>Σχήμα 2.9.</b> Δημιουργία τμήματος σύγκλισης.	<b>35</b>
<b>Σχήμα 2.10.</b> Μοντέλο νέας συγχώνευσης $\epsilon$ -μη-κυρτών τμημάτων και δημιουργίας τελικών $\epsilon$ -τμημάτων.	<b>37</b>
<b>Σχήμα 3.1.</b> Τυπική περίπτωση τομής μεταξύ γραμμής και κύκλου.	<b>41</b>
<b>Σχήμα 3.2.</b> Τέσσερις εναλλακτικές περιπτώσεις τομής μεταξύ κύκλου και γραμμής.	<b>42</b>
<b>Σχήμα 3.3.</b> Μεταβολή του αναπτύγματος $L$ και της χορδής $S$ συναρτήσει της καμπυλότητας.	<b>43</b>
<b>Σχήμα 3.4.</b> Εντοπισμός κρίσιμων σημείων σε διάγραμμα LR.	<b>44</b>
<b>Σχήμα 3.5.</b> Τρεις τύποι χαρακτηριστικών σημείων: Σημείο τύπου 1: θέση πολύ έντονης μεταβολής κλίσης Σημείο τύπου 2: θέση έντονης μεταβολής κλίσης Σημείο τύπου 3: θέση ευθυγράμμου τμήματος.	<b>44</b>
<b>Σχήμα 3.6.</b> Μεταβολή του αναπτύγματος $L$ και του μήκους χορδής $S$ , αυξανόμενης της ακτίνας, για τρεις τύπους χαρακτηριστικών σημείων.	<b>45</b>
<b>Σχήμα 3.7.</b> Ποσοστιαία αύξηση του δείκτη LR, αυξανόμενης της ακτίνας, για τρεις τύπους χαρακτηριστικών σημείων.	<b>46</b>
<b>Σχήμα 3.8.</b> Χαρακτηριστικές μορφές των καμπυλών για τις τρεις ομάδες τιμών του δείκτη LR.	<b>47</b>
<b>Σχήμα 3.9.</b> Σύγκριση μεταξύ της ταξινόμησης των τιμών του δείκτη LR και των τιμών ημιτονικότητας SV.	<b>48</b>

<b>Σχήμα 3.10.</b> Γραμμές από τις μελέτες των Marino και Thara.	50
<b>Σχήμα 3.11.</b> Χάρτες θερμότητας των πέντε υπό μελέτη γραμμών.	59
<b>Σχήμα 3.12.</b> Η γεωμετρία του συστήματος υποκείμενο-οθόνη κατά τη διάρκεια του πειράματος.	60
<b>Σχήμα 3.13.</b> Η θεωρητική γραμμή του Thara, τα κρίσιμα σημεία και οι τέσσερις τύποι τμημάτων σύγκλισης.	62
<b>Σχήμα 3.14.</b> Παράδειγμα βαρών σημείων που συμμετέχουν στη μετατόπιση σημείου $i$ .	65
<b>Σχήμα 3.15.</b> Παράδειγμα χρήσης ‘εικονικών’ σημείων κατά την υλοποίηση του τελεστή $Gauss$ με $k=4\sigma$ και $\sigma=1$ .	66
<b>Σχήμα 3.16.</b> Καμπές κατά μήκος γραμμών και χαρακτηριστικά αυτών.	69
<b>Σχήμα 3.17.</b> Γωνίες διεύθυνσης ευθυγράμμων τμημάτων $(i-1)-i$ και $(i+1)-i$	69
<b>Σχήμα 3.18.</b> Παράδειγμα ομοπαράλληλου μετασχηματισμού μονοπλευρων $\epsilon$ -μη-κυρτών τμημάτων.	71
<b>Σχήμα 3.19.</b> Ο αλγόριθμος <i>Balloon</i> .	72
<b>Σχήμα 4.1.</b> Διεύρυνση μονόπλευρου $\epsilon$ -μη-κυρτού τμήματος.	75
<b>Σχήμα 4.2.</b> Μοντέλο διεύρυνσης μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων.	77
<b>Σχήμα 4.3.</b> Παραδείγματα εφαρμογής αλγορίθμου ‘ <i>Balloon</i> ’ σε μη εξομαλυσμένη ή μη επαρκώς εξομαλυσμένη γραμμή.	78
<b>Σχήμα 4.4.</b> Παράδειγμα εφαρμογής τελεστή εξομάλυνσης $Gauss$ σε μονόπλευρο $\epsilon$ -μη-κυρτό τμήμα.	80
<b>Σχήμα 4.5.</b> Οι κορυφές $K_{Σ_E}$ και $K_{Σ_A}$ εντοπίζονται εκτός του δίσκου.	84
<b>Σχήμα 4.6.</b> Η κορυφή $K_{Σ_E}$ εντοπίζεται πάνω στο δίσκο και η κορυφή $K_{Σ_A}$ εκτός αυτού.	85
<b>Σχήμα 4.7.</b> Η κορυφή $K_{Σ_E}$ εντοπίζεται εντός του δίσκου και η κορυφή $K_{Σ_A}$ εκτός αυτού.	86
<b>Σχήμα 4.8.</b> Οι κορυφές $K_{Σ_E}$ και $K_{Σ_A}$ εντοπίζονται εντός του δίσκου.	88
<b>Σχήμα 4.9.</b> Περιπτώσεις εφαρμογής αλγορίθμου ‘Ανεξάρτητης Διόγκωσης’.	90
<b>Σχήμα 4.10.</b> Παράδειγμα εφαρμογής του αλγορίθμου ‘ <i>Balloon</i> ’ και του ομοπαράλληλου μετασχηματισμού κατά την εκτέλεση του αλγορίθμου ‘Ανεξάρτητης Διόγκωσης’.	92
<b>Σχήμα 4.11.</b> Παράδειγμα καμπών που ανευρίσκονται στο αρχικό και σε εξομαλυσμένες εκδοχές αμφίπλευρου $\epsilon$ -μη-κυρτού τμήματος.	95
<b>Σχήμα 4.12.</b> Παράδειγμα καμπών που πληρούν το κριτήριο $L_{BL} \geq \epsilon$ .	97

<b>Σχήμα 4.13.</b> Παράδειγμα καμπών που δεν πληρούν το κριτήριο $BL \geq \epsilon$ .	<b>98</b>
<b>Σχήμα 4.14.</b> Παράδειγμα εξομάλυνσης κύριων καμπών.	<b>99</b>
<b>Σχήμα 4.15.</b> Παράδειγμα διόγκωσης κύριων καμπών.	<b>100</b>
<b>Σχήμα 4.16.</b> Παράδειγμα περιοχής σύγκλισης.	<b>102</b>
<b>Σχήμα 4.17.</b> Μοντέλο υπολογισμού ποσοτικών ιδιοτήτων και εντοπισμού χαρακτηριστικών θέσεων των περιοχών σύγκλισης.	<b>103</b>
<b>Σχήμα 4.18.</b> Προβλήματα κατά τη γενίκευση τμημάτων σύγκλισης.	<b>105</b>
<b>Σχήμα 4.19.</b> Μοντέλο δημιουργίας διευρυμένης περιοχής σύγκλισης και αποκοπής τελικών $\epsilon$ -τμημάτων.	<b>109</b>
<b>Σχήμα 4.20.</b> Ο αλγόριθμος 'Depress'.	<b>110</b>
<b>Σχήμα 4.21.</b> Μοντέλο γενίκευσης $\epsilon$ -κυρτών τμημάτων.	<b>111</b>
<b>Σχήμα 4.22.</b> Μοντέλοσυνένωσης γενικευμένων $\epsilon$ -τμημάτων.	<b>112</b>
<b>Σχήμα 4.23.</b> Παράδειγμα διόρθωσης ασυνεχειών ή σπασιμάτων γενικευμένης γραμμής.	<b>112</b>
<b>Σχήμα 4.24.</b> Μοντέλο εξομάλυνσης γενικευμένης γραμμής.	<b>113</b>
<b>Σχήμα 4.25.</b> Μοντέλο ελέγχου ευαναγνωσιμότητας γενικευμένης γραμμής στην παράγωγη κλίμακα.	<b>114</b>
<b>Σχήμα 5.1.</b> Ακτογραμμή της νήσου Περιστέρα και τα 135 κρίσιμα σημεία.	<b>116</b>
<b>Σχήμα 5.2.</b> Απλοποιημένες ακτογραμμές της νήσου Περιστέρα σε πέντε παράγωγες κλίμακες.	<b>129</b>
<b>Σχήμα 5.3.</b> Αποτέλεσμα εφαρμογής του μοντέλου κατάτμησης στην ακτογραμμή της νήσου Περιστέρας για το σενάριο γενίκευσης 1:1.000.000.	<b>125</b>
<b>Σχήμα 5.4.</b> Παράδειγμα γενίκευσης μονόπλευρου $\epsilon$ -μη-κυρτού τμήματος.	<b>127</b>
<b>Σχήμα 5.5.</b> Παραδείγματα αλληλεπίδρασης μετασχηματισμών εξομάλυνσης και διόγκωσης.	<b>128</b>
<b>Σχήμα 5.6.</b> Παράδειγμα εφαρμογής αλγορίθμου 'Ανεξάρτητης Διόγκωσης'.	<b>130</b>
<b>Σχήμα 5.7.</b> Τα μονόπλευρα $\epsilon$ -μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης.	<b>131</b>
<b>Σχήμα 5.8.</b> Οι κύριες καμπές αντιστοιχιζόμενες στα αρχικά αμφίπλευρα $\epsilon$ -μη-κυρτά τμήματα.	<b>134</b>
<b>Σχήμα 5.9.</b> Τα αμφίπλευρα $\epsilon$ -μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης.	<b>136</b>

<b>Σχήμα 5.10.</b> $\epsilon$ -μη-κυρτά τμήματα που προκύπτουν μετά τον έλεγχο ευαναγνωσιμότητας.	<b>137</b>
<b>Σχήμα 5.11.</b> Παράδειγμα επίλυσης προβλήματος ευαναγνωσιμότητας σε 'διπλό' αμφίπλευρο $\epsilon$ -μη-κυρτό-τμήμα.	<b>139</b>
<b>Σχήμα 5.12.</b> Νέα $\epsilon$ -μη-κυρτά τμήματα μετά τη γενίκευση.	<b>140</b>
<b>Σχήμα 5.13.</b> Προεπεξεργασία και γενίκευση τμημάτων σύγκλισης.	<b>141</b>
<b>Σχήμα 5.14.</b> Παραδείγματα γενίκευσης $\epsilon$ -κυρτών τμημάτων.	<b>143</b>
<b>Σχήμα 5.15.</b> Ακτογραμμή της νήσου Περιστεράς όπως προκύπτει από την εφαρμογή του μοντέλου για σενάριο γενίκευσης 1:1.000.000.	<b>143</b>
<b>Σχήμα 5.16 .</b> Αποτέλεσμα εφαρμογής του μοντέλου κατάτμησης στην ακτογραμμή της νήσου Περιστεράς για σενάριο γενίκευσης 1:500.000.	<b>145</b>
<b>Σχήμα 5.17.</b> Τα μονόπλευρα $\epsilon$ -μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης (σενάριο γενίκευσης 1:500.00).	<b>148</b>
<b>Σχήμα 5.18.</b> Οι κύριες καμπές αντιστοιχισμένες στα αρχικά αμφίπλευρα $\epsilon$ -μη-κυρτά τμήματα (σενάριο γενίκευσης 1:500.000).	<b>151</b>
<b>Σχήμα 5.19.</b> Τα αμφίπλευρα $\epsilon$ -μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης (σενάριο γενίκευσης 1:500.000).	<b>152</b>
<b>Σχήμα 5.20.</b> $\epsilon$ -μη-κυρτά τμήματα που προκύπτουν μετά τον έλεγχο ευαναγνωσιμότητας (σενάριο γενίκευσης 1:500.000).	<b>153</b>
<b>Σχήμα 5.21.</b> Νέα $\epsilon$ -μη-κυρτά τμήματα μετά τη γενίκευση (σενάριο γενίκευσης 1:500.000).	<b>154</b>
<b>Σχήμα 5.22.</b> Ακτογραμμή της νήσου Περιστεράς όπως προκύπτει από την εφαρμογή του μοντέλου για σενάριο γενίκευσης 1:500.000.	<b>156</b>
<b>Σχήμα 5.23.</b> Η ακτογραμμή της νήσου Περιστεράς παραγμένη από τέσσερις διαφορετικές μεθόδους για σενάριο γενίκευσης 1:1.000.000.	<b>158</b>
<b>Σχήμα 5.24.</b> Η ακτογραμμή της νήσου Περιστεράς παραγμένη από τέσσερις διαφορετικές μεθόδους (σενάριο γενίκευσης 1:500.000).	<b>162</b>

## Πίνακας Περιεχομένων Πινάκων

<b>Πίνακας 3.1.</b> Παράμετροι εφαρμογής του δείκτη LR και αποτελέσματα της σύγκρισης.	<b>51</b>
<b>Πίνακας 3.2.</b> Τιμές του δείκτη LR σε κοινά κρίσιμα σημεία των μελετών των Marino και Thara.	<b>52</b>
<b>Πίνακας 3.3.</b> Διάρκεια προβολής πειραματικών γραμμών.	<b>58</b>
<b>Πίνακας 3.4.</b> Διάρκεια σάρωσης κάθε γραμμής και θέσεων γύρω από κρίσιμα σημεία όπως προκύπτει από τα πρωτογενή δεδομένα.	<b>61</b>
<b>Πίνακας 3.5.</b> Πυκνότητα χρονικής διάρκειας προσηλώσεων ανά pixel γύρω από κάθε γραμμή και σε θέσεις γύρω από κρίσιμα και μη κρίσιμα σημεία.	<b>61</b>
<b>Πίνακας 5.1.</b> Διατηρούμενες κορυφές για τα πέντε σενάρια απλοποίησης μετά από την εφαρμογή των 'Αρχών της Επιλογής'.	<b>119</b>
<b>Πίνακας 5.2.</b> Διατηρούμενα κρίσιμα σημεία στα πέντε σενάρια απλοποίησης.	<b>119</b>
<b>Πίνακας 5.3.</b> Διατηρούμενα κρίσιμα σημεία ταξινομημένα στις τρεις ομάδες τιμών του δείκτη LR μετά από τα πέντε σενάρια απλοποίησης.	<b>122</b>
<b>Πίνακας 5.4.</b> Αριθμός καμπών και παράμετροι εφαρμογής του τελεστή εξομάλυνσης στα μονόπλευρα $\epsilon$ -μη-κυρτά τμήματα.	<b>129</b>
<b>Πίνακας 5.5.</b> Παράμετροι διόγκωσης μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων.	<b>130</b>
<b>Πίνακας 5.6.</b> Αριθμός καμπών και παράμετροι εφαρμογής εξομάλυνσης στα αμφίπλευρα $\epsilon$ -μη-κυρτά τμήματα.	<b>132</b>
<b>Πίνακας 5.7.</b> Ποσοτικά χαρακτηριστικά κύριων καμπών αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων.	<b>133</b>
<b>Πίνακας 5.8.</b> Παράμετροι διόγκωσης αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων.	<b>135</b>
<b>Πίνακας 5.9.</b> Παράμετροι γενίκευσης νέων μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων.	<b>138</b>
<b>Πίνακας 5.10.</b> Παράμετροι γενίκευσης νέων αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων.	<b>138</b>
<b>Πίνακας 5.11.</b> Δεδομένα απλοποίησης $\epsilon$ -κυρτών τμημάτων.	<b>142</b>
<b>Πίνακας 5.12.</b> Αριθμός καμπών και παράμετροι εφαρμογής του τελεστή εξομάλυνσης στα μονόπλευρα $\epsilon$ -μη-κυρτά τμήματα (σενάριο γενίκευσης 1:500.000).	<b>146</b>
<b>Πίνακας 5.13.</b> Παράμετροι διόγκωσης μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).	<b>147</b>
<b>Πίνακας 5.14.</b> Αριθμός καμπών και παράμετροι εφαρμογής εξομάλυνσης στα αμφίπλευρα $\epsilon$ -μη-κυρτά τμήματα (σενάριο γενίκευσης 1:500.000).	<b>149</b>

---

<b>Πίνακας 5.15.</b> Ποσοτικά χαρακτηριστικά κύριων καμπών αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).	<b>150</b>
<b>Πίνακας 5.16.</b> Παράμετροι διόγκωσης αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).	<b>151</b>
<b>Πίνακας 5.17.</b> Παράμετροι γενίκευσης νέων μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).	<b>154</b>
<b>Πίνακας 5.18.</b> Δεδομένα απλοποίησης $\epsilon$ -κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).	<b>155</b>

---

## Περίληψη

Η παρούσα διατριβή εστιάζει στο ερευνητικό πεδίο της γενίκευσης ψηφιακών χαρτογραφικών γραμμών εισάγοντας μια ολοκληρωμένη μεθοδολογία γενίκευσης φυσικών γραμμικών οντοτήτων. Πραγματεύεται δύο βασικές έννοιες που απασχολούν την χαρτογραφική κοινότητα στο εν λόγω πεδίο: 1. τη γενίκευση γραμμικών οντοτήτων με 'οδηγό' το εννοιολογικό πλαίσιο 'κατάτμηση - χαρακτηρισμός - γενίκευση' και 2. τον εντοπισμό κρίσιμων σημείων κατά μήκος χαρτογραφικών γραμμών.

Η διατριβή αναπτύσσεται σε τρία επίπεδα. Αρχικά, γίνεται μία ανασκόπηση μελετών που έχουν πραγματοποιηθεί γύρω από το αντικείμενο. Συγκεκριμένα, παρουσιάζεται η έννοια της γενίκευσης γραμμικών οντοτήτων και καταγράφονται και αναλύονται οι σύγχρονες ερευνητικές τάσεις στο αντικείμενο της χαρτογραφικής γενίκευσης. Εισάγεται η έννοια των κρίσιμων σημείων και αναδεικνύεται η σημασία αυτών στη διαδικασία της γενίκευσης. Στη συνέχεια γίνεται αναφορά στις μεθόδους που ακολουθήθηκαν για την αυτοματοποίηση σε ψηφιακό περιβάλλον, αρχικά, μέσω των αλγορίθμων απλοποίησης. Περιγράφεται το θεωρητικό πλαίσιο λειτουργίας των πιο διαδεδομένων εξ αυτών και στη συνέχεια, με βάση την έννοια των κρίσιμων σημείων και την ανάλυση μελετών γύρω από το αντικείμενο, επιδιώκεται μία αξιολόγησή τους. Όπως προκύπτει από τη βιβλιογραφία, οι ανεπάρκειες των αλγορίθμων απλοποίησης οδηγούν τους μελετητές στην αναζήτηση νέων τρόπων επεξεργασίας για τη γενίκευση των χαρτογραφικών γραμμών. Μία από τις καινοτόμες ιδέες που ερευνώνται είναι ο εντοπισμός ομοιογενών γραμμών ή ομοιογενών τμημάτων κατά μήκος των γραμμών, ο χαρακτηρισμός αυτών μέσω ποιοτικών και ποσοτικών κριτηρίων και η γενίκευσή τους μέσω κατάλληλων τελεστών κατά περίπτωση.

Σε δεύτερο επίπεδο, παρατίθενται δύο μεθοδολογίες που αναφέρονται στα δύο ανωτέρω ζητούμενα. Η δομή και η αξιολόγησή τους αναπτύσσεται στο κύριο μέρος του κειμένου.

Η πρώτη έχει στόχο τη δημιουργία ενός μοντέλου γενίκευσης ψηφιακών γραμμών, η οποία στηρίζεται στο εννοιολογικό πλαίσιο 'κατάτμηση-χαρακτηρισμός-γενίκευση'. Πιο συγκεκριμένα, σχεδιάζεται μία μεθοδολογία κατά την οποία η εκάστοτε γραμμή κατατέμενεται σε ομοιογενή τμήματα. Ως κριτήριο κατάτμησης χρησιμοποιείται η ευαναγνωσιμότητα των γραμμών, η οποία ορίζεται μέσω χαρτογραφικών κανόνων που στηρίζονται στην οπτική αντίληψη. Στη συνέχεια, θεσπίζονται παράμετροι χαρακτηρισμού των τμημάτων κατάτμησης της γραμμικής οντότητας (ονομάζονται  $\in$ -τμήματα). Ορίζονται τέσσερις κατηγορίες τμημάτων, στις οποίες αυτά εντάσσονται ανάλογα με τη μορφή και τη γεωμετρία τους. Παράλληλα δε, καθορίζονται και υπολογίζονται ποσοτικά μέτρα που τα χαρακτηρίζουν. Η γενίκευση της γραμμής ολοκληρώνεται με την εφαρμογή κατάλληλων τελεστών γενίκευσης (μεγέθυνση, συγχώνευση, απλοποίηση, διεύρυνση κ.λπ.) σε κάθε  $\in$ -τμήμα, ανάλογα με την κατηγορία στην οποία ανήκει και τα ποσοτικά χαρακτηριστικά του. Για κάθε κατηγορία, προτείνονται τελεστές γενίκευσης (διεύρυνση, απλοποίηση, διόγκωση, εξομάλυνση κτλ.) που κρίνονται απαραίτητοι για τη γενίκευσή τους και σχεδιάζονται οι αλγόριθμοι εφαρμογής τους. Οι επιμέρους διεργασίες επεξεργασίας των

γραμμών κωδικοποιούνται, αλληλοσυνδέονται και ενσωματώνονται σε κατάλληλο λογισμικό περιβάλλον, ώστε να δημιουργηθεί μία ενιαία πλατφόρμα υλοποίησης των τεχνικών. Ο σχεδιασμός της μεθόδου πραγματοποιείται με άξονα την εφαρμογή της σε γραμμές που απεικονίζουν φυσικές οντότητες, οι οποίες, λόγω της πολυπλοκότητας και της τυχαίας μορφής τους, παρουσιάζουν ιδιαίτερο ερευνητικό ενδιαφέρον.

Με τη δεύτερη μεθοδολογία προτείνεται ένας τρόπος εντοπισμού των κρίσιμων σημείων χαρτογραφικών γραμμών. Η μέθοδος είναι βασισμένη σε γεωμετρικές αρχές και χρησιμοποιεί τον δείκτη αναλογίας μήκους LR ως μέτρο εκτίμησης της αλλαγής κλίσης κατά μήκος μιας ψηφιακής γραμμής. Ο δείκτης LR εφαρμόζεται, τοπικά, σε κάθε σημείο της γραμμής και είναι δομημένος με τρόπο, ώστε η τιμή του να αυξάνει ανάλογα με την αλλαγή της κλίσης της. Η εξέταση σε τοπικό επίπεδο διασφαλίζεται με την αποκοπή της γραμμής μέσω ενός κύκλου σταθερής ακτίνας γύρω από κάθε της σημείο. Οι κορυφές της γραμμής, στις οποίες οι τιμές του δείκτη εμφανίζουν τοπικά μέγιστα, μεγαλύτερα από ένα προκαθορισμένο όριο, θεωρούνται ως κρίσιμα σημεία. Στο παρόν κείμενο αναλύεται η δομή του δείκτη LR και καθορίζονται οι παράμετροι λειτουργίας του.

Σε τρίτο επίπεδο πραγματοποιείται πιλοτική εφαρμογή όπου ελέγχεται η αποτελεσματικότητα των προτεινόμενων μεθόδων. Η μέθοδος εντοπισμού κρίσιμων σημείων, αρχικά, εφαρμόζεται σε τέσσερις γραμμές (τρεις γεωμορφολογικές και μία γεωμετρική), οι οποίες επιλέχθηκαν από δύο σχετικές με το αντικείμενο μελέτες, προκειμένου να ελεγχθεί και να αξιολογηθεί με βάση εκ των προτέρων γνωστά κρίσιμα σημεία. Στη συνέχεια, η μέθοδος εφαρμόζεται πιλοτικά στην ακτογραμμή της νήσου Περιστεράς κλίμακας 1:50.000 και τα αποτελέσματα αναλύονται. Τέλος, ο δείκτης LR χρησιμοποιείται για να συγκριθούν τα αποτελέσματα της εφαρμογής σε πέντε διαδοχικά σενάρια γενίκευσης, δύο αλγορίθμων απλοποίησης γραμμών ('pointremove' και 'bendsimplify' του λογισμικού ArcGIS). Η μέθοδος 'κατάτμησης - χαρακτηρισμού - γενίκευσης' εφαρμόζεται στην ακτογραμμή της νήσου Περιστερά κλίμακας 1:50.000, για δύο σενάρια γενίκευσης με παράγωγες κλίμακες 1:500.000 και 1:1.000.000. Στόχος είναι η αποτίμηση, τόσο των επιμέρους διεργασιών, όσο και του τελικού προϊόντος που προκύπτει από την τεχνική, μετά την εφαρμογή της σε μία γραμμή τυχαίας μορφής και γεωμετρίας και για σενάρια μεγάλης αλλαγής κλίμακας. Τα αποτελέσματα αξιολογούνται με βάση τους κανόνες που έχουν ορισθεί, αλλά και συγκρινόμενα με: α) εκείνα που προκύπτουν από την εφαρμογή δύο αλγορίθμων απλοποίησης γραμμών ('pointremove' και 'bendsimplify') και β) ψηφιοποιημένες αναλογικές γραμμές τοπογραφικών χαρτών της ΓΥΣ στις αντίστοιχες κλίμακες.

Η διατριβή περιλαμβάνει εκτός της Εισαγωγής, έξι Κεφάλαια, την Βιβλιογραφία και ένα Παράρτημα.

Στο Κεφάλαιο 1 πραγματοποιείται βιβλιογραφική ανασκόπηση, αναπτύσσεται η θεωρητική αναφορά του αντικειμένου έρευνας της διατριβής και αναλύονται οι στόχοι της έρευνας.

Το Κεφάλαιο 2 αναφέρεται στη μέθοδο κατάτμησης των χαρτογραφικών γραμμών. Στο πρώτο μέρος, περιγράφεται το θεωρητικό πλαίσιο της τεχνικής, θεσπίζονται οι κατηγορίες των  $\epsilon$ -τμημάτων και ορίζεται η διάμετρος εφαρμογής  $\epsilon$ . Στο δεύτερο



μέρος περιγράφεται η υλοποίηση της τεχνικής σε περιβάλλον ηλεκτρονικού υπολογιστή.

Στο Κεφάλαιο 3 αναφέρονται μέθοδοι επεξεργασίας των χαρτογραφικών γραμμών και περιλαμβάνει δύο μέρη. Το πρώτο σχετίζεται με τη μέθοδο εντοπισμού κρίσιμων σημείων. Ορίζεται ο δείκτης αναλογίας μήκους LR και θεσπίζονται οι παράμετροι λειτουργίας του (μέγεθος ακτίνας κύκλου R, ταξινόμηση τιμών και ορισμός ορίου επιλογής, δείκτες LLR και ALR). Στη συνέχεια, γίνεται έλεγχος της λειτουργικότητας της μεθόδου και σύγκρισή της με σχετικές μελέτες. Στο δεύτερο μέρος του Κεφαλαίου 3 περιγράφονται τα 'εργαλεία' επεξεργασίας των γραμμών (τελεστής εξομάλυνσης Gauss, μέθοδος εντοπισμού καμπών, ομοπαράλληλος μετασχηματισμός, αλγόριθμοι Balloon και 'bendsimplify') και ο τρόπος με τον οποίο αυτά έχουν προσαρμοσθεί, ώστε να ενσωματωθούν στο μοντέλο γενίκευσης γραμμών.

Το Κεφάλαιο 4 αναφέρεται στη γενίκευση των  $\epsilon$ -τμημάτων που προκύπτουν από τη μέθοδο κατάτμησης των χαρτογραφικών γραμμών. Αναπτύσσεται η μέθοδος και περιγράφονται οι τελεστές γενίκευσης που εφαρμόζονται σε κάθε μία από τις τέσσερις κατηγορίες  $\epsilon$ -τμημάτων (μονόπλευρα και αμφίπλευρα  $\epsilon$ -μη-κυστά τμήματα, τμήματα σύγκλισης και  $\epsilon$ -κυστά τμήματα). Στη συνέχεια, περιγράφονται οι διεργασίες για τη συνένωση των γενικευμένων  $\epsilon$ -τμημάτων ώστε να δημιουργηθεί η τελική γραμμή, καθώς και η μέθοδος ελέγχου ευαναγνωσιμότητας αυτής βάσει των κανόνων που έχουν οριστεί.

Το Κεφάλαιο 5 αναφέρεται στις δύο πιλοτικές εφαρμογές που πραγματοποιήθηκαν στα πλαίσια της διατριβής. Το κεφάλαιο χωρίζεται σε τρία μέρη. Στο πρώτο περιγράφεται το αρχείο αναφοράς. Στο δεύτερο αναπτύσσεται η εφαρμογή του δείκτη αναλογίας μήκους LR στην ακτογραμμή της νήσου Περιστερά κλίμακας 1:50.000 και αναλύονται τα αποτελέσματα. Σε ένα δεύτερο επίπεδο, βάσει των κρίσιμων σημείων που έχουν εντοπισθεί συγκρίνονται δύο αλγόριθμοι απλοποίησης γραμμών (οι 'pointremove' και 'bendsimplify'). Στο τρίτο μέρος περιγράφεται η εφαρμογή του μοντέλου κατάτμησης - γενίκευσης γραμμών στην ακτογραμμή της νήσου Περιστερά κλίμακας 1:50.000, για δύο σενάρια γενίκευσης σε κλίμακες 1:1.000.000 και 1:500.000 και τα αποτελέσματα αναλύονται και αξιολογούνται.

Το 6ο Κεφάλαιο περιέχει τα συμπεράσματα που προκύπτουν από την διατριβή καθώς επίσης επισημαίνονται ανοικτά ζητήματα για μελλοντική έρευνα.

Τέλος στο Παράρτημα παρουσιάζονται οι κώδικες υλοποίησης των 'εργαλείων' επεξεργασίας γραμμικών οντοτήτων και γενίκευσης που έχουν αναπτυχθεί σε γλώσσα προγραμματισμού *Python*.

## **Extended Abstract**

### **Cartographic generalization of natural occurring lines with legibility constraints.**

This thesis focuses on the research field of digital cartographic line generalization, by introducing a comprehensive method for generalizing natural linear entities. It addresses two key concepts of concern of the cartographic community in this field: 1. the generalization of linear entities 'guided' by the conceptual framework 'segmentation - characterization - generalization' and 2. the detection of critical points along cartographic lines.

The thesis is developed at three levels: Firstly, a review of studies around this subject, which have been carried out, is presented. In particular, the concept of generalization of linear entities is introduced and current research trends in the field of cartographic generalization are listed and analyzed. The concept of critical points is introduced and their importance in the generalization process is highlighted. Then, there are presented the methods followed for automation in a digital environment, initially through simplification algorithms. The theoretical framework of operation of the most widespread of them is described and, based on the concept of critical points and the analysis of the relevant studies, an evaluation of them is been conducted. According to bibliography, the inadequacies of simplification algorithms lead researchers to look for new methods of generalizing cartographic lines. One of the innovative ideas being investigated is the identification of homogeneous lines or homogeneous segments along the lines, their classification according to qualitative and quantitative criteria, and their generalization via the relevant operators as appropriate.

At the second level, two methodologies, that address the above defined questions are presented. Their structure and evaluation are developed in the main part of the dissertation.

The first of these methodologies, aims to create a model of digital lines generalization, which is based on the conceptual framework 'segmentation-classification-generalization'. More specifically, a specific method has been designed, according to which each line is divided into homogeneous segments. The criteria for segmentation is the legibility of the lines, which is defined based on cartographic rules according to visual perception. Then, specific parameters of classification of the linear entity's divided segments are introduced (the so called  $\epsilon$ -segments) and four categories of segments (sections) are defined for classification, based on their shape and geometry. At the same time, quantitative metrics have been identified and calculated in order to define them. The generalisation of the line is completed by applying appropriate generalisation operators (merging, simplification, enlargement, etc.) to each part, depending on the section to which it belongs and its quantitative characteristics. For each segment generalization operators (enlargement, simplification, expansion, smoothing etc.) that are considered necessary for their

generalization are proposed and their implementation algorithms are designed. The individual line processing operations are coded, interconnected and been integrated in an appropriate software environment so as to create a unified platform which will implement the techniques. The design of the method is carried out based on its implementation to lines representing physical entities, which, due to their complexity and randomness, are of particular research interest.

The second method suggests a way of detecting the critical points of cartographic lines. The method is based on geometric principles and uses the length ratio index LR as a metric to estimate the slope change along a digital line. The LR index is applied, locally, to each point on the line and is structured in such a way that its value increases according to the change in slope of the line. The local examination is ensured by cutting the line through a circle of constant radius around each point of the line. The peaks of the line at which the index values show local maximum greater than a predetermined threshold are considered as critical points. In this document the structure of the LR indicator is analyzed and its operating parameters are defined.

At the third level, a case study is carried out in order to test the effectiveness of the proposed methodologies. The critical points detection method is initially applied to four lines (three geomorphological and one geometrical), selected from two relevant studies, in order to be tested and evaluated from previously known critical points. The method is then piloted on the 1:50,000 scale coastline of the island of Peristera and the results are analysed. Finally, the LR index is used to compare the results of the implementation in five successive generalization scenarios of two line simplification algorithms ('pointremove' and 'bendsimplify' of ArcGIS software). The 'segmentation-classification-generalization' method is applied to the coastline of the island of Peristera at a scale of 1:50,000, for two generalisation scenarios with derived scales of 1:500,000 and 1:1,000,000. The objective is to evaluate both the individual processes and the final outcome resulting from the implementation of the technique to a random line of random shape and geometry and the scenarios of large scale change. The results are evaluated based on the predefined rules, but they are also compared with a) the results deriving from the implementation of two algorithms for line simplification ('pointremove' and 'bendsimplify') and b) the digitalization of analogue lines of GIS topographic maps at the respective scales.

## ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ

### 1.1 Βιβλιογραφική Ανασκόπηση - Θεωρητική αναφορά

Ένας χάρτης αποτελεί μία υπό κλίμακα αναπαράσταση χωρικών φαινομένων. Ο όρος 'υπό κλίμακα' είναι το στοιχείο εκείνο που χαρακτηρίζει τη χαρτογραφική διαδικασία και ουσιαστικά, καθορίζει (σε συνδυασμό με το θέμα και το αναγνωστικό κοινό του χάρτη) το επίπεδο πληροφορίας που απεικονίζεται. Αν και η 'ένα προς ένα' καταγραφή της πραγματικότητας είναι πρακτικά αδύνατη, η συλλογή πληροφορίας, στις μέρες μας, μέσω διαφόρων μεθόδων (π.χ. μετρήσεις υπαίθρου, αεροφωτογραφίες, δορυφορικές εικόνες, παλαιότεροι χάρτες, χάρτες μεγαλύτερης κλίμακας κτλ.), προσφέρει στον χαρτογράφο ένα πλήθος πρωτογενούς υλικού. Το πρωτογενές αυτό υλικό πρέπει να μετατραπεί σε δεδομένα χρήσιμα για τη σύνθεση του εκάστοτε χάρτη. Παράλληλα, οι σύγχρονες απαιτήσεις και εφαρμογές απαιτούν την παραγωγή χαρτών σε μια ποικιλία δεδομένων και λεπτομερειών.

Οι χαρτογράφοι στην προσπάθειά τους να διαχειριστούν τον τεράστιο όγκο πληροφορίας, αλλά και να καλύψουν τις απαιτήσεις παραγωγής, προσανατολίζονται στη δημιουργία χαρτών μεγάλης κλίμακας, οι οποίοι θα λειτουργούν ως 'πρωτογενές υλικό' για τη σύνθεση παράγωγων χαρτών. Στόχος, δηλαδή, είναι η δημιουργία ορθά σχεδιασμένων χαρτών, οι οποίοι θα αποτελούν μια έγκυρη 'βάση' παραγωγής νέων, σε μικρότερη κλίμακα, που θα εξυπηρετούν τις εκάστοτε ανάγκες. Σε κάθε περίπτωση, γίνεται φανερό πως η χαρτογραφική γενίκευση, δηλαδή, 'η επιλεγμένη και απλοποιημένη αναπαράσταση των λεπτομερειών που είναι κατάλληλες ως προς την κλίμακα και το σκοπό του χάρτη' (ICA 1973), αποτελεί πρωτεύουσας σημασίας λειτουργία στη σύγχρονη χαρτογραφία.

Η επεξεργασία γραμμών θεωρείται το κύριο μέρος της χαρτογραφικής γενίκευσης. Και τούτο, αφού, περίπου, το 80% των δεδομένων που περιέχει ένας χάρτης απεικονίζεται με γραμμικές παραστάσεις (δρόμοι, ακτογραμμές, ποτάμια, σύνορα κτλ.) (Thara 1988a). Η οπτικά αποτελεσματική απεικόνιση γραμμικών φαινομένων μέσω ενός χάρτη απαιτεί πολύπλευρη προσέγγιση, τόσο από θεωρητική, όσο και από πρακτική σκοπιά. Θέτοντας ως βάση τον τρόπο και τις διαδικασίες οπτικής αντίληψης, ο χαρτογράφος, οφείλει να δημιουργήσει μια αναπαράσταση του φαινομένου, λειτουργικά έγκυρη και οπτικά αποδεκτή από τον αναγνώστη.

Η γενίκευση χαρτογραφικών γραμμών θεωρείται ως μία από τις πιο πολύπλοκες χαρτογραφικές διαδικασίες. Σύμφωνα με τους Brassel και Weibel (1988), πρόκειται για μια διανοητική διαδικασία επεξεργασίας της πληροφορίας, η οποία περιλαμβάνει λειτουργίες όπως η σύγκριση, η διάκριση, ο συνδυασμός, η αναγνώριση σχέσεων, η τακτοποίηση, η αφαίρεση ή η απόδοση. Κατά τη χειροκίνητη γενίκευση ο χαρτογράφος επιδιώκει ένα γεωγραφικά ακριβές, αλλά και οπτικά ορθό αποτέλεσμα, λαμβάνοντας υπόψη παράγοντες όπως ο βαθμός αλλαγής κλίμακας, ο σκοπός του χάρτη, αλλά και η μορφή και ο χαρακτήρας της γραμμής. Έτσι, ακολουθώντας μια ολιστική διαδικασία, με τη βοήθεια της οποίας εξετάζεται και αναλύεται η γραμμή τόσο σε τοπικό, όσο και σε γενικό επίπεδο και αξιολογείται το κατά το πόσο η διατήρηση, η τροποποίηση ή η αφαίρεση του κάθε χαρακτηριστικού της θα επηρεάσει τις γύρω θέσεις αλλά και ολόκληρη τη γραμμή.

Η διεργασία αυτή λαμβάνει χώρα συνεχώς και επαναληπτικά κατά την πορεία ολοκλήρωσης της γενίκευσης. Γίνεται φανερό, εκτός των άλλων, πως η χειροκίνητη γενίκευση ενέχει μεγάλη δόση υποκειμενικότητας, αφού εξαρτάται κατά πολύ από τα λογικά και αισθητικά κριτήρια του χαρτογράφου.

Παρόλα αυτά, έρευνες έχουν δείξει πως υπάρχει ένας γενικός κανόνας που καθορίζει τον τρόπο επιλογής των θέσεων που θα διαμορφώσουν τη γενικευμένη γραμμή. Στην κατεύθυνση αυτή η Marino (1979) παρουσίασε μια εμπειρική μελέτη, προσπαθώντας να προσδιορίσει πώς οι χρήστες των χαρτών συλλέγουν θέσεις που διαμορφώνουν τις γενικευμένες γραμμές. Στην έρευνά της, σε ένα σύνολο χαρτογράφων και μη-χαρτογράφων δόθηκαν έξι γραμμές, απεικονίσεις φυσικά ιστάμενων οντοτήτων, διαφορετικού μορφολογικού χαρακτήρα. Ζητήθηκε να επιλέξουν τα σημεία που θεωρούν κατάλληλα και αναγκαία, ώστε οι γραμμές να διατηρούν το χαρακτήρα τους. Η διαδικασία έλαβε χώρα σε τρία επίπεδα γενίκευσης και ο αριθμός των προς διατήρηση σημείων ήταν κάθε φορά συγκεκριμένος. Η στατιστική ανάλυση των αποτελεσμάτων οδήγησε σε τρία βασικά συμπεράσματα: (i) εμφανίζεται ταύτιση απόψεων μεταξύ χαρτογράφων και μη χαρτογράφων, (ii) τα σημεία που επιλέγονται βρίσκονται σε θέσεις μέγιστης μεταβολής της κλίσης της γραμμής και (iii) η διατήρηση των σημείων ιεραρχικά και στις τρεις αλλαγές κλίμακας αναδεικνύει τη σημασία που από τη φύση τους αυτά έχουν στον καθορισμό του χαρακτήρα της γραμμής.

Σε μία μελέτη, η οποία σκοπό έχει τη σύγκριση τριών αλγορίθμων απλοποίησης μέσω μαθηματικών, αλλά και εμπειρικών κριτηρίων, η White (1985) καταλήγει σε αντίστοιχα συμπεράσματα με την Marino. Σημαντικά στοιχεία που προκύπτουν από την έρευνα είναι ότι: (i) Εμφανίζεται μια ιεραρχία κατά τη διαδικασία επιλογής σημείων από τους ερωτηθέντες. Ορισμένα θεωρούνται 'περισσότερο κρίσιμα' ή 'περισσότερο ενδιαφέροντα' από άλλα και επιλέγονται περισσότερο. (ii) Υπάρχουν ελάχιστες διαφορές στα σημεία που επιλέγουν χαρτογράφοι και μη χαρτογράφοι. Οι όποιες διαφορές έχουν να κάνουν με την αντίληψη του παρατηρητή και στην συγκεκριμένη περίπτωση στο ότι οι πρώτοι εντοπίζουν τα ιδιαίτερα χαρακτηριστικά των γραμμών.

Τα αποτελέσματα της έρευνας της Marino επιβεβαιώνουν τις ιδέες του Attneave γύρω από την οπτική αντίληψη. Σε μια μελέτη διευκρίνισης διαφόρων αρχών της οπτικής αντίληψης στο πεδίο της θεωρίας της πληροφορίας, ο Attneave (1954) αναφέρει ότι κατά τη διάρκεια παρατήρησης ενός αντικειμένου, ο ανθρώπινος εγκέφαλος δέχεται ένα τεράστιο αριθμό ερεθισμάτων, τα οποία μεταφέρουν μια αντίστοιχα μεγάλη ποσότητα πληροφορίας. Η πληροφορία αυτή, λόγω του όγκου της είναι αδύνατο να αποθηκευτεί και να χρησιμοποιηθεί, ενώ συγχρόνως, ένα ελάχιστο μόνο μέρος της απαιτείται για την κατανόηση του αντικειμένου ως τη συγκεκριμένη-ξεχωριστή οντότητα. Ο ανθρώπινος εγκέφαλος πραγματοποιεί ένα σύνολο ενεργειών (συνειδητών ή μη), με τις οποίες η περίσσεια διέγερσης απομακρύνεται και η εισερχόμενη πληροφορία περιγράφεται και κωδικοποιείται, ώστε να διατηρηθεί η ελάχιστη απαραίτητη. Η μεταξύ άλλων, βασική αρχή (υπόθεση), βάση της οποίας ολοκληρώνεται η παραπάνω διαδικασία είναι ότι: η πληροφορία συγκεντρώνεται κατά μήκος του περιγράμματος και ειδικότερα, σε εκείνα τα σημεία, όπου η διεύθυνση μεταβάλλεται περισσότερο. Ο Attneave (1954) τονίζει ότι: 'αντικείμενα μπορούν να αναπαρασταθούν με μεγάλη οικονομία και

πιστότητα, αντιγράφοντας τα σημεία, στα οποία το περίγραμμά τους αλλάζει κατεύθυνση κατά το μέγιστο και ενώνοντας αυτά τα σημεία με κατάλληλο τρόπο με μια ευθεία' (Attneave 1954, 185). Τέλος, ο Attneave (1954) καταλήγει στο συμπέρασμα, πως τα σημεία (θέσεις) που διαμορφώνουν το σχήμα ενός γραφικού αντικειμένου, μπορούν να διαχωριστούν ποιοτικά, ανάλογα με την ποσότητα πληροφορίας που παρέχουν στον παρατηρητή. Θέσεις που μεταφέρουν υψηλή ποσότητα πληροφορίας θεωρούνται χαρακτηριστικές και είναι ικανές από μόνες τους να περιγράψουν και να χαρακτηρίσουν το απεικονιζόμενο σχήμα και διαχωρίζονται από τις υπόλοιπες (πλεονάζουσες). Η αρχή αυτή που διατύπωσε ο Attneave βρίσκει εφαρμογή στο πεδίο της χαρτογραφίας, στην βιβλιογραφία της οποίας τα χαρακτηριστικά σημεία συναντώνται ως 'κρίσιμα σημεία'.

Ο Freeman (1978) καταλήγει, στον ορισμό του για τα κρίσιμα σημεία μιας ψηφιακής γραμμής, πως αυτά εντοπίζονται στα: (i) μέγιστα και ελάχιστα της καμπυλότητας, (ii) σημεία τέλους ανοικτών γραμμών, (iii) σημεία τομής, (iv) σημεία επαφής, (v) σημεία καμπής, (vi) ασυνέχειες της καμπυλότητας. Κατά τους Hoffman και Richards (1982): 'Τα κρίσιμα σημεία που έχουν εντοπιστεί με την εύρεση των μέγιστων, των ελάχιστων και των μηδενισμών της καμπυλότητας παραμένουν σταθερά σε μετασχηματισμό στροφής, μετάθεσης και ομοιόμορφης κλίμακας.' Ο Jenks (1981), αναφέρει πως σε κάθε γραμμή υπάρχουν σημεία 'υψηλής σημασίας', τα οποία καθορίζουν τη γεωγραφική διαμόρφωσή της. Στην παραδοχή αυτή στηρίζεται το γεγονός ότι 'ένα αραιό, αλλά προσεκτικά επιλεγμένο σύνολο σημείων μπορεί να χρησιμοποιηθεί στη δημιουργία μιας πιστής αναπαράστασης κάθε γραμμής' (Jenks 1981, 4). Ο Jenks διακρίνει δύο κατηγορίες χαρακτηριστικών σημείων: (i) Σημεία, όπου υπάρχουν σημαντικές οικονομικές, πολιτικές κ.τ.λ. δραστηριότητες. Η επιλογή των σημείων αυτών εξαρτάται από τον σκοπό και το περιεχόμενο του χάρτη. (ii) Σημεία φυσικά, ενδιαφέροντα ή βασικά για τη δομή της γραμμής. Τα σημεία αυτά δίνουν στη γραμμή την ξεχωριστή ή χαρακτηριστική μορφή της. Η θέση τους σχετίζεται με αλλαγές στην κλίση (κατεύθυνση) της γραμμής ή μεγάλους 'κυματισμούς' της.

Η μετάβαση από την αναλογική στην ψηφιακή χαρτογραφία δημιούργησε την ανάγκη, αλλά συγχρόνως και πρόσφορο έδαφος για αυτοματοποίηση της διαδικασίας γενίκευσης. Στην προσπάθεια αυτή, οι χαρτογράφοι καλούνται να διαχειριστούν και να δώσουν λύσεις σε δύο ζητούμενα. Όπως αναφέρθηκε και παραπάνω, κατά τη χειροκίνητη διαδικασία, ο χαρτογράφος εξετάζει τη σημαντικότητα της εκάστοτε θέσης ως προς το σύνολο της γραμμής, αλλά και ως προς τα επιμέρους τμήματά της. Αξιολογεί, δηλαδή, τις πληροφορίες που το εκάστοτε σημείο του παρέχει για τη διατήρηση του βασικού σχήματός της, συναρτήσει των προσδοκώμενων αποτελεσμάτων, του βαθμού γενίκευσης, καθώς και της μορφής αυτής. Η πολυπλοκότητα της παραπάνω διαδικασίας συνιστά σημαντικές δυσκολίες στη δημιουργία ενός μοντέλου αυτοματοποιημένης ολοκλήρωσής της. Σε αυτό έρχεται να προστεθεί και το 'πρόβλημα' της καταγραφής, διαχείρισης και απόδοσης συνεχών χωρικών φαινομένων, όπως είναι οι χαρτογραφικές γραμμές, μέσω απεικονίσεων διανυσματικής (vector) δομής σε ψηφιακό περιβάλλον. Οι χαρακτηριστικές, φυσικές και γεωμετρικές ιδιότητες των γραμμικών οντοτήτων, όπως η συνέχεια, η καμπυλότητα κ.λπ., παύουν να υφίστανται στο περιβάλλον ηλεκτρονικού υπολογιστή. Αυτή η ασυμβατότητα

μεταξύ πραγματικότητας και αναπαράστασης της χαρτογραφικής γραμμής δημιουργεί σημαντικά προβλήματα στην αντίληψή της από τον εκάστοτε χρήστη του χάρτη. Δημιουργείται, λοιπόν, η ανάγκη εύρεσης νέων μεθόδων ανάλυσης και επεξεργασίας των ψηφιακών γραμμών, βασισμένων σε νέα δεδομένα.

Στην ψηφιακή χαρτογραφία, η έρευνα στο πεδίο της γενίκευσης εστιάζει στην εύρεση μεθόδων επεξεργασίας των γραμμών που αποδίδονται μέσω διακριτής δομής αρχείων, τεχνικών που θα προσομοιώνουν τις λειτουργίες της αντίστοιχης χειροκίνητης διαδικασίας, καθώς και μέτρων αποτίμησης της ορθότητας των αποτελεσμάτων. Στη διεθνή βιβλιογραφία συναντάται ένα πλήθος μεθοδολογιών που σχετίζονται με τα παραπάνω θέματα. Η αρχική προσπάθεια αυτοματοποίησης εστιάζει στην απλοποίηση των γραμμικών οντοτήτων. Με τον όρο 'απλοποίηση' νοείται η απαλοιφή σημείων της αρχικής γραμμής, ώστε αυτή να αποδοθεί στην παράγωγη κλίμακα. Έχει δημιουργηθεί ένας μεγάλος αριθμός αλγορίθμων απλοποίησης, η πλειοψηφία των οποίων στηρίζεται στην ανάλυση της δομής των χαρτογραφικών γραμμών μέσω γεωμετρικών κριτηρίων (μήκος, κάθετη απόσταση, γωνιακή μεταβολή, επιφανειακή μετάθεση κ.λπ.), καθώς επίσης και σε δεσμεύσεις που τίθενται από τον χρήστη ανάλογα με το βαθμό απλοποίησης και το σκοπό του χάρτη, για την επιλογή των σημείων που θα αφαιρεθούν και αυτών που θα διατηρηθούν στην απλοποιημένη γραμμή (McMaster 1987, Li 2007). Σύμφωνα με μία κατηγοριοποίηση των αλγορίθμων που παρουσιάζει ο Li (1995), οι περισσότεροι από αυτούς χαρακτηρίζονται ως 'πολυγωνικής προσέγγισης'. Σημαντικότερος και ευρέως εφαρμοζόμενος αλγόριθμος απλοποίησης γραμμών είναι αυτός που έχει προταθεί από τους Douglas και Peucker (1973).

Η χαρτογραφική έρευνα στο αντικείμενο της γενίκευσης των γραμμών έχει στηριχθεί από καιρό στη θεωρία του Attneave, υιοθετώντας την έννοια των κρίσιμων σημείων προκειμένου να αξιολογηθούν ποιοτικά οι διαδικασίες απλοποίησης. Από την άλλη μεριά, η ποσοτική αξιολόγηση οποιουδήποτε αλγορίθμου απλοποίησης γραμμών μπορεί να πραγματοποιηθεί με τη χρησιμοποίηση διάφορων ποσοτικών κριτηρίων, όπως τα μαθηματικά μέτρα που προτείνονται από τον McMaster (1986). Μετά από πλήθος εφαρμογών, αρκετοί αλγόριθμοι θεωρούνται ικανοποιητικοί. Εντούτοις, υπάρχουν ακόμα μερικές ερωτήσεις που αξίζει να διερευνηθούν σε περισσότερο βάθος: Είναι οι δύο γραμμές πριν και μετά την απλοποίηση οπτικά παρόμοιες; Είναι οι παράγωγες γραμμικές απεικονίσεις οπτικά και αισθητικά αποτελεσματικές και αποδεκτές από τους αναγνώστες;

Η εγκυρότητα των αλγορίθμων απλοποίησης αποτελεί θέμα συζήτησης για τους χαρτογράφους. Ένα γενικό συμπέρασμα εστιάζει στο ότι δεν καλύπτουν από μόνοι τους τις απαιτήσεις για την παραγωγή οπτικά αποτελεσματικών χαρτών. Και τούτο οφείλεται στην ίδια τη δομή της λειτουργίας τους, η οποία συνιστά αδυναμίες στην διαδικασία επιλογής σημείων.

Έχει παρατηρηθεί πως οι υπάρχοντες αλγόριθμοι απλοποίησης παρέχουν ικανοποιητικά αποτελέσματα εφαρμοζόμενοι σε ορισμένες γραμμές και μη αποδεκτά σε άλλες. Αλλά ακόμα και από την εφαρμογή τους στην ίδια γραμμή, η ποιότητα της απλοποίησης δεν είναι η ίδια σε όλες τις θέσεις. Αυτό οφείλεται στο γεγονός πως οι γραμμικές οντότητες έχουν ξεχωριστή δομή και χαρακτήρα. Για παράδειγμα, μια γραμμή που αναπαριστά ένα ποταμό ημιτονοειδούς μορφής έχει

διαφορετική δομή από μία που αναπαριστά μια πολύπλοκη ακτογραμμή ή σε μια γραμμή που αναπαριστά ένα δρόμο που ξεκινά από μια πεδινή περιοχή και καταλήγει σε μια ορεινή, η δομή της μεταβάλλεται κατά μήκος της. Πάνω σε αυτό το πρόβλημα, η Buttenfield (1989) τονίζει πως οι χαρτογραφικές γραμμές 'συμπεριφέρονται' διαφορετικά κατά τη διαδικασία γενίκευσης, ανάλογα με τη μορφολογία και το χαρακτήρα τους. Τις διαχωρίζει σε εκείνες των οποίων η δομή μεταβάλλεται με την αλλαγή κλίμακας (εξαρτώμενες κλίμακας) και σε εκείνες που δεν μεταβάλλεται (ανεξάρτητες κλίμακας). Γίνεται, λοιπόν, φανερό πως δεν μπορεί να διατυπωθεί ένας γενικός κανόνας σχετικά με το 'ποια' σημεία πρέπει να επιλέγονται κατά τη διαδικασία απλοποίησης.

Οι περισσότεροι, και οι πλέον έγκυροι αλγόριθμοι απλοποίησης (όπως για παράδειγμα ο Douglas-Peucker), στηρίζονται στην ανάλυση της γεωμετρίας της γραμμής εφαρμόζοντας ένα γεωμετρικό κριτήριο σε όλο το εύρος της. Αυτό έχει ως αποτέλεσμα τη μη αποτίμηση της σημαντικότητας, σε τοπικό επίπεδο, του εκάστοτε σημείου. Κάθε σημείο είναι έντονα συνδεδεμένο με τα γειτονικά του. Έτσι, η διατήρησή του ή μη μπορεί να επηρεάσει τη σημαντικότητα των προηγούμενων ή των επόμενων. Και αυτό μπορεί να επηρεάσει το τελικό σχήμα και την αισθητική και γεωγραφική εγκυρότητα της γενικευμένης γραμμής. Απαιτείται, λοιπόν, ανάλυσή της και επιλογή των προς διατήρηση σημείων μέσω κατάλληλου γεωμετρικού κριτηρίου ή συνδυασμού τέτοιων ανά περίπτωση και όχι μέσω ενός, προκαθορισμένου για το σύνολο της. Αυτό τονίζει σε έρευνα του ο Dutton (1999), ο οποίος αναφέρει πως η ανάλυση μιας γραμμής ως σύνολο για την επιλογή των σημείων που τη διαμορφώνουν είναι δυνατό να οδηγήσει σε οπτικά μη-αποδεκτά ή ακόμη και λανθασμένα αποτελέσματα. Και τούτο, διότι η οπτική ποιότητα επηρεάζεται από ένα συνδυασμό παραγόντων, όπως ο βαθμός αλλαγής κλίμακας, η πολυπλοκότητα της γραμμής ή το είδος του φαινομένου που απεικονίζεται. Συγχρόνως, τα αποτελέσματα εξαρτώνται τόσο από το γεωμετρικό κριτήριο που χρησιμοποιείται, όσο και από τις δεσμεύσεις που τίθενται από τον χρήστη. Για παράδειγμα, η επιλογή του αρχικού σημείου, βάση του οποίου εξετάζεται η γραμμή ή η τιμή της παραμέτρου (ανοχή) που εφαρμόζεται στο γεωμετρικό κριτήριο διαφοροποιούν τα σημεία που κάθε φορά εντοπίζονται. Παράλληλα, η παρέμβαση του χρήστη μέσω του ορισμού της παραμέτρου περιορίζει την έννοια της αντικειμενικότητας που επιδιώκεται μέσω της αυτοματοποίησης της διαδικασίας γενίκευσης.

Ένα γεγονός που αξίζει να επισημανθεί είναι πως πολλοί αλγόριθμοι απλοποίησης έχουν ως αρχή λειτουργίας τους τον εντοπισμό και διατήρηση σημείων σε θέσεις μεγάλης αλλαγής κλίσης της γραμμής. Πρακτικά δηλαδή επιδιώκεται, με τη χρήση γεωμετρικών κριτηρίων και κανόνων, να υλοποιηθεί η θεωρία του Attneave για την επιλογή σημείων. Εκ του αποτελέσματος, οι αλγόριθμοι αυτοί θεωρούνται από τους πλέον έγκυρους. Η πράξη όμως έχει δείξει πως η εφαρμογή ακόμα και του συγκεκριμένου ως κανόνα επιλογής σημείων δεν είναι πάντοτε επαρκής, γεγονός που οφείλεται στους παράγοντες που περιγράφονται στην προηγούμενη παράγραφο. Έτσι, πολλές φορές, σημεία που διατηρούνται επειδή εντοπίζονται σε θέσεις μεγάλης αλλαγής κλίσης είναι πιθανό να υποβαθμίσουν ποιοτικά και αισθητικά το αποτέλεσμα. Σύμφωνα με τον Thara (1988a): 'ορισμένα από τα κρίσιμα σημεία που προκαλούν 'αιχμές' στις απλοποιημένες γραμμές πρέπει να



απαλείφονται, αν αυτές θέλουμε να είναι ομαλές, εναργείς και οπτικά αποδεκτές'. Αυτό τονίζουν και οι Visvalingam και Whyatt (1990), σε έρευνά τους σχετικά με τη εγκυρότητα του αλγορίθμου Douglas-Peucker: 'Τα σημεία που επιλέγονται από τον αλγόριθμο Douglas-Peucker δεν είναι πάντα κρίσιμα. Η χειροκίνητη απλοποίηση λαμβάνει υπόψη τη σχετική σημαντικότητα των χαρακτηριστικών. Αυτό εν μέρει εξαρτάται από το σκοπό του χάρτη' (Visvalingam and Whyatt 1990, 224). Επιπλέον, έχοντας ως λειτουργική αρχή τον εντοπισμό των θέσεων μεγάλης αλλαγής κλίσης μιας γραμμής, αρκετοί αλγόριθμοι δεν επιλέγουν σημεία που βρίσκονται σε ομαλά τμήματα ('σιωπηλά' σημεία), τα οποία μπορεί να είναι χρήσιμα για τη διατήρηση του σχήματος της γραμμής (Dutton, 1999).

Συμπεραίνεται, λοιπόν, πως κάθε σημείο που εντοπίζεται σε μια θέση μέγιστης αλλαγής κλίσης δεν είναι απαραίτητως ένα σημείο που θα πρέπει να διατηρηθεί, και, επιπλέον τα προς διατήρηση σημεία δεν εντοπίζονται αποκλειστικά και μόνο σε αυτές τις θέσεις της γραμμής. Η μηχανική λειτουργία των αλγορίθμων δεν παρέχει τη δυνατότητα επιλεκτικής εφαρμογής των κανόνων εντοπισμού, όπως λαμβάνει χώρα κατά τη χειροκίνητη διαδικασία, ούτε προεπισκόπησης της οπτικής ποιότητας των αποτελεσμάτων. Ένα τέτοιο συμπέρασμα μπορεί να διαφανεί μέσω της έρευνας της White (1985), όπου λιγότερα από τα μισά κρίσιμα σημεία που επιλέχθηκαν από τους συμμετέχοντες στην εμπειρική μελέτη ταυτίζονται με αυτά που διατηρήθηκαν από τον εγκυρότερο (βάση της έρευνας) αλγόριθμο που ελέγχθηκε (Douglas-Peucker).

Η εν γένει αναποτελεσματικότητα των αλγορίθμων απλοποίησης και οι λόγοι από τους οποίους αυτή πηγάζει, άνοιξε νέους ορίζοντες στην έρευνα γύρω από τη γενίκευση. Ένα πρώτο συμπέρασμα είναι πως η απλοποίηση από μόνη της δεν ολοκληρώνει μια διαδικασία γενίκευσης. Αποτελεί έναν από τους τελεστές της. Σύμφωνα με τους McMaster και Shea (1992) και άλλες διεργασίες, όπως η εξομάλυνση, η μεγέθυνση, η ενίσχυση, η συγχώνευση και η μετάθεση ή κατά περίπτωση ο συνδυασμός αυτών, εμπεριέχονται στην έννοια της γενίκευσης. Αποτελούν άλλωστε ενέργειες που λαμβάνουν χώρα και στην αντίστοιχη χειροκίνητη διαδικασία. Η έρευνα, λοιπόν, οφείλει να εστιάσει στη δημιουργία ανάλογων τελεστών ή και μοντέλων γενίκευσης όπου αυτοί συνδυάζονται.

Η έρευνα τις τελευταίες τρεις δεκαετίες πραγματοποιείται κάτω από το εννοιολογικό πλαίσιο της ανάλυσης και του χαρακτηρισμού των γραμμικών οντοτήτων, την απομόνωση ομοιογενών τμημάτων και τελικά, την εφαρμογή κατάλληλων κάθε φορά τελεστών γενίκευσης. Μια σειρά προτάσεων και εφαρμογών παρουσιάζεται στη διεθνή βιβλιογραφία. Η διαδικασία της γενίκευσης προτείνεται να λαμβάνει χώρα σε κάθε τμήμα με τον κατάλληλο τελεστή. Πρώτος διατύπωσε αυτή την ιδέα ο Dutton (1999, 36) καθώς αναφέρει πως 'χωρίζοντας τη γραμμή σε τμήματα, ώστε να είναι κάθε ένα από αυτά περισσότερο ομοιογενές, και εφαρμόζοντας κατάλληλους αλγορίθμους και παραμέτρους σε κάθε ένα τμήμα ανεξάρτητα, τα αποτελέσματα της γενίκευσης θα είναι καλύτερα'.

Οι Wang και Müller (1998) προτείνουν έναν αλγόριθμο γενίκευσης, στον οποίο θεωρούνται τμήματα της γραμμής οι καμπές (bends). Αυτές ορίζονται ως το σύνολο των διαδοχικών κορυφών της γραμμής που έχουν ομόσημη γωνία κάμψης (θετική ή αρνητική) και εμπεριέχονται μεταξύ δύο κορυφών αντίθετου πρόσημου. Σε κάθε καμπή υπολογίζονται οι ποσοτικές ιδιότητες (μέγεθος και σχήμα) και ορίζονται οι

σχέσεις με τις γειτονικές. Με βάση το χαρακτηρισμό αυτό, οι καμπές απαλείφονται ή διατηρούνται, ενώ οι τελευταίες παίρνουν την τελική μορφή τους μετά από εφαρμογή τελεστών απαλοιφής, συνδυασμού και μεγέθυνσης ή συνδυασμού αυτών όπου κρίνεται αναγκαίο.

Ο Qian και οι συνεργάτες του (Qian et al. 2016) προτείνουν την τεχνική ODC (Oblique-Dividing-Curve), στην οποία υλοποιείται η ιδέα της κατάτμησης - απλοποίησης. Σαν τμήματα της γραμμής θεωρούνται οι καμπύλες (curves). Κάθε καμπύλη ορίζεται μεταξύ ενός αρχικού σημείου A και του σημείου επαφής B της εφαπτομένης από το πρώτο στη γραμμή. Η διαδικασία επαναλαμβάνεται με αρχή το σημείο B, έως ότου διαχωρισθεί σε καμπύλες ολόκληρη η γραμμή. Οι προκύπτουσες καμπύλες ταξινομούνται σε κατηγορίες ανάλογα με το σχήμα τους (U-shaped και V-shaped) και το μέγεθός τους (large, small, minimum), βάσει μέτρων που ορίζονται. Η απλοποίησή τους πραγματοποιείται μέσω γνωστών αλγορίθμων (π.χ. Douglas-Peucker), με παραμέτρους εφαρμογής που συνδέονται με τα χαρακτηριστικά τους. Η διαδικασία εφαρμόζεται επαναληπτικά, μέχρι τη δημιουργία της επιθυμητής παράγωγης γραμμής.

Η Plazanet και οι συνεργάτες της (Plazanet et al. 1995) υποστηρίζουν πως για το σχεδιασμό ενός πλήρους εργαλείου γενίκευσης γραμμικών οντοτήτων είναι απαραίτητη η δημιουργία ενός μοντέλου, το οποίο θα κατατέμνει τη γραμμή σε ομοιογενή και χαρακτηρισμένα τμήματα. Για κάθε κατηγορία θα πρέπει να καθορισθεί ο καταλληλότερος αλγόριθμος γενίκευσης. Έτσι, 'επιτυγχάνεται η τελική αρμονία αν η γραμμή γενικευθεί ανά τμήματα' (Plazanet et al. 1995, 305). Πάνω σε αυτή την ιδέα προτείνουν κανόνες χαρακτηρισμού των γραμμών. Αρχικά αναφέρουν πως κάθε γραμμή μπορεί να αναλυθεί σε τρία επίπεδα αντίληψης: το γενικό (αντιστοιχεί σε ολόκληρη τη γραμμή), το ενδιάμεσο (τμήμα της γραμμής) και το τοπικό (καμπή της γραμμής). Στη συνέχεια, προτείνουν τους κυματισμούς, την ομοιογένεια, την πυκνότητα και την πολυπλοκότητα ως κριτήρια χαρακτηρισμού. Αναπτύσσοντας τα κριτήρια αυτά στα τρία επίπεδα αντίληψης, καταλήγουν σε μια μέθοδο κατάτμησης των γραμμών η οποία, υπολογίζοντας και τις γεωμετρικές ιδιότητες των τμημάτων, μετατρέπεται σε ένα μοντέλο προεπεξεργασίας των γραμμών, ώστε αυτές να γενικευθούν σε ένα αυτοματοποιημένο σύστημα.

Στηριζόμενος στην ιδέα κατάτμηση-χαρακτηρισμός-εφαρμογή κατάλληλου τελεστή, ο Mustière (2005) παρουσιάζει μια μεθοδολογία γενίκευσης γραμμικών οντοτήτων που αναπαριστούν οδικό δίκτυο. Μεταξύ άλλων, τονίζει πως θα ήταν χρήσιμο να βρεθούν εργαλεία τυποποίησης της περιγραφής του χώρου (spatial-analysis tools), τα οποία θα υποβοηθούν τις λειτουργίες των αλγορίθμων γενίκευσης. Στη μέθοδό του προτείνεται ένα κατάλληλο εργαλείο, το οποίο χρησιμοποιεί την έννοια της ευαναγνωσιμότητας για να αποκόψει τμήματα (τα ονομάζει καμπές - bends) κατά μήκος των γραμμών. Ορίζει πως μια γραμμή είναι μη-ευανάγνωστη από τη μία ή και από τις δύο πλευρές της όταν ο άξονας της 'είναι μακριά' από τα όρια (εσωτερικά ή εξωτερικά) του συμβόλου απεικόνισής της και καθορίζει εμπειρικά την ανοχή. Με τη μέθοδο αυτή, κάθε γραμμή αποκόπτεται σε τμήματα που ανήκουν σε μία εκ των τριών κατηγοριών: 1) απομονωμένη καμπή, 2) αλληλουχία καμπών και 3) μικρές καμπές που αποτελούν θόρυβο. Τα τμήματα της γραμμής που ανήκουν σε κάθε μία από αυτές μεταβάλλονται διαφορετικά κατά τη διαδικασία γενίκευσης, ώστε να διατηρείται, τόσο η οριζοντιογραφική ακρίβεια, όσο και η ευαναγνωσιμότητα του

τελικού προϊόντος. Ο Mustière (2005) καθορίζει πως: 1) οι απομονωμένες καμπές είναι αναγκαίο να μεγεθύνονται. Για το λόγο αυτό εφαρμόζει τους τελεστές γενίκευσης *MinBreak* και *MaxBreak*. 2) σε περίπτωση αλληλουχίας καμπών, αυτή θα πρέπει είτε να μεγεθύνεται, είτε να της αφαιρούνται καμπές, μέσω των τελεστών *Accordion* και *Schematization*, ενώ, 3) οι μικρές καμπές θα πρέπει να εξομαλύνονται μέσω φίλτρων εξομάλυνσης *Gauss* ή *Plaster*. Η μέθοδος γενίκευσης χαρτογραφικών γραμμών ονομάζεται *GALBE* και αποτελεί ένα χαρακτηριστικό παράδειγμα καθολικού μοντέλου γενίκευσης, βασισμένο στο εννοιολογικό πλαίσιο της σύγχρονης χαρτογραφικής έρευνας.

Το μοντέλο που παρουσιάζεται από τον Mustière (2005) έχει τις βάσεις του στο σύστημα γενίκευσης *PlaGe* (Lecordix et al. 1997). Στόχος της είναι η δημιουργία ενός διαδραστικού συστήματος, το οποίο παρέχει μεθόδους προεπεξεργασίας και ανάλυσης των γραμμών και κωδικοποιημένου τελεστές γενίκευσης, σε ένα περιβάλλον άμεσης πρόσβασης και οπτικοποίησης των εφαρμογών αυτών, ώστε ο χρήστης να παρεμβαίνει όποτε κρίνεται απαραίτητο. Αρχικά, εφαρμόστηκαν μέσω του συστήματος γνωστοί αλγόριθμοι απλοποίησης σε γραμμές που αναπαριστούν οδικό δίκτυο. Αξιολογώντας τα αποτελέσματα κατέληξαν στο συμπέρασμα πως υπάρχουν θέσεις όπου απαιτείται μεγέθυνση ή σμίκρυνση της παράγωγης γραμμής, όπως και θέσεις όπου η πολυπλοκότητα επιβάλλει την εφαρμογή μιας σειράς αλγορίθμων και όχι μόνον ενός, διεργασίες που δεν καλύπτουν οι υπάρχοντες αλγόριθμοι. Καταλήγουν πως είναι απαραίτητη η κατάτμηση της γραμμής και η εφαρμογή διαφορετικών αλγορίθμων ή διαφορετικών τιμών ανοχών σε κάθε τμήμα. Για την κάλυψη των παραπάνω αναγκών η ομάδα του Lecordix (Lecordix et al. 1997) εισάγουν στο σύστημα μια μέθοδο κατάτμησης γραμμών που στηρίζεται στα σημεία καμπής (*inflection points*) και μια σειρά τελεστών γενίκευσης (αλγόριθμοι *Lowe*, *Accordion*, *Balloon* και *Schematization*) που καλύπτουν τις διεργασίες μεγέθυνσης και σμίκρυνσης μεμονωμένων ή αλληλουχίας καμπών. Η τεχνική του συστήματος *PlaGe* περιλαμβάνει την επαναλαμβανόμενη κατάτμηση των γραμμών μέχρι να δημιουργηθούν ομοιογενή τμήματα, το χαρακτηρισμό και την ταξινόμηση αυτών μέσω των ποσοτικών τους ιδιοτήτων που υπολογίζονται και την εφαρμογή των απαραίτητων τελεστών γενίκευσης. Σε κάθε βήμα ο χρήστης μπορεί να παρεμβαίνει, ελέγχοντας και βελτιστοποιώντας το τελικό αποτέλεσμα. Θα μπορούσε να ειπωθεί πως σε ένα τέτοιο σύστημα, η υποκειμενική παρέμβαση του χρήστη δε συνάδει με την αντικειμενικότητα που αναμένεται στην αυτοματοποιημένη γενίκευση. Στην πραγματικότητα, όμως, συμβαίνει το εντελώς αντίθετο. Ο χρήστης λαμβάνει αποφάσεις βάσει πληροφοριών που προκύπτουν μετά από επεξεργασία της γραμμής μέσω σειράς αντικειμενικών (γεωμετρικών, μαθηματικών κτλ.) κριτηρίων, που ουσιαστικά τον 'καθοδηγούν' στη λήψη των καλύτερων αποφάσεων. Οι χαρτογράφοι έχουν καταλήξει πως η βέλτιστη λύση στον τομέα της ψηφιακής γενίκευσης μπορεί να δοθεί μέσω διαδραστικών συστημάτων, στα οποία ο υπολογιστής θα δίνει πληροφορίες και ο άνθρωπος θα παίρνει τις τελικές αποφάσεις.

Μια ολοκληρωμένη εφαρμογή γενίκευσης τεχνικών χαρτογραφικών γραμμών βασισμένη στην τεχνολογία των πολλαπλών ευφυών διαμεσολαβητών (*Multi-Agent Systems*) αναπτύχθηκε στο πλαίσιο του ερευνητικού προγράμματος *Agent Project* (*Automatic Generalization New Technology*) (Ruas and Duchêne 2007). Σύμφωνα με

τους Russell και Norvig (2003), ευφυής διαμεσολαβητής (agent) είναι οποιαδήποτε οντότητα, η οποία μπορεί να αντιληφθεί το περιβάλλον της μέσω αισθητήρων και να ενεργήσει σε αυτό μέσω επιδραστών. Θεωρείται ότι κάθε ορθολογική οντότητα ευφυή διαμεσολαβητή μπορεί να αντιληφθεί τις πράξεις της, οι οποίες λογίζονται ως ορθές. Ως 'ορθή' πράξη ορίζεται αυτή που προξενεί τα βέλτιστα αποτελέσματα. Στο πλαίσιο της χαρτογραφίας, κάθε χαρτογραφική οντότητα είναι αυτόνομη και αποτελεί έναν ευφυή διαμεσολαβητή. Χαρακτηρίζεται από περιορισμούς (constraints) που περιγράφουν τις επιβεβλημένες τιμές ποσοτικών χαρακτηριστικών της (μέγεθος, απόσταση, πυκνότητα κ.λπ.), οι οποίοι απαιτούνται για την αναπαράστασή της. Σε περίπτωση που αυτές οι τιμές δεν είναι κατάλληλες, η οντότητα χρειάζεται να γενικευθεί. Η διαδικασία λαμβάνει χώρα επαναληπτικά, εφαρμόζοντας τον κατάλληλο, μεταξύ ενός πλήθους τελεστών γενίκευσης. Κάθε φορά η ποιότητα του αποτελέσματος αξιολογείται, μέχρι να βρεθεί η βέλτιστη λύση, μέχρι δηλαδή, να καλυφθούν οι περιορισμοί που χαρακτηρίζουν την οντότητα. Οι Ruas και Duchêne (2007), ορίζουν τρεις κατηγορίες οντοτήτων: 1) τις οντότητες ευφυείς μικρο-διαμεσολαβητές (micro-agent), οι οποίες είναι υπεύθυνες αποκλειστικά για τη δική τους γενίκευση (π.χ. ένας δρόμος), 2) τις οντότητες ευφυείς μέσο-διαμεσολαβητές (meso-agent), οι οποίοι επηρεάζουν τη γενίκευση μιας ομάδας ευφυών διαμεσολαβητών (π.χ. οδικό δίκτυο) και 3) τις οντότητες ευφυών μακρο-διαμεσολαβητών (macro-agent) που ελέγχουν ή επηρεάζουν τη γενίκευση ενός μεγάλου αριθμού ευφυών διαμεσολαβητών. Στις δύο τελευταίες περιπτώσεις, οι περιορισμοί (constraints) κάθε οντότητας εμπεριέχουν περιορισμούς και άλλων οντοτήτων, με αποτέλεσμα η διαδικασία να γίνεται αρκετά πιο πολύπλοκη. Η εφαρμογή *Agent Project*, που συνοπτικά περιγράφηκε, είναι πρωτοποριακή, η έρευνα για την εξέλιξή της συνεχίζεται και είναι σχεδιασμένη, ώστε να καλύπτει τις ανάγκες γενίκευσης ενός οδικού δικτύου.

---

## 1.2 Στόχος της διατριβής

Στις μέρες μας, είναι κοινώς αποδεκτό στη χαρτογραφική κοινότητα, πως οι αλγόριθμοι απλοποίησης που συνήθως εφαρμόζονται για τη γενίκευση ψηφιακών γραμμών, δεν καλύπτουν τις απαιτήσεις παραγωγής χαρτογραφικά και αισθητικά ορθών προϊόντων, αφού η δομή και η λειτουργία τους δεν υποστηρίζει μία τόσο σύνθετη και ολιστική διεργασία. Αντικείμενο έρευνας, πλέον, για τους χαρτογράφους είναι η δημιουργία αυτοματοποιημένων, καθολικών συστημάτων γενίκευσης, τα οποία θα προσομοιώνουν την αντίστοιχη χειροκίνητη διαδικασία. Για το λόγο αυτό, οι χαρτογράφοι επιδιώκουν να καταγράψουν τους κανόνες που διέπουν τη γενίκευση, να τους παραμετροποιήσουν και να δημιουργήσουν εργαλεία υλοποίησής τους, πάντα υπό τις δεσμεύσεις που θέτει η ψηφιακή αναπαράσταση των γραμμών.

Δύο από τα ζητούμενα που έχουν ανακύψει αναφορικά με την παραπάνω ερευνητική τάση είναι η εύρεση μεθόδων ή τεχνικών:

1. εντοπισμού ομοιογενών γραμμών ή ομοιογενών τμημάτων κατά μήκος των γραμμών, ο χαρακτηρισμός αυτών μέσω ποιοτικών και ποσοτικών κριτηρίων και η γενίκευσή τους μέσω κατάλληλων τελεστών κατά περίπτωση και
2. εντοπισμού κρίσιμων σημείων κατά μήκος χαρτογραφικών γραμμών.

Στην παρούσα έρευνα προτείνονται δύο μεθοδολογίες οι οποίες έχουν ως στόχο να επιλύσουν τα δύο παραπάνω ζητούμενα.

Η πρώτη αφορά στη δημιουργία μιας μεθόδου γενίκευσης ψηφιακών χαρτογραφικών γραμμών, η οποία στηρίζεται στο εννοιολογικό πλαίσιο κατάτμηση-χαρακτηρισμός-γενίκευση. Πιο συγκεκριμένα, σχεδιάζεται μία μεθοδολογία κατά την οποία η εκάστοτε γραμμή κατατέμνεται σε ομοιογενή τμήματα. Ως κριτήριο κατάτμησης χρησιμοποιείται η ευαναγνωσιμότητα των γραμμών, η οποία ορίζεται μέσω χαρτογραφικών κανόνων που στηρίζονται στην οπτική αντίληψη, ενώ η υλοποίηση της διαδικασίας βασίζεται θεωρητικά, στην ιδέα του κυλιόμενου δίσκου που έχει επινοηθεί από τον Perkal (1966a). Τα δημιουργούμενα τμήματα κατηγοριοποιούνται με βάση τη γεωμετρία τους και χαρακτηρίζονται μέσω ποσοτικών κριτηρίων. Η γενίκευση της γραμμής ολοκληρώνεται με την εφαρμογή κατάλληλων τελεστών γενίκευσης (διεύρυνση, απλοποίηση, διόγκωση, εξομάλυνση κτλ.) σε κάθε τμήμα, ανάλογα με την κατηγορία στην οποία ανήκει και τα ποσοτικά χαρακτηριστικά του.

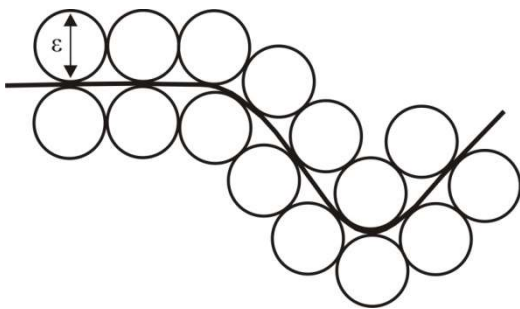
Η μέθοδος αναπτύσσεται με σκοπό την εφαρμογή της σε γραμμές που αναπαριστούν φυσικές οντότητες (π.χ. ακτογραμμές, ποτάμια κ.λπ.). Χαρακτηριστικό των εν λόγω γραμμών είναι η πολυπλοκότητα στη δομή τους και το μη προβλέψιμο του σχήματός τους. Για παράδειγμα, μια ακτογραμμή είναι δυνατό να περιλαμβάνει μια σειρά διαδοχικών κόλπων και χερσονήσων, διαφόρων μεγεθών και διαμορφώσεων ή ένας ποταμός να έχει τυχαίο σχηματισμό, ανάλογα με το ανάγλυφο του εδάφους στο οποίο ρέει. Αντιθέτως, η μορφή των γραμμών που αναπαριστούν τεχνητά φαινόμενα (π.χ. δρόμοι), συνήθως χαρακτηρίζεται από ομαλότητα και συμμετρία λόγω του τρόπου κατασκευής τους. Τα χαρακτηριστικά των φυσικών γραμμών μετατρέπουν την ανάλυση και επεξεργασία τους σε μια σύνθετη διαδικασία, αφού είναι αρκετά δύσκολη η θέσπιση γενικών κανόνων τυποποίησης της μορφής και της γεωμετρίας τους, καθώς και η δημιουργία εργαλείων γενίκευσής τους, τα οποία συνήθως έχουν ως βάση την ανάλυση λιγότερο πολύπλοκων γραμμών. Η επιλογή τους γίνεται, αφενός για την εγκυρότερη αποτίμηση των εφαρμογών και αφετέρου, διότι αντίστοιχες έρευνες δεν συναντώνται στη διεθνή βιβλιογραφία.

Στο στάδιο σχεδιασμού της μεθόδου, ορίζονται και τεκμηριώνονται θεωρητικά όσοι παράμετροι (κριτήρια κατάτμησης, χαρακτηρισμού και τελεστές γενίκευσης) απαιτούνται για την εφαρμογή της προτεινόμενης τεχνικής. Στο στάδιο ανάπτυξης, λαμβάνει χώρα η αυτοματοποίηση των διαδικασιών σε ψηφιακό περιβάλλον. Στο λογισμικό πακέτο διαχείρισης χωρικών δεδομένων *ArcGIS* δομείται ένα μοντέλο κατάτμησης γραμμών και υπολογισμού ποσοτικών χαρακτηριστικών και κατηγοριοποίησης των προκύπτων τμημάτων. Παράλληλα, σε γλώσσα προγραμματισμού *Python*, κωδικοποιούνται οι ρουτίνες μέσω των οποίων υλοποιούνται οι τελεστές γενίκευσης. Τέλος, τα τρία στάδια επεξεργασίας αλληλοσυνδέονται, ώστε να προκύψει μία ενιαία λειτουργία που θα είναι δυνατό να εκτελεσθεί σε οποιαδήποτε γραμμική οντότητα. Η προτεινόμενη μεθοδολογία εφαρμόζεται πιλοτικά στην ακτογραμμή της νήσου Περιστεράς. Σκοπός είναι η αξιολόγηση των αποτελεσμάτων των επιμέρους διεργασιών, αλλά και του τελικού προϊόντος που προκύπτει από την τεχνική.

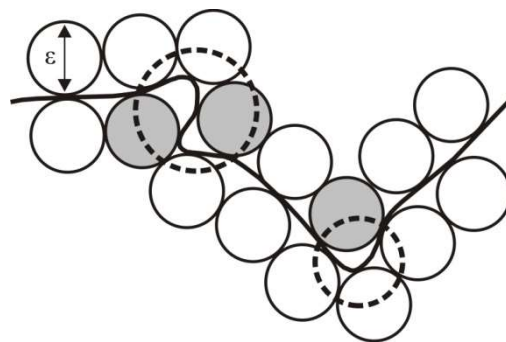
Η δεύτερη μέθοδος που προτείνεται στην παρούσα διατριβή αφορά τον εντοπισμό κρίσιμων σημείων κατά μήκος χαρτογραφικών γραμμών. Η μέθοδος είναι βασισμένη σε γεωμετρικές αρχές και χρησιμοποιεί τον δείκτη αναλογίας μήκους LR ως μέτρο εκτίμησης της αλλαγής κλίσης κατά μήκος μιας ψηφιακής γραμμής. Ο δείκτης LR εφαρμόζεται, τοπικά, σε κάθε σημείο της γραμμής και είναι δομημένος με τρόπο, ώστε η τιμή του να αυξάνει ανάλογα με την κλίση της. Η εξέταση σε τοπικό επίπεδο διασφαλίζεται με την αποκοπή της γραμμής μέσω ενός κύκλου σταθερής ακτίνας γύρω από κάθε της σημείο. Οι κορυφές της γραμμής, στις οποίες οι τιμές του δείκτη εμφανίζουν τοπικά μέγιστα, μεγαλύτερα από ένα προκαθορισμένο όριο, θεωρούνται ως κρίσιμα σημεία. Στο παρόν κείμενο αναλύεται η δομή του δείκτη LR και καθορίζονται οι παράμετροι λειτουργίας του. Στο πρακτικό στάδιο, η μέθοδος κωδικοποιείται σε γλώσσα προγραμματισμού Python και εφαρμόζεται σε τέσσερις γραμμές των οποίων τα κρίσιμα σημεία είναι εκ των προτέρων γνωστά, ώστε να αξιολογηθεί η εγκυρότητά της. Επίσης, εφαρμόζεται πιλοτικά στην ακτογραμμή της νήσου Περιστερά, με σκοπό την εκτίμηση της αποτελεσματικότητάς της. Τα αποτελέσματα της παραπάνω εφαρμογής χρησιμοποιούνται ως μέτρο για να συγκριθούν δύο ευρύτερα διαδεδομένοι αλγόριθμοι απλοποίησης (οι 'pointremove' και 'bendsimplify').

## ΚΕΦΑΛΑΙΟ 2: ΜΕΘΟΔΟΣ ΚΑΤΑΤΜΗΣΗΣ ΧΑΡΤΟΓΡΑΦΙΚΩΝ ΓΡΑΜΜΩΝ

Στο κεφάλαιο αυτό παρουσιάζεται η μέθοδος κατάτμησης μιας γραμμής σε τμήματα που έχουν κοινά χαρακτηριστικά. Η δομή της μεθόδου βασίζεται στην ιδέα της ‘ $\epsilon$ -περιοχής’, που εισήχθη από τον Πολωνό μαθηματικό Julian Perkal (1966a), σε έρευνά του σχετική με την εύρεση μεθόδων μέτρησης του αναπτύγματος αναλογικών γραμμών. Ο Perkal ορίζει την ‘ $\epsilon$ -περιοχή’ ως ‘το γεωμετρικό τόπο των σημείων του επιπέδου που απέχουν απόσταση μικρότερη κατά  $\epsilon$  από τη γραμμή’ (Perkal 1966a, 8). Θεωρητικά, η ‘ $\epsilon$ -περιοχή’ γύρω από μία γραμμή στο επίπεδο δημιουργείται όταν ένας δίσκος διαμέτρου  $\epsilon$  κυλά κατά μήκος της και από τις δύο πλευρές της στο επίπεδο. Το μέγεθος της περιοχής εξαρτάται από το μέγεθος του  $\epsilon$ . Επεκτείνοντας την ιδέα της ‘ $\epsilon$ -περιοχής’, ο Perkal εισάγει την έννοια της  $\epsilon$ -κυρτότητας, βάσει της οποίας χωρίζει τις γραμμές (ή μέρη των γραμμών) σε  $\epsilon$ -κυρτά και  $\epsilon$ -μη-κυρτά τμήματα. Ένα τμήμα μιας γραμμής είναι  $\epsilon$ -κυρτό ‘εάν δίσκος διαμέτρου  $\epsilon$  μπορεί να εφάπτεται και στις δύο πλευρές αυτής’ (Perkal 1966a, 9) (Σχήμα 2.1). Αντίθετα, όταν υπάρχουν τμήματα στα οποία ο δίσκος δεν εφάπτεται, τότε τα συγκεκριμένα τμήματα της γραμμής είναι  $\epsilon$ -μη-κυρτά (τμήματα εντός διακεκομμένου κύκλου στο Σχήμα 2.2).



**Σχήμα 2.1:** Παράδειγμα  $\epsilon$ -κυρτού τμήματος γραμμής.



**Σχήμα 2.2:** Παράδειγμα  $\epsilon$ -μη-κυρτού τμήματος γραμμής.

Όπως φαίνεται στο Σχήμα 2.2, τα  $\epsilon$ -μη-κυρτά τμήματα είναι λιγότερο ομαλά, καθώς αποτελούν έντονες ανωμαλίες του σχήματος της γραμμής και παρουσιάζουν μεγαλύτερη πολυπλοκότητα σε σχέση με τα  $\epsilon$ -κυρτά. Έτσι, σε μια πιθανή διαδικασία γενίκευσης χρήζουν ιδιαίτερης επεξεργασίας. Αυτό τονίζει σε μελέτη της και η Beard (1991), σύμφωνα με την οποία ‘η έλλειψη  $\epsilon$ -κυρτότητας σε μια γραμμή αποτελεί μια άμεση απόδειξη ότι αυτή δεν είναι ευαναγνώστη και ότι απαιτείται να γενικευθεί’ (Beard 1991, 9). Άλλωστε, και ο Perkal χρησιμοποιεί την ιδέα της  $\epsilon$ -κυρτότητας σε μια μέθοδο γενίκευσης, όπως την παρουσίασε σε μεταγενέστερη μελέτη του (Perkal 1966b). Σύμφωνα με αυτή, γύρω από κάθε κλειστή γραμμή ορίζονται δύο πεδία, ένα εσωτερικό ( $D$ ) και ένα εξωτερικό ( $D'$ ). Κατά τη διαδικασία γενίκευσης ( $\epsilon$ -γενίκευση, όπως ονομάζεται), ένας δίσκος διαμέτρου  $\epsilon$ , κυλάει, ξεχωριστά, στην εσωτερική και στην εξωτερική πλευρά της γραμμής. ‘Το σύνολο των σημείων που περιλαμβάνονται πλήρως στο πεδίο  $D$  και έχουν την ιδιότητα να περιέχονται μέσα στους δίσκους διαμέτρου  $\epsilon$ , ορίζονται ως  $\epsilon$ -γενίκευση του πεδίου  $D'$  (Perkal 1966b, 4). Αντίστοιχα ισχύουν για το πεδίο  $D'$ . Με άλλα λόγια, με την κύλιση του δίσκου εντοπίζονται τα  $\epsilon$ -

κυρτά τμήματα της γραμμής, τα οποία και διατηρούνται. Αντίθετα, τα  $\epsilon$ -μη-κυρτά τμήματα, απαλείφονται και αντικαθίστανται από το τμήμα του δίσκου που ενώνει τα σημεία επαφής. Παρόλο που οι μελέτες του Perkal σχετίζονται με την επεξεργασία αναλογικών γραμμών, οι έννοιες που εισάγει αποτελούν χρήσιμο εργαλείο στη σύγχρονη, ψηφιακή χαρτογραφία.

Η ιδέα της  $\epsilon$ -κυρτότητας χρησιμοποιείται στην παρούσα έρευνα ως θεωρητικό πλαίσιο για την κατάτμηση των ψηφιακών χαρτογραφικών γραμμών. Θεωρητικά, σε μια διαδικασία αντίστοιχη του Perkal, δίσκος διαμέτρου  $\epsilon$  κυλάει και από τις δύο πλευρές της γραμμής στο επίπεδο, κατατέμνοντάς την σε  $\epsilon$ -τμήματα,  $\epsilon$ -κυρτά και  $\epsilon$ -μη-κυρτά. Το μέγεθος της διαμέτρου  $\epsilon$  ορίζεται βάσει χαρτογραφικών παραμέτρων, οι οποίες σχετίζονται άμεσα με την κλίμακα του χάρτη, το πάχος του συμβόλου και το όριο οπτικής αντίληψης. Επιτυγχάνεται, με τον τρόπο αυτό, ο διαχωρισμός της γραμμής σε τμήματα όπου εμφανίζονται προβλήματα οπτικής ευαναγνωσιμότητας ( $\epsilon$ -μη-κυρτά) και σε τμήματα όπου δεν εμφανίζονται ( $\epsilon$ -κυρτά), στη συγκεκριμένη κλίμακα. Ολοκληρώνεται, έτσι, μία πρώτη επεξεργασία της γραμμής (κατάτμηση και χαρακτηρισμός με βάση ένα κοινό κριτήριο), ώστε να εφαρμοσθούν στη συνέχεια, κατάλληλοι τελεστές κατά τη γενίκευση.

## 2.1 Κατηγορίες $\epsilon$ -τμημάτων

Με την περάτωση της τεχνικής του κυλιόμενου δίσκου του Perkal, η ψηφιακή χαρτογραφική γραμμή κατατέμνεται καθ' όλο το μήκος της, σε  $\epsilon$ -κυρτά και  $\epsilon$ -μη-κυρτά τμήματα. Τα τελευταία, λόγω της πολυπλοκότητάς τους, αναπαριστούν μια ποικιλία γεωμετρικών σχηματισμών, οι οποίοι ομαδοποιούνται βάσει των χαρακτηριστικών τους σε τρεις τυπικές κατηγορίες. Συγκεντρωτικά, λοιπόν, κάθε  $\epsilon$ -τμήμα της γραμμής, ανάλογα με τη μορφή και τον τρόπο δημιουργίας του, ανήκει σε μία εκ των τεσσάρων κατηγοριών:

- κατηγορία Α: Μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα

Τμήματα της γραμμής που η  $\epsilon$ -μη-κυρτότητα αφορά σε μία απλή κύρτωση ή κοίλωση, η οποία εμφανίζεται στη μία (αριστερή ή δεξιά) πλευρά της γραμμής,

- κατηγορία Β: Αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα

Τμήματα της γραμμής που η  $\epsilon$ -μη-κυρτότητα αφορά σε μία επαλληλία κυρτώσεων/κοιλώσεων, οι οποίες εμφανίζονται και στις δύο πλευρές της γραμμής,

- κατηγορία Γ: Τμήματα σύγκλισης

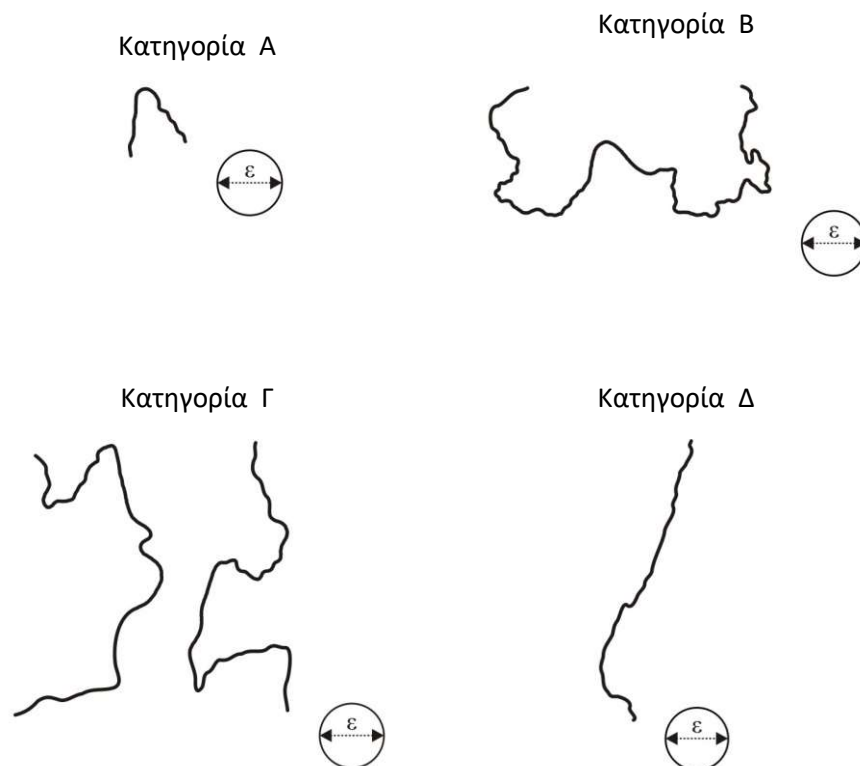
Τμήματα της γραμμής, τα οποία πλησιάζουν μεταξύ τους σε απόσταση μικρότερη από την κρίσιμη απόσταση της οπτικής ευαναγνωσιμότητας εκφρασμένης στην κλίμακα του χάρτη και

- κατηγορία Δ:  $\epsilon$ -κυρτά τμήματα.

Τμήματα της γραμμής που χαρακτηρίζονται εξολοκλήρου ως  $\epsilon$ -κυρτά.



Στο Σχήμα 2.3 παρουσιάζονται χαρακτηριστικά παραδείγματα των τεσσάρων κατηγοριών  $\epsilon$ -τμημάτων που δημιουργούνται κατά την κατάτμηση των ψηφιακών γραμμών μέσω της προτεινόμενης τεχνικής.



**Σχήμα 2.3:** Τυπικά παραδείγματα κατηγοριών  $\epsilon$ -τμημάτων που αποκóπτονται με την προτεινόμενη τεχνική.

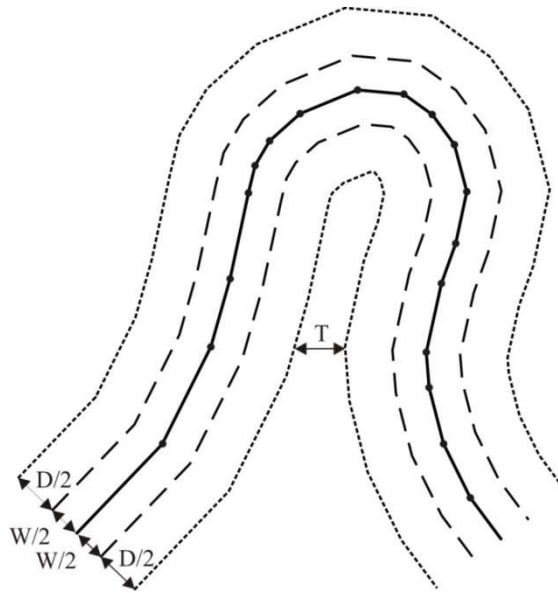
## 2.2 Ορισμός διαμέτρου εφαρμογής $\epsilon$

Ο ορισμός της διαμέτρου  $\epsilon$  του δίσκου που εφαρμόζεται αποτελεί σημαντική μεταβλητή για την προτεινόμενη μεθοδολογία. Είναι αναγκαίο να βασίζεται σε χαρτογραφικούς κανόνες, ώστε να έχει αντικειμενικό χαρακτήρα, να είναι δηλαδή, η διάμετρος του δίσκου ανεξάρτητη της γραμμής και παρέμβασης του χρήστη. Οι παράγοντες που συνεκτιμώνται για το ορισμό της τιμής του  $\epsilon$  είναι το πάχος του χαρτογραφικού συμβόλου της γραμμής, το όριο διακριτικής ικανότητας του ανθρώπινου οφθαλμού (δηλαδή, η ελάχιστη απόσταση μεταξύ δύο συμβόλων, ώστε να γίνονται αντιληπτά ως ξεχωριστές γραφικές οντότητες από το ανθρώπινο μάτι) και μια τιμή ανοχής, ώστε να αποφεύγονται περιπτώσεις συμπτώσεων ή σφαλμάτων στρογγύλευσης του υπολογιστή (Σχήμα 2.4). Η διάμετρος, λοιπόν, του δίσκου, με τρόπο που να εξασφαλίζεται η οπτική ευαναγνωσιμότητα του χάρτη, ορίζεται από τη σχέση:  $\epsilon = W + D + T$

όπου:  $W$ : το πάχος χαρτογραφικής γραμμής

$D$ : το όριο οπτικής αντίληψης

και  $T$ : τιμή ανοχής.



Σχήμα 2.4: Επιλογή διαμέτρου εφαρμογής  $\epsilon$  του δίσκου του Perkal.

## 2.3 Υλοποίηση της τεχνικής του κυλιόμενου δίσκου σε ηλεκτρονικό υπολογιστή

### 2.3.1 Βασική αρχή και περιβάλλον υλοποίησης

Σε περιβάλλον ηλεκτρονικού υπολογιστή, η υλοποίηση της αναλυτικής διαδικασίας του Perkal πραγματοποιείται σε λογισμικό πακέτο συστήματος διαχείρισης χωρικών δεδομένων. Η διαδικασία του κυλιόμενου δίσκου επιτυγχάνεται με τη εφαρμογή του εργαλείου 'ζώνη επιρροής' (buffer) που υποστηρίζεται από το συγκεκριμένο λογισμικό. Η γενική ιδέα της εφαρμογής παρουσιάζεται στο Σχήμα 2.5. Αρχικά, γύρω από κάθε πλευρά της γραμμής (Σχήμα 2.5-συνεχής γραμμή) δημιουργείται μία 'ζώνη επιρροής' (διακεκομμένη γραμμή), με πλάτος ίσο με το μισό της διαμέτρου  $\epsilon$  του δίσκου του Perkal. Στη συνέχεια, μία νέα 'ζώνη επιρροής' με το ίδιο πλάτος, δημιουργείται γύρω από τα όρια της αρχικής (εστιγμένη γραμμή). Τα εσωτερικά όρια της νέας 'ζώνης επιρροής' τέμνουν τη γραμμή σε ορισμένες θέσεις. Οι θέσεις αυτές οριοθετούν τα  $\epsilon$ -μήματα της γραμμής. Αξίζει να σημειωθεί, ότι η μέθοδος που περιγράφεται είναι μια ικανοποιητική προσέγγιση της τεχνικής του Perkal σε ψηφιακό περιβάλλον. Η δεύτερη 'ζώνη επιρροής' που εφαρμόζεται προσομοιώνει τον κυλιόμενο κύκλο και η τομή μεταξύ 'ζώνης επιρροής' και γραμμής αντιστοιχεί στα επαπτόμενα σημεία του δίσκου και της γραμμής, όπως αναφέρεται στην μελέτη του Perkal (1966a).

Με άξονα τη γενική ιδέα, όπως περιγράφεται παραπάνω, επιχειρείται η αυτοματοποίηση της διαδικασίας κατάτμησης χαρτογραφικών γραμμών. Στόχος είναι η δόμηση ενός μοντέλου επεξεργασίας γραμμικών οντοτήτων, στο οποίο η παρέμβαση του χρήστη αφορά μόνο τον αρχικό καθορισμό των παραμέτρων εφαρμογής. Η υλοποίηση πραγματοποιείται στο περιβάλλον του λογισμικού πακέτου ArcGIS® της εταιρίας ESRI, με τη βοήθεια του εργαλείου *Model Builder*. Το *Model Builder* αποτελεί μία εφαρμογή του λογισμικού, η οποία παρέχει τη δυνατότητα δημιουργίας και εκτέλεσης μοντέλων επεξεργασίας χωρικών δεδομένων. Με τον όρο 'μοντέλο' νοείται μια αλληλουχία χωρικών διεργασιών, που υλοποιούνται με τη εφαρμογή κατάλληλων εργαλείων και σκοπό έχουν τη

μετατροπή των αρχικών δεδομένων στα παράγωγα επιθυμητά προϊόντα. Το *Model Builder* επιλέχθηκε, αφού τα εργαλεία που παρέχει καλύπτουν τις απαιτήσεις για τη δημιουργία του μοντέλου κατάτμησης γραμμών. Παράλληλα, υποστηρίζει λειτουργίες που συμβάλλουν στην αυτοματοποίηση, συνολικά, της προτεινόμενης τεχνικής γενίκευσης. Σημαντικότερη είναι η δυνατότητα ενσωμάτωσης στο μοντέλο εργαλείων και ρουτινών, που δομούνται από το χρήστη για την κάλυψη συγκεκριμένων αναγκών (π.χ. τελεστές γενίκευσης γραμμών στη συγκεκριμένη έρευνα) κατά την επεξεργασία των οντοτήτων. Επιπλέον, ο υπολογισμός και η καταγραφή σε βάσεις, ποσοτικών και τοπολογικών πληροφοριών για τις γεωγραφικές οντότητες σε όλα τα στάδια επεξεργασίας, καθώς η άμεση οπτικοποίηση του τελικού, αλλά και των ενδιάμεσων αποτελεσμάτων, αποτελούν ουσιαστικές παροχές για την εύκολη και έγκυρη διαχείριση των δεδομένων. Έτσι, στο περιβάλλον του *Model Builder* καθίσταται εφικτή η δημιουργία ενός ολοκληρωμένου συστήματος επεξεργασίας γραμμικών οντοτήτων.



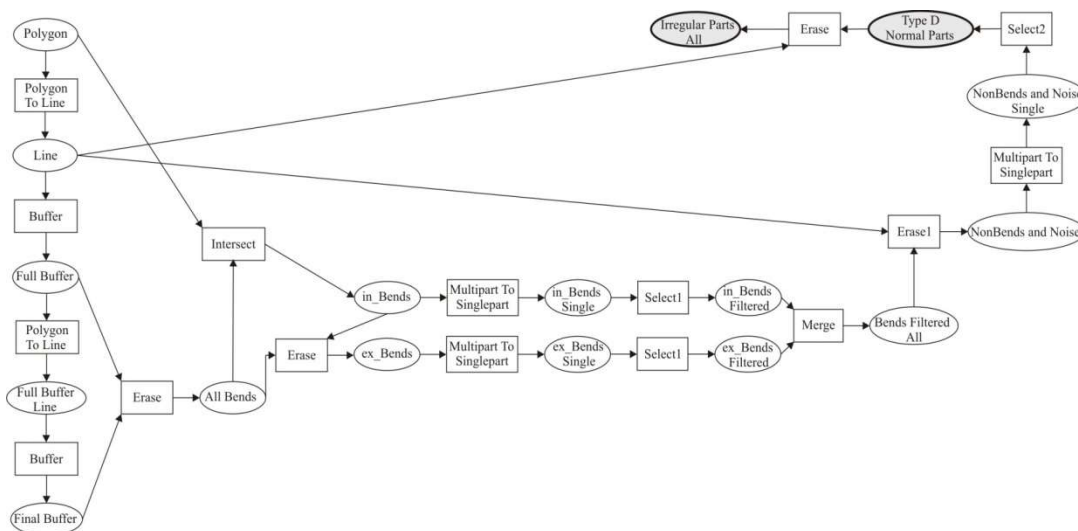
**Σχήμα 2.5:** Παράδειγμα εφαρμογής της τεχνικής του Perkal σε ψηφιακό περιβάλλον.

Για την αυτοματοποίηση της προτεινόμενης μεθόδου κατάτμησης γραμμών δημιουργείται ένα μοντέλο, με το οποίο εκτελείται μια αλληλουχία εργαλείων, κατάλληλα δομημένων, ώστε να λαμβάνεται το επιθυμητό αποτέλεσμα. Στο παρόν κείμενο το μοντέλο παρουσιάζεται τμηματικά, ανάλογα με το στάδιο της έρευνας που περιγράφεται. Αναφορά γίνεται στις πιο σημαντικές διεργασίες που λαμβάνουν χώρα. Στα διαγράμματα ροής που ακολουθούν (Σχήματα 2.6, 2.7, 2.10, 4.2, 4.17, 4.19, 4.21, 4.22, 4.24 και 4.25), οι οντότητες που εισάγονται ή προκύπτουν μετά από μια επιμέρους επεξεργασία αναπαριστώνται με έλλειψη, ενώ τα εργαλεία που χρησιμοποιούνται με ορθογώνιο. Τα βέλη απεικονίζουν τη ροή των πράξεων και τις εκάστοτε οντότητες που συμμετέχουν σε αυτές.

### 2.3.2 Εντοπισμός, φιλτράρισμα και συγχώνευση ε-μη-κυρτών τμημάτων και εντοπισμός ε-κυρτών τμημάτων

Στο Σχήμα 2.6 παρουσιάζεται η διαδικασία εφαρμογής διαδοχικών ‘ζωνών επιρροής’ (*Buffer, Full Buffer, Final Buffer*) εσωτερικά και εξωτερικά της γραμμής, όπως αναφέρεται στη γενική περιγραφή της μεθόδου. Αποτέλεσμα είναι η αποκοπή ε-μη-κυρτών τμημάτων και από τις δύο πλευρές αυτής (*in\_Bends* και *ex\_Bends*, αντίστοιχα).

Ένα μεγάλο μέρος από τα ε-μη-κυρτά τμήματα που δημιουργούνται δεν είναι ορατά στην κλίμακα του χάρτη, αφού έχουν πολύ μικρό μέγεθος και στενό/επίμηκες σχήμα, αποτελώντας, ουσιαστικά, ‘θόρυβο’. Για το λόγο αυτό, πριν από οποιαδήποτε περαιτέρω εφαρμογή, υπόκεινται σε μια διαδικασία φιλτραρίσματος (*Select1*). Για την επίτευξη αυτού, κατά τη διαδικασία κατάτμησης υπολογίζεται το εμβαδό του πολυγώνου που δημιουργείται μεταξύ του ε-τμήματος και του μέρους της ‘ζώνης επιρροής’ που αποκόπτεται μεταξύ της αρχής και του πέρατος αυτού. Το τελευταίο αποτελεί το τόξο του θεωρητικού δίσκου του Perkal που έχει αποκοπεί μεταξύ των σημείων τομής του με τη γραμμή. Τα ε-μη-κυρτά τμήματα με εμβαδό μικρότερο μίας ανοχής, θεωρείται ότι δεν χρήζουν ιδιαίτερης επεξεργασίας, αφού, λόγω μεγέθους, δεν είναι ορατά στην παράγωγη κλίμακα και επομένως, δεν δημιουργούν πρόβλημα ευαναγνωσιμότητας. Τα τμήματα αυτά απομονώνονται, συνενώνονται (*Erase1*) με τα ε-κυρτά και ως τέτοια αντιμετωπίζονται στις επόμενες διεργασίες. Έτσι, με την ολοκλήρωση του συγκεκριμένου σταδίου, κατά μήκος της γραμμής εντοπίζονται διαδοχικά, φιλτραρισμένα ε-μη-κυρτά τμήματα (*Bends Filtered All*) και ‘ενημερωμένα’ ε-κυρτά (*NonBends and Noise*), τα οποία είναι προϊόν συνένωσης των αρχικών και των ε-μη-κυρτών που αποτελούν ‘θόρυβο’.



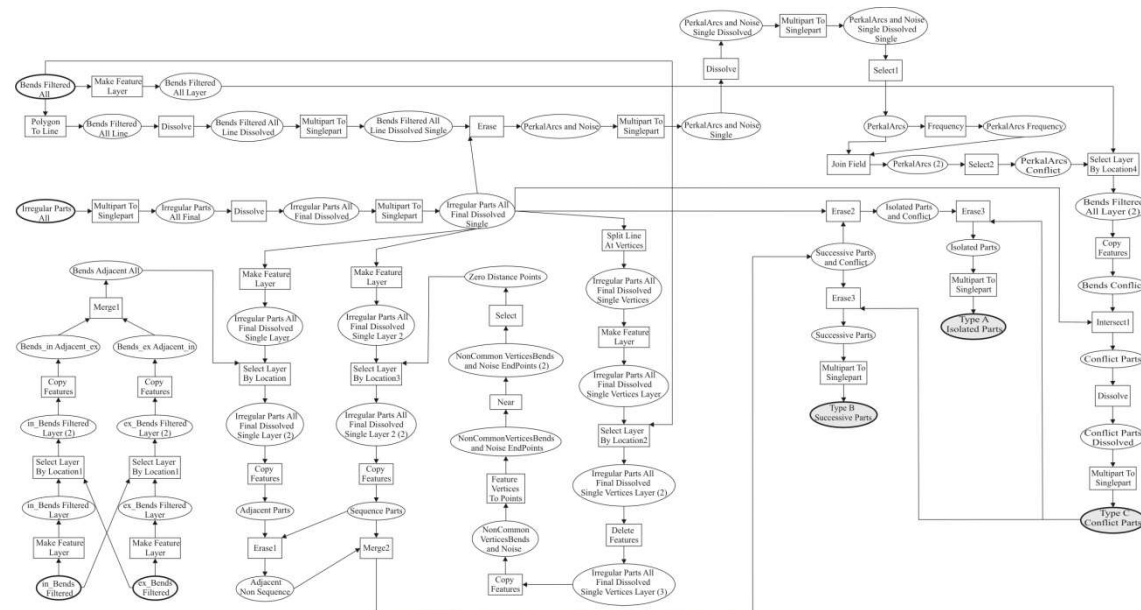
Σχήμα 2.6: Μοντέλο εντοπισμού, φιλτραρίσματος, συγχώνευσης ε-μη-κυρτών τμημάτων και εντοπισμού ε-κυρτών τμημάτων.

Σε επόμενο στάδιο, για την αποφυγή εμφάνισης κατακερματισμένων ε-μη-κυρτών τμημάτων που μεταξύ τους παρεμβάλλονται ε-κυρτά τμήματα με μήκος που

πλησιάζει το μέγεθος του ορίου της οπτικής αντίληψης, εφαρμόζεται μια διαδικασία συγχώνευσης στα φιλτραρισμένα  $\epsilon$ -μη-κυρτά τμήματα. Στόχος είναι,  $\epsilon$ -μη-κυρτά τμήματα που απέχουν μεταξύ τους απόσταση στο όριο της οπτικής αντίληψης, να υπόκεινται σε περαιτέρω επεξεργασία ως ενιαίες οντότητες, αφού ως τέτοιες γίνονται αντιληπτές από τον αναγνώστη του χάρτη. Για την υλοποίηση αυτού, υπολογίζεται και εξετάζεται το ανάπτυγμα των  $\epsilon$ -κυρτών τμημάτων, όπως έχουν δομηθεί μετά τη διαδικασία φιλτραρίσματος. Σε περιπτώσεις που τα εν λόγω  $\epsilon$ -τμήματα είναι οριακά αναγνώσιμα, το μήκος τους αποτελεί μια επαρκή προσέγγιση της απόστασης μεταξύ των  $\epsilon$ -μη-κυρτών τμημάτων. Τα  $\epsilon$ -κυρτά τμήματα των οποίων το ανάπτυγμα είναι μικρότερο ενός κατωφλίου, το οποίο εξαρτάται από το όριο οπτικής αντίληψης (D) και μια τιμή ανοχής (T), ανηγμένα στην κλίμακα του χάρτη, επιλέγονται, διαχωρίζονται (*Select2*) της κατηγορίας και συνενώνονται με τα εκατέρωθεν  $\epsilon$ -μη-κυρτά. Επιτυγχάνεται, έτσι, η συγχώνευση των  $\epsilon$ -μη-κυρτών τμημάτων. Με την ολοκλήρωση της διαδικασίας, η χαρτογραφική γραμμή δομείται από διαδοχικά  $\epsilon$ -μη-κυρτά (στην αρχική τους μορφή ή ως αλληλουχίες μετά τη συγχώνευση) (*Irregular Parts All*) και  $\epsilon$ -κυρτά (εναπομείναντα μετά τη συγχώνευση) (*TypeD Normal Parts*) τμήματα.

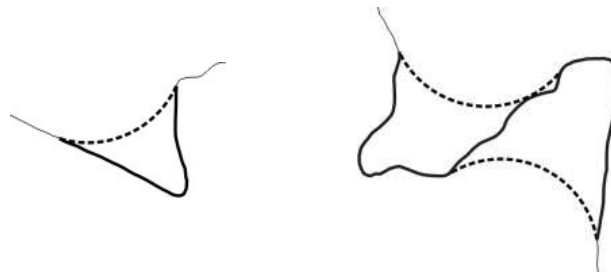
### 2.3.3 Διαχωρισμός $\epsilon$ -μη-κυρτών τμημάτων σε κατηγορίες

Η εφαρμογή της ιδέας του Perkal, με την τεχνική και τους κανόνες που περιγράφονται στην προηγούμενη ενότητα, έχει ως αποτέλεσμα την κατάτμηση των ψηφιακών χαρτογραφικών γραμμών καθ' όλο το μήκος τους. Τα τμήματα που προκύπτουν χαρακτηρίζονται είτε ως  $\epsilon$ -κυρτά, είτε ως  $\epsilon$ -μη-κυρτά. Σε ένα δεύτερο επίπεδο ανάλυσης, τα  $\epsilon$ -μη-κυρτά τμήματα είναι δυνατό να διαχωρισθούν σε τρεις επιμέρους κατηγορίες, ανάλογα με τη μορφή και τον τρόπο δημιουργίας τους. Στο Σχήμα 2.7 παρουσιάζεται η διαδικασία εντοπισμού και διάκρισης αυτών ως ανεξάρτητες οντότητες.



Σχήμα 2.7: Μοντέλο διαχωρισμού  $\epsilon$ -μη-κυρτών τμημάτων σε κατηγορίες.

Από την περιγραφή της μεθόδου κατάτμησης γίνεται αντιληπτό πως ένα μέρος των  $\epsilon$ -μη-κυρτών τμημάτων δημιουργείται από την κύλιση του δίσκου του Perkal εσωτερικά της γραμμής και ένα δεύτερο μέρος, από την αντίστοιχη διαδικασία εξωτερικά αυτής (*in\_Bends Filtered* και *ex\_Bends Filtered*, αντίστοιχα). Τα εσωτερικά και εξωτερικά τμήματα είναι δυνατό να εμφανίζονται απομονωμένα (να γειτνιάζουν, δηλαδή, με  $\epsilon$ -κυρτά τμήματα) ή να επικαλύπτονται σε ένα μέρος τους (Σχήμα 2.8). Το γεγονός αυτό διαφοροποιεί τα  $\epsilon$ -τμήματα ως προς τη μορφή τους, αφού τα πρώτα χαρακτηρίζονται από μία κύρτωση ή κοίλωση, ενώ τα επικαλυπτόμενα, μετά την ένωση τους, εμφανίζονται ως επαλληλία κυρτώσεων και κοιλώσεων. Με βάση την ιδιότητα αυτή πραγματοποιείται ο κύριος διαχωρισμός των  $\epsilon$ -μη-κυρτών τμημάτων. Πιο συγκεκριμένα, τα απομονωμένα έχουν τη δομή μονόπλευρων  $\epsilon$ -τμημάτων (κατηγορία Α), ενώ τα επικαλυπτόμενα αμφίπλευρων (κατηγορία Β). Έτσι, τα επικαλυπτόμενα  $\epsilon$ -τμήματα (*Bends\_in\_Adjacent\_ex* και *Bends\_ex\_Adjacent\_in*) εντοπίζονται (*Select Layer by Location1*), συνενώνονται (*Merge1, Bends Adjacent All*), διαμορφώνονται ως ανεξάρτητες οντότητες (*Adjacent Parts*) και διαχωρίζονται από τα απομονωμένα.



**Σχήμα 2.8:** Παράδειγμα απομονωμένου (αριστερά) και επικαλυπτόμενου (δεξιά) τμήματος.

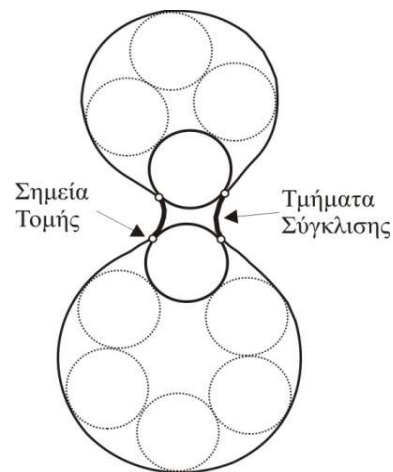
Τα επικαλυπτόμενα και τα απομονωμένα  $\epsilon$ -τμήματα είναι πιθανό να συμμετέχουν στις αλληλουχίες  $\epsilon$ -τμημάτων, που έχουν διαμορφωθεί από τη διαδικασία συγχώνευσης, όπως αυτή περιγράφεται στην προηγούμενη ενότητα. Αναλυτικότερα, οι αλληλουχίες είναι δυνατό να δημιουργηθούν από τη συγχώνευση δύο απομονωμένων, δύο επικαλυπτόμενων, ενός απομονωμένου και ενός επικαλυπτόμενου ή συνδυασμού περισσότερων των δύο  $\epsilon$ -μη-κυρτών τμημάτων και των ενδιάμεσων αυτών  $\epsilon$ -κυρτών τμημάτων. Γίνεται εύκολα αντιληπτό πως, ανεξαρτήτως του τρόπου δημιουργίας τους, οι αλληλουχίες έχουν τη μορφή επαλληλίας κυρτώσεων και κοιλώσεων και ως εκ τούτου, χαρακτηρίζονται ως αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα. Το γεγονός ισχύει και στην περίπτωση που οι αλληλουχίες είναι προϊόν συγχώνευσης δύο ή περισσότερων απομονωμένων  $\epsilon$ -τμημάτων, αφού διαδοχικά τέτοια τμήματα, τα οποία απέχουν απόσταση που προσεγγίζει το όριο του αντιληπτού, έχουν πάντα αντίθετη καμπυλότητα. Συνοψίζοντας λοιπόν, στην κατηγορία των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων (κατηγορία Β) εντάσσονται οι αλληλουχίες  $\epsilon$ -τμημάτων και τα επικαλυπτόμενα που δε μετέχουν σε αυτές, ενώ σε αυτή των μονόπλευρων (κατηγορία Α) τα απομονωμένα  $\epsilon$ -τμήματα που δεν έχουν ενταχθεί σε κάποια αλληλουχία.

Σε ένα δεύτερο επίπεδο επεξεργασίας, ολοκληρώνεται ο διαχωρισμός των  $\epsilon$ -μη-κυρτών τμημάτων σε μονόπλευρα και αμφίπλευρα, σύμφωνα με τις αρχές που αναφέρονται παραπάνω. Για την επίτευξη αυτού, αρχικά, από τα  $\epsilon$ -μη-κυρτά



τμήματα, ανευρίσκονται εκείνα που αποτελούν αλληλουχίες. Τα εν λόγω  $\epsilon$ -τμήματα χαρακτηρίζονται από το γεγονός πως, μεταξύ των επιμέρους τμημάτων που τα δομούν, υπάρχει τουλάχιστον ένα  $\epsilon$ -κυρτό. Η ιδιότητα αυτή αποτελεί κριτήριο για την ανεύρεση τους. Υλοποιείται μια σειρά διεργασιών, κατά τις οποίες από το σύνολο των  $\epsilon$ -μη-κυρτών τμημάτων (*Irregular Parts All*), εντοπίζονται (*Select Layer by Location2*) και διαγράφονται (*Delete Features*) τα απομονωμένα και τα επικαλυπτόμενα  $\epsilon$ -τμήματα (*Bends Filtered All*). Αποτέλεσμα είναι η διατήρηση μόνο των  $\epsilon$ -κυρτών τμημάτων που μετέχουν σε αυτά (*Zero Distance Points*). Έτσι, εκτελώντας μια διαδικασία ελέγχου, εντοπίζονται και διαχωρίζονται από τα υπόλοιπα (*Select Layer by Location3*), τα  $\epsilon$ -μη-κυρτά τμήματα στα οποία ανήκουν τα  $\epsilon$ -κυρτά, δηλαδή, οι αλληλουχίες  $\epsilon$ -τμημάτων (*Sequence Parts*). Εν συνεχεία, όντας γνωστές οι αλληλουχίες, εξετάζεται ποια από τα απομονωμένα και τα επικαλυπτόμενα  $\epsilon$ -τμήματα συμμετέχουν σε αυτές, ώστε να διαγραφούν από τα αντίστοιχα αρχεία. Για το λόγο αυτό, από τα επικαλυπτόμενα  $\epsilon$ -τμήματα (*Adjacent Parts*) αποκόπτονται (*Erase1*) εκείνα που αποτελούν μέρος αλληλουχιών και προκύπτουν τα αμιγώς επικαλυπτόμενα  $\epsilon$ -τμήματα (*Adjacent Non Sequence*). Τα αμιγώς επικαλυπτόμενα  $\epsilon$ -τμήματα και οι αλληλουχίες συνενώνονται (*Merge2*), αφού, όπως αναφέρεται παραπάνω, συνθέτουν την κατηγορία των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων (*Successive Parts and Conflict*). Στα τελευταία είναι πιθανό να περιλαμβάνονται απομονωμένα  $\epsilon$ -τμήματα, ως μέρη αλληλουχιών. Υλοποιώντας μια διαδικασία διαγραφής (*Erase2*) των αμφίπλευρων από τα αρχικά  $\epsilon$ -μη-κυρτά τμήματα (*Irregular Parts All*), διατηρούνται και διαχωρίζονται ως ανεξάρτητες οντότητες τα απομονωμένα που δε συμμετέχουν σε αλληλουχίες, δηλαδή, τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα (*Isolated Parts and Conflict*).

Εκτός των δύο παραπάνω περιπτώσεων, είναι πιθανό να εμφανίζονται θέσεις κατά μήκος της γραμμής, όπου η απόσταση μεταξύ δύο ή περισσότερων απέναντι  $\epsilon$ -μη-κυρτών τμημάτων είναι μικρότερη από το όριο οπτικής ευαναγνωσιμότητας. Οι θέσεις αυτές χρήζουν ειδικής επεξεργασίας κατά τη γενίκευση και για το λόγο αυτό, τα τμήματα που τις διαμορφώνουν θεωρούνται ξεχωριστή κατηγορία. Τα 'τμήματα σύγκλισης', όπως ονομάζονται, εντοπίζονται από την εφαρμογή της τεχνικής του Perkal εσωτερικά της γραμμής. Θεωρητικά, ο δίσκος του Perkal κυλά κατά μήκος της, δημιουργώντας  $\epsilon$ -τμήματα. Αν η γραμμή, σε κάποια θέση της, παρουσιάζει 'στένωση', ο δίσκος σταματά να κυλά, αδυνατώντας να τη διαπεράσει. Είναι φανερό, πως το γεγονός παρατηρείται και στο έτερο μέρος της γραμμής. Τα σημεία όπου αποκόπτεται η κύλιση του δίσκου οριοθετούν τις προβληματικές θέσεις (Σχήμα 2.9). Σε πρακτικό επίπεδο, ο εντοπισμός των τμημάτων που πιθανά ανήκουν στην κατηγορία, πραγματοποιείται με βάση αυτή τη λογική. Ιδιότητα που τα χαρακτηρίζει είναι πως η δημιουργία τους είναι



Σχήμα 2.9: Δημιουργία τμήματος σύγκλισης.

αποτέλεσμα τομής της γραμμής με δύο ή περισσότερες 'ζώνες επιρροής'. Σε όλες τις άλλες περιπτώσεις, κάθε  $\epsilon$ -μη-κυρτό τμήμα προκύπτει από την τομή μίας 'ζώνης επιρροής' με τη γραμμή. Για το λόγο αυτό, εκτελείται μια σειρά διεργασιών, κατά την οποία απομονώνονται (*Select1*) τα τόξα των 'ζωνών επιρροής' μεταξύ των άκρων των  $\epsilon$ -μη-κυρτών τμημάτων (*Perkal Arcs*). Κάθε τόξο αντιστοιχίζεται στο  $\epsilon$ -τμήμα που ανήκει. Στη συνέχεια, πραγματοποιείται έλεγχος (*Frequency, PerkalArcs Frequency*), επιλέγονται (*Select2*) και εντοπίζονται (*Select Layer by Location4*) τα  $\epsilon$ -τμήματα στα οποία αναλογούν δύο ή περισσότερα τόξα (*Bends Conflict*). Τα τμήματα αυτά διαχωρίζονται (*Intersect1*) από τα υπόλοιπα  $\epsilon$ -μη-κυρτά, δημιουργώντας την κατηγορία των τμημάτων σύγκλισης (*Type C Conflict Parts*). Πριν τον εντοπισμό τους, τα εν λόγω  $\epsilon$ -τμήματα, είναι δυνατό να εμφανίζονται απομονωμένα ή να αποτελούν μέρος αλληλουχίας ή επικαλυπτόμενων  $\epsilon$ -τμημάτων. Ενυπάρχουν, δηλαδή, στα μονόπλευρα ή/και στα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα. Έτσι, σε ένα τελευταίο στάδιο επεξεργασίας, πραγματοποιείται η απαλοιφή τους (*Erase3*) από αυτά, ώστε να προκύψουν οι τελικές οντότητες των άλλων δύο κατηγοριών (*Type A Isolated Parts* και *Type B Successive Parts*).

### **2.3.4 Νέα συγχώνευση $\epsilon$ -μη-κυρτών τμημάτων και δημιουργία τελικών $\epsilon$ -τμημάτων**

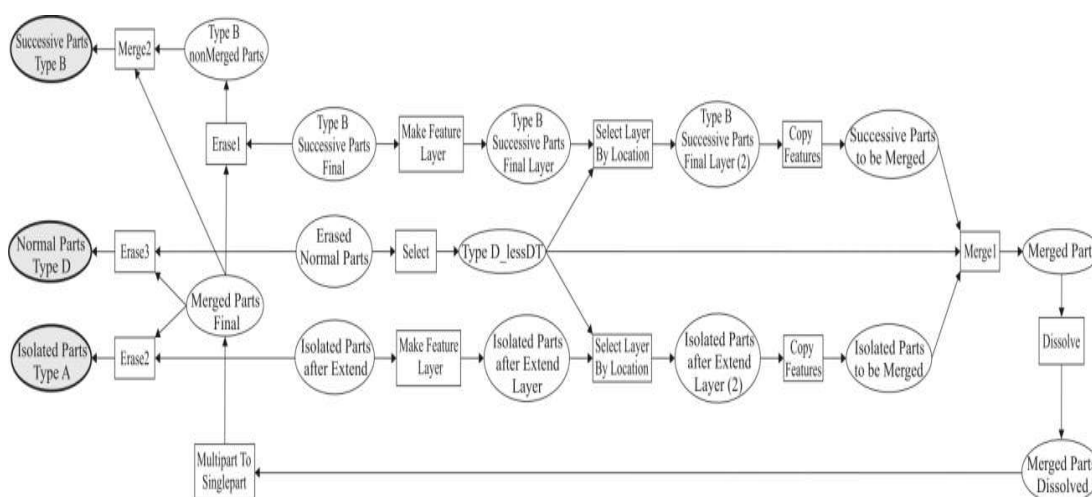
Με την ολοκλήρωση των διεργασιών κατάτμησης, οι χαρτογραφικές γραμμές διαμερίζονται σε  $\epsilon$ -τμήματα, τα οποία διακρίνονται σε μονόπλευρα ή αμφίπλευρα  $\epsilon$ -μη-κυρτά, σύγκλισης και  $\epsilon$ -κυρτά. Σε επόμενα στάδια επεξεργασίας, τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα και τα τμήματα σύγκλισης υπόκεινται σε διαδικασίες διεύρυνσης, στα πλαίσια των μετασχηματισμών που υφίστανται για τη γενίκευσή τους. Αποτέλεσμα αυτών είναι η μεταβολή του αρχικού ανάπτγματος, τόσο των εν λόγω  $\epsilon$ -τμημάτων (αυξάνει), όσο και των γειτονικών τους (μειώνεται). Πιο συγκεκριμένα, τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα διευρύνονται προσαρτώντας μέρη από τα παρακείμενα  $\epsilon$ -κυρτά (βλ. κεφ. 4.1.1), ενώ στη διεύρυνση των τμημάτων σύγκλισης είναι δυνατό να 'συμμετέχουν'  $\epsilon$ -τμήματα που ανήκουν στις υπόλοιπες τρεις κατηγορίες (A, B και Δ) (βλ. κεφ. 4.3.2). Με την ολοκλήρωση της διαδικασίας, είναι πιθανό το ανάπτγμα ορισμένων γειτονικών  $\epsilon$ -τμημάτων να εμφανίζεται μικρότερο ή να προσεγγίζει το μέγεθος του ορίου της οπτικής αντίληψης. Το γεγονός αφορά μόνο σε γειτονικά  $\epsilon$ -κυρτά τμήματα. Δεδομένου πως, και στις δύο περιπτώσεις διεύρυνσης, τα μέρη που αποκόπτονται από τα γειτονικά  $\epsilon$ -τμήματα έχουν αρκετά μικρό μήκος (δεν υπερβαίνει το μισό του ορίου της οπτικής αντίληψης), το φαινόμενο παρατηρείται όταν τα  $\epsilon$ -τμήματα, στην αρχική τους μορφή, είναι οριακά ορατά. Λόγω του τρόπου δημιουργίας τους, κάτι τέτοιο είναι δυνατό να παρατηρηθεί μόνο στα  $\epsilon$ -κυρτά τμήματα. Αντίθετα, το αρχικό ανάπτγμα των μονόπλευρων και των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων είναι επαρκές, ώστε να εξασφαλίζεται η αναγνωσιμότητά τους μετά την αποκοπή του απαραίτητου για τη διεύρυνση μέρους αυτών.

Τα  $\epsilon$ -κυρτά τμήματα, των οποίων το ανάπτγμα βρίσκεται στο όριο της οπτικής αντίληψης, δεν γίνονται αντιληπτά από τον αναγνώστη του χάρτη. Για το λόγο αυτό, κρίνεται αναγκαία η συνένωσή τους με τα εκατέρωθεν γειτονικά  $\epsilon$ -τμήματα. Βάσει της μεθόδου κατάτμησης, μια χαρτογραφική γραμμή κατατέμνεται σε διαδοχικά  $\epsilon$ -



κυρτά και  $\epsilon$ -μη-κυρτά τμήματα (βλ. κεφ. 2.3.3). Εξαιρέση αποτελούν τα τμήματα σύγκλισης, τα οποία είναι δυνατό να γειτνιάζουν με  $\epsilon$ -τμήματα όλων των κατηγοριών. Έτσι, τα  $\epsilon$ -κυρτά τμήματα συνενώνονται με παρακείμενα μονόπλευρα ή/και αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα. Πιθανή συνένωση με τμήματα σύγκλισης ορίζεται να μην πραγματοποιείται αφού, λόγω των ιδιαίτερων χαρακτηριστικών της δομής και της γεωμετρίας τους, τα εν λόγω  $\epsilon$ -τμήματα χρήζουν ειδικής επεξεργασίας ως ανεξάρτητες οντότητες, στην αρχική τους μορφή. Στις περιπτώσεις αυτές, το  $\epsilon$ -κυρτό τμήμα συνενώνεται μόνο με το έτερο γειτονικό  $\epsilon$ -μη-κυρτό τμήμα.

Ουσιαστικά, μέσω της διαδικασίας συνένωσης, λαμβάνει χώρα μια περαιτέρω συγχώνευση των  $\epsilon$ -μη-κυρτών τμημάτων (εξαιρουμένων των τμημάτων σύγκλισης) που έχουν δημιουργηθεί κατά το στάδιο κατάτμησης της γραμμής. Στις περιπτώσεις, δηλαδή, που η αρχική απόσταση μεταξύ των  $\epsilon$ -μη-κυρτών τμημάτων (η οποία προσεγγίζεται επαρκώς μέσω του αναπτύγματος των  $\epsilon$ -κυρτών τμημάτων όταν κυμαίνεται στα όρια του ορατού) έχει ελαττωθεί σε βαθμό, που να γίνονται αντιληπτά ως μία, ενιαία οντότητα από τον αναγνώστη του χάρτη, αυτά συγχωνεύονται και ως τέτοια υπόκεινται στην επεξεργασία γενίκευσής τους. Η συγχώνευση είναι δυνατό να πραγματοποιηθεί μεταξύ δύο μονόπλευρων, δύο αμφίπλευρων, ενός μονόπλευρου κι ενός αμφίπλευρου ή συνδυασμού περισσότερων των δύο  $\epsilon$ -μη-κυρτών τμημάτων, με την προσάρτηση του ή των ενδιαμέσων  $\epsilon$ -κυρτών τμημάτων. Τα  $\epsilon$ -τμήματα που δημιουργούνται έχουν τη μορφή επάλληλων κυρτώσεων και κοιλώσεων και ως εκ τούτου, εντάσσονται στην κατηγορία των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων (κατηγορία Β). Με την ολοκλήρωση της διαδικασίας, διαμορφώνονται τα τελικά  $\epsilon$ -τμήματα, στα οποία κατατέμενται η χαρτογραφική γραμμή.



Σχήμα 2.10: Μοντέλο νέας συγχώνευσης  $\epsilon$ -μη-κυρτών τμημάτων και δημιουργίας τελικών  $\epsilon$ -τμημάτων.

Η διαδικασία συγχώνευσης υλοποιείται στο περιβάλλον του λογισμικού πακέτου ArcGIS, όπως παρουσιάζεται στο Σχήμα 2.10. Σε ένα πρώτο επίπεδο επεξεργασίας, από το σύνολο των  $\epsilon$ -κυρτών τμημάτων (Erased Normal Parts), όπως αυτά έχουν διαμορφωθεί μετά τη διεύρυνση των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων και των

τμημάτων σύγκλισης, επιλέγονται εκείνα (Select, TypeD\_ lessDT), των οποίων το ανάπτυγμα είναι μικρότερο μιας τιμής κατωφλίου. Το κατώφλι εξαρτάται από το όριο οπτικής αντίληψης (D) και μια τιμή ανοχής (T), ανηγμένα στην κλίμακα του χάρτη. Τα επιλεγμένα €-κυρτά τμήματα δεν είναι ορατά από τον αναγνώστη του χάρτη και συνενώνονται με τα εκατέρωθεν γειτονικά €-τμήματα, τα οποία μπορεί να είναι μονόπλευρα (Isolated Parts after Extend) ή/και αμφίπλευρα (Type B Successive Parts Final) €-μη-κυρτά τμήματα. Τα γειτονικά €-μη-κυρτά τμήματα εντοπίζονται (Select Layer By Location, Isolated Parts to be Merged, Successive Parts to be Merged) και συγχωνεύονται (*Merge1*), μαζί με τα αντίστοιχα €-κυρτά τμήματα. Κατά μήκος μιας χαρτογραφικής γραμμής, υπάρχει πιθανότητα να εντοπίζονται περισσότερα του ενός €-κυρτά τμήματα που χρήζουν συνένωσης και ως εκ τούτου, περισσότερες της μίας ομάδες €-τμημάτων που συγχωνεύονται. Για το λόγο αυτό, τα συγχωνευμένα €-τμήματα που αρχικά δημιουργούνται (Merged Parts), διαχωρίζονται (Multipart to Singlepart) στις ανάλογες ανεξάρτητες οντότητες (Merged Parts Final).

Σε ένα δεύτερο στάδιο επεξεργασίας, πραγματοποιείται η δημιουργία των τελικών €-τμημάτων. Όπως αναφέρεται παραπάνω, τα συγχωνευμένα €-τμήματα έχουν τη μορφή αμφίπλευρων €-μη-κυρτών τμημάτων και σε αυτά εντάσσονται. Για την αποφυγή καταγραφής στο τελικό αρχείο, των αμφίπλευρων €-μη-κυρτών τμημάτων που έχουν συγχωνευθεί, στην αρχική τους μορφή, εντοπίζονται (*Erase1*) τα €-τμήματα της κατηγορίας που δε μετέχουν σε συγχώνευση (Type B non Merged Parts). Σε αυτά προσαρτώνται (*Merge2*) τα συγχωνευμένα €-τμήματα (Merged Parts Final), δημιουργώντας τα νέα αμφίπλευρα €-μη-κυρτά τμήματα (Successive Parts Type B). Στη συνέχεια, από τα αρχικά μονόπλευρα €-μη-κυρτά τμήματα αποκόπτονται (*Erase2*) εκείνα που πιθανά μετέχουν σε συγχώνευση και προκύπτουν τα αντίστοιχα τελικά (Isolated Parts Type A). Ανάλογη διεργασία λαμβάνει χώρα και για τα €-κυρτά τμήματα (*Erase3*, Normal Parts Type D). Έτσι, συμπεριλαμβανομένων των τμημάτων σύγκλισης που έχουν εξαιρεθεί της συγκεκριμένης διαδικασίας, προκύπτουν τα τελικά €-τμήματα της χαρτογραφικής γραμμής.

## ΚΕΦΑΛΑΙΟ 3: ΜΕΘΟΔΟΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΧΑΡΤΟΓΡΑΦΙΚΩΝ ΓΡΑΜΜΩΝ

### 3.1 Μέθοδος εντοπισμού κρίσιμων σημείων κατά μήκος χαρτογραφικών γραμμών μέσω του δείκτη αναλογίας μήκους LR

Στο κεφάλαιο αυτό παρουσιάζεται η προτεινόμενη μέθοδος εντοπισμού κρίσιμων σημείων κατά μήκος χαρτογραφικών γραμμών. Η μέθοδος βασίζεται θεωρητικά στη μελέτη του ψυχολόγου Attneave (1954), ο οποίος καταλήγει, μεταξύ άλλων, στο συμπέρασμα πως ένα γραφικό αντικείμενο μπορεί να αναπαρασταθεί με μεγάλη οικονομία και πιστότητα αντιγράφοντας τα σημεία στα οποία το περιγράμμα τους αλλάζει σημαντικά κατεύθυνση και ενώνοντας αυτά τα σημεία με κατάλληλο τρόπο με μια ευθεία (βλ. κεφ. 1.1).

Πάνω σε αυτό το πλαίσιο, το κριτήριο, βάσει του οποίου γίνεται ο εντοπισμός των κρίσιμων σημείων κατά μήκος γραμμής έχει επιλεγεί με τέτοιο τρόπο, ώστε να ανιχνεύονται οι κορυφές που χαρακτηρίζονται από έντονη μεταβολή της κλίσης. Η κεντρική ιδέα είναι η αποκοπή της γραμμής γύρω από κάθε κορυφή και ο προσδιορισμός της καμπυλότητας ανεξάρτητα, χρησιμοποιώντας το μήκος ως γεωμετρικό κριτήριο. Η μέθοδος αξιοποιεί την έννοια της 'περιοχής εξέτασης', η οποία εμφανίζεται σε πολλούς αλγορίθμους εντοπισμού κρίσιμων σημείων της επιστήμης της πληροφορικής.

Στο πεδίο της επιστήμης της πληροφορικής, και ειδικότερα σε αντικείμενα, όπως η όραση υπολογιστών, η αναγνώριση μορφών ή η επεξεργασία σημάτων, έχει αναπτυχθεί ένας μεγάλος αριθμός αλγορίθμων εντοπισμού κρίσιμων σημείων, οι οποίοι κυρίως απευθύνονται σε προβλήματα, όπως η προσέγγιση γραμμών, η κατάτμηση καμπυλών ή η ανίχνευση χαρακτηριστικών γνωρισμάτων. Η πλειοψηφία αυτών των αλγορίθμων αυτών στηρίζεται στον υπολογισμό της καμπυλότητας σε κάθε σημείο της γραμμής, μέσω της ανάλυσης των γωνιών θλάσεών της. Τα πιθανά κρίσιμα σημεία εντοπίζονται σε θέσεις μεγίστων ή ελαχίστων της καμπυλότητας. Σύμφωνα με μία ταξινόμηση αλγορίθμων εντοπισμού κρίσιμων σημείων που παρουσιάστηκε από τον Li (1995), η πλειοψηφία τους ανήκει στην κατηγορία των αλγορίθμων 'ανίχνευσηςγωνιών'.

Στους πρώτους αλγορίθμους που δημιουργήθηκαν (Rosenfeld and Johnston 1973, Freeman and Davis 1977), η εκτίμηση της καμπυλότητας βασίζεται στον υπολογισμό της γωνίας θλάσης  $\theta$  (ή το συνημίτονό της  $-\cos\theta$ ) με κορυφή κάθε σημείου  $i$  μιας καμπύλης γραμμής, και άκρατα δύο σημεία της καμπύλης:  $i-k$  και  $i+k$ . Το εύρος της περιοχής ( $i-k$  -  $i+k$ ) ορίζεται μέσω παραμέτρων που ορίζονται από τον χρήστη. Όπως αποδείχθηκε, η χρήση ορισμένων παραμέτρων παρουσιάζουν αρκετά μειονεκτήματα, όπως η αδυναμία επεξεργασίας δεδομένων σε διαφορετικά επίπεδα λεπτομέρειας (μεγάλη τιμή της παραμέτρου οδηγεί σε απαλοιφή κρίσιμων σημείων, ενώ μικρή σε διατήρηση πλεοναζόντων σημείων) και η απαίτηση πολλαπλών δοκιμών. Στη χαρτογραφία, ο Thara (1988b) δημιούργησε έναν αλγόριθμο βασισμένο σε αυτή τη φιλοσοφία.

Η ιδέα της δημιουργίας μιας 'περιοχής εξέτασης' (region of support) γύρω από κάθε σημείο για τον υπολογισμό της καμπυλότητας, αποτέλεσε τη βάση, πάνω στην

οποία στηρίζονται οι περισσότεροι αλγόριθμοι εντοπισμού κρίσιμων. Οι Teh και Chin (1989) αναφέρουν ότι: 'ο ακριβής προσδιορισμός της περιοχής εξέτασης είναι πιο σημαντικός από το ίδιο το μέτρο υπολογισμού της καμπυλότητας'. Βασισμένοι σε αυτήν την ιδέα και για να αποφύγουν τις ανεπάρκειες της χρήσης παραμέτρων στην αντιμετώπιση του προβλήματος, δημιούργησαν ένα μη-παραμετρικό αλγόριθμο, στον οποίο η περιοχή εξέτασης για κάθε κορυφή ορίζεται με βάση τοπικές ιδιότητες της καμπύλης και δευτερευόντως, χρησιμοποιούνται μέτρα για τον υπολογισμό της καμπυλότητας. Οι Ray και Ray (1992), πρότειναν τη χρήση μη-συμμετρικής περιοχής εξέτασης.

Για την καλύτερη αντιμετώπιση του προβλήματος του 'θορύβου' των αλγορίθμων που βασίζονται σε αυτήν την λογική, και ειδικότερα, για να ξεπεραστεί η αδυναμία άμεσης εφαρμογής μαθηματικών σχέσεων για τον υπολογισμό της καμπυλότητας πάνω στις διακριτές αναπαραστάσεις των γραμμών σε ψηφιακό περιβάλλον, σε αρκετές έρευνες (για παράδειγμα σε αυτή των Ansari και Huang (1991)), προτείνεται η εξομάλυνση της γραμμής μέσω φίλτρων (με συνηθέστερο το φίλτρο Gauss), πριν τον υπολογισμό της καμπυλότητας. Σύμφωνα με τον Cornic (1997), η παραπάνω λύση θίγει το πρόβλημα της επιλογής ενός κατάλληλου εύρους του εφαρμοζόμενου φίλτρου, δεδομένου ότι ένα στενό σε εύρος φίλτρο Gauss είναι δυνατό να οδηγήσει σε εντοπισμό κρίσιμων σημείων που μπορούν να θεωρηθούν ως ασήμαντες λεπτομέρειες, ενώ ένα μεγάλο σε εύρος φίλτρο Gauss μπορεί να αποκλείσει ορισμένα κρίσιμα σημεία από τον εντοπισμό τους. Για το λόγο αυτό, σε άλλες έρευνες, (π.χ. Rattarangi και Chin (1992) και Pei και Lin (1992)), προτείνεται η εξομάλυνση της γραμμής μέσω φίλτρου Gauss σε διάφορα επίπεδα με κλιμακούμενα εύρη (από ένα ελάχιστο έως έναμέγιστο).

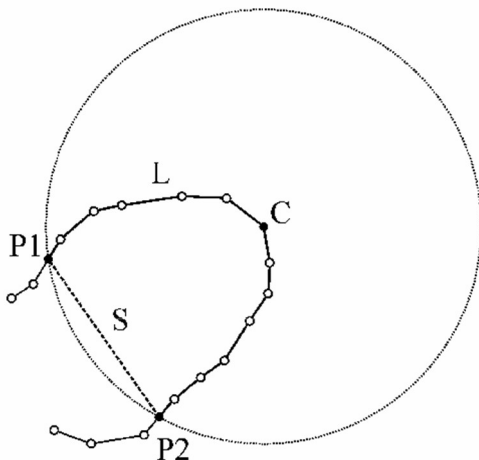
Ακολουθώντας την αρχή της επεξεργασίας της γραμμής σε τοπικό επίπεδο, έχουν δημιουργηθεί αρκετοί αλγόριθμοι, οι οποίοι δεν βασίζονται στον υπολογισμό της καμπυλότητας για τον εντοπισμό των κρίσιμων σημείων. Ο Cornic (1997) παρουσιάζει έναν μη-παραμετρικό αλγόριθμο, κατά τον οποίο η αξιολόγηση της σημαντικότητας κάθε σημείου δεν βασίζεται στον υπολογισμό παραμέτρων του ίδιου, αλλά στη 'σχέση' του με τα γειτονικά του. Το κάθε σημείο της καμπύλης κρίνεται με βάση πόσες φορές αποτελεί ακραίο σημείο περιοχής εξέτασης (αριστερό και δεξί όριο) γειτονικών του σημείων. Βάση παραδοχών, κρίσιμα θεωρούνται εκείνα που εμφανίζονται τις περισσότερες φορές ως άκρα περιοχών εξέτασης. Ο Cronin (1999) παρουσιάζει αντίστοιχο αλγόριθμο, κατά τον οποίο κάθε σημείο της γραμμής κατατάσσεται σε μία από δεκαοχτώ κατηγορίες, με κριτήριο τη θέση και τον προσανατολισμό του σε σχέση με το προηγούμενο και το επόμενο. Γενικά, τα σημεία κωδικοποιούνται ως σημεία κυρτότητας (τοπικό μέγιστο), ως σημεία κοιλότητας (τοπικό ελάχιστο) και ως τρέχοντα σημεία (μηδενική γωνία). Μέσω της κωδικοποίησης εντοπίζονται τα μέγιστα και τα ελάχιστα και μετά από φιλτράρισμα με τη βοήθεια μιας προϋπολογισμένης συνάρτησης σφάλματος, ώστε να απαλειφθούν όσες κορυφές χαρακτηρίζονται πολύ έντονα αιχμηρές ή έντονα αβαθείς, επιλέγονται τα κρίσιμα σημεία. Ο Antoine (1997) προτείνει μία μέθοδο, κατά την οποία εφαρμόζεται η θεωρία των μικροκυματικών συναρτήσεων για τον εντοπισμό των κρίσιμων σημείων.

Αν και στην περιοχή της ψηφιακής επεξεργασίας εικόνας η έννοια των κρίσιμων σημείων συναντάται σε διαφορετικά πεδία εφαρμογής από αυτά της χαρτογραφίας

και η διαχείριση των γραμμικών οντοτήτων πραγματοποιείται σε αρχεία κανονικοποιημένης (raster) δομής, οι διαδικασίες προσέγγισής τους παρουσιάζουν ιδιαίτερο ενδιαφέρον. Η ανάπτυξη μεθόδων εντοπισμού κρίσιμων σημείων θέτει χρήσιμα στοιχεία και νέες λογικές για αντίστοιχες προσπάθειες στη χαρτογραφία. Πρώτα απ' όλα, ο ορισμός μιας περιοχής εξέτασης γύρω από το εκάστοτε ελεγχόμενο σημείο είναι μια χρήσιμη τεχνική. Σύμφωνα με αυτήν, η καμπυλότητα της γραμμής υπολογίζεται μέσω της ανάλυσης της γωνιακότητας, και με βάση τις αρχές του Attneave εντοπίζονται οι κρίσιμες θέσεις. Έτσι, πραγματοποιείται διερεύνηση της σημαντικότητας των σημείων σε τοπικό και όχι σε καθολικό επίπεδο, γεγονός που αποτελεί ζητούμενο για τους χαρτογράφους. Παράλληλα, ενδιαφέρον παρουσιάζουν οι προσπάθειες αντιμετώπισης του 'προβλήματος' της διακριτής απόδοσης συνεχών φαινομένων μέσω υπολογιστή, με την χρήση τεχνικών εξομάλυνσης των γραμμών μέσω φίλτρων και μάλιστα σε διαφορετικά επίπεδα, ανάλογα με την τοπική μορφολογία τους. Τέλος, πρωτοποριακές μπορούν να θεωρηθούν οι πρόσφατες προσπάθειες δημιουργίας μη-παραμετρικών αλγορίθμων, οι οποίες είναι ανεξαρτημένες από δοκιμές και υποκειμενικές παρεμβάσεις του χρήστη.

### 3.1.1 Ορισμός δείκτη αναλογίας μήκους LR

Σε ψηφιακό περιβάλλον, ο δείκτης αναλογίας μήκους LR εφαρμόζεται σε γραμμές διανυσματικής (vector) δομής, δηλαδή, σε ένα σύνολο διακριτών σημείων, ενωμένων μέσω διανυσμάτων. Η



**Σχήμα 3.1:** Τυπική περίπτωση τομής μεταξύ γραμμής και κύκλου.

εφαρμογή κύκλου που φέρεται με κέντρο (C) την εκάστοτε κορυφή της γραμμής. Η ακτίνα (R) του κύκλου είναι σταθερή και προκαθορισμένη. Τα όρια της 'περιοχής εξέτασης' (P1 και P2) ορίζονται ως οι δύο διαδοχικές, εκατέρωθεν του κέντρου (C), τομές του κύκλου και της γραμμής. Μεταξύ των σημείων τομής P1 και P2 είναι δυνατό να υπολογισθεί το ανάπτυγμα της γραμμής L και το μήκος της μεταξύ τους χορδής S (Σχήμα 3.1).

Ο δείκτης αναλογίας μήκους (LR) ορίζεται από τη σχέση:  $LR = \frac{L}{S}$ .

Ένας κύκλος είναι δυνατό:

1. Να αποκόπτει τη γραμμή σε δύο σημεία P1 και P2, εκατέρωθεν της εξεταζόμενης κορυφής (Σχήμα 3.1). Στην περίπτωση αυτή, η οποία είναι η πιο συνηθής για τις προτεινόμενες ακτίνες, εντοπίζονται οι τομές P1-P2 και υπολογίζονται το

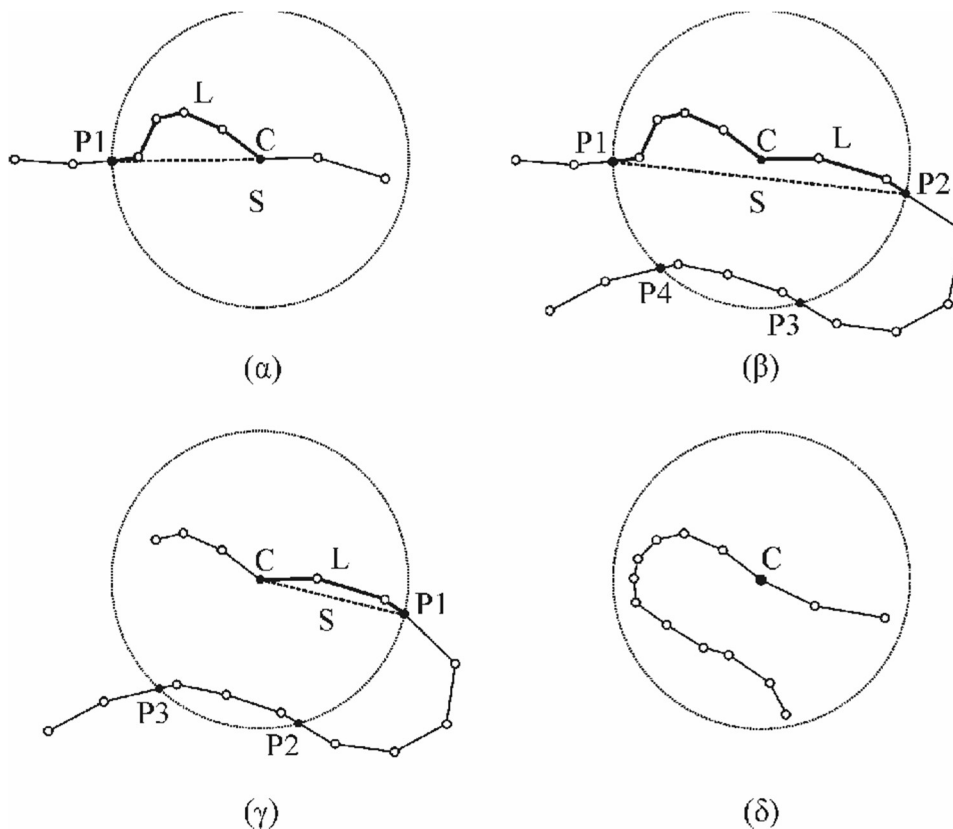
ανάπτυγμα  $L_{P1-P2}$  (ως άθροισμα των ενδιάμεσων ευθυγράμμων τμημάτων) και η απόσταση χορδής  $S_{P1-P2}$ . Ο δείκτης LR είναι:  $LR = \frac{L_{P1-P2}}{S_{P1-P2}}$ .

2. Να αποκόπτει τη γραμμή σε ένα μόνο σημείο P1 (Σχήμα 3.2α). Στην περίπτωση αυτή, η οποία εμφανίζεται σε ακραίες θέσεις ανοικτών γραμμών, εντοπίζεται η τομή κύκλου γραμμής P1 και υπολογίζεται το ανάπτυγμα  $L_{P1-C}$ , μεταξύ κέντρου κύκλου και σημείου τομής και η απόσταση τομής  $S_{P1-C}$ , η οποία είναι ίση με την ακτίνα του κύκλου R. Ο δείκτης LR είναι:  $LR = \frac{L_{P1-C}}{S_{P1-C}} = \frac{L_{P1-C}}{R}$ .

3. Να αποκόπτει τη γραμμή σε παραπάνω των δύο σημείων P1, P2, P3 ... Pn, εκατέρωθεν κορυφής (Σχήμα 3.2β). Στην περίπτωση αυτή, εντοπίζονται τα εγγύτερα σημεία τομής P1-P2 εκατέρωθεν της κορυφής και ακολουθείται διαδικασία ανάλογη της περίπτωσης 1.

4. Να αποκόπτει τη γραμμή σε παραπάνω των δύο σημείων P1, P2, P3 ... Pn, από τη μια πλευρά της κορυφής (Σχήμα 3.2γ). Στην περίπτωση αυτή, εντοπίζεται το εγγύτερο στην κορυφή σημείο τομής P1 και ακολουθείται διαδικασία ανάλογη της περίπτωσης 2.

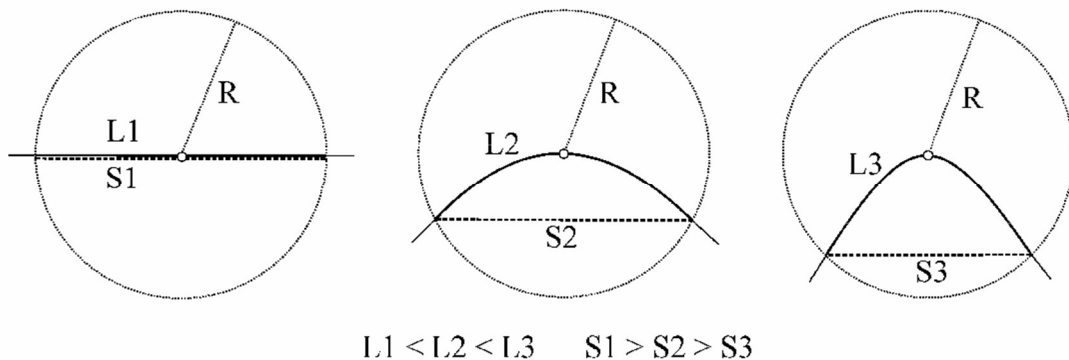
5. Να μην αποκόπτει τη γραμμή (Σχήμα 3.2δ). Στην περίπτωση αυτή, το μέτρο δεν εφαρμόζεται και απαιτείται μικρότερη ακτίνα κύκλου.



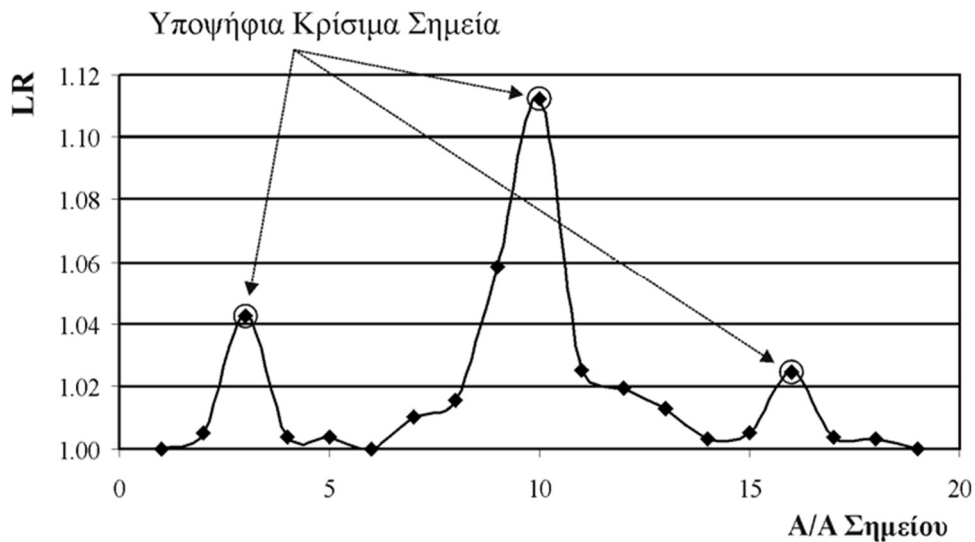
Σχήμα 3.2: Τέσσερις εναλλακτικές περιπτώσεις τομής μεταξύ κύκλου και γραμμής.

Με την εφαρμογή ενός κύκλου σταθερής ακτίνας, ο οποίος ‘επισκέπτεται’ διαδοχικά όλες τις κορυφές μιας γραμμής, η γραμμή τμηματοποιείται ισομερώς γύρω από κάθε κορυφή, βάσει ενός κοινού μέτρου. Δημιουργείται, έτσι, μία σταθερή και ισοδύναμη ‘περιοχή εξέτασης’ γύρω από κάθε σημείο, μέσα στην οποία υπολογίζεται το ανάπτυγμα  $L$ , η απόσταση  $S$  και τελικά, ο δείκτης  $LR$ . Ο δείκτης αναλογίας μήκους  $LR$ , σύμφωνα με τον ορισμό του, συσχετίζεται έμμεσα με την έννοια της αυτό-όμοιας κλασματικής διάστασης (self-similar fractal dimension) (Mandelbrot 1982) και κατά συνέπεια, εκφράζει ποσοτικά το βαθμό πολυπλοκότητας του αποκοπτόμενου τμήματος των γραμμών.

Υποθέτοντας πως το βήμα ψηφιοποίησης μιας ψηφιακής γραμμής παραμένει κατά προσέγγιση σταθερό (ώστε η μονάδα μέτρησης του αναπτύγματος της γραμμής να είναι κοινή) και μικρότερο της ( $R$ ) ακτίνας του κύκλου, το ανάπτυγμα  $L$  και το μήκος χορδής  $S$  μεταβάλλονται ανάλογα με τη μορφή της γραμμής στην περιοχή εξέτασης. Αυξανόμενη της καμπυλότητας, το ανάπτυγμα  $L$  αυξάνεται, ενώ αντίστοιχα η απόσταση  $S$  ελαττώνεται (Σχήμα 3.3). Συνεπώς και ο δείκτης  $LR$  μεταβάλλεται ανάλογα με την καμπυλότητα της γραμμής στην περιοχή εξέτασης και πιο συγκεκριμένα, αυξάνει όσο πιο μεγάλη είναι η καμπυλότητα. Ο δείκτης  $LR$  είναι αδιάστατος και πάντα μεγαλύτερος ή ίσος της μονάδας ( $LR \geq 1$ ) (η ισότητα εμφανίζεται στην ευθυγραμμία). Συνεπώς, ο δείκτης  $LR$  μπορεί να λειτουργήσει σαν ένα μέτρο προσδιορισμού της καμπυλότητας. Υπολογίζοντας τον δείκτη  $LR$  σε κάθε περιοχή εξέτασης (δηλαδή, σε κάθε κορυφή) και εντοπίζοντας τα τοπικά μέγιστα, προσδιορίζονται τα σημεία της γραμμής, τα οποία μπορούν να θεωρηθούν κρίσιμα. (Σχήμα 3.4).



**Σχήμα 3.3:** Μεταβολή του αναπτύγματος  $L$  και της χορδής  $S$  συναρτήσει της καμπυλότητας.



Σχήμα 3.4: Εντοπισμός κρίσιμων σημείων σε διάγραμμα LR.

### 3.1.2 Ορισμός μεγέθους ακτίνας κύκλου R

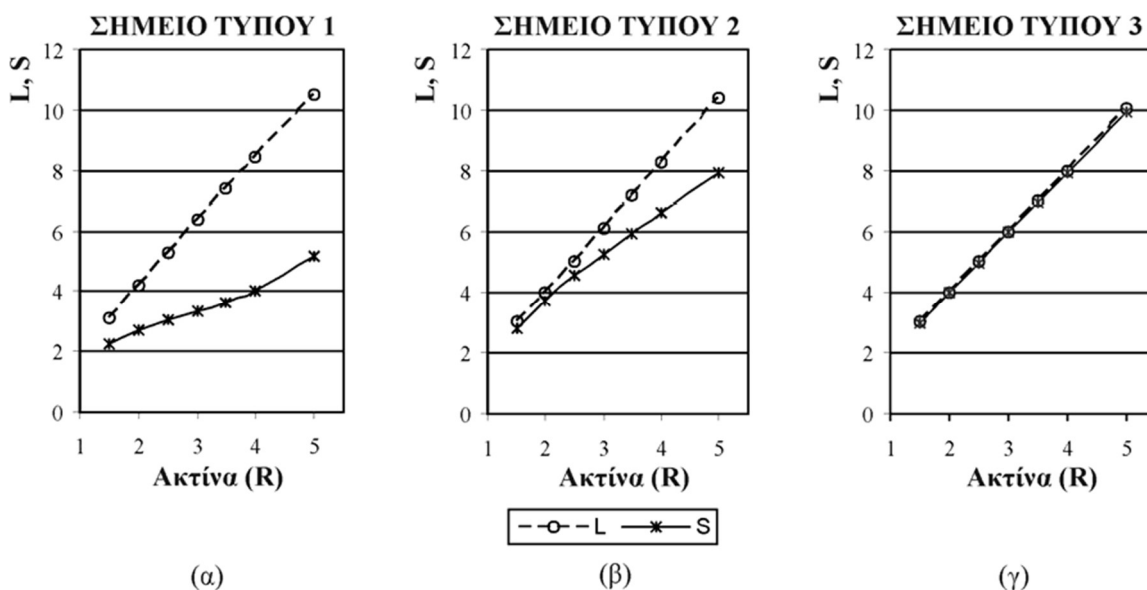
Βασική αρχή της προτεινόμενης μεθόδου εντοπισμού κρίσιμων σημείων είναι η εκτίμηση της καμπυλότητας της γραμμής μέσω του δείκτη LR, σε τοπικό επίπεδο, γύρω από κάθε κορυφή της γραμμής. Το μέγεθος της ακτίνας (R) του κύκλου εφαρμογής αποτελεί παράγοντα που επιδρά άμεσα στην ποιότητα των αποτελεσμάτων, αφού αυτό ορίζει το εύρος της περιοχής εξέτασης. Εύκολα μπορεί να παρατηρηθεί, πως αυξάνοντας την εφαρμοζόμενη ακτίνα το ανάπτυγμα της γραμμής (L) και το μήκος χορδής (S) μεταβάλλονται θετικά, αφού η περιοχή εξέτασης επεκτείνεται. Όμως, η μεταβολή τους δεν είναι ισόποση. Αυξανόμενη της καμπυλότητας της γραμμής, η αναλογία αύξησης του L εμφανίζεται μεγαλύτερη από αυτή του S. Το ποσοστό αύξησής τους, όμως, γύρω από κάθε θέση, διαφέρει, γεγονός που εξαρτάται από την καμπυλότητα της γραμμής. Για την ευκολότερη κατανόηση των παραπάνω, παρατίθεται ένα παράδειγμα μεταβολής του αναπτύγματος L και της χορδής S για ένα ευρύ φάσμα ακτινών (R) εφαρμογής. Στο παράδειγμα αυτό, το μέγεθος της ακτίνας ξεκινά από τιμή ίση με το μέσο βήμα ψηφιοποίησης και αυξάνεται αναλογικά ως προς αυτό, για τρία χαρακτηριστικά σημεία, αντίστοιχα (Σχήμα 3.5).



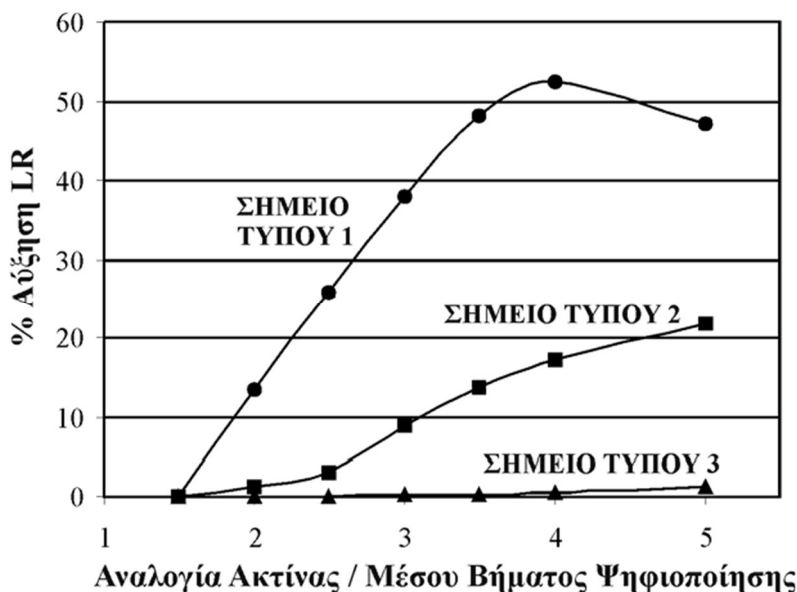
Σχήμα 3.5: Τρεις τύποι χαρακτηριστικών σημείων:  
 Σημείο τύπου 1: θέση πολύ έντονης μεταβολής κλίσης  
 Σημείο τύπου 2: θέση έντονης μεταβολής κλίσης  
 Σημείο τύπου 3: θέση ευθυγράμμου τμήματος.



Ερμηνεύοντας το Σχήμα 3.6, παρατηρείται πως το ποσοστό αύξησης του αναπτύγματος  $L$ , σε σχέση με αυτό του μήκους της χορδής  $S$ , εμφανίζεται μεγαλύτερο σε θέσεις μεγάλης καμπυλότητας (Σχήμα 3.6α). Η διαφορά αυτή ελαττώνεται σε σημεία μέσης καμπυλότητας (Σχήμα 3.6β) και τείνει να μηδενιστεί στα ευθύγραμμα τμήματα της γραμμής (Σχήμα 3.6γ). Αντίστοιχα μεταβάλλεται και η ποσοστιαία αύξηση των τιμών του δείκτη  $LR$ , για τις αντίστοιχες ακτίνες εφαρμογής, όπως φαίνεται στο Σχήμα 3.7. Το ζητούμενο είναι κρίσιμα σημεία που αντιστοιχούν σε χαμηλές τιμές του δείκτη  $LR$  να διακρίνονται από τις υπόλοιπες, απλές κορυφές της γραμμής. Ο τρόπος μεταβολής των τιμών του λόγου  $LR$  (όπως παρουσιάζεται στο Σχήμα 3.6), οδηγεί στο συμπέρασμα, πως κάτι τέτοιο μπορεί να επιτευχθεί με την εφαρμογή κύκλων με ακτίνες ίσες με δύο έως τέσσερις φορές το μέσο βήμα ψηφιοποίησης της γραμμής. Μέσα σε αυτά τα όρια παρατηρούνται οι μέγιστες αποκλίσεις μεταξύ των μηκών  $L$  και  $S$  και, ως εκ τούτου, με αυτά τα μεγέθη των ακτίνων εμφανίζονται οι μεγαλύτερες διαφοροποιήσεις στις τιμές του δείκτη  $LR$  για τον εντοπισμό των κρίσιμων σημείων. Μέσα σε αυτά τα όρια ανοχής εντοπίζονται ευκολότερα κρίσιμα σημεία χαμηλού δείκτη  $LR$  (π.χ. Σημείο τύπου 2). Όταν οι ακτίνες του κύκλου υπερβαίνουν περίπου κατά πέντε φορές το μέσο βήμα ψηφιοποίησης, τότε οι τιμές του δείκτη  $LR$  τείνουν να εκφράσουν γενικά παρά τοπικά χαρακτηριστικά της γραμμής.



**Σχήμα 3.6:** Μεταβολή του αναπτύγματος  $L$  και του μήκους χορδής  $S$ , αυξανομένης της ακτίνας, για τρεις τύπους χαρακτηριστικών σημείων.



**Σχήμα 3.7:** Ποσοστιαία αύξηση του δείκτη LR, αυξανόμενης της ακτίνας, για τρεις τύπους χαρακτηριστικών σημείων.

### 3.1.3 Ταξινόμηση τιμών του δείκτη LR και ορισμός ορίου επιλογής

Οι τιμές του δείκτη LR μπορούν να ταξινομηθούν σε τρεις ομάδες (Α, Β και Γ), μιας κλίμακας τάξης, σύμφωνα με τα μορφολογικά χαρακτηριστικά των γραμμών. Μετά από εμπειρική διερεύνηση, επιλέχθηκαν τα ακόλουθα όρια των τιμών του LR για τον ορισμό των τριών ομάδων:

- Ομάδα Α: Τιμές του δείκτη LR μεταξύ 1.04 και 1.15:

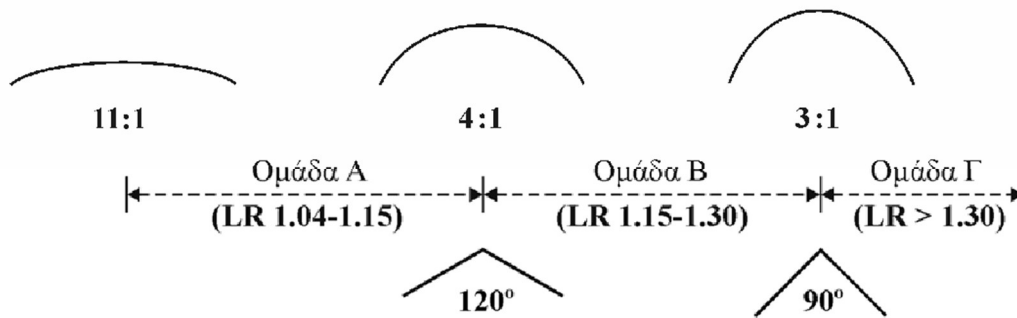
Τα κρίσιμα σημεία αντιστοιχούν σε θέσεις ομαλών κλίσεων (γωνίες θλάσης μέχρι 120°), με αναλογία βάσης ως προς ύψος από 4:1 έως 11:1. (Σχήμα 3.8)

- Ομάδα Β: Τιμές του δείκτη LR μεταξύ 1.15 και 1.30:

Τα κρίσιμα σημεία αντιστοιχούν σε θέσεις έντονων αλλαγών των κλίσεων (γωνίες θλάσης από 90° έως 120°), με αναλογία βάσης ως προς ύψος από 3:1 έως 4:1. (Σχήμα 3.8)

- Ομάδα Γ: Τιμές του δείκτη LR μεγαλύτερες από 1.30:

Τα κρίσιμα σημεία αντιστοιχούν σε θέσεις πολύ έντονων αλλαγών των κλίσεων (γωνίες θλάσης μικρότερες από 90°), με αναλογία βάσης ως προς ύψος μικρότερη από 3:1. (Σχήμα 3.8)



**Σχήμα 3.8:** Χαρακτηριστικές μορφές των καμπυλών για τις τρεις ομάδες τιμών του δείκτη LR.

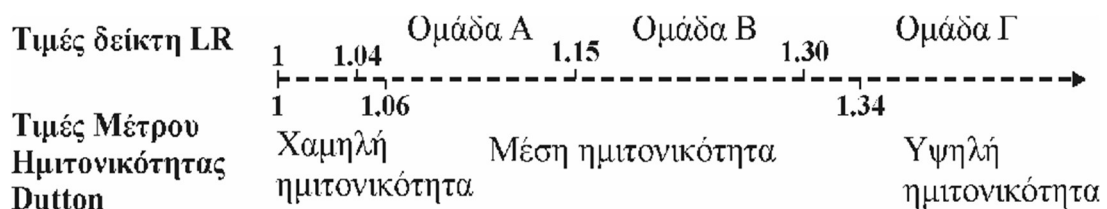
Πρέπει να σημειωθεί ότι στην παραπάνω ταξινόμηση χρησιμοποιείται ένα κατώτατο όριο για τις τιμές του δείκτη LR ίσο με 1,04. Μια εμπειρική ανάλυση διαφορετικών δοκιμαστικών γραμμών, οι οποίες ποικίλουν ως προς το πάχος τους σύμφωνα με τα πρότυπα των γραμμικών χαρτογραφικών συμβόλων και ως προς τη μορφή τους, έδειξε ότι μόνον οι κορυφές που σχετίζονται με κυρτώσεις της γραμμής που έχουν αναλογία βάσης ως προς ύψος μικρότερη από 11:1 μπορούν να θεωρηθούν κρίσιμες. Υπολογίζοντας την τιμή του δείκτη LR σε αυτές τις κυρτώσεις προκύπτει το κατώτατο όριο ίσο με την τιμή 1,04. Αυτό το κατώτατο όριο αποκλείει από το σύνολο των κρίσιμων σημείων εκείνες τις κορυφές που εντοπίζονται σε θέσεις πολύ μικρών κλίσεων ή ευθύγραμμα τμήματα της γραμμής, ή ακόμα, και τον 'θόρυβο' που παρουσιάζεται σε κάθε ψηφιακή αναπαράσταση των αναλογικών γραμμών.

### 3.1.4 Σύγκριση δείκτη LR με 'μέτρο ημιτονικότητας' του Dutton

Ο Dutton (1999) αναφέρει ότι ένα σημαντικό κριτήριο που πρέπει να λαμβάνεται υπόψη για την ορθή επιλογή των διατηρούμενων σημείων κατά τη διαδικασία απλοποίησης γραμμών είναι η σημείο προς σημείο εκτίμηση της τοπικής ημιτονικότητας της γραμμής. Για αυτόν το λόγο, ο Dutton (1999) θέσπισε το στατιστικό 'μέτρο ημιτονικότητας'. Σε κάθε σημείο υπολογίζεται ο λόγος του μήκους της ψηφιοποιημένης γραμμής, μεταξύ  $\pm k$  γειτονικών σημείων, προς το μήκος της ευθείας βάσης που ενώνει τα δύο αυτά ακραία σημεία (βλ. Dutton (1999), Σχήμα 4). Οι 'τιμές ημιτονικότητας' (SV) που υπολογίζονται με αυτόν τον τρόπο είναι αδιάστατοι πραγματικοί αριθμοί και εκφράζουν και αυτές τις μεταβολές της καμπυλότητας της γραμμής γύρω από κάθε κορυφή. Ο αριθμός (k) των γειτονικών σημείων που συμμετέχουν στον υπολογισμό του μέτρου καθορίζουν το εύρος της περιοχής εξέτασης και κατ' επέκταση τις τιμές SV αυτού. Ο Dutton (1999) καταλήγει στο συμπέρασμα, ότι μια πιο ισχυρή εκτίμηση της ημιτονικότητας των γραμμών επιτυγχάνεται με τον υπολογισμό του μέσου όρου των τιμών του 'μέτρου ημιτονικότητας', ύστερα από διαδοχική εφαρμογή του σε ένα μικρό εύρος παρακείμενων σημείων.

Η δομή του 'μέτρου ημιτονικότητας' παρουσιάζει αρκετές ομοιότητες με αυτή του δείκτη LR. Οι δύο δείκτες εκτιμούν την καμπυλότητα της γραμμής με σχετικά παρόμοιο τρόπο. Με την εφαρμογή ενός συγκεκριμένου μετασχηματισμού ταξινόμησης, ο Dutton συνδέει τις τιμές SV του μέτρου σε τρία επίπεδα

ημιτονικότητας (βλ. Dutton (1999), Σχήμα 5). Μια σύγκριση μεταξύ των τριών ομάδων (Α, Β και Γ) των τιμών του LR και της ταξινόμησης που ακολουθεί ο Dutton παρουσιάζεται στο Σχήμα 3.9. Όπως φαίνεται, οι τιμές SV του μέτρου μεταξύ 1,06 και 1,34 αντιστοιχούν σε τμήματα των γραμμών μέσης 'ημιτονικότητας'. Αυτές οι τιμές είναι πολύ κοντά στο όριο μεταξύ των ομάδων Α και Β των τιμών LR (1,04 έως 1,15 και 1,15 έως 1,30 αντιστοίχα). Για τιμές SV του μέτρου μεγαλύτερες από 1,34, ο Dutton (1999) καταλήγει ότι αντιστοιχούν σε περιοχές της γραμμής που χαρακτηρίζονται από μεγάλη 'ημιτονικότητα'. Η τιμή αυτή βρίσκεται πολύ κοντά στο κάτω όριο των τιμών του δείκτη LR για την ομάδα Γ (1,30). Είναι προφανές, ότι οι τιμές του δείκτη LR, που αντιστοιχούν σε κρίσιμα σημεία μιας γραμμής, συμπίπτουν με τις κατηγορίες μέσης και μεγάλης 'ημιτονικότητας' της ταξινόμησης του Dutton. Το όριο επιλογής του δείκτη LR (1,04) εμφανίζεται στο άνω άκρο της κατηγορίας χαμηλής 'ημιτονικότητας'. Αντίθετα στην υπόλοιπη περιοχή χαμηλής 'ημιτονικότητας' του Dutton δεν αντιστοιχούν τιμές του δείκτη LR, αφού, από τον ορισμό τους, κρίσιμα σημεία δεν εντοπίζονται σε περιοχές χαμηλής καμπυλότητας.



Σχήμα 3.9: Σύγκριση μεταξύ της ταξινόμησης των τιμών του δείκτη LR και των τιμών ημιτονικότητας SV.

### 3.1.5 Ορισμός δεικτών LLR και ALR

Τέλος, με στόχο να προσδιοριστεί η καμπυλότητα της γραμμής σε τοπικό επίπεδο, ορίζονται δύο είδη δεικτών LR:

- Ο τοπικός δείκτης LR (LLR), ο οποίος υπολογίζεται όταν εφαρμόζεται ένας κύκλος ακτίνας ίσης με δύο φορές το μέσο βήμα της ψηφιοποιημένης αναλογικής γραμμής.
- Ο μέσος δείκτης LR (ALR), ο οποίος ορίζεται ως ο μέσος όρος των δεικτών LR όταν εφαρμόζονται διαδοχικά στη γραμμή κύκλοι με ακτίνες: ίσες με το μέσο βήμα ψηφιοποίησης (R1), δύο φορές το μέσο βήμα της ψηφιοποίησης (R2), τρεις φορές το μέσο βήμα της ψηφιοποίησης (R3) και τέλος, τέσσερις φορές το μέσο βήμα της ψηφιοποίησης (R4) της αναλογικής γραμμής. Κατά συνέπεια, ο δείκτης ALR εκφράζεται από τη σχέση:

$$ALR = \frac{LR_{R1} + LR_{R2} + LR_{R3} + LR_{R4}}{4}$$

Η εκτίμηση της καμπυλότητας της γραμμής πραγματοποιείται σε τοπικό επίπεδο, γύρω από κάθε κορυφή, με την εφαρμογή και των δύο δεικτών. Η διαφορά μεταξύ των δύο δεικτών εστιάζεται στο εύρος της περιοχής εξέτασης. Με την εφαρμογή του δείκτη LLR, η αλλαγή των κλίσεων υπολογίζεται σε μια μικρή περιοχή γύρω από

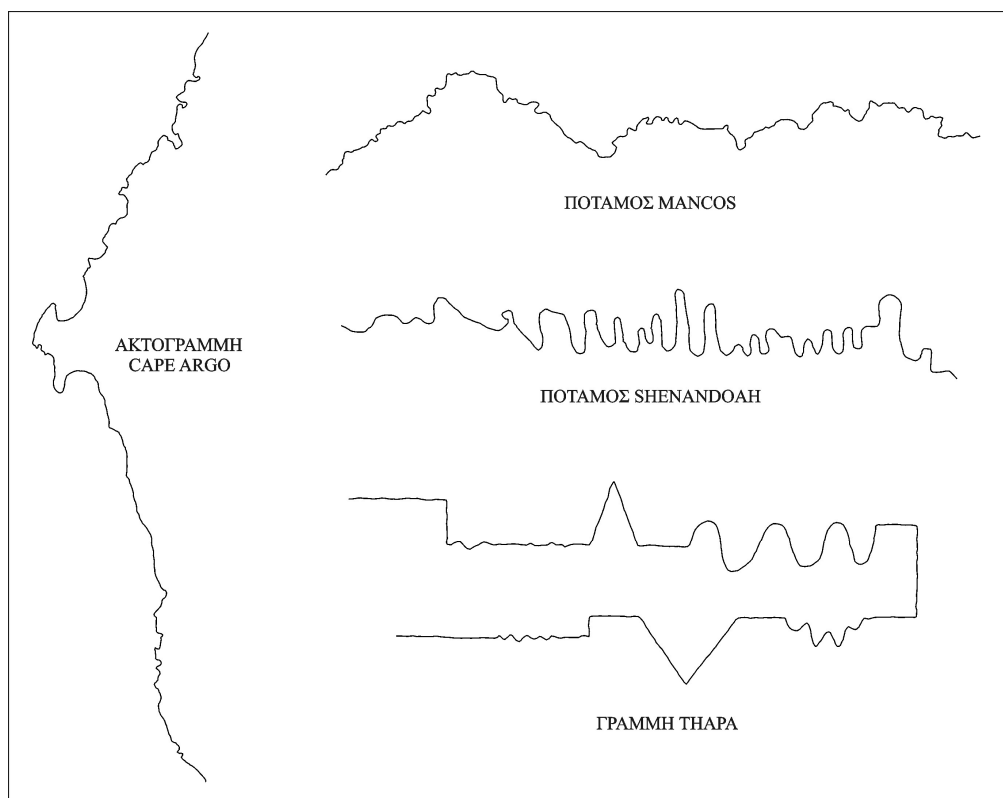
κάθε κορυφή. Κατά συνέπεια, η επιρροή των τοπικών ιδιοτήτων της γραμμής είναι υψηλή στο δείκτη. Με εφαρμογή του δείκτη ALR, η καμπυλότητα υπολογίζεται σε τοπικό επίπεδο (R1), καθώς επίσης και σε ευρύτερα επίπεδα (R3, R4). Κατά συνέπεια, μειώνεται η επιρροή της μορφολογίας των παρακείμενων κορυφών. Τέλος, η σημασία της κάθε κορυφής υπολογίζεται σε έναν ευρύτερο τομέα και έτσι, οι τιμές του δείκτη ALR είναι περισσότερο συστηματοποιημένες από αυτές του δείκτη LLR.

Η ρουτίνα της μεθόδου κωδικοποιήθηκε σε γλώσσα προγραμματισμού *Python* και παρατίθεται στο Παράρτημα Π3.1.

### **3.1.6 Έλεγχος λειτουργικότητας μεθόδου**

#### **3.1.6.1 Σύγκριση με σχετικές μελέτες**

Η αξιοπιστία των αποτελεσμάτων της προτεινόμενης μεθόδου ελέγχεται εφαρμόζοντας τον δείκτη LR σε γραμμικές οντότητες, των οποίων τα κρίσιμα σημεία είναι εκ των προτέρων γνωστά. Ως αρχεία αναφοράς χρησιμοποιούνται τρεις γεωμορφολογικές χαρτογραφικές γραμμές, οι ποταμοί Mancos και Shenandoah και η ακτογραμμή Cape Arago, οι οποίες προέρχονται από την εμπειρική έρευνα της Marino (1979), καθώς και η γεωμετρική γραμμή του Thara (Thara 1987). Οι τρεις πρώτες απεικονίζουν φυσικές οντότητες και αποτελούν διαφορετικά γεωγραφικά και γεωμορφολογικά δείγματα γραμμών. Ο ποταμός Shenandoah έχει μορφή ημιτονοειδούς καμπύλης, ο ποταμός Mancos χαρακτηρίζεται από μεγάλη πολυπλοκότητα, ενώ η ακτογραμμή Cape Arago περιέχει συγχρόνως περιοχές μεγάλης πολυπλοκότητας και ευθύγραμμα τμήματα. Τα κρίσιμα σημεία είναι προϊόν της εμπειρικής μελέτης της Marino (1979), όπως αυτή περιγράφεται στο Κεφάλαιο 1.1. Πιο συγκεκριμένα, στην παρούσα έρευνα χρησιμοποιούνται τα κρίσιμα σημεία που προέρχονται από το πρώτο επίπεδο απλοποίησης, ώστε να υπάρχει το μεγαλύτερο δυνατό δείγμα. Η θεωρητική γραμμή του Thara αποτελεί ένα γεωμετρικό μοντέλο, σχεδιασμένο με τρόπο, ώστε να περιέχει το μεγαλύτερο μέρος των σχημάτων (σφήνες, απότομες αλλαγές κλίσεων, μικροί κυματισμοί, συνεχείς μεγάλες καμπύλες, ευθύγραμμα τμήματα κτλ.), τα οποία μπορούν, είτε ανεξάρτητα, είτε σε συνδυασμό, να εντοπισθούν σε μια τυχαία γραμμή. Στην παρούσα έρευνα χρησιμοποιούνται τα κρίσιμα σημεία που προέρχονται από την εφαρμογή του μαθηματικού μοντέλου, που ο Thara προτείνει στην έρευνά του (Thara 1988b, 64). Οι απεικονίσεις των τεσσάρων γραμμών παρουσιάζονται στο Σχήμα 3.10.



**Σχήμα 3.10:** Γραμμές από τις μελέτες των Marino και Thara.

Οι γραμμές από τις έρευνες της Marino (1979) και του Thara (1987) επιλέχθηκαν προκειμένου να ελεγχθούν άμεσα τα αποτελέσματα της προτεινόμενης μεθόδου εντοπισμού κρίσιμων σημείων. Η πρώτη είναι μια έρευνα που απεικονίζει τον τρόπο, με τον οποίο οι άνθρωποι αντιλαμβάνονται και επιλέγουν τα κρίσιμα σημεία (Marino), ενώ η δεύτερη μια παρόμοια με την προτεινόμενη μέθοδο, βασισμένη σε μαθηματικές αρχές (Thara). Κατά συνέπεια, και οι δύο έρευνες αποτελούν ένα ισχυρό υπόβαθρο για τον έλεγχο της προτεινόμενης μεθόδου.

Τα πρωτογενή δεδομένα των γραμμών ελέγχου δημιουργήθηκαν με ψηφιοποίηση των σαρωμένων, με ανάλυση 300dpi, εικόνων των κειμένων της Marino και του Thara. Οι τελικές ψηφιακές γραμμές δημιουργήθηκαν μετά από επεξεργασία διόρθωσης και εξομάλυνσης των πρωτογενών δεδομένων, προκειμένου να απαλειφθούν περιττές κορυφές και να εξομαλυνθούν, σύμφωνα με την τυποποιημένη χαρτογραφική διαδικασία που περιγράφεται από τον Jenks (1981). Τα μέσα βήματα ψηφιοποίησης των γραμμών ελέγχου παρουσιάζονται στον Πίνακα 3.1. Εντούτοις, υπάρχουν σημεία ανεπαρκούς ψηφιοποίησης των αναλογικών γραμμών λόγω της ποιότητας των πηγών. Τα σημεία αυτά εντοπίστηκαν και χαρακτηρίστηκαν εξ αρχής, ώστε οι τιμές του δείκτη LR να ελεγχθούν και να μην επηρεαστεί η τελική ποιότητα του αποτελέσματος της έρευνας.

Ο εντοπισμός των κρίσιμων σημείων των τεσσάρων γραμμών πραγματοποιήθηκε με την εφαρμογή της προτεινόμενης μεθόδου, χρησιμοποιώντας τόσο τον δείκτη LLR, όσο και τον ALR. Στόχος είναι η αξιολόγηση της λειτουργικότητάς τους, αλλά και η μεταξύ τους σύγκριση, βάσει γνωστών δεδομένων. Οι παράμετροι εφαρμογής, καθώς και τα αποτελέσματα που προκύπτουν φαίνονται στον Πίνακα 3.1.

Γραμμές	Γραμμή Thara	Ποταμός Shenandoah	Ποταμός Mancos	Ακτογραμμή Cape Arago
<b>Μέσο Βήμα Ψηφιοποίησης*</b>	0.44	0.22	0.18	0.3
<b>Ακτίνα LLR*</b>	0.9	0.5	0.4	0.6
<b>Ακτίνες ALR*</b>	0.45, 0.9, 1.35 & 1.8	0.25, 0.5, 0.75 & 1	0.2, 0.4, 0.6 & 0.8	0.3, 0.6, 0.9 & 1.2
<b>Γνωστά Κρίσιμα Σημεία</b>	45	53	40	53
<b>LLR</b>				
<b>Εντοπισμένα</b>	45	65	88	63
<b>Κοινά</b>	41 (91%)	50 (94%)	38 (95%)	43 (81%)
<b>Πρόσθετα</b>	4	15	50	20
<b>ALR</b>				
<b>Εντοπισμένα</b>	45	59	72	49
<b>Κοινά</b>	40 (89%)	51 (96%)	35 (88%)	38 (72%)
<b>Πρόσθετα</b>	5	8	37	11

\* σε mm στο χαρτί

**Πίνακας 3.1.** Παράμετροι εφαρμογής του δείκτη LR και αποτελέσματα της σύγκρισης.

Παρατηρώντας τον Πίνακα 3.1, διαπιστώνεται πως τα αποτελέσματα από την εφαρμογή της μεθόδου LR είναι σε γενική συμφωνία με αυτά των δύο σχετικών μελετών (Marino 1979, Thara 1987). Με εξαίρεση την ακτογραμμή Cape Arago, τόσο με την εφαρμογή του LLR, όσο και του ALR, το ποσοστό επιτυχίας στον εντοπισμό των κρίσιμων σημείων που παρουσιάζονται στις δύο μελέτες αγγίζει ή ξεπερνάει το 90%.

Εντούτοις, υπάρχουν ορισμένες περιπτώσεις απόκλισης, στις οποίες εντοπίζονται ως κρίσιμα, σημεία τα οποία δεν χαρακτηρίζονται ως τέτοια στις αντίστοιχες μελέτες, και κυρίως σε αυτή της Marino. Τα σημεία αυτά αναφέρονται ως 'πρόσθετα' στον Πίνακα 3.1. Η απόκλιση αυτή παρατηρείται, κυρίως, στη γραμμή του ποταμού Mancos. Το LR αποτελεί μια αυτοματοποιημένη μέθοδο εντοπισμού κρίσιμων σημείων. Εκτιμά την καμπυλότητα κάθε σημείου με βάση κοινά κριτήρια και περιορισμούς. Δεν υπάρχει η δυνατότητα απόρριψης σημείων που καλύπτουν αυτούς τους κανόνες, αλλά δεν είναι κρίσιμα. Για παράδειγμα, η μέθοδος του δείκτη LR εντοπίζει μικρά, απομονωμένα σπασίματα της γραμμής, τα οποία οι ερωτηθέντες της Marino ενδεχομένως να μην μπορούσαν να διακρίνουν. Συγχρόνως, όμως, η Marino θέτει ένα ανώτατο όριο στον αριθμό σημείων που τα υποκείμενα της έρευνας πρέπει να επιλέξουν. Ο περιορισμός αυτός επιτρέπει να εικάζουμε πως μέρος των 'πρόσθετων' σημείων θα μπορούσαν να θεωρηθούν κρίσιμα, εάν επιτρεπόταν η επιλογή μεγαλύτερου ή απεριόριστου αριθμού σημείων. Παραδείγματος χάριν, με την εφαρμογή του δείκτη LR εντοπίζονται όλες οι κορυφές συνεχών διακυμάνσεων της γραμμής. Στη μελέτη της Marino διατηρείται μέρος αυτών. Θεωρώντας ως έγκυρο μέτρο σύγκρισης τη μελέτη της Marino, καταλήγουμε στο συμπέρασμα πως ορισμένα εκ των 'πρόσθετων' σημείων θα μπορούσαν να μην θεωρηθούν κρίσιμα, αφού βρίσκονται σε θέσεις χαμηλής σημαντικότητας ή προέρχονται από ατέλειες του δείκτη LR. Παρόλα αυτά, ένα μέρος των 'πρόσθετων' σημείων μπορούν να θεωρηθούν ως κρίσιμα, συγκρίνοντάς τα με τα κρίσιμα σημεία που βρέθηκαν σε παρόμοιες περιοχές των γραμμών.

Εξετάζοντας τον συνολικό αριθμό των κρίσιμων σημείων που εντοπίζονται από την εφαρμογή των δεικτών LLR και ALR, εξάγονται σημαντικά συμπεράσματα για την ποιότητά τους. Με μια πρώτη ματιά, γίνεται φανερό πως το ποσοστό επιτυχίας των δύο δεικτών στον εντοπισμό των κρίσιμων σημείων που παρουσιάζονται στις μελέτες των Marino και Thapa είναι σχεδόν το ίδιο (ο δείκτης LLR παρουσιάζει ελαφρώς καλύτερα αποτελέσματα στις τρεις από τις τέσσερις γραμμές). Εκτός ελαχίστων εξαιρέσεων, τα κρίσιμα σημεία που εντοπίζονται είναι κοινά. Η διαφοροποίηση μεταξύ των δύο δεικτών έγκειται στον συνολικό αριθμό σημείων που επιλέγονται ως κρίσιμα. Παρατηρείται, πως ο αριθμός των κρίσιμων σημείων που προκύπτουν από την εφαρμογή του ALR είναι σημαντικά μικρότερος. Εξαιρεση αποτελεί μόνο η γραμμή του Thapa, όπου ανιχνεύεται ίσος αριθμός κρίσιμων σημείων από τους δύο δείκτες. Εξετάζοντας τα επιπλέον κρίσιμα σημεία που εντοπίζονται από την εφαρμογή του δείκτη LLR, παρατηρείται πως σημαντικό μέρος αυτών αποδίδουν τοπικές εξάρσεις της γραμμής, ελάσσονος σημαντικότητας για τη διατήρηση, συνολικά, της μορφής της. Για παράδειγμα, πολλά από τα σημεία αυτά εντοπίζονται σε πολύ κοντινές θέσεις 'κοινών' κρίσιμων σημείων ή σε θέσεις ομαλών ή μέσων διαταραχών που προηγούνται ή έπονται απότομων σπασμάτων της γραμμής. Οι τιμές του δείκτη ALR, σε αυτές τις περιπτώσεις, παρουσιάζονται ομαλοποιημένες. Δηλαδή, δεν εμφανίζουν τοπικό μέγιστο, αλλά είναι αύξουσες ή φθίνουσες, προς ή από το τοπικό μέγιστο του σημείου μεγαλύτερης μεταβολής κλίσης. Εμφανίζονται, βέβαια, (ελάχιστες) περιπτώσεις που, λόγω της συγκεκριμένης συμπεριφοράς των τιμών του ALR, δεν επιλέγονται κρίσιμα σημεία που εμφανίζονται σε αντίστοιχες θέσεις. Παράλληλα, η εκτίμηση της καμπυλότητας της ευρύτερης περιοχής γύρω από κάθε σημείο, μέσω της εφαρμογής του δείκτη ALR, βοηθά στον ευκολότερο εντοπισμό κρίσιμων σημείων που βρίσκονται σε απομονωμένες, ομαλές διακυμάνσεις της γραμμής ή θέσεις μικρής αλλαγής κλίσης. Τέλος, παρατηρείται πως, μέσω του δείκτη ALR, στις μεγάλες, ανοιχτές καμπύλες, τα κρίσιμα σημεία εντοπίζονται ακριβώς στην κορυφή αυτών (σε πολλές περιπτώσεις η χρήση του δείκτη LLR οδηγεί στον εντοπισμό γειτονικών σημείων).

Γραμμή	LLR	ALR
Γραμμή Thapa	1.061	1.059
Ποταμός Shenandoah	1.048	1.046
Ποταμός Mancos	1.048	1.049
Ακτογραμμή Cape Arago	1.040	1.048

**Πίνακας 3.2.** Τιμές του δείκτη LR σε κοινά κρίσιμα σημεία των μελετών των Marino και Thapa.

Στον Πίνακα 3.2 παρουσιάζονται οι τιμές του δείκτη LR, πάνω από τις οποίες εντοπίζονται τα κοινά κρίσιμα σημεία με τις μελέτες των Marino και Thapa. Επιβεβαιώνεται, με τον τρόπο αυτό, ο ορισμός του ορίου επιλογής στην τιμή 1,04 και κατ' επέκταση η ιδέα πως κρίσιμα σημεία εντοπίζονται σε θέσεις με λόγο  $B:H < 11:1$ .

Παράλληλα, θετικό μπορεί να χαρακτηριστεί το γεγονός πως ένα μεγάλο μέρος των θέσεων που επιλέχθηκαν από περισσότερους ερωτηθέντες κατά την εμπειρική διαδικασία της Marino, αντιστοιχούν σε υψηλές τιμές του δείκτη LR (ομάδες B και



Γ). Επιβεβαιώνεται, δηλαδή, η αρχή πως θέσεις υψηλής καμπυλότητας νοούνται ως σημαντικές από τον άνθρωπο. Η επιλογή, όμως, από τους ερωτηθέντες σημαντικών σημείων που αντιστοιχούν σε χαμηλές τιμές του δείκτη LR, καθώς και η ύπαρξη αρκετών σημείων υψηλών τιμών του LR που δεν θεωρούνται κρίσιμα αποτρέπει την θέσπιση μιας γενικής αρχής συσχετισμού τιμής LR και σημαντικότητας του σημείου.

Μια διεξοδικότερη, ποιοτική, ανάλυση των αποτελεσμάτων του δείκτη LR, σε σχέση με τη μορφολογία των απεικονιζόμενων γραμμών, οδηγεί στα παρακάτω:

- Λόγω της γεωμετρικής μορφής της θεωρητικής γραμμής του Thara, η διαβάθμιση των τιμών του δείκτη LR είναι σε πλήρη συμφωνία με τη μορφή της γραμμής, ειδικά όταν εξετάζεται συνολικά. Εξετάζοντας τα αποτελέσματα λεπτομερέστερα, παρατηρείται ότι τα κρίσιμα σημεία της ομάδας A αντιστοιχούν, κυρίως, στα μικρά σπασίματα της γραμμής, σε ομαλές κλίσεις (γωνίες θλάσης μέχρι  $120^\circ$ ) και σε συνεχείς, μικρές καμπύλες της γραμμής. Η ομάδα B αποτελείται κυρίως από τις συνεχείς, μεγάλες διακυμάνσεις και τις έντονες αλλαγές κλίσεων. Τέλος, η ομάδα Γ αντιστοιχεί σε θέσεις πολύ έντονων σπασμάτων της κλίσης της γραμμής, σε οξείες ακίδες και σε μεσαίες και μεγάλες διακυμάνσεις. Τα κρίσιμα σημεία που δεν ανιχνεύονται από την εφαρμογή των δεικτών LLR και ALR είναι κοινά και εντοπίζονται σε θέσεις σχεδόν μηδενικής καμπυλότητας της γραμμής. Τα 'πρόσθετα' σημεία που λαμβάνονται ως κρίσιμα από την εφαρμογή του δείκτη ALR βρίσκονται σε μικρές, απότομες καμπύλες της γραμμής (με λόγο B:H=5.5:1) και σε θέσεις κορυφών συνεχών, μικρών κυματισμών της γραμμής, οι οποίες θα μπορούσαν να θεωρηθούν κρίσιμες (γειτονικές λαμβάνονται ως κρίσιμες από τον Thara). Αντίστοιχα ισχύουν και για τα 'πρόσθετα' σημεία που επιλέγονται από τον δείκτη LLR, εκτός ενός που έχει να κάνει με την ευαισθησία του δείκτη στα τοπικά χαρακτηριστικά της γραμμής. Παρόλα αυτά, είναι σημαντικό το γεγονός, πως ο συνολικός αριθμός κρίσιμων σημείων που εντοπίζονται από τις δύο μεθόδους (Thara και LR) είναι ακριβώς ο ίδιος (45). Γενικότερα, τα δύο μαθηματικά μοντέλα παρουσιάζουν μια άκρως ικανοποιητική ταύτιση στην επιλογή κρίσιμων σημείων.

- Αναφορικά με την ημιτονοειδούς μορφής γραμμή του ποταμού Shenandoah, το ποσοστό επιτυχίας στον εντοπισμό των κρίσιμων σημείων που παρουσιάζονται στη μελέτη της Marino αγγίζει το απόλυτο, τόσο με την εφαρμογή του δείκτη LLR (50/53), όσο και του δείκτη ALR (51/53). Διατηρούνται όλες οι κορυφές των μεγάλων καμπυλών και των απότομων σπασμάτων που χαρακτηρίζουν τη γραμμή. Μάλιστα, οι τιμές δείκτη LR που αντιστοιχούν στις θέσεις αυτές εντάσσονται στις ομάδες B και Γ και αποτελούν την πλειοψηφία. Οι υπόλοιπες, που ανήκουν στην ομάδα A, αντιστοιχούν σε ελαφρές κλίσεις και απότομα, μικρά σπασίματα της γραμμής. Η κατηγοριοποίηση του δείκτη ανάλογα με το απεικονιζόμενο σχήμα επαληθεύεται, ενώ, αξίζει να σημειωθεί πως εξετάζοντας τις καμπύλες όχι τοπικά, αλλά στο σύνολό τους πάνω στη γραμμή, παρατηρείται πως οι τιμές του δείκτη αυξάνουν όσο πιο 'οξείες' είναι αυτές. Τα σημεία που δεν εντοπίζονται από τον δείκτη LLR βρίσκονται σε θέσεις χαμηλής μεταβολής της καμπυλότητας με λόγο B:H>11:1 (δεν εντοπίζεται ούτε από τον δείκτη ALR) και σε θέσεις απομονωμένων, ομαλών καμπυλών. Το ένα επιπλέον σημείο που δεν εντοπίζεται από τον δείκτη ALR βρίσκεται σε θέση που ακολουθεί απότομη καμπύλη, όπου η τιμές του εμφανίζονται φθίνουσες. Από την εφαρμογή των δεικτών LLR και ALR εντοπίζονται σημεία που, κατά τη μελέτη της Marino, δεν θεωρούνται κρίσιμα. Ο αριθμός τους

δεν είναι μεγάλος, όμως παρουσιάζουν ιδιαίτερο ενδιαφέρον. Ένα μικρό μέρος των 'πρόσθετων' σημείων εντοπίζεται σε θέσεις μικρών, απότομων σπασιμάτων της γραμμής, αντιστοιχούν σε χαμηλές τιμές των δεικτών LR (περίπου 1,05) και δεν γίνονται εύκολα αντιληπτά από τους ερωτηθέντες στη μελέτη της Marino. Τα υπόλοιπα 'πρόσθετα' σημεία που εντοπίζονται από την εφαρμογή του δείκτη LLR αποδίδουν τοπικές διαταραχές της γραμμής. Η πλειοψηφία τους εντοπίζεται σε άμεσα γειτονικές θέσεις κρίσιμων σημείων ή θέσεις ελάσσονος σημασίας για την απόδοση της μορφής της γραμμής. Με την εφαρμογή του δείκτη ALR, σχεδόν όλα τα παραπάνω σημεία εξαλείφονται. Στην γραμμή του ποταμού Shenandoah φαίνεται καθαρά πώς η εξέταση της καμπυλότητας σε ένα εύρος περιοχών γύρω από κάθε σημείο από τον δείκτη ALR βοηθάει στην απαλοιφή αρκετών 'περιττών' κρίσιμων σημείων που εντοπίζονται από τον δείκτη LLR. Συγχρόνως, όμως, στην εν λόγω γραμμή παρουσιάζεται μία σπανίως εμφανιζόμενη ατέλεια του δείκτη ALR. Τα πέντε από τα οκτώ 'πρόσθετα' σημεία ανήκουν στην ομάδα Γ και εντοπίζονται σε γειτονικές θέσεις κορυφών μεγάλων καμπυλών που θεωρούνται κρίσιμες. Το πρόβλημα δημιουργείται όταν, λόγω της μορφής της καμπύλης, η περιοχή εξέτασης για την ακτίνα R4 ξεπερνά τα όρια του 'τοπικού' και εκφράζει περισσότερο γενικευμένα χαρακτηριστικά της μορφής της γραμμής. Έτσι, ο δείκτης  $LR_{R4}$  εμφανίζεται πολύ μεγάλος και αυξάνει την τιμή του δείκτη ALR. Το ζήτημα αυτό, αν και σπανίως εμφανιζόμενο, χρήζει διεξοδικότερης διερεύνησης.

- Όσον αφορά την μεγάλης πολυπλοκότητας γραμμή του ποταμού Mancos, η πλειοψηφία των σημείων που θεωρούνται κρίσιμα στη μελέτη της Marino εντοπίζεται από τους δείκτες LR (38/40 χρησιμοποιώντας τον δείκτη LLR και 35/40 χρησιμοποιώντας τον δείκτη ALR). Τα κρίσιμα σημεία που δεν επιλέγονται εντοπίζονται σε θέσεις καμπυλών, οι οποίες γειτνιάζουν άμεσα με θέσεις καμπυλών μικρότερου λόγου B:H (άρα μεγαλύτερης τιμής LR) που θεωρούνται κρίσιμες. Η τιμή του δείκτη LR στις θέσεις αυτές δεν παρουσιάζει τοπικό μέγιστο, αλλά είναι φθίνουσα ή αύξουσα. Αυτό που χαρακτηρίζει την εφαρμογή του δείκτη LR στην γραμμή του ποταμού Mancos είναι ο εντοπισμός μεγάλου αριθμού 'πρόσθετων' σημείων. Το γεγονός αυτό οφείλεται στη μορφολογία της. Πρόκειται για μία γραμμή μεγάλης πολυπλοκότητας, με συνεχείς, μεγάλες ή μικρές κυρτώσεις. Η δομή του δείκτη LR οδηγεί στον εντοπισμό μεγάλου μέρους αυτών (όσων ο λόγος B:H είναι μικρότερος του 11:1). Κατά την έρευνα της Marino, όμως, περίπου οι μισές χαρακτηρίζονται ως κρίσιμες. Με εξαίρεση δύο, η πλειοψηφία των κρίσιμων αυτών σημείων αντιστοιχούν σε τιμές του δείκτη LR μεγαλύτερες του 1,08. Συγχρόνως, ο κύριος όγκος των 'πρόσθετων' σημείων ανήκει στην ομάδα Α, και λιγότερα στις ομάδες Β και Γ. Σε μια γραμμή μεγάλης πολυπλοκότητας σαν αυτή του ποταμού Mancos, ο περιορισμός στον αριθμό των προς επιλογή σημείων που θέτει η Marino στην έρευνά της θα μπορούσε να αποτελέσει μια εύκολη ερμηνεία της απόκλισης του αριθμού των κρίσιμων σημείων που παρουσιάζεται μεταξύ των δύο διαδικασιών. Δεχόμενοι, όμως, τη μελέτη της Marino ως βάση σύγκρισης και αναλύοντας τα αποτελέσματα, προκύπτει ένα βασικό χαρακτηριστικό της προτεινόμενης μεθόδου. Ο δείκτης LR παρουσιάζει ευαισθησία σε θέσεις μεγάλης πολυπλοκότητας της γραμμής. Όπως αποδεικνύεται από την εφαρμογή στην γραμμή του Ποταμού Mancos, σε μια τέτοια περίπτωση, μόνο οι θέσεις μεγάλης μεταβολής της καμπυλότητας θα πρέπει να χαρακτηρίζονται σαν κρίσιμες. Έτσι εξηγείται το γεγονός ότι τα περισσότερα 'πρόσθετα' σημεία ανήκουν στις χαμηλές

τιμές της ομάδας A ή σε τιμές σημείων που γειτνιάζουν με θέσεις μεγαλύτερης μεταβολής κλήσης. Απαιτείται, δηλαδή, ένα περαιτέρω 'φιλτράρισμα' των κρίσιμων σημείων. Το ζήτημα αυτό χρήζει διεξοδικότερης ανάλυσης σε μελλοντική έρευνα.

- Αναφερόμενοι στην ακτογραμμή Cape Arago, οι τιμές του δείκτη LR αντιστοιχούν άμεσα στη μορφολογία της γραμμής. Σε αυτή τη γραμμή, όμως, εμφανίζεται το μικρότερο ποσοστό επιτυχίας στον εντοπισμό των κρίσιμων σημείων που παρουσιάζονται στη μελέτη της Marino (43/53 χρησιμοποιώντας τον δείκτη LLR και 38/53 τον δείκτη ALR). Το γεγονός αυτό οφείλεται, εν μέρει, στη μορφολογία της γραμμής (περιλαμβάνει τμήματα μεγάλης πολυπλοκότητας, αλλά και ευθύγραμμα τμήματα). Πολλά από τα κρίσιμα σημεία που παρουσιάζονται στη μελέτη της Marino εντοπίζονται σε θέσεις χαμηλής ή μηδενικής καμπυλότητας, όπου οι τιμές του δείκτη LR παρουσιάζονται μικρότερες από το όριο επιλογής. Παράλληλα, σε θέσεις μεγάλης πολυπλοκότητας επιλέγονται από τους ερωτηθέντες διαδοχικά, άμεσα γειτονικά σημεία ως κρίσιμα. Μέρος αυτών δεν εντοπίζεται από την εφαρμογή του δείκτη LR, αφού οι τιμές του παρουσιάζονται φθίνουσες ή αύξουσες, προς ή από το τοπικό μέγιστο αντίστοιχα. Όπως αναφέρεται παραπάνω, το φαινόμενο αυτό παρουσιάζεται κυρίως στον δείκτη ALR. Με την εφαρμογή του δείκτη LR εντοπίζονται σημεία που δεν χαρακτηρίζονται σαν κρίσιμα στη μελέτη της Marino. Ο αριθμός τους δεν είναι μεγάλος. Μάλιστα, μέσω του δείκτη ALR ο συνολικός αριθμός των κρίσιμων σημείων που εντοπίζονται είναι μικρότερος από αυτών της Marino, χωρίς να μεταβάλλεται η συνολική εικόνα της γραμμής. Τα 'πρόσθετα' σημεία ανήκουν και στις τρεις ομάδες των τιμών του LR. Εντοπίζονται είτε σε μικρές, ομαλές διαταραχές της γραμμής με μεγάλο λόγο B:H, είτε σε θέσεις απότομων μεταβολών της καμπυλότητας ή σε κορυφές συνεχών καμπύλων (μέρος των οποίων θεωρούν κρίσιμες οι ερωτηθέντες της Marino). Μέρος των 'πρόσθετων' σημείων που εντοπίζονται από το δείκτη LLR προέρχονται από την ευαισθησία του δείκτη στον εντοπισμό τοπικών ανωμαλιών της γραμμής.

Πρέπει να αναφερθεί ότι η μέθοδος του δείκτη LR εφαρμόστηκε και σε μη εξομαλυσμένα δεδομένα. Το συγκεκριμένο πείραμα έδειξε ότι εντοπίζονται τα ίδια ακριβώς κρίσιμα σημεία. Αυτό οφείλεται κυρίως στο γεγονός ότι οι παράμετροι της διαδικασίας διόρθωσης και εξομάλυνσης επιλέχτηκαν, ώστε να είναι κοντά στην ανοχή της ψηφιοποίησης της αναλογικής γραμμής, καθώς επίσης και στον ορισμό του κατώτατου ορίου των τιμών του δείκτη LR.

### **3.1.6.2 Σύγκριση με μέθοδο οφθαλμικών καταγραφών (Eye-Tracking)**

Στη μελέτη του σχετικά με τις αρχές οπτικής αντίληψης, ο Attneave (1954) αναφέρει ότι κατά τη διάρκεια της παρατήρησης ενός γραφικού αντικειμένου, ο ανθρώπινος εγκέφαλος λαμβάνει έναν τεράστιο αριθμό ερεθισμάτων, τα οποία μεταφέρουν μια αντίστοιχα μεγάλη ποσότητα πληροφορίας. Η πληροφορία αυτή, αφενός δεν μπορεί να αποθηκευθεί και να χρησιμοποιηθεί και αφετέρου, ένα ελάχιστο μέρος της είναι απαραίτητο για την ταυτοποίηση του αντικειμένου. Ο Attneave επισημαίνει ότι κάθε θέση που διαμορφώνει ένα γραφικό αντικείμενο μεταφέρει διαφορετική ποσότητα πληροφορίας στον παρατηρητή. Καταλήγει στο συμπέρασμα ότι η πληροφορία που συγκεντρώνονται κατά μήκος του περιγράμματος ενός γραφικού αντικειμένου και

ιδιαίτερα σε θέσεις μεγάλης αλλαγής κλίσης είναι διακριτή και αρκεί από μόνη της για να περιγραφεί το σχήμα του. Έτσι, ο ανθρώπινος εγκέφαλος πραγματοποιεί μια σειρά ενεργειών (συνειδητών ή μη), με τρόπο ώστε να επιλέγονται και να αποθηκεύονται τα ερεθίσματα που βρίσκονται στις διακριτές αυτές θέσεις και να απομακρύνονται τα περιττά.

Λαμβάνοντας υπόψη ότι η γενίκευση της γραμμής είναι μια διαδικασία κατά την οποία ένας χαρτογράφος διατηρεί μόνο τα απαραίτητα σημεία ώστε να διαμορφώσει τη παράγωγη γραμμή, μια ενδιαφέρουσα εφαρμογή της θεωρίας του Attneave είναι ο εντοπισμός των θέσεων αυτών, όταν αναλύονται οι οφθαλμικές κινήσεις ενός χρήστη, ενώ σαρώνει μια χαρτογραφική γραμμή που προβάλλεται σε μια οθόνη υπολογιστή.

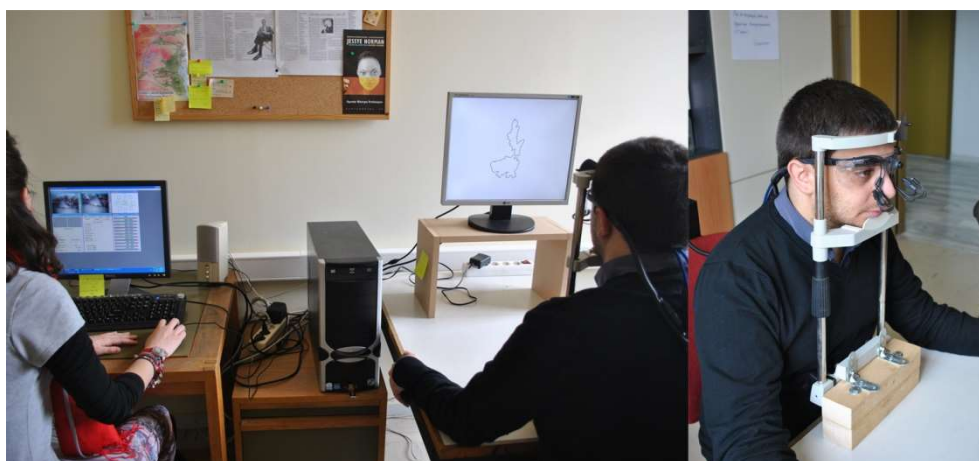
Πάνω σε αυτό το πλαίσιο, διεξήχθη μία εμπειρική έρευνα για να εξεταστεί το κατά πόσο οι μετρήσεις των οφθαλμικών κινήσεων κατά την παρατήρηση των συγκεκριμένων χαρτογραφικών γραμμών κάτω από μια απλή διαδικασία σάρωσης σχετίζονται με τις θέσεις που εντοπίζονται κρίσιμα σημεία. Τα ερεθίσματα του πειράματος αποτελούνται από τις γραμμές που χρησιμοποιούνται ως αρχεία αναφοράς στη διαδικασία ελέγχου της αξιοπιστίας του δείκτη αναλογίας LR (βλ. κεφ. 3.1.6.1, Σχήμα 3.10), καθώς και την ακτογραμμή της νήσου Περιστέρα (βλ. κεφ. 5.1). Τα υποκείμενα καλούνται να σαρώσουν αυτές τις χαρτογραφικές γραμμές σε έναν προκαθορισμένο χρόνο. Σκοπός του πειράματος είναι να εξεταστεί αν η παρατήρηση των χαρτογραφικών γραμμών σχετίζεται με τις θέσεις που εντοπίζονται τα κρίσιμα σημεία μέσω της εφαρμογής του δείκτη αναλογίας LR, και πιο συγκεκριμένα του δείκτη αναλογίας ALR. Η εμπειρική έρευνα σχεδιάστηκε και εκτελέστηκε στο Εργαστήριο Χαρτογραφίας.

Η διαδικασία ανάγνωσης ενός χάρτη είναι μια περίπλοκη διαδικασία, η οποία εξαρτάται από διάφορους παράγοντες. Αυτοί οι παράγοντες ποικίλλουν ανάλογα με τις συνθήκες προβολής και την εργασία που πρέπει να ολοκληρωθεί. Σε συνθήκες ελεύθερης θέασης οι αναγνώστες του χάρτη αντιδρούν στις φυσικές ιδιότητες των ερεθισμάτων (Castner and Eastman 1984), ενώ κατά τη διάρκεια μιας ειδικής ανάγνωσης, ο σκοπός της ανάγνωσης έχει μεγάλη σημασία στις γνωστικές και αντιληπτικές δραστηριότητες (Steinke 1985). Κατά τη διάρκεια της απόδοσης μιας οπτικής αναζήτησης συμβαίνει μια 'από πάνω προς τα κάτω' διαδικασία. Ο έλεγχος προσοχής της από πάνω προς τα κάτω επεξεργασίας καθοδηγείται από παράγοντες 'εσωτερικούς' για τον παρατηρητή και δεν έχει σχέση με το ερέθισμα (Awh et al. 2012).

Στη βιβλιογραφία στα πεδία της ψυχολογίας και της οπτικής αντίληψης προτείνονται δύο διαφορετικές μεθοδολογίες για τη μέτρηση της οπτικής αναζήτησης. Σύμφωνα με την πρώτη, οι επιδόσεις ενός υποκειμένου κατά τη διάρκεια παρατήρησης ερεθισμάτων μπορεί να μετρηθεί μέσω του χρόνου αντίδρασης και της ευστοχίας των απαντήσεων σε μια διαδικασία αναζήτησης (Greene and Wolfe 2011). Η δεύτερη μέθοδος χρησιμοποιεί την ανάλυση των οφθαλμικών κινήσεων του υποκειμένου (Zelinsky 1996). Η εμπλουτισμένη μεθοδολογία σύλληψης και ανάλυσης της θέσης που κοιτάζει ένα άτομο στο χώρο ('σημείο παρακολούθησης') ονομάζεται οφθαλμική καταγραφή (eye tracking) (Young and Sheena 1975). Και οι δύο μεθοδολογίες, η μέτρηση του χρόνου αντίδρασης και της ακρίβειας (Lloyd 1997,

Michaelidou et al. 2005) και οι οφθαλμικές καταγραφές (Coltiken et al. 2009, Ooms et al. 2012), έχουν ήδη χρησιμοποιηθεί σε χαρτογραφικά πειράματα. Σε μια σχετική εμπειρική έρευνα, ο Wood (1995) μελετά την απλοποίηση γραμμών με μετρήσεις οφθαλμικών καταγραφών. Ένας αριθμός χαρτών με τέσσερα επίπεδα λεπτομέρειας παρουσιάζονται σε μια ομάδα υποκειμένων. Από τα υποκείμενα ζητείται να εκτελέσουν τρεις προκαθορισμένες εργασίες: να μετρήσουν, να υπολογίσουν και να συγκρίνουν. Κατά τη διάρκεια του πειράματος καταγράφονται οι οφθαλμικές κινήσεις χρησιμοποιώντας ένα σύστημα παρακολούθησης ματιών (eye tracking system). Με βάση τη στατιστική ανάλυση των μετρήσεων, ο Wood συμπεραίνει ότι στην διαδικασία καταμέτρησης, ο αριθμός και η διάρκεια των προσηλώσεων μειώνονται καθώς μειώνεται το επίπεδο λεπτομέρειας των χαρτών. Αντιθέτως, η μείωση των προσηλώσεων κατά τη διάρκεια των διαδικασιών υπολογισμού και σύγκρισης πρόκυψε στατιστικά ασήμαντη.

Στην εμπειρική έρευνα, η μέθοδος που χρησιμοποιείται για τη μέτρηση της αντίδρασης των συμμετεχόντων είναι η 'οφθαλμική καταγραφή' (eye tracking). Συγκεκριμένα, χρησιμοποιείται το σύστημα καταγραφής View Point Eye Tracker® της Arrington Research. Οι οφθαλμικές κινήσεις καταγράφονται με συχνότητα δειγματοληψίας 60Hz, η οποία μεταφράζεται σε μία χρονική απόσταση περίπου 16,67msec μεταξύ δύο διαδοχικών σημείων καταγραφής. Η ακρίβεια του Eye Tracker ποικίλλει μεταξύ 0,25-1.00 βαθμούς οπτικού τόξου, όταν η ανάλυση της εγγραφής είναι 0,15 μοίρες του οπτικού τόξου. Για την προβολή ερεθισμάτων χρησιμοποιείται μια οθόνη 19 ιντσών. Η απόσταση μεταξύ υποκειμένου και οθόνης προβολής των ερεθισμάτων είναι 60cm. Η εικόνα 3.1 απεικονίζει το 'σύστημα οφθαλμικών καταγραφών' στις συνθήκες μέτρησης του πειράματος. Ο εξοπλισμός του συστήματος περιγράφεται λεπτομερέστερα σε σχετική μελέτη της ομάδας του Krassanakis (Krassanakis et al. 2011).



**Εικόνα 3.1.** Δύο όψεις της ρύθμισης παραμέτρων του συστήματος οφθαλμικών καταγραφών που χρησιμοποιείται για το πείραμα.

Οι πέντε γραμμές που συνθέτουν τα ερεθίσματα της έρευνας είναι η ακτογραμμή Cape Arago σε κλίμακα 1:25.000 περίπου, ο ποταμός Mancos σε κλίμακα 1:25.000 περίπου και ο ποταμός Shenandoah σε κλίμακα 1:100.000 περίπου (Marino 1979), η γραμμή ελέγχου του Thara (1988) (βλ. Σχήμα 3.10) και η ακτογραμμή της νήσου Περιστέρα σε κλίμακα 1:50.000 (βλ. κεφ. 5.1). Κάθε γραμμή παρουσιάζεται στα

υποκείμενα σε τέσσερις περιστροφές (0, 45, 90, 135 μοίρες), με αυτή των 0 μοιρών να αντιστοιχεί στην αρχική γραμμή. Έτσι, τα ερεθίσματα του πειράματος αποτελούνται από συνολικά 20 εικόνες. Το πάχος κάθε γραμμής διαμορφώθηκε λαμβάνοντας υπόψη την απόσταση των 60cm μεταξύ του υποκειμένου και της οθόνης προβολής. Τα ερεθίσματα έχουν ανάλυση 1280x1024 pixels. Η διάρκεια προβολής της κάθε γραμμής καθορίστηκε μετά από ένα προ-πείραμα με λίγα υποκείμενα, κατά το οποίο καλούνται να σαρώσουν τις γραμμές που παρουσιάζονται, με τη διάρκεια κάθε σάρωσης να καταγράφεται. Η διάρκεια προβολής των γραμμών ορίζεται έτσι, ώστε να εξασφαλίζεται ότι κάθε υποκείμενο είναι σε θέση να εκτελέσει μια πλήρη σάρωση κάθε γραμμής. Στον Πίνακα 3.3 παρουσιάζεται η διάρκεια προβολής που επιλέχθηκε για κάθε γραμμή.

<b>Γραμμή</b>	<b>Διάρκεια δοκιμής (msec)</b>
<b>Ακτογραμμή Cape Arago</b>	10.000
<b>Ποταμός Mancos</b>	10.000
<b>Ποταμός Shenandoah</b>	12.000
<b>Γραμμή Thapa</b>	10.000
<b>Ακτογραμμή Ν. Περιστεράς</b>	15.000

**Πίνακας 3.3:** Διάρκεια προβολής πειραματικών γραμμών.

Επιπλέον, δημιουργήθηκαν δύο ακόμη ερεθίσματα με την προβολή πέντε σταθερών στόχων πριν και μετά τις πειραματικές γραμμές, προκειμένου να ελεγχθεί η ακρίβεια του συστήματος καταγραφής σε σχέση με τη θέση των συμμετεχόντων. Μια υψηλή τιμή σφάλματος υποδεικνύει την πιθανή κίνηση ενός υποκειμένου κατά τη διάρκεια της πειραματικής διαδικασίας. Μετά τον καθορισμό μιας τιμής-κατωφλίου σφάλματος, τα δεδομένα που προέρχονται από υποκείμενα που δεν πληρούν αυτόν τον περιορισμό δεν περιλαμβάνονται στη διαδικασία ανάλυσης.

Στο πείραμα συμμετείχαν 52 άτομα ηλικίας μεταξύ 18 και 30 ετών, τα οποία έχουν ακριβή όραση χωρίς την χρήση γυαλιών ή φακών επαφής. Κάθε υποκείμενο εκτελεί το πείραμα ξεχωριστά. Τα υποκείμενα καλούνται να σαρώσουν το σύνολο των πειραματικών γραμμών από την αρχή μέχρι το τέλος. Σε περίπτωση που ένα υποκείμενο τερματίσει τη σάρωση μίας γραμμής πριν από τη λήξη του χρόνου προβολής καλείται, είτε να επαναλάβει τη σάρωση, είτε να κοιτάξει σε ένα τυχαίο σημείο στην οθόνη για το υπολειπόμενο χρονικό διάστημα. Η βαθμονόμηση του συστήματος πραγματοποιείται με τη χρήση 9 σταθερών στόχων για κάθε συμμετέχοντα. Τέλος, δημιουργούνται τέσσερα σύνολα εικόνων και δοκιμάζονται σε τέσσερις ομάδες των 12, 13, 13 και 14 ατόμων αντίστοιχα. Κάθε σύνολο εικόνων περιέχει πέντε εικόνες που απεικονίζουν τις πέντε δοκιμαστικές γραμμές. Για να αποφευχθούν οι προκαταλήψεις, η περιστροφή και η ακολουθία της παρουσίασης είναι τυχαίες. Μετά την εξέταση της ακρίβειας των μετρήσεων κάθε υποκειμένου για τα ερεθίσματα των πέντε στόχων, επιλέγονται για την ανάλυση τα πρωτογενή δεδομένα των 40 ατόμων (10 ανά ομάδα) με σφάλμα μικρότερο από 50 pixels.

Τα δεδομένα που χρησιμοποιούνται στην ανάλυση είναι εκείνα που προκύπτουν από την πρώτη σάρωση κάθε γραμμής. Τα πρόσθετα δεδομένα που προέρχονται από επιπλέον απόπειρες σάρωσης δεν συμπεριλαμβάνονται στην περαιτέρω επεξεργασία. Τα βασικά μεγέθη που χρησιμοποιούνται για την ανάλυση της καταγραφής των οφθαλμικών κινήσεων κάθε υποκειμένου είναι οι προσηλώσεις

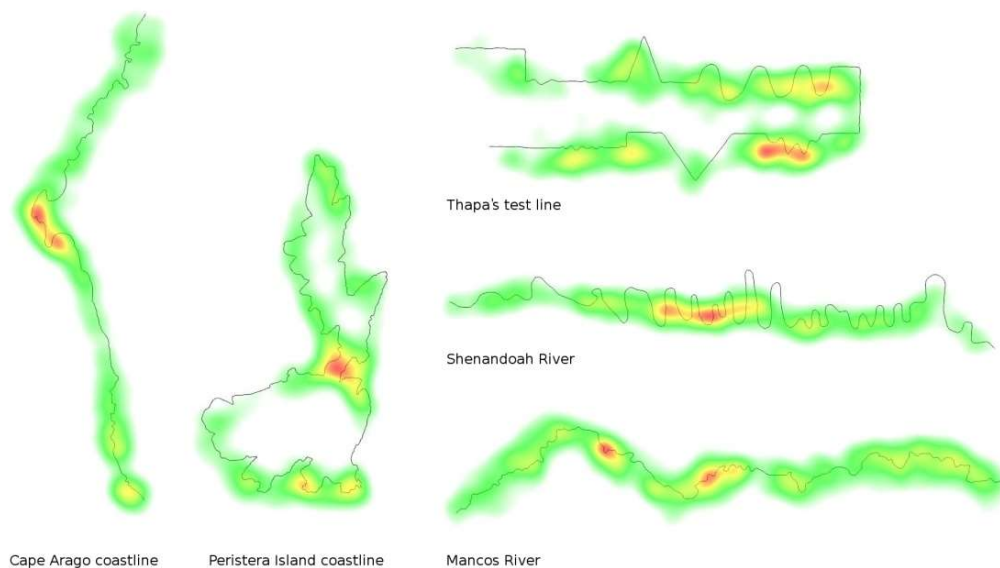
(fixations) και οι σακκαδικές κινήσεις (saccadic movements). Η πλήρης ακολουθία μεταξύ προσηλώσεων και σακκαδικών κινήσεων συνθέτουν τη διαδρομή σάρωσης της οπτικής αναζήτησης. Για την ανάλυση των πρωτογενών δεδομένων και την μετατροπή τους στα βασικά μεγέθη (προσηλώσεις και σακκαδικές κινήσεις) χρησιμοποιείται το λογισμικό Ogama (Voßkübler et al. 2008).

Οι παράμετροι υπολογισμού των προσηλώσεων ορίζονται ως εξής:

- η μέγιστη απόσταση που ένα σημείο μπορεί να απέχει από το μέσο σημείο προσήλωσης ώστε να θεωρείται μέρος αυτής ορίζεται σε 20pixel
- ο ελάχιστος αριθμός δειγμάτων που μπορεί να θεωρηθεί ότι απαρτίζουν μια προσήλωση ορίζεται σε 5
- το μέγεθος του δακτυλίου ανίχνευσης μίας προσήλωσης ορίζεται σε 31pixel. Το μέγεθος του δακτυλίου ορίζει το μέγεθος της ζώνης (buffer) που χρησιμοποιείται για τον εντοπισμό των προσηλώσεων.

Οι προσηλώσεις που εξάγονται συγκρίνονται με τα κρίσιμα σημεία που έχουν εντοπισθεί στις πέντε πειραματικές γραμμές μέσω της εφαρμογής του δείκτη αναλογίας μήκους ALR.

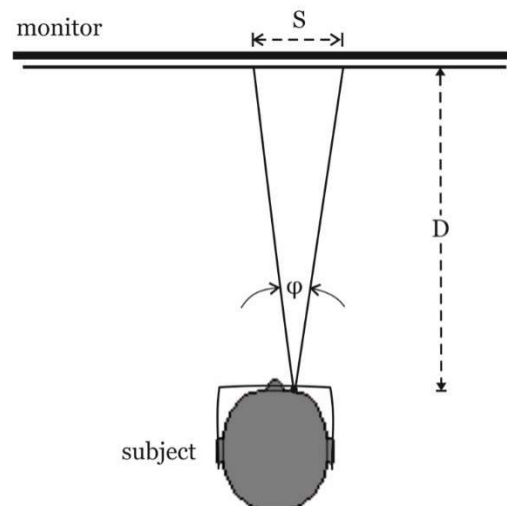
Επιπλέον, για όλες τις γραμμές παράγονται οι απεικονίσεις των χαρτών θερμότητας (Σχήμα 3.11). Ο χάρτης θερμότητας είναι μια τεχνική απεικόνισης που αναπαριστά διαφορετικά επίπεδα έντασης της παρατήρησης χρησιμοποιώντας χρώματα (Sprakon and Miniotas 2007). Τα χρώματα των χαρτών θερμότητας που παρουσιάζονται στο Σχήμα 3.11 μεταβάλλονται από πράσινο σε κόκκινο όσο αυξάνεται η πυκνότητα των προσηλώσεων. Παρατηρείται ότι οι προσηλώσεις είναι πυκνότερες σε περιοχές όπου δύο αντίθετα μέρη της γραμμής συγκρούονται ή σε θέσεις διαδοχικών διακυμάνσεων. Σημειώνεται ότι αυτές οι θέσεις χαρακτηρίζονται από υψηλή συγκέντρωση κρίσιμων σημείων που εντοπίζονται από την εφαρμογή του δείκτη ALR.



**Σχήμα 3.11:** Χάρτες θερμότητας των πέντε υπό μελέτη γραμμών.



Ένα σημαντικό ζήτημα που αφορά στην επεξεργασία των δεδομένων που συλλέγονται στο πείραμα μέσω του συστήματος οφθαλμικών καταγραφών έχει να κάνει με το γεγονός ότι η υποδιαίρεση τόσο των πρωτογενών δεδομένων, όσο και των εξαγόμενων προσηλώσεων είναι πιο χονδροειδείς από ότι τα σημεία που δομούν τις υπό μελέτη γραμμές. Το γεγονός οφείλεται στην ακρίβεια καταγραφής του συστήματος, σε συνδυασμό με το γεγονός ότι τα μάτια δεν είναι απόλυτα, αλλά σχετικά σταθερά κατά τη διάρκεια μιας οπτικής προσήλωσης. Για να ξεπεραστεί το πρόβλημα αυτό, δημιουργείται μια ζώνη εξέτασης (buffer zone) γύρω από κάθε γραμμή, αλλά και γύρω από τα κρίσιμα σημεία αυτής. Οι οφθαλμικές καταγραφές που εντοπίζονται εντός της ζώνης εξέτασης αντιστοιχούν στην παρατήρηση των υπό μελέτη στοιχείων (γραμμές, κρίσιμα σημεία). Το μέγεθος ( $S$ ) της ζώνης εξέτασης καθορίζεται λαμβάνοντας υπόψη την ακρίβεια του συστήματος καταγραφής ( $\phi$ ) και την απόσταση μεταξύ του υποκειμένου και των ερεθισμάτων ( $D$ ) (βλ. Σχήμα 3.12). Δεδομένου ότι η απόσταση  $D$  είναι 60cm κατά τη διάρκεια του πειράματος, το μέγεθος ( $S$ ) της ζώνης εξέτασης ορίζεται σε 20pixel.



**Σχήμα 3.12:** Η γεωμετρία του συστήματος υποκείμενο-οθόνη κατά τη διάρκεια του πειράματος.

Στον Πίνακα 3.4 παρουσιάζονται η διάρκεια παρατήρησης κάθε γραμμής, καθώς και των θέσεων που εντοπίζονται κρίσιμα σημεία, για όλα τα υποκείμενα (συνολικά) και ανά υποκείμενο (μέσος όρος), όπως προκύπτουν από την ανάλυση των πρωτογενών δεδομένων του πειράματος. Ένα πρώτο συμπέρασμα που προκύπτει είναι πως τα υποκείμενα αφιερώνουν ένα μεγάλο μέρος από το συνολικό χρόνο σάρωσης της γραμμής, προκειμένου να παρατηρούν τις θέσεις που περιέχουν κρίσιμα σημεία.



	Συνολική διάρκεια (msec)		Μέσος όρος & τυπική διάρκεια (msec)	
	Γραμμή	Κρίσιμα σημεία	Γραμμή	Κρίσιμα σημεία
<b>Ακτογραμμή Cape Arago</b>	364.993	154.000	9.125 ( $\pm 1.251$ )	3.850 ( $\pm 2.201$ )
<b>Ποταμός Mancos</b>	371.361	180.368	9.284 ( $\pm 904$ )	4.509 ( $\pm 2.645$ )
<b>Ποταμός Shenandoah</b>	447.567	256.317	11.189 ( $\pm 1.615$ )	6.408 ( $\pm 1.812$ )
<b>Γραμμή Thapa</b>	360.591	114.333	9.015 ( $\pm 1.140$ )	2.858 ( $\pm 1.250$ )
<b>Ακτογραμμή Ν. Περιστεράς</b>	510.005	265.000	12.750 ( $\pm 2.332$ )	6.625 ( $\pm 1.963$ )

**Πίνακας 3.4:** Διάρκεια σάρωσης κάθε γραμμής και θέσεων γύρω από κρίσιμα σημεία όπως προκύπτει από τα πρωτογενή δεδομένα.

Στη συνέχεια, λαμβάνει χώρα ανάλυση των προσηλώσεων που προκύπτουν από την πειραματική διαδικασία. Έτσι, υπολογίζεται η πυκνότητα χρονικής διάρκειας προσηλώσεων ανά pixel σε θέσεις γύρω από κάθε γραμμή, καθώς και σε θέσεις όπου εντοπίζονται κρίσιμα σημεία. Οι 'θέσεις' οριοθετούνται από τις αντίστοιχες ζώνες εξέτασης. Επιπλέον, για κάθε γραμμή υπολογίζεται η πυκνότητα διάρκειας προσηλώσεων ανά pixel σε θέσεις γύρω από μη κρίσιμα σημεία, ως η διαφορά των μεταξύ των παραπάνω υπολογισμένων τιμών. Τα αποτελέσματα, τα οποία αφορούν όλα τα υποκείμενα που συμμετέχουν στο πείραμα, παρουσιάζονται στον Πίνακα 3.5.

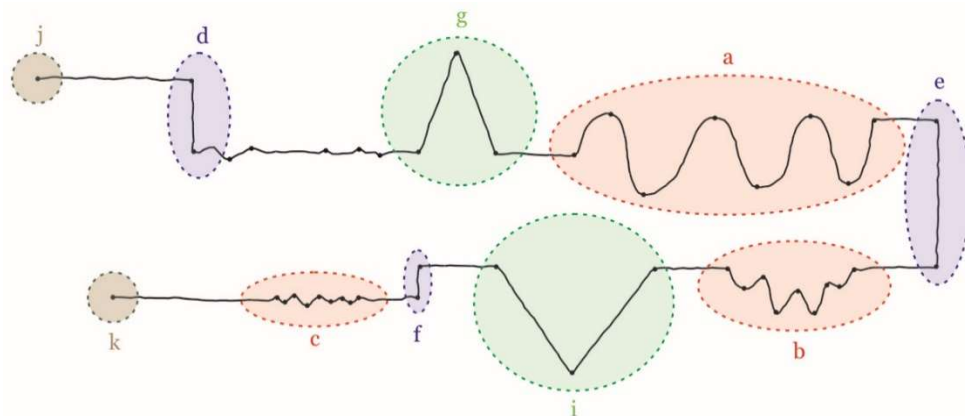
	Πυκνότητα διάρκειας προσηλώσεων (msec) ανά pixel		
	Γραμμή	Θέσεις κρίσιμων σημείων	Θέσεις μη κρίσιμων σημείων
<b>Ακτογραμμή</b>	3,98	4,35	3,14
<b>Ποταμός Mancos</b>	3,21	3,30	2,65
<b>Ποταμός Shenandoah</b>	4,45	4,48	4,35
<b>Γραμμή Thapa</b>	2,14	2,38	1,89
<b>Ακτογραμμή Ν. Περιστεράς</b>	3,88	4,33	3,01

**Πίνακας 3.5:** Πυκνότητα χρονικής διάρκειας προσηλώσεων ανά pixel γύρω από κάθε γραμμή και σε θέσεις γύρω από κρίσιμα και μη κρίσιμα σημεία.

Ένα γενικό συμπέρασμα που εξάγεται από την εξέταση του Πίνακα 3.5 είναι ότι τα υποκείμενα εστιάζουν την προσοχή τους περισσότερο σε θέσεις κρίσιμων σημείων κατά την παρατήρηση μιας γραμμής. Το γεγονός ισχύει για όλες τις υπό μελέτη γραμμές. Ενδιαφέρον παρουσιάζουν τα αποτελέσματα που αφορούν στη γραμμή του ποταμού Shenandoah. Παρατηρείται ότι η διάρκεια προσηλώσεων σε θέσεις κρίσιμων και μη κρίσιμων σημείων είναι σχεδόν ίση. Αυτό πιθανά οφείλεται στο σχεδόν ημιτονοειδές σχήμα που έχει η συγκεκριμένη γραμμή (βλ. Σχήμα 3.10). Τα κρίσιμα σημεία εντοπίζονται στην πλειοψηφία τους στις κορυφές των καμπυλών. Λαμβάνοντας υπόψη ότι οι καμπύλες είναι αρκετά κοντά η μία με την άλλη και ότι τα ενδιάμεσα ευθύγραμμα τμήματα είναι μικρού μήκους, η διάκριση μεταξύ κρίσιμων και μη κρίσιμων θέσεων γίνεται δύσκολη για τον παρατηρητή. Αντίθετα, τα κρίσιμα σημεία των υπολοίπων τεσσάρων γραμμών εντοπίζονται συνήθως σε πιο

ευδιάκριτες θέσεις. Έτσι, τα υποκείμενα προφανώς εστιάζουν ευκολότερα στις θέσεις των κρίσιμων σημείων.

Όπως αναφέρεται στην ενότητα 3.1.6.1, η γραμμή του Thara είναι ένα γεωμετρικό μοντέλο που περιέχει διάφορα αντιπροσωπευτικά δείγματα σχημάτων που μπορεί να εμφανίσει μια γραμμή. Για τον λόγο αυτό, επιχειρείται μια πιο διεξοδική ανάλυση των αποτελεσμάτων της πειραματικής διαδικασίας σε αυτή τη γραμμή. Οι θέσεις της γραμμής του Thara όπου εντοπίζονται κρίσιμα σημεία ομαδοποιούνται, έτσι ώστε να δημιουργηθούν γραμμικά τμήματα κοινών γεωμετρικών χαρακτηριστικών. Όπως φαίνεται στο Σχήμα 3.13 η γραμμή διαιρείται σε: (i) τμήματα διαδοχικών διακυμάνσεων (a, b, c), (ii) τμήματα απότομης αλλαγής διεύθυνσης (μεταβολές κλίσης πλησίον των 90 μοιρών) (d, e, f), (iii) τμήματα που σχηματίζεται ένα τρίγωνο (g, i) και (iv) τα δύο άκρα (j, k). Στο Σχήμα 3.13 τα κρίσιμα σημεία απεικονίζονται με μαύρες κουκίδες.



**Σχήμα 3.13:** Η θεωρητική γραμμή του Thara, τα κρίσιμα σημεία και οι τέσσερις τύποι τμημάτων γραμμής.

Συγκρίνοντας τη διάρκεια προσηλώσεων που αντιστοιχούν στις θέσεις των κρίσιμων σημείων, παρατηρείται ότι τα υποκείμενα εστιάζουν περισσότερο στα τμήματα διαδοχικών διακυμάνσεων της γραμμής (a, b και c στο Σχήμα 3.13). Πιο συγκεκριμένα, το τμήμα «b», το οποίο χαρακτηρίζεται από πυκνές, αιχμηρές και μεσαίου μήκους διακυμάνσεις, εξετάζεται για τον αξιοσημείωτο χρόνο των 676msec ανά υποκείμενο. Αξίζει να σημειωθεί ότι τα κρίσιμα σημεία που εντοπίζονται στο τμήμα «b» έχουν τις μέγιστες τιμές του δείκτη αναλογίας μήκους ALR. Στο τμήμα «a», το οποίο χαρακτηρίζεται από μεγάλου μήκους και ομαλές διακυμάνσεις, επίσης παρατηρείται μεγάλη χρονική διάρκεια προσηλώσεων, της τάξης των 754msec ανά υποκείμενο. Ωστόσο, όπως φαίνεται στο Σχήμα 3.13, το μήκος του τμήματος «b» είναι πολύ μικρότερο από εκείνο του τμήματος «a». Ως εκ τούτου, η πυκνότητα προσηλώσεων στο τμήμα «b» είναι προφανώς υψηλότερη, γεγονός που απεικονίζεται καλά και στον χάρτη θερμότητας που έχει δημιουργηθεί για τη γραμμή του Thara (Σχήμα 3.11). Τέλος, τα υποκείμενα δαπανούν λιγότερο σε σχέση με τα δύο προηγούμενα, αλλά επίσης σημαντικό χρόνο (214msec ανά υποκείμενο) για την παρατήρηση των διακυμάνσεων της γραμμής που εμφανίζονται στο τμήμα «c», οι οποίες είναι μεν πυκνές, αλλά μικρές σε μέγεθος.

Σχετικά με τα τμήματα της γραμμής που χαρακτηρίζονται από απότομη αλλαγή διεύθυνσης (e, d, και f στο Σχήμα 3.13), από την ανάλυση των αποτελεσμάτων προκύπτει ότι η διάρκεια των προσηλώσεων αυξάνει (65msec, 101msec και 131msec ανά υποκείμενο, αντίστοιχα) όταν η απόσταση μεταξύ των σημείων αλλαγής κλίσης μειώνεται. Το γεγονός αυτό υποδηλώνει ότι τα υποκείμενα τείνουν να εστιάζουν περισσότερο σε περιοχές όπου η αλλαγή διεύθυνσης λαμβάνει χώρα σε μικρότερη περιοχή.

Όσον αφορά το τμήμα της γραμμής «i», το οποίο ομοιάζει με ένα ισόπλευρο τρίγωνο, οι προσηλώσεις συγκεντρώνονται γύρω από τις τρεις κορυφές του. Πιο συγκεκριμένα, τα υποκείμενα αφιερώνουν περισσότερο χρόνο παρατηρώντας την κορυφή του τριγώνου (109msec ανά υποκείμενο) και λιγότερο τις δύο απολήξεις του (περίπου 40msec για κάθε μία). Αντίθετα, εξετάζοντας τις διάρκειες των προσηλώσεων στο τμήμα «g» της γραμμής (το οποίο ομοιάζει με ένα ισοσκελές τρίγωνο), προκύπτει ότι αυτές είναι μεγαλύτερες στις δύο κορυφές βάσης (περίπου 75msec ανά υποκείμενο για κάθε μία) και μικρότερες στην κορυφή (20msec ανά υποκείμενο). Παράλληλα, παρατηρείται ότι οι προσηλώσεις εντοπίζονται όχι μόνο στις τρεις κορυφές του τριγώνου, αλλά και στις ενδιάμεσες περιοχές. Αυτό μπορεί να εξηγηθεί από το γεγονός ότι η απόσταση μεταξύ των δύο κορυφών βάσης είναι μικρή, με αποτέλεσμα τα υποκείμενα πιθανά να μην είναι σε θέση να τις διακρίνουν από την κορυφή.

Τέλος, η αρχή και το πέρας της θεωρητικής γραμμής του Thara (j και k στο Σχήμα 3.13) ανιχνεύονται ικανοποιητικά από τα υποκείμενα. Μάλιστα, από την εξέταση των γραμμών σάρωσης προκύπτει πως από τις θέσεις αυτές τα υποκείμενα αρχίζουν την εξέταση της γραμμής. Πιο συγκεκριμένα, η απόληξη «j» αποτελεί το σημείο εκκίνησης σάρωσης όταν η γραμμή παρουσιάζεται στην αρχική της μορφή, ενώ η απόληξη «k» για όλες τις υπόλοιπες στραμμένες εκδοχές της. Κάτι τέτοιο παρατηρείται και για τις τρεις γραμμές από την εμπειρική μελέτη της Marino. Το γεγονός καταδεικνύει ότι τα υποκείμενα τείνουν να ξεκινούν τη διαδικασία σάρωσης από την πάνω αριστερή πλευρά της εικόνας. Αξίζει να σημειωθεί ότι στην περίπτωση της ακτογραμμής της νήσου Περιστερά, η οποία αποτελεί μια κλειστή γραμμή, η διαδικασία σάρωσης ξεκινά συνήθως από το κέντρο της εικόνας, και πραγματοποιείται εξίσου αριστερόστροφα και δεξιόστροφα.

Τα αποτελέσματα της εμπειρικής έρευνας έρχονται κατ' αρχήν σε συμφωνία με τη θεωρία του Attneave, ότι δηλαδή η μεγαλύτερη ποσότητα πληροφορίας συγκεντρώνεται σε θέσεις μεγάλης αλλαγής κλίσης μιας γραμμής. Όσον αφορά τα κρίσιμα σημεία που έχουν εντοπισθεί από την εφαρμογή του δείκτη αναλογίας μήκους ALR, όπως φαίνεται στους Πίνακες 3.4 και 3.5, τα υποκείμενα αφιερώνουν περισσότερο χρόνο για να παρατηρήσουν αυτά, σε σχέση με τις υπόλοιπες θέσεις των γραμμών. Η ταύτιση των προσηλώσεων των υποκειμένων σε θέσεις γύρω από τα κρίσιμα σημεία επιβεβαιώνεται και μέσω των χαρτών θερμότητας (Σχήμα 3.11). Έτσι, μέσω της εμπειρικής έρευνας εκτιμάται πως η λειτουργικότητα της μεθόδου εντοπισμού κρίσιμων σημείων μέσω του δείκτη αναλογίας μήκους ALR είναι αρκετά ικανοποιητική.

### 3.2 Εξομάλυνση €-τμημάτων

#### 3.2.1 Θεωρητική αναφορά

Ο τελεστής εξομάλυνσης εντάσσεται στο μοντέλο γενίκευσης, ώστε να χρησιμοποιηθεί σε τρία διαφορετικά στάδια επεξεργασίας των γραμμικών οντοτήτων: στην απαλοιφή του θορύβου ψηφιοποίησης, στη διαδικασία εντοπισμού καμπών και στην εξομάλυνση μετά τη συνένωση των γενικευμένων €-τμημάτων.

Μία γραμμή που αποδίδεται στην αρχική της μορφή χαρακτηρίζεται από λεπτομέρεια ανάλογη της κλίμακας. Η λεπτομέρεια αυτή, η οποία μπορεί, είτε να αναπαριστά την πραγματικότητα, είτε να είναι προϊόν μη ορθής ψηφιοποίησης, είναι δυνατό να επηρεάσει τις μετέπειτα διαδικασίες ανάλυσης, χαρακτηρισμού και γενίκευσης. Για παράδειγμα, ένας πολύ μικρός κόλπος που αναπαρίσταται σε μια αρχική ακτογραμμή είναι σχεδόν βέβαιο πως δε θα αποδοθεί στην παράγωγη κλίμακα. Υπάρχει περίπτωση, όμως, ο τελεστής γενίκευσης που θα εφαρμοσθεί να εξαρτηθεί από αυτόν, γεγονός που θα μειώσει την ποιότητα του αποτελέσματος. Για την αντιμετώπιση ανάλογων προβλημάτων κρίνεται σκόπιμο, οι γραμμές να υπόκεινται σε μια διαδικασία εξομάλυνσης, πριν από την τελική επεξεργασία τους· σε μία, δηλαδή, μικρο-μετατόπιση των σημείων που τις δομούν, ώστε να αποκτήσουν μια πιο ομαλή μορφή, χωρίς να μεταβληθεί ο χαρακτήρας και η γεωμετρία τους. Πρέπει να τονισθεί, πως το αποτέλεσμα της εξομάλυνσης στις περισσότερες εφαρμογές δεν επαρκεί να εξαντλήσει τις ανάγκες της χαρτογραφικής γενίκευσης. Η χρήση του όμως, ως ενδιάμεσο βήμα για την επιλογή των προς διατήρηση θέσεων μιας γραμμής, αποδεικνύεται ιδιαίτερα χρήσιμη, αφού αυτή γίνεται μέσω ενός αντικειμενικού κριτηρίου και απαλείφονται ορισμένα τα ανθρώπινα σφάλματα της ψηφιοποίησης.

#### 3.2.2 Τελεστής εξομάλυνσης Gauss – Προσαρμογή και ενσωμάτωση στο μοντέλο

Η εξομάλυνση πραγματοποιείται με την εφαρμογή του τελεστή *Gauss*, ο οποίος δίνεται από τις σχέσεις (Lecordix et al. 1997, Fritsch 1997):

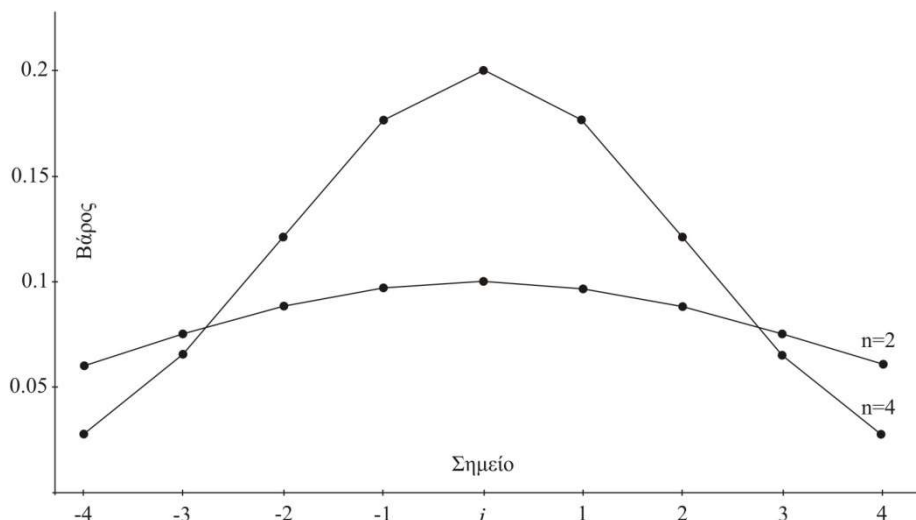
$$X_1(i) = \sum_{k=-4\sigma}^{4\sigma} X(i-k) \cdot g_{\sigma}(k), \quad \text{όπου} \quad g_{\sigma}(k) = \frac{1}{\sigma \cdot \sqrt{2\pi}} e^{-k^2/2\sigma^2}$$

$$Y_1(i) = \sum_{k=-4\sigma}^{4\sigma} Y(i-k) \cdot g_{\sigma}(k)$$

Ο τελεστής *Gauss* είναι ένα φίλτρο εξομάλυνσης κινούμενου μέσου όρου με βάρη (Li 2007). Η γενική αρχή λειτουργίας του έγκειται στη μετατόπιση του εκάστοτε σημείου  $i$  της γραμμής, με βάση τις συντεταγμένες του ίδιου, αλλά και των γειτονικών του. Οι συντεταγμένες της νέας θέσης ( $X_1(i)$ ,  $Y_1(i)$ ) υπολογίζονται από το μέσο όρο των συντεταγμένων του προς εξέταση, των  $k$  πριν και των  $k$  μετά αυτού σημείων. Η μεταβλητή  $k$  ορίζεται από τη σχέση  $k = n \cdot \sigma$ . Κάθε σημείο συμβάλει στον

υπολογισμό του μέσου όρου υπό ένα συντελεστή βάρους  $g_{\sigma}(k)$ , η τιμή του οποίου εξαρτάται από τις τιμές των  $\sigma$  και  $n$ , και από το πλήθος των σημείων που παρεμβάλλονται μεταξύ αυτού και του προς μετατόπιση. Το βάρος των εκατέρωθεν του  $i$  σημείων είναι μικρότερο, όσο μεγαλύτερο είναι το  $n$  (βλ. Σχήμα 3.14). Όπως φαίνεται στις σχέσεις, στο μοντέλο η τιμή του  $n$  επιλέγεται σταθερή και ίση με 4 ( $k = 4 \cdot \sigma$ ). Η τιμή αυτή θεωρείται ικανοποιητική ώστε να περιορίζεται επιρροή των εκατέρωθεν σημείων στο αποτέλεσμα, δεδομένου ότι ο τελεστής *Gauss* προορίζεται για την εξομάλυνση γραμμών τυχαίας γεωμετρίας και συγχρόνως, για εφαρμογές στις οποίες συμμετέχει μεγάλος αριθμός σημείων. Όπως φαίνεται στο Σχήμα 3.14, τα βάρη των συμμετεχόντων σημείων ακολουθούν κανονική κατανομή με κεντρικό αυτό του οποίου οι συντεταγμένες υπολογίζονται. Αποτέλεσμα της διαδικασίας είναι μία μετάθεση, το διάνυσμα της οποίας επηρεάζεται όλο και λιγότερο από τις συντεταγμένες των πιο απομακρυσμένων σημείων. Επιπρόσθετα, το άθροισμα των βαρών ισούται με τη μονάδα ( $\sum g_{\sigma}(k) = 1$ ). Στην πράξη όμως, λόγω σφαλμάτων στρογγυλοποίησης, το άθροισμα παρουσιάζει μικρές αποκλίσεις από αυτή. Το γεγονός έχει ως συνέπεια η νέα θέση των σημείων να είναι, έστω και κατ' ελάχιστο, μετατοπισμένη από την ορθή. Με βάση τις προδιαγραφές του μοντέλου, κάτι τέτοιο δημιουργεί πρόβλημα, ιδιαίτερα σε ό,τι αφορά τα άκρα και την κορυφή του  $\epsilon$ -τμήματος, αφού τα μεν πρώτα αποτελούν σημεία σύνδεσης μεταξύ των  $\epsilon$ -τμημάτων, ενώ στην κορυφή στηρίζονται οι μετέπειτα διεργασίες γενίκευσης. Για την επίλυση του προβλήματος, οι προκύπτουσες συντεταγμένες  $X_1(i)$ ,  $Y_1(i)$  διαιρούνται με το άθροισμα των βαρών, ώστε να διορθωθούν από σφάλματα στρογγυλοποίησης. Έτσι, τελικά, οι νέες ορθές συντεταγμένες  $X'_1(i)$ ,  $Y'_1(i)$  κάθε σημείου  $i$  μίας γραμμής υπολογίζονται από τις σχέσεις:

$$X'_1(i) = \frac{\sum_{k=-4\sigma}^{4\sigma} X(i-k) \cdot g_{\sigma}(k)}{\sum_{k=-4\sigma}^{4\sigma} g_{\sigma}(k)} \quad (1\alpha) \quad \text{και} \quad Y'_1(i) = \frac{\sum_{k=-4\sigma}^{4\sigma} Y(i-k) \cdot g_{\sigma}(k)}{\sum_{k=-4\sigma}^{4\sigma} g_{\sigma}(k)} \quad (1\beta)$$



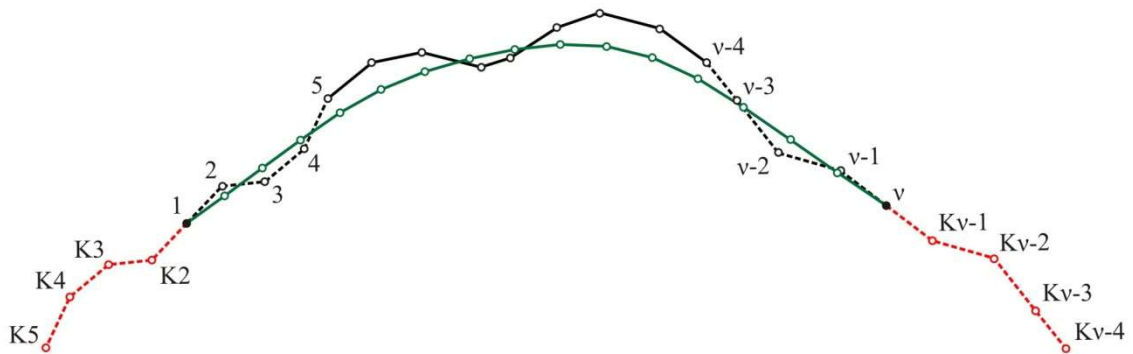
Σχήμα 3.14: Παράδειγμα βαρών σημείων που συμμετέχουν στη μετατόπιση σημείου  $i$ .

Σύμφωνα με την αρχή λειτουργίας του τελεστή *Gauss*, ο βαθμός εξομάλυνσης κάθε γραμμής εξαρτάται από τον αριθμό ( $k$ ) των σημείων που συμμετέχουν στον

υπολογισμό των νέων συντεταγμένων  $X'_1(i)$  και  $Y'_1(i)$ . Συγκεκριμένα, ο βαθμός εξομάλυνσης μεγαλώνει με την αύξηση του  $k$ . Όπως προκύπτει από τις σχέσεις (1α) και (1β), μετά την προσαρμογή του τελεστή στις απαιτήσεις του μοντέλου, η παράμετρος  $k$  καθορίζεται αποκλειστικά από τη μεταβλητή  $\sigma$  ( $k = 4\sigma$ ). Έτσι, κατά την εκτέλεση του τελεστή, ο χρήστης καλείται να θέσει την τιμή της μεταβλητής  $\sigma$ , ανάλογα με το επιθυμητό μέγεθος εξομάλυνσης. Με δεδομένο πως η παράμετρος  $k$  αντιστοιχεί σε φυσικό αριθμό, οι τιμές  $\sigma$  πρέπει να είναι πολλαπλάσια του 0.25. Το εύρος τους κυμαίνεται από την ελάχιστη έως μία μέγιστη ( $\frac{\nu-1}{8}$ ), η οποία εξαρτάται από τον αριθμό  $\nu$  των σημείων που δομούν την εκάστοτε γραμμή ( $0.25 \leq \sigma \leq \frac{\nu-1}{8}$ ).

Για τιμές που δεν πληρούν τις παραπάνω συνθήκες ο τελεστής δεν εκτελείται.

Κατά την ενσωμάτωση του τελεστή εξομάλυνσης *Gauss* στο μοντέλο ανακύπτει ένα πρόβλημα, αναφορικά με την υλοποίησή του στις ακραίες θέσεις των γραμμικών τμημάτων. Στο Σχήμα 3.15, παρουσιάζεται ένα παράδειγμα εξομάλυνσης ενός γραμμικού τμήματος, εκτελώντας τον τελεστή με τιμή του  $\sigma=1$ . Βάσει των κανόνων λειτουργίας του, στην περίπτωση αυτή η επεξεργασία θα λάβει χώρα μόνο μεταξύ των σημείων 5 και  $\nu-4$  της γραμμής. Στα τέσσερα πρώτα σημεία δεν είναι δυνατό να πραγματοποιηθεί, αφού πριν από αυτά δεν υπάρχουν τα σημεία που απαιτούνται (τέσσερα πριν το σημείο 1, τρία πριν το σημείο 2 κοκ) για τον υπολογισμό των νέων συντεταγμένων. Αντίστοιχα, η έλλειψη των απαραίτητων σημείων μετά το σημείο  $\nu-3$ , κάνει μη εφικτή την επεξεργασία των τεσσάρων τελευταίων σημείων του τμήματος. Γίνεται εύκολα αντιληπτό πως το γεγονός αφορά στα  $4\sigma$  πρώτα και στα  $4\sigma$  τελευταία σημεία οποιασδήποτε ανοιχτής γραμμής.



**Σχήμα 3.15:** Παράδειγμα χρήσης ‘εικονικών’ σημείων κατά την υλοποίηση του τελεστή εξομάλυνσης *Gauss* με  $k=4\sigma$  και  $\sigma=1$ .

Δεδομένου πως μια γραμμή πρέπει να εξομαλύνεται ισοβαρώς, καθ’ όλο το μήκος της, ώστε να αποφεύγονται φαινόμενα γραφικής ανομοιογένειας και ασυνεχειών, στη λειτουργία του τελεστή *Gauss* ενσωματώνεται μια τεχνική για την αντιμετώπιση του προβλήματος. Η τεχνική στηρίζεται στη δημιουργία ‘εικονικών’ σημείων εξωτερικά των γραμμικών τμημάτων. Το πλήθος αυτών καθορίζεται από την τιμή της παραμέτρου  $\sigma$  με την οποία εφαρμόζεται κάθε φορά ο τελεστής. Ειδικότερα, δημιουργούνται  $4\sigma$  σημεία πριν την αρχή και  $4\sigma$  μετά το πέρας του τμήματος, ακριβώς, δηλαδή, όσα είναι απαραίτητα για την επίλυση του προβλήματος. Οι

συντεταγμένες των ‘εικονικών’ σημείων υπολογίζονται σύμφωνα με την αρχή της συμμετρίας, βάσει των σχέσεων:

$$XK(i) = 2 \cdot X(1) - X(i)$$

$$YK(i) = 2 \cdot Y(1) - Y(i)$$

, για τα σημεία πριν την αρχή και

$$XK(v-i) = 2 \cdot X(v) - X(v-i)$$

$$YK(v-i) = 2 \cdot Y(v) - Y(v-i)$$

, για τα σημεία μετά το πέρας του γραμμικού τμήματος.

Σύμφωνα με τις παραπάνω σχέσεις, τα ‘εικονικά’ σημεία είναι συμμετρικά των  $4\sigma$  μετά το πρώτο και των  $4\sigma$  πριν το τελευταίο σημείο της γραμμής, τόσο ως προς τον άξονα X, όσο και ως προς τον άξονα Y. Στο παράδειγμα του Σχήματος 3.15, για την εξομάλυνση των ακραίων θέσεων, δημιουργούνται μέσω της τεχνικής τα σημεία K2, K3, K4, K5 και K $v-1$ , K $v-2$ , K $v-3$ , K $v-4$ , τα οποία είναι συμμετρικά των σημείων 2, 3, 4, 5 και  $v-1$ ,  $v-2$ ,  $v-3$  και  $v-4$ , αντίστοιχα. Είναι εμφανές, πως τα ‘εικονικά’ σημεία, με τον τρόπο που ορίζονται, ακολουθούν την κατεύθυνση της υφιστάμενης γραμμής, δημιουργώντας μια ομαλή και φυσική συνέχειά της. Με τη χρήση τους, λοιπόν, εκτός της συμμετοχής στη διαδικασία εξομάλυνσης των  $4\sigma$  ακραίων σημείων, επιτυγχάνεται το αποτέλεσμα αυτής να είναι το δυνατόν πιο ρεαλιστικό. Παράλληλα, όπως παρατηρείται στο Σχήμα 3.15, τα πρόσθετα τμήματα που θεωρητικά διαμορφώνονται (γραμμές κόκκινης απόχρωσης) είναι κατοπτρικά των υφιστάμενων (2...5 και  $v-1$ ... $v-4$ ). Συνέπεια αυτού είναι το πρώτο και το τελευταίο σημείο της γραμμής να παραμένουν αμετάθετα μετά την εξομάλυνση. Το γεγονός είναι απαραίτητο σε επόμενο στάδιο επεξεργασίας, κατά το οποίο διενεργείται η συνένωση των γενικευμένων  $\epsilon$ -τμημάτων (βλ. κεφ. 4.5).

Ένα χαρακτηριστικό του τελεστή εξομάλυνσης *Gauss* είναι η συρρίκνωση που επιφέρει όταν εφαρμόζεται σε κλειστές ή καμπύλες γραμμικές οντότητες. Όπως επισημαίνει σε σχετική έρευνα ο Lowe (1988), σε τέτοιας μορφής γραμμές, τα εκατέρωθεν κάθε σημείου κλίνουν προς το τοπικό κέντρο καμπυλότητας. Κάθε σημείο λοιπόν μετατοπίζεται προς αυτό, δεδομένου πως η νέα θέση του προκύπτει από το μέσο όρο των θέσεων των γειτονικών του. Ο Li (2007) αναφέρει χαρακτηριστικά πως αν το φίλτρο *Gauss* εφαρμοσθεί σε ένα κύκλο, το αποτέλεσμα θα είναι ένας μικρότερος κύκλος. Το μέγεθος της συρρίκνωσης είναι ανάλογο της καμπυλότητας της γραμμής και του βαθμού εξομάλυνσης.

Η παραπάνω ιδιότητα λαμβάνεται υπόψη στο σχεδιασμό του μοντέλου, αφού οι συνθήκες κάτω από τις οποίες εφαρμόζεται ο τελεστής ευνοούν την εμφάνισή της. Η συρρίκνωση παρατηρείται ιδιαίτερα κατά την επεξεργασία για τη δημιουργία καμπών. Και τούτο διότι η διαδικασία συνίσταται στην εξομάλυνση συνήθως σε μεγάλο βαθμό, γραμμών μεγάλης καμπυλότητας, όπως είναι τα  $\epsilon$ -μη-κυρτά τμήματα. Όπως περιγράφεται στα κεφάλαια 4.1.2.1 και 4.2.2.1, το γεγονός πως τα προκύπτοντα τμήματα εμφανίζονται πάντα συρρικνωμένα επιτρέπει τον προκαθορισμό των ‘εργαλείων’ γενίκευσης που έπονται του σταδίου δημιουργίας καμπών, ενώ συγχρόνως βοηθά στον προσδιορισμό των παραμέτρων εφαρμογής αυτών. Η συρρίκνωση παρατηρείται επίσης και κατά τη διεργασία συνένωσης των γενικευμένων  $\epsilon$ -τμημάτων, ειδικά όταν η παράγωγη γραμμή είναι κλειστή (πχ ακτογραμμή ενός νησιού). Στην περίπτωση αυτή, το φαινόμενο είναι σαφώς πιο

ήπιο, αφού τόσο η καμπυλότητα της γραμμής, όσο και ο βαθμός εξομάλυνσης είναι μικρότερα. Δεδομένου, όμως, πως το μοντέλο απευθύνεται στη διαχείριση γραμμών μεγάλης πολυπλοκότητας και τυχαίας γεωμετρίας, μία έστω και ελάχιστη συρρίκνωση είναι πιθανό να δημιουργήσει εκ νέου πρόβλημα ευαναγνωσιμότητας στο τελικό προϊόν. Για το λόγο αυτό, μετά από κάθε εφαρμογή πραγματοποιείται έλεγχος με βάση τους κανόνες της οπτικής αντίληψης που έχουν ορισθεί (βλ. κεφ. 4.6).

Ο τελεστής εξομάλυνσης *Gauss*, όπως προσαρμόσθηκε για την ενσωμάτωσή του στο διαδραστικό σύστημα γενίκευσης, δημιουργήθηκε σε γλώσσα προγραμματισμού *Python*. Ο κώδικας υλοποίησης του παρουσιάζεται στο Παράρτημα Π3.2 (ανοιχτές γραμμές) και Π3.2Κ (κλειστές γραμμές).

### **3.3 Μέθοδος εντοπισμού καμπών κατά μήκος των γραμμών**

Η επιλογή κατάλληλου τελεστή γενίκευσης, καθώς και των παραμέτρων εφαρμογής του προϋποθέτει, σε αρκετές περιπτώσεις, τη γνώση χαρακτηριστικών της δομής των υπό εξέταση  $\epsilon$ -τμημάτων. Μία μέθοδος ανάλυσης της γεωμετρίας ψηφιακών γραμμικών οντοτήτων, ώστε να εξαχθούν πληροφορίες απαραίτητες για την περαιτέρω επεξεργασία, είναι ο εντοπισμός καμπών κατά μήκος τους.

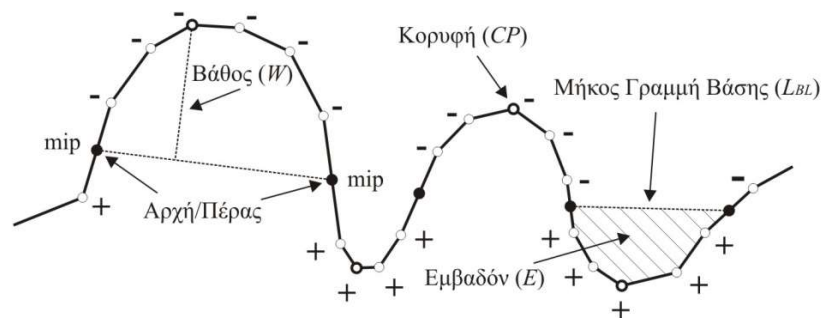
Στο διαδραστικό σύστημα γενίκευσης προσαρμόζεται και υλοποιείται σε κώδικα ο ορισμός που προτείνουν οι Wang και Müller (1998) για τις καμπές. Σύμφωνα με αυτόν, μία καμπή (bend) είναι 'εκείνο το τμήμα της γραμμής που περιέχει έναν αριθμό διαδοχικών κορυφών, με τις γωνίες κάμψης σε όλες τις κορυφές που περιέχονται στην καμπή να είναι είτε θετικές είτε αρνητικές και η κάμψη των ακραίων κορυφών της να είναι με αντίθετα πρόσημα' (Wang and Müller 1998, 5). Πιο απλά, μία καμπή είναι το τμήμα μιας γραμμής, το οποίο αποτελείται από έναν αριθμό σημείων με ομόσημες γωνίες κλίσης. Ως 'κλίση σημείου' ορίζεται η γωνιακή διαφορά της διεύθυνσης των ευθυγράμμων τμημάτων 'σημείο-επόμενο σημείο' και 'σημείο-προηγούμενο σημείο'. Από τα παραπάνω απορρέουν δύο βασικά χαρακτηριστικά των ψηφιακών γραμμών. Το πρώτο είναι πως δύο διαδοχικές καμπές είναι πάντα ετερόσημες και το δεύτερο, ότι μια γραμμή αποτελείται από διαδοχικές καμπές.

Η μεθοδολογία των Wang και Müller αποτελεί χρήσιμο εργαλείο προεπεξεργασίας και ανάλυσης των μονόπλευρων (βλ. κεφ. 4.1.2.1) και των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων κατά τη διαδικασία γενίκευσής τους (βλ. κεφ. 4.2.1), αφού οι έννοιες που εισάγει χρησιμοποιούνται για τον εντοπισμό των επιμέρους καμπών που διαμορφώνουν τις επαλληλίες, αλλά και χαρακτηριστικών θέσεων αυτών.

Η προσαρμογή της μεθόδου στις ανάγκες της έρευνας συνίσταται στη δημιουργία ενός κώδικα, μέσω του οποίου υπολογίζονται οι κλίσεις των σημείων και απομονώνονται οι καμπές, ως το σύνολο των σημείων με ομόσημες κλίσεις. Μία διαφοροποίηση σε σχέση με τους ορισμούς των Wang και Müller έχει να κάνει με την οριοθέτηση των καμπών. Σύμφωνα με αυτούς, η αρχή και το πέρας κάθε καμπής θεωρούνται τα σημεία αλλαγής του προσήμου κλίσης (σημεία



κάμψης/inflection points). Κάτι τέτοιο, όμως, έχει ως αποτέλεσμα οι καμπές που απομονώνονται να αλληλεπικαλύπτονται (βλ. Wang and Müller 1998, 6, σχήματα 3 και 4), γεγονός που δε συνάδει με τη φιλοσοφία και τις απαιτήσεις της παρούσας έρευνας. Για το λόγο αυτό, ως αρχή ή πέρασ καμπής ορίζονται τα μέσα των διανυσμάτων, των οποίων τα άκρα έχουν διαφορετικού πρόσημου κλίσεις (Σχήμα 3.16). Έτσι, εξασφαλίζεται πως το πέρασ μιας καμπής συμπίπτει με την αρχή της επόμενης, με τις δύο αυτές καμπές να μην επικαλύπτονται. Το πρώτο σημείο του  $\in$ -τμήματος ορίζεται ως η αρχή της πρώτης καμπής και το τελευταίο ως το πέρασ της τελευταίας. Στο παράδειγμα του Σχήματος 3.16 παρουσιάζεται μία αλληλουχία καμπών, καθώς και τα χαρακτηριστικά τους.

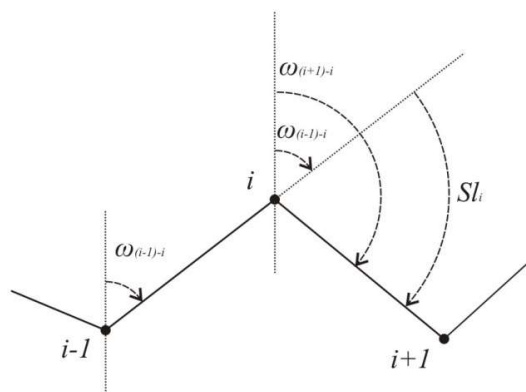


Σχήμα 3.16: Καμπές κατά μήκος γραμμών και χαρακτηριστικά αυτών.

Πιο αναλυτικά, για κάθε σημείο  $i$  ( $=2$  έως  $n-1$ ) του  $\in$ -τμήματος υπολογίζεται η γωνία κλίσης  $Sl$  βάσει της σχέσης:  $Sl = \omega_{(i+1)-i} - \omega_{(i-1)-i}$ , όπου  $\omega$  η γωνία διεύθυνσης των ευθυγράμμων τμημάτων που δημιουργούνται μεταξύ του σημείου  $i$  με το επόμενο και το προηγούμενο σημείο και υπολογίζονται από τις σχέσεις:

$$\omega_{(i+1)-i} = \arctan \left| \frac{dX_{(i+1)-i}}{dY_{(i+1)-i}} \right| \cdot \left( \frac{200}{\pi} \right) \quad \text{και} \quad \omega_{(i-1)-i} = \arctan \left| \frac{dX_{(i-1)-i}}{dY_{(i-1)-i}} \right| \cdot \left( \frac{200}{\pi} \right), \quad \text{αντίστοιχα}$$

(Σχήμα 3.17). Να σημειωθεί πως σε περιπτώσεις που η γωνία κλίσης είναι μεγαλύτερη του  $\pi$  ( $Sl > \pi$ ), διορθώνεται κατά  $2\pi - |Sl|$ . Διασφαλίζεται έτσι, πως η διαδικασία είναι ανεξάρτητη της φοράς σάρωσης της γραμμής. Στη συνέχεια, ανευρίσκονται τα γειτονικά σημεία με αντίθετη γωνία κλίσης και βάσει αυτών εντοπίζεται το μέσο τους, το οποίο αποτελεί το άκρο της καμπής (mip). Κάθε καμπή απαρτίζεται από τουλάχιστον τρία σημεία.



Σχήμα 3.17: Γωνίες διεύθυνσης ευθυγράμμων τμημάτων  $(i-1)-i$  και  $(i+1)-i$

Για κάθε καμπή καταγράφεται το πρόσημο της γωνίας κλίσης και ο αριθμός των σημείων που την απαρτίζουν. Παράλληλα, υπολογίζονται τα γεωμετρικά χαρακτηριστικά τους (κορυφή, μήκος γραμμής βάσης, εμβαδό και βάθος) τα οποία συνεκτιμώνται για την επιλογή των παραμέτρων εφαρμογής των τελεστών γενίκευσης. Πιο συγκεκριμένα:

- Για την εύρεση της κορυφής (CP) της καμπής χρησιμοποιείται ο δείκτης αναλογίας μήκους ALR (κεφ. 3.1.5), ο οποίος έχει ενσωματωθεί στη διαδικασία. Ως κορυφή ορίζεται το σημείο στο οποίο υπολογίζεται η μέγιστη τιμή του δείκτη ALR, αντιστοιχεί δηλαδή στο σημείο που εμφανίζεται η μέγιστη αλλαγή κλίσης της καμπής.

- Ως Μήκος της Γραμμής Βάσης ( $L_{BL}$ ) ορίζεται το μήκος της ευθείας μεταξύ της αρχής (A) και του πέρατος (Π) της γραμμής. Υπολογίζεται από τη σχέση:

$$L_{BL} = \sqrt{(X_{\Pi} - X_A)^2 + (Y_{\Pi} - Y_A)^2}$$

- Ως Βάθος (W) ορίζεται η απόσταση μεταξύ της κορυφής (CP) και της γραμμής βάσης (BL) της καμπής. Υπολογίζεται από τη σχέση:

$$W = \frac{\left| \left( -\frac{Y_{\Pi} - Y_A}{X_{\Pi} - X_A} \right) \cdot X_{CP} + Y_{CP} + \frac{Y_{\Pi} - Y_A}{X_{\Pi} - X_A} \cdot X_A - Y_A \right|}{\sqrt{\left( -\frac{Y_{\Pi} - Y_A}{X_{\Pi} - X_A} \right)^2 + 1}}$$

- Ως Εμβαδόν (E) ορίζεται το εμβαδόν του χωρίου που δημιουργείται μεταξύ καμπής και γραμμής βάσης (BL) της καμπής. Υπολογίζεται από τη σχέση:

$$E = \left| \frac{1}{2} \cdot \sum_1^k X_i \cdot (Y_{i-1} - Y_{i+1}) \right|$$

Η μέθοδος εντοπισμού καμπών δημιουργήθηκε σε γλώσσα προγραμματισμού *Python* και παρουσιάζεται στο Παράρτημα Π3.3.

### 3.4 Ομοπαράλληλος μετασχηματισμός

Ο ομοπαράλληλος μετασχηματισμός είναι ένας από τους διαδομένους μετασχηματισμούς στην αναλυτική χαρτογραφική διαδικασία, αφού επιφέρει ομογενείς παραμορφώσεις στα γραφικά αντικείμενα που εφαρμόζεται. Η μαθηματική σχέση μέσω της οποίας ορίζεται είναι (Νάκος 2006):

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} T_x \\ T_y \end{pmatrix} + \begin{pmatrix} \cos \theta_x & \sin \theta_y \\ -\sin \theta_x & \cos \theta_y \end{pmatrix} \begin{pmatrix} S_x & 0 \\ 0 & S_y \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \text{ όπου:}$$

$T_x$  και  $T_y$  η μετάθεση,

$\theta_x$  και  $\theta_y$  στροφές των αξόνων:  $X$  και  $Y$ , αντίστοιχα και

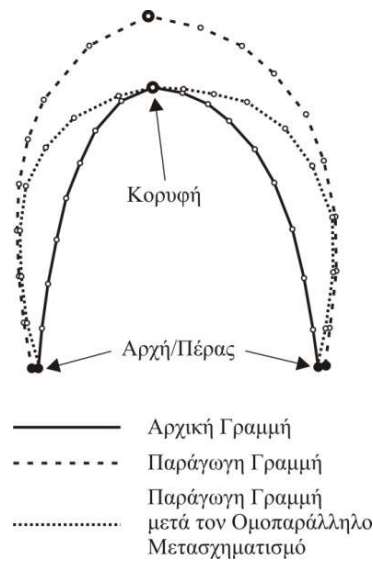
$S_x$  και  $S_y$  κλίμακες κατά τους άξονες:  $X$  και  $Y$ , αντίστοιχα.

Οι γεωμετρικές παράμετροι του ομοπαράλληλου μετασχηματισμού, είναι έξι: δύο μεταθέσεις, δύο στροφές και δύο κλίμακες, κατά τους άξονες  $X$  και  $Y$ , αντίστοιχα. Συχνά χρησιμοποιείται και με τις ακόλουθες γραμμικές σχέσεις:

$$x' = ax + by + c,$$

$$y' = dx + ey + f.$$

Οι συντελεστές των γραμμικών σχέσεων ( $a, b, c, d, e$  και  $f$ ) συσχετίζονται με τις έξι γεωμετρικές παραμέτρους από τις ακόλουθες σχέσεις (Νάκος, 2006):



**Σχήμα 3.18:** Παράδειγμα ομοπαράλληλου μετασχηματισμού μονοπλευρων  $\in$ -μη-κυρτών τμημάτων.

Μετάθεση:  $T_x = c$  και  $T_y = f$

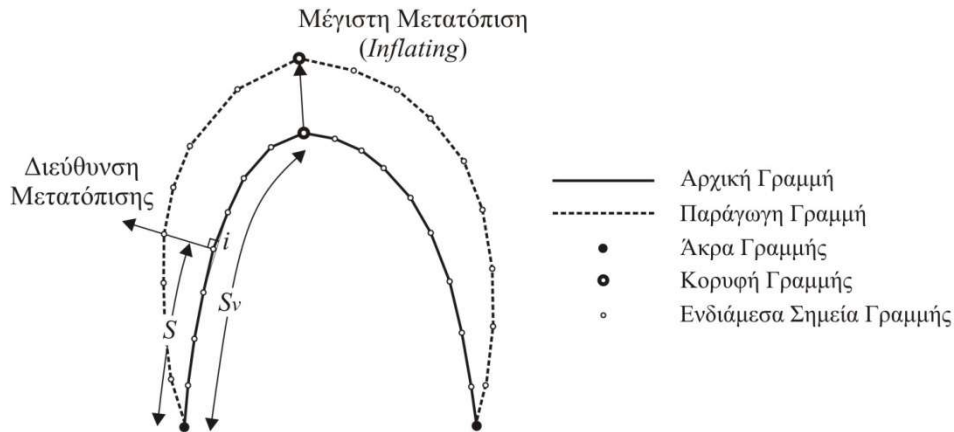
Στροφές:  $\theta_x = \arctan\left(-\frac{d}{a}\right)$  και  $\theta_y = \arctan\left(\frac{b}{e}\right)$

Κλίμακες:  $S_x = \sqrt{a^2 + d^2}$  και  $S_y = \sqrt{b^2 + e^2}$

Ο ομοπαράλληλος μετασχηματισμός μπορεί να δώσει σημαντικές λύσεις στο θέμα της ορθής απόδοσης των γενικευμένων γραμμών, αφού επιδρά ομογενώς στη γεωμετρία και τη θέση ενός αντικειμένου, καθώς επιφέρει στρεβλή μετάθεση, στροφή και αλλαγή κλίμακας. Για την ένταξή του στο μοντέλο γενίκευσης, ο ομοπαράλληλος μετασχηματισμός προσαρμόσθηκε, ούτως ώστε το πρώτο και το τελευταίο σημείο των γραμμών να παραμένουν αμετάθετα, αφού αυτά αποτελούν τα σημεία σύνδεσης των  $\in$ -τμημάτων. Ο κώδικας υλοποίησής του δημιουργήθηκε σε γλώσσα προγραμματισμού *Python* και παρουσιάζεται στο Παράρτημα Π3.4.

### 3.5 Αλγόριθμος Balloon

Ο αλγόριθμος 'Balloon' εισάγεται το 1997 (Lecordix et al. 1997) και αποτελεί έναν από τους αλγόριθμους που εμπεριέχονται στο διαδραστικό σύστημα γενίκευσης 'PlaGe' (βλ. κεφ. 1.1). Εφαρμόζεται σε περιπτώσεις που απαιτείται η διόγκωση γραμμικών τμημάτων που έχουν τη μορφή μιας μεμονωμένης καμψής (κύρτωσης ή κοίλωσης). Ο σχεδιασμός του είναι προσανατολισμένος στη γενίκευση γραμμών που απεικονίζουν τεχνητά φαινόμενα, όπως για παράδειγμα, οδικό δίκτυο.



Σχήμα 3.19: Ο αλγόριθμος *Balloon*.

Σύμφωνα με την αρχή λειτουργίας του αλγορίθμου '*Balloon*', η μετατόπιση του εκάστοτε σημείου  $i$  της γραμμής δίνεται μέσω των μαθηματικών σχέσεων:

$$X_c = X_i + Inflating \cdot Sense \cdot u \cdot S/S_v$$

$$Y_c = Y_i + Inflating \cdot Sense \cdot v \cdot S/S_v, \text{ όπου:}$$

$X_c, Y_c$ : οι συντεταγμένες του σημείου  $i$  στην προκύπτουσα γραμμή,

$X_i, Y_i$ : οι συντεταγμένες του σημείου  $i$  στην αρχική γραμμή,

*Inflating*: ο συντελεστής διόγκωσης της γραμμής,

*Sense*: +1 ή -1, ανάλογα με το πρόσημο της καμπυλότητας,

$u, v$ : ο συντελεστής διεύθυνσης της κάθετου στη γραμμή στο σημείο  $i$ ,

$S$ : το επικαμπύλιο μήκος ανάμεσα στο σημείο  $i$  και το εγγύτερο σε αυτό άκρο της γραμμής,

$S_v$ : το επικαμπύλιο μήκος μεταξύ του εγγύτερου στο σημείο  $i$  άκρο και την κορυφή της γραμμής.

Πιο αναλυτικά, κάθε σημείο της γραμμής μετατοπίζεται κάθετα σε αυτή, με κατεύθυνση εξωτερικά της (ορίζεται μέσω των συντελεστών  $u, v$  και *Sense*, αντίστοιχα). Το μέγεθος της μετατόπισης εξαρτάται από τρεις παράγοντες: το συντελεστή διόγκωσης '*Inflating*', ο οποίος τίθεται από τον χρήστη, την επικαμπύλια απόσταση του σημείου από το εγγύτερο σε αυτό άκρο ( $S$ ) και την επικαμπύλια απόσταση του συγκεκριμένου άκρου από την κορυφή ( $S_v$ ). Αποτέλεσμα, όπως φαίνεται στο Σχήμα 3.19, είναι η γραμμή να διογκώνεται σταδιακά, από μία μηδενική τιμή στα δύο άκρα έως μία μέγιστη (ίση με το συντελεστή διόγκωσης '*Inflating*') στην κορυφή της.

### **3.6 Αλγόριθμος Bendsimplify**

Ο αλγόριθμος 'bendsimplify' αποτελεί ένα τελεστή απλοποίησης γραμμικών οντοτήτων, ο οποίος υποστηρίζεται από το λογισμικό πακέτο *ArcGIS*. Πρόκειται για μια περαιτέρω ανάπτυξη της ιδέας που παρουσιάστηκε από την *Visvalingam* και τον *Whyatt* (1993) και είναι βασισμένος στην έρευνα που επινοήθηκε από τους *Wang* και *Müller* (1998). Η εφαρμογή του στηρίζεται στον εντοπισμό καμπών (*bends*) κατά μήκος των γραμμών και την ανάλυση ιδιοτήτων τους που έχουν ορισθεί (μέγεθος, σχήμα, ομοιότητα με γειτονικές). Βάσει αυτών, εκτελούνται, όπου απαιτείται, διαδοχικά διεργασίες αφαίρεσης, σύνθεσης και μεγιστοποίησης, εκ των οποίων η πρώτη ελαττώνει τις κορυφές, ενώ οι άλλες δύο βελτιώνουν αισθητικά τη γραμμή. Η εξέταση και επεξεργασία συνόλων σημείων (καμπές) και όχι μεμονωμένων, όπως συμβαίνει με πολλούς αλγορίθμους απλοποίησης, έχει ως αποτέλεσμα η τελική εικόνα της γραμμής να παρουσιάζεται αρκετά ομαλή και να αποφεύγεται η εμφάνιση αιχμών ή σφηνών. Το γεγονός αποτελεί πλεονέκτημα, ιδιαίτερα σε περιπτώσεις μεγάλης αλλαγής κλίμακας. Ο αλγόριθμος 'bendsimplify' έχει χρησιμοποιηθεί στο παρελθόν (*Nakos and Mitropoulos 2005*) για τη γενίκευση χαρτογραφικών γραμμών και τα αποτελέσματα της εφαρμογής του κρίνονται ως αρκετά ικανοποιητικά.

## ΚΕΦΑΛΑΙΟ 4: ΓΕΝΙΚΕΥΣΗ $\epsilon$ -ΤΜΗΜΑΤΩΝ ΧΑΡΤΟΓΡΑΦΙΚΩΝ ΓΡΑΜΜΩΝ

### 4.1 Γενίκευση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων (Κατηγορία Α)

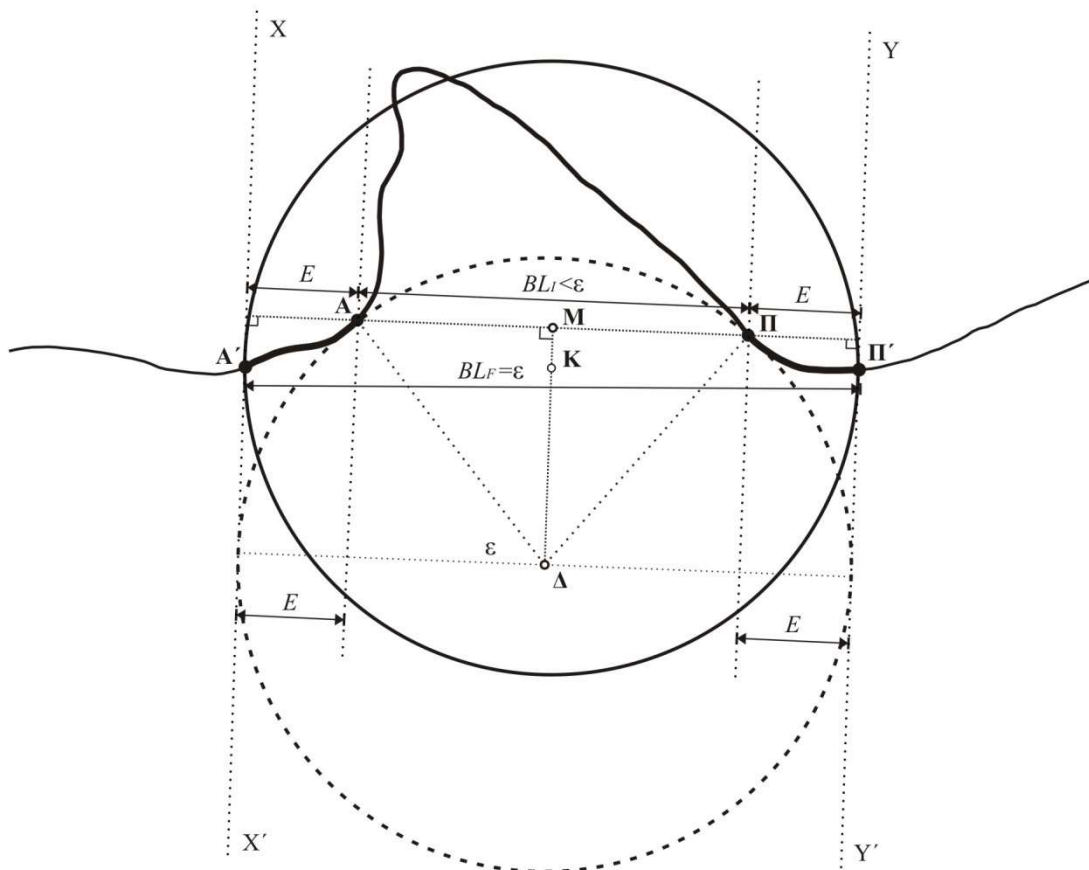
Τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα, συνήθως, έχουν τη μορφή μιας απομονωμένης κύρτωσης ή κοίλωσης. Κατά τις διεργασίες διαχωρισμού των  $\epsilon$ -τμημάτων σε κατηγορίες (βλ. κεφ. 2.3.3 και 2.3.4), σε αυτά εντάσσονται τα απομονωμένα  $\epsilon$ -μη-κυρτά τμήματα, τα οποία δεν έχουν συγχωνευθεί με γειτονικά τους για το σχηματισμό αμφίπλευρων και δεν αποτελούν μέρος τμημάτων σύγκλισης. Θεωρητικά, ένα μονόπλευρο  $\epsilon$ -μη-κυρτό τμήμα αποκόπτεται μεταξύ δύο θέσεων μιας γραμμής, όταν η κύλιση δίσκου του Perkal διαμέτρου  $\epsilon$  που εφάπτεται εσωτερικά ή εξωτερικά αυτής, διακόπτεται μεταξύ των δύο αυτών θέσεων, χωρίς να συμβαίνει το ίδιο στην άλλη πλευρά της γραμμής. Βάσει των κανόνων οπτικής αντίληψης που έχουν ορισθεί στο κεφάλαιο 2, το γεγονός της μη κύλισης του δίσκου του Perkal κατά μήκος των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων αναδεικνύει το πρόβλημα ευαναγνωσιμότητάς τους. Για την ευκρινή απόδοση τους σε χάρτες μικρότερης κλίμακας απαιτείται η μεγέθυνση τους, ώστε να εξασφαλίζεται το όριο οπτικής αντίληψης, θεωρητικά δηλαδή, ο δίσκος του Perkal διαμέτρου  $\epsilon$  να εισέρχεται σε αυτά και να κυλά κατά μήκος τους. Η επεξεργασία στην οποία υπόκεινται για την επίτευξη αυτού, συνίσταται κατά σειρά στη διεύρυνση, την εξομάλυνση, τη διόγκωση και την εφαρμογή του ομοπαράλληλου μετασχηματισμού.

#### 4.1.1 Διεύρυνση μονόπλευρων $\epsilon$ -μη κυρτών τμημάτων

Όπως αναφέρεται στην προηγούμενη ενότητα, η ευαναγνωσιμότητα των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων στην παράγωγη κλίμακα διασφαλίζεται αν ο θεωρητικός δίσκος του Perkal εισέρχεται εντός αυτών και κυλά κατά μήκος τους. Προϋπόθεση για την επίτευξη του πρώτου είναι η διεύρυνση του  $\epsilon$ -τμήματος. Η διαδικασία, λόγω της σημαντικότητάς της, προηγείται των υπολοίπων διεργασιών γενίκευσης (εξομάλυνση, διόγκωση και ομοπαράλληλο μετασχηματισμό).

Στο Σχήμα 4.1 παρουσιάζεται ένα χαρακτηριστικό παράδειγμα μονόπλευρου  $\epsilon$ -μη-κυρτού τμήματος. Το  $\epsilon$ -τμήμα (απεικονίζεται με γραμμή μέσου πάχους) έχει αποκοπεί μεταξύ των σημείων Α και Β, όπου ο δίσκος του Perkal διαμέτρου  $\epsilon$  (διακεκομμένος κύκλος) σταματά να κυλά κατά μήκος της γραμμής, στη μία πλευρά αυτής. Τα σημεία Α και Β αποτελούν την αρχή και το πέρας του, ενώ το σημείο Δ είναι το θεωρητικό κέντρο του δίσκου, στη θέση που 'δημιουργεί' το  $\epsilon$ -τμήμα. Είναι εμφανές πως η γραμμή βάσης ΑΒ του  $\epsilon$ -τμήματος, δηλαδή η γραμμή που ενώνει τα άκρα του, έχει μήκος ( $BL_1$ ) μικρότερο από τη διάμετρο  $\epsilon$  του δίσκου του Perkal. Για να επιτευχθεί η 'εισχώρηση' του δίσκου εντός του  $\epsilon$ -τμήματος απαιτείται η διεύρυνση του τελευταίου, σε βαθμό που το μήκος της γραμμής βάσης του προκύπτοντος  $\epsilon$ -τμήματος να είναι τουλάχιστον ίσο με τη διάμετρο  $\epsilon$  ( $BL_F = \epsilon$ ). Το μήκος που υπολείπεται για την πραγμάτωση αυτού είναι  $\epsilon - BL_1$ . Η διεύρυνση πραγματοποιείται με την προσάρτηση μέρους από τα γειτονικά  $\epsilon$ -τμήματα (γραμμές μεγάλου πάχους στο Σχήμα 4.1). Για λόγους διατήρησης της συμμετρίας

και της γεωμετρίας του  $\epsilon$ -τμήματος, η διεύρυνση ορίζεται να λαμβάνει χώρα ισοβαρώς εκατέρωθεν των άκρων του, όπως παρουσιάζεται στο Σχήμα 4.1. Πιο συγκεκριμένα, το  $\epsilon$ -τμήμα διευρύνεται με τρόπο, ώστε η γραμμή βάσης του να επεκτείνεται κατά μήκος  $E = \frac{\epsilon - BL_I}{2}$ , εξωτερικά του κάθε άκρου του. Το μήκος αυτό τίθεται ως η τιμή εφαρμογής, αφού είναι το ελάχιστο και αναγκαίο για την επίλυση του προβλήματος, ενώ παράλληλα, εξασφαλίζει την αποκοπή των ελάχιστων τμημάτων από τα γειτονικά, ώστε να μην επηρεάζονται οι μετέπειτα διεργασίες γενίκευσης.



Σχήμα 4.1: Διεύρυνση μονόπλευρου  $\epsilon$ -μη-κυρτού τμήματος.

Βάσει των παραπάνω, η διεύρυνση του μονόπλευρου  $\epsilon$ -μη-κυρτού τμήματος ανάγεται στον εντοπισμό δύο σημείων  $A'$  και  $\Pi'$  (Σχήμα 4.1) κατά μήκος της γραμμής, τα οποία απέχουν απόσταση  $E$  κατά τη διεύθυνση της ευθείας μεταξύ των  $A$  και  $\Pi$  από τα σημεία  $A$  και  $\Pi$  αντίστοιχα και συγχρόνως, απόσταση  $\epsilon$  μεταξύ τους. Θεωρητικά, το πρόβλημα επιλύεται αν ο δίσκος του Perkal κυλήσει με αφετηρία τη θέση όπου 'δημιουργεί' το  $\epsilon$ -τμήμα (το κέντρο του στο σημείο  $\Delta$ ) και κατεύθυνση προς τη γραμμή, ώστε να εισχωρήσει στο  $\epsilon$ -τμήμα. Κατά την κύλισή του, ο δίσκος τέμνει τη γραμμή σε μία σειρά από ζεύγη σημείων. Τα ζητούμενα σημεία  $A'$  και  $\Pi'$  εντοπίζονται ως το μοναδικό ζεύγος εξ αυτών, το οποίο ικανοποιεί το κριτήριο ορισμού τους.

Η διεύθυνση κύλισης του δίσκου του Perkal (η οποία καθορίζεται από τη διεύθυνση μετατόπισης του κέντρου του) αποτελεί σημαντική παράμετρο της διαδικασίας, αφού από αυτήν εξαρτάται η ισοβαρής διεύρυνση του  $\epsilon$ -τμήματος. Ως διεύθυνση κύλισης ορίζεται η ευθεία που ενώνει το κέντρο του δίσκου στην αρχική του θέση ( $\Delta$ ) με το μέσο ( $M$ ) της γραμμής βάσης ΑΠ του  $\epsilon$ -τμήματος. Η ευθεία  $\Delta M$  είναι κάθετη στη γραμμή βάσης, αφού το νοητό τρίγωνο  $\Delta\Gamma M$  είναι ισοσκελές ( $\Delta A = \Delta \Gamma = \frac{\epsilon}{2}$ ) (Σχήμα 4.1). Έτσι, ο δίσκος κινείται κάθετα της αρχικής πορείας του και

κατ' επέκταση κάθετα στο  $\epsilon$ -τμήμα. Με τον τρόπο αυτό εξασφαλίζεται πως καθώς το  $\epsilon$ -τμήμα διευρύνεται, η γραμμή βάσης του επεκτείνεται ισόποσα εξωτερικά των δύο άκρων του ( $E_A = E_{\Pi}$ ). Η κύλιση του δίσκου κατά οποιαδήποτε άλλη διεύθυνση θα είχε ως αποτέλεσμα την επέκταση της γραμμής βάσης κατά διαφορετική απόσταση από το κάθε άκρο ( $E_A \neq E_{\Pi}$ ) και ως εκ τούτου, την ανισοβαρή διεύρυνση του  $\epsilon$ -τμήματος, γεγονός που θα παραμόρφωνε τη γεωμετρία και το σχήμα του.

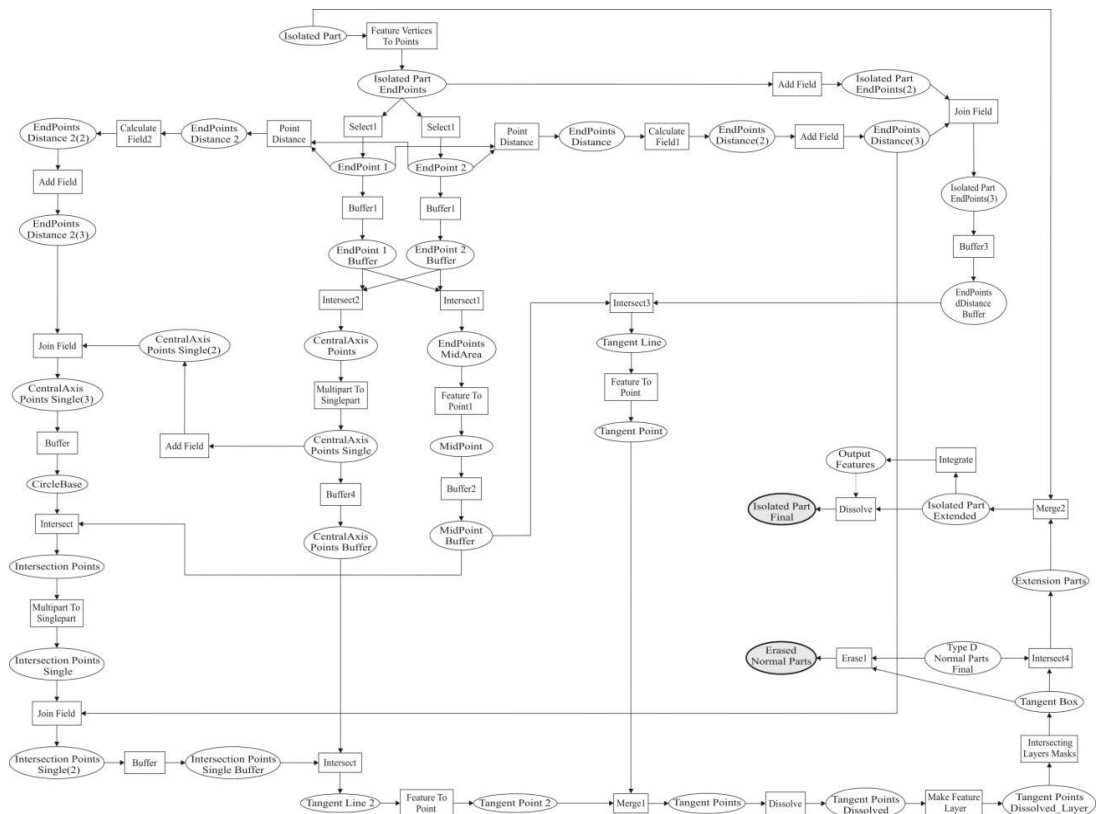
Κατά την κύλιση του μεταξύ των θέσεων  $\Delta$  και  $M$ , ο δίσκος του Perkal σαρώνει μία επιφάνεια, της οποίας τα όρια διαμορφώνουν δύο ευθείες  $XX'$  και  $YY'$  (Σχήμα 4.1). Οι ευθείες αυτές αποτελούν το γεωμετρικό τόπο των σημείων που απέχουν απόσταση  $\frac{\epsilon}{2}$  από τον άξονα κύλισης  $\Delta M$  και είναι παράλληλες με αυτόν. Η μεταξύ

τους απόσταση είναι ίση με τη διάμετρο  $\epsilon$  του δίσκου. Το κάθε άκρο ( $A$  και  $\Pi$ ) του  $\epsilon$ -τμήματος απέχει απόσταση  $E = \frac{\epsilon - BL_I}{2}$  από την εγγύτερη αυτού ευθεία. Ως εκ

τούτου, τα σημεία  $A'$  και  $\Pi'$  βρίσκονται στις ευθείες  $XX'$  και  $YY'$  αντίστοιχα, και συγκεκριμένα, εντοπίζονται ως τα σημεία τομής των δύο ευθειών με τη χαρτογραφική γραμμή. Θεωρητικά, κατά την κύλισή του, ο δίσκος τέμνει τη γραμμή σε μία σειρά θέσεων, εξωτερικά των άκρων  $A$  και  $\Pi$ . Οι αποστάσεις μεταξύ των εκάστοτε σημείων τομής ξεκινούν από τιμή ίση με το μήκος  $BL_I$  της γραμμής βάσης (αρχική θέση), σταδιακά αυξάνουν, παίρνουν μία μέγιστη τιμή  $\epsilon$  και στη συνέχεια φθίνουν. Τα ζητούμενα σημεία  $A'$  και  $\Pi'$  είναι εκείνα που απέχουν τη μέγιστη απόσταση  $\epsilon$  και ανευρίσκονται όταν ο δίσκος τέμνει τη γραμμή στη διάμετρό του. Το κέντρο  $K$  του δίσκου (συνεχής κύκλος στο Σχήμα 4.1) στη θέση αυτή εντοπίζεται μεταξύ των σημείων  $\Delta$  και  $M$ .

Βάσει της λογικής που περιγράφεται παραπάνω, διασφαλίζεται πως η διεύρυνση του μονόπλευρου  $\epsilon$ -μη-κυρτού τμήματος ολοκληρώνεται ισοβαρώς, με την επέκταση της γραμμής βάσης κατά απόσταση  $E$  εξωτερικά των δύο άκρων του και πως η γραμμή βάσης του προκύπτοντος  $\epsilon$ -τμήματος, το οποίο δημιουργείται με την ένωση των τμημάτων  $AA'$  και  $\Pi\Pi'$  με το αρχικό, έχει μήκος  $BL_F = \epsilon$ . Η διαδικασία υλοποιείται στο περιβάλλον του λογισμικού πακέτου ArcGIS®, μέσω του μοντέλου που παρουσιάζεται στο Σχήμα 4.2. Αναλυτική περιγραφή του μοντέλου παρατίθεται στο Παράρτημα Π4.1.1.





Σχήμα 4.2: Μοντέλο διεύρυνσης μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων.

#### 4.1.2 Επεξεργασία διευρυμένων μονόπλευρων $\epsilon$ -μη κυρτών τμημάτων

Με την ολοκλήρωση της διαδικασίας διεύρυνσης, έχει εξασφαλισθεί η δυνατότητα εισχώρησης του θεωρητικού δίσκου του Perkal διαμέτρου  $\epsilon$  στα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα. Όπως αναφέρεται στο κεφάλαιο 4.1, η ευκρινής απόδοση των τελευταίων στην παράγωγη κλίμακα προϋποθέτει, εκτός της εισχώρησης, και την κύλιση του θεωρητικού δίσκου κατά μήκος τους. Για τη διασφάλιση της δεύτερης αυτής συνθήκης ευαναγνωσιμότητας, τα διευρυμένα  $\epsilon$ -τμήματα υπόκεινται σε μια αλληλουχία μετασχηματισμών, η οποία περιλαμβάνει την εξομάλυνση, τη διόγκωση και, στις περιπτώσεις που απαιτείται, τον ομοπαράλληλο μετασχηματισμό τους. Οι παραπάνω τελεστές εφαρμόζονται σειριακά, στην εκάστοτε προκύπτουσα εκδοχή του  $\epsilon$ -τμήματος. Η διαδικασία εντάσσεται σε ένα ενιαίο διαδραστικό μοντέλο, στο οποίο αρχικά ο χρήστης θέτει την παράμετρο εξομάλυνσης. Εν συνέχεια, το σύστημα υπολογίζει χαρακτηριστικά του  $\epsilon$ -τμήματος (καμπές, κρίσιμα σημεία) και με βάση αυτά (και σε συνδυασμό με το μέγεθος του δίσκου του Perkal) ορίζει την παράμετρο διόγκωσης, επιλέγει την εφαρμογή ή μη του ομοπαράλληλου μετασχηματισμού και θέτει τις παραμέτρους στην περίπτωση που ο τελευταίος εκτελείται. Το μοντέλο γενίκευσης της γραμμής παρέχει τη δυνατότητα ελέγχου της ευαναγνωσιμότητας του γενικευμένου  $\epsilon$ -τμήματος (βλ. κεφ. 4.6) δηλαδή, απρόσκοπτη κύλιση του δίσκου του Perkal ή δημιουργία  $\epsilon$ -μη-κυρτών τμημάτων με μήκος που πλησιάζει το μέγεθος του ορίου της οπτικής αντίληψης. Σε ενδεχόμενο μη επίτευξης αυτής, ο χρήστης μπορεί να παρέμβει, ορίζοντας εκ νέου την παράμετρο εξομάλυνσης ή θέτοντας εξωτερικά την παράμετρο εφαρμογής του τελεστή διόγκωσης. Σε κάθε περίπτωση, επιδιώκεται ο ελάχιστος αλλά αναγκαίος

μετασχηματισμός του  $\epsilon$ -τμήματος, ώστε να διασφαλίζονται οι συνθήκες ευκρινούς απόδοσής του στην παράγωγη κλίμακα. Αναλυτική περιγραφή της διαδικασίας παρουσιάζεται στις ενότητες που ακολουθούν.

#### 4.1.2.1 Εξομάλυνση και υπολογισμός γεωμετρικών χαρακτηριστικών των μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων

Τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα στην αρχική τους μορφή, αποδίδονται με λεπτομέρεια ανάλογη της κλίμακας. Η λεπτομέρεια αυτή είναι δυνατό να επηρεάσει τον μετασχηματισμό διόγκωσης που εφαρμόζεται στο επόμενο στάδιο επεξεργασίας και υλοποιείται μέσω του αλγορίθμου 'Balloon' (Lecordix et al. 1997) (βλ. κεφ. 4.1.2.3). Και τούτο διότι, ο αλγόριθμος έχει σχεδιασθεί για την επεξεργασία ομαλών ή μη πολύπλοκων γραμμικών οντοτήτων. Λόγω της αρχής λειτουργίας του (βλ. κεφ. 3.5), ο αλγόριθμος εμφανίζεται αρκετά ευαίσθητος σε ανωμαλίες που πιθανά υπάρχουν στη γραμμή, με αποτέλεσμα να προκύπτουν χαρτογραφικά και αισθητικά μη αποδεκτές παράγωγες γραμμές, όπως αυτή που παρουσιάζεται στα παράδειγμα του Σχήματος 4.3α. Για την επίλυση του προβλήματος, σε ένα πρώτο στάδιο επεξεργασίας, τα διευρυμένα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα εξομαλύνονται, ώστε να εξαλειφθούν οι τοπικές λεπτομέρειες. Η εξομάλυνση πραγματοποιείται μέσω του τελεστή *Gauss*, όπως αυτός έχει προσαρμοσθεί και ενσωματωθεί στο μοντέλο γενίκευσης (βλ. κεφ. 3.2.2).



**Σχήμα 4.3:** Παραδείγματα εφαρμογής αλγορίθμου 'Balloon' σε μη εξομαλυσμένη ή μη επαρκώς εξομαλυσμένη γραμμή.

Ο βαθμός της εξομάλυνσης αποτελεί μία παράμετρο που επιδρά σημαντικά στην ποιότητα του τελικού προϊόντος για δύο λόγους. Ο πρώτος αφορά στη λειτουργία του αλγορίθμου 'Balloon' που αναφέρεται παραπάνω. Δοκιμαστικές εφαρμογές του αλγορίθμου σε γραμμές εξομαλυσμένες σε διάφορα επίπεδα, αναδεικνύουν το γεγονός πως ακόμη και μικρές ανωμαλίες της γραμμής μπορεί να επιφέρουν μη αποδεκτά αποτελέσματα (βλ. Σχήμα 4.3β). Για το λόγο αυτό επιλέγεται τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα να εξομαλύνονται σε βαθμό που δυνητικά να εξαλείφονται όλες οι τοπικές λεπτομέρειες. Για την ποσοτικοποίηση των τοπικών λεπτομερειών εντοπίζονται και υπολογίζεται ο αριθμός των καμπών (bends) που απαρτίζουν το μονόπλευρο  $\epsilon$ -μη-κυρτό τμήμα, τόσο στην αρχική, όσο και στην εκάστοτε εξομαλυσμένη εκδοχή του. Η διαδικασία υλοποιείται μέσω της μεθόδου

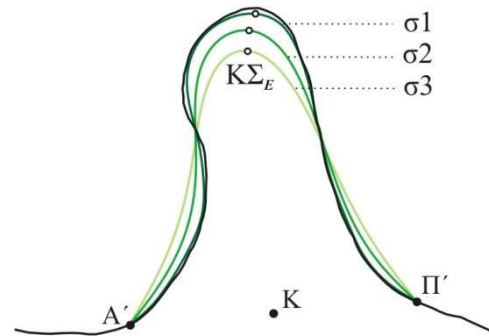
εντοπισμού καμπών, η οποία βασίζεται στον ορισμό που προτείνουν οι Wang και Müller (1998) για τις καμπές και έχει προσαρμοσθεί και ενσωματωθεί στο μοντέλο γενίκευσης, όπως περιγράφεται στο κεφάλαιο 3.3. Κάθε καμπή που ανευρίσκεται κατά μήκος του  $\epsilon$ -τμήματος αντιστοιχεί σε ένα κυματισμό/ανωμαλία αυτού. Ο αριθμός των καμπών λοιπόν, αυξάνει ανάλογα με την πολυπλοκότητά του. Εντοπίζοντας τις καμπές στο εκάστοτε εξομαλυσμένο  $\epsilon$ -τμήμα παρέχεται η δυνατότητα στο χρήστη να εκτιμήσει την καταλληλότητα του βαθμού εξομάλυνσης που έχει επιλέξει, ώστε να εφαρμοσθούν οι μετέπειτα μετασχηματισμοί γενίκευσης. Πρακτικά, τα  $\epsilon$ -τμήματα θα πρέπει να εξομαλύνονται σε βαθμό που να ελαχιστοποιείται ο αριθμός των καμπών και ιδανικά να εντοπίζεται μία. Η διαδικασία λαμβάνει χώρα και στο αρχικό  $\epsilon$ -τμήμα. Ο αριθμός των καμπών δηλώνει την πολυπλοκότητά του και παρέχεται ως πληροφορία στο χρήστη, ώστε να προεκτιμηθούν τα αποτελέσματα της εξομάλυνσης.

Εκτός των καμπών, σε κάθε εξομαλυσμένο  $\epsilon$ -τμήμα ανευρίσκονται και καταγράφονται η κορυφή και τα άκρα του. Ως κορυφή τίθεται το κρίσιμο σημείο ( $K_{\Sigma\epsilon}$  στο Σχήμα 4.4) που εντοπίζεται μετά από εφαρμογή του δείκτη αναλογίας μήκους ALR (βλ. κεφ. 3.1). Σύμφωνα με τον ορισμό του τελευταίου, πρόκειται για το σημείο όπου το εξομαλυσμένο  $\epsilon$ -τμήμα παρουσιάζει τη μέγιστη καμπυλότητα. Στο βαθμό εξομάλυνσης που συνήθως επιλέγεται (ώστε στο  $\epsilon$ -τμήμα να ανιχνεύεται ο ελάχιστος αριθμός καμπών), η κορυφή του εξομαλυσμένου  $\epsilon$ -τμήματος εντοπίζεται σε θέσεις γύρω από το μέσο του, όπως φαίνεται στο παράδειγμα του Σχήματος 4.4. Παράλληλα, καταγράφεται ο αύξοντας αριθμός της κορυφής  $K_{\Sigma\epsilon}$  και αντιστοιχίζεται με αυτόν στο αρχικό  $\epsilon$ -τμήμα. Το σημείο αυτό ( $K_{\Sigma\alpha}$ ) ορίζεται ως η κορυφή του μονόπλευρου  $\epsilon$ -τμήματος. Στη συνήθη μορφή των μονόπλευρων  $\epsilon$ -τμημάτων και λόγω του τρόπου που μετασχηματίζονται μέσω του τελεστή εξομάλυνσης *Gauss*, η περιοχή γύρω από την κορυφή αντιστοιχεί στην περιοχή του αρχικού  $\epsilon$ -τμήματος που χαρακτηρίζεται από τη μεγαλύτερη 'στένωση'. Στα παραδείγματα των Σχημάτων 4.5 $\alpha_1$ , 4.6 $\alpha_1$  και 4.7 $\alpha_1$ , το γεγονός γίνεται αντιληπτό αν παρατεθούν η θέση της κορυφής με αυτή του κρίσιμου σημείου ( $K_{\Sigma\alpha}$ ) στο αρχικό  $\epsilon$ -τμήμα. Σε εξαιρετικές περιπτώσεις όπως του παραδείγματος στο Σχήμα 4.8 $\alpha_1$ , όπου στο μονόπλευρο  $\epsilon$ -τμήμα δεν διακρίνεται κάποια εμφανής 'στένωση', η κορυφή συνήθως αντιστοιχεί σε θέση στο μέσο του αρχικού  $\epsilon$ -τμήματος. Τέλος, καταγράφονται τα άκρα του εξομαλυσμένου  $\epsilon$ -τμήματος (Α' και Π' στο Σχήμα 4.4), τα οποία ταυτίζονται με εκείνα του αρχικού αφού, βάσει της αρχής λειτουργίας του τελεστή εξομάλυνσης *Gauss* όπως αυτός έχει προσαρμοσθεί και ενσωματωθεί στο μοντέλο γενίκευσης, παραμένουν σταθερά κατά τη διαδικασία εξομάλυνσης.

Όπως αναφέρεται στο κεφάλαιο 3.2.2, ένα χαρακτηριστικό του τελεστή εξομάλυνσης *Gauss* είναι πως όταν εφαρμόζεται σε καμπύλες ή κλειστές γραμμικές οντότητες επιφέρει τη συρρίκνωσή τους, αφού κάθε σημείο κλίνει προς το τοπικό κέντρο καμπυλότητας. Λόγω του σχήματός τους, το φαινόμενο παρατηρείται κατά την εφαρμογή του τελεστή στα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα. Εξετάζοντας το  $\epsilon$ -τμήμα ως ενιαία οντότητα (και ανεξάρτητα της διεύθυνσης μετάθεσης του

εκάστοτε σημείου) συμπεραίνεται πως η εξομαλυσμένη εκδοχή του εμφανίζεται μετατοπισμένη προς το κέντρο  $K$  του δίσκου του Perkal (βλ. Σχήματα 4.4 έως 4.8). Το γεγονός ότι ο μετασχηματισμός του  $\epsilon$ -τμήματος είναι καθορισμένος συμβάλει στη σύνδεση της διαδικασίας εξομάλυνσης με τις μετέπειτα διεργασίες γενίκευσης.

Ιδιαίτερο ενδιαφέρον παρουσιάζουν οι θέσεις όπου εντοπίζεται η κορυφή του εξομαλυσμένου  $\epsilon$ -τμήματος. Όπως φαίνεται στο παράδειγμα του Σχήματος 4.4, αυξανόμενου του βαθμού εξομάλυνσης η κορυφή  $K\Sigma_E$  εντοπίζεται πάντα εγγύτερα στο κέντρο  $K$ . Στο κεφάλαιο 4.1.2.2.1 περιγράφεται αναλυτικά πως από τη γεωμετρία που διαμορφώνεται μεταξύ της κορυφής  $K\Sigma_E$  του εξομαλυσμένου  $\epsilon$ -τμήματος, της κορυφής  $K\Sigma_A$  της αρχικής εκδοχής του και του δίσκου του Perkal στη θέση  $K$ , εξαρτώνται τόσο η μέθοδος διόγκωσης που θα εφαρμοσθεί, όσο και η τιμή του συντελεστή 'Inflating' του αλγορίθμου 'Balloon' και ο ομοπαράλληλος μετασχηματισμός. Ουσιαστικά, η κορυφή  $K\Sigma_E$  έχει θεμελιώδη ρόλο στον μετασχηματισμό του  $\epsilon$ -τμήματος. Ο βαθμός της εξομάλυνσης λοιπόν, από τον οποίο εξαρτάται η θέση της κορυφής, επηρεάζει άμεσα το αποτέλεσμα της γενίκευσης.



**Σχήμα 4.4:** Παράδειγμα εφαρμογής τελεστή εξομάλυνσης Gauss σε μονόπλευρο  $\epsilon$ -μη-κυρτό τμήμα.

Οι τιμές της μεταβλητής  $\sigma$  του τελεστή Gauss που μπορούν να χρησιμοποιηθούν για την εξομάλυνση του εκάστοτε  $\epsilon$ -τμήματος, υπολογίζονται (βλ. κεφ. 3.2.2) και παρέχονται στον χρήστη. Μετά από κάθε δοκιμαστική εφαρμογή του τελεστή καταγράφονται και επίσης παρέχονται ως πληροφορία, ο αριθμός καμπών του εξομαλυσμένου  $\epsilon$ -τμήματος, οι κορυφές  $K\Sigma_E$  και  $K\Sigma_A$ , καθώς και οι βάσει αυτών παράμετροι διόγκωσης. Συνεκτιμώντας τα παραπάνω, ο χρήστης καλείται να επιλέξει τον κατάλληλο βαθμό εξομάλυνσης, με τον οποίο θα επιτευχθεί ο βέλτιστος μετασχηματισμός του  $\epsilon$ -τμήματος. Δεδομένης της άμεσης συνάρτησης των διεργασιών εξομάλυνσης και διόγκωσης, κρίνεται αναγκαίο αυτές να αντιμετωπίζονται ενιαία.

#### 4.1.2.2. Διόγκωση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων

Στο τελευταίο επίπεδο επεξεργασίας, το εξομαλυσμένο μονόπλευρο  $\epsilon$ -τμήμα υπόκειται στη διαδικασία διόγκωσης. Πρόκειται για το στάδιο με το οποίο ολοκληρώνονται οι διεργασίες για τη δημιουργία των συνθηκών κύλισης του δίσκου του Perkal κατά μήκος του. Για την υλοποίηση του τελεστή δημιουργούνται και ενσωματώνονται στο διαδραστικό σύστημα γενίκευσης οι αλγόριθμοι 'Δεσμευμένης' και 'Ανεξάρτητης' διόγκωσης. Ο αλγόριθμος 'Δεσμευμένης'

διόγκωσης αποτελεί το βασικό 'εργαλείο' περάτωσης της διαδικασίας και εκτελείται σε κάθε εφαρμογή, ενώ ο δεύτερος χρησιμοποιείται επικουρικά και μόνο στις περιπτώσεις μη επίτευξης των συνθηκών ευαναγνωσιμότητας από τον πρώτο. Η λειτουργία και των δύο στηρίζεται στο συνδυασμό του ομοπαράλληλου μετασχηματισμού με αυτόν της διόγκωσης, όπως πραγματοποιείται μέσω του αλγορίθμου 'Balloon' (Lecordix et al. 1997). Ο κάθε ένας από τους προαναφερθέντες γεωμετρικούς μετασχηματισμούς συμβάλλει με διαφορετικό τρόπο στη διαμόρφωση του παράγωγου  $\epsilon$ -τμήματος.

Σύμφωνα με το θεωρητικό πλαίσιο του αλγορίθμου 'Balloon' (βλ. κεφ. 3.5), κάθε σημείο που δομεί το γραμμικό τμήμα μετατοπίζεται κάθετα σε αυτό, με κατεύθυνση εξωτερικά του και σε απόσταση που εξαρτάται από τη θέση του ως προς το εγγύτερο άκρο και ένα συντελεστή ('Inflating') που τίθεται από τον χρήστη. Αποτέλεσμα είναι η σταδιακή διόγκωση του γραμμικού τμήματος από μία μηδενική τιμή στα άκρα μέχρι μία μέγιστη στην κορυφή. Η αρχή λειτουργίας του αλγορίθμου καλύπτει θεμελιώδεις απαιτήσεις όσον αφορά την επιδιωκόμενη διόγκωση των μονόπλευρων  $\epsilon$ -τμημάτων. Πιο συγκεκριμένα, η μη μετάθεση των δύο άκρων είναι απαραίτητη ώστε να επιτυγχάνεται η σύνδεσή τους με τα γειτονικά  $\epsilon$ -τμήματα στη μετέπειτα επεξεργασία της γραμμής. Η διεύθυνση μετατόπισης των σημείων διασφαλίζει πως η καμπυλότητα των εξομαλυσμένων (και κατ' επέκταση των αρχικών)  $\epsilon$ -τμημάτων διατηρείται και στη διογκωμένη εκδοχή τους. Τέλος, η συνήθης μορφή των μονόπλευρων  $\epsilon$ -τμημάτων χαρακτηρίζεται από ένα μέγιστο 'άνοιγμα' στη γραμμή βάσης, το οποίο ελαττώνεται στις ενδιάμεσες θέσεις και καταλήγει σε μια περιοχή όπου παρατηρείται η μεγαλύτερη 'στένωση' (βλ. Σχήματα 4.1 έως 4.7). Όπως περιγράφεται παρακάτω, εντός αυτής ανευρίσκεται η κορυφή τους. Η σταδιακά αυξανόμενη, λοιπόν, διόγκωση των  $\epsilon$ -τμημάτων από τα άκρα στην κορυφή κρίνεται ως ένας αποτελεσματικός μετασχηματισμός, αν ληφθεί υπόψη και το θεωρητικό ζητούμενο της κύλισης του δίσκου του Perkal κατά μήκος τους.

Όπως αναφέρεται στο κεφάλαιο 4.1.1, τα άκρα του διευρυμένου μονόπλευρου  $\epsilon$ -μη-κυρτού τμήματος (Α' και Π' στο Σχήμα 4.1) εντοπίζονται όταν το κέντρο του δίσκου του Perkal με διάμετρο  $\epsilon$  βρίσκεται στη θέση Κ. Με αφετηρία τη χαρακτηριστική αυτή θέση, όπου διασφαλίζεται η δυνατότητα εισχώρησης, ο δίσκος θεωρητικά πρέπει να κυλήσει κατά μήκος του  $\epsilon$ -τμήματος. Στην πραγματικότητα, όμως, η πορεία κύλισης του δίσκου δεν είναι δυνατό να προβλεφθεί και κατ' επέκταση να υλοποιηθεί. Για το λόγο αυτό, ακολουθείται μία τεχνική προσομοίωσης της διαδικασίας. Θεωρώντας το δίσκο σταθερό στη θέση Κ, λαμβάνει χώρα η διόγκωση του εξομαλυσμένου  $\epsilon$ -τμήματος μέχρις ότου το παράγωγο  $\epsilon$ -τμήμα: α) να αναπτύσσεται εξωτερικά ή/και να εφάπτεται του δίσκου, β) να αναπτύσσεται εξωτερικά ή/και εσωτερικά ή/και να εφάπτεται του δίσκου και το ή τα πολύγωνα που δημιουργούνται μεταξύ τους να μην είναι ορατά στην παράγωγη κλίμακα. Με την επίτευξη ενός εκ των δύο κριτηρίων, διασφαλίζεται η ευαναγνωσιμότητα του  $\epsilon$ -τμήματος, όταν ο δίσκος παραμένει στην αρχική του θέση. Αυτή είναι η ελάχιστη διόγκωση που υφίσταται κατά την επεξεργασία του το εξομαλυσμένο  $\epsilon$ -τμήμα.

Τα πολύγωνα που αναφέρονται παραπάνω διαμορφώνονται ανάμεσα στα μέρη του διογκωμένου  $\epsilon$ -τμήματος και του δίσκου, τα οποία αποκόπτονται μεταξύ των σημείων όπου αυτά τέμνονται. Η εμφάνισή τους, λοιπόν, υποδηλώνει τη μη διασφάλιση των προϋποθέσεων απρόσκοπτης κύλισης του δίσκου και τη δημιουργία νέων  $\epsilon$ -μη-κυρτών τμημάτων. Σύμφωνα με τους κανόνες της οπτικής αντίληψης (βλ. κεφ. 2.4.2), τα τελευταία δε γίνονται αντιληπτά σαν ξεχωριστές οντότητες από τον αναγνώστη, όταν το εμβαδό των πολυγώνων είναι μικρότερο μίας ανοχής. Στις περιπτώσεις αυτές, το διογκωμένο  $\epsilon$ -τμήμα θεωρείται ενιαίο και ευαναγνωστο στην παράγωγη κλίμακα. Κάτι τέτοιο δεν ισχύει όταν το εμβαδό των πολυγώνων είναι μεγαλύτερο της ανοχής. Το γεγονός φανερώνει πως είτε το  $\epsilon$ -τμήμα δεν έχει διογκωθεί επαρκώς ώστε να επιτυγχάνονται οι προϋποθέσεις ευκρινούς απόδοσής του όταν ο δίσκος βρίσκεται στη θέση Κ, είτε πως αυτό έχει επιτευχθεί, ο δίσκος θεωρητικά έχει κυλήσει από την αρχική του θέση, όμως έχει τμήσει σε εσωτερικά του σημεία το  $\epsilon$ -τμήμα. Σε κάθε περίπτωση, απαιτείται η μεγαλύτερη διόγκωση του εξομαλυσμένου  $\epsilon$ -τμήματος. Βάσει αυτής της λογικής, ελέγχοντας κάθε φορά το μέγεθος ή την ύπαρξη πολυγώνων, ουσιαστικά προσομοιώνεται η διαδικασία κύλισης του δίσκου.

Στην πράξη, λοιπόν, η εφαρμογή του γεωμετρικού μετασχηματισμού στηρίζεται στην παραδοχή πως ο δίσκος του Perkal παραμένει σταθερός στη θέση Κ. Με αυτό ως μέτρο, το εξομαλυσμένο  $\epsilon$ -τμήμα διογκώνεται μέχρι η προκύπτουσα εκδοχή του να ικανοποιεί ένα από τα ακόλουθα κριτήρια:

1. να αναπτύσσεται εξωτερικά ή/και να εφάπτεται του δίσκου και να μη δημιουργούνται πολύγωνα μεταξύ τους,
2. να αναπτύσσεται εξωτερικά ή/και να εφάπτεται του δίσκου και το πολύγωνο που δημιουργείται μεταξύ τους να μην είναι ορατό στην παράγωγη κλίμακα,
3. να αναπτύσσεται εξωτερικά ή/και εσωτερικά ή/και να εφάπτεται του δίσκου και το ή τα πολύγωνα που δημιουργούνται μεταξύ τους να μην είναι ορατά στην παράγωγη κλίμακα.

Το μέγεθος της διόγκωσης επιδιώκεται να είναι το ικανό και αναγκαίο, ώστε να δημιουργούνται οι συνθήκες κύλισης με τη δυνατόν μικρότερη παραμόρφωση του  $\epsilon$ -τμήματος. Η επίτευξη αυτών ελέγχεται σε επόμενο στάδιο επεξεργασίας (βλ. κεφ. 4.6), κατά το οποίο αξιολογείται ποιοτικά η γενικευμένη γραμμή στο σύνολό της, με βάση τους κανόνες οπτικής αντίληψης που έχουν ορισθεί. Επιγραμματικά, σε ό,τι αφορά τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα, η διαδικασία περιλαμβάνει την εξέταση εμφάνισης ή μη πολυγώνων και, σε περίπτωση που παρατηρείται το πρώτο, τον υπολογισμό και σύγκριση του εμβαδού τους με την ανοχή. Το συμπέρασμα του ελέγχου οπτικοποιείται, ώστε ο χρήστης να πληροφορείται άμεσα και εύκολα σχετικά με την αποτελεσματικότητα της διόγκωσης.

#### **4.1.2.2.1 Διόγκωση με δεσμεύσεις – Αλγόριθμος ‘Δεσμευμένης Διόγκωσης’**

Ο γεωμετρικός μετασχηματισμός που περιγράφεται στην προηγούμενη ενότητα υλοποιείται μέσω του αλγορίθμου ‘Δεσμευμένης Διόγκωσης’. Ο αλγόριθμος είναι ενταγμένος στο σύστημα γενίκευσης των μονόπλευρων  $\epsilon$ -τμημάτων. Το γεγονός

παρέχει τη δυνατότητα αυτόματης εκτέλεσής του, χωρίς την παρέμβαση του χρήστη, αφού οι παράμετροι εφαρμογής του καθορίζονται βάσει πληροφοριών/δεσμεύσεων, με τις οποίες το ίδιο το σύστημα τον τροφοδοτεί.

Σύμφωνα με το θεωρητικό πλαίσιο του αλγορίθμου, η διαδικασία περατώνεται σε δύο αλληλένδετα επίπεδα. Το πρώτο, αυτό της προεπεξεργασίας, αφορά στην ανάλυση της γεωμετρίας που διαμορφώνεται μεταξύ της κορυφής ΚΣ<sub>Ε</sub> του εξομαλυσμένου  $\epsilon$ -τμήματος, της κορυφής ΚΣ<sub>Α</sub> της αρχικής εκδοχής του και του δίσκου του Perkal στη θέση Κ (στο εξής αναφέρεται ως γεωμετρία 'θέσης Κ'). Οι παραπάνω σταθερές έχουν ανευρεθεί στα προηγούμενα στάδια επεξεργασίας (βλ. κεφ. 4.1.1 και 4.1.2.1) και εισάγονται αυτόματα, ως δεδομένα στον αλγόριθμο. Με βάση την ανάλυση της γεωμετρίας στη 'θέση Κ' επιλέγεται η μέθοδος και προσδιορίζεται ο βαθμός της διόγκωσης του εξομαλυσμένου  $\epsilon$ -τμήματος.

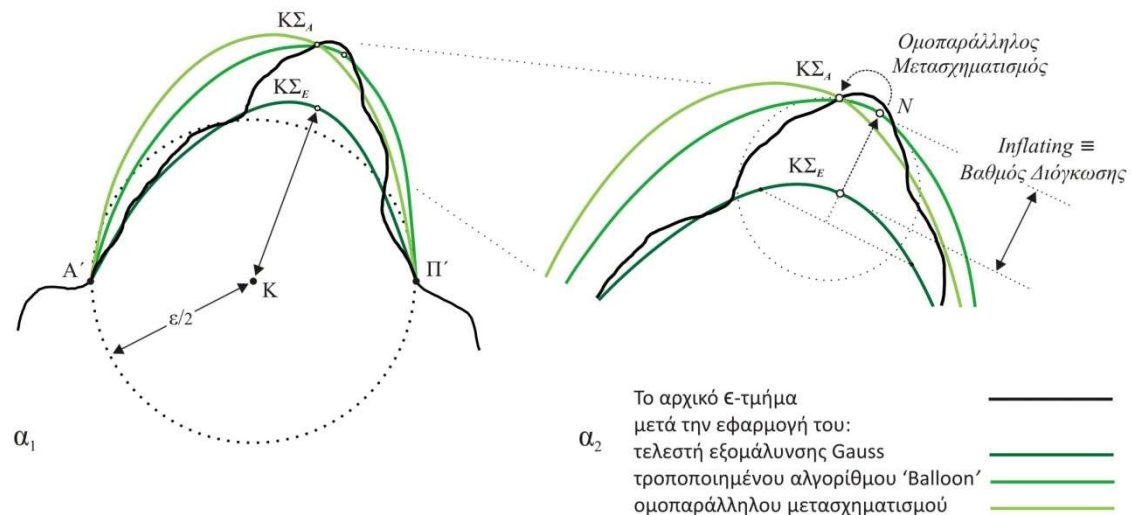
Όσον αφορά τη μέθοδο, ο αλγόριθμος περιλαμβάνει και παρέχει τη δυνατότητα εκτέλεσης μίας τροποποιημένης έκδοσης του αλγορίθμου 'Balloon', καθώς και του ομοπαράλληλου μετασχηματισμού. Ανάλογα με τη γεωμετρία στη 'θέση Κ', η διαδικασία διόγκωσης περατώνεται είτε εφαρμόζοντας διαδοχικά (κατά τη σειρά που αναφέρονται) τους δύο μετασχηματισμούς, είτε χρησιμοποιώντας αποκλειστικά τον πρώτο. Επιπρόσθετα, για κάθε περίπτωση καθορίζεται ο τρόπος υλοποίησης του ή των μετασχηματισμών. Η τροποποιημένη έκδοση του αλγορίθμου 'Balloon' διατηρεί τη φιλοσοφία και την αρχή λειτουργίας του πρωτότυπου. Η διαφορά έγκειται στο γεγονός ότι η τιμή του συντελεστή 'Inflating' δεν τίθεται εξωτερικά από το χρήστη. Όπως περιγράφεται στα επόμενα, ορίζεται αυτόματα από τον αλγόριθμο, βάσει κανόνων και υπό δεσμεύσεις που απορρέουν από την ανάλυση της γεωμετρίας στη 'θέση Κ'.

Σύμφωνα με τη φιλοσοφία της μεθόδου εντοπισμού της, η κορυφή ΚΣ<sub>Ε</sub> αποτελεί το πλέον χαρακτηριστικό σημείο του εξομαλυσμένου  $\epsilon$ -τμήματος. Το γεγονός αιτιολογεί το δομικό της ρόλο στη λειτουργία τόσο του τροποποιημένου αλγορίθμου 'Balloon', όσο και του ομοπαράλληλου μετασχηματισμού. Από τη μετάθεση της κορυφής εξαρτάται εκείνη των υπολοίπων σημείων του  $\epsilon$ -τμήματος. Αυτή, λοιπόν, χρησιμοποιείται ως μέτρο για τον προσδιορισμό του βαθμού διόγκωσης. Ένας βασικός κανόνας που ακολουθείται κατά την εκτέλεση του αλγορίθμου είναι η κορυφή ΚΣ<sub>Ε</sub>, μετά το γεωμετρικό μετασχηματισμό, να εντοπίζεται εκτός ή κατά το ελάχιστο πάνω στο δίσκο του Perkal. Θεωρητικά δηλαδή, στη νέα της θέση να μην ανακόπτει την κύλισή του. Λαμβάνοντας αυτό σαν κατώφλι, επιδιώκεται η μετάθεσή της σε μεγαλύτερες αποστάσεις, κατάλληλες ώστε και τα ενδιάμεσα σημεία του  $\epsilon$ -τμήματος να εντοπίζονται σε θέσεις εκτός, πάνω ή πλησίον του δίσκου. Σε επίπεδο υλοποίησης, η κορυφή μετατοπίζεται υπό συγκεκριμένους κανόνες. Κατά την εφαρμογή του τροποποιημένου αλγορίθμου 'Balloon' κάθετα στο  $\epsilon$ -τμήμα, με κατεύθυνση εξωτερικά αυτού και σε απόσταση ίση με την τιμή του συντελεστή 'Inflating', ενώ μέσω του ομοπαράλληλου μετασχηματισμού κατά την απαιτούμενη διεύθυνση και απόσταση, ώστε να ταυτισθεί με την κορυφή ΚΣ<sub>Α</sub> του αρχικού  $\epsilon$ -τμήματος (βλ. Σχήματα 4.5, 4.6, 4.7). Το μέγεθος της μετάθεσής της, λοιπόν, εξαρτάται από τη μέθοδο διόγκωσης που ακολουθείται και από τα διαστήματα που απέχουν τα σημεία από τα οποία διέρχεται. Ουσιαστικά, δηλαδή, από τη γεωμετρία στη 'θέση Κ'. Ο βαθμός

διόγκωσης του  $\epsilon$ -τμήματος ισοδυναμεί με την απόσταση μεταξύ της αρχικής θέσης της κορυφής  $K_{\Sigma_E}$  και αυτής που εντοπίζεται μετά τη διόγκωση.

Εξετάζοντας τη θέση των κορυφών του εξομαλυσμένου και του αρχικού  $\epsilon$ -τμήματος ( $K_{\Sigma_E}$  και  $K_{\Sigma_A}$ , αντίστοιχα), σε σχέση με το δίσκο του Perkal με κέντρο στο σημείο  $K$  παρατηρούνται τέσσερα ενδεχόμενα:

1. οι κορυφές  $K_{\Sigma_E}$  και  $K_{\Sigma_A}$  εντοπίζονται εκτός του δίσκου, δηλαδή:  $(K K_{\Sigma_E}) > \frac{\epsilon}{2}$   
και  $(K K_{\Sigma_A}) > \frac{\epsilon}{2}$ .



**Σχήμα 4.5:** Οι κορυφές  $K_{\Sigma_E}$  και  $K_{\Sigma_A}$  εντοπίζονται εκτός του δίσκου.

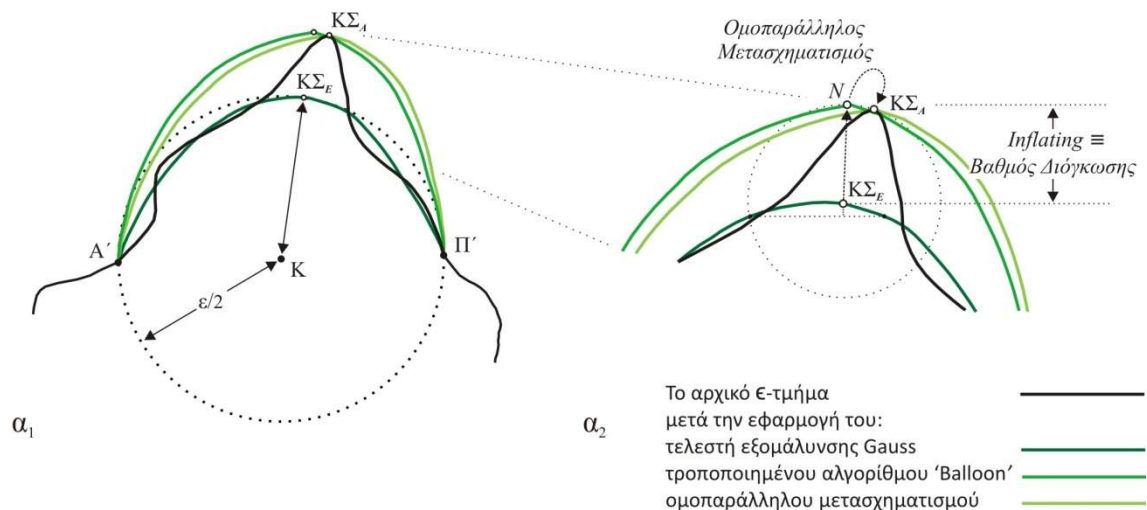
Όπως φαίνεται στο παράδειγμα του Σχήματος 4.5α<sub>1</sub>, στην περίπτωση αυτή η γειτονική στην κορυφή  $K_{\Sigma_E}$  περιοχή του εξομαλυσμένου  $\epsilon$ -τμήματος αναπτύσσεται εκτός του δίσκου. Εσωτερικά αυτού βρίσκονται τα υπόλοιπα, ενδιάμεσα μέρη του  $\epsilon$ -τμήματος και είναι εκείνα που παρεμποδίζουν την κύλισή του. Απαιτείται, λοιπόν, η μετατόπιση των σημείων που δομούν τα εν λόγω μέρη σε θέσεις εκτός, πάνω ή πλησίον του δίσκου, ώστε το  $\epsilon$ -τμήμα να αποδίδεται με ευκρίνεια στην παράγωγη κλίμακα. Η διόγκωση περατώνεται με την εκτέλεση διαδοχικά του τροποποιημένου αλγορίθμου 'Balloon' και του ομοπαράλληλου μετασχηματισμού. Κατά την υλοποίηση του πρώτου, η τιμή του συντελεστή διόγκωσης υπολογίζεται αυτόματα από το σύστημα βάσει της σχέσης:  $Inflating = (K_{\Sigma_E} K_{\Sigma_A})'$  ορίζεται δηλαδή ίση με την απόσταση μεταξύ των κορυφών  $K_{\Sigma_E}$  και  $K_{\Sigma_A}$ . Τα υπόλοιπα σημεία μετατοπίζονται σε μικρότερες αποστάσεις, ανάλογες αυτής, σύμφωνα με τη φιλοσοφία του αλγορίθμου 'Balloon'. Αποτέλεσμα είναι η διόγκωση του εξομαλυσμένου  $\epsilon$ -τμήματος. Όπως φαίνεται στο Σχήμα 4.5α<sub>2</sub>, η κορυφή  $K_{\Sigma_E}$  εντοπίζεται σε μία νέα θέση  $N$ . Ο ομοπαράλληλος μετασχηματισμός του προκύπτοντος  $\epsilon$ -τμήματος που ακολουθεί, πραγματοποιείται με γνώμονα τη μεταφορά αυτής, από την εν λόγω θέση στο σημείο  $K_{\Sigma_A}$ . Με την ολοκλήρωση της διαδικασίας διαμορφώνεται το τελικό, γενικευμένο  $\epsilon$ -τμήμα.

Με το συγκεκριμένο τρόπο επεξεργασίας, το εξομαλυσμένο  $\epsilon$ -τμήμα υπόκειται στη μέγιστη δυνατή διόγκωση μέσω του αλγορίθμου 'Balloon'. Το  $K_{\Sigma_A}$  είναι το πιο



απομακρυσμένο από την κορυφή  $KΣ_E$ , και συγχρόνως, γνωστό σημείο της περιοχής ενδιαφέροντος. Έτσι, η τιμή  $Inflating = (KΣ_E KΣ_A)$  είναι η ανώτερη με την οποία μπορεί να εφαρμοσθεί ο αλγόριθμος στα πλαίσια της ‘δεσμευμένης’ διαδικασίας. Επισημαίνεται πως, στην πραγματικότητα, η τιμή ορίζεται ίση με την απόσταση που μετατέθηκε η κορυφή  $KΣ_A$  κατά την εξομάλυνση του αρχικού  $\epsilon$ -τμήματος. Επιβεβαιώνεται, λοιπόν, η επίδραση της τελευταίας στη διαδικασία διόγκωσης, όπως αναφέρεται στο κεφάλαιο 4.1.2.1. Το γεγονός ισχύει και για τις τρεις περιπτώσεις που ακολουθούν. Ο ομοπαράλληλος μετασχηματισμός, υπό τους όρους που υλοποιείται, δεν επιφέρει επιπλέον διόγκωση στο  $\epsilon$ -τμήμα. Όπως φαίνεται στο Σχήμα 4.5 $\alpha_2$ , η κορυφή μετατοπίζεται από τη θέση  $N$  στη θέση  $KΣ_A$ , η οποία επίσης απέχει απόσταση  $(KΣ_E KΣ_A)$  από την αρχική. Με αυτή λοιπόν, ισοδυναμεί και ο συνολικός βαθμός διόγκωσης του  $\epsilon$ -τμήματος. Ουσιαστικά, το θεμελιώδη ρόλο στη διαδικασία έχει ο αλγόριθμος ‘Balloon’. Ο ομοπαράλληλος μετασχηματισμός ‘διορθώνει’ χαρτογραφικά το αποτέλεσμα, ταυτίζοντας την κορυφή του γενικευμένου  $\epsilon$ -τμήματος με εκείνη του αρχικού. Με τη χρήση του η μορφή του  $\epsilon$ -τμήματος δε μεταβάλλεται αισθητά, αφού στην πλειοψηφία των περιπτώσεων, το διάστημα μεταξύ των σημείων  $N$  και  $KΣ_A$  δεν είναι μεγάλο. Παρατηρώντας το παράγωγο  $\epsilon$ -τμήμα στο παράδειγμα του Σχήματος 4.5 $\alpha_1$ , είναι εμφανές πως η κορυφή  $KΣ_E$  και τα δύο άκρα ( $A'$  και  $\Pi'$ ) εντοπίζονται στην αρχική τους θέση. Παράλληλα, γίνεται εφικτή η ευκρινής απόδοσή του στην παράγωγη κλίμακα, αφού αναπτύσσεται σε θέσεις που δεν εμποδίζουν την κύλιση του δίσκου του Perkal κατά μήκος του.

2. η κορυφή  $KΣ_E$  εντοπίζεται πάνω στο δίσκο και η κορυφή  $KΣ_A$  εκτός αυτού, δηλαδή:  $(K KΣ_E) = \frac{\epsilon}{2}$  και  $(K KΣ_A) > \frac{\epsilon}{2}$ .

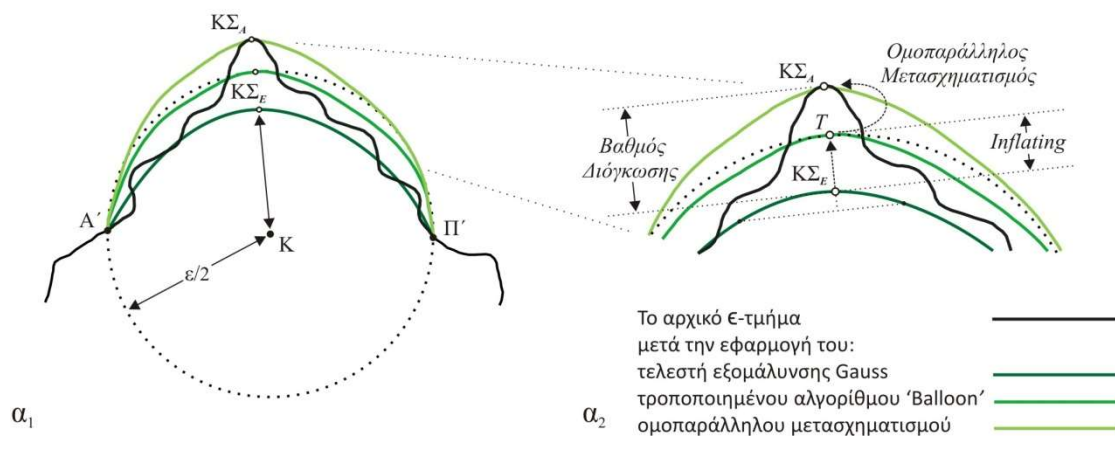


Σχήμα 4.6: Η κορυφή  $KΣ_E$  εντοπίζεται πάνω στο δίσκο και η κορυφή  $KΣ_A$  εκτός αυτού.

Όπως φαίνεται στο παράδειγμα του Σχήματος 4.6 $\alpha_1$ , η κύλιση του δίσκου δεν παρεμποδίζεται (έστω και οριακά) από την κορυφή  $KΣ_E$ , αλλά από ενδιάμεσα σημεία του εξομαλυσμένου  $\epsilon$ -τμήματος. Ουσιαστικά, το γεωμετρικό πρόβλημα είναι πανομοιότυπο με εκείνο της πρώτης περίπτωσης. Για την επίλυσή του, λοιπόν, ακολουθείται η ίδια μεθοδολογία, ακριβώς όπως περιγράφεται παραπάνω. Αξίζει

να σημειωθεί, πως το φαινόμενο του εντοπισμού της κορυφής  $K_{Σ_E}$  πάνω στο δίσκο του Perkal συναντάται εξαιρετικά σπάνια στην πράξη. Αντιμετωπίζεται, όμως, από τον αλγόριθμο 'Δεσμευμένης Διόγκωσης', ώστε να καλύπτονται όλα τα θεωρητικά ενδεχόμενα.

3. η κορυφή  $K_{Σ_E}$  εντοπίζεται εντός του δίσκου και η κορυφή  $K_{Σ_A}$  εκτός αυτού, δηλαδή:  $(KK_{Σ_E}) < \frac{\epsilon}{2}$  και  $(KK_{Σ_A}) > \frac{\epsilon}{2}$ .



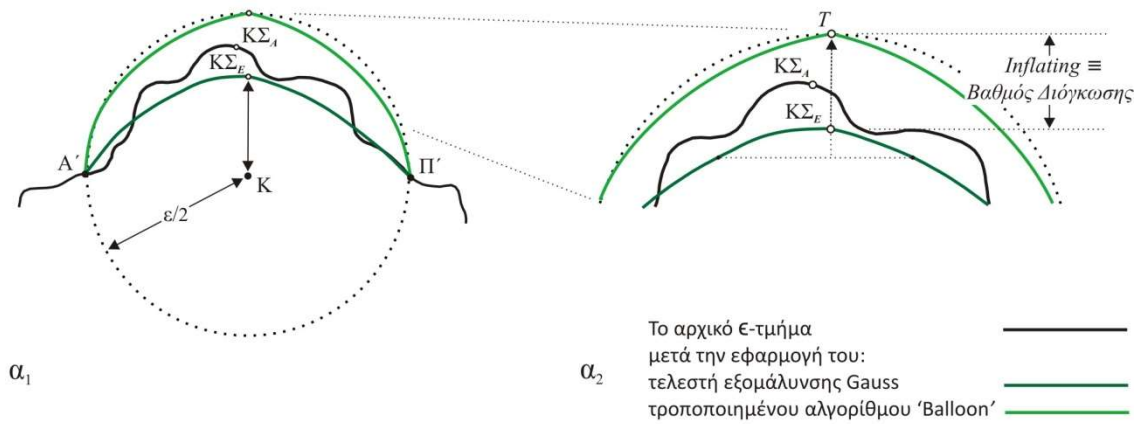
**Σχήμα 4.7:** Η κορυφή  $K_{Σ_E}$  εντοπίζεται εντός του δίσκου και η κορυφή  $K_{Σ_A}$  εκτός αυτού.

Όπως φαίνεται στο παράδειγμα του Σχήματος 4.7α<sub>1</sub>, με εξαίρεση τα δύο άκρα, το εξομαλυσμένο  $\epsilon$ -τμήμα βρίσκεται εξολοκλήρου εντός του δίσκου του Perkal. Είναι, λοιπόν, απαραίτητη η μετατόπιση του συνόλου των σημείων που το δομούν σε θέσεις εκτός, πάνω ή πλησίον του δίσκου, ώστε το  $\epsilon$ -τμήμα που δημιουργείται να αποδίδεται ευκρινώς στην παράγωγη κλίμακα. Η διόγκωση πραγματοποιείται με οδηγό τη μετάθεση της κορυφής  $K_{Σ_E}$  από την αρχική της θέση στην κορυφή  $K_{Σ_A}$ . Η διαδικασία περατώνεται σε δύο στάδια, με την εκτέλεση κατά σειρά του τροποποιημένου αλγορίθμου 'Balloon' και του ομοπαράλληλου μετασχηματισμού. Κατά την υλοποίηση του πρώτου, τίθεται ως μέτρο η μετατόπιση της κορυφής  $K_{Σ_E}$  πάνω στο δίσκο. Σε αυτό το πλαίσιο, το σύστημα εντοπίζει το σημείο  $T$ , όπου τέμνονται η κάθετος στο  $\epsilon$ -τμήμα που διέρχεται από την κορυφή  $K_{Σ_E}$  και ο δίσκος (Σχήμα 4.7α<sub>2</sub>). Ο συντελεστής διόγκωσης υπολογίζεται αυτόματα, βάσει της σχέσης  $Inflating = (K_{Σ_E}T)$  ορίζεται δηλαδή ίσος με την απόσταση μεταξύ των σημείων  $K_{Σ_E}$  και  $T$ . Σύμφωνα με τη φιλοσοφία του αλγορίθμου 'Balloon', τα υπόλοιπα σημεία του  $\epsilon$ -τμήματος μετατοπίζονται σε μικρότερες αποστάσεις, ανάλογες αυτής. Λαμβάνοντας υπόψη τη συνήθη μορφή των εξομαλυσμένων  $\epsilon$ -τμημάτων, γίνεται αντιληπτό πως μετά την εφαρμογή του αλγορίθμου υπό τις προαναφερθείσες δεσμεύσεις, ένα μέρος των ενδιαμέσων σημείων είναι πιθανό να εντοπίζεται σε γειτονικές, αλλά εσωτερικά του δίσκου θέσεις. Αυτό έχει σαν αποτέλεσμα τη μη επίτευξη της αδιάκοπτης κύλισης του τελευταίου κατά μήκος του προκύπτοντος  $\epsilon$ -τμήματος. Για το λόγο αυτό, στο δεύτερο στάδιο επεξεργασίας, λαμβάνει χώρα ο

ομοπαράλληλος μετασχηματισμός του  $\epsilon$ -τμήματος. Η διαδικασία πραγματοποιείται με γνώμονα τη μετατόπιση της κορυφής από τη θέση  $T$  στην κορυφή  $K\Sigma_A$  (Σχήμα 4.7α<sub>2</sub>). Υπό αυτούς τους όρους, το  $\epsilon$ -τμήμα υφίσταται μία επιπλέον διόγκωση, με αποτέλεσμα τη διαμόρφωση της τελικής, γενικευμένης εκδοχής του. Παρατηρώντας το παράγωγο  $\epsilon$ -τμήμα στο παράδειγμα του Σχήματος 4.7α<sub>1</sub>, γίνεται φανερό πως η κορυφή  $K\Sigma_E$  εντοπίζεται στην αρχική της θέση ( $K\Sigma_A$ ), ενώ τα δύο άκρα ( $A'$  και  $\Pi'$ ) παραμένουν αμετάθετα. Ο συνολικός βαθμός διόγκωσης είναι ( $K\Sigma_E K\Sigma_A$ ). Συγχρόνως, δημιουργούνται οι προϋποθέσεις για την απρόσκοπτη κύλιση του δίσκου του Perkal κατά μήκος του, οι προϋποθέσεις, δηλαδή, για την ευκρινή απεικόνισή του στην παράγωγη κλίμακα. Σημειώνεται πως η διαδικασία εφαρμόζεται και στην περίπτωση που η κορυφή  $K\Sigma_A$  του αρχικού  $\epsilon$ -τμήματος εντοπίζεται πάνω στο δίσκο του Perkal, όταν δηλαδή  $(K\Sigma_A) = \frac{\epsilon}{2}$ . Αν και στην πράξη κάτι τέτοιο εμφανίζεται εξαιρετικά σπάνια, αντιμετωπίζεται από τον αλγόριθμο για την κάλυψη όλων των θεωρητικών ενδεχόμενων.

Ένας εναλλακτικός τρόπος αντιμετώπισης του προβλήματος ευαναγνωσιμότητας των  $\epsilon$ -τμημάτων της κατηγορίας είναι η εφαρμογή της διόγκωσης κατά το πρότυπο των δύο προηγούμενων περιπτώσεων. Ορίζοντας, δηλαδή, την τιμή του συντελεστή 'Inflating' ίση με ( $K\Sigma_E K\Sigma_A$ ) κατά την εκτέλεση του τροποποιημένου αλγορίθμου 'Balloon' και στη συνέχεια υλοποιώντας τον ομοπαράλληλο μετασχηματισμό στο προκύπτον  $\epsilon$ -τμήμα, ούτως ώστε η κορυφή να μετατοπίζεται από τη θέση  $N$  στο σημείο  $K\Sigma_A$  (βλ. περιγραφή περίπτωσης 1 και Σχήμα 4.5α<sub>2</sub>). Το εξομαλυσμένο  $\epsilon$ -τμήμα υπόκειται στη μέγιστη δυνατή διόγκωση μέσω του αλγορίθμου 'Balloon', αφού η παραπάνω τιμή 'Inflating' είναι η ανώτερη με την οποία αυτός μπορεί να εφαρμοσθεί στα πλαίσια της 'δεσμευμένης' διαδικασίας. Ο συνολικός βαθμός διόγκωσης ισοδυναμεί με ( $K\Sigma_E K\Sigma_A$ ), δεδομένου πως δε μεταβάλλεται με την υλοποίηση του ομοπαράλληλου μετασχηματισμού. Υπό αυτές τις προϋποθέσεις, είναι εφικτή η διαμόρφωση  $\epsilon$ -τμημάτων που απεικονίζονται ευκρινώς στην παράγωγη κλίμακα. Όπως παρατηρείται, ο βαθμός διόγκωσης και οι γεωμετρικοί μετασχηματισμοί που χρησιμοποιούνται είναι ίδιοι με τους αντίστοιχους της βασικής επεξεργασίας. Η διαφορά έγκειται στον τρόπο εφαρμογής των τελευταίων. Στη συγκεκριμένη περίπτωση, το εξομαλυσμένο  $\epsilon$ -τμήμα υφίσταται μεγαλύτερη διόγκωση μέσω του αλγορίθμου 'Balloon'. Αυτό έχει ως αποτέλεσμα να παραμορφώνεται περισσότερο, ενώ υπάρχει πιθανότητα ορισμένα σημεία του να μετατίθενται σε απόσταση μεγαλύτερη από εκείνη που απαιτείται ώστε να εντοπίζονται εκτός του δίσκου του Perkal' το  $\epsilon$ -τμήμα, δηλαδή, να διογκώνεται πλέον του ικανού και αναγκαίου. Στη βασική επεξεργασία τα μειονεκτήματα αυτά περιορίζονται, αφού ο αλγόριθμος 'Balloon' 'επιδρά' λιγότερο, ενώ δίνεται μεγαλύτερη βαρύτητα στον ομοπαράλληλο μετασχηματισμό κατά τη διαδικασία διαμόρφωσης του παράγωγου προϊόντος. Αυτός είναι και ο λόγος που εντάσσεται ως κύρια λειτουργία στον αλγόριθμο 'Δεσμευμένης Διόγκωσης'. Ο εναλλακτικός τρόπος διόγκωσης μπορεί να είναι περισσότερο επαρκής σε μεμονωμένες περιπτώσεις, χωρίς όμως αυτές να ορίζουν κάποιο γενικό κανόνα εφαρμογής του.

4. οι κορυφές  $K_{Σ_E}$  και  $K_{Σ_A}$  εντοπίζονται εντός του δίσκου, δηλαδή:  $(K_{Σ_E}) < \frac{\epsilon}{2}$   
 και  $(K_{Σ_A}) < \frac{\epsilon}{2}$ .



**Σχήμα 4.8:** Οι κορυφές  $K_{Σ_E}$  και  $K_{Σ_A}$  εντοπίζονται εντός του δίσκου.

Όπως φαίνεται στο παράδειγμα του Σχήματος 4.8α<sub>1</sub>, ο δίσκος του Perkal περιβάλλει στο σύνολό τους τόσο το αρχικό, όσο και το εξομαλυσμένο  $\epsilon$ -τμήμα. Για τη διόγκωση του τελευταίου τίθεται ως μέτρο η μετατόπιση της κορυφής  $K_{Σ_E}$  σε θέση πάνω στο δίσκο. Η διαδικασία περατώνεται με την εκτέλεση αποκλειστικά του τροποποιημένου αλγορίθμου 'Balloon'. Το σύστημα εντοπίζει το σημείο  $T$ , όπου τέμνονται η κάθετος στο  $\epsilon$ -τμήμα που διέρχεται από την κορυφή  $K_{Σ_E}$  και ο δίσκος (Σχήμα 4.8α<sub>2</sub>). Η τιμή του συντελεστή διόγκωσης ορίζεται ίση με την απόσταση μεταξύ των σημείων  $K_{Σ_E}$  και  $T$  και υπολογίζεται αυτόματα από τη σχέση:  $Inflating = (K_{Σ_E}T)$ . Εφαρμόζοντας τον αλγόριθμο με τις παραπάνω δεσμεύσεις διαμορφώνεται το τελικό, γενικευμένο  $\epsilon$ -τμήμα. Στο παράδειγμα του Σχήματος 4.8α<sub>1</sub> παρατηρείται πως η κορυφή του εντοπίζεται στη θέση  $T$ . Σε αντίθεση δηλαδή με τις τρεις προηγούμενες περιπτώσεις, δεν επαναφέρεται στην κορυφή  $K_{Σ_A}$  του αρχικού  $\epsilon$ -τμήματος. Δεδομένου πως η τελευταία βρίσκεται εντός του δίσκου, γίνεται εύκολα αντιληπτό πως η υλοποίηση της διόγκωσης με κριτήριο τη μετάθεση της κορυφής  $K_{Σ_E}$  σε αυτή δε θα επέλυε το πρόβλημα ευαναγνωσιμότητας. Ο συνολικός βαθμός διόγκωσης, λοιπόν, είναι  $(K_{Σ_E}T)$ . Τα ενδιάμεσα σημεία του  $\epsilon$ -τμήματος μετατοπίζονται σε αποστάσεις μικρότερες της τιμής 'Inflating', σύμφωνα με την αρχή λειτουργίας του αλγορίθμου 'Balloon'. Στη συνήθη μορφή των εξομαλυσμένων  $\epsilon$ -τμημάτων, αυτό έχει σαν αποτέλεσμα, μετά την υλοποίηση του μετασχηματισμού, τα εν λόγω σημεία να εντοπίζονται κατά το βέλτιστο, σε περιοχές γύρω από τα όρια του δίσκου του Perkal. Στην πραγματικότητα, η πλειοψηφία τους εντοπίζεται σε γειτονικές και εντός αυτού θέσεις. Έτσι, όπως φαίνεται στο Σχήμα 4.8α<sub>1</sub>, το μεγαλύτερο μέρος του διογκωμένου  $\epsilon$ -τμήματος αναπτύσσεται εσωτερικά του δίσκου, σε κοντινή όμως απόσταση από αυτόν. Στις περισσότερες περιπτώσεις, λοιπόν, η απρόσκοπτη κύλιση του δίσκου κατά μήκος του δεν επιτυγχάνεται. Είναι όμως δυνατό το μήκος των  $\epsilon$ -μη-κυρτών τμημάτων που δημιουργούνται να είναι μικρότερο από το όριο της οπτικής αντίληψης. Σε ένα τέτοιο ενδεχόμενο, η γενίκευση του  $\epsilon$ -τμήματος θεωρείται επιτυχής, αφού εξασφαλίζονται οι προϋποθέσεις ευκρινούς απόδοσής του στην παράγωγη κλίμακα.

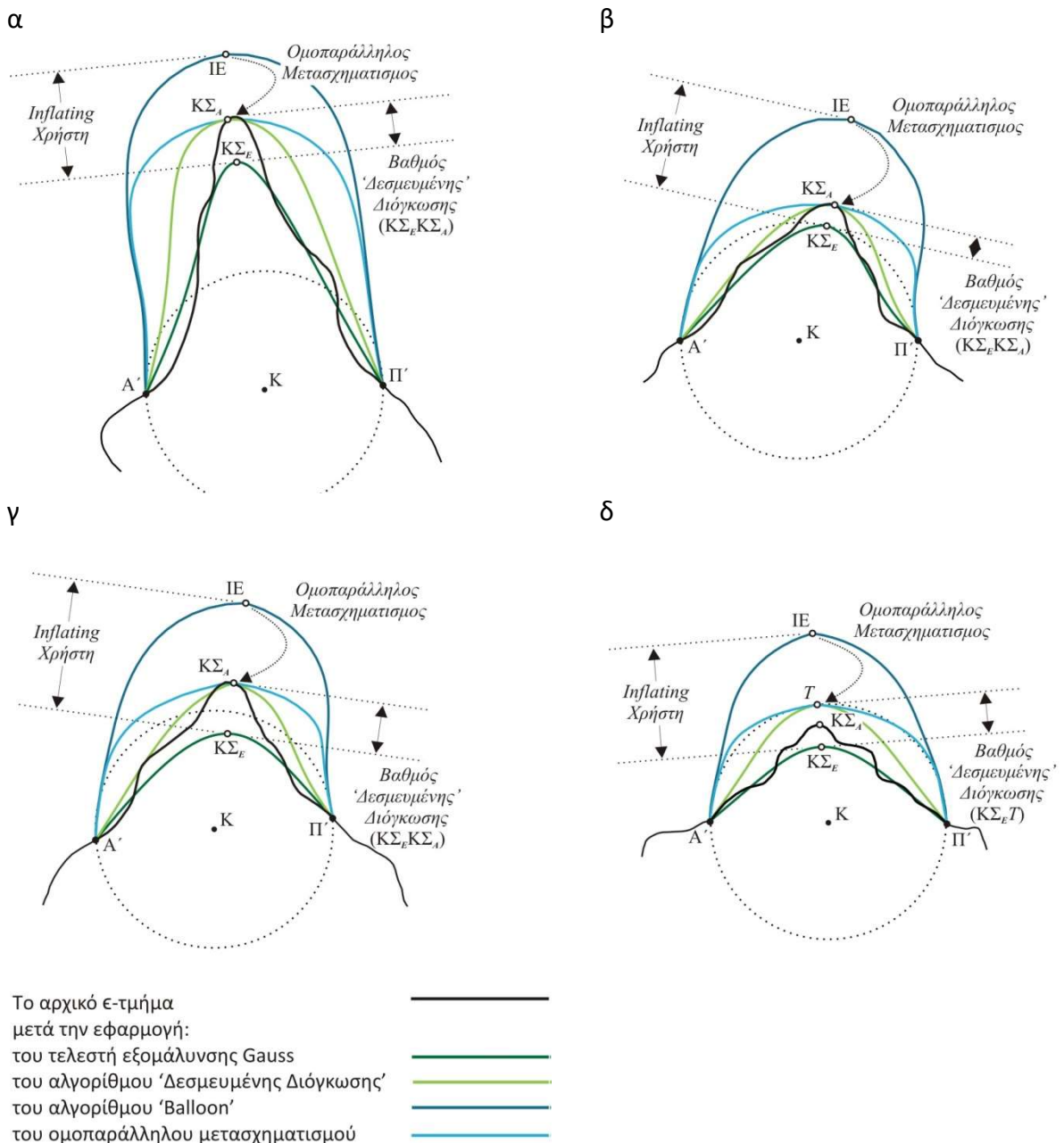
#### 4.1.2.2.2 Διόγκωση χωρίς δεσμεύσεις – Αλγόριθμος ‘Ανεξάρτητης Διόγκωσης’

Το σύστημα γενίκευσης περιλαμβάνει μία δεύτερη, εναλλακτική μέθοδο επεξεργασίας των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων, η οποία υλοποιείται μέσω του αλγορίθμου ‘Ανεξάρτητης Διόγκωσης’. Ο αλγόριθμος είναι σχεδιασμένος για την αντιμετώπιση του γεωμετρικού προβλήματος που περιγράφεται στο κεφάλαιο 4.1.2.3. Τη διόγκωση, δηλαδή, των εξομαλυσμένων  $\epsilon$ -τμημάτων, με τρόπο ώστε τα παράγωγα να ικανοποιούν ένα από τα κριτήρια που απαιτούνται για να χαρακτηρίζονται ως ‘ευανάγνωστα’, όταν ο δίσκος του Perkal διαμέτρου  $\epsilon$  εντοπίζεται στη θέση  $K$ . Ο ρόλος της μεθόδου είναι επικουρικός, καθώς χρησιμοποιείται σε περιπτώσεις που δε διασφαλίζονται οι προϋποθέσεις ευαναγνωσιμότητας των  $\epsilon$ -τμημάτων μέσω της αντίστοιχης, ‘υπό δεσμεύσεις’ διαδικασίας. Ωστόσο, στην εφαρμογή της ενσωματώνεται το σύνολο των απαραίτητων διεργασιών διόγκωσης, γεγονός που παρέχει τη δυνατότητα αυτοτελούς εκτέλεσής της από το χρήστη.

Η διόγκωση των  $\epsilon$ -τμημάτων περατώνεται με τη χρήση του αλγορίθμου ‘Balloon’ (Lecordix et al. 1997) και του ομοπαράλληλου μετασχηματισμού, οι οποίοι έχουν προσαρμοσθεί και ενταχθεί στον αλγόριθμο ‘Ανεξάρτητης Διόγκωσης’. Ο γεωμετρικός μετασχηματισμός υλοποιείται με την εκτέλεση διαδοχικά αυτών, σε μία, ενιαία διαδικασία. Συγκεκριμένα, το εξομαλυσμένο  $\epsilon$ -τμήμα διογκώνεται μέσω του αλγορίθμου ‘Balloon’ και στη συνέχεια, το  $\epsilon$ -τμήμα που προκύπτει υπόκειται σε ομοπαράλληλο μετασχηματισμό, ώστε να διαμορφωθεί η γενικευμένη εκδοχή του. Η λειτουργία του αλγορίθμου είναι ημιαυτόματη. Ο χρήστης θέτει την τιμή του συντελεστή ‘Inflating’ του πρώτου (βλ. κεφ. 3.5), ενώ οι υπόλοιπες παράμετροι εφαρμογής προσδιορίζονται μέσω εσωτερικών διεργασιών από το σύστημα. Οι τελευταίες προηγούνται της διαδικασίας διόγκωσης και αφορούν στην ανάλυση της γεωμετρίας στη ‘θέση  $K$ ’. Της γεωμετρίας, δηλαδή, που διαμορφώνεται μεταξύ του δίσκου του Perkal του οποίου το κέντρο εντοπίζεται στο σημείο  $K$  και των κορυφών της εξομαλυσμένης και της αρχικής εκδοχής του  $\epsilon$ -τμήματος ( $K_{\Sigma\epsilon}$  και  $K_{\Sigma\alpha}$ , αντίστοιχα). Τα τρία σημεία έχουν ανευρεθεί σε προηγούμενα στάδια επεξεργασίας (βλ. κεφ. 4.1.1 και 4.1.2.2) και εισάγονται ως δεδομένα στον αλγόριθμο. Με κριτήριο τη σχετική θέση αυτών, τα  $\epsilon$ -τμήματα χαρακτηρίζονται και εντάσσονται στην κατηγορία: (1) όταν οι κορυφές  $K_{\Sigma\epsilon}$  και  $K_{\Sigma\alpha}$  εντοπίζονται εκτός του δίσκου (Σχήμα 4.9α), (2) όταν η κορυφή  $K_{\Sigma\epsilon}$  εντοπίζεται πάνω στο δίσκο και η κορυφή  $K_{\Sigma\alpha}$  εκτός αυτού (Σχήμα 4.9β), (3) όταν η κορυφή  $K_{\Sigma\epsilon}$  εντοπίζεται εντός του δίσκου και η κορυφή  $K_{\Sigma\alpha}$  εκτός ή πάνω σε αυτόν (Σχήμα 4.9γ) και (4) όταν οι κορυφές  $K_{\Sigma\epsilon}$  και  $K_{\Sigma\alpha}$  εντοπίζονται εντός του δίσκου (Σχήμα 4.9δ).

Μία πρώτη διεργασία που πραγματοποιείται με βάση τη γεωμετρία στη ‘θέση  $K$ ’ είναι ο υπολογισμός του βαθμού ‘Δεσμευμένης Διόγκωσης’. Λαμβάνοντας υπόψη πως η μέθοδος ‘Ανεξάρτητης Διόγκωσης’ επιλέγεται σε περιπτώσεις που το παράγωγο από την εκτέλεση του αλγορίθμου ‘Δεσμευμένης Διόγκωσης’  $\epsilon$ -τμήμα δεν ικανοποιεί τα κριτήρια ευαναγνωσιμότητας, αυτός αποτελεί κατώφλι κατά την εφαρμογή της. Είναι λοιπόν χρήσιμο ο χρήστης, ο οποίος ‘συμμετέχει’ στον προσδιορισμό του μεγέθους διόγκωσης του  $\epsilon$ -τμήματος, να γνωρίζει εκ των προτέρων την τιμή του. Για το λόγο αυτό, με δεδομένη τη γεωμετρία στη ‘θέση  $K$ ’ (η

οποία είναι συγκεκριμένη για το εκάστοτε  $\epsilon$ -τμήμα) εκτελείται εικονικά το στάδιο προεπεξεργασίας του αλγορίθμου ‘Δεσμευμένης Διόγκωσης’ (βλ. κεφ. 4.1.2.3.1). Όταν το  $\epsilon$ -τμήμα ανήκει στην κατηγορία (1), (2) ή (3), ο ομώνυμος βαθμός διόγκωσης είναι ίσος με την απόσταση ( $K\Sigma_E K\Sigma_A$ ). Στην περίπτωση που ανήκει στην κατηγορία (4), αρχικά εντοπίζεται το σημείο  $T$  όπου τέμνονται η κάθετος στο  $\epsilon$ -τμήμα που διέρχεται από την κορυφή  $K\Sigma_E$  και ο δίσκος του Perkal και στη συνέχεια προσδιορίζεται ο βαθμός βάσει της σχέσης ( $K\Sigma_E T$ ) (Σχήμα 4.9). Η τιμή που υπολογίζεται εξάγεται σαν πληροφορία μέσω του συστήματος.



Σχήμα 4.9: Περιπτώσεις εφαρμογής αλγορίθμου ‘Ανεξάρτητης Διόγκωσης’.

Κατά την εκτέλεση του αλγορίθμου ‘Ανεξάρτητης Διόγκωσης’, ο χρήστης, λαμβάνοντας ως ελάχιστο το βαθμό ‘Δεσμευμένης Διόγκωσης’, καλείται να θέσει



την τιμή του συντελεστή *'Inflating'*. Βάσει αυτής υλοποιείται ο μετασχηματισμός του εξομαλυσμένου  $\epsilon$ -τμήματος μέσω του αλγορίθμου *'Balloon'*. Έτσι, η κορυφή του ΚΣ<sub>ε</sub> μετατοπίζεται σε απόσταση *'Inflating'*, ενώ τα υπόλοιπα σημεία που το δομούν σε μικρότερες, ανάλογες αυτής αποστάσεις, σύμφωνα με τη φιλοσοφία του αλγορίθμου (βλ. κεφ. 3.5). Όπως φαίνεται στα παραδείγματα του Σχήματος 4.9, στην προκύπτουσα εκδοχή του η κορυφή ΚΣ<sub>ε</sub> εντοπίζεται στη θέση ΙΕ.

Εξετάζοντας τα μη ευανάγνωστα  $\epsilon$ -τμήματα που διαμορφώνονται από την εφαρμογή του αλγορίθμου *'Δεσμευμένης Διόγκωσης'* (γραμμή πράσινης απόχρωσης στο Σχήμα 4.9), παρατηρείται πως η κορυφή τους, ως μεμονωμένο σημείο (ΚΣ<sub>Α</sub> ή *T*), δεν ανακόπτει την κύλιση του δίσκου του Perkal, αφού εντοπίζεται εκτός ή πάνω σε αυτόν. Το πρόβλημα ευαναγνωσιμότητας δημιουργείται από ενδιάμεσα σημεία τους. Στις περισσότερες περιπτώσεις (χωρίς αυτό να συνιστά κανόνα), σημεία που εντοπίζονται σε μεσαίες θέσεις μεταξύ του άκρου και της κορυφής, είναι εκείνα που επιδέχονται τη μεγαλύτερη επιπρόσθετη μετάθεση. Αν υποθεθεί πως το πρόβλημα αντιμετωπίζεται με τη χρήση αποκλειστικά του αλγορίθμου *'Balloon'*, ο χρήστης πρέπει να επιλέξει εκείνη την τιμή του συντελεστή *'Inflating'*, με την οποία το σημείο του εξομαλυσμένου  $\epsilon$ -τμήματος που απαιτείται να μετατοπισθεί κατά τη μέγιστη απόσταση, να εντοπίζεται στην κατάλληλη θέση μετά την εφαρμογή. Είναι προφανές πως για την επίτευξη αυτού, και με δεδομένο τον τρόπο λειτουργίας του αλγορίθμου (το διάστημα μετάθεσης των σημείων ελαττώνεται προσεγγίζοντας τα άκρα), η κορυφή ΚΣ<sub>ε</sub> μετατοπίζεται πέραν του αναγκαίου, σε θέση δηλαδή εξωτερικά του σημείου ΚΣ<sub>Α</sub> ή *T*. Το ίδιο συμβαίνει και με γειτονικά της σημεία. Όπως φαίνεται στα παραδείγματα του Σχήματος 4.9 (γραμμή μπλε απόχρωσης), τα  $\epsilon$ -τμήματα μετασχηματίζονται ώστε να διασφαλίζεται η κύλιση του δίσκου του Perkal κατά μήκος τους, όμως εμφανίζουν μεγαλύτερη της απαιτούμενης παραμόρφωση, κυρίως σε περιοχές γύρω από την κορυφή. Το γεγονός είναι πιθανό να υποβαθμίζει αισθητικά ή να καθιστά χαρτογραφικά μη αποδεκτό το προϊόν. Το  $\epsilon$ -τμήμα, δηλαδή, να διογκώνεται σε τέτοιο βαθμό που να έρχεται σε σύμπτωση με παρακείμενα τμήματα ή η έκτασή του να είναι δυσανάλογη σε σχέση με την αρχική του στο σύνολο της γραμμής. Σε κάθε περίπτωση ανακύπτει θέμα οπτικής ποιότητας του αποτελέσματος. Επιπρόσθετα, και λαμβάνοντας υπόψη πως η μεθοδολογία απευθύνεται στην επεξεργασία γραμμών που απεικονίζουν φυσικά φαινόμενα, κάτι τέτοιο μπορεί να επηρεάσει την ακρίβεια του τελικού προϊόντος. Για παράδειγμα, αν το  $\epsilon$ -τμήμα αποτελεί άκρο μιας χερσονήσου, η διόγκωσή του σε βαθμό που να απεικονίζεται μέσα στη θάλασσα θα μπορούσε να μειώσει την αξιοπιστία του αποτελέσματος της γενίκευσης ενός ναυτικού χάρτη.

Για την αποφυγή τέτοιων ενδεχόμενων λαμβάνει χώρα ο ομοπαράλληλος μετασχηματισμός του  $\epsilon$ -τμήματος που προκύπτει από την εφαρμογή του αλγορίθμου *'Balloon'*. Η διαδικασία υλοποιείται υπό τον όρο της διατήρησης σταθερών των δύο άκρων Α' και Π' και της μεταφοράς της κορυφής από τη θέση ΙΕ στη θέση όπου θα εντοπίζονταν αν εκτελούνταν ο αλγόριθμος *'Δεσμευμένης Διόγκωσης'*, δηλαδή:

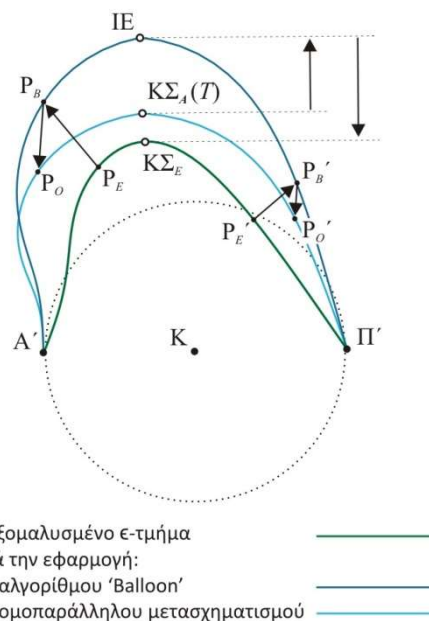
α) στο σημείο ΚΣ<sub>Α</sub>, όταν το  $\epsilon$ -τμήμα ανήκει στην κατηγορία (1), (2) ή (3) (Σχήμα 4.9α, 4.9β και 4.9γ, αντίστοιχα) ή

β) στο σημείο  $T$ , το οποίο έχει εντοπισθεί κατά το προηγούμενο στάδιο επεξεργασίας, όταν αυτό είναι ενταγμένο στην κατηγορία (4) (Σχήμα 4.9δ).

Όπως φαίνεται στα παραδείγματα, σε κάθε περίπτωση το σημείο  $K_{\Sigma_A}$  ή  $T$  εντοπίζεται εγγύτερα στο κέντρο ( $K$ ) του δίσκου του Perkal, χωρίς όμως να ανακόπτει την κύλιση του τελευταίου. Στη συνήθη μορφή των  $\epsilon$ -τμημάτων, με τις παραπάνω δεσμεύσεις η κορυφή μετατίθεται κατά τη μέγιστη απόσταση, στη διεύθυνση της ευθείας  $IEK_{\Sigma_A}$  (ή  $IE_T$ ). Τα υπόλοιπα σημεία, επίσης μετατοπίζονται προς τον άξονα που διέρχεται από τα άκρα  $A'$  και  $\Pi'$ , σε διευθύνσεις που με μικρές αποκλίσεις είναι παράλληλες με εκείνη της κορυφής και σε διαστήματα που ελαττώνονται καθώς εντοπίζονται πλησιέστερα στα άκρα (βλ. Σχήμα 4.10).

Εξετάζοντας τις νέες θέσεις των σημείων, παρατηρείται πως η κορυφή ( $K_{\Sigma_A}$  ή  $T$ ) απέχει τη μεγαλύτερη απόσταση από τον άξονα  $A'\Pi'$ , γεγονός που φανερώνει πως μεταξύ των δύο αναπτύσσεται το παράγωγο  $\epsilon$ -τμήμα. Επιτυγχάνεται, έτσι, η επίλυση του προβλήματος πλεονάζουσας παραμόρφωσης, αφού οι περιοχές που έχουν μετατεθεί πέραν του απαραίτητου (θέση  $K_{\Sigma_A}$  ή  $T$ ) μέσω του αλγορίθμου 'Balloon' διαμορφώνονται εσωτερικά αυτής. Η 'συμπίεση', όμως, του  $\epsilon$ -τμήματος προς τον άξονα  $A'\Pi'$  είναι πιθανό να δημιουργήσει εκ νέου πρόβλημα στην κύλιση του δίσκου του Perkal. Μέρος των σημείων που το δοθούν, δηλαδή, θα έπρεπε να εντοπίζεται σε πιο απομακρυσμένες από το κέντρο του τελευταίου θέσεις. Ένα τέτοιο ενδεχόμενο αντιμετωπίζεται με τον ορισμό μεγαλύτερης τιμής του συντελεστή 'Inflating'. Αυξάνοντας το μέγεθος διόγκωσης του εξομαλυσμένου  $\epsilon$ -τμήματος μέσω του αλγορίθμου 'Balloon', δημιουργούνται οι προϋποθέσεις επαναφοράς του συνόλου των σημείων σε θέσεις που δεν ανακόπτουν την κύλιση του δίσκου, κατά την υλοποίηση του ομοπαράλληλου μετασχηματισμού. Δεδομένου πως οι δύο μετασχηματισμοί λαμβάνουν χώρα αυτόματα, σε μία διαδικασία, ο χρήστης οφείλει να λάβει υπόψη εξ αρχής το γεγονός και να επιλέξει την ελάχιστη αναγκαία τιμή του συντελεστή 'Inflating', ώστε να επιτυγχάνονται οι συνθήκες ευαναγνωσιμότητας με τη δυνατόν μικρότερη μεταβολή της μορφής και της γεωμετρίας του  $\epsilon$ -τμήματος.

Στο Σχήμα 4.10 παρουσιάζεται ένα παράδειγμα του τρόπου που μετασχηματίζεται το εξομαλυσμένο  $\epsilon$ -τμήμα κατά την εκτέλεση του αλγορίθμου 'Ανεξάρτητης Διόγκωσης', με τη βοήθεια της αναπαράστασης της πορείας μετάθεσης δύο σημείων του  $P$  και  $P'$ . Χαρακτηριστικό της λειτουργίας τόσο του αλγορίθμου 'Balloon', όσο και του ομοπαράλληλου μετασχηματισμού είναι πως η απόσταση μετάθεσης του εκάστοτε σημείου είναι ανάλογη μίας μέγιστης, η οποία εμφανίζεται



**Σχήμα 4.10:** Παράδειγμα εφαρμογής του αλγορίθμου 'Balloon' και του ομοπαράλληλου μετασχηματισμού κατά την εκτέλεση του αλγορίθμου 'Ανεξάρτητης Διόγκωσης'.



στην κορυφή. Όπως φαίνεται στο σχήμα όμως, σε κάθε περίπτωση ισχύει:  $(K\Sigma_{EIE}) > (IEK\Sigma_A)$  (ή  $(K\Sigma_{EIE}) > (IE T)$ ) και κατά συνέπεια:  $(P_{EPB}) > (P_{BPo})$  για το τυχαίο σημείο P. Είναι προφανές λοιπόν, πως η διόγκωση του  $\epsilon$ -τμήματος μέσω του αλγορίθμου 'Balloon' είναι πάντοτε μεγαλύτερη από τη 'συμπίεση' που υφίσταται από τον ομοπαράλληλο μετασχηματισμό. Επιπρόσθετα, και στις δύο περιπτώσεις, η απόσταση μετάθεσης των σημείων ελαττώνεται καθώς αυτά εντοπίζονται πλησιέστερα στα άκρα ( $(P_{EPB}) > (P_{E'P'B'})$  και  $(P_{BPo}) > (P_{B'Po'})$ ), με το ρυθμό όμως της ελάττωσης να είναι μεγαλύτερος κατά την εφαρμογή του ομοπαράλληλου μετασχηματισμού ( $\Delta(P_{Bo}) > \Delta(P_{EB})$ ). Τα παραπάνω, σε συνδυασμό με το γεγονός πως, μέσω του τελευταίου, τα σημεία δεν μετατοπίζονται προς την αρχική τους θέση, αλλά σε διεύθυνση που με μικρές αποκλίσεις είναι κοινή και παράλληλη με εκείνη της κορυφής (βλ. Σχήμα 4.10), έχει ως αποτέλεσμα ο ομοπαράλληλος μετασχηματισμός να επιφέρει μία σταδιακή 'εκτόνωση' της αρχικής διόγκωσης προς τα πλάγια του  $\epsilon$ -τμήματος. Η διόγκωση, δηλαδή, περιορίζεται περισσότερο σε περιοχές που είναι πλεονάζουσα (γύρω από την κορυφή) και λιγότερο σε περιοχές (συνήθως στο μέσο του  $\epsilon$ -τμήματος), όπου απαιτείται να είναι μέγιστη ώστε να μην ανακόπτεται η κύλιση του δίσκου του Perkal. Ουσιαστικά, εφαρμόζοντας διαδοχικά τους δύο μετασχηματισμούς με τους κανόνες που αναφέρονται, επιτυγχάνεται κάθε θέση του εξομαλυσμένου  $\epsilon$ -τμήματος να διογκώνεται σε βαθμό που προσεγγίζει τον ελάχιστο απαραίτητο με τον οποίο διασφαλίζονται οι συνθήκες ευαναγνωσιμότητας, μειώνοντας έτσι την παραμόρφωση του τελικού προϊόντος.

Ο κώδικας υλοποίησης του αλγορίθμου επεξεργασίας των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων δημιουργήθηκε σε γλώσσα προγραμματισμού *Python* και παρουσιάζεται στο Παράρτημα Π4.1.2.

#### 4.2 Γενίκευση αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων (Κατηγορία Β)

Τα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα έχουν τη μορφή μιας επαλληλίας δύο ή περισσότερων κυρτώσεων και κοιλώσεων κατά μήκος της γραμμής. Θεωρητικά δημιουργούνται όταν η κύλιση του δίσκου του Perkal διαμέτρου  $\epsilon$  διακόπτεται μεταξύ διαδοχικών θέσεων, εσωτερικά και εξωτερικά της γραμμής. Βάσει των κανόνων οπτικής αντίληψης που έχουν ορισθεί στο κεφάλαιο 2, για την ευκρινή απόδοσή τους σε χάρτες μικρότερης κλίμακας απαιτείται ο μετασχηματισμός τους, ώστε ο δίσκος του Perkal να κυλά εκατέρωθεν της γραμμής. Η επίτευξη αυτού αποτελεί από μόνη της μια σύνθετη διαδικασία. Επιπρόσθετα, η δομή και η γεωμετρία των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων είναι συνήθως αρκετά πολύπλοκη, γεγονός που καθιστά τη ανάλυση και τη διαχείρισή τους ιδιαίτερα δύσκολη.

Για το λόγο αυτό, η επεξεργασία τους πραγματοποιείται σε δύο στάδια. Στο πρώτο λαμβάνει χώρα μια περαιτέρω κατάτμηση των  $\epsilon$ -τμημάτων. Ως τμήματα τίθενται οι καμπές (bends) όπως προκύπτουν από την εφαρμογή της μεθόδου Wang και Müller (1988). Οι καμπές αποτελούν τις θεμελιώδεις οντότητες που δομούν το  $\epsilon$ -τμήμα και

από την επεξεργασία αυτών εξάγονται οι παράγωγες γραμμές. Η διαδικασία περιλαμβάνει την εξομάλυνση των  $\epsilon$ -τμημάτων και τον εντοπισμό κάθε φορά των καμπών. Κατ' αυτό τον τρόπο, και βάσει κριτηρίων που βασίζονται στις αρχές οπτικής αντίληψης, επιλέγονται οι κύριες, ανάλογα με το βαθμό γενίκευσης και την παράγωγη κλίμακα. Στην πραγματικότητα, το αμφίπλευρο  $\epsilon$ -μη-κυρτό τμήμα αποδομείται στις κύριες κυρτώσεις και κοιλώσεις. Στο δεύτερο στάδιο επεξεργασίας, κάθε κύρτωση και κοίλωση υπόκειται ανεξάρτητα σε ένα μετασχηματισμό διόγκωσης, ο οποίος βασίζεται στον αντίστοιχο των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων και τέλος επανασυνδέονται, ώστε να προκύψει η γενικευμένη εκδοχή του  $\epsilon$ -τμήματος.

#### **4.2.1. Εντοπισμός καμπών και απλοποίηση των $\epsilon$ -μη-κυρτών τμημάτων**

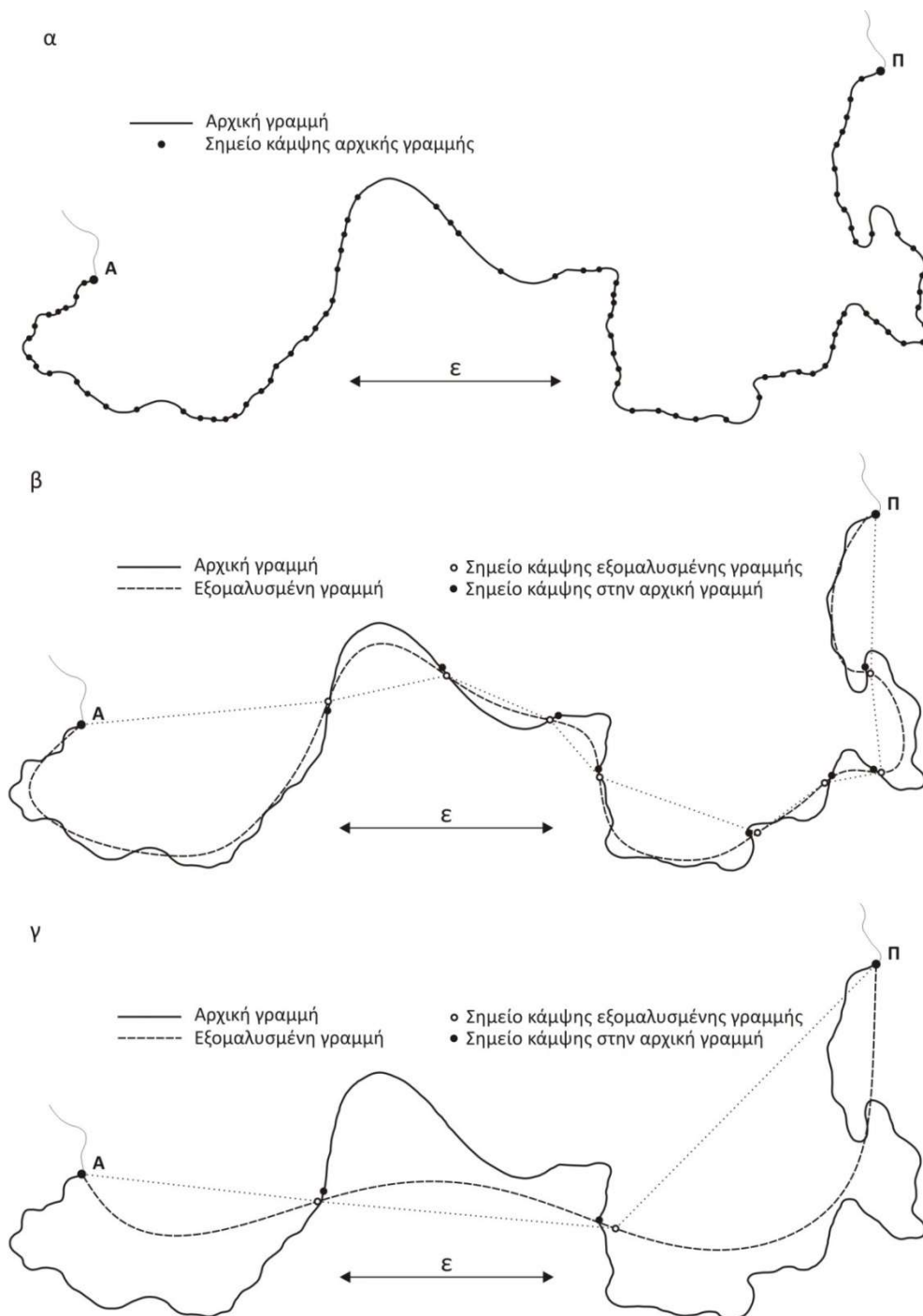
##### **4.2.1.1. Αρχή λειτουργίας**

Εξετάζοντας τον τρόπο δημιουργίας των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων (βλ. κεφ. 2.3.3, 2.3.4) συμπεραίνεται πως το σχήμα τους είναι αρκετά πολύπλοκο, αφού διαμορφώνονται από μια σειρά (κατ' ελάχιστο δύο) διαδοχικών κυρτώσεων και κοιλώσεων. Το γεγονός καθιστά την ανάλυση και την επεξεργασία τους ως ενιαία οντότητα ιδιαίτερα δύσκολη. Για το λόγο αυτό, προτείνεται η εκ νέου κατάτμησή τους. Ως τμήματα ορίζονται οι καμπές (bends), όπως προκύπτουν από τη μέθοδο που έχει προσαρμοσθεί και ενσωματωθεί στο μοντέλο γενίκευσης και βασίζεται στην έρευνα των Wang και Müller (1998) (βλ. κεφ. 3.3). Οι καμπές αποτελούν τις θεμελιώδεις οντότητες πάνω στις οποίες στηρίζεται η επεξεργασία των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων.

Τα  $\epsilon$ -τμήματα, στην αρχική τους μορφή, αποδίδονται με λεπτομέρεια ανάλογη της κλίμακας. Εφαρμόζοντας τη μέθοδο των Wang και Müller, αυτή 'μετατρέπεται' σε ένα πλήθος εντοπιζόμενων καμπών. Όπως φαίνεται στο Σχήμα 4.11α, αυτές αντιστοιχούν σε διαδοχικές κυρτώσεις και κοιλώσεις της γραμμής, η πλειοψηφία των οποίων, είτε αποτελούν θόρυβο (μικρολεπτομέρεια), είτε δεν είναι ευδιάκριτες στην παράγωγη κλίμακα βάσει των κανόνων ευαναγνωσιμότητας. Ένα ζητούμενο λοιπόν, είναι ο μετασχηματισμός του αρχικού  $\epsilon$ -τμήματος, ώστε να 'ανιχνεύονται' οι κύριες, κατάλληλες για την εκάστοτε εφαρμογή καμπές.

Ο μετασχηματισμός λαμβάνει χώρα με την εφαρμογή του τελεστή εξομάλυνσης *Gauss*, όπως έχει προσαρμοσθεί και ενσωματωθεί στο μοντέλο γενίκευσης (βλ. κεφ. 3.2.2). Στην πλειοψηφία των περιπτώσεων, ο τελεστής λειτουργεί ως φίλτρο καθαρισμού του 'θορύβου' που χαρακτηρίζει τις γραμμικές οντότητες στην πρωτογενή τους μορφή. Στο μοντέλο, χρησιμοποιείται ως 'εργαλείο' αφαίρεσης/συγχώνευσης των καμπών των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων. Η διαφορά έγκειται στον αναλογικά μεγαλύτερο βαθμό, με τον οποίο υλοποιείται. Η εφαρμογή στηρίζεται στην ιδέα, πως η λεπτομέρεια ενός  $\epsilon$ -τμήματος, η οποία είναι άμεσα συνδεδεμένη με την κλίμακα, εμμέσως ποσοτικοποιείται με τη δημιουργία των επιμέρους καμπών κατά μήκος του. Εφαρμόζοντας, λοιπόν, έναν τελεστή

εξομάλυνσης (έναν τελεστή, δηλαδή, αφαίρεσης λεπτομέρειας) σε κατάλληλο βαθμό, είναι δυνατό να αφαιρούνται οι μη απαραίτητες και να λαμβάνονται οι σημαντικές για την απόδοση του  $\epsilon$ -τμήματος στην παράγωγη κλίμακα, καμπές. Υπό αυτή την έννοια, η διαδικασία θα μπορούσε να χαρακτηριστεί και ως απλοποίηση των  $\epsilon$ -τμημάτων. Με τον τρόπο αυτό, επιτυγχάνεται η απαλοιφή/διατήρηση καμπών βάσει ενός αντικειμενικού, κοινού για όλο το  $\epsilon$ -τμήμα κριτηρίου, περιορίζοντας την παρέμβαση του χρήστη στην επιλογή του βαθμού εξομάλυνσης.



Σχήμα 4.11: Παράδειγμα καμπών που ανευρίσκονται στο αρχικό και σε εξομαλυσμένες εκδοχές αμφίπλευρου  $\epsilon$ -μη-κυρτού τμήματος.

Η απλοποίηση του εκάστοτε  $\epsilon$ -τμήματος λαμβάνει χώρα εφαρμόζοντας δοκιμαστικά τον τελεστή *Gauss*, ώστε να επιλεγεί η κατάλληλη, για τις διεργασίες που ακολουθούν, εξομαλυσμένη εκδοχή του. Το εύρος τιμών της μεταβλητής  $\sigma$  εξαρτάται από τον αριθμό των σημείων που δομούν το  $\epsilon$ -τμήμα (βλ. κεφ. 3.2.2), υπολογίζεται και παρέχεται ως πληροφορία στον χρήστη. Σε κάθε εξομαλυσμένο  $\epsilon$ -τμήμα που προκύπτει εντοπίζονται οι καμπές μέσω της μεθόδου Wang και Müller. Συγχρόνως, υπολογίζονται οι ποσοτικές ιδιότητες των καμπών (μήκος γραμμής βάσης, εμβαδόν, βάθος, καμπυλότητα, αριθμός σημείων (βλ. κεφ. 3.3), οι οποίες βοηθούν στην αξιολόγηση του αποτελέσματος εξομάλυνσης. Επιπλέον, εντοπίζεται η κορυφή τους μέσω της εφαρμογής του δείκτη αναλογίας μήκους ALR και τα σημεία κάμψης (inflection points). Όπως αναφέρεται στο κεφάλαιο 1.1, οι θέσεις αυτές θεωρούνται χαρακτηριστικές, αφού εμπεριέχουν και παρέχουν στον παρατηρητή μεγάλη ποσότητα πληροφορίας και ως εκ τούτου, παίζουν σημαντικό ρόλο στη διαμόρφωση της γενικευμένης γραμμής. Για το λόγο αυτό καταγράφονται, αντιστοιχίζονται με τα σημεία στο αρχικό  $\epsilon$ -τμήμα και καταχωρίζονται ως δεδομένα στο μοντέλο, ώστε να χρησιμοποιηθούν στις μετέπειτα διεργασίες γενίκευσης.

Όπως αναφέρεται παραπάνω, αν εφαρμοσθεί η μέθοδος των Wang και Müller στο αρχικό  $\epsilon$ -τμήμα, αυτό κατατέμενεται σε ένα πλήθος καμπών, οι οποίες βάσει των ποσοτικών τους ιδιοτήτων, δεν είναι ευδιάκριτες στην παράγωγη κλίμακα (Σχήμα 4.11α). Εξομαλύνοντας το  $\epsilon$ -τμήμα η αρχική λεπτομέρεια (άρα και οι αντίστοιχες καμπές) αφαιρούνται, με αποτέλεσμα να εντοπίζονται λιγότερες, αλλά μεγαλύτερες καμπές (Σχήμα 4.11β). Εφαρμόζοντας τον τελεστή *Gauss* με τη μέγιστη επιτρεπτή τιμή του ( $\sigma_{max} = \frac{\nu - 1}{8}$ , όπου  $\nu$  ο αριθμός των σημείων του  $\epsilon$ -τμήματος) είναι δυνατό

να εντοπισθούν οι ελάχιστες καμπές κατά μήκος του  $\epsilon$ -τμήματος (Σχήμα 4.11γ). Η ιδιότητα του τελεστή εξομάλυνσης *Gauss* να επιφέρει συρρίκνωση όταν εφαρμόζεται σε κλειστές ή καμπύλες γραμμικές οντότητες (βλ. κεφ. 3.2.2), ευνοεί τη χρησιμοποίησή του στη διαδικασία εντοπισμού των κύριων καμπών. Και τούτο διότι στην πλειοψηφία των περιπτώσεων, τα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα αποτελούν γραμμές μεγάλης καμπυλότητας και συγχρόνως, επιλέγεται η εξομάλυνση τους σε μεγάλο βαθμό. Έτσι, παρά τους περιορισμούς που μπορεί να επιφέρει η εφαρμογή του τελεστή με συγκεκριμένες τιμές της παραμέτρου  $\sigma$  ή πιθανές ιδιαιτερότητες στη γεωμετρία του  $\epsilon$ -τμήματος, η διαδικασία κρίνεται ιδιαίτερα αποτελεσματική.

#### **4.2.1.2. Επιλογή προς διατήρηση/αφαίρεση καμπών**

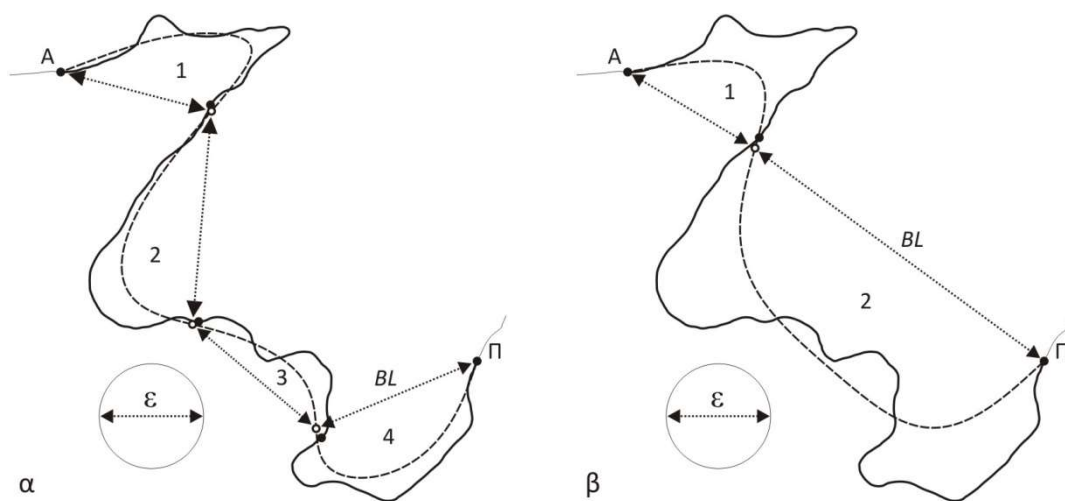
Η ποιότητα της γενίκευσης των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων εξαρτάται κατά πολύ από το αποτέλεσμα της εξομάλυνσης. Ανάλογα με τη ζητούμενη αλλαγή κλίμακας και τη γεωμετρία της γραμμής, ο χρήστης καλείται να επιλέξει τον κατάλληλο βαθμό εξομάλυνσης, από τον οποίο καθορίζονται οι παράμετροι 'πόσες' και 'ποιές' καμπές επιλέγονται για περαιτέρω επεξεργασία. Σε ένα δεύτερο επίπεδο, όταν οι καμπές έχουν επιλεγεί, μικρές διαφοροποιήσεις στο βαθμό εξομάλυνσης

επιφέρουν αλλαγές της γεωμετρίας στη 'θέση Κ' στην οποία στηρίζεται η διαδικασία διόγκωσης που ακολουθεί.

Ένα βασικό κριτήριο πάνω στο οποίο στηρίζεται η επιλογή για διατήρηση ή αφαίρεση κάθε καμπής είναι η αναλογία του μήκους της γραμμής βάσης της ( $L_{BL}$ ), ως προς τη διάμετρο εφαρμογής  $\epsilon$  του θεωρητικού δίσκου του Perkal. Βάσει των κανόνων οπτικής αντίληψης που έχουν ορισθεί (βλ. κεφ. 2), μια καμπή είναι δυνατό να αποδοθεί ευκρινώς στην παράγωγη κλίμακα όταν ο δίσκος μπορεί να εισέλθει και να κυλήσει κατά μήκος της. Για την εξασφάλιση του πρώτου λοιπόν, επιλέγεται να διατηρούνται οι καμπές με γραμμή βάσης μεγαλύτερη ή ίση της διαμέτρου  $\epsilon$  ( $L_{BL} \geq \epsilon$ ). Οι υπόλοιπες ( $L_{BL} < \epsilon$ ) είναι υποψήφιες προς αφαίρεση. Πάνω σε αυτή τη λογική, ο χρήστης εφαρμόζει τον τελεστή εξομάλυνσης *Gauss* με σταδιακά αυξανόμενες τιμές της μεταβλητής  $\sigma$ . Σε κάθε μία από τις εξομαλυσμένες εκδοχές του  $\epsilon$ -τμήματος εντοπίζονται οι καμπές και ελέγχεται η αναλογία  $\frac{L_{BL}}{\epsilon}$ .

Κατά τη διαδικασία είναι δυνατό:

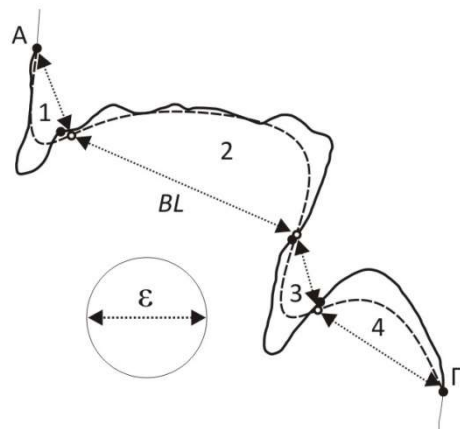
1. σε μία τιμή της μεταβλητής  $\sigma$  να δημιουργούνται καμπές, για το σύνολο των οποίων η γραμμή βάσης είναι μεγαλύτερη ή ίση της διαμέτρου  $\epsilon$  ( $L_{BL} \geq \epsilon$ ). Η τιμή αυτή τίθεται ως κατώφλι ( $\sigma_{kmin}$ ), αφού είναι η ελάχιστη με την οποία το  $\epsilon$ -τμήμα αποκόπτεται σε καμπές που πληρούν το κριτήριο επιλογής-διατήρησης και μπορούν να θεωρηθούν ως κύριες (Σχήμα 4.12α). Λαμβάνοντας ως βάση το κατώφλι  $\sigma_{kmin}$ , το  $\epsilon$ -τμήμα μπορεί να εξομαλυνθεί σε μεγαλύτερο βαθμό, μέχρι τη μέγιστη τιμή της μεταβλητής  $\sigma$  ( $\sigma_{max}$ ). Η περαιτέρω εξομάλυνση είναι δυνατό να επιφέρει: α) μικρομετατοπίσεις στα όρια των ήδη εντοπισμένων κύριων καμπών, διατηρώντας όμως τον αριθμό τους, β) επιπλέον αφαίρεση/συγχώνευση καμπών και δημιουργία νέων, μεγαλύτερων (Σχήμα 4.12β). Οι ελάχιστες τιμές της μεταβλητής  $\sigma$  με τις οποίες επιτυγχάνεται αυτό ( $\sigma_{kmin}$ ) καταγράφονται. Για όλες τις καμπές που εντοπίζονται εξακολουθεί να ισχύει η συνθήκη  $L_{BL} \geq \epsilon$ .



Σχήμα 4.12: Παράδειγμα καμπών που πληρούν το κριτήριο  $L_{BL} \geq \epsilon$ .

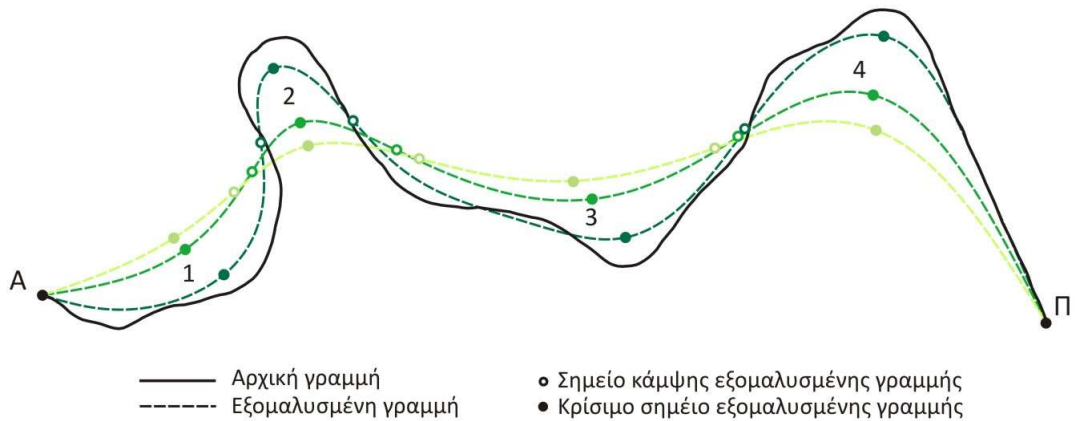
Στη δεύτερη περίπτωση, ο χρήστης καλείται να επιλέξει ποια εκδοχή κύριων καμπών θα χρησιμοποιηθούν στο επόμενο στάδιο επεξεργασίας. Η επιλογή πραγματοποιείται συνεκτιμώντας δύο παράγοντες. Ο πρώτος έχει να κάνει με το αν οι καμπές, μετά τη διαδικασία διόγκωσης, καλύπτουν τα κριτήρια ευαναγνωσιμότητας και συγχρόνως, το γενικευμένο  $\epsilon$ -τμήμα στο σύνολό του είναι αισθητικά και χαρτογραφικά αποδεκτό. Επιπρόσθετα, λαμβάνονται υπόψη δεσμεύσεις υποκειμενικού ενδιαφέροντος, οι οποίες συνδέονται με τα απεικονιζόμενα φαινόμενα και το σκοπό του χάρτη. Για παράδειγμα, ο χρήστης καλείται να αποφασίσει αν η αλληλουχία χερσόνησος-κόλπος-χερσόνησος που αποδίδεται μέσω των καμπών '2', '3' και '4' του Σχήματος 4.12α θα αποδοθεί στον παράγωγο χάρτη ή θα απεικονισθεί ως μία οντότητα (καμπή '2' στο Σχήμα 4.12β).

2. για ορισμένες ή για το σύνολο των καμπών που δημιουργούνται έχοντας εξομαλύνει το  $\epsilon$ -τμήμα στο μέγιστο βαθμό (τιμή  $\sigma_{max}$  της μεταβλητής  $\sigma$ ), να ισχύει  $L_{BL} < \epsilon$  (Σχήμα 4.13). Το γεγονός οφείλεται σε ένα συνδυασμό παραγόντων που έχουν να κάνουν με τη γεωμετρία του  $\epsilon$ -τμήματος, τον αριθμό των σημείων που το δομούν και τον τρόπο που το μετασχηματίζει ο τελεστής εξομάλυνσης *Gauss*. Στην περίπτωση αυτή επιλέγεται προς περαιτέρω επεξεργασία ο ελάχιστος αριθμός καμπών που δύναται να δημιουργηθεί. Η ελάχιστη τιμή της μεταβλητής  $\sigma$  με την οποία αυτό επιτυγχάνεται ( $\sigma_{min}$ ) καταγράφεται στο μοντέλο.



**Σχήμα 4.13:** Παράδειγμα καμπών που δεν πληρούν το κριτήριο  $BL \geq \epsilon$ .

Μετά την ολοκλήρωση του σταδίου επιλογής των κύριων καμπών ακολουθεί η τελική διαμόρφωσή τους. Η διαδικασία είναι εξίσου σημαντική και συνδέεται άμεσα με την επεξεργασία διόγκωσης που ακολουθεί. Όπως αναφέρεται παραπάνω, στο μοντέλο έχει καταγραφεί η ελάχιστη τιμή της μεταβλητής  $\sigma$  με την οποία έχει επιτευχθεί ο εντοπισμός των κύριων καμπών του εκάστοτε  $\epsilon$ -τμήματος. Με βάση αυτή δίνεται η δυνατότητα επιπλέον εξομάλυνσης του  $\epsilon$ -τμήματος, μέχρι τον μέγιστο εφικτό βαθμό, δηλαδή την τιμή  $\sigma_{max}$  ή την αμέσως προηγούμενη της τιμής  $\sigma_{kmin}$ , όταν έχει επιλεγεί ενδιάμεση εκδοχή κύριων καμπών της περίπτωσης 1β.



**Σχήμα 4.14:** Παράδειγμα εξομάλυνσης κύριων καμπών.

Η διαδικασία επιφέρει αλλαγές στη γεωμετρία των καμπών, χωρίς όμως να τις επηρεάζει ως τις χαρακτηριστικές οντότητες που περιγράφουν το αμφίπλευρο  $\epsilon$ -μη-κυρτό τμήμα. Όπως φαίνεται στο παράδειγμα του Σχήματος 4.14, αυξάνοντας τον βαθμό εξομάλυνσης, οι τέσσερις κύριες καμπές στις οποίες έχει αποδομηθεί το  $\epsilon$ -τμήμα εξακολουθούν να υφίστανται. Η μεταβολή έγκειται στη θέση των σημείων κάμψης και των κρίσιμων σημείων των καμπών και κατά συνέπεια, και των αντίστοιχων θέσεων στην αρχική εκδοχή του  $\epsilon$ -τμήματος. Ουσιαστικά, μεταβάλλεται το μήκος της γραμμής βάσης ( $L_{BL}$ ) και η γεωμετρία στη 'θέση Κ' (δηλαδή, η γεωμετρία που διαμορφώνεται μεταξύ των κορυφών  $K_{ΣΕ}$  και  $K_{ΣΑ}$  του εξομαλυσμένου και του αρχικού  $\epsilon$ -τμήματος αντίστοιχα και του δίσκου του Perkal στη θέση Κ, βλ. κεφ. 4.1.1 και 4.1.2.1) της εκάστοτε καμπής. Τα δύο αυτά γεωμετρικά χαρακτηριστικά παίζουν καθοριστικό ρόλο στο αποτέλεσμα της γενίκευσης του  $\epsilon$ -τμήματος, αφού επηρεάζουν τόσο τη δυνατότητα εισχώρησης του δίσκου του Perkal εντός της καμπής, όσο και τη μέθοδο και το βαθμό διόγκωσής της που ακολουθεί.

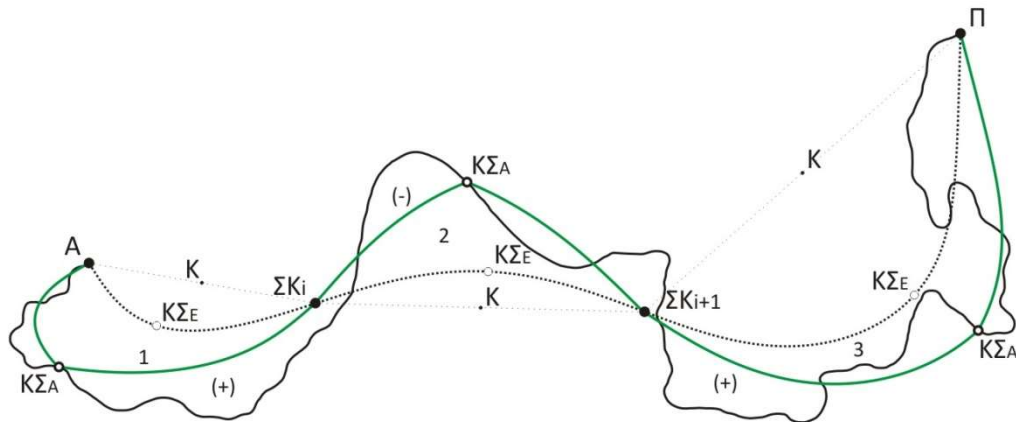
Η επιλογή της βέλτιστης μορφής των κύριων καμπών αποτελεί μια αρκετά σύνθετη διαδικασία για το χρήστη. Και τούτο διότι, ενώ η εξομάλυνση λαμβάνει χώρα στο σύνολο του  $\epsilon$ -τμήματος, τα χαρακτηριστικά κάθε καμπής αξιολογούνται μεμονωμένα, δεδομένου ότι ο μετασχηματισμός διόγκωσης πραγματοποιείται ανεξάρτητα σε κάθε μία. Έτσι, είναι πιθανό να αλλοιώνεται η αποδεκτή μορφή μίας καμπής, στην προσπάθεια να βελτιωθεί εκείνη μίας όμορης. Απαιτείται, λοιπόν, η επιλογή της κατάλληλης τιμής της μεταβλητής  $\sigma$ , με την οποία διασφαλίζεται το δυνατόν ορθότερο αποτέλεσμα για όλες τις κύριες καμπές. Η διαδικασία πραγματοποιείται συνδυαστικά με την επεξεργασία διόγκωσης, ώστε ο χρήστης να έχει την εποπτεία του τελικού προϊόντος.

#### 4.2.2 Διόγκωση καμπών

Μετά την περάτωση του σταδίου απλοποίησης, έχουν καθορισθεί οι κύριες καμπές μέσω των οποίων θα αποδοθούν τα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα στην



παράγωγη κλίμακα. Σε ένα τελευταίο στάδιο επεξεργασίας, τα  $\epsilon$ -τμήματα, όπως έχουν διαμορφωθεί μετά την εξομάλυνση, υπόκεινται στο μετασχηματισμό διόγκωσης, ώστε να επιτευχθούν οι συνθήκες ευαναγνωσιμότητάς τους.



Σχήμα 4.15: Παράδειγμα διόγκωσης κύριων καμπών.

Στο μοντέλο εισάγονται το αρχικό και η εξομαλυσμένη εκδοχή του  $\epsilon$ -τμήματος, η οποία είναι κατατετημημένη στις κύριες καμπές. Ο χρήστης τις απομονώνει βάσει του αύξοντα αριθμού τους, αφού η διεργασία διόγκωσης λαμβάνει χώρα ανεξάρτητα σε κάθε μία. Κατά το προηγούμενο στάδιο επεξεργασίας για την εκάστοτε καμπή έχουν καταγραφεί τα γεωμετρικά χαρακτηριστικά της, δηλαδή: α) η αρχή και το πέρας (A,  $\Sigma K_i$  για την πρώτη καμπή,  $\Sigma K_{i+1}, \Pi$  για την τελευταία και  $\Sigma K_i, \Sigma K_{i+1}$  για τις ενδιάμεσες, όπου A,  $\Pi$  τα άκρα του  $\epsilon$ -τμήματος και  $\Sigma K_i, \Sigma K_{i+1}$  τα σημεία κάμψης), β) η κορυφή  $K\Sigma_E$ , η οποία έχει εντοπισθεί μέσω της εφαρμογής του δείκτη αναλογίας μήκους ALR, γ) το πρόσημο της καμπυλότητας και δ) το κέντρο K του θεωρητικού δίσκου του Perkal, το οποίο ορίζεται ως το μέσο της ευθείας που ενώνει την αρχή και το πέρας. Επιπλέον, η κορυφή έχει αντιστοιχισθεί με εκείνη στην αρχική καμπή ( $K\Sigma_A$ ).

Όπως φαίνεται στο Σχήμα 4.15, κάθε καμπή αποτελεί μία ανεξάρτητη κύρτωση ή κοίλωση. Λαμβάνοντας, λοιπόν, ως δεδομένα τα παραπάνω χαρακτηριστικά, υλοποιείται η διεργασία διόγκωσης αντίστοιχη με αυτή που εφαρμόζεται στα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα. Ανάλογα με τη γεωμετρία της καμπής και το επίπεδο αλλαγής κλίμακας, ο χρήστης εφαρμόζει έναν εκ των αλγορίθμων 'Δεσμευμένης Διόγκωσης' ή 'Ανεξάρτητης Διόγκωσης', όπως περιγράφονται στα κεφάλαια 4.1.2.2.1 και 4.1.2.2.2, αντίστοιχα. Κατά την επεξεργασία των καμπών, τόσο μέσω του αλγορίθμου διόγκωσης *Balloon* (Lecordix et al. 1997), όσο και του ομοπαράλληλου μετασχηματισμού, οι οποίοι εμπεριέχονται στους παραπάνω αλγορίθμους, τα άκρα τους δεν μετατίθενται. Έτσι, μετά την περάτωση της διαδικασίας διόγκωσης για το σύνολο των καμπών, αυτές επανασυνδέονται, ώστε να προκύψει η γενικευμένη εκδοχή του αμφίπλευρου  $\epsilon$ -μη-κυρτού τμήματος (γραμμή πράσινης απόχρωσης στο Σχήμα 4.15).

Ο κώδικας υλοποίησης του αλγορίθμου επεξεργασίας των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων δημιουργήθηκε σε γλώσσα προγραμματισμού *Python* και παρουσιάζεται στο Παράρτημα Π4.2.



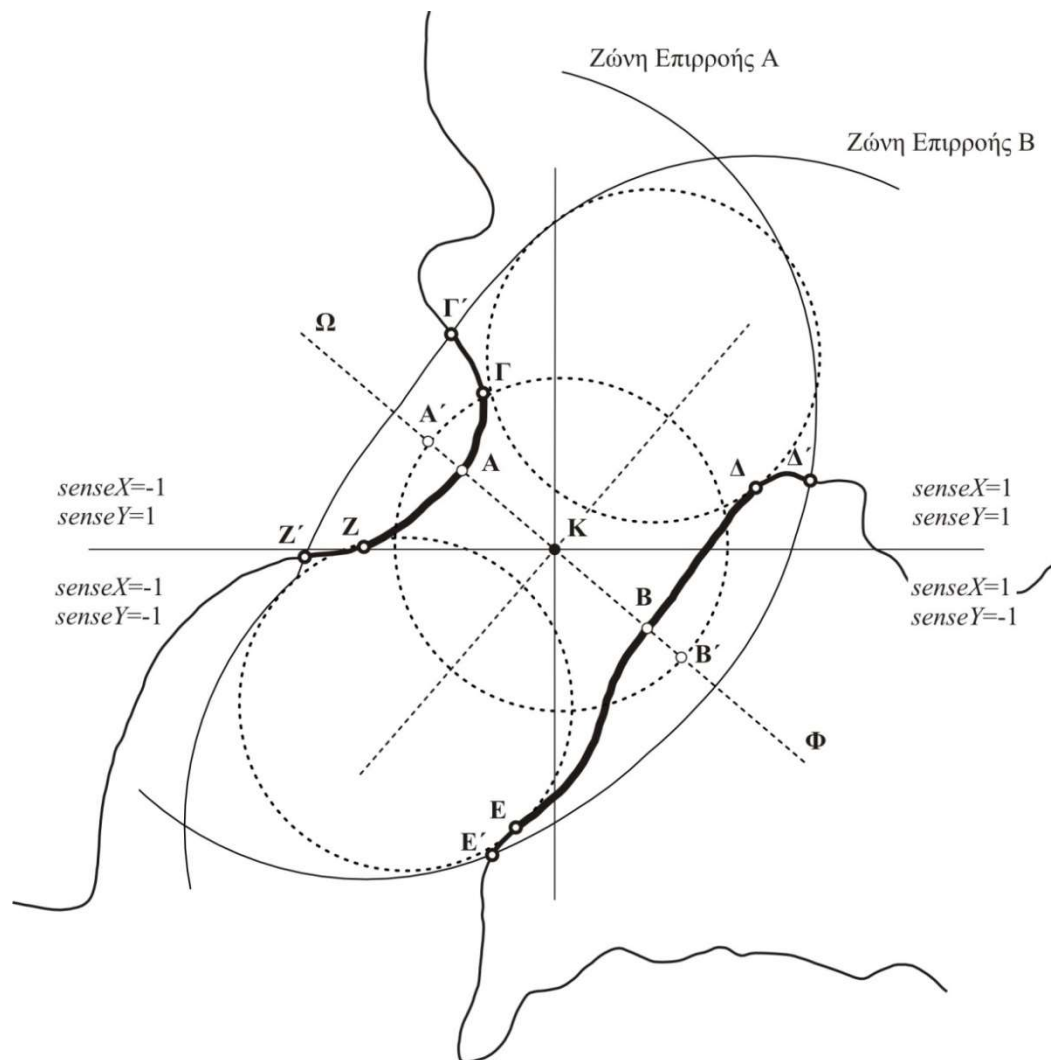
### 4.3 Γενίκευση τμημάτων σύγκλισης (Κατηγορία Γ)

Ως τμήματα σύγκλισης χαρακτηρίζονται δύο ή περισσότερα απέναντι  $\epsilon$ -μη-κυρτά τμήματα, τα οποία απέχουν απόσταση μικρότερη από το όριο οπτικής ευαναγνωσιμότητας. Στην πραγματικότητα, συνθέτουν περιοχές όπου παρατηρείται στένωση της γραμμής. Κατά την απεικόνισή τους σε χάρτες μικρότερης κλίμακας, τα τμήματα σύγκλισης ταυτίζονται ή είναι δυσδιάκριτα ως ανεξάρτητες οντότητες, γεγονός χαρτογραφικά και αισθητικά μη αποδεκτό. Για την αντιμετώπιση του προβλήματος εισάγεται μια νέα μέθοδος γενίκευσής τους. Η μέθοδος στηρίζεται στη γενική ιδέα της μετάθεσης των γραμμικών τμημάτων με τρόπο κατάλληλο, ώστε η περιοχή ανάμεσα τους να διαπλατύνεται, σε βαθμό που να εξασφαλίζεται η οπτική ευαναγνωσιμότητα. Η εφαρμογή της περιλαμβάνει τρία στάδια επεξεργασίας των τμημάτων: την ανεύρεση χαρακτηριστικών ιδιοτήτων τους, τη διεύρυνση και τη συμπίεσή τους μέσω του αλγορίθμου 'Depress' που δημιουργείται. Ένα γεγονός που πρέπει να επισημανθεί είναι πως, κατά την υλοποίηση της διαδικασίας, το εκάστοτε τμήμα σύγκλισης δεν αντιμετωπίζεται ανεξάρτητα, όπως συμβαίνει με τις διεργασίες γενίκευσης των  $\epsilon$ -τμημάτων των υπολοίπων τριών κατηγοριών (Α, Β και Δ). Αντίθετα, εισάγεται η έννοια της 'περιοχής σύγκλισης', η οποία διαμορφώνεται από δύο ή παραπάνω αλληλοεπηρεαζόμενα τμήματα σύγκλισης. Κάθε τμήμα, ως επιμέρους οντότητα της περιοχής, 'συμβάλει' στη δημιουργία των παραμέτρων εφαρμογής και βάσει αυτών υπόκειται σε επεξεργασία, ώστε να επιτευχθεί η γενίκευσή της.

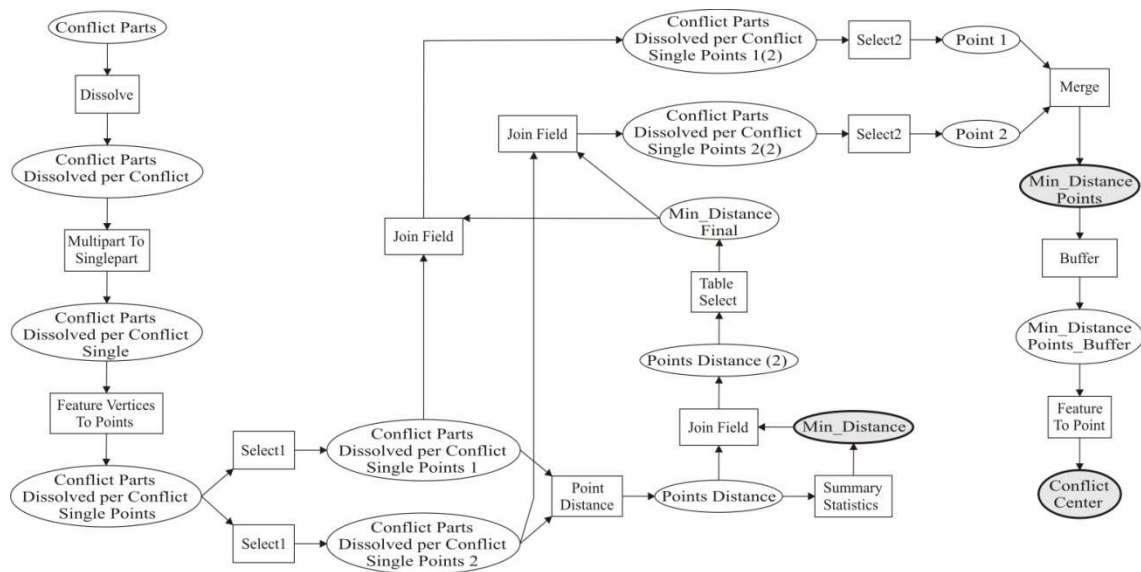
#### 4.3.1 Ιδιότητες περιοχής σύγκλισης

Όπως αναφέρεται στο κεφάλαιο 2.4.3, τα τμήματα σύγκλισης θεωρητικά δημιουργούνται όταν δίσκος του Perkal διαμέτρου  $\epsilon$ , που κυλά στο εσωτερικό της γραμμής, αδυνατεί να διαπεράσει στένωση που αυτή παρουσιάζει, με αποτέλεσμα να την τέμνει σε τέσσερα ή περισσότερα (ανάλογα με τη μορφή της στένωσης) σημεία. Σε συνέχεια αυτής της λογικής, το πρόβλημα ευαναγνωσιμότητας επιλύεται, αν η στένωση διαπλατυνθεί, ώστε ο ίδιος δίσκος του Perkal να διέρχεται από αυτήν. Γίνεται φανερό πως, ουσιαστικά, η γενίκευση των τμημάτων σύγκλισης αποσκοπεί στην ευκρινή απόδοση της περιοχής που αυτά οριοθετούν και διαμορφώνουν και όχι αυτών καθ'αυτών των τμημάτων, ως αυτοτελείς οντότητες. Για το λόγο αυτό, οι μετασχηματισμοί γενίκευσης εφαρμόζονται ανεξάρτητα στο εκάστοτε τμήμα, βάσει, όμως, παραμέτρων που προκύπτουν από την ανάλυση των ιδιοτήτων της περιοχής σύγκλισης. Έτσι, στο πρώτο στάδιο επεξεργασίας, ανευρίσκονται οι βασικές ιδιότητες που χαρακτηρίζουν την περιοχή σύγκλισης. Πιο συγκεκριμένα, υπολογίζεται η ελάχιστη απόσταση μεταξύ των τμημάτων σύγκλισης, εντοπίζονται τα σημεία μεταξύ των οποίων αυτή εμφανίζεται (Α και Β στο Σχήμα 4.16), καθώς και το μέσο μεταξύ των σημείων αυτών, το οποίο αποτελεί το 'κέντρο' της περιοχής σύγκλισης (Κ στο Σχήμα 4.16). Το κέντρο περιοχής σύγκλισης αποτελεί ιδιαίτερης σημαντικότητας θέση, αφού, όπως προκύπτει και από τον ορισμό του, σε αυτή εμφανίζεται σε μέγιστο βαθμό το πρόβλημα οπτικής ευαναγνωσιμότητας. Τα σημεία ελάχιστης απόστασης ορίζονται ως οι 'κορυφές' των τμημάτων στα οποία ανήκουν, ενώ τα σημεία τομής του δίσκου του Perkal με τη γραμμή (Γ, Δ, Ζ, Ε στο Σχήμα 4.16) αποτελούν τα όρια της περιοχής σύγκλισης.

Η επεξεργασία λαμβάνει χώρα στο περιβάλλον του λογισμικού πακέτου ArcGIS®, όπως παρουσιάζεται στο Σχήμα 4.17. Τα γραμμικά τμήματα που διαμορφώνουν την περιοχή σύγκλισης (*Conflict Parts*) διασπώνται (*Feature Vertices To Points*) στα σημεία από τα οποία δομούνται (*Conflict Parts Dissolved per Conflict Single Points*) και διαχωρίζονται (*Select1*) σε ανεξάρτητες οντότητες. Στη συνέχεια, υπολογίζονται (*Point Distance*), μία προς μία, όλες οι αποστάσεις μεταξύ των σημείων των γραμμών και επιλέγεται (*Summary Statistics*) η ελάχιστη (*Min\_Distance*). Τα σημεία, μεταξύ των οποίων αυτή εμφανίζεται, εντοπίζονται (*Table Select*) και αποκόπτονται (*Select2*) από τα υπόλοιπα, αφού αποτελούν τις κορυφές των τμημάτων σύγκλισης (*Min\_Distance Points*). Τέλος, γύρω από τις κορυφές, δημιουργείται (*Buffer*) ο γεωμετρικός τόπος των θέσεων που απέχουν από αυτές απόσταση ίση ή μικρότερη της ελάχιστης. Το κέντρο του πεδίου που προκύπτει αποτελεί το κέντρο της περιοχής σύγκλισης (*Conflict Center*), το οποίο και εντοπίζεται (*Feature To Point*).



Σχήμα 4.16: Παράδειγμα περιοχής σύγκλισης.



Σχήμα 4.17: Μοντέλο υπολογισμού ποσοτικών ιδιοτήτων και εντοπισμού χαρακτηριστικών θέσεων των περιοχών σύγκλισης.

#### 4.3.2 Επεξεργασία περιοχής σύγκλισης και διεύρυνση τμημάτων σύγκλισης

Όπως αναφέρεται παραπάνω, η ευκρινής απεικόνιση της περιοχής σύγκλισης σε χάρτη μικρότερης κλίμακας επιτυγχάνεται με την κατάλληλη διαπλάτυνσή της, ώστε θεωρητικά, να τη διαπερνά δίσκος του Perkal διαμέτρου  $\epsilon$ . Η διαπλάτυνση πραγματοποιείται μέσω της μετάθεσης 'προς τα έξω' ή αλλιώς, μέσω της 'συμπίεσης' των τμημάτων σύγκλισης, διαδικασία η οποία, στην πράξη, συνίσταται στη μετατόπιση των σημείων που τα δομούν. Η συμπίεση ορίζεται να λαμβάνει χώρα ισοβαρώς μεταξύ των τμημάτων, ώστε να μην αλλοιώνεται η μορφή και η γεωμετρία της περιοχής και συγχρόνως, με μέγεθος που να εξασφαλίζει πως η μεταξύ τους απόσταση είναι πάντοτε μεγαλύτερη ή ίση της τιμής  $\epsilon$ . Το μέγεθος και η διεύθυνση μετατόπισης του εκάστοτε σημείου κάθε τμήματος εξαρτάται από τη σχετική του θέση ως προς το κέντρο της περιοχής σύγκλισης. Σαν βάση, δηλαδή, υπολογισμού των παραμέτρων μετατόπισης, ορίζονται εκείνες που επιλύουν το πρόβλημα ευαναγνωσιμότητας στη θέση όπου αυτό παρατηρείται στο μέγιστο βαθμό.

Για τη διέλευση δίσκου του Perkal διαμέτρου  $\epsilon$  από το κέντρο της περιοχής σύγκλισης, απαιτείται η μετάθεση των σημείων ελάχιστης απόστασης A και B σε δύο νέες θέσεις A' και B' αντίστοιχα, οι οποίες απέχουν απόσταση, τουλάχιστον, ίση με τη διάμετρο  $\epsilon$  ( $(A'B') \geq \epsilon$ ). Συγχρόνως, τα σημεία A και B πρέπει να μετατοπισθούν κατά ίσα διαστήματα, ώστε να διασφαλισθεί η ισοβαρής διαπλάτυνση της περιοχής. Οι δύο παραπάνω δεσμεύσεις ικανοποιούνται από κοινού, μόνον όταν τα νέα σημεία A', B' και το κέντρο (K) της περιοχής σύγκλισης είναι συνευθειακά. Επιπλέον, το ελάχιστο διάστημα που υπολείπεται για την επίλυση του προβλήματος ευαναγνωσιμότητας στη συγκεκριμένη θέση είναι  $\epsilon - Min\_Distance$ . Ως εκ τούτου, κάθε μία εκ των κορυφών A, B των τμημάτων σύγκλισης αρκεί να μετατοπισθεί 'προς τα έξω', σε απόσταση  $(AA') = (BB') = \frac{\epsilon - Min\_Distance}{2}$ . Η διεύθυνση

μετατόπισης των κορυφών, κατά την οποία κάτι τέτοιο επιτυγχάνεται είναι μοναδική και ορίζεται από την ευθεία ΩΦ που διέρχεται από αυτές. Όπως φαίνεται στο Σχήμα 4.16, με την εφαρμογή των παραπάνω κανόνων, τα νέα σημεία Α' και Β' εντοπίζονται στα όρια του δίσκου του Perkal και απέχουν ίσες αποστάσεις από τα αντίστοιχα αρχικά, στοιχεία που ικανοποιούν τους κανόνες συμπίεσης που έχουν τεθεί. Σε περίπτωση υλοποίησης της διαδικασίας υπό οποιαδήποτε άλλη διεύθυνση, το απαιτούμενο μέγεθος μετάθεσης είναι μεγαλύτερο, ενώ οι θέσεις Α' και Β' εντοπίζονται εκατέρωθεν της ευθείας ΩΦ. Δεδομένου πως η μετατόπιση των υπολοίπων σημείων των τμημάτων σύγκλισης εξαρτάται από τις παραμέτρους μετατόπισης των κορυφών, κάτι τέτοιο θα υποβάθμιζε την ποιότητα της γενικευμένης περιοχής, η οποία θα εμφανίζονταν αλλοιωμένη ως προς το μέγεθος, τον προσανατολισμό και το σχήμα.

Θεωρητικά, ο δίσκος του Perkal, εισερχόμενος στην περιοχή σύγκλισης, τέμνει τα τμήματα σε ένα άπειρο πλήθος από ζεύγη σημείων. Ιδανικά, λοιπόν, η διαπλάτυνση της περιοχής έγκειται στην υλοποίηση μιας διαδικασίας αντίστοιχης με εκείνη που πραγματοποιείται για τη μετατόπιση των κορυφών Α και Β, σε κάθε ένα από τα ζεύγη αυτά. Είναι προφανές, πως κάτι τέτοιο είναι μη εφαρμόσιμο σε γραμμές που αποδίδονται διανυσματικά και κατά συνέπεια, δομούνται από πεπερασμένα σε αριθμό και συγκεκριμένα κάθε φορά σημεία. Επιπλέον, η διαδικασία δεν είναι εφικτό να υλοποιηθεί σε ζεύγη που δημιουργούνται μεταξύ των υπαρχόντων σημείων των τμημάτων. Και τούτο, διότι η τυχαία μορφή των τελευταίων καθιστά αδύνατη την 'ένα προς ένα' αντιστοίχιση και συνάρτηση των σημείων τους βάσει αντικειμενικών κριτηρίων, αφού ακόμη και ο αριθμός αυτών, στη συντριπτική πλειοψηφία των περιπτώσεων, διαφέρει. Το μοναδικό ζεύγος σημείων που μπορεί να εντοπισθεί μεταξύ των τμημάτων είναι οι κορυφές (καθώς και τα άκρα, τα οποία όμως δεν υπόκεινται σε μετάθεση). Για το λόγο αυτό, η διαπλάτυνση της περιοχής σύγκλισης πραγματοποιείται με την ανεξάρτητη μετατόπιση του εκάστοτε σημείου, βάσει παραμέτρων που εξαρτώνται εκείνων που έχουν υπολογισθεί για τη μετάθεση των κορυφών. Η διαδικασία αποτελεί την πλέον αποτελεσματική προσέγγιση της αντίστοιχης θεωρητικής, αφού η συσχέτιση των παραμέτρων μετατόπισης κάθε σημείου με αυτές που επιλύουν το πρόβλημα ευαναγνωσιμότητας στη θέση μέγιστης εμφάνισής του, αποτελεί εχέγγυο για την επίλυσή του σε όλο το εύρος της περιοχής. Έτσι, λαμβάνοντας υπόψη πως το μέγεθος μετάθεσης των κορυφών ( $\frac{\epsilon - \text{Min} - \text{Distance}}{2}$ ) είναι το μέγιστο και αναγκαίο που απαιτείται για τη

διαπλάτυνση της περιοχής, τα υπόλοιπα σημεία κάθε τμήματος σύγκλισης μετατοπίζονται σε αποστάσεις ανάλογες αυτής, οι οποίες φθίνουν μέχρι μιας μηδενικής στα άκρα. Για λόγους διατήρησης του προσανατολισμού και της γεωμετρίας της παράγωγης περιοχής, η διεύθυνση μετατόπισης ορίζεται να είναι κοινή για το σύνολο των σημείων. Θεωρητικά, αυτό επιτυγχάνεται όταν ο δίσκος του Perkal κυλά δια μέσου της περιοχής σύγκλισης υπό μία, συγκεκριμένη ευθεία. Ιδανικά, η εν λόγω ευθεία θα προσδιορίζονταν μέσω μαθηματικών μοντέλων (π.χ. συνθήκη ελαχίστων τετραγώνων), αν ήταν γνωστή η ακριβής πορεία κύλισης του δίσκου. Δεδομένου πως κάτι τέτοιο δεν είναι εφικτό, επιχειρείται η εύρεση της βέλτιστης δυνατής προσέγγισης αυτής, με κριτήριο τις ιδιότητες της περιοχής σύγκλισης. Βάση αυτών, ως άξονας κύλισης του δίσκου τίθεται η μεσοκάθετος στο ευθύγραμμο τμήμα ΑΒ που συνδέει τις κορυφές των τμημάτων, το οποίο, όπως

αναφέρεται παραπάνω, είναι ευθυγραμμισμένο με την ευθεία ΩΦ. Όπως φαίνεται στο Σχήμα 4.16, οι δύο ευθείες ορίζουν ένα σύστημα αξόνων με αρχή το κέντρο της περιοχής σύγκλισης. Η θέση του, σε συνδυασμό με το γεγονός πως η ευθεία ΩΦ αποτελεί τη διεύθυνση μετάθεσης των κορυφών, καθιστούν το εν λόγω σύστημα στην πιο έγκυρη βάση για τον προσανατολισμό της παράγωγης περιοχής. Έτσι, τα σημεία των τμημάτων σύγκλισης μετατοπίζονται κάθετα στον άξονα κύλισης ή διαφορετικά, σε διεύθυνση παράλληλη της ευθείας ΩΦ. Τέλος, η κατεύθυνση μετατόπισης κάθε σημείου εξαρτάται από τη σχετική θέση της κορυφής του τμήματος στο οποίο ανήκει, ως προς το κέντρο της περιοχής σύγκλισης και είναι τέτοια που να εξασφαλίζει πως η νέα του θέση εντοπίζεται εξωτερικά της αρχικής. Αναλυτική περιγραφή της διαδικασίας, η οποία υλοποιείται μέσω του αλγορίθμου 'Depress', παρουσιάζεται στην επόμενη ενότητα.

Εφαρμόζοντας την τεχνική συμπίεσης στα τμήματα σύγκλισης, με τους κανόνες που περιγράφονται παραπάνω, ανακύπτουν δύο ζητήματα που χρήζουν περαιτέρω διερεύνησης. Το πρώτο, και πιο σημαντικό, αφορά στην αδυναμία επίλυσης του προβλήματος ευαναγνωσιμότητας στα άκρα της περιοχής σύγκλισης και, ενδεχομένως, σε γειτονικές αυτών θέσεις. Το γεγονός οφείλεται στη λειτουργία του αλγορίθμου 'Depress' και συγκεκριμένα, στη σταδιακή ελάττωση του μεγέθους μετατόπισης των σημείων κάθε τμήματος, από μία μέγιστη τιμή στην κορυφή του, σε μηδενική στα δύο του άκρα. Η μη μετατόπιση των τελευταίων, εξ ορισμού συνεπάγεται τη διατήρηση του προβλήματος στα όρια της περιοχής σύγκλισης. Όπως φαίνεται στο παράδειγμα του Σχήματος 4.16, ο δίσκος του Perkal δεν είναι δυνατό να εισχωρήσει στην περιοχή χωρίς να μετατεθούν τα σημεία τομής του (E, Z και Γ, Δ) με τη γραμμή. Επιπρόσθετα, είναι πιθανό το μέγεθος μετατόπισης παρακείμενων των άκρων σημείων να μην επαρκεί, ώστε τα μέρη των τμημάτων που δομούν να απέχουν απόσταση ίση ή μεγαλύτερη της διαμέτρου  $\epsilon$ .

Το δεύτερο ζήτημα δεν παρατηρείται κατά κανόνα, σε κάθε εφαρμογή της τεχνικής συμπίεσης. Αφορά περιπτώσεις τμημάτων σύγκλισης, των οποίων η καμπυλότητα δε διατηρείται μετά το μετασχηματισμό μετάθεσης ή μεταβάλλεται σε βαθμό που να δημιουργούνται ασυνέχειες στις θέσεις σύνδεσης με τα γειτονικά  $\epsilon$ -τμήματα. Στο Σχήμα 4.18 παρουσιάζεται ένα χαρακτηριστικό παράδειγμα μιας τέτοιας περίπτωσης. Το γενικευμένο τμήμα έχει αντίθετη καμπυλότητα της αρχικής, ενώ στα σημεία σύνδεσης με την υπόλοιπη γραμμή είναι εμφανής η ασυνέχεια στην ομαλότητα. Η αλλαγή καμπυλότητας έχει άμεση επίπτωση στη χαρτογραφική ποιότητα του παράγωγου προϊόντος, αφού αλλοιώνεται η μορφή και η γεωμετρία της γραμμής. Επιπλέον, η εμφάνιση ασυνεχειών, οι οποίες, εκτός της αλλαγής, είναι δυνατό να δημιουργούνται και από μικρότερες μεταβολές της καμπυλότητας όταν το σχήμα του τμήματος είναι πρόσφορο, υποβαθμίζει αισθητικά το



Σχήμα 4.18: Προβλήματα κατά τη γενίκευση τμημάτων σύγκλισης.

αποτέλεσμα. Τα προβλήματα αυτά είναι απόρροια του τρόπου λειτουργίας του αλγορίθμου 'Depress' συναρτήσει της γεωμετρίας του προς επεξεργασία τμήματος. Όπως αναφέρεται και παραπάνω, τα σημεία που δομούν τα τμήματα μετατοπίζονται υπό σταθερή διεύθυνση και σε αποστάσεις που μεταπίπτουν από μια μέγιστη τιμή στην κορυφή, στη μηδενική στις θέσεις σύνδεσης (άκρα). Με δεδομένο πως οι παράμετροι μετατόπισης είναι συγκεκριμένοι, αφού καθορίζονται βάσει των ιδιοτήτων της περιοχής σύγκλισης, το πρόβλημα ανακύπτει όταν το ανάπτγμα του τμήματος είναι αναλογικά μικρό σε σχέση με το μέγεθος μέγιστης μετατόπισης. Σε μια τέτοια περίπτωση, η μετάπτωση από τη μέγιστη στη μηδενική μετατόπιση λαμβάνει χώρα απότομα, σε μικρό διάστημα, με αποτέλεσμα να δημιουργούνται μεγάλες μεταβολές ή αλλαγή στην καμπυλότητα του τμήματος. Το φαινόμενο γίνεται εντονότερο όσο η αναλογία μέγιστης μετατόπισης/ανάπτγματος αυξάνει.

Τα δύο προβλήματα που ανακύπτουν κατά την εφαρμογή της τεχνικής συμπίεσης αντιμετωπίζονται από κοινού, μέσω της διεύρυνσης των τμημάτων σύγκλισης. Η διαδικασία υλοποιείται με την προσάρτηση μέρους από τα εκατέρωθεν γειτονικά  $\epsilon$ -τμήματα. Με τον τρόπο αυτό, εξασφαλίζεται η ευαναγνωσιμότητα στα όρια της περιοχής σύγκλισης, αφού τα άκρα της, ως ενδιάμεσα πλέον σημεία των τμημάτων, υπόκεινται σε μετατόπιση μέσω του αλγορίθμου 'Depress'. Τα νέα άκρα, της διευρυμένης περιοχής σύγκλισης, απέχουν απόσταση μεγαλύτερη της διαμέτρου  $\epsilon$ , γεγονός που επιτρέπει στο θεωρητικό δίσκο του Perkal να εισχωρήσει σε αυτή. Επιπλέον, το πρόβλημα αλλαγής καμπυλότητας και εμφάνισης ασυνεχειών που παρατηρείται σε ορισμένα τμήματα απαλείφεται ή έστω περιορίζεται σε αποδεκτό βαθμό. Και τούτο, διότι με τη διεύρυνση των τμημάτων αντιμετωπίζεται η γενεσιουργός αιτία του, δηλαδή το αναλογικά μικρό μήκος τους σε σχέση με το μέγεθος μετατόπισης. Δεδομένου πως το τελευταίο καθορίζεται βάσει της αρχικής περιοχής σύγκλισης κατά την εφαρμογή του αλγορίθμου 'Depress', η αναλογία μέγιστης μετατόπισης/ανάπτγματος ελαττώνεται, διορθώνοντας το συγκεκριμένο μειονέκτημα της διαδικασίας συμπίεσης. Να σημειωθεί, πως η διεύρυνση λαμβάνει χώρα σε όλα τα τμήματα της περιοχής σύγκλισης, αφού είναι αναγκαία για την επίλυση του προβλήματος ευαναγνωσιμότητας των ορίων της. Αυτό δεν έχει αρνητική επίπτωση στη μορφή τμημάτων που δεν παρατηρείται πρόβλημα αλλαγής καμπυλότητας ή εμφάνισης ασυνεχειών. Αντιθέτως, συμβάλει στην αισθητική βελτίωση του αποτελέσματος γενίκευσής τους.

Σημαντικό ζητούμενο αποτελεί το μέγεθος διεύρυνσης των τμημάτων σύγκλισης. Και τούτο, διότι τα μέρη που προσαρτώνται σε αυτά θα πρέπει να είναι τα ικανά και αναγκαία για την επίλυση των προβλημάτων που ανακύπτουν κατά την εφαρμογή της τεχνικής συμπίεσης, χωρίς όμως να επηρεάζεται σε μεγάλο βαθμό η δομή και η γεωμετρία των γειτονικών  $\epsilon$ -τμημάτων από τα οποία αποκόπτονται και κατ'επέκταση, το σύνολο της γενικευμένης γραμμής. Παράλληλα, είναι απαραίτητη η θέσπιση ενός παραμετροποιημένου τρόπου υπολογισμού του, ώστε να μην είναι αυθαίρετος και να είναι ανεξάρτητος, τόσο από τη μορφολογία της εκάστοτε γραμμής, όσο και από το επίπεδο γενίκευσης.

Η διεύρυνση των τμημάτων σύγκλισης στηρίζεται στην εφαρμογή 'ζωνών επιρροής'. Η διαδικασία υλοποιείται στο περιβάλλον του λογισμικού πακέτου ArcGIS®, όπως παρουσιάζεται στο Σχήμα 4.19. Γύρω από κάθε τμήμα της περιοχής σύγκλισης (*Conflict Parts Dissolved per Conflict Single*) δημιουργείται 'ζώνη επιρροής' (*Buffer*,

*Buffer Conflict Parts Dissolved per Conflict Single*), εύρους ίσου με την τιμή  $\epsilon$ . Η 'ζώνη επιρροής' τέμνει τη χαρτογραφική γραμμή σε δύο ή περισσότερα σημεία. Τα σημεία τομής που εντοπίζονται εγγύτερα στο έτερο τμήμα σύγκλισης, ορίζουν τα νέα άκρα αυτού. Στην πράξη, οι 'ζώνες επιρροής' της περιοχής σύγκλισης υπόκεινται σε μια διαδικασία αλληλοεπικάλυψης (*Intersect*). Το κοινό τους πεδίο απομονώνεται, καθώς οριοθετεί τη 'διευρυμένη περιοχή σύγκλισης' (*Common Buffer Area*). Τα  $\epsilon$ -τμήματα που αποκόπτονται στο εσωτερικό αυτής συνενώνονται, διαμορφώνοντας τα διευρυμένα τμήματα σύγκλισης. Εντός της διευρυμένης περιοχής σύγκλισης εντοπίζονται:

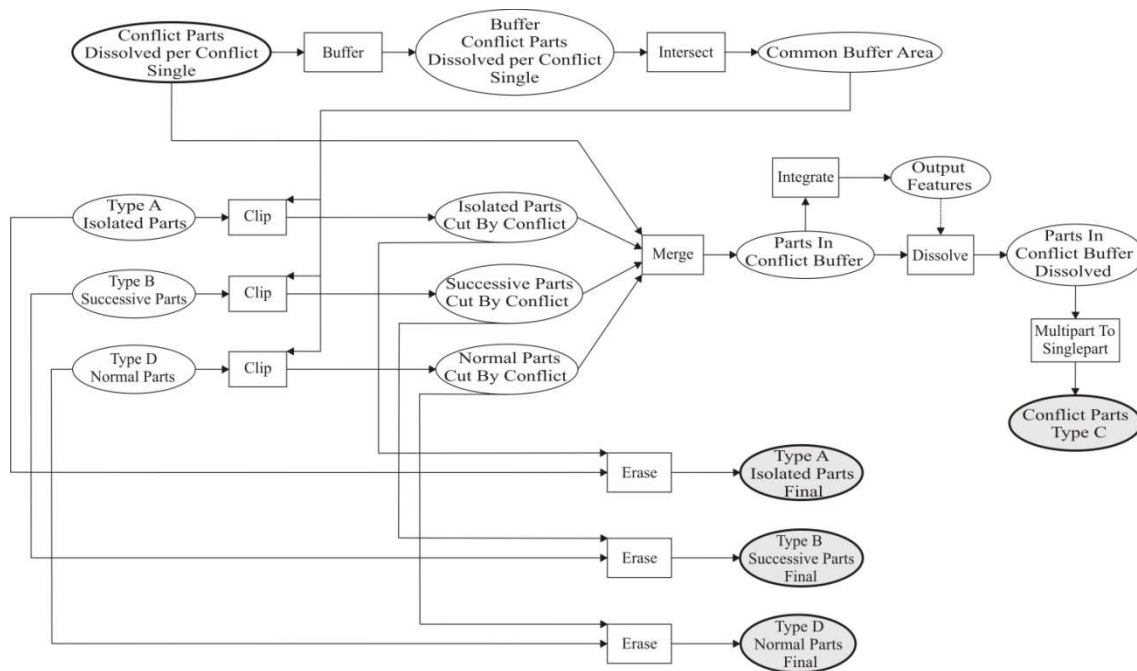
- πάντοτε ακέραια τα αρχικά τμήματα σύγκλισης. Δεν συναντάται, δηλαδή, το ενδεχόμενο η αρχή ή το πέρας τους να βρίσκεται εκτός αυτής. Το γεγονός επαληθεύεται και στην ακραία περίπτωση, που κάποιο εκ των τμημάτων έχει 'εκφυλισθεί' σε σημείο και η 'ζώνη επιρροής' γύρω από αυτό έχει τη μορφή κύκλου ακτίνας  $\epsilon$ . Τα άκρα του έτερου τμήματος της περιοχής απέχουν από το τμήμα-σημείο αποστάσεις μικρότερες της τιμής  $\epsilon$  (αφού διαφορετικά δε θα είχαν δημιουργηθεί τμήματα σύγκλισης στη συγκεκριμένη θέση) και ως εκ τούτου, εντοπίζονται εντός της 'ζώνης επιρροής'-κύκλου.
- μέρη από τα παρακείμενα  $\epsilon$ -τμήματα, τα οποία μπορεί να είναι αμφίπλευρα  $\epsilon$ -μη-κυρτά ή/και  $\epsilon$ -κυρτά. Πιο συγκεκριμένα, ένα τμήμα σύγκλισης γειτνιάζει με δύο αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα όταν, πριν το διαχωρισμό του ως ανεξάρτητη οντότητα κατά τη διαδικασία κατάτμησης (βλ. κεφ. 2.4.3), αποτελεί μέρος αλληλουχίας ή επικαλυπτόμενων  $\epsilon$ -τμημάτων και συγχρόνως, εντοπίζεται στο μέσον αυτών. Στην αντίστοιχη περίπτωση που έχει αποκοπεί από ακραία θέση αλληλουχίας ή επικάλυψης, συνορεύει με αμφίπλευρο  $\epsilon$ -μη-κυρτό τμήμα στο ένα του άκρο και με  $\epsilon$ -κυρτό στο έτερο. Τέλος, όταν αρχικά εμφανίζεται απομονωμένο, τα γειτονικά του τμήματα εξακολουθούν να είναι  $\epsilon$ -κυρτά.
- μέρη από  $\epsilon$ -τμήματα και πέραν των άμεσα γειτονικών. Το γεγονός παρατηρείται όταν τόσο η απόσταση μεταξύ των τμημάτων σύγκλισης, όσο και το ανάπτυγμα τους τείνουν στο ελάχιστο. Μια τέτοια περίπτωση είναι εξαιρετικά σπάνιο να εμφανισθεί σε μια χαρτογραφική γραμμή που απεικονίζει φυσική οντότητα. Εξετάζεται όμως, αφού υφίσταται ως θεωρητικό ενδεχόμενο. Για το λόγο αυτό, έλεγχος αποκοπής από τη διευρυμένη περιοχή σύγκλισης, πραγματοποιείται, εκτός των άλλων, και για τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα, τα οποία δύναται να συνορεύουν με τα άμεσα γειτονικά.

Πρακτικά και βάσει των παραπάνω, τα εντός περιοχής γειτονικά  $\epsilon$ -τμήματα αποκόπτονται (*Clip, ... Parts Cut By Conflict*) και συνενώνονται (*Merge*) με τα αρχικά τμήματα σύγκλισης, δημιουργώντας τα 'διευρυμένα τμήματα σύγκλισης' (*Conflict Parts Type C*). Αντίθετα, τα μέρη των γειτονικών  $\epsilon$ -τμημάτων που βρίσκονται εκτός αυτής απομονώνονται (*Erase*), δημιουργώντας τα τελικά, προς επεξεργασία  $\epsilon$ -τμήματα, των υπολοίπων τριών κατηγοριών (A, B και Δ) (*Type A Isolated Parts Final*, *Type B Successive Parts Final* και *Type D Normal Parts Final*). Στο παράδειγμα του Σχήματος 4.16, η 'ζώνη επιρροής' του τμήματος με κορυφή B τέμνει τη γραμμή πλησίον του τμήματος με κορυφή A στα σημεία Γ' και Ζ'. Αντίστοιχα, η 'ζώνη επιρροής' του τμήματος A στα σημεία Δ' και Ε'. Μεταξύ αυτών αποκόπτονται

τμήματα από τα γειτονικά (γραμμές μέσου πάχους), συνενώνονται με τα αρχικά (γραμμές μεγάλου πάχους) και διαμορφώνονται τα διευρυμένα τμήματα σύγκλισης. Τα σημεία τομής οριοθετούν τη διευρυμένη περιοχή σύγκλισης και παίζουν σημαντικό ρόλο στην επεξεργασία της γραμμής, αφού αποτελούν τα νέα σημεία σύνδεσης της 'προβληματικής' με τις γειτονικές περιοχές και παραμένουν σταθερά κατά τη διαδικασία γενίκευσης.

Η εφαρμογή στηρίζεται θεωρητικά στην ιδέα πως, εκατέρωθεν κάθε τμήματος σύγκλισης προσαρτώνται μέρη των γειτονικών, μεγέθους ικανού και αναγκαίου, ώστε ο δίσκος του Perkal να εισέρχεται στη διευρυμένη περιοχή που δημιουργείται, αν μετατεθεί μόνο το συγκεκριμένο τμήμα σύγκλισης. Η ζώνη επιρροής εύρους  $\epsilon$  σε κάθε τμήμα προσομοιώνει, ακριβώς, αυτή τη διαδικασία. Γίνεται φανερό πως, με την υλοποίηση σε όλα τα τμήματα, εξασφαλίζεται η επίλυση του προβλήματος οπτικής ευαναγνωσιμότητας στα όρια της αρχικής περιοχής σύγκλισης. Συγχρόνως, υπολογίζοντας το μέγεθος της διεύρυνσης με βάση την τιμή  $\epsilon$ , αντιμετωπίζεται επιτυχώς το ζήτημα αλλαγής καμπυλότητας και εμφάνισης ασυνεχειών. Και τούτο, διότι το ανάπτυγμα των τμημάτων αυξάνει σημαντικά σε σχέση με τη μέγιστη μετατόπιση, η οποία κατά την εφαρμογή του αλγορίθμου 'Depress' ορίζεται βάσει του μισού της τιμής  $\epsilon$ . Η φιλοσοφία λειτουργίας της 'ζώνης επιρροής' διασφαλίζει πως η μορφή και το μέγεθος κάθε διευρυμένης περιοχής εξαρτάται αποκλειστικά από τη μορφή των τμημάτων της αρχικής περιοχής σύγκλισης στα οποία εφαρμόζεται. Παράλληλα, με τον ορισμό του εύρους εφαρμογής ίσο με την τιμή  $\epsilon$ , επιτυγχάνεται η συνάρτηση της διαδικασίας με το εκάστοτε επιθυμητό επίπεδο γενίκευσης. Έτσι, συμπερασματικά, η επίλυση του προβλήματος σύγκλισης ολοκληρώνεται με τη γενίκευση των τμημάτων που ανήκουν στη διευρυμένη περιοχή, βάσει των παραμέτρων που προκύπτουν από την ανάλυση των ιδιοτήτων των αρχικών τμημάτων σύγκλισης. Στα μέρη των γειτονικών  $\epsilon$ -τμημάτων που βρίσκονται εκτός 'ζώνης επιρροής' πραγματοποιείται η κατάλληλη επεξεργασία, ανάλογα με την κατηγορία (A, B ή Δ) στην οποία αρχικά ανήκαν. Για το λόγο αυτό, οι διεργασίες για τη γενίκευση των τμημάτων σύγκλισης προηγούνται της γενίκευσης των τμημάτων των υπολοίπων τριών κατηγοριών. Να σημειωθεί πως κάτι τέτοιο δε συνιστά πρόβλημα στην ποιότητα του τελικού προϊόντος, αφού η περίπτωση σύγκλισης αποτελεί το πιο σημαντικό, αλλά και δύσκολο διαχειρίσιμο ζητούμενο στη διαδικασία γενίκευσης μιας χαρτογραφικής γραμμής.

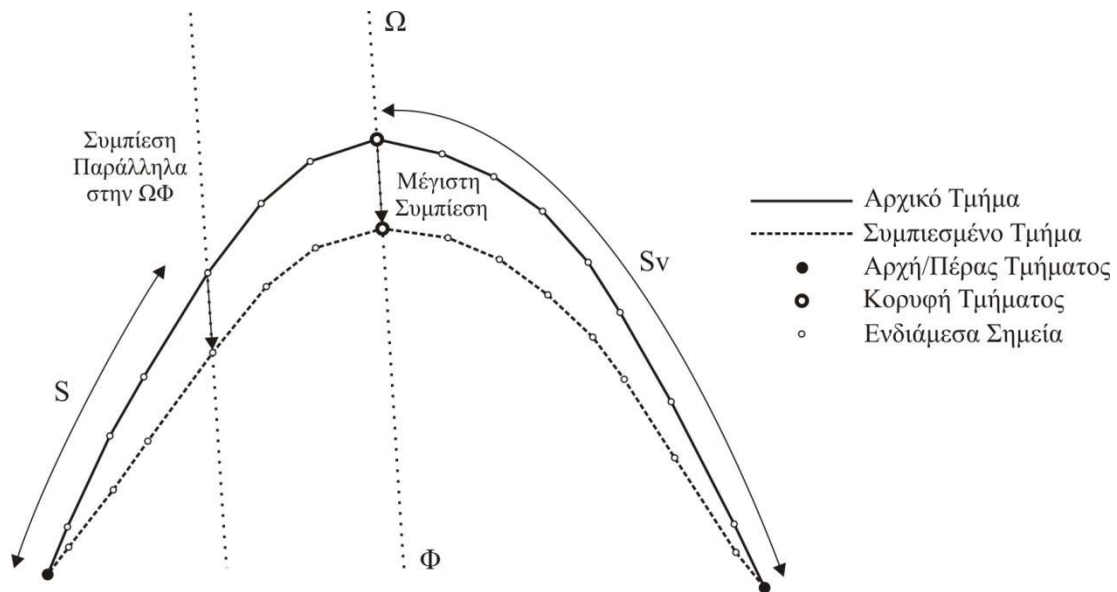




Σχήμα 4.19: Μοντέλο δημιουργίας διευρυμένης περιοχής σύγκλισης και αποκοπής τελικών ε-τμημάτων.

#### 4.3.3 Συμπίεση τμήματος σύγκλισης - Αλγόριθμος 'Depress'

Τα διευρυμένα τμήματα σύγκλισης υπόκεινται στη διαδικασία συμπίεσης μέσω του αλγορίθμου 'Depress', ώστε να ολοκληρωθεί η γενίκευσή τους. Κάθε σημείο που δομεί το εκάστοτε τμήμα μετατοπίζεται βάσει ενός συντελεστή συμπίεσης. Το μέγεθος της μετατόπισης είναι ανάλογο της επικαμπύλιας απόστασης του μετατοπιζόμενου σημείου από την αρχή ή το πέρας του τμήματος. Αν το σημείο εντοπίζεται πριν την κορυφή αυτού χρησιμοποιείται η αρχή, ενώ σε αντίθετη περίπτωση το πέρας. Η μετατόπιση πραγματοποιείται σε διεύθυνση παράλληλη της ευθείας ΩΦ που ενώνει τις κορυφές των τμημάτων σύγκλισης και σε κατεύθυνση που ορίζεται ανάλογα με τη θέση της κορυφής, ως προς το κέντρο της περιοχής σύγκλισης.



Σχήμα 4.20: Ο αλγόριθμος 'Depress'.

Η μετατόπιση των σημείων κατά την εφαρμογή δίνεται μέσω των μαθηματικών σχέσεων:

$$X_c = X_i + Depress \cdot Sense \cdot a \cdot S/S_v$$

$$Y_c = Y_i + Depress \cdot Sense \cdot b \cdot S/S_v, \text{ όπου:}$$

$X_c, Y_c$ : οι συντεταγμένες της προκύπτουσας γραμμής.

$X_i, Y_i$ : οι συντεταγμένες της αρχικής γραμμής.

*Depress*: ο συντελεστής συμπίεσης του τμήματος. Η τιμή του είναι σταθερή σε κάθε εφαρμογή, εξαρτάται από τις ιδιότητες της περιοχής σύγκλισης και το επίπεδο γενίκευσης και είναι ίση με  $Depress = \frac{\epsilon - Min\_Distance}{2}$ .

*Sense*: +1 ή -1, ανάλογα με τη θέση της κορυφής του τμήματος, σε σχέση με το κέντρο της περιοχής σύγκλισης (βλ. Σχήμα 4.16),

*a, b*: ο συντελεστής διεύθυνσης της ευθείας  $\Omega\Phi$  που διέρχεται από τις κορυφές των τμημάτων σύγκλισης,

*S*: το επικαμπύλιο μήκος ανάμεσα στην αρχή ή το πέρας του τμήματος και το εκάστοτε σημείο *i*. Για τα σημεία που βρίσκονται πριν την κορυφή, το μήκος *S* δίνεται από τη σχέση  $S_i = S_{i-1} + \sqrt{(X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2}$ , ενώ για τα σημεία που βρίσκονται μετά την κορυφή από τη σχέση  $S_i = S_{i+1} + \sqrt{(X_i - X_{i+1})^2 + (Y_i - Y_{i+1})^2}$ .

*S<sub>v</sub>*: το επικαμπύλιο μήκος ανάμεσα στην αρχή ή το πέρας του τμήματος και την κορυφή.

Η διαδικασία που παρουσιάζεται αφορά τη γενίκευση περιοχών σύγκλισης που διαμορφώνονται από δύο τμήματα. Ο αλγόριθμος *Depress* είναι σχεδιασμένος για εφαρμογές και σε περιοχές που διαμορφώνονται από περισσότερα των δύο τμημάτων. Κατά την υλοποίησή του, υπολογίζονται τα χαρακτηριστικά της περιοχής σύγκλισης και γενικεύονται τα τμήματα, μεταξύ των οποίων εμφανίζεται η ελάχιστη

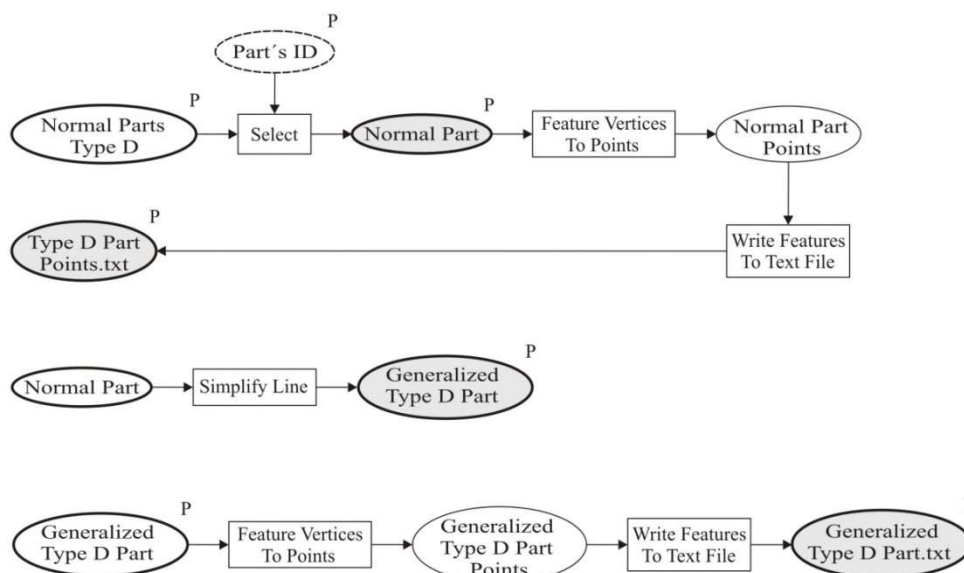
απόσταση. Στη συνέχεια, υπολογίζονται τα χαρακτηριστικά της νέας περιοχής που προκύπτει. Η διαδικασία λαμβάνει χώρα επαναληπτικά, μέχρι να δημιουργηθεί η περιοχή, από την οποία θα διέρχεται ο θεωρητικός δίσκος του Perkal.

Ο κώδικας υλοποίησης του αλγορίθμου επεξεργασίας των τμημάτων σύγκλισης δημιουργήθηκε σε γλώσσα προγραμματισμού *Python* και παρουσιάζεται στο Παράρτημα Π4.3.

#### 4.4 Γενίκευση $\epsilon$ -κυρτών τμημάτων (Κατηγορία Δ)

Τα  $\epsilon$ -κυρτά τμήματα, όπως αναφέρεται και στον ορισμό τους (βλ. κεφ. 2.1) χαρακτηρίζονται από ομαλότητα και μικρή πολυπλοκότητα. Η επεξεργασία για την απόδοση τους σε μικρότερη κλίμακα έγκειται στη γενίκευσή τους μέσω ενός τελεστή απλοποίησης. Η διαδικασία εκτελείται μέσω του αλγορίθμου *bendsimplify* που υποστηρίζεται από το λογισμικό πακέτο *ArcGIS®*. Ο συγκεκριμένος αλγόριθμος προτιμάται διότι, λόγω του τρόπου λειτουργίας του (βλ. κεφ. 3.6), δημιουργεί αρκετά ομαλές γραμμικές οντότητες, ιδιαίτερα σε περιπτώσεις μεγάλου βαθμού γενίκευσης. Το γεγονός συνάδει με τη φιλοσοφία και τα αποτελέσματα από την εφαρμογή των τριών μεθόδων επεξεργασίας των  $\epsilon$ -μη-κυρτών τμημάτων.

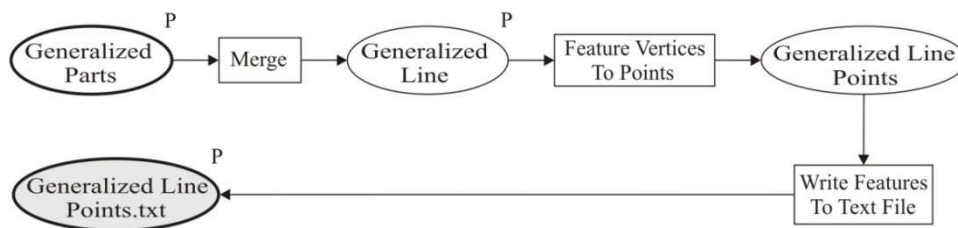
Με τη βοήθεια του εργαλείου *Model Builder* έχει δημιουργηθεί μοντέλο (Σχήμα 4.21), το οποίο δίνει τη δυνατότητα στο χρήστη να επιλέγει το  $\epsilon$ -τμήμα και να θέτει την ανοχή για την υλοποίηση του αλγορίθμου. Επικουρικά, για την επιλογή της ανοχής, υπολογίζεται και παρέχεται ως πληροφορία ο αριθμός των σημείων που δομούν το  $\epsilon$ -τμήμα πριν και μετά την απλοποίηση, ενώ η παράγωγη γραμμή οπτικοποιείται για την αξιολόγηση του αποτελέσματος.



Σχήμα 4.21: Μοντέλο γενίκευσης  $\epsilon$ -κυρτών τμημάτων.

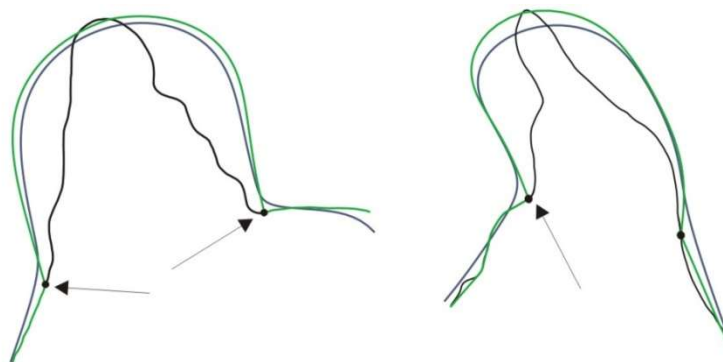
#### 4.5 Συνένωση γενικευμένων $\epsilon$ -τμημάτων - Δημιουργία γενικευμένης γραμμής

Σε ένα τελευταίο στάδιο επεξεργασίας, τα γενικευμένα  $\epsilon$ -τμήματα επανενώνονται, ώστε να συνθέσουν τη γενικευμένη εκδοχή της αρχικής γραμμής, η οποία αποτελεί και το ζητούμενο της προτεινόμενης μεθοδολογίας. Όπως αναφέρεται σε προηγούμενες ενότητες, ένας βασικός περιορισμός κατά την επεξεργασία του συνόλου των  $\epsilon$ -τμημάτων είναι η διατήρηση της αρχικής θέσης των άκρων τους. Με τον τρόπο αυτό, γίνεται εφικτή η ορθή και χωρίς τοπολογικές αστοχίες συνένωσή τους. Η διαδικασία λαμβάνει χώρα στο περιβάλλον του λογισμικού πακέτου ArcGIS®, όπως φαίνεται στο Σχήμα 4.22.



Σχήμα 4.22: Μοντέλο συνένωσης γενικευμένων  $\epsilon$ -τμημάτων.

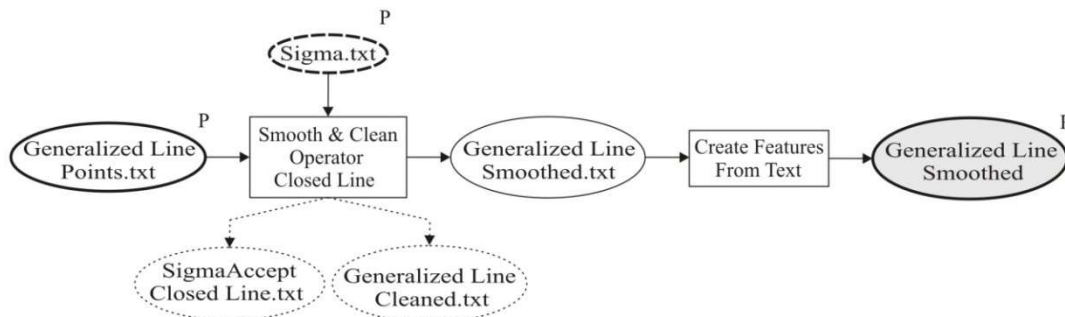
Η παράγωγη γραμμή (γραμμή πράσινης απόχρωσης στο Σχήμα 4.23) είναι πιθανό να παρουσιάζει ασυνέχειες σε σημεία σύνδεσης των  $\epsilon$ -τμημάτων ή ανεπαίσθητες αιχμές και σπασίματα κατά μήκος της. Το γεγονός αυτό οφείλεται στους σύνθετους, αλλά και διαφορετικούς μετασχηματισμούς που έχουν υποστεί τα  $\epsilon$ -τμήματα κατά τη γενίκευσή τους. Για τη διόρθωση τέτοιου είδους ατελειών, η γραμμή εξομαλύνεται σε μικρό βαθμό, με την εφαρμογή του τελεστή Gauss (βλ. κεφ. 3.2.2), ώστε να διαμορφωθεί η τελική, γενικευμένη εκδοχή της (γραμμή μπλε απόχρωσης στο Σχήμα 4.23).



Σχήμα 4.23: Παράδειγμα διόρθωσης ασυνεχειών ή σπασιμάτων γενικευμένης γραμμής.

Παράλληλα με τη διόρθωση των ασυνεχειών, η εξομάλυνση σε αυτό το επίπεδο είναι δυνατό, υπό προϋποθέσεις, να δώσει λύση και σε προβλήματα ευαναγνωσιμότητας. Όπως φαίνεται στο Σχήμα 4.23, εξομαλύνοντας το γενικευμένο  $\epsilon$ -τμήμα η απόσταση μεταξύ των δύο άκρων του αυξάνει. Το γεγονός μπορεί να δημιουργήσει τις συνθήκες εισχώρησης του θεωρητικού δίσκου του Perkal εντός αυτού. Η λύση θεωρείται αποδεκτή υπό τον όρο ότι η κάθετη απόσταση της κορυφής ΚΣ<sub>A</sub> του αρχικού  $\epsilon$ -τμήματος από την τελική γενικευμένη γραμμή είναι μικρότερη από μία τιμή κατωφλίου (D+T). Κατ' αυτό τον τρόπο, τίθεται ένα όριο στο

βαθμό εξομάλυνσης της όλης διαδικασίας και συγχρόνως, διασφαλίζεται η διατήρηση (εντός του ορίου διακριτικής ικανότητας του ανθρώπινου οφθαλμού) χαρακτηριστικών θέσεων που διαμορφώνουν τη γραμμή. Για τον καθορισμό της τιμής κατωφλίου λαμβάνεται υπόψη το όριο διακριτικής ικανότητας  $D$  και η τιμή ανοχής  $T$  (βλ. κεφ. 2.2), ανηγμένα στην κλίμακα του χάρτη.



**Σχήμα 4.24:** Μοντέλο εξομάλυνσης γενικευμένης γραμμής.

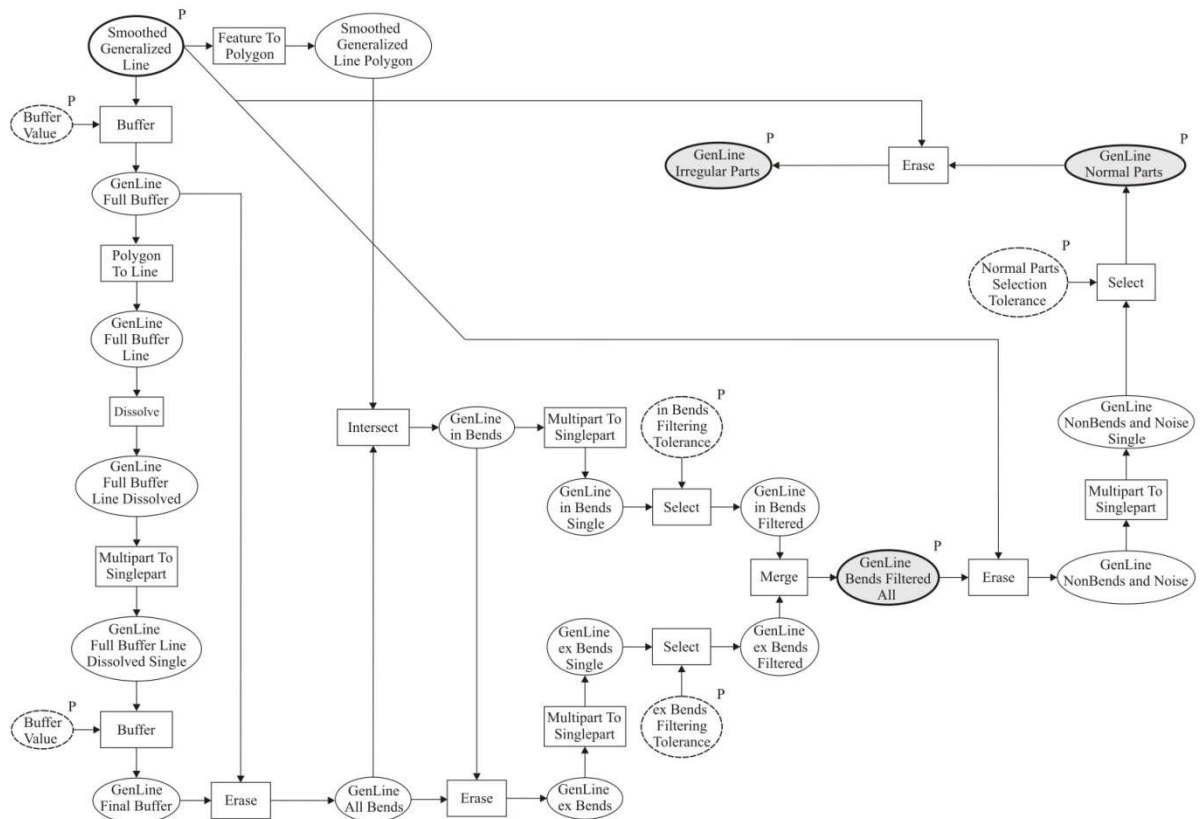
Η διαδικασία λαμβάνει χώρα στο περιβάλλον του λογισμικού πακέτου *ArcGIS®*, όπως φαίνεται στο Σχήμα 4.24, ενώ ο κώδικας υλοποίησης του τελεστή εξομάλυνσης Gauss για κλειστές γραμμές δημιουργήθηκε σε γλώσσα προγραμματισμού *Python* και παρουσιάζεται στο Παράρτημα Π3.2Κ.

#### 4.6 Έλεγχος ευαναγνωσιμότητας γενικευμένης γραμμής στην παράγωγη κλίμακα.

Μετά την ολοκλήρωση των διεργασιών γενίκευσης που περιγράφονται στις προηγούμενες ενότητες, λαμβάνει χώρα ο έλεγχος ευαναγνωσιμότητας της προκύπτουσας γραμμής. Βάσει των κανόνων οπτικής αντίληψης που έχουν ορισθεί στο κεφάλαιο 2, η γραμμή αποδίδεται ευκρινώς στην παράγωγη κλίμακα αν, θεωρητικά, ο δίσκος του Perkal διαμέτρου  $\epsilon$  κυλά κατά μήκος της χωρίς να δημιουργούνται  $\epsilon$ -μη-κυρτά τμήματα.

Η διαδικασία υλοποιείται στο περιβάλλον του λογισμικού πακέτου *ArcGIS®*, μέσω ενός μοντέλου (Σχήμα 4.25), αντίστοιχου με εκείνο της μεθόδου κατάτμησης γραμμικών οντοτήτων (βλ. κεφ. 2.3.2.). Στο μοντέλο εισάγεται η γενικευμένη γραμμή και ο χρήστης θέτει τις παραμέτρους εφαρμογής, δηλαδή: α) το πλάτος της 'ζώνης επιρροής' (*Buffer Value*), β) το ελάχιστο εμβαδό πολυγώνου (*in/exBends Filtering Tolerance*) για τη διεργασία φιλτραρίσματος των  $\epsilon$ -μη-κυρτών τμημάτων και γ) το ελάχιστο μήκος αναπτύγματος για τη διεργασία συγχώνευσης των  $\epsilon$ -τμημάτων (*Normal Parts Selection Tolerance*). Με την περάτωση της διαδικασίας είναι δυνατό να δημιουργηθούν  $\epsilon$ -κυρτά (*Gen Line Normal Parts*) ή/και  $\epsilon$ -μη-κυρτά τμήματα (*Gen Line Irregular Parts*). Η διασφάλιση της ευαναγνωσιμότητας της γενικευμένης γραμμής επιβεβαιώνεται εάν προκύψει μόνο ένα  $\epsilon$ -κυρτό τμήμα και κανένα  $\epsilon$ -μη-κυρτό. Αντίθετα, εάν παραχθούν ένα ή περισσότερα  $\epsilon$ -μη-κυρτά τμήματα, συμπεραίνεται πως το πρόβλημα οπτικής ευαναγνωσιμότητας δεν έχει επιλυθεί στις θέσεις που αυτά εντοπίζονται. Για τις συγκεκριμένες θέσεις, ο χρήστης

καλείται να εφαρμόσει εκ νέου τους τελεστές γενίκευσης, με διαφορετικές τιμές των παραμέτρων υλοποίησης, ώστε να καταλήξει στο επιθυμητό αποτέλεσμα.



Σχήμα 4.25: Μοντέλο ελέγχου ευαναγνωσιμότητας γενικευμένης γραμμής στην παράγωγη κλίμακα.

## ΚΕΦΑΛΑΙΟ 5: ΠΙΛΟΤΙΚΗ ΕΦΑΡΜΟΓΗ

### 5.1. Το αρχείο αναφοράς

Οι τεχνικές εντοπισμού κρίσιμων σημείων μέσω του δείκτη LR και κατάτμησης - γενίκευσης χαρτογραφικών γραμμών, εφαρμόζεται πιλοτικά στην ακτογραμμή της νήσου Περιστεράς, η οποία ανήκει στο σύμπλεγμα των Σποράδων. Πρόκειται για μια γραμμή που αναπαριστά μια φυσική χωρική οντότητα και χαρακτηρίζεται από υψηλό βαθμό πολυπλοκότητας. Συνδυάζει απομονωμένες κοιλώσεις και κυρτώσεις, αλληλουχίες αυτών και θέσεις όπου είναι δυνατό να εμφανισθεί σύγκλιση τμημάτων, γεγονός που την καθιστά αντιπροσωπευτική για τον έλεγχο των αποτελεσμάτων της συγκεκριμένης πρότασης.

Η ακτογραμμή έχει δημιουργηθεί μετά από ψηφιοποίηση αναλογικού χάρτη της ΓΥΣ, σαρωμένου στα 600dpi, κλίμακας 1:50.000. Στην περίπτωση της εφαρμογής του δείκτη LR το μέσο βήμα ψηφιοποίησης είναι 0,3mm στον χάρτη (5m στο έδαφος), ενώ σε αυτή της κατάτμησης - γενίκευσης είναι 0,1mm στον χάρτη (5m στο έδαφος). Τα τελικά αρχεία αναφοράς έχουν προέλθει μετά από καθαρισμό, ώστε να απαλειφθούν σφάλματα ψηφιοποίησης (επαναχαράξεις, κόμβοι, αιχμές κτλ.) και εξομάλυνση (ανεπαίσθητη μετατόπιση των σημείων, ώστε η γραμμή να αποκτήσει μία πιο ομαλή και 'φυσική' μορφή, χωρίς να μεταβληθεί η γεωγραφική θέση των σημείων και ο χαρακτήρας της, όπως προτείνεται από τον Jenks (1981). Τα δεδομένα διορθώθηκαν και εξομαλύνθηκαν με παραμέτρους κοντά στην τιμή ανοχής της ψηφιοποίησης της αναλογικής γραμμής.

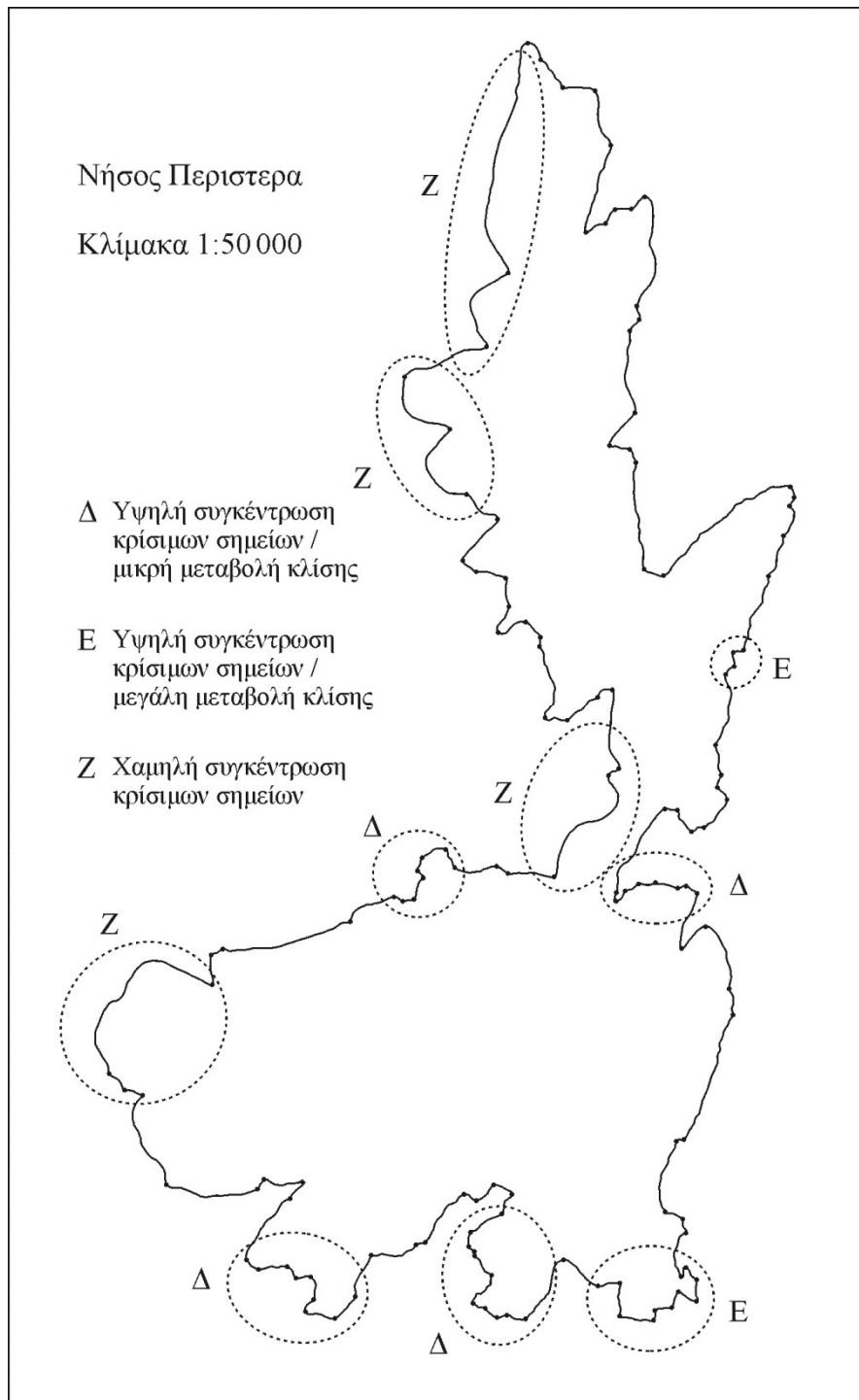
### 5.2. Πιλοτική εφαρμογή του δείκτη αναλογίας μήκους LR

#### 5.2.1. Εντοπισμός κρίσιμων σημείων μέσω του δείκτη ALR

Ο δείκτης LR εφαρμόσθηκε στην ακτογραμμή της νήσου Περιστεράς. Ο εντοπισμός των κρίσιμων σημείων της ακτογραμμής πραγματοποιήθηκε με τη εφαρμογή του δείκτη ALR. Εφαρμόσθηκαν ακτίνες  $R=15m$ ,  $R=30m$ ,  $R=45m$ ,  $R=60m$  στο έδαφος (0,3mm, 0,6mm, 0,9mm και 1,2mm στον χάρτη, αντίστοιχα) και κατώτερο όριο του δείκτη LR για την επιλογή κρίσιμου σημείου ορίστηκε τιμή 1,04. Τελικά, με την εφαρμογή της μεθόδου εντοπίζονται 135 κρίσιμα σημεία (5,6% των αρχικών). Στο Σχήμα 5.1 παρουσιάζεται η ακτογραμμή, καθώς και τα 135 κρίσιμα σημεία που εντοπίστηκαν.

Αξιολογώντας τη θέση των 135 κρίσιμων σημείων, παρατηρείται πως ικανοποιούν την βασική αρχή της 'διατήρησης της μορφής και του χαρακτήρα της γραμμής'. Θέσεις έντονων αλλαγών κλίσης ή σπασίματα της γραμμής -σημαντικά για την αποτελεσματική αναπαράσταση του βασικού σχήματος της γραμμής- εντοπίζονται με επιτυχία. Επίσης, τα αποτελέσματα της εφαρμογής συμβαδίζουν με την λογική της ομαδοποίησης των τιμών του δείκτη LR ανάλογα με το απεικονιζόμενο σχήμα. Γενικά, από τα 135 εντοπιζόμενα κρίσιμα σημεία, τα 87 ανήκουν στην ομάδα Α, τα 28 στην ομάδα Β και τα 20 στην ομάδα Γ.





Σχήμα 5.1: Ακτογραμμή της νήσου Περιστεράς και τα 135 κρίσιμα σημεία.

Μια διεξοδικότερη αξιολόγηση των 135 κρίσιμων σημείων σε τοπικό επίπεδο, οδηγεί σε διάφορα σχόλια, τα οποία κυρίως έχουν να κάνουν με όσα τμήματα της γραμμής παρουσιάζεται πυκνότερη ή πιο αραιή από την αναμενόμενη συγκέντρωσή τους. Υπάρχουν θέσεις, όπου παρουσιάζονται υψηλές συγκεντρώσεις κρίσιμων σημείων, όπως αυτές που υποδεικνύονται με τα γράμματα Δ και Ε στο Σχήμα 5.1. Αυτό καταδεικνύει πως η μέθοδος του δείκτη LR οδηγεί σε υπερεκτίμηση των κρίσιμων σημείων σε περιοχές υψηλής πολυπλοκότητας της γραμμής, γεγονός που



μπορεί να θεωρηθεί ως μειονέκτημα, αφού η διατήρηση της λεπτομέρειας δεν συμβαδίζει πάντα την έννοια των κρίσιμων σημείων. Οι τιμές του δείκτη ALR που αντιστοιχούν στις περιοχές αυτές δεν κατατάσσονται σε μια συγκεκριμένη ομάδα του δείκτη LR, αλλά συμβαδίζουν με τη μορφή της γραμμής στην εκάστοτε περιοχή. Σε περιοχές ομαλής μεταβολής της καμπυλότητας (υποδεικνύονται με το γράμμα Δ στο Σχήμα 5.1), οι τιμές του δείκτη ALR ανήκουν κατά κύριο λόγο στην ομάδα Α και δευτερευόντως στην ομάδα Β. Αντίθετα, σε περιοχές απότομων μεταβολών της καμπυλότητας (υποδεικνύονται με το γράμμα Ε στο Σχήμα 5.1), οι τιμές του δείκτη ALR ανήκουν στην ομάδα Β ή και στην ομάδα Γ. Η ίδια συμπεριφορά παρατηρείται, επίσης, και κατά την ανάλυση των αποτελεσμάτων της εφαρμογής της μεθόδου LR στην γραμμή του Ποταμού Mancos (γραμμή υψηλής πολυπλοκότητας), όπου αρκετές κορυφές, οι οποίες συνδέονται με τοπικά μέγιστα του δείκτη LR, δεν είχαν επιλεγεί ως κρίσιμα στην εμπειρική μελέτη της Marino. Αυτή η ανεπάρκεια μπορεί να εξηγηθεί από το γεγονός ότι το ανάπτυγμα L, και κατ' επέκταση ο δείκτης LR, εκτός της καμπυλότητας, αυξάνει ανάλογα και με την πολυπλοκότητα της γραμμής. Φαίνεται ότι ο δείκτης LR είναι περισσότερο ευαίσθητος σε αυτές τις περιπτώσεις.

Σε σύγκριση με παλαιότερες εφαρμογές του δείκτη LLR στη γραμμή, μέσω του δείκτη ALR εντοπίζονται σαφώς λιγότερα σημεία στις θέσεις υψηλής πολυπλοκότητας. Κάτι τέτοιο είναι λογικό, αφού εξετάζεται μια ευρύτερη περιοχή στην εκτίμηση της καμπυλότητας. Οι τιμές του ALR που αντιστοιχούν στις θέσεις πολυπλοκότητας εμφανίζονται ομαλοποιημένες (φθίνουσες ή αύξουσες προς σημεία μεγαλύτερης μεταβολής κλίσης ή κάτω από το όριο επιλογής). Η ιδιότητα αυτή του ALR ανοίγει τις προοπτικές για περαιτέρω διερεύνηση των περιοχών εξέτασης (δηλαδή των ακτινών εφαρμογής), ώστε να περιορισθεί ή να απαλειφθεί το φαινόμενο. Η δυσκολία έγκειται στην εύρεση κατάλληλων ακτινών, οι οποίες να μην επιδρούν αρνητικά στην εκτίμηση της καμπυλότητας των υπολοίπων θέσεων της γραμμής.

Μία δεύτερη παρατήρηση αφορά σε περιοχές της γραμμής με πιο αραιή, από το αναμενόμενο, συγκέντρωση κρίσιμων σημείων (υποδεικνύονται με το γράμμα Ζ στο Σχήμα 5.1). Οι περιπτώσεις αυτές έχουν, κυρίως, να κάνουν με διευρυμένες, επιμήκεις χερσονήσους με σταθερή αλλαγή κλίσης, οι οποίες δεν εντοπίζονται. Το γεγονός αυτό, εν μέρει, αποτελεί μειονέκτημα της μεθόδου, αφού αυτές οι καμπύλες θα μπορούσαν να θεωρηθούν απαραίτητες για τη διατήρηση του βασικού σχήματος της γραμμής. Οι τιμές του δείκτη LR που αντιστοιχούν στις κορυφές τους είναι σταθερές (ή χαρακτηρίζονται από πολύ μικρές διακυμάνσεις) και, ταυτόχρονα είναι μικρότερες από την τιμή του ορίου επιλογής (κυμαίνονται από τιμές του δείκτη μεταξύ 1,025 έως 1,039). Το πρόβλημα θα μπορούσε να επιλυθεί με μείωση του ορίου επιλογής. Χρειάζεται, όμως, ιδιαίτερη προσοχή και ανάλυση, αφού κάτι τέτοιο θα σήμαινε διατήρηση αριθμού κορυφών ως κρίσιμες πέραν του επιθυμητού. Συγχρόνως, λόγω των παραπάνω, οι καμπύλες αποδίδονται με τον ελάχιστο αριθμό κορυφών (στα ενδιάμεσα σημεία μεταξύ των κρίσιμων σημείων χαρακτηρίζονται από χαμηλές τιμές του δείκτη LR), με αποτέλεσμα να έχουν την μορφή μεγάλων σφηνών, οι οποίες οπτικά δεν είναι απόλυτα αποδεκτές. Το φαινόμενο αυτό είχε παρατηρηθεί και κατά την εφαρμογή του δείκτη στην ημιτονοειδούς μορφής γραμμή του Ποταμού Shenandoah. Όμως, σε μια μεθοδολογία εντοπισμού κρίσιμων σημείων το συγκεκριμένο πρόβλημα

περιορίζεται, γεγονός που φαίνεται και από την απόλυτη ταύτιση των σημείων που εντοπίζονται μέσω του δείκτη LR, με τα αντίστοιχα της εμπειρικής μελέτης της Marino και του Thara. Ο σχεδιασμός του δείκτη LR είναι δομημένος με τέτοιο τρόπο, ώστε να εντοπίζονται οι θέσεις των χαρακτηριστικών αλλαγών της κλίσης, στις οποίες, σύμφωνα με τον Attneave (1954), εμφανίζονται τα κρίσιμα σημεία. Κατά συνέπεια, αυτές οι χαρακτηριστικές μορφές που παρουσιάζονται στην ακτογραμμή της νήσου Περιστέρας δεν περιέχουν κρίσιμα σημεία.

### 5.2.2 Σύγκριση δύο αλγορίθμων απλοποίησης γραμμών με μέτρο τα κρίσιμα σημεία που έχουν εντοπισθεί μέσω του δείκτη ALR

Η ακτογραμμή της νήσου Περιστέρας γενικεύεται μέσω δύο αλγορίθμων απλοποίησης (τους 'pointremove' και 'bendsimplify') χρησιμοποιώντας το λογισμικό ArcGIS. Ο αλγόριθμος απλοποίησης 'pointremove' βασίζεται στον αλγόριθμο που επινοήθηκε από τους Douglas και Peucker (1973), με ορισμένες μικρές τροποποιήσεις απαραίτητες για την προσαρμογή του στο περιβάλλον του συγκεκριμένου λογισμικού. Οι Douglas και Peucker (1973), αρχικά, δημιούργησαν τον αλγόριθμο για το φιλτράρισμα των πρωτογενών ψηφιακών δεδομένων, δηλαδή την απαλοιφή των περιττών σημείων (λεπτομερειών) που 'εισάγονται' κατά τη διαδικασία ψηφιοποίησης των αναλογικών γραμμών, ώστε να παράγονται οι τελικές απεικονίσεις των γραμμών. Στη συνέχεια, χρησιμοποιήθηκε από τους χαρτογράφους ως αλγόριθμος απλοποίησης. Ο αλγόριθμος είναι δομημένος με τέτοιο τρόπο, ώστε να διατηρούνται τα σημεία που βρίσκονται στα τμήματα αλλαγής της κλίσης της γραμμής και απαλείφονται τα υπόλοιπα. Ο αλγόριθμος απλοποίησης 'bendsimplify' περιγράφεται στο Κεφάλαιο 3.6.

Στόχος του συγκεκριμένου μέρους της έρευνας είναι η σύγκριση των δύο αλγορίθμων απλοποίησης με μέτρο τα κρίσιμα σημεία που έχουν εντοπισθεί από την εφαρμογή του δείκτη LR στην αρχική ακτογραμμή. Βάσει αυτών, αναλύονται οι διατηρούμενες κορυφές των απλοποιημένων γραμμών, ως προς την ποσότητα, αλλά και ως προς την ποιότητά τους. Τα σενάρια της γενίκευσης περιλαμβάνουν την απλοποίηση της ακτογραμμής σε πέντε διαφορετικά επίπεδα, μέσω του εκάστοτε αλγορίθμου. Οι παράγωγες ακτογραμμές παρουσιάζονται στις ονομαστικές κλίμακες 1:100.000, 1:250.000, 1:1.000.000 και 1:2.000.000 αντίστοιχα. Σε κάθε περίπτωση, ο αριθμός των διατηρούμενων κορυφών καθορίζεται σύμφωνα με τις 'Αρχές της Επιλογής' (Törpfer and Pillewizer 1966), η οποία για την περίπτωση των ακτογραμμών (γραμμικά σύμβολα ίσου πάχους) εκφράζεται από τη σχέση (Jones and Abraham 1987):  $n = n_0 \frac{S}{S_0}$ , όπου  $S_0$  και  $n_0$  η κλίμακα και ο αριθμός κορυφών του αρχικού χάρτη και  $S$  και  $n$  του παράγωγου, αντίστοιχα. Ο αριθμός των σημείων που απαιτούνται για να αναπαρασταθεί μια γενικευμένη γραμμή δεν μεταβάλλεται πάντα γραμμικά ως προς την κλίμακα. Εντούτοις, όταν η γραμμή και η αναλογία αλλαγής της κλίμακας είναι συγκεκριμένες, οι 'Αρχές της Επιλογής' μπορούν να χρησιμοποιηθούν ως γενικός αποδεκτός χαρτογραφικός κανόνας για να υπολογιστούν τα χαρακτηριστικά γνωρίσματα της παράγωγης γραμμής. Στην εφαρμογή, χρησιμοποιούνται οι 'Αρχές της Επιλογής' με σκοπό, αποκλειστικά, να προσδιοριστούν οι κατάλληλες τιμές ανοχών για την εφαρμογή των αλγορίθμων. Οι ανοχές καθορίζονται με τρόπο, ώστε να εξισωθεί ο αριθμός των διατηρούμενων

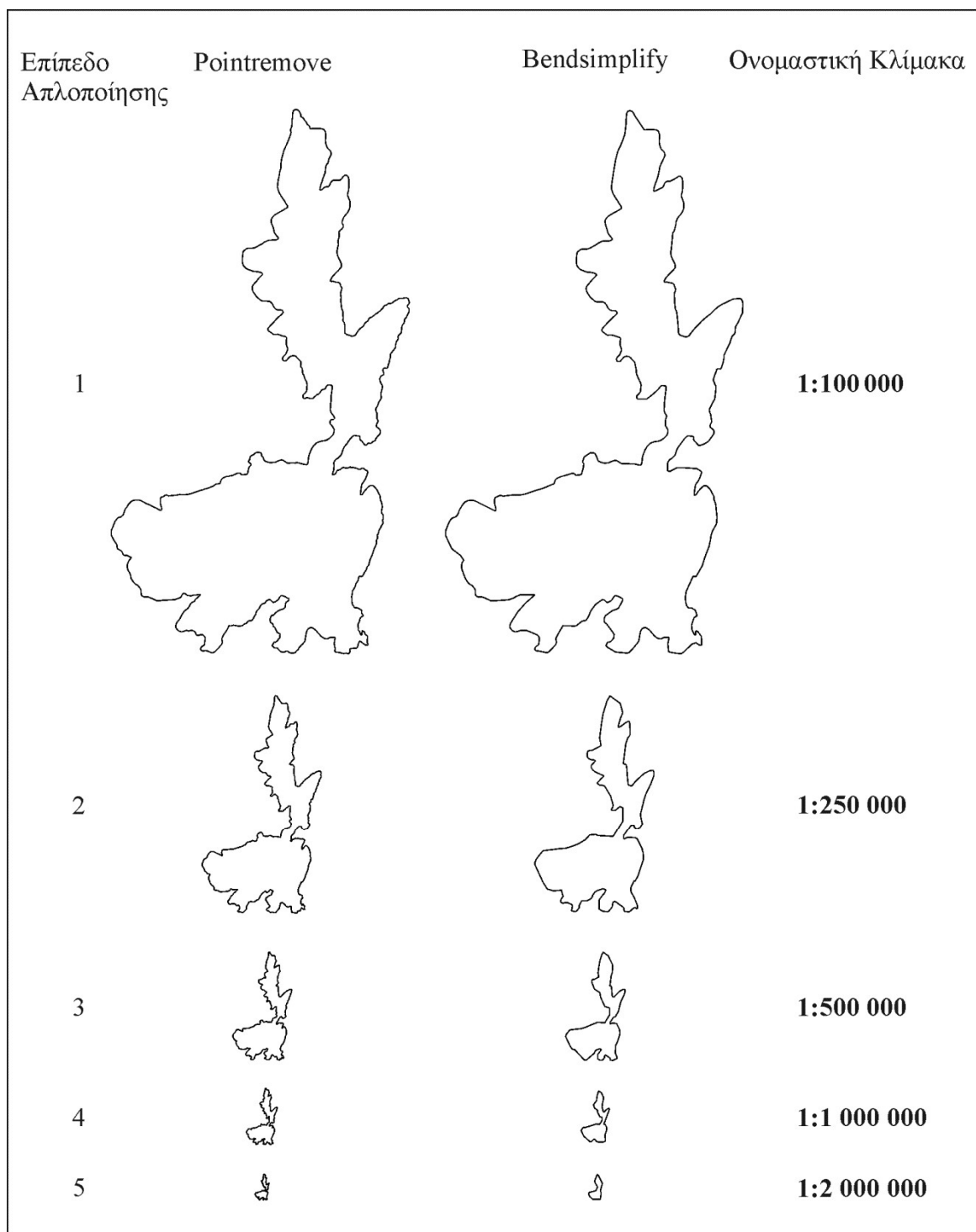
κορυφών σε κάθε επίπεδο απλοποίησης και για τους δύο αλγορίθμους. Στον Πίνακα 5.1 παρουσιάζονται οι παράμετροι των πέντε επιπέδων απλοποίησης της ακτογραμμής και στο Σχήμα 5.2 απεικονίζονται οι απλοποιημένες ακτογραμμές.

Επίπεδο	Ονομαστική κλίμακα χάρτη	Διατηρούμενες κορυφές
1	1:100.000	1207 (50%)
2	1:250.000	485 (20%)
3	1:500.000	246 (10%)
4	1:1.000.000	129 (5%)
5	1:2.000.000	61 (2.5%)

**Πίνακας 5.1.** Διατηρούμενες κορυφές για τα πέντε σενάρια απλοποίησης μετά από την εφαρμογή των 'Αρχών της Επιλογής'.

Επίπεδο Απλοποίησης	Διατηρούμενα Κρίσιμα Σημεία	
	Pointremove	Bendsimplify
1	135 (100%)	113 (84%)
2	130 (96%)	66 (49%)
3	114 (84%)	39 (29%)
4	73 (54%)	20 (15%)
5	35 (26%)	8 (6%)

**Πίνακας 5.2:** Διατηρούμενα κρίσιμα σημεία στα πέντε σενάρια απλοποίησης.



Σχήμα 5.2: Απλοποιημένες ακτογραμμές της νήσου Περιστεράς σε πέντε παράγωγες κλίμακες.

Στον Πίνακα 5.2 παρουσιάζεται ο αριθμός των κοινών σημείων που μοιράζεται κάθε επίπεδο απλοποίησης με το σύνολο των εντοπισμένων, μέσω του δείκτη LR, κρίσιμων σημείων στην αρχική γραμμή κλίμακας 1:50.000. Στο πρώτο επίπεδο απλοποίησης, όπου ο αριθμός των σημείων που διατηρούνται είναι μεγάλος (50% των αρχικών), παρατηρείται πως ο αλγόριθμος 'pointremove' διατηρεί το απόλυτο των κρίσιμων σημείων, ενώ ο 'bendsimplify' το 84% αυτών. Με την αύξηση του επιπέδου απλοποίησης, τα ποσοστά διατήρησης κρίσιμων σημείων από τον αλγόριθμο 'bendsimplify' μειώνονται σε μεγάλο βαθμό. Ήδη, από το δεύτερο επίπεδο απλοποίησης, ο 'bendsimplify' διατηρεί λιγότερα από τα μισά κρίσιμα σημεία (49%). Ενδιαφέρον παρουσιάζει το τέταρτο επίπεδο απλοποίησης, όπου (συμπωματικά) ο αριθμός των σημείων που διαμορφώνουν τις απλοποιημένες γραμμές (129) είναι περίπου ίσος με αυτόν των κρίσιμων σημείων (135). Παρατηρείται πως ο αλγόριθμος 'bendsimplify' διατηρεί μόλις των 15% των κρίσιμων σημείων. Στο πέμπτο επίπεδο, όπου ο αριθμός των σημείων που διαμορφώνουν τις απλοποιημένες γραμμές είναι μικρότερος από αυτών των κρίσιμων σημείων (και επομένως είναι πρακτικά αδύνατο να υπάρξει απόλυτη σύμπτωση των αποτελεσμάτων) ο 'bendsimplify' διατηρεί ελάχιστα κρίσιμα σημεία. Αντίθετα, ο αλγόριθμος 'pointremove' διατηρεί σχεδόν το απόλυτο των κρίσιμων σημείων και στο δεύτερο επίπεδο απλοποίησης, ενώ στο τρίτο το ποσοστό υπερβαίνει το 80%. Στο τέταρτο επίπεδο, διατηρεί το 54% των κρίσιμων σημείων. Δηλαδή, σε ένα επίπεδο διατήρησης, σχεδόν, ίσου αριθμού σημείων, πάνω από τα μισά που εντοπίζονται από τον αλγόριθμο 'pointremove' είναι κρίσιμα. Ακόμη και στο πέμπτο επίπεδο, πάνω από τα μισά σημεία που διατηρούνται από τον αλγόριθμο 'pointremove' (35 από τα 61) θεωρούνται κρίσιμα.

Παρόμοια αποτελέσματα συναντώνται και στην έρευνα της White (1985), όπου αξιολογήθηκαν διάφοροι αλγόριθμοι απλοποίησης γραμμών. Η αξιολόγηση των δύο αλγορίθμων σε απόλυτους αριθμούς, με μια πρώτη ματιά, υποδηλώνει ότι ο αλγόριθμος 'bendsimplify' σε σχέση με τον 'pointremove' παρουσιάζει συγκεκριμένα μειονεκτήματα, δεδομένου ότι ένας σημαντικός αριθμός των σημείων που διατηρούνται από την εφαρμογή του δεν συμπίπτει με τα προκαθορισμένα κρίσιμα σημεία. Ωστόσο, ποια κρίσιμα σημεία απαλείφει ο αλγόριθμος 'bendsimplify'; Επιπλέον, είναι αυτά τα σημεία σημαντικής σπουδαιότητας για τη διατήρηση της βασικής μορφής της ακτογραμμής;

Το σύνολο των σημείων που διατηρούνται από τους δύο αλγορίθμους απλοποίησης και είναι, συγχρόνως, κρίσιμα σημεία της αρχικής γραμμής ταξινομούνται στις τρεις ομάδες (Α, Β και Γ) των τιμών του δείκτη LR. Τα αποτελέσματα, για τα πέντε σενάρια απλοποίησης, παρουσιάζονται στον Πίνακα 5.3.

Επίπεδο Απλοποίησης	Ομάδα Α		Ομάδα Β		Ομάδα Γ	
	P*	B**	P*	B**	P*	B**
1	87	75	28	20	20	18
2	83	46	28	13	19	7
3	69	29	26	7	19	3
4	37	14	17	5	19	1
5	15	6	6	1	14	1
<b>Αρχική Γραμμή</b>	87		28		20	

\* Pointremove, \*\* Bendsimplify

**Πίνακας 5.3:** Διατηρούμενα κρίσιμα σημεία ταξινομημένα στις τρεις ομάδες τιμών του δείκτη LR μετά από τα πέντε σενάρια απλοποίησης.

Παρατηρώντας τον Πίνακα 5.3, γίνεται προφανές ότι η πλειοψηφία των σημείων που επιλέγονται από τον αλγόριθμο 'bendsimplify' αντιστοιχεί στην ομάδα Α, ενώ ο αριθμός των κορυφών που αντιστοιχούν σε μεσαίες ή υψηλές τιμές του δείκτη LR (ομάδες Β και Γ) είναι μικρός. Ουσιαστικά, από το δεύτερο επίπεδο απλοποίησης και μετά, ο αλγόριθμος 'bendsimplify' δεν διατηρεί σημεία υψηλών τιμών του δείκτη LR. Σε κάθε περίπτωση, περισσότερο από το 70% των διατηρούμενων σημείων ανήκουν στην ομάδα Α. Στην πραγματικότητα, η πλειοψηφία των τιμών της ομάδας Α κυμαίνεται σε αρκετά χαμηλά επίπεδα (από 1,04 μέχρι 1,08). Πρακτικά, αυτό υποδηλώνει ότι διατηρούνται κορυφές που αναπαριστούν θέσεις ομαλών αλλαγών της κλίσης (αυτές οι περιοχές καθορίζουν, συνήθως, τα άκρα και τις αιχμές των καμπυλών με μεγάλες διακυμάνσεις), που είναι απαραίτητες για τη διατήρηση της μορφής της γραμμής. Οι απλοποιημένες γραμμές παρουσιάζονται, σε μεγάλο βαθμό, εξομαλυμένες. Στα δύο πρώτα επίπεδα απλοποίησης, ο αλγόριθμος 'bendsimplify' διατηρεί όλες τις μεγάλες, ανοιχτές καμπύλες (Σχήμα 5.2 και περιοχές που υποδεικνύονται με το γράμμα F στο Σχήμα 5.1). Παράλληλα, διατηρούνται οι περιοχές μεγάλης πολυπλοκότητας που αντιστοιχούν σε μικρές τιμές του δείκτη LR (Σχήμα 5.2 και περιοχές που υποδεικνύονται με το γράμμα D στο Σχήμα 5.1), ενώ αυτές που αντιστοιχούν σε μεγάλες τιμές του δείκτη LR (Σχήμα 5.2 και περιοχές που υποδεικνύονται με το γράμμα E στο Σχήμα 5.1), δεν διατηρούνται. Στα επόμενα επίπεδα απλοποίησης, οι θέσεις έντονης πολυπλοκότητας απαλείφονται. Συγχρόνως, όμως, διαμορφώνονται μόνο οι πολύ μεγάλες, ανοιχτές καμπύλες (μάλιστα ορισμένες συνενώνονται με τις γειτονικές τους). Ορισμένοι απότομοι κόλποι και χερσόνησοι δεν διατηρούνται. Στα δύο τελευταία επίπεδα απλοποίησης, το τελικό σχήμα είναι στρογγυλοποιημένο (ομοιάζει με περίγραμμα γύρω από τα ακραία σημεία της ακτογραμμής).

Ο αλγόριθμος 'pointremove' διατηρεί την πλειοψηφία των κρίσιμων σημείων στα δύο πρώτα επίπεδα απλοποίησης. Επιπλέον, οι παράγωγες γραμμές διατηρούν την ίδια κατανομή των κρίσιμων σημείων στις τρεις ομάδες με την αρχική γραμμή. Στον Πίνακα 4 φαίνεται πως το μεγαλύτερο μέρος των διατηρημένων κορυφών ανήκει στην ομάδα Α. Αυτό συνεπάγεται τη διατήρηση των χαρακτηριστικών κλίσεων της υπό εξέταση γραμμής. Με την αύξηση του επιπέδου απλοποίησης, όπου ο αριθμός των κρίσιμων σημείων μειώνεται, παρατηρείται πως το ποσοστό αυτών που ανήκουν στην ομάδα Α μειώνεται, ενώ, αντίστοιχα, αυξάνει το ποσοστό των

σημείων που ανήκουν στις ομάδες Β και Γ. Αυτό δείχνει πως δεν διατηρούνται θέσεις ομαλών κλίσεων, σε αντίθεση με τις μεγάλες κλίσεις της γραμμής. Ο αλγόριθμος 'rointremove' διατηρεί και στα πέντε επίπεδα απλοποίησης τις μεγάλες, ευρείες καμπύλες (Σχήμα 5.2 και περιοχές που υποδεικνύονται με το γράμμα Ζ στο Σχήμα 5.1). Στα δύο τελευταία επίπεδα, όμως, αυτές αποδίδονται με τα ελάχιστα σημεία, με αποτέλεσμα η μορφή τους να είναι περισσότερο οξεία. Η πολυπλοκότητα της γραμμής στα δύο πρώτα επίπεδα απλοποίησης διατηρείται στο απόλυτο. Στο τρίτο επίπεδο απλοποίησης, τα σημεία αυτών με πολύ μικρές τιμές του δείκτη ALR απαλείφονται, στο τέταρτο εντοπίζονται μόνο αυτές που ανήκουν στην ομάδα Γ (και μερικές της ομάδας Β), ενώ στο τελευταίο δεν εντοπίζονται.

Συγκρίνοντας τους δύο αλγορίθμους απλοποίησης γραμμών, βάσει των κρίσιμων σημείων που εντοπίζονται από το δείκτη LR, παρατηρούνται αρκετές διαφορές. Αυτό προσφέρει έδαφος για να εξεταστούν οι αποκλίσεις μεταξύ των δύο αλγορίθμων όσον αφορά τη δομή τους. Επιπλέον, με γνώση των ιδιοτήτων των κρίσιμων σημείων ολοκληρώνεται μια ποσοτική αξιολόγηση των διαφοροποιήσεων που παρουσιάζονται. Τα κρίσιμα σημεία που διατηρούνται από τους δύο αλγορίθμους σε κάθε επίπεδο απλοποίησης, και η ταξινόμησή τους στις τρεις ομάδες τιμών του δείκτη LR, μπορούν να μας επιτρέψουν να προβούμε σε μια ποσοτική ανάλυση αυτού που γίνεται αντιληπτό από το ανθρώπινο μάτι.

Η φιλοσοφία της δομής του αλγορίθμου 'rointremove' είναι αρκετά κοντά σε αυτή του δείκτη LR και έτσι, ως ένα βαθμό, τα αποτελέσματα συγκλίνουν. Αντίθετα, ο αλγόριθμος 'bendsimplify', έχοντας σαφώς διαφορετική δομή (δημιουργία και εξέταση κυρτώσεων της γραμμής) παρουσιάζει σημαντικές αποκλίσεις. Η διατήρηση των θέσεων υψηλής πολυπλοκότητας της γραμμής από τον αλγόριθμο 'rointremove' μπορεί να θεωρηθεί μειονέκτημα του αλγορίθμου, ειδικά στα υψηλά επίπεδα απλοποίησης. Ακόμη και η διατήρηση των μεγάλων, ανοιχτών καμπυλών σε μικρές κλίμακες θα μπορούσε να θεωρηθεί σαν διατήρηση της λεπτομέρειας. Παράλληλα, η διατήρηση σημείων μεγάλης τιμής του δείκτη LR από τον 'rointremove' δημιουργεί μια πιο 'οξεία' μορφή της τελικής γραμμής, κυρίως στα μεγάλα επίπεδα απλοποίησης. Αντίθετα, ο αλγόριθμος 'bendsimplify' εξαλείφει τη λεπτομέρεια, ανάλογα με το επίπεδο απλοποίησης. Επιπλέον, ελαχιστοποιεί τη πολυπλοκότητα της γραμμής, αλλά διατηρεί τη βασική μορφή της (μέγιστο της καμπυλότητάς της), αφού το μεγαλύτερο μέρος των κρίσιμων σημείων από αυτά που εντοπίζονται στην αρχική γραμμή και διατηρούνται στις παράγωγες ανήκει στην ομάδα Α.

Κατά συνέπεια, συνοψίζοντας τα αποτελέσματα της σύγκρισης των δύο αλγορίθμων, με βάση την μέθοδο εντοπισμού κρίσιμων σημείων μέσω του δείκτη LR και την ταξινόμησή τους στις τρεις ομάδες τιμών του δείκτη (Α, Β, Γ), συμπεραίνεται πως ο αλγόριθμος 'bendsimplify' μπορεί να θεωρηθεί χαρτογραφικά ως πιο κατάλληλος από τον αλγόριθμο 'rointremove', ειδικά για την απεικόνιση των γραμμών σε πολύ μικρές κλίμακες. Οι παράγωγες γραμμές με την εφαρμογή του αλγορίθμου 'bendsimplify' μπορούν να θεωρηθούν οπτικά και 'αισθητικά' εγκυρότερες.

Αξιολογώντας τα κρίσιμα σημεία που εντοπίζονται με την εφαρμογή του ALR, με βάση τα αποτελέσματα της εφαρμογής και την φιλοσοφία των δύο αλγορίθμων, συμπεραίνεται πως ένα μεγάλο μέρος τους είναι πραγματικά αναγκαίο για τη

διατήρηση της μορφής και του χαρακτήρα της γραμμής. Η διατήρηση από τον 'bendsimplify' σημείων που, κατά κύριο λόγο, ανήκουν στην ομάδα Α αναδεικνύει το γεγονός πως είναι απαραίτητα για τον προσδιορισμό ομαλών θέσεων της γραμμής. Αντίστοιχα, η διατήρηση θέσεων που ανήκουν στις ομάδες Β και Γ από τον *pointremove* αναδεικνύει το γεγονός πως 'απότομα' σπασίματα αποτελούν τα ακραία, χαρακτηριστικά σημεία του σχήματος της γραμμής. Επιβεβαιώνεται, όμως, η αρχική εκτίμηση πως ο αριθμός τους θα μπορούσε να είναι μικρότερος αν εξαλείφονταν μέρος αυτών που προσδίδουν λεπτομέρεια στη γραμμή. Επίσης, θα ήταν θετικό αν εντοπιζόνταν οι κορυφές σε όλες τις μεγάλες, ανοιχτές καμπύλες της γραμμής, αφού αυτές διατηρούνται και από τους δύο αλγορίθμους σε όλα τα επίπεδα απλοποίησης. Όπως αναφέρθηκε και παραπάνω, όμως, ο σκοπός του δείκτη LR δεν είναι η παραγωγή απλοποιημένων γραμμών, αλλά ο εντοπισμός των πιο σημαντικών τους σημείων, τα οποία αποτελούν ένα σκελετό που συνοδεύεται και από άλλα ('σιωπηλά σημεία') σε μία απλοποιημένη γραμμή. Έτσι, εκτιμάται πως η λειτουργικότητά του είναι αρκετά ικανοποιητική.

### 5.3 Πιλοτική εφαρμογή μοντέλου κατάτμησης – γενίκευσης γραμμών

#### 5.3.1 Σενάριο γενίκευσης 1:1.000.000

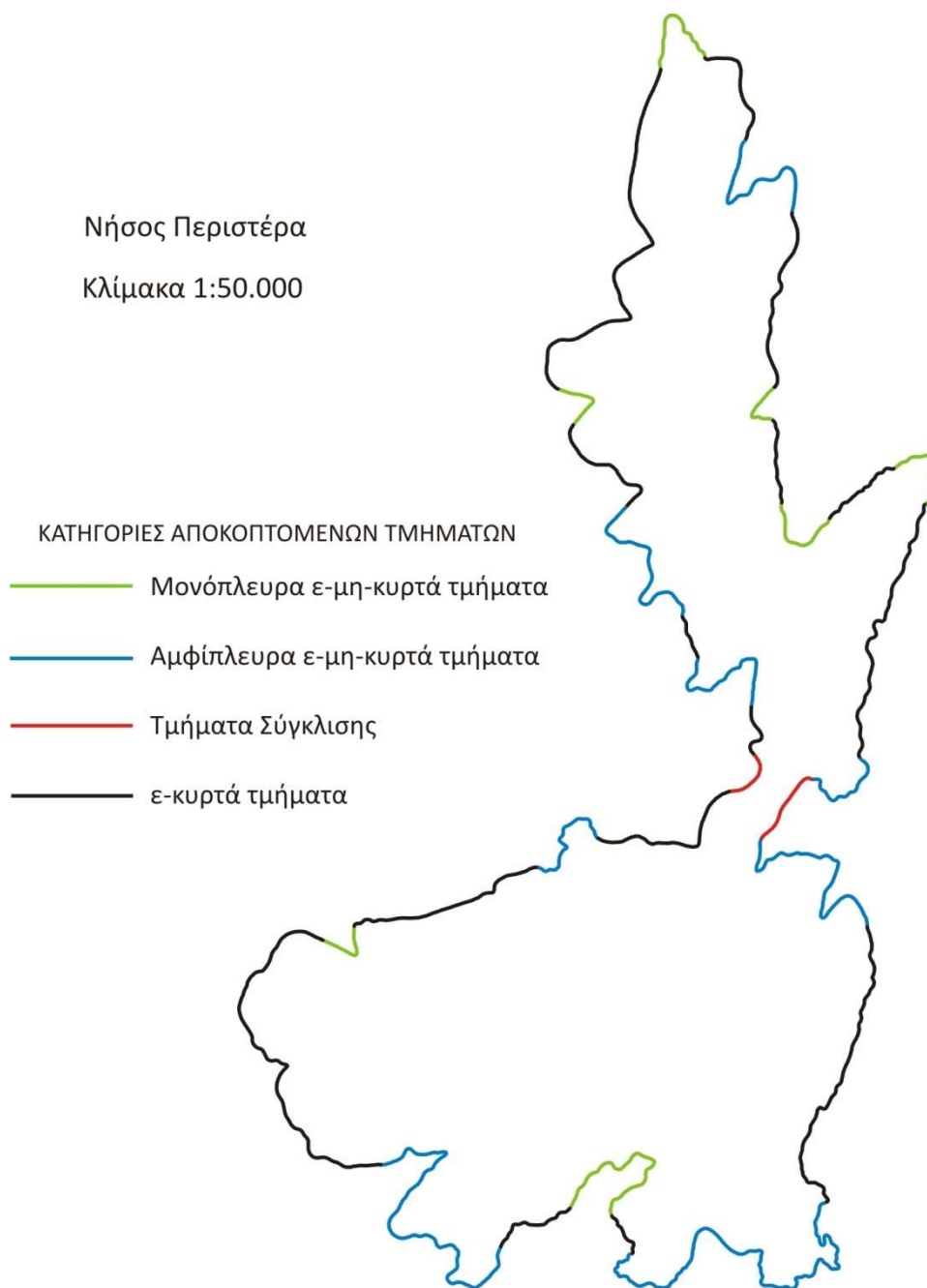
##### 5.3.1.1 Κατάτμηση γραμμής

Η τεχνική κατάτμησης (βλ. κεφ. 2.3.2, Σχήμα 2.6) υλοποιείται στην αρχική ακτογραμμή της νήσου Περιστεράς. Η διαδικασία λαμβάνει χώρα για ένα σενάριο γενίκευσης, στο οποίο ο παράγωγος χάρτης είναι κλίμακας 1:1.000.000. Η διάμετρος εφαρμογής  $\epsilon$  ορίζεται από τη σχέση  $\epsilon = W + D + T$  (βλ. κεφ. 2.2). Θεωρώντας πως σε ένα χάρτη κλίμακας 1:1.000.000, μια ακτογραμμή, συνήθως, αποδίδεται με γραμμή πάχους  $W=0,1\text{mm}$ , το όριο οπτικής αντίληψης είναι  $D=0,2\text{mm}$  (Rouleau 1984) και η τιμή ανοχής  $T=0,05\text{mm}$ , η τιμή της τίθεται  $\epsilon=0,35\text{mm}$  (στον χάρτη) ή  $\epsilon=350\text{m}$  (στο έδαφος).

Κατά την κύλιση του θεωρητικού δίσκου του Perkal εσωτερικά και εξωτερικά της ακτογραμμής αποκόπτονται 659  $\epsilon$ -μη-κυρτά τμήματα (345 και 313, αντίστοιχα). Τα τμήματα αυτά υπόκεινται σε μια διαδικασία φιλτραρίσματος, ώστε να εντοπισθούν εκείνα που, λόγω μεγέθους, δεν είναι ορατά και αποτελούν, ουσιαστικά, 'θόρυβο'. Με δεδομένο πως, σε χάρτη κλίμακας 1:1.000.000, η γραμμή αποδίδεται με σύμβολο πάχους  $W=0,1\text{mm}$ , το ελάχιστο εμβαδό που μπορεί να γίνει αντιληπτό από τον αναγνώστη είναι  $0,01\text{mm}^2$  στον χάρτη. Έτσι, προς περαιτέρω επεξεργασία λαμβάνονται 32  $\epsilon$ -μη-κυρτά τμήματα (16 εσωτερικά και 16 εξωτερικά της ακτογραμμής), των οποίων το εμβαδό είναι μεγαλύτερο ή ίσο των  $10.000\text{m}^2$  (στο έδαφος). Τα υπόλοιπα συνενώνονται με τα  $\epsilon$ -κυρτά, δημιουργώντας τελικά 21  $\epsilon$ -κυρτά τμήματα. Στη συνέχεια, εξετάζεται η απόσταση μεταξύ των φιλτραρισμένων  $\epsilon$ -μη-κυρτών τμημάτων. Για το λόγο αυτό υπολογίζεται το ανάπτυγμα των  $\epsilon$ -κυρτών τμημάτων. Στις περιπτώσεις που αυτό είναι μικρότερο ενός κατωφλίου των  $0,25\text{mm}$  στον χάρτη (ή  $250\text{m}$  στο έδαφος), τα  $\epsilon$ -μη-τμήματα συγχωνεύονται, προσαρτώντας και τα ενδιάμεσα  $\epsilon$ -κυρτά, αφού εκλαμβάνονται από τον αναγνώστη του χάρτη ως ενιαίες και όχι ως ανεξάρτητες οντότητες. Για τον καθορισμό της τιμής του κατωφλίου λαμβάνεται υπόψη το όριο διακριτικής ικανότητας  $D=0,2\text{mm}$  και μία



τιμή ανοχής  $T=0,05\text{mm}$ , στο συγκεκριμένο χάρτη. Με την ολοκλήρωση της διαδικασίας συγχώνευσης, η ακτογραμμή διαμορφώνεται από 17  $\epsilon$ -μη-κυρτά και 16  $\epsilon$ -κυρτά τμήματα.



**Σχήμα 5.3:** Αποτέλεσμα εφαρμογής του μοντέλου κατάτμησης στην ακτογραμμή της νήσου Περιστέρας για το σενάριο γενίκευσης 1:1.000.000.

Στο επόμενο στάδιο επεξεργασίας, εφαρμόζοντας το κατάλληλο μοντέλο (βλ. κεφ. 2.3.3, Σχήμα 2.7), τα  $\epsilon$ -μη-κυρτά τμήματα διαχωρίζονται σε τρεις επιμέρους κατηγορίες, ανάλογα με τη μορφή και τον τρόπο δημιουργίας τους. Έτσι, τελικά, κατά μήκος της ακτογραμμής αποκόπτονται 8 μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα

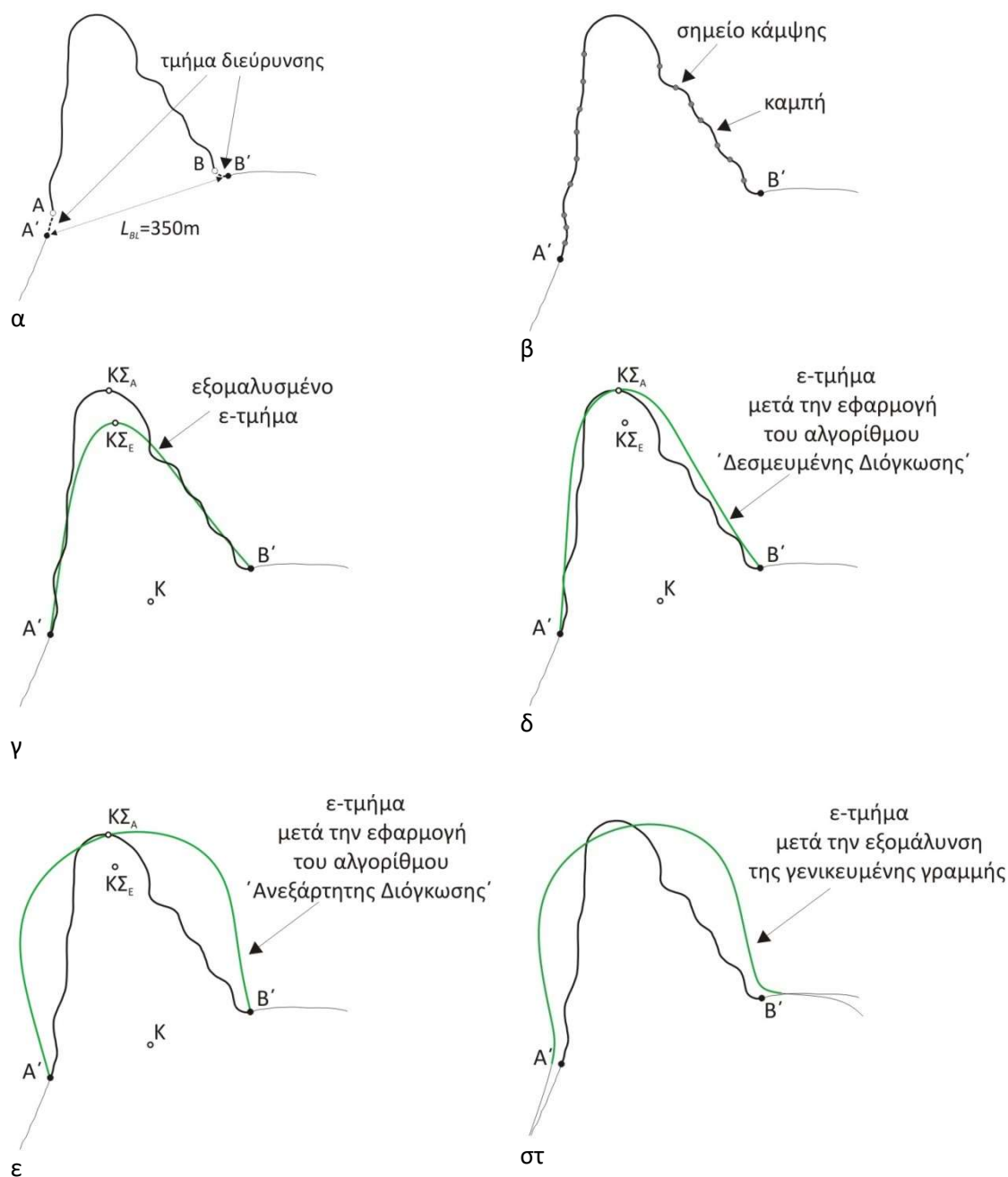
(κατηγορία Α), 8 αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα (κατηγορία Β), 2 τμήματα σύγκλισης που ορίζουν μία περιοχή σύγκλισης (κατηγορία Γ) και 16  $\epsilon$ -κυρτά τμήματα (κατηγορία Δ). Τέλος, για τις ανάγκες γενίκευσης των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων και των τμημάτων σύγκλισης λαμβάνει χώρα η διεύρυνσή τους, προσαρτώντας μέρος των γειτονικών  $\epsilon$ -τμημάτων (βλ. κεφ. 4.1.1 και 4.3.2, αντίστοιχα). Όπως αναφέρεται στο κεφάλαιο 2.3.4, αυτό έχει ως πιθανό αποτέλεσμα τη δημιουργία  $\epsilon$ -κυρτών τμημάτων που δεν γίνονται αντιληπτά από τον αναγνώστη του χάρτη, αφού το ανάπτυγμα τους βρίσκεται στο όριο της οπτικής αντίληψης. Για το λόγο αυτό συνενώνονται με τα εκατέρωθεν γειτονικά  $\epsilon$ -τμήματα. Η τιμή κατωφλίου του αναπτύγματος που τίθεται στην εφαρμογή είναι τα 0,25mm στον χάρτη (ή 250m στο έδαφος). Με την ολοκλήρωση της διαδικασίας προκύπτουν τα τελικά  $\epsilon$ -τμήματα στα οποία αποκόπτεται η ακτογραμμή, τα οποία είναι 7 μονόπλευρα, 8 αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα, 2 τμήματα σύγκλισης που ορίζουν μία περιοχή σύγκλισης και 15  $\epsilon$ -κυρτά τμήματα. Το αποτέλεσμα της εφαρμογής του μοντέλου κατάτμησης στην ακτογραμμή της νήσου Περιστεράς παρουσιάζεται στο Σχήμα 5.3.

### 5.3.1.2 Γενίκευση γραμμής

#### 5.3.1.2.1 Γενίκευση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων

Για την ευκρινή απόδοσή τους στην κλίμακα 1:1.000.000, τα επτά μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα υπόκεινται διαδοχικά στους μετασχηματισμούς διεύρυνσης, εξομάλυνσης και διόγκωσης. Στο Σχήμα 5.4 παρουσιάζεται ένα αντιπροσωπευτικό παράδειγμα γενίκευσης ενός εξ αυτών. Επισημαίνεται πως οι γραμμικές οντότητες των σχημάτων του κεφαλαίου αναπαριστώνται μεγεθυμένες για την καλύτερη παρατήρηση των αποτελεσμάτων.

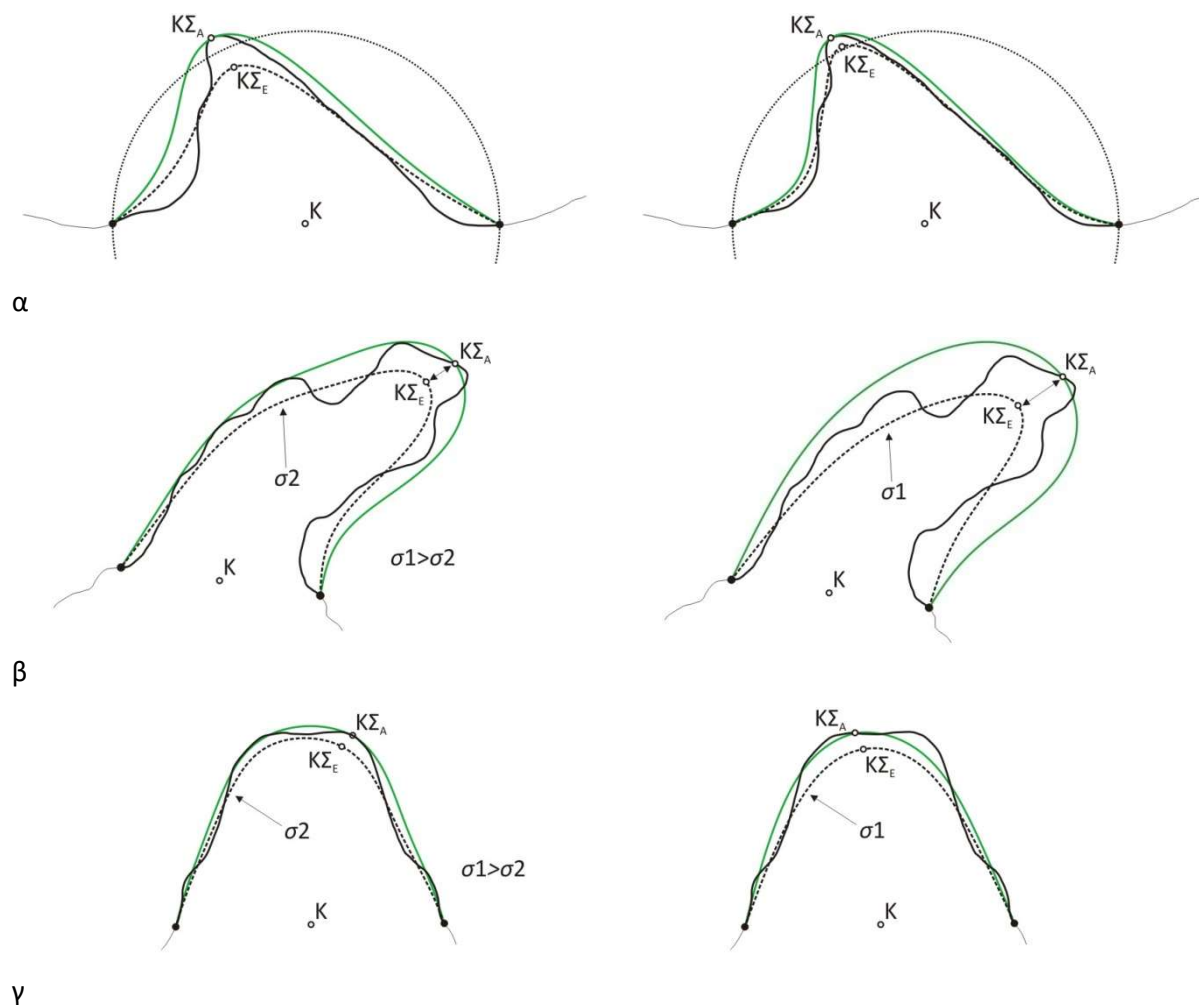
Η διεύρυνση πραγματοποιείται κατά τρόπο, ώστε η γραμμή βάσης των  $\epsilon$ -τμημάτων να είναι ίση με τη διάμετρο του δίσκου του Perkal ( $L_{BL}=\epsilon$ ), όπου  $\epsilon=350m$  στο έδαφος (ή  $\epsilon=0,35mm$  στον χάρτη) (Σχήμα 5.4α). Εν συνεχεία, τα διευρυμένα  $\epsilon$ -τμήματα εξομαλύνονται. Σκοπός είναι η εξάλειψη των τοπικών λεπτομερειών της αρχικής γραμμής, οι οποίες πιθανά επηρεάσουν την μετέπειτα διαδικασία διόγκωσης (βλ. Σχήμα 4.3, κεφ. 4.1.2.1). Όπως φαίνεται στο Σχήμα 5.4β, η λεπτομέρεια, η οποία αποτελεί ουσιαστικά μικρές διακυμάνσεις της γραμμής, μπορεί να ποσοτικοποιηθεί εντοπίζοντας τις καμπές (bends) κατά μήκος της. Επιδίωξη είναι η εξομάλυνση των  $\epsilon$ -τμημάτων σε βαθμό που να ελαχιστοποιείται ο αριθμός των καμπών και ιδανικά να εντοπίζεται μία. Η διεργασίες εξομάλυνσης και εντοπισμού καμπών υλοποιούνται μέσω του τελεστή *Gauss* και της μεθόδου που στηρίζεται στον ορισμό των Wang και Müller (1998) αντίστοιχα, και οι οποίες έχουν προσαρμοσθεί και ενσωματωθεί στο μοντέλο γενίκευσης (βλ. κεφ. 3.2.2 και 3.3). Η διαδικασία λαμβάνει χώρα εφαρμόζοντας δοκιμαστικά τον τελεστή *Gauss* και εντοπίζοντας κάθε φορά τις καμπές στο εξομαλυσμένο  $\epsilon$ -τμήμα.



Σχήμα 5.4: Παράδειγμα γενίκευσης μονόπλευρου ε-μη-κυρτού τμήματος.

Συγχρόνως, όπως αναφέρεται στο κεφάλαιο 4.1.2.1, εκτός της απαλοιφής των τοπικών ανωμαλιών της αρχικής γραμμής, από την εξομάλυνση εξαρτάται η διαμόρφωση της γεωμετρίας στη θέση  $K$ . Από τη σχετική θέση μεταξύ των κορυφών  $KΣ_E$  (η οποία εντοπίζεται μέσω του δείκτη αναλογίας μήκους  $ALR$ , βλ. κεφ. 3.1) και  $KΣ_A$  του εξομαλυσμένου και του αρχικού ε-τμήματος αντίστοιχα, και του δίσκου του Perkal στη θέση  $K$ , καθορίζεται η μέθοδος και ο βαθμός του μετασχηματισμού διόγκωσης που ακολουθεί (βλ. κεφ. 4.1.2.2.1). Στο Σχήμα 5.5 παρουσιάζονται ορισμένα χαρακτηριστικά παραδείγματα του τρόπου με τον οποίο μπορεί να επηρεάσει ο βαθμός εξομάλυνσης τη διαδικασία διόγκωσης. Στο Σχήμα 5.5α παρατηρείται πως, ελαττώνοντας από το μέγιστο τον βαθμό εξομάλυνσης,

μεταβάλλεται η σχετική θέση μεταξύ της κορυφής  $K_{ΣΕ}$  σχέση με το δίσκο του Perkal στη θέση  $K$ . Αυτό έχει ως αποτέλεσμα να αλλάζει η μέθοδος υπολογισμού της τιμής 'Inflating' του αλγορίθμου 'Balloon', αλλά και ο τρόπος εφαρμογής του ομοπαράλληλου μετασχηματισμού κατά την υλοποίηση του αλγορίθμου 'Δεσμευμένης Διόγκωσης'. Στο Σχήμα 5.5β παρουσιάζεται πώς μεταβάλλεται η τιμή 'Inflating' σε σχέση με το βαθμό εξομάλυνσης. Στις περιπτώσεις που η κορυφή  $K_{ΣΕ}$  εντοπίζεται εκτός του δίσκου του Perkal (οι οποίες αποτελούν πλειοψηφία κατά την εφαρμογή), η τιμή 'Inflating' αυξάνει ανάλογα με το βαθμό εξομάλυνσης, με αποτέλεσμα τη μεγαλύτερη διόγκωση του  $\epsilon$ -τμήματος. Το γεγονός βέβαια δεν αποτελεί κανόνα. Για παράδειγμα, μεταβάλλοντας το βαθμό εξομάλυνσης μπορεί οι κορυφές  $K_{ΣΕ}$  να εντοπίζονται σε θέσεις, των οποίων οι αντίστοιχες στο αρχικό  $\epsilon$ -τμήμα ( $K_{ΣΑ}$ ) να διαφέρουν σημαντικά. Έτσι, όπως φαίνεται στο Σχήμα 5.5γ, υπάρχει πιθανότητα να επιτυγχάνεται μεγαλύτερη διόγκωση επιλέγοντας την λιγότερο εξομαλυσμένη εκδοχή του  $\epsilon$ -τμήματος, γεγονός που οφείλεται στον ομοπαράλληλο μετασχηματισμό ως προς την κορυφή  $K_{ΣΑ}$ .



Σχήμα 5.5: Παραδείγματα αλληλεπίδρασης μετασχηματισμών εξομάλυνσης και διόγκωσης.

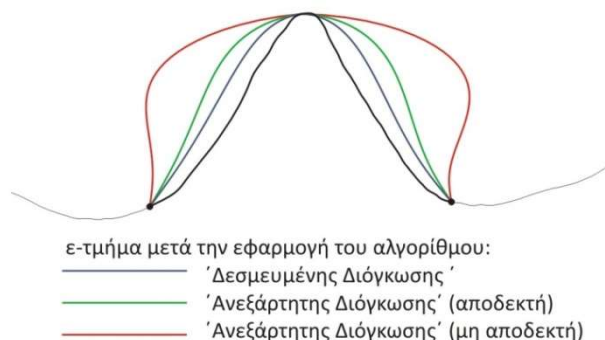
Από τα παραπάνω γίνεται φανερό πως υπάρχει άμεση συνάρτηση των διεργασιών εξομάλυνσης και διόγκωσης. Κρίνεται, λοιπόν, σκόπιμο να αντιμετωπίζονται ενιαία από τον χρήστη. Η επιλογή του κατάλληλου βαθμού εξομάλυνσης γίνεται κατόπιν δοκιμών, εφαρμόζοντας τον τελεστή *Gauss* με διάφορες τιμές της μεταβλητής  $\sigma$ . Επικουρικά, προτείνονται τρεις χαρακτηριστικές τιμές, οι οποίες χρησιμοποιούνται ως 'οδηγός' για την εφαρμογή: η μέγιστη ( $\sigma_{max}$ ), η ελάχιστη με την οποία εντοπίζεται ο ελάχιστος αριθμός καμπών ( $\sigma_{min1}$ ) και η ελάχιστη με την οποία εντοπίζεται ο αμέσως επόμενος αριθμός καμπών ( $\sigma_{min2}$ ). Από την πειραματική διαδικασία έχει προκύψει πως εντός του συγκεκριμένου εύρους τιμών επιτυγχάνονται, αφενός η εξάλειψη της λεπτομέρειας της αρχικής γραμμής και αφετέρου, οι βέλτιστες λύσεις διόγκωσης των  $\epsilon$ -τμημάτων. Στον Πίνακα 5.4 παρουσιάζονται οι παράμετροι υλοποίησης του τελεστή εξομάλυνσης κατά την εφαρμογή του στα επτά μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα.

A/A $\epsilon$ -τμήματος	αριθμός αρχικών καμπών	$\sigma_{max}$	ελάχιστος αριθμός καμπών	$\sigma_{min1}$	επόμενος αριθμός καμπών	$\sigma_{min2}$	$\sigma$ εφαρμογής
1	86	34.5	2	27	3	24	<b>34.5</b>
2	40	12.25	2	12	4	5	<b>12.25</b>
3	32	12.5	3	3.5	5	3	<b>12.5</b>
4	37	17	1	16.75	2	12.25	<b>17</b>
5	35	14.75	1	12.5	3	9.75	<b>9.75</b>
6	41	19.5	3	13.5	5	12.25	<b>19.5</b>
7	35	13	3	6	5	5.75	<b>13</b>

**Πίνακας 5.4:** Αριθμός καμπών και παράμετροι εφαρμογής του τελεστή εξομάλυνσης στα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα.

Έχοντας επιλεγεί ο βαθμός εξομάλυνσης και εντοπισθεί τα σημεία  $K_E$ ,  $K_A$  και  $K$  (Σχήμα 5.4γ), λαμβάνει χώρα η διόγκωση του εξομαλυσμένου  $\epsilon$ -τμήματος. Σε ένα πρώτο επίπεδο επεξεργασίας, αυτή επιλέγεται να υλοποιηθεί μέσω του αλγορίθμου 'Δεσμευμένης Διόγκωσης' (βλ. κεφ.4.1.2.2.1). Ανάλογα με τη γεωμετρία στη 'θέση  $K$ ' ορίζονται οι παράμετροι εφαρμογής του, δηλαδή η τιμή του συντελεστή 'Inflating' του αλγορίθμου 'Balloon' ( $Inflating = (K \Sigma_E K \Sigma_A)$  ή  $Inflating = (K \Sigma_E T)$ ) και η εφαρμογή ή μη του ομοπαράλληλου μετασχηματισμού.

Στην περίπτωση που το αποτέλεσμα από την εφαρμογή του αλγορίθμου 'Δεσμευμένης Διόγκωσης' (Σχήμα 5.4δ) δεν καλύπτει τις προϋποθέσεις ευαναγνωσιμότητας, η διόγκωση υλοποιείται μέσω του αλγορίθμου 'Ανεξάρτητης Διόγκωσης' (βλ. κεφ.4.1.2.2.2). Λαμβάνοντας ως δεδομένο τη γεωμετρία στη 'θέση  $K$ ', τίθεται από τον χρήστη η τιμή του συντελεστή 'Inflating' του αλγορίθμου 'Balloon'. Η τιμή που επιλέγεται πρέπει να είναι η ελάχιστη αναγκαία, ώστε να επιτυγχάνονται οι συνθήκες ευαναγνωσιμότητας του  $\epsilon$ -τμήματος στην παράγωγη κλίμακα με τη δυνατόν μικρότερη μεταβολή της μορφής και της γεωμετρίας του (Σχήμα 5.4ε). Σε αντίθετη περίπτωση, το αποτέλεσμα μπορεί να είναι αισθητικά και χαρτογραφικά μη αποδεκτό, όπως φαίνεται στο παράδειγμα του Σχήματος 5.6.

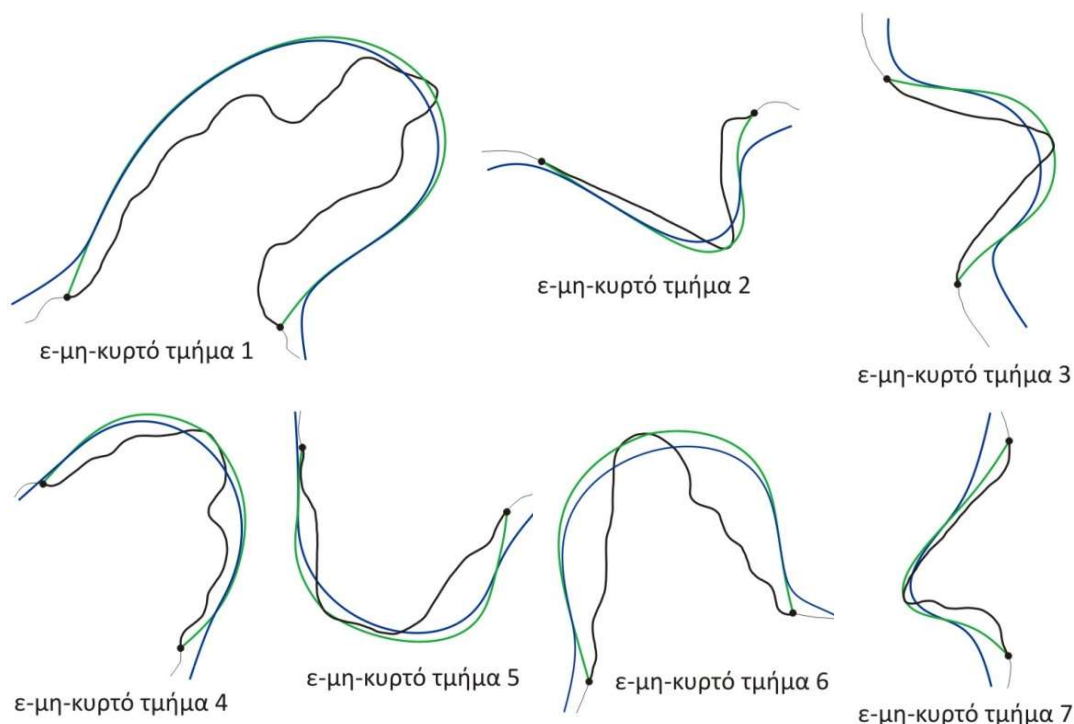


Σχήμα 5.6: Παράδειγμα εφαρμογής αλγορίθμου 'Ανεξάρτητης Διόγκωσης'.

Στον Πίνακα 5.5 παρουσιάζονται οι παράμετροι που χρησιμοποιήθηκαν για τη διόγκωση των επτά μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων. Το αποτέλεσμα της επεξεργασίας για τη γενίκευση των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων παρουσιάζεται στο Σχήμα 5.7. Με γραμμή πράσινης απόχρωσης αποδίδονται τα  $\epsilon$ -τμήματα όπως έχουν προκύψει από τους μετασχηματισμούς γενίκευσης και με γραμμή μπλε απόχρωσης όπως διαμορφώνονται μετά την εξομάλυνση του συνόλου της ακτογραμμής (βλ. κεφ. 4.5 και κεφ. 5.3.1.2.5).

A/A $\epsilon$ -τμήματος	τιμή συντελεστή 'Inflating' αλγορίθμου 'Δεσμευμένης Διόγκωσης' (m)	αλγόριθμος Διόγκωσης που εφαρμόζεται	τιμή συντελεστή 'Inflating' αλγορίθμου 'Ανεξάρτητης Διόγκωσης' (m)
1	93.26	Ανεξάρτητη	150
2	37.63	Δεσμευμένη	
3	35.88	Ανεξάρτητη	90
4	40.85	Ανεξάρτητη	120
5	19.50	Ανεξάρτητη	120
6	54.46	Ανεξάρτητη	170
7	18.41	Δεσμευμένη	

Πίνακας 5.5: Παράμετροι διόγκωσης μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων.



Σχήμα 5.7: Τα μονόπλευρα ε-μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης.

### 5.3.1.2.2 Γενίκευση αμφίπλευρων ε-μη-κυρτών τμημάτων

Για την ευκρινή απόδοσή τους στην κλίμακα 1:1.000.000, τα οκτώ αμφίπλευρα ε-μη-κυρτά τμήματα υπόκεινται διαδοχικά στους μετασχηματισμούς απλοποίησης, δηλαδή στην αποδόμησή τους στις κύριες καμπές, διόγκωσης των τελευταίων ως ανεξάρτητες οντότητες και επανένωσής τους για τη δημιουργία του γενικευμένου ε-τμήματος.

Η απλοποίηση περιλαμβάνει, σε συνδυασμό, την εξομάλυνση και τον εντοπισμό των καμπών κατά μήκος του εξομαλυσμένου ε-τμήματος. Η διεργασία υλοποιούνται μέσω του τελεστή *Gauss* και της μεθόδου που στηρίζεται στον ορισμό των Wang και Müller (1998) αντίστοιχα, όπως έχουν προσαρμοσθεί και ενσωματωθεί στο μοντέλο γενίκευσης (βλ. κεφ. 3.2.2 και 3.3). Στόχος είναι η εξάλειψη της λεπτομέρειας της αρχικής γραμμής, η οποία ποσοτικοποιείται μέσω των καμπών, και η επιλογή εκείνων (κύριων) που είναι απαραίτητες για τη διαμόρφωση του ε-τμήματος στην παράγωγη κλίμακα. Όπως αναφέρεται στο κεφάλαιο 4.2.1.2, ένα βασικό κριτήριο για την επιλογή των προς διατήρηση καμπών είναι η αναλογία του μήκους της γραμμής βάσης τους ( $L_{BL}$ ), ως προς τη διάμετρο εφαρμογής  $\epsilon$  του θεωρητικού δίσκου του Perkal. Έτσι, για το παρόν σενάριο γενίκευσης αναζητούνται καμπές για τις οποίες ισχύει  $L_{BL} \geq 350m$  στο έδαφος. Η διαδικασία λαμβάνει χώρα εφαρμόζοντας δοκιμαστικά τον τελεστή *Gauss* και εντοπίζοντας κάθε φορά τις καμπές στο εξομαλυσμένο ε-τμήμα. Ως 'οδηγός' για την εφαρμογή χρησιμοποιούνται τρεις χαρακτηριστικές τιμές της μεταβλητής  $\sigma$  του τελεστή *Gauss*: η μέγιστη ( $\sigma_{max}$ ), η ελάχιστη με την οποία εντοπίζεται ο ελάχιστος αριθμός καμπών ( $\sigma_{min1}$ ) και η ελάχιστη με την οποία εντοπίζεται ο αμέσως επόμενος αριθμός καμπών ( $\sigma_{min2}$ ). Εντός του συγκεκριμένου εύρους τιμών, οι καμπές που εντοπίζονται πληρούν ή

προσεγγίζουν το κριτήριο επιλογής/διατήρησης. Για μικρότερες τιμές οι καμπές που εντοπίζονται δεν είναι ευδιάκριτες στην παράγωγη κλίμακα. Ο αριθμός καμπών και οι τιμές της μεταβλητής  $\sigma$  που χρησιμοποιούνται παρουσιάζονται στον Πίνακα 5.6.

A/A ε-τμήματος	αριθμός αρχικών καμπών	$\sigma_{max}$	ελάχιστος αριθμός καμπών	$\sigma_{min1}$	επόμενος αριθμός καμπών	$\sigma_{min2}$	$\sigma$ εφαρμογής	αριθμός κύριων καμπών
1	152	59	2	56.75	4	24	<b>59</b>	<b>2</b>
2	156	80.5	3	74.25	4	33.75	<b>45</b>	<b>4</b>
3	37	18	2	8.75	4	7.25	<b>15</b>	<b>2</b>
4	105	41.25	4	12	6	11.5	<b>41.25</b>	<b>4</b>
5	48	20.25	2	13.75	3	9	<b>13.75</b>	<b>2</b>
6	63	24.25	2	17.25	3	13.75	<b>17.25</b>	<b>2</b>
7	75	28.75	2	18.75	4	14	<b>28.75</b>	<b>2</b>
8	93	32.5	4	10.75	6	8.75	<b>32.5</b>	<b>4</b>

**Πίνακας 5.6:** Αριθμός καμπών και παράμετροι εφαρμογής εξομάλυνσης στα αμφίπλευρα ε-μη-κυρτά τμήματα.

Όπως παρατηρείται στον Πίνακα 5.6, ο αριθμός των κύριων καμπών που επιλέγονται ταυτίζεται με τον ελάχιστο αριθμό καμπών που μπορούν να εντοπισθούν εφαρμόζοντας το μέγιστο βαθμό εξομάλυνσης. Κατ' αυτό τον τρόπο, τα ε-τμήματα αποκόπτονται σε καμπές με το δυνατόν μεγαλύτερο μήκος γραμμής βάσης. Εξαιρέση αποτελεί το ε-τμήμα 2, στο οποίο υπήρχε η δυνατότητα οι καμπές '1' και '2' (βλ. Σχήμα 5.8) να εντοπισθούν ως μία, ενιαία. Η επιλογή έγινε μετά από αξιολόγηση του τελικού προϊόντος, αφού με τον τρόπο αυτό, αφενός η γενικευμένη γραμμή διέρχεται από μία θέση μεγάλης πολυπλοκότητας και αφετέρου διότι με τον βαθμό εξομάλυνσης που επιλέχθηκε δημιουργήθηκαν οι προϋποθέσεις πιο ικανοποιητικής διόγκωσης του συνόλου του ε-τμήματος.

Μετά τον προσδιορισμό του αριθμού των κύριων καμπών, λαμβάνει χώρα η επιλογή της τελικής τιμής εφαρμογής της μεταβλητής  $\sigma$  του τελεστή *Gauss*. Όπως φαίνεται στο Σχήμα 4.14 (βλ. κεφ. 4.2.1.2), η αλλαγή της τιμής  $\sigma$  επιφέρει μεταβολή, τόσο τα άκρα των καμπών (σημεία κάμψης), βάσει των οποίων υπολογίζεται το μήκος της γραμμής βάσης, όσο και στη γεωμετρία στη 'θέση Κ', από την οποία εξαρτάται η επεξεργασία διόγκωσης που ακολουθεί. Η επιλογή του βαθμού εξομάλυνσης πραγματοποιήθηκε μετά από δοκιμές, λαμβάνοντας υπόψη τα ποσοτικά χαρακτηριστικά των προκυπτουσών καμπών, αλλά και συνεκτιμώντας κάθε φορά το αποτέλεσμα της διόγκωσης.

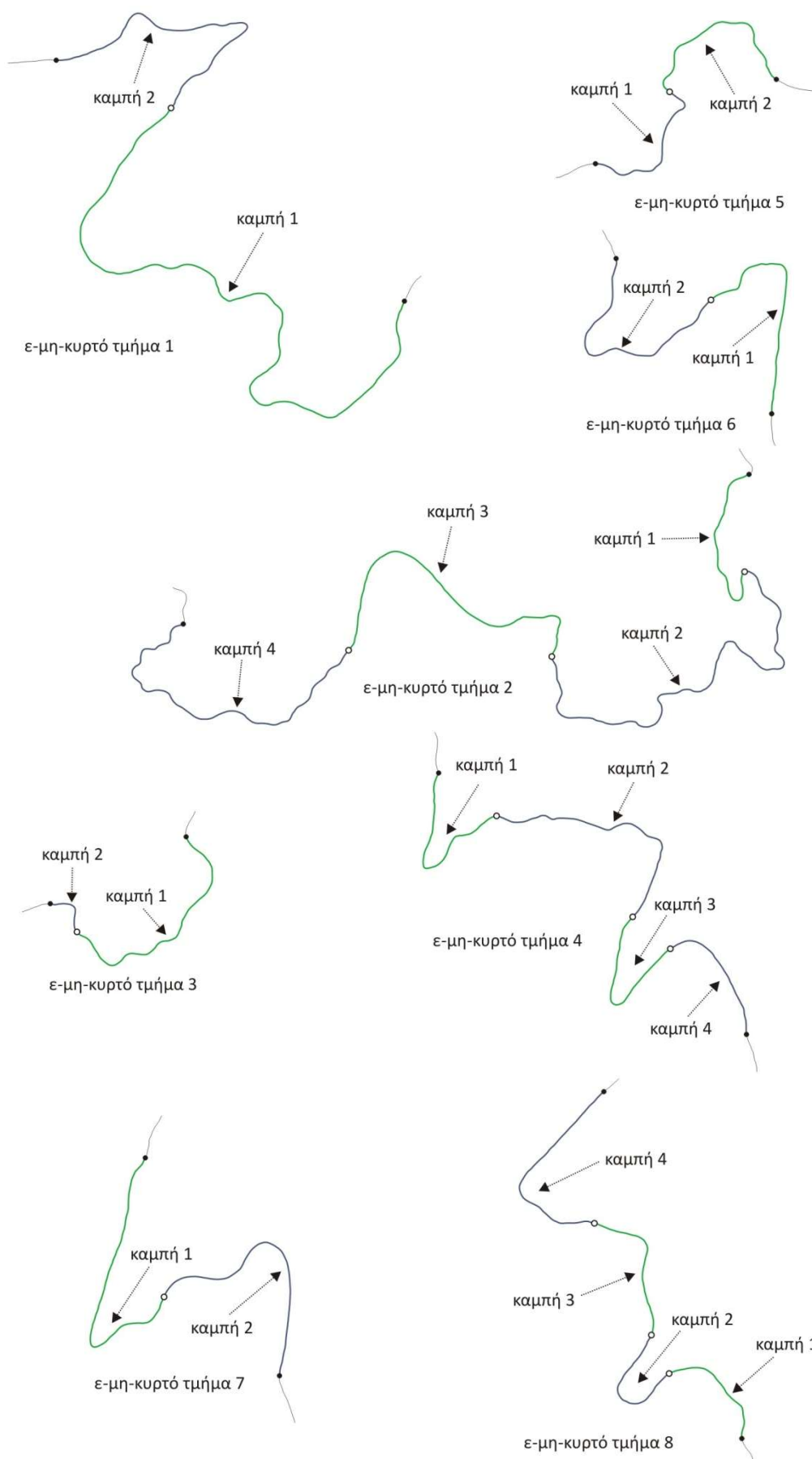
Στον Πίνακα 5.7 αναγράφονται τα ποσοτικά χαρακτηριστικά των κύριων καμπών που προκύπτουν από την εφαρμογή του τελεστή *Gauss* με τις τιμές της μεταβλητής  $\sigma$  που επιλέχθηκαν. Παρατηρείται πως σε ορισμένες περιπτώσεις, τα αμφίπλευρα ε-μη-κυρτά τμήματα αποδομούνται σε κύριες καμπές των οποίων το μήκος της γραμμής βάσης είναι αρκετά μικρότερο του κριτηρίου επιλογής (πχ ε-τμήματα 5 και 6). Το γεγονός οφείλεται στον περιορισμό που επιφέρει η εφαρμογή του τελεστή με *Gauss* συγκριμένες/πεπερασμένες τιμές της παραμέτρου  $\sigma$ , οι οποίες εξαρτώνται από τον αριθμό των σημείων που δομούν το ε-τμήμα.



A/A ε-τμήματος	A/A καμπής	μήκος γραμμής βάσης $L_{βι}(m)$	βάθος (m)	εμβαδόν (m <sup>2</sup> )
1	1	686.1	229.2	142801.2
	2	284.8	125.2	23340.6
2	1	259.5	14.5	2542.0
	2	485.8	96.9	61407.6
	3	486.1	107.0	36336.6
	4	358.5	106.4	38386.8
3	1	324.8	91.5	37396.5
	2	91.0	4.6	268.3
4	1	206.6	57.9	7515.9
	2	344.1	77.3	18107.0
	3	176.7	25.4	2819.3
	4	204.8	6.1	822.5
5	1	237.8	70.2	11495.9
	2	238.4	121.5	22563.5
6	1	284.7	133.8	25929.3
	2	237.2	174.3	28411.8
7	1	325.5	122.8	26925.5
	2	304.8	136.8	27046.4
8	1	184.9	10.7	1294.3
	2	159.9	27.4	2772.9
	3	252.2	23.7	3841.5
	4	318.6	75.1	15684.9

**Πίνακας 5.7:** Ποσοτικά χαρακτηριστικά κύριων καμπών  
αμφίπλευρων ε-μη-κυρτών τμημάτων.

Στο Σχήμα 5.8 παρουσιάζονται οι κύριες καμπές στις οποίες αποδομούνται τα οκτώ αμφίπλευρα ε-μη-κυρτά τμήματα, αν αντιστοιχισθούν τα σημεία κάμψης στην αρχική εκδοχή τους. Το αποτέλεσμα της διαδικασίας απλοποίησης κρίνεται αρκετά ικανοποιητικό, υπό την έννοια πως αν ζητούνταν από κάποιον παρατηρητή να διαχωρίσει τα ε-τμήματα στις βασικές κυρτώσεις και κοιλώσεις τους, αυτές θα μπορούσαν να ταυτισθούν με τις κύριες καμπές που έχουν εντοπισθεί. Εξαίρεση ενδεχομένως αποτελεί η 'καμπή 1' του πρώτου ε-τμήματος, η οποία θα μπορούσε να διαχωρισθεί σε τρεις επιμέρους και η 'καμπή 2' του τρίτου ε-τμήματος, η οποία λόγω μεγέθους, πιθανά δεν θα είχε εντοπισθεί. Επισημαίνεται πως για την καλύτερη παρατήρηση, οι γραμμικές οντότητες των σχημάτων του κεφαλαίου αποδίδονται σε μεγέθυνση.



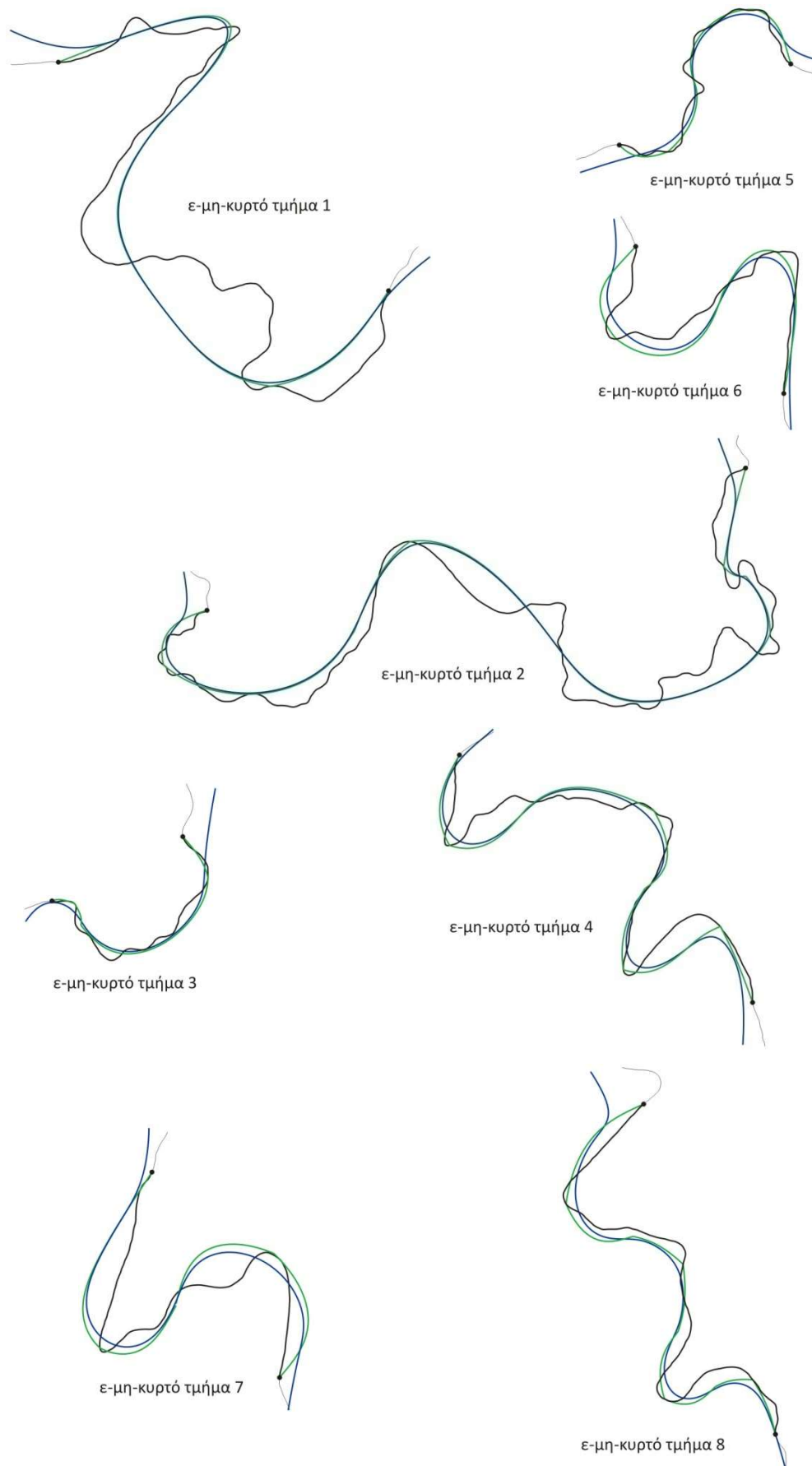
Σχήμα 5.8: Οι κύριες καμπές αντιστοιχισμένες στα αρχικά αμφίπλευρα ε-μη-κυρτά τμήματα.

Στο δεύτερο στάδιο επεξεργασίας η εξομαλυσμένη εκδοχή του εκάστοτε αμφίπλευρου  $\epsilon$ -μη-κυρτού τμήματος υπόκειται στο μετασχηματισμό διόγκωσης. Η διαδικασία υλοποιείται ανεξάρτητα σε κάθε κύρια καμπή του  $\epsilon$ -τμήματος και είναι όμοια με εκείνη των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων. Όπως αναφέρεται στο κεφάλαιο 4.2.2 (Σχήμα 4.15) για την εκάστοτε καμπή έχουν υπολογισθεί και εισάγονται στο σύστημα τα γεωμετρικά χαρακτηριστικά της, δηλαδή, η αρχή και το πέρας, η κορυφή  $K_{\Sigma\epsilon}$  (η οποία έχει εντοπισθεί μέσω της εφαρμογής του δείκτη αναλογίας μήκους ALR), το πρόσημο της καμπυλότητας, το κέντρο  $K$  του θεωρητικού δίσκου του Perkal και η κορυφή  $K_{\Sigma\alpha}$  της αρχικής καμπής. Η διαδικασία υλοποιείται μέσω των αλγορίθμων 'Δεσμευμένης Διόγκωσης' (βλ. κεφ. 4.1.2.2.1) ή 'Ανεξάρτητης Διόγκωσης' (βλ. κεφ. 4.1.2.2.2). Οι παράμετροι διόγκωσης των οκτώ αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων παρουσιάζονται στον Πίνακα 5.8.

A/A $\epsilon$ -τμήματος	A/A καμπής	τιμή συντελεστή 'Inflating' αλγορίθμου 'Δεσμευμένης Διόγκωσης' (m)	αλγόριθμος Διόγκωσης που εφαρμόζεται	τιμή συντελεστή 'Inflating' αλγορίθμου 'Ανεξάρτητης Διόγκωσης' (m)
1	1	124.34	Δεσμευμένη	
	2	28.90	Δεσμευμένη	
2	1	153.62	Δεσμευμένη	
	2	64.88	Δεσμευμένη	
	3	41.43	Δεσμευμένη	
	4	109.60	Ανεξάρτητη	150
3	1	20.02	Δεσμευμένη	
	2	170.34	Δεσμευμένη	
4	1	117.04	Δεσμευμένη	
	2	83.80	Ανεξάρτητη	240
	3	149.52	Δεσμευμένη	
	4	166.56	Δεσμευμένη	
5	1	96.15	Δεσμευμένη	
	2	45.75	Δεσμευμένη	
6	1	42.71	Δεσμευμένη	
	2	44.63	Ανεξάρτητη	120
7	1	99.00	Δεσμευμένη	
	2	37.33	Ανεξάρτητη	200
8	1	162.70	Δεσμευμένη	
	2	146.42	Δεσμευμένη	
	3	151.23	Δεσμευμένη	
	4	85.26	Ανεξάρτητη	200

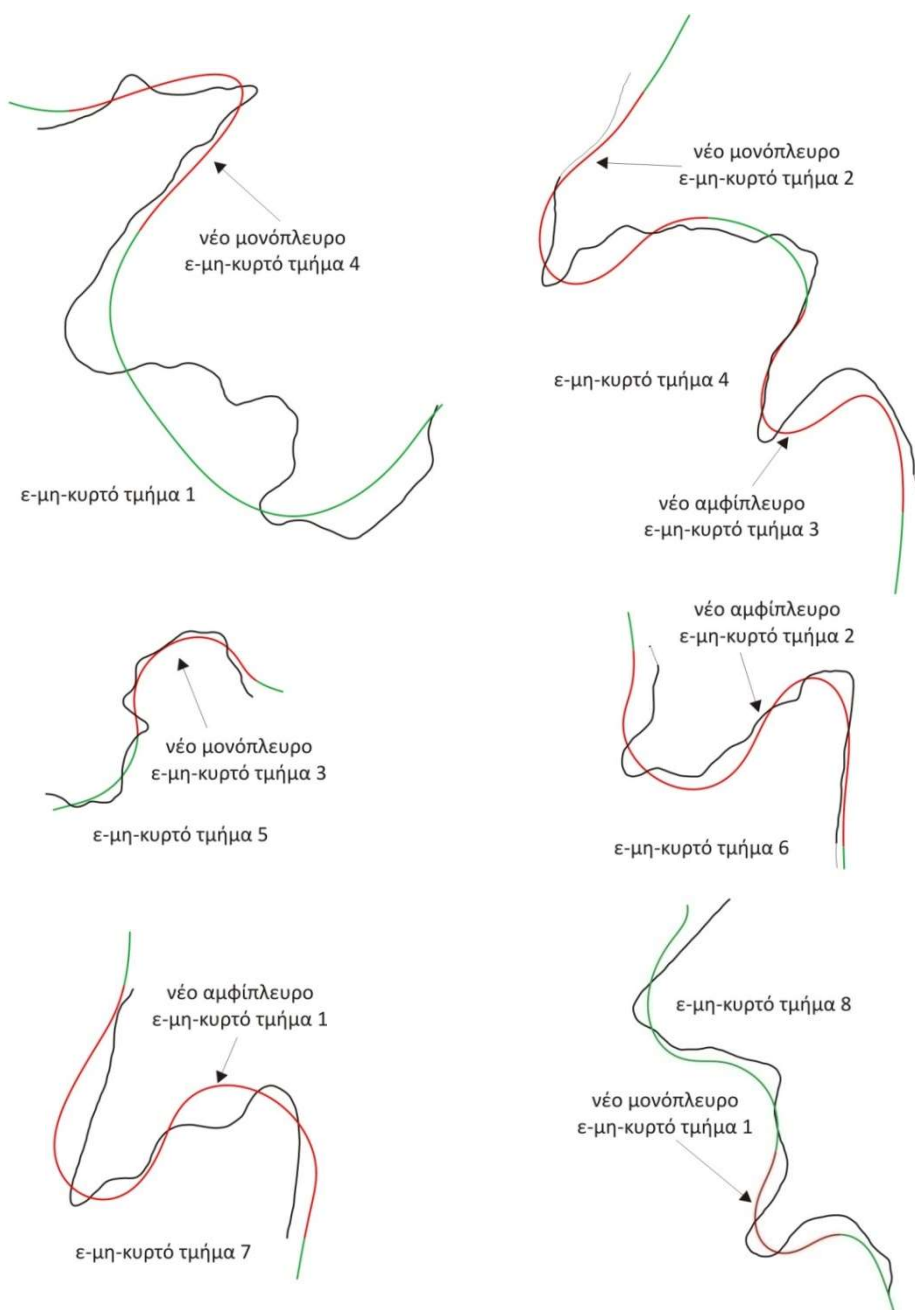
**Πίνακας 5.8:** Παράμετροι διόγκωσης αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων.

Το αποτέλεσμα της επεξεργασίας για τη γενίκευση των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων παρουσιάζεται στο Σχήμα 5.9. Με γραμμή πράσινης απόχρωσης παρουσιάζονται τα  $\epsilon$ -τμήματα όπως έχουν προκύψει από τους μετασχηματισμούς γενίκευσης και με γραμμή μπλε απόχρωσης όπως διαμορφώνονται μετά την εξομάλυνση του συνόλου της ακτογραμμής (βλ. κεφ. 4.5 και 5.3.1.2.5).



**Σχήμα 5.9:** Τα αμφίπλευρα ε-μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης.

Μετά την εφαρμογή του μοντέλου ελέγχου για την ευαναγνωσιμότητα των παράγωγων γραμμών (βλ. κεφ. 4.6, Σχήμα 4.25) προέκυψαν νέα  $\epsilon$ -μη-κυρτά τμήματα. Το γεγονός καταδεικνύει πως οι θέσεις αυτές της παράγωγης ακτογραμμής δεν αποδίδονται ευκρινώς στην κλίμακα 1:1.000.000. Επειδή τα παραπάνω  $\epsilon$ -τμήματα εντοπίζονται σε θέσεις που σχετίζονται με τα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα, η επεξεργασία για τη διόρθωση του προβλήματος περιγράφεται στο συγκεκριμένο κεφάλαιο.



**Σχήμα 5.10:**  $\epsilon$ -μη-κυρτά τμήματα που προκύπτουν μετά τον έλεγχο ευαναγνωσιμότητας.

Παρατηρώντας τα νέα  $\epsilon$ -μη-κυρτά τμήματα (γραμμές κόκκινης απόχρωσης) στο Σχήμα 5.10, συμπεραίνονται τα εξής: α) οι διεργασίες γενίκευσης δεν είναι επαρκείς στα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα τα οποία αποτελούνται από δύο, σχεδόν όμοιες σε σχήμα και μέγεθος κύριες καμπές (βλ. Πίνακα 5.7), με αποτέλεσμα να δημιουργείται πρόβλημα είτε στην μία ( $\epsilon$ -τμήμα 5), είτε και στις δύο καμπές ( $\epsilon$ -τμήματα 6 και 7). Το γεγονός οφείλεται σε ένα συνδυασμό παραγόντων που έχουν να κάνουν με το ότι το μήκος της γραμμής βάσης τους είναι αρκετά μικρότερο από το τη διάμετρο εφαρμογής  $\epsilon$ , η γεωμετρία στη 'θέση Κ' είναι τέτοια που δεν δημιουργεί τις προϋποθέσεις για εφαρμογή του αλγορίθμου 'Δεσμευμένης Διόγκωσης' με ικανοποιητικές τιμές του συντελεστή 'Inflating', ενώ η εφαρμογή του αλγορίθμου 'Ανεξάρτητης Διόγκωσης' δημιουργεί πρόβλημα 'σύγκρουσης' μεταξύ των διογκωμένων καμπών, λόγω του περιορισμένου χώρου. β) πρόβλημα ευαναγνωσιμότητας εμφανίζεται σε μεμονωμένες καμπές αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων, οι οποίες χαρακτηρίζονται από μεγάλο βάθος και μήκος γραμμής βάσης μικρότερο της διαμέτρου εφαρμογής  $\epsilon$  (περιπτώσεις  $\epsilon$ -τμημάτων 1, 4 και 8).

Για την επίλυση του προβλήματος ευαναγνωσιμότητάς τους στην κλίμακα 1:1.000.000, τα νέα  $\epsilon$ -μη-κυρτά τμήματα αποκόπτονται από την υπόλοιπη γραμμή και υπόκεινται στις διεργασίες γενίκευσης αντίστοιχες με εκείνες των αρχικών  $\epsilon$ -μη-κυρτών τμημάτων και ανάλογα με την κατηγορία στην οποία ανήκουν. Όπως φαίνεται στο Σχήμα 5.10, από τη διαδικασία έχουν προκύψει τέσσερα μονόπλευρα και τρία αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα.

Στον Πίνακα 5.9 και 5.10 παρουσιάζονται οι παράμετροι γενίκευσης των τεσσάρων μονόπλευρων και των τριών αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων, αντίστοιχα.

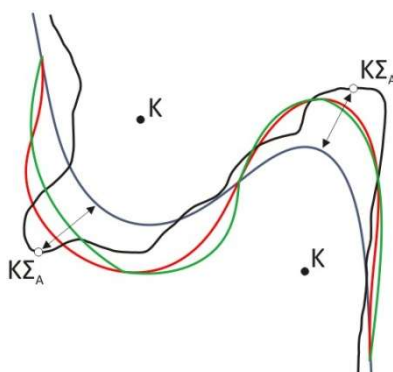
A/A $\epsilon$ -τμήματος	$\sigma$ εφαρμογής τελεστή εξομάλυνσης	αλγόριθμος Διόγκωσης που εφαρμόζεται	τιμή συντελεστή 'Inflating' αλγορίθμου Διόγκωσης (m)
1	15.25	Δεσμευμένη	29.75
2	24.75	Δεσμευμένη	56.10
3	15.5	Δεσμευμένη	8.47
4	21.75	Ανεξάρτητη	160

**Πίνακας 5.9:** Παράμετροι γενίκευσης νέων μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων.

A/A $\epsilon$ -τμήματος	$\sigma$ εφαρμογής τελεστή εξομάλυνσης	αριθμός κύριων καμπών	A/A καμπής	αλγόριθμος Διόγκωσης που εφαρμόζεται	τιμή συντελεστή 'Inflating' αλγορίθμου Διόγκωσης (m)
1	26.25	2	1	Δεσμευμένη	69.03
			2	Δεσμευμένη	39.85
2	22.5	2	1	Δεσμευμένη	9.30
			2	Δεσμευμένη	35.21
3	17.5	2	1	Δεσμευμένη	56.09
			2	Δεσμευμένη	83.72

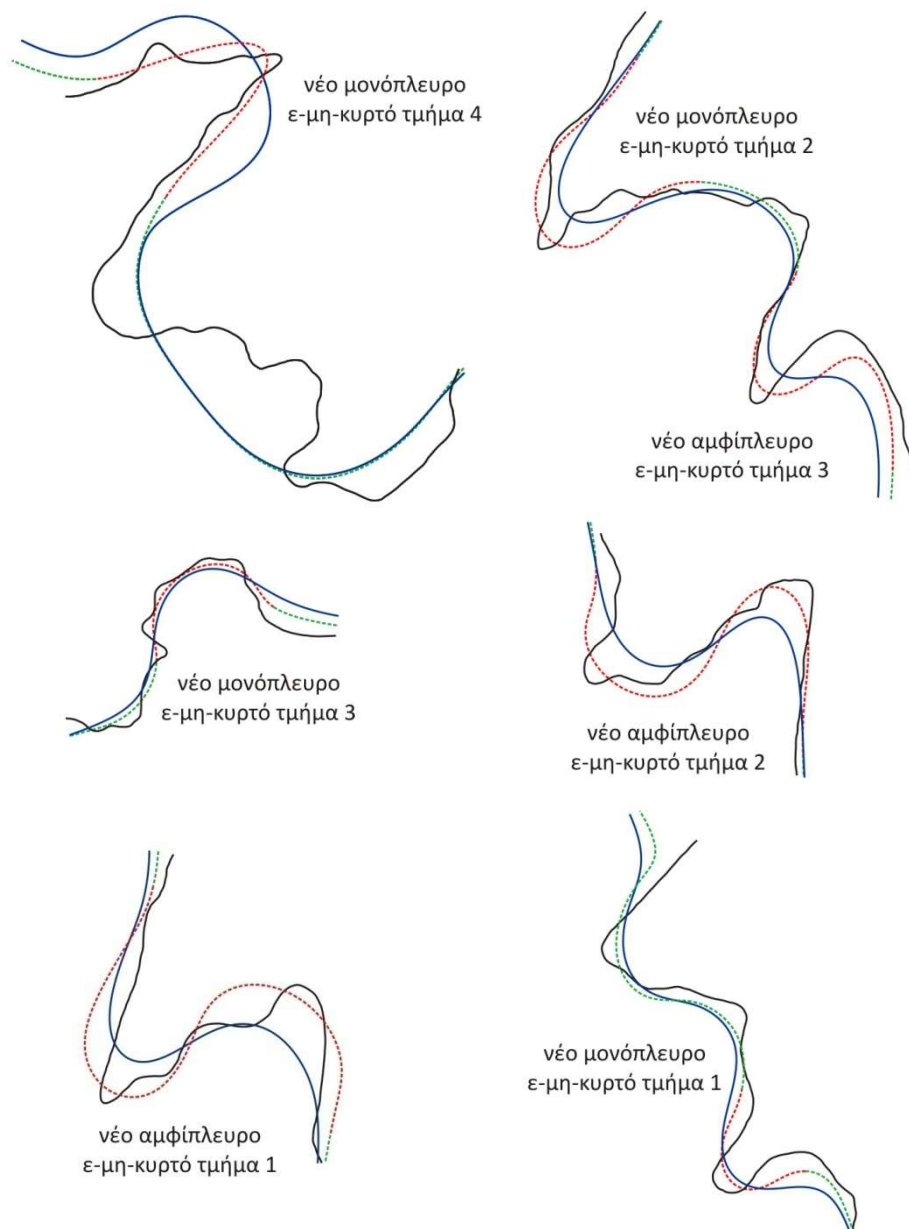
**Πίνακας 5.10:** Παράμετροι γενίκευσης νέων αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων.

Ιδιαίτερο ενδιαφέρον παρουσιάζουν οι περιπτώσεις των νέων αμφίπλευρων 1 και 2, στα οποία με τους μετασχηματισμούς που εφαρμόζονται, βελτιώνεται, αλλά δεν επιλύεται το πρόβλημα ευαναγνωσιμότητας (γραμμή πράσινης απόχρωσης στο Σχήμα 5.11). Αυτό επιτυγχάνεται με την εξομάλυνση της γενικευμένης γραμμής που λαμβάνει χώρα στο σύνολο της ακτογραμμής (βλ. κεφ. 4.5 και κεφ. 5.3.1.2.5). Όπως φαίνεται στο παράδειγμα του Σχήματος 5.11, η διαδικασία έχει ως αποτέλεσμα να αμβλύνεται η θέση σύνδεσης των δύο καμπών και συγχρόνως, αυτές να συρρικνώνονται προς το κέντρο τους  $K$ . Κατ' αυτό τον τρόπο δημιουργούνται οι συνθήκες κύλισης του δίσκου του Perkal κατά μήκος του προκύπτοντος  $\epsilon$ -τμήματος (γραμμή μπλε απόχρωσης). Η λύση θεωρείται αποδεκτή υπό την προϋπόθεση ότι η κάθετη απόσταση της κορυφής  $K\Sigma_A$  του αρχικού  $\epsilon$ -τμήματος από τη γενικευμένη γραμμή είναι μικρότερη από μία τιμή κατωφλίου των 0,25mm στον χάρτη (ή 250m στο έδαφος). Για τον καθορισμό της τιμής του κατωφλίου λαμβάνεται υπόψη το όριο διακριτικής ικανότητας  $D=0,2\text{mm}$  και μία τιμή ανοχής  $T=0,05\text{mm}$ , στον χάρτη κλίμακας 1:1.000.000. Αξίζει να σημειωθεί πως η νέα εξομάλυνση, λόγω του αναλογικά μικρού βαθμού της, επιφέρει ελάχιστες διαφοροποιήσεις στην τελική μορφή των υπολοίπων  $\epsilon$ -μη-κυρτών τμημάτων (βλ. Σχήμα 5.12).



**Σχήμα 5.11:** Παράδειγμα επίλυσης προβλήματος ευαναγνωσιμότητας σε 'διπλό' αμφίπλευρο  $\epsilon$ -μη-κυρτό-τμήμα.

Στο Σχήμα 5.12 παρουσιάζονται τα επτά  $\epsilon$ -μη-κυρτά τμήματα μετά τους μετασχηματισμούς που έλαβαν χώρα για τη γενίκευσή τους. Επισημαίνεται πως για λόγους ευκολότερης παρατήρησης απεικονίζονται τα  $\epsilon$ -τμήματα μόνο στην αρχική και στην τελική (μετά και τη διαδικασία εξομάλυνσης) μορφή τους, χωρίς τους ενδιάμεσους μετασχηματισμούς.



Σχήμα 5.12: Νέα ε-μη-κυρτά τμήματα μετά τη γενίκευση.

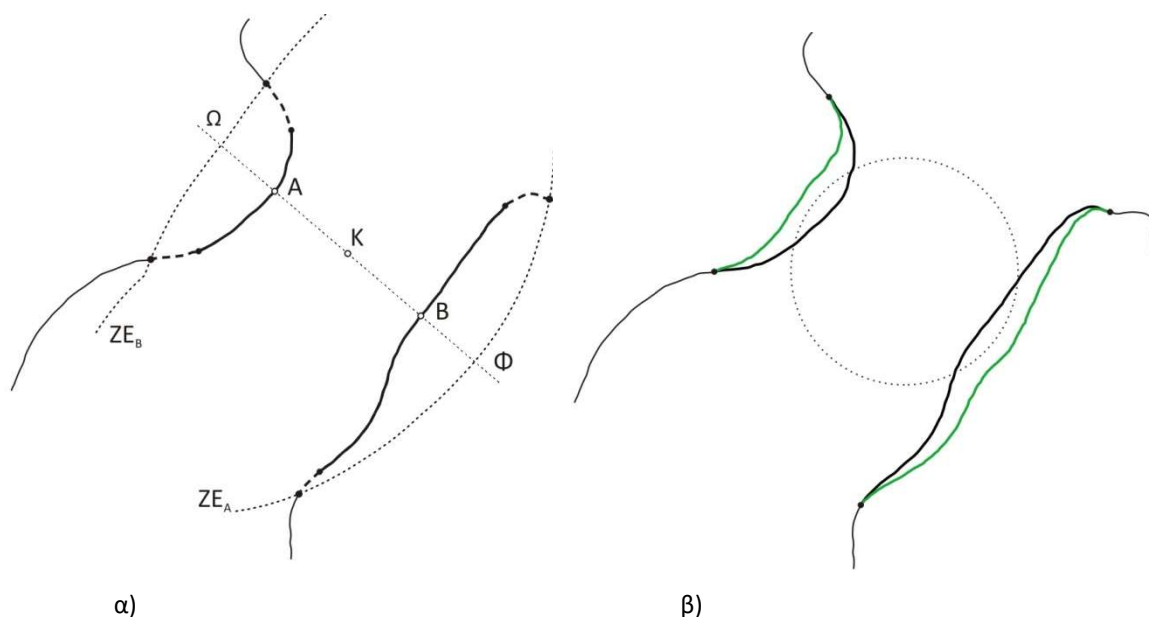
### 5.3.1.2.3 Γενίκευση τμημάτων σύγκλισης

Κατά μήκος της ακτογραμμής της νήσου Περιστεράς αποκόπτονται δύο τμήματα σύγκλισης, τα οποία διαμορφώνουν μία ομώνυμη περιοχή. Στο πρώτο στάδιο επεξεργασίας, ανευρίσκονται και αναλύονται τα χαρακτηριστικά της περιοχής σύγκλισης, βάσει των οποίων καθορίζονται οι παράμετροι γενίκευσης. Μέσω του μοντέλου που έχει δημιουργηθεί για το σκοπό αυτό (βλ. Σχήμα 4.17), υπολογίζεται η ελάχιστη απόσταση μεταξύ των δύο τμημάτων ( $Min\_Distance = 255,87m$ , στο έδαφος) και εντοπίζονται οι κορυφές τους (A, B) και το κέντρο της περιοχής σύγκλισης (K) (Σχήμα 5.13α). Παράλληλα, υπολογίζεται η διεύθυνση της νοητής ευθείας ΚΛ που διέρχεται από τις δύο κορυφές. Στη συνέχεια, υλοποιείται το μοντέλο διεύρυνσης της περιοχής σύγκλισης (βλ. Σχήμα 4.19). Όπως παρουσιάζεται στο Σχήμα 5.13α, γύρω από τα δύο τμήματα σύγκλισης φέρονται ζώνες επιρροής



( $ZE_A$ ,  $ZE_B$ ), με εύρος ίσο με την διάμετρο εφαρμογής  $\epsilon$ , δηλαδή 350m στο έδαφος ή 0,35mm στον χάρτη (βλ. κεφ.4.3.1). Οι ζώνες επιρροής τέμνουν την ακτογραμμή σε θέσεις εκατέρωθεν των τμημάτων. Τα μέρη των γειτονικών  $\epsilon$ -τμημάτων που εντοπίζονται εντός των ζωνών επιρροής (διακεκομμένες γραμμές) αποκόπτονται και συνενώνονται με τα αρχικά (συνεχείς γραμμές), διαμορφώνοντας τα διευρυμένα τμήματα σύγκλισης. Τα τμήματα των γειτονικών  $\epsilon$ -τμημάτων που εντοπίζονται εκτός ζωνών επιρροής συστήνουν την τελική μορφή αυτών.

Τα προκύπτοντα τμήματα υπόκεινται σε μία διαδικασία συμπίεσης με την εφαρμογή του αλγορίθμου *Depress*. Με βάση τα δεδομένα που προέκυψαν από την ανάλυση της περιοχής σύγκλισης, η τιμή του συντελεστή συμπίεσης ( $Depress = \frac{\epsilon - Min\_Distance}{2}$ ) ορίζεται 47,065m (στο έδαφος), η μετατόπιση των σημείων πραγματοποιείται παράλληλα της ευθείας  $\Omega\Phi$  και σε κατεύθυνση ανάλογη της θέσης της κορυφής του εκάστοτε τμήματος ως προς το κέντρο της περιοχής σύγκλισης. Το αποτέλεσμα της εφαρμογής παρουσιάζεται με γραμμή πράσινης απόχρωσης στο Σχήμα 5.13β.



**Σχήμα 5.13:** Προεπεξεργασία και γενίκευση τμημάτων σύγκλισης.

Παρατηρείται πως τα τμήματα που διαμορφώνουν την περιοχή σύγκλισης έχουν μετατοπισθεί κατάλληλα, ώστε να διέρχεται από αυτή ο θεωρητικός δίσκος του Perkal, γεγονός που εξασφαλίζει την οπτική ευαναγνωσιμότητά της στην παράγωγη κλίμακα. Επιπρόσθετα, τα γενικευμένα τμήματα διατηρούν την καμπυλότητα των αρχικών, ενώ στις θέσεις σύνδεσης με τα γειτονικά δεν εμφανίζονται ασυνέχειες. Διασφαλίζεται, έτσι, η γεωμετρική και 'αισθητική' εγκυρότητα του τελικού προϊόντος. Επισημαίνεται πως οι γραμμές του Σχήματος 5.13 αποδίδονται σε μεγαλύτερη κλίμακα, για την καλύτερη παρατήρηση των αποτελεσμάτων των διεργασιών που λαμβάνουν χώρα κατά την υλοποίηση του μοντέλου γενίκευσης.

### 5.3.1.2.4 Γενίκευση $\epsilon$ -κυρτών τμημάτων

Η επεξεργασία στην οποία υπόκεινται τα δεκαπέντε  $\epsilon$ -κυρτά τμήματα της ακτογραμμής, ώστε να αποδοθούν στον παράγωγο χάρτη, συνίσταται στην απλοποίηση τους, η οποία πραγματοποιείται με την εφαρμογή του αλγορίθμου 'bendsimplify' (βλ. κεφ. 3.6). Η δομή και η λειτουργία του συγκεκριμένου αλγορίθμου έχει ως αποτέλεσμα τη απαλοιφή της λεπτομέρειας και τη διατήρηση της καμπυλότητας των γραμμών, γεγονός που συνάδει με τη φιλοσοφία του μοντέλου γενίκευσης. Επίσης, σύμφωνα με παλαιότερες μελέτες (Nakos and Mitropoulos 2005), ο αλγόριθμος μπορεί να θεωρηθεί κατάλληλος για την απεικόνιση γραμμών σε πολύ μικρές κλίμακες, όπως συμβαίνει με το παρόν σενάριο γενίκευσης.

Η επιλογή της παραμέτρου εφαρμογής (ανοχή) πραγματοποιείται ανεξάρτητα για κάθε  $\epsilon$ -κυρτό τμήμα, βάσει εμπειρικών κριτηρίων. Παράγοντες που λαμβάνονται υπόψη είναι η μορφή του  $\epsilon$ -τμήματος, καθώς και ο αριθμός των σημείων που το δομούν. Παράλληλα, επιδιώκεται οι παράγωγες γραμμές να διατηρούν την καμπυλότητα των αρχικών και να μη δημιουργούνται ασυνέχειες στις θέσεις σύνδεσης με τα γειτονικά  $\epsilon$ -τμήματα. Τα δεδομένα απλοποίησης των δεκαπέντε  $\epsilon$ -κυρτών τμημάτων παρουσιάζονται στον Πίνακα 5.11.

Α/Α $\epsilon$ -τμήματος	Αριθμός σημείων $\epsilon$ -τμήματος		Ανοχή (m)
	Αρχικό	Γενικευμένο	
1	135	30	145
2	470	39	390
3	195	2	1200
4	81	2	250
5	118	2	500
6	114	2	250
7	355	2	1000
8	57	4	80
9	72	2	1000
10	209	52	210
11	222	13	390
12	412	42	290
13	456	58	230
14	103	13	120
15	55	2	1000

**Πίνακας 5.11:** Δεδομένα απλοποίησης  $\epsilon$ -κυρτών τμημάτων.

Στο Σχήμα 5.14 παρουσιάζονται τρία χαρακτηριστικά παραδείγματα γενίκευσης των  $\epsilon$ -κυρτών τμημάτων (γραμμές πράσινης απόχρωσης). Οι γραμμές αποδίδονται σε μεγαλύτερη κλίμακα για την ευκολότερη παρατήρηση του αποτελέσματος.



Σχήμα 5.14: Παραδείγματα γενίκευσης  $\epsilon$ -κυρτών τμημάτων.

#### 5.3.1.2.5 Συνένωση γενικευμένων $\epsilon$ -τμημάτων - Εξομάλυνση γενικευμένης γραμμής - Δημιουργία γενικευμένης γραμμής

Με την ολοκλήρωση των διεργασιών γενίκευσης, τα 32  $\epsilon$ -τμήματα έχουν υποστεί την κατάλληλη επεξεργασία, ανάλογα με την κατηγορία (Α, Β, Γ ή Δ) στην οποία ανήκουν και τα ιδιαίτερα χαρακτηριστικά τους, ώστε να αποδίδονται χαρτογραφικά και αισθητικά ορθά στο χάρτη κλίμακας 1:1.000.000. Επόμενο στάδιο είναι η συνένωσή τους, ώστε να ανασυνθέσουν τη γενικευμένη, πλέον, εκδοχή της ακτογραμμής της νήσου Περιστεράς. Η διαδικασία πραγματοποιείται στο περιβάλλον του λογισμικού πακέτου ArcGIS®, μέσω του μοντέλου που έχει δημιουργηθεί (βλ. κεφ. 4.5, Σχήμα 4.22).

Εξετάζοντας την προκύπτουσα ακτογραμμή, εντοπίζονται θέσεις όπου το αποτέλεσμα χρήζει διορθώσεων. Τα προβλήματα αφορούν, κυρίως, ασυνέχειες που εμφανίζονται σε θέσεις σύνδεσης των  $\epsilon$ -τμημάτων ή αιχμές και απότομα σπασίματα κατά μήκος της γραμμής. Τέτοιου είδους ατέλειες θεωρούνται αναμενόμενες, αφού οφείλονται στους σύνθετους και διαδοχικούς μετασχηματισμούς που έχουν υποστεί η γραμμή και τα  $\epsilon$ -τμήματα κατά την επεξεργασία τους. Μολονότι το μέγεθος και η σπουδαιότητά τους ως προς το σύνολο της γραμμής δεν είναι τέτοια που να επηρεάζουν σημαντικά το αποτέλεσμα, κρίνεται σκόπιμη η διόρθωσή τους, ώστε να βελτιωθεί αισθητικά το τελικό προϊόν. Αυτό επιτυγχάνεται με την εξομάλυνση, σε μικρό βαθμό, της ακτογραμμής (βλ. κεφ. 4.5, Σχήμα 4.24). Η διαδικασία υλοποιείται με τη χρήση του τελεστή Gauss, με τιμές της μεταβλητής  $\sigma=9$  στην πρώτη



Σχήμα 5.15: Ακτογραμμή της νήσου Περιστεράς όπως προκύπτει από την εφαρμογή του μοντέλου για σενάριο γενίκευσης 1:1.000.000.

εκδοχή της γενικευμένης ακτογραμμής και  $\sigma=10$  σε εκείνη που προέκυψε μετά τις διορθώσεις των  $\epsilon$ -μη-κυρτών τμημάτων που δημιουργήθηκαν μετά τον έλεγχο ευαναγνωσιμότητας (βλ. κεφ. 5.3.1.2.2). Ο τελευταίος επαναλαμβάνεται (βλ. κεφ. 4.6), με το αποτέλεσμα του να πιστοποιεί την ευκρινή απόδοση της ακτογραμμής στην κλίμακα 1:1.000.000, βάσει των κανόνων οπτικής αντίληψης που έχουν ορισθεί. Με την ολοκλήρωση της διεργασίας προκύπτει η τελική, γενικευμένη μορφή της ακτογραμμής της νήσου Περιστεράς, όπως παρουσιάζεται στα Σχήματα 5.15 και 5.23α. Να σημειωθεί πως η ακτογραμμή παρουσιάζεται σε μεγέθυνση, με προσαρμοσμένο πάχος γραμμής, για την ευκολότερη παρατήρηση και αξιολόγηση του αποτελέσματος.

### 5.3.2 Σενάριο γενίκευσης 1:500.000

#### 5.3.2.1 Κατάτμηση γραμμής

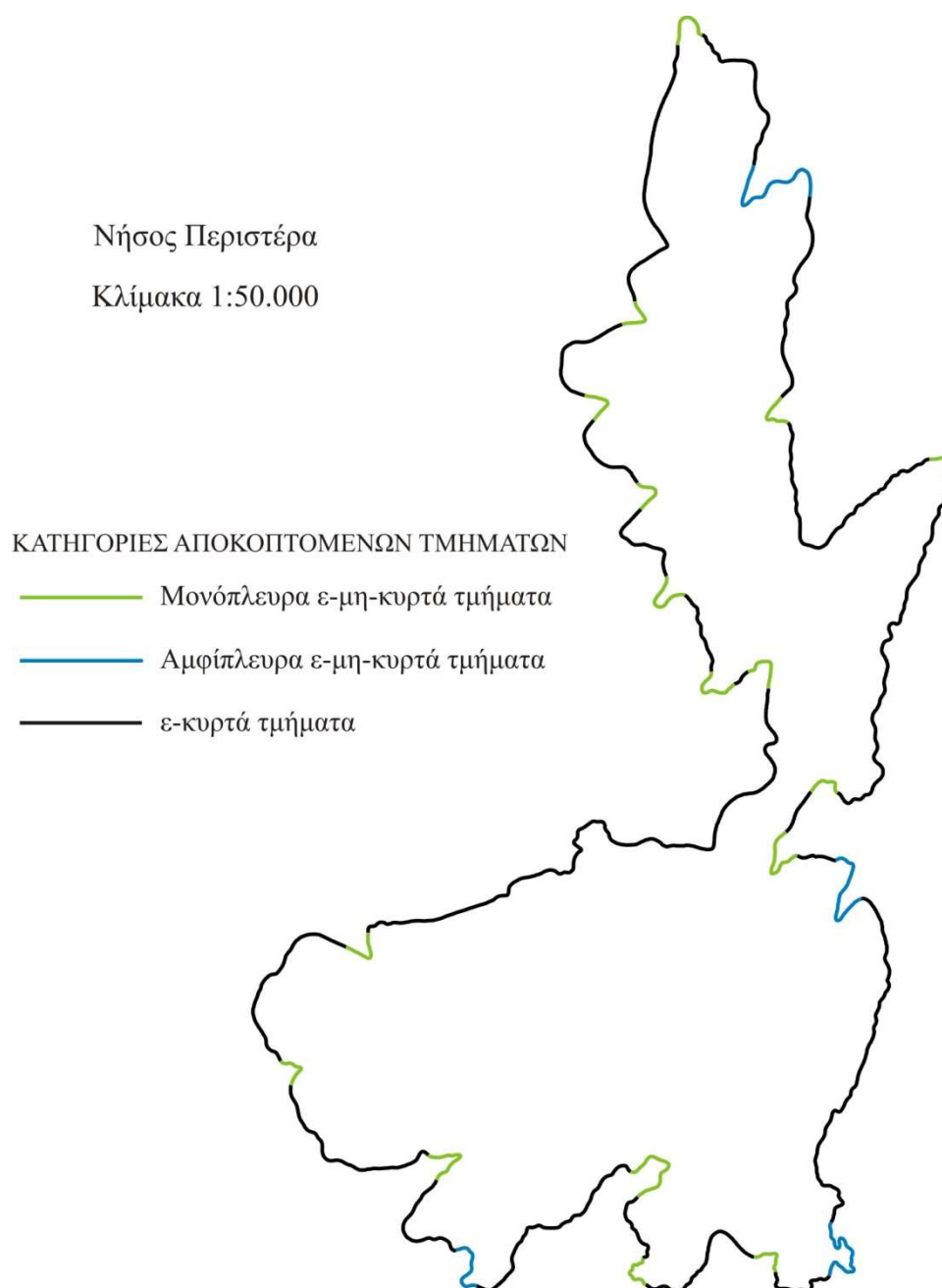
Η τεχνική κατάτμησης υλοποιείται στην αρχική γραμμή της νήσου Περιστεράς για σενάριο γενίκευσης, στο οποίο ο παράγωγος χάρτης είναι κλίμακας 1:500.000. Οι διεργασίες που ακολουθούνται αναλύονται στα κεφάλαια 2.3.2, 2.3.3 και 2.3.4 και είναι αντίστοιχες με εκείνες που περιγράφονται εκτενέστερα στο κεφάλαιο 5.3.1.1 (εφαρμογή για σενάριο γενίκευσης 1:1.000.000).

Για τον ορισμό των παραμέτρων εφαρμογής λαμβάνονται υπόψη: α) το πάχος της γραμμής  $W$ , το οποίο σε ένα χάρτη κλίμακας 1:500.000 είναι, συνήθως,  $W=0,1\text{mm}$ , β) το όριο οπτικής αντίληψης  $D=0,2\text{mm}$  (Rouleau 1984) και γ) η τιμή ανοχής  $T=0,05\text{mm}$  (βλ. κεφ. 2.2). Βάσει αυτών ορίζονται η διάμετρος εφαρμογής ( $\epsilon=W+D+T$ )  $\epsilon=0,175\text{mm}$  στον χάρτη ή  $\epsilon=175\text{m}$  στο έδαφος, το ελάχιστο εμβαδό που μπορεί να γίνει αντιληπτό από τον αναγνώστη  $W^2=0,01\text{mm}^2$  στον χάρτη ή  $2500\text{m}^2$  στο έδαφος και το ελάχιστο ανάπτυγμα ενός γραμμικού τμήματος που μπορεί να γίνει αντιληπτό από τον αναγνώστη  $D+T=0,25\text{mm}$  στον χάρτη ή  $125\text{m}$  στο έδαφος.

Κατά την κύλιση του θεωρητικού δίσκου του Perkal αποκόπτονται 678  $\epsilon$ -μη-κυρτά τμήματα (394 εσωτερικά και 284 εξωτερικά της ακτογραμμής). Τα τμήματα υπόκεινται στη διαδικασία φιλτραρίσματος, ώστε να εξαιρεθούν εκείνα των οποίων το εμβαδό είναι μικρότερο από  $2500\text{m}^2$  στο έδαφος. Έτσι, προς περαιτέρω επεξεργασία επιλέγονται 26  $\epsilon$ -μη-κυρτά τμήματα (9 εσωτερικά και 17 εξωτερικά της ακτογραμμής), ενώ τα υπόλοιπα συνενώνονται με τα  $\epsilon$ -κυρτά, δημιουργώντας 24  $\epsilon$ -κυρτά τμήματα. Εν συνεχεία, τα  $\epsilon$ -μη-κυρτά τμήματα μεταξύ των οποίων παρεμβάλλονται  $\epsilon$ -κυρτά με ανάπτυγμα μικρότερο των  $125\text{m}$  στο έδαφος συγχωνεύονται. Με την ολοκλήρωση της διαδικασίας συγχώνευσης, η ακτογραμμή διαμορφώνεται από 23  $\epsilon$ -μη-κυρτά και 22  $\epsilon$ -κυρτά τμήματα.

Στο επόμενο στάδιο επεξεργασίας τα  $\epsilon$ -μη-κυρτά τμήματα διαχωρίζονται στις επιμέρους κατηγορίες, ανάλογα με τη μορφή και τον τρόπο δημιουργίας τους. Έτσι, τελικά, κατά μήκος της ακτογραμμής αποκόπτονται 19 μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα (κατηγορία Α), 3 αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα (κατηγορία Β) και 22  $\epsilon$ -κυρτά τμήματα (κατηγορία Δ). Τέλος, για τις ανάγκες γενίκευσης των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων λαμβάνει χώρα η διεύρυνσή τους, προσαρτώντας μέρος των γειτονικών  $\epsilon$ -τμημάτων (βλ. κεφ. 4.1.1). Αυτό έχει ως αποτέλεσμα τη

δημιουργία  $\epsilon$ -κυρτών τμημάτων με ανάπτυγμα μικρότερο από 125m στο έδαφος. Τα  $\epsilon$ -τμήματα αυτά συγχωνεύονται με τα εκατέρωθεν γειτονικά. Με την ολοκλήρωση της διαδικασίας προκύπτουν τα τελικά  $\epsilon$ -τμήματα στα οποία αποκόπτεται η ακτογραμμή, τα οποία είναι 17 μονόπλευρα, 4 αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα και 21  $\epsilon$ -κυρτά τμήματα. Το αποτέλεσμα της εφαρμογής του μοντέλου κατάτμησης στην ακτογραμμή της νήσου Περιστέρας παρουσιάζεται στο Σχήμα 5.16.



**Σχήμα 5.16:** Αποτέλεσμα εφαρμογής του μοντέλου κατάτμησης στην ακτογραμμή της νήσου Περιστέρας για σενάριο γενίκευσης 1:500.000.

### 5.3.2.2 Γενίκευση γραμμής

#### 5.3.2.2.1 Γενίκευση μονόπλευρων $\epsilon$ -μη-κυρτών τμημάτων

Για την ευκρινή απόδοσή τους στην κλίμακα 1:500.000, τα 17 μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα υπόκεινται διαδοχικά στους μετασχηματισμούς διεύρυνσης, εξομάλυνσης και διόγκωσης. Οι κανόνες και η λογική πάνω στην οποία στηρίζεται η επιλογή των παραμέτρων εφαρμογής είναι αντίστοιχη με εκείνη που περιγράφεται στο κεφάλαιο 5.3.1.2.1(σενάριο γενίκευσης 1:1.000.000).

Η διεύρυνση πραγματοποιείται κατά τρόπο, ώστε η γραμμή βάσης των  $\epsilon$ -τμημάτων να είναι ίση με τη διάμετρο του δίσκου του Perkal ( $L_{BL}=\epsilon$ ), όπου  $\epsilon=175\text{m}$  στο έδαφος (ή  $\epsilon=0,175\text{mm}$  στον χάρτη).Εν συνεχεία, τα διευρυμένα  $\epsilon$ -τμήματα εξομαλύνονται, αφενός για να εξαλειφθούν οι τοπικές ανωμαλίες της αρχικής γραμμής και αφετέρου, για να διαμορφωθεί η δυνατόν βέλτιστη γεωμετρία στη 'θέση Κ', από την οποία εξαρτάται ο μετασχηματισμός διόγκωσης που ακολουθεί. Η διαδικασία λαμβάνει χώρα εφαρμόζοντας δοκιμαστικά τον τελεστή *Gauss* και εντοπίζοντας κάθε φορά τις καμπές στο εξομαλυσμένο  $\epsilon$ -τμήμα μέσω της μεθόδου που στηρίζεται στον ορισμό των Wang και Müller (1998). Για την επιλογή της κατάλληλης τιμής της μεταβλητής  $\sigma$  του τελεστή *Gauss* χρησιμοποιούνται ως 'οδηγός' η μέγιστη ( $\sigma_{max}$ ) και η ελάχιστη με την οποία εντοπίζεται ο ελάχιστος αριθμός καμπών ( $\sigma_{min1}$ ). Από την πειραματική διαδικασία προέκυψε πως για τιμές μικρότερες της  $\sigma_{min1}$ , το αποτέλεσμα της εξομάλυνσης δεν είναι επαρκές για τις ανάγκες της εφαρμογής. Στον Πίνακα 5.12 παρουσιάζονται οι παράμετροι υλοποίησης του τελεστή εξομάλυνσης κατά την εφαρμογή του στα 17 μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα.

A/A $\epsilon$ -τμήματος	αριθμός αρχικών καμπών	$\sigma_{max}$	ελάχιστος αριθμός καμπών	$\sigma_{min1}$	$\sigma$ εφαρμογής
1	16	9	3	5.25	<b>9</b>
2	24	7	3	2	<b>7</b>
3	24	8.5	2	8.5	<b>8.5</b>
4	15	6.75	2	5.5	<b>6.75</b>
5	12	6.5	3	2.5	<b>6.5</b>
6	13	6.25	3	2	<b>6.25</b>
7	25	10.75	3	2.25	<b>10.75</b>
8	21	7.5	3	3.25	<b>7.5</b>
9	23	9.5	2	8.75	<b>9.5</b>
10	16	7	2	4.75	<b>7</b>
11	28	13	2	10.25	<b>13</b>
12	25	7.5	3	3.25	<b>7.5</b>
13	18	7.25	3	4.75	<b>7.25</b>
14	31	11.5	3	8.5	<b>11.5</b>
15	47	16.5	2	15.25	<b>16.5</b>
16	18	7.5	3	4	<b>7.5</b>
17	13	8.25	3	4.25	<b>8.25</b>

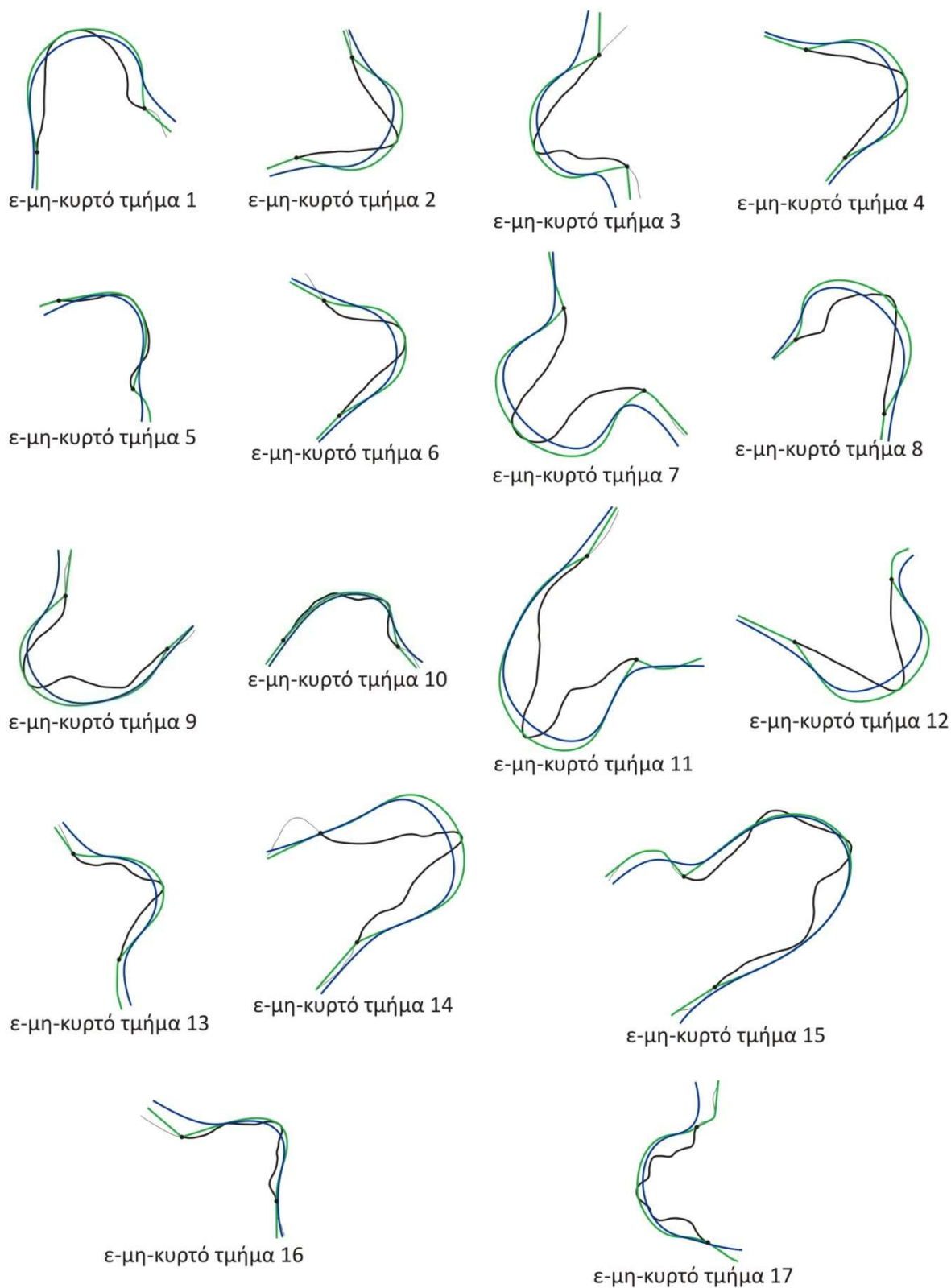
**Πίνακας 5.12:** Αριθμός καμπών και παράμετροι εφαρμογής του τελεστή εξομάλυνσης στα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα (σενάριο γενίκευσης 1:500.000).

Μετά την επιλογή του βαθμού εξομάλυνσης, εντοπίζονται τα σημεία ΚΣ<sub>Ε</sub> (με την εφαρμογή του δείκτη αναλογίας μήκους ARL), ΚΣ<sub>Α</sub> και Κ, τα οποία διαμορφώνουν τη γεωμετρία στη 'θέση Κ'. Λαμβάνοντας αυτήν ως δεδομένο, πραγματοποιείται η διόγκωση των εξομαλυσμένων  $\epsilon$ -τμημάτων, εφαρμόζοντας έναν από τους αλγόριθμους 'Δεσμευμένης Διόγκωσης' ή 'Ανεξάρτητης Διόγκωσης', ώστε να δημιουργηθούν οι προϋποθέσεις ευαναγνωσιμότητας στην κλίμακα 1.500.000.

Στον Πίνακα 5.13 παρουσιάζονται οι παράμετροι που χρησιμοποιήθηκαν για τη διόγκωση των 17 μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων. Το αποτέλεσμα της επεξεργασίας για τη γενίκευση των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων παρουσιάζεται στο Σχήμα 5.17. Με γραμμή πράσινης απόχρωσης αποδίδονται τα  $\epsilon$ -τμήματα όπως έχουν προκύψει από τους μετασχηματισμούς γενίκευσης και με γραμμή μπλε απόχρωσης όπως διαμορφώνονται μετά την εξομάλυνση του συνόλου της ακτογραμμής (βλ. κεφ. 5.3.2.2.4). Επισημαίνεται πως οι γραμμικές οντότητες αναπαριστώνται μεγεθυμένες για την καλύτερη παρατήρηση των αποτελεσμάτων.

A/A $\epsilon$ -τμήματος	τιμή συντελεστή 'Inflating' αλγορίθμου 'Δεσμευμένης Διόγκωσης' (m)	αλγόριθμος Διόγκωσης που εφαρμόζεται	τιμή συντελεστή 'Inflating' αλγορίθμου 'Ανεξάρτητης Διόγκωσης' (m)
1	17.47	Ανεξάρτητη	70
2	17.89	Ανεξάρτητη	70
3	19.20	Ανεξάρτητη	70
4	14.61	Ανεξάρτητη	70
5	9.56	Δεσμευμένη	
6	11.12	Ανεξάρτητη	50
7	21.50	Ανεξάρτητη	100
8	16.44	Ανεξάρτητη	90
9	26.05	Ανεξάρτητη	60
10	5.39	Δεσμευμένη	
11	37.84	Ανεξάρτητη	90
12	19.08	Ανεξάρτητη	90
13	17.50	Ανεξάρτητη	40
14	32.22	Ανεξάρτητη	90
15	42.53	Δεσμευμένη	
16	0.92	Ανεξάρτητη	20
17	1.30	Ανεξάρτητη	70

**Πίνακας 5.13:** Παράμετροι διόγκωσης μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).



**Σχήμα 5.17:** Τα μονόπλευρα ε-μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης (σενάριο γενίκευσης 1:500.00).



### 5.3.2.2.2 Γενίκευση αμφίπλευρων $\epsilon$ -μη-κυρτών τμημάτων

Για την ευκρινή απόδοσή τους στην κλίμακα 1:500.000, τα τέσσερα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα υπόκεινται διαδοχικά στους μετασχηματισμούς απλοποίησης, διόγκωσης των κύριων καμπών που εντοπίζονται και επανένωσής τους για τη δημιουργία του γενικευμένου  $\epsilon$ -τμήματος. Οι διεργασίες που ακολουθούνται είναι αντίστοιχες με εκείνες που περιγράφονται στο κεφάλαιο 5.3.1.2.1 (σενάριο γενίκευσης 1:1.000.000).

Η απλοποίηση περιλαμβάνει, σε συνδυασμό, την εξομάλυνση και τον εντοπισμό καμπών κατά μήκος των εξομαλυσμένων  $\epsilon$ -τμημάτων. Στόχος είναι η εξάλειψη των λεπτομερειών της αρχικής γραμμής και επιλογή των κύριων καμπών, οι οποίες είναι απαραίτητες για τη διαμόρφωση των  $\epsilon$ -τμημάτων στην κλίμακα 1:500.000. Μέτρο για την αναζήτηση των κύριων καμπών τίθεται η αναλογία της γραμμής βάσης ( $L_{BL}$ ) ως προς τη διάμετρο εφαρμογής  $\epsilon$  του θεωρητικού δίσκου του Perkal και συγκεκριμένα  $L_{BL} \geq 175m$  (στο έδαφος). Η διαδικασία λαμβάνει χώρα εφαρμόζοντας δοκιμαστικά τον τελεστή εξομάλυνσης *Gauss* και εντοπίζοντας κάθε φορά τις καμπές στο εξομαλυσμένο  $\epsilon$ -τμήμα μέσω της μεθόδου που στηρίζεται στον ορισμό των Wang και Müller (1998). Ως 'οδηγός' για την εφαρμογή χρησιμοποιούνται τρεις χαρακτηριστικές τιμές της μεταβλητής  $\sigma$  του τελεστή *Gauss*: η μέγιστη ( $\sigma_{max}$ ), η ελάχιστη με την οποία εντοπίζεται ο ελάχιστος αριθμός καμπών ( $\sigma_{min1}$ ) και η ελάχιστη με την οποία εντοπίζεται ο αμέσως επόμενος αριθμός καμπών ( $\sigma_{min2}$ ). Στον Πίνακα 5.14 παρουσιάζονται ο αριθμός καμπών και οι τιμές της μεταβλητής  $\sigma$  που χρησιμοποιούνται.

A/A $\epsilon$ -τμήματος	αριθμός αρχικών καμπών	$\sigma_{max}$	ελάχιστος αριθμός καμπών	$\sigma_{min1}$	επόμενος αριθμός καμπών	$\sigma_{min2}$	$\sigma$ εφαρμογής	αριθμός κύριων καμπών
1	21	12	2	8.5	3	5.75	<b>12</b>	<b>2</b>
2	45	17.25	2	16.75	3	6.25	<b>17.25</b>	<b>2</b>
3	40	21	3	8.75	5	7.25	<b>21</b>	<b>3</b>
4	55	22.75	2	14.75	4	12.5	<b>22.75</b>	<b>2</b>

**Πίνακας 5.14:** Αριθμός καμπών και παράμετροι εφαρμογής εξομάλυνσης στα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα (σενάριο γενίκευσης 1:500.000).

Όπως παρατηρείται στον Πίνακα 5.14, ο αριθμός των κύριων καμπών που επιλέγονται ταυτίζεται με τον ελάχιστο αριθμό καμπών που μπορούν να εντοπισθούν εφαρμόζοντας το μέγιστο βαθμό εξομάλυνσης. Κατ' αυτό τον τρόπο, τα  $\epsilon$ -τμήματα αποκόπτονται σε καμπές με το δυνατόν μεγαλύτερο μήκος γραμμής βάσης. Επίσης παρατηρείται πως, σε όλες τις περιπτώσεις, η τιμή εφαρμογής της μεταβλητής  $\sigma$  ταυτίζεται με τη μέγιστη ( $\sigma_{max}$ ). Η επιλογή έγινε μετά από δοκιμές, λαμβάνοντας υπόψη τα ποσοτικά χαρακτηριστικά των προκυπτουσών καμπών, αλλά και συνεκτιμώντας κάθε φορά το αποτέλεσμα διόγκωσης. Πιο συγκεκριμένα, με τη μέγιστη εξομάλυνση, η γεωμετρία στη 'θέση Κ' που προκύπτει και για τα τέσσερα  $\epsilon$ -τμήματα, δημιουργεί τις προϋποθέσεις για τη δυνατόν αποτελεσματικότερη διόγκωση των καμπών. Αντίθετα, οι μεταθέσεις των άκρων

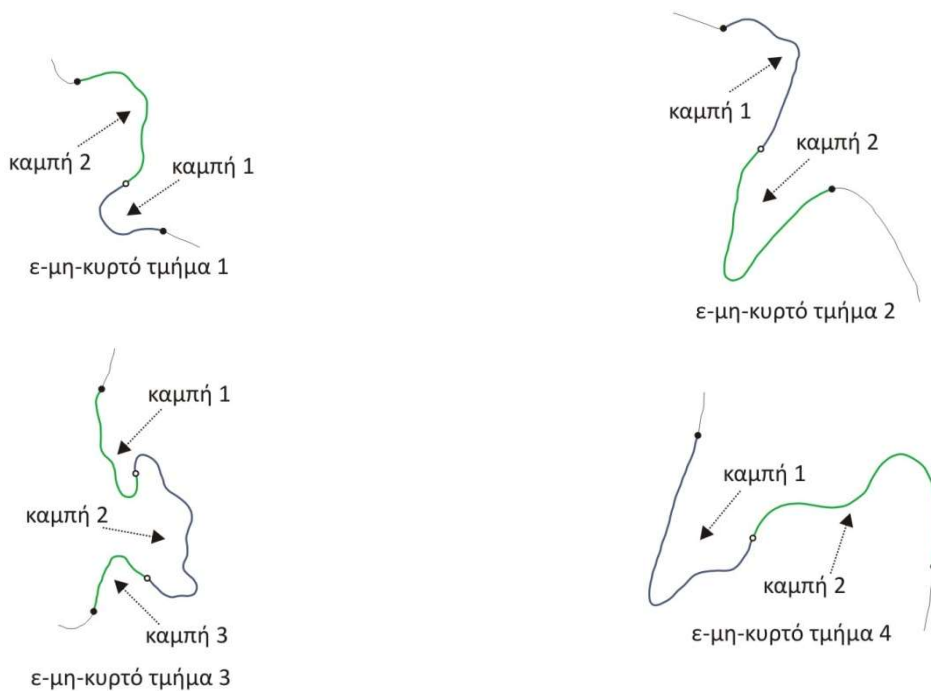
των καμπών (σημεία κάμψης) είναι αμελητέες και δεν μεταβάλουν το μήκος τηςγραμμής βάσης τους σε βαθμό που να επηρεάζεται το τελικό αποτέλεσμα.

Στον Πίνακα 5.15 αναγράφονται τα ποσοτικά χαρακτηριστικά των κύριων καμπών που προκύπτουν από την εφαρμογή του τελεστή *Gauss* με τις τιμές της μεταβλητής  $\sigma$  που επιλέχθηκαν. Παρατηρείται πως στην πλειοψηφία των περιπτώσεων, τα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα αποδομούνται σε κύριες καμπές των οποίων το μήκος της γραμμής βάσης υπερβαίνει ή προσεγγίζει το κριτήριο επιλογής ( $L_{BL} \geq 175m$  στο έδαφος). Εξάιρεση αποτελούν οι καμπές '1' και '3' των  $\epsilon$ -τμημάτων 1 και 3 αντίστοιχα. Το γεγονός οφείλεται στον περιορισμό που επιφέρει η εφαρμογή του τελεστή με *Gauss* συγκριμένες/πεπερασμένες τιμές της παραμέτρου  $\sigma$ , οι οποίες εξαρτώνται από τον αριθμό των σημείων που δομούν το  $\epsilon$ -τμήμα.

A/A $\epsilon$ -τμήματος	A/A καμπής	μήκος γραμμής βάσης $L_{BL}(m)$	βάθος (m)	εμβαδόν (m <sup>2</sup> )
1	1	99.85	42.16	2756.60
	2	175.16	62.38	7783.29
2	1	198.64	60.03	7867.18
	2	130.82	145.35	12298.13
3	1	153.10	13.39	1644.31
	2	161.83	41.67	6463.27
	3	97.23	9.01	561.85
4	1	187.18	122.24	14941.65
	2	285.36	94.94	18236.11

**Πίνακας 5.15:** Ποσοτικά χαρακτηριστικά κύριων καμπών αμφίπλευρων  $\epsilon$ -μη-κυρτώντμημάτων (σενάριο γενίκευσης 1:500.000).

Στο Σχήμα 5.18 παρουσιάζονται οι κύριες καμπές στις οποίες αποδομούνται τα οκτώ αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα, αν αντιστοιχισθούν τα σημεία κάμψης στην αρχική εκδοχή τους. Το αποτέλεσμα της διαδικασίας απλοποίησης κρίνεται αρκετά ικανοποιητικό, υπό την έννοια πως αν ζητούνταν από κάποιον παρατηρητή να διαχωρίσει τα  $\epsilon$ -τμήματα στις βασικές κυρτώσεις και κοιλώσεις τους, αυτές θα μπορούσαν να ταυτισθούν με τις κύριες καμπές που έχουν εντοπισθεί. Επισημαίνεται πως για την καλύτερη παρατήρηση, οι γραμμικές οντότητες των σχημάτων του κεφαλαίου αποδίδονται σε μεγέθυνση.



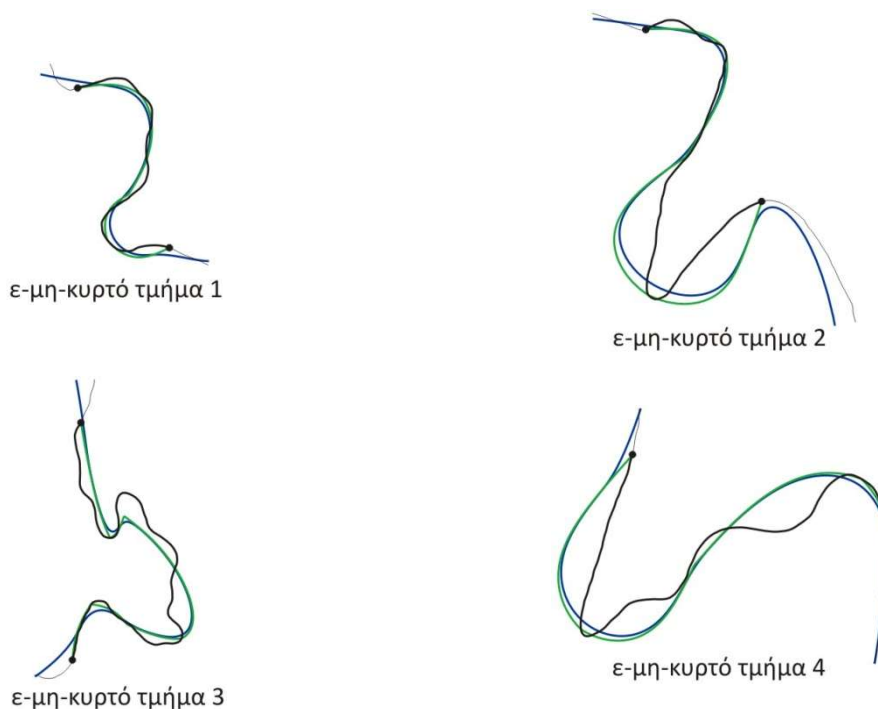
**Σχήμα 5.18:** Οι κύριες καμπές αντιστοιχισμένες στα αρχικά αμφίπλευρα ε-μη-κυρτά τμήματα (σενάριο γενίκευσης 1:500.000).

Στο επόμενο στάδιο επεξεργασίας η εξομαλυσμένη εκδοχή του εκάστοτε αμφίπλευρου ε-μη-κυρτού τμήματος υπόκειται στο μετασχηματισμό διόγκωσης. Η διαδικασία υλοποιείται ανεξάρτητα σε κάθε κύρια καμπή του ε-τμήματος. Έχοντας ως βάση τη γεωμετρία στη 'θέση Κ', πραγματοποιείται η διόγκωσή τους, εφαρμόζοντας έναν από τους αλγορίθμους 'Δεσμευμένης Διόγκωσης' ή 'Ανεξάρτητης Διόγκωσης', ώστε να δημιουργηθούν οι προϋποθέσεις ευαναγνωσιμότητας στην κλίμακα 1.500.000. Οι παράμετροι διόγκωσης των τεσσάρων αμφίπλευρων ε-μη-κυρτών τμημάτων παρουσιάζονται στον Πίνακα 5.16.

A/A ε-τμήματος	A/A καμπής	τιμή συντελεστή 'Inflating' αλγορίθμου 'Δεσμευμένης Διόγκωσης' (m)	αλγόριθμος Διόγκωσης που εφαρμόζεται	τιμή συντελεστή 'Inflating' αλγορίθμου 'Ανεξάρτητης Διόγκωσης' (m)
1	1	43.49	Δεσμευμένη	
	2	17.03	Δεσμευμένη	
2	1	19.33	Δεσμευμένη	
	2	59.82	Ανεξάρτητη	100
3	1	48.59	Δεσμευμένη	
	2	13.50	Δεσμευμένη	
	3	78.06	Δεσμευμένη	
4	1	76.04	Ανεξάρτητη	90
	2	72.30	Δεσμευμένη	

**Πίνακας 5.16:** Παράμετροι διόγκωσης αμφίπλευρων ε-μη-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).

Το αποτέλεσμα της επεξεργασίας για τη γενίκευση των αμφίπλευρων  $\epsilon$ -μη-κυρτών τμημάτων παρουσιάζεται στο Σχήμα 5.19. Με γραμμή πράσινης απόχρωσης παρουσιάζονται τα  $\epsilon$ -τμήματα όπως έχουν προκύψει από τους μετασχηματισμούς γενίκευσης και με γραμμή μπλε απόχρωσης όπως διαμορφώνονται μετά την εξομάλυνση του συνόλου της ακτογραμμής (βλ. κεφ. 5.3.2.2.4).

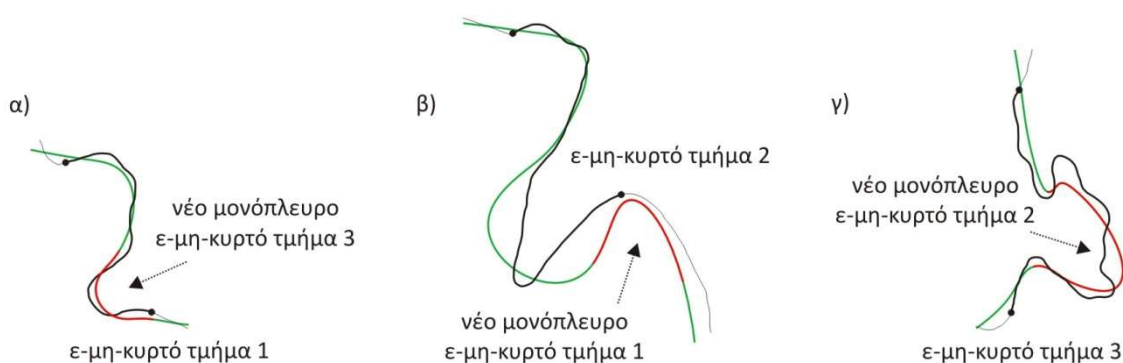


**Σχήμα 5.19:** Τα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα μετά τη διαδικασία γενίκευσης (σενάριο γενίκευσης 1:500.000).

Μετά την εφαρμογή του μοντέλου ελέγχου για την ευαναγνωσιμότητα των παράγωγων γραμμών (βλ. κεφ. 4.6) προέκυψαν τρία νέα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα. Το γεγονός καταδεικνύει πως οι θέσεις αυτές της παράγωγης ακτογραμμής δεν αποδίδονται ευκρινώς στην κλίμακα 1:500.000. Επειδή τα παραπάνω  $\epsilon$ -τμήματα εντοπίζονται σε θέσεις που σχετίζονται με τα αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα, η επεξεργασία για τη διόρθωση του προβλήματος περιγράφεται στο συγκεκριμένο κεφάλαιο.

Παρατηρώντας τα νέα  $\epsilon$ -μη-κυρτά τμήματα (γραμμές κόκκινης απόχρωσης) στο Σχήμα 5.20, συμπεραίνονται τα εξής: α) πρόβλημα ευαναγνωσιμότητας εμφανίζεται στην 'καμπή 1' του  $\epsilon$ -τμήματος 1 (Σχήμα 5.20α), της οποίας το μήκος της γραμμής βάσης είναι αρκετά μικρότερο της διαμέτρου εφαρμογής  $\epsilon$ , σε συνδυασμό με το, αναλογικά, μεγάλο βάθος της. Αξίζει να σημειωθεί πως το πρόβλημα υφίσταται και μετά τη διόγκωση της καμπής μέσω του αλγορίθμου 'Ανεξάρτητης Διόγκωσης'. β) ιδιαίτερο ενδιαφέρον παρουσιάζει η περίπτωση του  $\epsilon$ -μη-κυρτού τμήματος που συνδέεται με το  $\epsilon$ -τμήμα 2. Όπως φαίνεται στο Σχήμα 5.20β, ενώ το γενικευμένο  $\epsilon$ -τμήμα αποδίδεται ευκρινώς στην παράγωγη κλίμακα, δημιουργείται πρόβλημα στο γειτονικό  $\epsilon$ -κυρτό τμήμα. Το γεγονός οφείλεται σε

ένα συνδυασμό παραγόντων που έχουν να κάνουν με το βαθμό διόγκωσης της καμπής, το επίπεδο απλοποίησης του όμορου  $\epsilon$ -κυρτού τμήματος, το βαθμό εξομάλυνσης της γενικευμένης ακτογραμμής, σε συνάρτηση με τη γεωμετρία της γραμμής στη συγκεκριμένη θέση. Πρόκειται για ένα χαρακτηριστικό παράδειγμα του τρόπου με τον οποίο οι μετασχηματισμοί που υφίσταται το εκάστοτε  $\epsilon$ -τμήμα μπορεί να επηρεάσουν τα όμορα. γ) πρόβλημα ευαναγνωσιμότητας εμφανίζεται στην καμπή '2' του  $\epsilon$ -τμήματος 3. Όπως φαίνεται στο Σχήμα 5.20γ, το  $\epsilon$ -τμήμα στο σύνολό του χαρακτηρίζεται από έντονη πολυπλοκότητα. Λόγω της γεωμετρίας τους (αναλογικά μικρό βάθος και επιφάνεια), αλλά και της γεινιάσής τους με  $\epsilon$ -κυρτά τμήματα, οι καμπές '1' και '3' αποδίδονται ευκρινώς στην παράγωγη κλίμακα μετά τη διαδικασία διόγκωσης. Κάτι τέτοιο δεν επιτυγχάνεται για την ενδιάμεση καμπή '2', της οποίας το μήκος της γραμμής βάσης είναι μικρότερο από τη διάμετρο εφαρμογής  $\epsilon$ , ενώ συγχρόνως η επιφάνειά της είναι αναλογικά μεγάλη (βλ. Πίνακα 5.15). Αυτό έχει ως αποτέλεσμα να μην επιτυγχάνονται οι προϋποθέσεις εισχώρησης και κύλισης του θεωρητικού δίσκου του Perkal κατά μήκος της μετά τους αρχικούς μετασχηματισμούς.



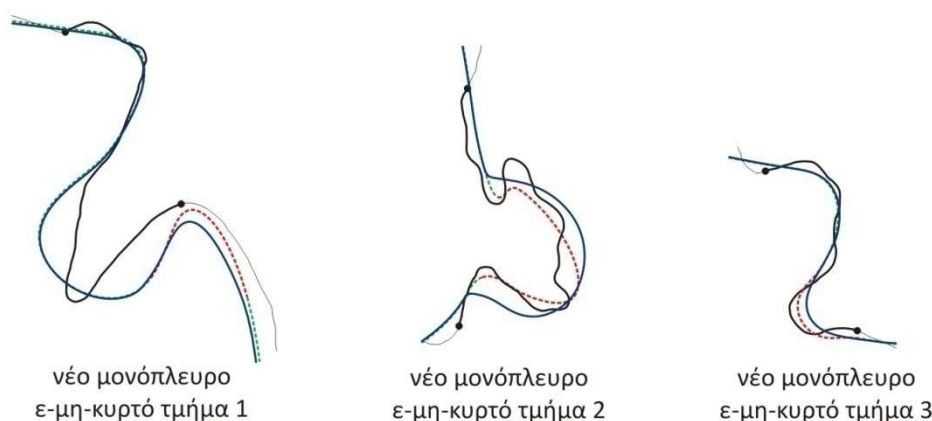
**Σχήμα 5.20:**  $\epsilon$ -μη-κυρτά τμήματα που προκύπτουν μετά τον έλεγχο ευαναγνωσιμότητας (σενάριο γενίκευσης 1:500.000).

Για την επίλυση του προβλήματος ευαναγνωσιμότητάς τους στην κλίμακα 1:500.000, τα τρία νέα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα αποκόπτονται από την υπόλοιπη γραμμή και υπόκεινται στις διεργασίες γενίκευσης αντίστοιχες με εκείνες των αρχικών  $\epsilon$ -μη-κυρτών τμημάτων. Στον Πίνακα 5.17 παρουσιάζονται οι παράμετροι γενίκευσης των τριών μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων.

A/A ε-τμήματος	$\sigma$ εφαρμογής τελεστή εξομάλυνσης	αλγόριθμος Διόγκωσης που εφαρμόζεται	τιμή συντελεστή 'Inflating' αλγορίθμου Διόγκωσης (m)
1	5.5	Δεσμευμένη	12.87
2	14.5	Ανεξάρτητη	70
3	7.5	Δεσμευμένη	10.66

**Πίνακας 5.17:** Παράμετροι γενίκευσης νέων μονόπλευρων ε-μη-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).

Στο Σχήμα 5.21 παρουσιάζονται τα τρία ε-μη-κυρτά τμήματα μετά τους μετασχηματισμούς που έλαβαν χώρα για τη γενίκευσή τους. Επισημαίνεται πως για λόγους ευκολότερης παρατήρησης απεικονίζονται τα ε-τμήματα μόνο στην αρχική και στην τελική (μετά και τη διαδικασία εξομάλυνσης) μορφή τους, χωρίς τους ενδιάμεσους μετασχηματισμούς.



**Σχήμα 5.21:** Νέα ε-μη-κυρτά τμήματα μετά τη γενίκευση (σενάριο γενίκευσης 1:500.000).

### 5.3.2.2.3 Γενίκευση ε-κυρτών τμημάτων

Η επεξεργασία στην οποία υπόκεινται τα 21 ε-κυρτά τμήματα της ακτογραμμής, ώστε να αποδοθούν στον παράγωγο χάρτη, συνίσταται στην απλοποίηση τους, η οποία πραγματοποιείται με την εφαρμογή του αλγορίθμου 'bendsimplify' (βλ. κεφ. 3.6). Η επιλογή της παραμέτρου εφαρμογής (ανοχή) πραγματοποιείται ανεξάρτητα για κάθε ε-κυρτό τμήμα, βάσει εμπειρικών κριτηρίων. Παράγοντες που λαμβάνονται υπόψη είναι η μορφή του ε-τμήματος, καθώς και ο αριθμός των σημείων που το δομούν. Παράλληλα, επιδιώκεται οι παράγωγες γραμμές να διατηρούν την καμπυλότητα των αρχικών και να μη δημιουργούνται ασυνέχειες στις θέσεις σύνδεσης με τα γειτονικά ε-τμήματα. Τα δεδομένα απλοποίησης των 21 ε-κυρτών τμημάτων παρουσιάζονται στον Πίνακα 5.18.

Α/Α €-τμήματος	Αριθμός σημείων €-τμήματος		Ανοχή (m)
	Αρχικό	Γενικευμένο	
1	218	27	250
2	324	93	80
3	237	18	340
4	147	46	280
5	108	33	280
6	360	50	170
7	517	43	220
8	127	57	90
9	104	32	70
10	23	2	80
11	773	129	200
12	57	4	130
13	55	19	60
14	487	67	140
15	228	48	110
16	246	33	140
17	277	49	230
18	158	27	340
19	109	30	80
20	197	52	180
21	121	21	100

**Πίνακας 5.18:** Δεδομένα απλοποίησης €-κυρτών τμημάτων (σενάριο γενίκευσης 1:500.000).

#### 5.3.2.2.4 Συνένωση γενικευμένων €-τμημάτων - Εξομάλυνση γενικευμένης γραμμής - Δημιουργία γενικευμένης γραμμής

Μετά την ολοκλήρωση των διεργασιών γενίκευσης, τα 42 €-τμήματα συνενώνονται ώστε να ανασυνθέσουν τη ακτογραμμή, μέσω της οποίας θα αποδοθεί η νήσος Περιστέρα στο χάρτη κλίμακας 1:500.000. Η διαδικασία πραγματοποιείται στο περιβάλλον του λογισμικού πακέτου ArcGIS®, μέσω του μοντέλου που έχει δημιουργηθεί (βλ. κεφ. 4.5). Στη συνέχεια, η γενικευμένη ακτογραμμή υπόκειται σε μία μικρού βαθμού εξομάλυνση, η οποία στόχο έχει τη διόρθωση ατελειών, όπως ασυνέχειες σε θέσεις σύνδεσης των €-τμημάτων, αιχμές ή απότομα σπασίματα, ώστε να βελτιωθεί αισθητικά το τελικό προϊόν. Η διαδικασία υλοποιείται με τη χρήση του τελεστή *Gauss*, με τιμές της μεταβλητής  $\sigma=5$  στην πρώτη εκδοχή της γενικευμένης ακτογραμμής και  $\sigma=2$  σε εκείνη που προέκυψε μετά τις διορθώσεις των €-μη-κυρτών τμημάτων που δημιουργήθηκαν μετά τον έλεγχο ευαναγνωσιμότητας (βλ. κεφ. 5.3.2.2.2). Ο τελευταίος επαναλαμβάνεται (βλ. κεφ. 4.6), με το αποτέλεσμα του να πιστοποιεί την ευκρινή απόδοση της ακτογραμμής στην κλίμακα 1:500.000, βάσει των κανόνων οπτικής αντίληψης που έχουν ορισθεί. Με την ολοκλήρωση της διεργασίας προκύπτει η τελική, γενικευμένη μορφή της ακτογραμμής της νήσου Περιστέρας, όπως παρουσιάζεται στα Σχήματα 5.22 και

5.24α. Να σημειωθεί πως η ακτογραμμή παρουσιάζεται σε μεγέθυνση, με προσαρμοσμένο πάχος γραμμής, για την ευκολότερη παρατήρηση και αξιολόγηση του αποτελέσματος.



**Σχήμα 5.22:** Ακτογραμμή της νήσου Περιστεράς όπως προκύπτει από την εφαρμογή του μοντέλου για σενάριο γενίκευσης 1:500.000.

### 5.3.3 Αξιολόγηση αποτελέσματος

Η παράγωγη ακτογραμμή της νήσου Περιστεράς, όπως παρουσιάζεται στο Σχήμα 5.23α, αποτελεί το προϊόν σύνθεσης των επιμέρους, ανεξάρτητα γενικευμένων  $\epsilon$ -τμημάτων της. Κάθε ένα από αυτά έχει γενικευθεί μέσω κατάλληλων τελεστών, ανάλογα με την κατηγορία στην οποία ανήκει και τα ποσοτικά και γεωμετρικά χαρακτηριστικά του, με στόχο την ευανάγνωστη απεικόνισή του στο χάρτη κλίμακας 1:1.000.000. Το ερώτημα που τίθεται είναι το κατά πόσο τα 'ορθώς' γενικευμένα  $\epsilon$ -τμήματα συνθέτουν μία 'ορθώς' γενικευμένη ακτογραμμή, ως ενιαία οντότητα.

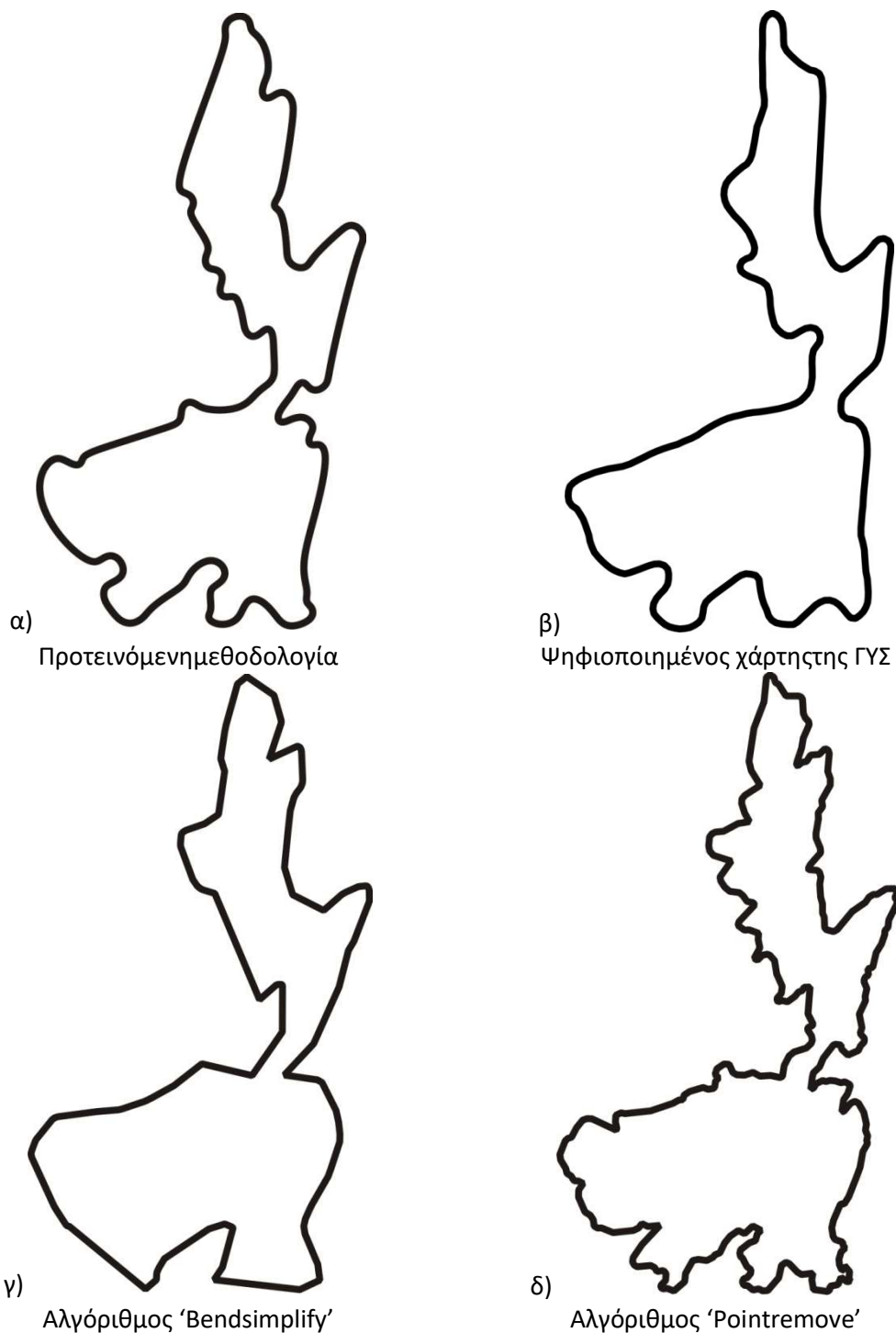
Παρατηρώντας την ακτογραμμή του Σχήματος 5.23α, θα μπορούσε να ειπωθεί ως ένα γενικό συμπέρασμα, πως το αποτέλεσμα της εφαρμογής είναι αρκετά ικανοποιητικό. Το μεγαλύτερο μέρος της λεπτομέρειας της αρχικής γραμμής (βλ. Σχήμα 5.3) έχει απαλειφθεί, γεγονός που συνάδει με τη μεγάλη μεταβολή κλίμακας που προέβλεπε το σενάριο γενίκευσης. Συγχρόνως, όμως, διατηρείται η βασική μορφή και ο χαρακτήρας της. Οι μικροί κυματισμοί και οι πολύ μικρού μεγέθους κόλποι και χερσόνησοι έχουν χαρακτηριστεί ως  $\epsilon$ -κυρτά τμήματα κατά τη διαδικασία κατάτμησης και έχουν αφαιρεθεί μέσω του κατάλληλου, για την εξάλειψη της λεπτομέρειας, αλγορίθμου απλοποίησης 'bendsimplify'. Οι αντίστοιχοι σχηματισμοί που αποτελούν μέρος των  $\epsilon$ -μη-κυρτών τμημάτων απαλείφονται κατά το στάδιο εξομάλυνσης που προηγείται των διεργασιών γενίκευσης. Οι μέσου και



μεγάλου μεγέθους κόλποι και χερσόνησοι, είτε έχουν διευρυνθεί όταν εμφανίζονται απομονωμένοι ώστε να είναι ευανάγνωστοι, είτε έχουν συγχωνευθεί με μεγαλύτερους γειτονικούς, όταν αποτελούν μέρος επαλληλίας. Έτσι, στην ακτογραμμή απεικονίζονται μόνο εκείνοι που είναι οπτικά αντιληπτοί στην παράγωγη κλίμακα και συγχρόνως, δε δημιουργούν προβλήματα σύμπτωσης με παρακείμενους. Θέσεις υψηλής πολυπλοκότητας, όπως για παράδειγμα το νότιο μέρος της γραμμής, αποδίδονται από τις βασικές, μεγαλύτερες καμπύλες. Παράλληλα, το τμήμα όπου εμφανίζεται στένωση (περιοχή σύγκλισης) έχει διευρυνθεί συμμετρικά ως προς το κέντρο του και σε κατάλληλο βαθμό, διατηρώντας τη συνέχειά του με τα γειτονικά τμήματα. Τέλος, η ακτογραμμή αποδίδεται σε ικανοποιητικό βαθμό ομαλή, δεδομένου και του μεγάλου βαθμού γενίκευσης. Απότομα σπασίματα ή ασυνέχειες, ακόμη και σε θέσεις μεγάλης μεταβολής της καμπυλότητάς της, δεν εμφανίζονται, γεγονός που οφείλεται στους περιορισμούς για διατήρηση της συνέχειας που τίθενται κατά τη γενίκευση των  $\epsilon$ -τμημάτων και την εξομάλυνση της γραμμής μετά την ανασύνθεσή της.

Για την εγκυρότερη αξιολόγηση του αποτελέσματος, η ακτογραμμή που έχει προκύψει από την εφαρμογή της προτεινόμενης τεχνικής συγκρίνεται με τρεις εκδοχές αυτής, οι οποίες έχουν παραχθεί από διαφορετικές διεργασίες. Η πρώτη (Σχήμα 5.23β) έχει δημιουργηθεί μετά από ψηφιοποίηση αναλογικού χάρτη της ΓΥΣ, κλίμακας 1:1.000.000. Η διαδικασία πραγματοποιήθηκε με μέσο βήμα ψηφιοποίησης 0,1mm στο χάρτη (100m στο έδαφος), ενώ η γραμμή έχει καθαρισθεί από σφάλματα ψηφιοποίησης και εξομαλυνθεί, ώστε να πάρει την τελική της μορφή. Οι άλλες δύο αποτελούν τα προϊόντα γενίκευσης της ακτογραμμής της νήσου Περιστεράς κλίμακας 1:50.000, μέσω των αλγορίθμων απλοποίησης 'bendsimplify' (Σχήμα 5.23γ) (βλ. κεφ. 3.6) και 'pointremove' (Σχήμα 5.23δ) (βλ. κεφ. 5.2.5). Οι τιμές των ανοχών εφαρμογής των δύο αλγορίθμων καθορίστηκαν με τρόπο, ώστε ο αριθμός των διατηρούμενων κορυφών να είναι ίσος. Το κριτήριο αυτό αποτελεί ένα αντικειμενικό μέτρο υλοποίησης και σύγκρισης αλγορίθμων, αφού βασίζεται σε ποσοτικές ιδιότητες της γραμμής. Ο αριθμός των διατηρούμενων κορυφών προσδιορίζεται σύμφωνα με τις 'Αρχές της Επιλογής' (Törfer and Pillewizer 1966), η οποία για την περίπτωση των ακτογραμμών (γραμμικά σύμβολα ίσου πάχους) εκφράζεται από τη σχέση (Jones and Abraham 1987):  $n = n_0 \frac{S}{S_0}$ , όπου

$S_0$  και  $n_0$  η κλίμακα και ο αριθμός κορυφών του αρχικού χάρτη και  $S$  και  $n$  του παράγωγου, αντίστοιχα. Η ακτογραμμή διαμορφώνεται από 326 σημεία στην περίπτωση του αλγορίθμου 'bendsimplify' και 330 στην περίπτωση του αλγορίθμου 'pointremove', αριθμός που προσεγγίζει και τον αριθμό σημείων της ψηφιοποιημένης γραμμής του χάρτη της ΓΥΣ κλίμακας 1:1.000.000 (316).



**Σχήμα 5.23:** Η ακτογραμμή της νήσου Περιστεράς παραγμένη από τέσσερις διαφορετικές μεθόδους για σενάριο γενίκευσης 1:1.000.000.

Η ακτογραμμή του Σχήματος 5.23β, ως προϊόν χειροκίνητης επεξεργασίας, αποτελεί ένα αξιόπιστο κριτήριο για την αξιολόγηση της προτεινόμενης τεχνικής. Συγκρίνοντάς την οπτικά με την παράγωγη ακτογραμμή (Σχήμα 5.23α), παρατηρείται πως η μορφή τους ομοιάζει σε μεγάλο βαθμό. Στις δύο γραμμές απεικονίζονται οι ίδιοι, μεγάλου μεγέθους χαρακτηριστικοί κόλποι και χερσόνησοι,

καθώς και οι ίδιες, σημαντικές ακραίες θέσεις (ακρωτήρια). Με άλλα λόγια, οι κύριες, αντιπροσωπευτικές γεωγραφικές οντότητες, οι οποίες είναι απαραίτητες για τη χαρτογραφικά και αισθητικά έγκυρη διαμόρφωση της γραμμής στη συγκεκριμένη κλίμακα, αποδίδονται και στις δύο περιπτώσεις. Το γεγονός αυτό, σε συνδυασμό με τη μη απόδοση σημαντικά περισσότερης λεπτομέρειας στην παράγωγη ακτογραμμή, αναδεικνύει την εν γένει αποτελεσματικότητα της προτεινόμενης τεχνικής. Παράλληλα, παρατηρείται πως η ακτογραμμή του Σχήματος 5.23α, προσεγγίζει σε ομαλότητα αυτή του ψηφιοποιημένου χάρτη της ΓΥΣ. Δεδομένου του μεγάλου βαθμού γενίκευσης, κάτι τέτοιο αποτελεί σημαντικό πλεονέκτημα της μεθόδου, αφού η επεξεργασία της γραμμής πραγματοποιείται μέσω εργαλείων που λειτουργούν βάσει αιτιοκρατικών κανόνων και δεν έχουν τη δυνατότητα 'παρέμβασης' και 'διόρθωσης', που έχει ο χαρτογράφος κατά τη χειροκίνητη σχεδίαση ενός χάρτη.

Οι απεικονιζόμενοι σχηματισμοί που περιγράφονται παραπάνω, αποτελούν, στην πλειοψηφία τους, μέρη  $\epsilon$ -μη-κυρτών τμημάτων που ανήκουν στις κατηγορίες Α και Β. Κατά τη διαδικασία, δηλαδή, κατάτμησης της γραμμής, εντοπίζονται ως θέσεις που παρουσιάζουν προβλήματα ευαναγνωσιμότητας και χρήζουν ιδιαίτερης επεξεργασίας. Η γενίκευσή τους πραγματοποιείται με αποτελεσματικό τρόπο, τόσο σε επίπεδο διατήρησης χαρακτηριστικών σημείων, όσο και σε επίπεδο επιλογής/επεξεργασίας και τελικά απόδοσης, των απαραίτητων καμπών από τις οποίες δομούνται. Οι επιμέρους, λοιπόν, διεργασίες κατάτμησης/χαρακτηρισμού, αλλά και γενίκευσης των συγκεκριμένων  $\epsilon$ -τμημάτων θα μπορούσαν να θεωρηθούν επιτυχείς. Επιπλέον, ένα πλεονέκτημα της προτεινόμενης μεθόδου είναι πως ορισμένα χαρακτηριστικά σημεία (ακρωτήρια, βαθύτερο σημείο κόλπων) των βασικών σχηματισμών που απεικονίζονται, αποδίδονται εγγύτερα στην αρχική τους θέση (σε αντίθεση με τη γραμμή του χάρτη της ΓΥΣ), γεγονός που βελτιώνει χαρτογραφικά το προϊόν.

Η επιπλέον λεπτομέρεια που παρουσιάζεται στην παράγωγη, από την προτεινόμενη τεχνική, ακτογραμμή έχει τη μορφή είτε μικρών κυματισμών, είτε μικρού μεγέθους κόλπων ή χερσονήσων. Οι πρώτοι, συνήθως εντοπίζονται κατά μήκος των  $\epsilon$ -κυρτών τμημάτων. Μία λύση για την εξάλειψή τους είναι η εφαρμογή μεγαλύτερου βαθμού απλοποίησης. Κάτι τέτοιο, όμως, θα είχε ως αποτέλεσμα την απώλεια της καμπυλότητας και της ομαλότητας των παράγωγων τμημάτων σε σχέση με την υπόλοιπη γραμμή, αφού κατά την απλοποίηση αφαιρούνται κορυφές, γεγονός που δε συμβαίνει στην γενίκευση των  $\epsilon$ -μη-κυρτών τμημάτων ή τη δημιουργία απότομων σπασιμάτων και ασυνεχειών στα σημεία σύνδεσης με τα γειτονικά, αφού θα μεταβάλλονταν η διεύθυνση των ακραίων θέσεών τους. Και τα δύο θα υποβάθμιζαν, αντί να βελτιώσουν, την ποιότητα του τελικού προϊόντος. Στη δεύτερη περίπτωση, οι μικροί κόλποι και χερσονήσοι, συνήθως, είναι μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα (σπανιότερα μπορεί να είναι και αμφίπλευρα  $\epsilon$ -μη-κυρτά τμήματα με δύο κύριες καμπές). Παρόλο που φαινομενικά προσδίδουν λεπτομέρεια στη γραμμή, η γενίκευσή τους έχει πραγματοποιηθεί με τρόπο κατάλληλο, ώστε να διασφαλίζεται η ευαναγνωσιμότητά τους στην παράγωγη κλίμακα. Παράλληλα, δεδομένου πως έχουν χαρακτηριστεί ως 'προβληματικά' κατά τη διαδικασία κατάτμησης, η γενίκευσή τους με την εφαρμογή ενός αλγορίθμου απλοποίησης, πιθανά, δε θα έδινε λύση στο ζητούμενο της ευκρινούς απόδοσής τους ή θα

δημιουργούσε αισθητικές και χαρτογραφικές ατέλειες (ασυνέχειες, σπασίματα κτλ.) στην παράγωγη γραμμή.

Οι ακτογραμμές που απεικονίζονται στα Σχήματα 5.23γ και 5.23δ αποτελούν προϊόντα γενίκευσης της αρχικής, μέσω των αλγορίθμων απλοποίησης 'bendsimplify' και 'pointremove', αντίστοιχα. Η σύγκριση της ακτογραμμής που παράγεται από την προτεινόμενη τεχνική με αυτές παρουσιάζει ενδιαφέρον, δεδομένου πως και οι τρεις αποτελούν αποτέλεσμα εφαρμογής τεχνητών μεθόδων γενίκευσης.

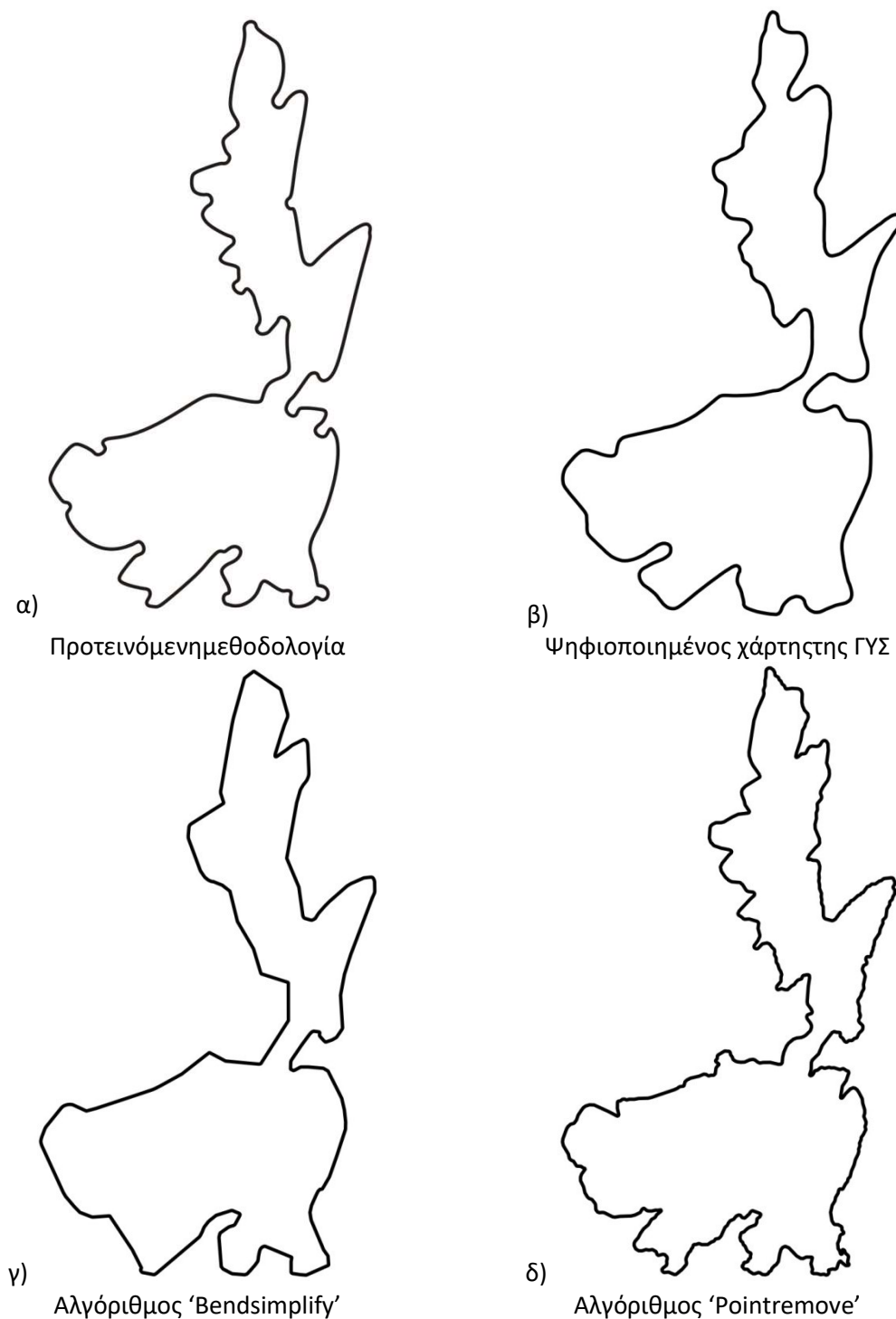
Η ακτογραμμή που προκύπτει από την εφαρμογή του αλγορίθμου 'pointremove' χαρακτηρίζεται από μεγάλο βαθμό πολυπλοκότητας. Αποδίδονται όλοι οι γεωγραφικοί σχηματισμοί (κόλποι, χερσόνησοι) και η πλειοψηφία των μικροκυματισμών της αρχικής γραμμής, ακόμη και σε ομαλά τμήματά της. Σε αρκετές θέσεις της, η σύμπτωση μεταξύ γραμμών είναι εμφανής. Η διατήρηση τόσο μεγάλου βαθμού λεπτομέρειας στη συγκεκριμένη κλίμακα, μετατρέπει το προϊόν σε χαρτογραφικά και αισθητικά μη αποδεκτό. Ωστόσο, πλεονέκτημα αποτελεί το γεγονός πως δεν εμφανίζονται απότομα σπασίματα και ασυνέχειες, ενώ, λόγω της δομής του αλγορίθμου, διατηρείται η πλειοψηφία των χαρακτηριστικών θέσεων της γραμμής. Αντίθετα, η παράγωγη ακτογραμμή από την εφαρμογή του αλγορίθμου 'bendsimplify' αποδίδεται με την ελάχιστη λεπτομέρεια, λιγότερη από αυτήν της ακτογραμμής του ψηφιοποιημένου χάρτη της ΓΥΣ (Σχήμα 5.23β). Βασικοί, όμως, κόλποι που χαρακτηρίζουν τη μορφολογία της ακτογραμμής, όπως αυτοί στο νότιο μέρος της, δεν απεικονίζονται, γεγονός που υποβαθμίζει χαρτογραφικά το προϊόν. Σημαντικό, επίσης, χαρακτηριστικό είναι η εμφάνιση απότομων σπασμάτων και ασυνεχειών κατά μήκος της γραμμής, που έχει ως αποτέλεσμα την ελάττωση της ομαλότητάς της. Το φαινόμενο αυτό, που οφείλεται στο μεγάλο βαθμό γενίκευσης, υποβαθμίζει αισθητικά το προϊόν, δεν επηρεάζει, όμως, σε μη αποδεκτό επίπεδο τη χαρτογραφική του εγκυρότητα.

Παρατηρώντας τις τρεις εκδοχές της ακτογραμμής, θα μπορούσε να ειπωθεί ως ένα γενικό σχόλιο, πως το σχήμα αυτής που έχει δημιουργηθεί από την προτεινόμενη τεχνική αποτελεί ένα συνδυασμό των υπόλοιπων δύο, έχοντας διατηρήσει τις θετικές τους ιδιότητες. Πιο συγκεκριμένα, η ακτογραμμή αποδίδεται με λεπτομέρεια που προσεγγίζει τα επίπεδα λεπτομέρειας εκείνης που προέρχεται από την εφαρμογή του αλγορίθμου 'bendsimplify'. Συγχρόνως, όμως, διατηρεί τους βασικούς γεωγραφικούς σχηματισμούς, ενώ παράλληλα, εμφανίζεται σημαντικά πιο εξομαλυσμένη. Σε σύγκριση με την ακτογραμμή που έχει δημιουργηθεί από τον αλγόριθμο 'pointremove', διατηρεί αναλόγως τη συνέχεια, αποδίδει μεγάλο μέρος των χαρακτηριστικών θέσεων, με σαφώς, όμως, λιγότερη λεπτομέρεια. Ως ένα βαθμό, ομοιάζει ως μια γενικευμένη εκδοχή αυτής.

Από τα παραπάνω γίνεται αντιληπτό πως οι δύο αλγόριθμοι απλοποίησης έχουν σημαντικά πλεονεκτήματα, αλλά και ατέλειες στη λειτουργία τους, οι οποίες υποβαθμίζουν την ποιότητα των παράγωγων προϊόντων. Εκ του αποτελέσματος κρινόμενη, η προτεινόμενη τεχνική καλύπτει ικανοποιητικά αυτές τις αδυναμίες, διατηρώντας, συγχρόνως, τα αντίστοιχα πλεονεκτήματα. Η ακτογραμμή που προκύπτει από την εφαρμογή της είναι χαρτογραφικά και αισθητικά πιο έγκυρη από εκείνες των δύο αλγορίθμων. Το γεγονός αυτό οφείλεται στο διαφορετικό τρόπο λειτουργίας τους. Ο αλγόριθμος 'pointremove' αποσκοπεί αποκλειστικά στην

απαλοιφή σημείων της γραμμής. Στηρίζεται στην καθολική ανάλυσή της βάσει ενός γεωμετρικού κριτηρίου, ώστε να επιλεγούν εκείνα τα σημεία που θα αφαιρεθούν. Αυτό έχει ως αποτέλεσμα να αντιμετωπίζονται όλες οι θέσεις της ομότροπα, γεγονός που δε συνάδει με 'τη διαφορετική συμπεριφορά των γραμμών κατά την αλλαγή κλίμακας, ανάλογα με τη μορφή και το χαρακτήρα τους' (βλ. κεφ. 1.1). Επιβεβαιώνεται, έτσι, η αναγκαιότητα δημιουργίας μεθόδων γενίκευσης που στηρίζονται στο εννοιολογικό πλαίσιο 'κατάτμηση-χαρακτηρισμός-γενίκευση'. Αντίθετα, η λειτουργία του αλγόριθμου 'bendsimplify' βασίζεται σε αυτή την αρχή και προσεγγίζει στη θεωρία του την προτεινόμενη τεχνική. Κατά την εφαρμογή του η γραμμή κατατέμνεται σε ομοιογενείς οντότητες (καμπές/bends), οι οποίες χαρακτηρίζονται μέσω ποσοτικών και ποιοτικών κριτηρίων και βάσει αυτών γενικεύονται μέσω κατάλληλων τελεστών. Η φιλοσοφία του όμως, αποσκοπεί κυρίως στην αφαίρεση των καμπών, δηλαδή στην αφαίρεση σημείων, γεγονός που έχει ως αποτέλεσμα τη δημιουργία των ασυνεχειών που παρουσιάζονται στη συγκεκριμένη εφαρμογή. Επιπλέον, εκτιμάται ως μειονέκτημα του αλγορίθμου η μη κατηγοριοποίηση των τμημάτων (καμπών) και κατά συνέπεια, η γενίκευσή τους σειριακά, ως προς το σύνολο της γραμμής. Στη λειτουργία αυτή, οφείλεται η απαλοιφή σημαντικών κόλπων από την παράγωγη ακτογραμμή. Αντίθετα, στην προτεινόμενη τεχνική, κάθε  $\epsilon$ -τμήμα αποκόπτεται από τη γραμμή, εντάσσεται σε μία από τις τέσσερις κατηγορίες και γενικεύεται ανεξάρτητα και με διαφορετικό τρόπο, ανάλογα με την κατηγορία που ανήκει. Η γενίκευση, συνήθως, δε συνίσταται στην αφαίρεση, αλλά στη μετατόπιση σημείων. Έτσι, εξασφαλίζεται η απόδοση σημαντικών γεωγραφικών σχηματισμών και η ορθή (στα πλαίσια των θεωρητικών κανόνων που έχουν τεθεί) γενίκευση όλων των τμημάτων.

Όμοια συμπεράσματα απορρέουν και από την παρατήρηση του Σχήματος 5.24, στο οποίο απεικονίζονται οι εκδοχές της ακτογραμμής της νήσου Περιστεράς, όπως έχουν παραχθεί από τις τέσσερις μεθόδους για την απόδοσή της σε χάρτη κλίμακας 1:500.000. Η πρώτη (Σχήμα 5.24α) αποτελεί το προϊόν από την εφαρμογή του προτεινόμενου μοντέλου (βλ. κεφ. 5.3.2). Η δεύτερη (Σχήμα 5.24β) έχει δημιουργηθεί από την ψηφιοποίηση του αναλογικού χάρτη της ΓΥΣ κλίμακας 1:500.000, με μέσο βήμα ψηφιοποίησης 0,1mm στο χάρτη (50m στο έδαφος) και διαμορφώνεται (μετά από τις διαδικασίες καθαρισμού και εξομάλυνσης) από 726 σημεία. Οι άλλες δύο αποτελούν τα προϊόντα γενίκευσης της ακτογραμμής της νήσου Περιστεράς κλίμακας 1:50.000, μέσω των αλγορίθμων απλοποίησης 'bendsimplify' (Σχήμα 5.24γ) και 'pointremove' (Σχήμα 5.24δ). Ο αριθμός των διατηρούμενων κορυφών προσδιορίσθηκε σύμφωνα με τις 'Αρχές της Επιλογής' (Töpfer and Pillewizer 1966) και είναι 613 για τον αλγόριθμο 'bendsimplify' και 667 για τον αλγόριθμο 'pointremove'.



**Σχήμα 5.24:** Η ακτογραμμή της νήσου Περιστεράς παραγμένη από τέσσερις διαφορετικές μεθόδους (σενάριο γενίκευσης 1:500.000).

Συνοψίζοντας, θα μπορούσε να ειπωθεί πως το αποτέλεσμα της εφαρμογής της προτεινόμενης τεχνικής στην ακτογραμμή της νήσου Περιστεράς για τα δύο σενάρια γενίκευσης είναι αρκετά ικανοποιητικό. Οι παράγωγες ακτογραμμές καλύπτουν τις απαιτήσεις ευαναγνωσιμότητας, οι οποίες αποτελούσαν ζητούμενο κατά το σχεδιασμό της μεθόδου, ενώ παράλληλα, αισθητικά θεωρούνται αποδεκτές. Η ομοιότητα, σε μεγάλο βαθμό, του σχήματός τους με το σχήμα των ακτογραμμών

των χαρτών της ΓΥΣ επιβεβαιώνει το γεγονός. Συγχρόνως, τα αποτελέσματα της σύγκρισής τους με τις ακτογραμμές που προκύπτουν από δύο ευρέως διαδεδομένους αλγορίθμους γενίκευσης είναι άκρως ενθαρρυντικά. Σημαντικά μειονεκτήματα που οφείλονται στη λειτουργία των αλγορίθμων δεν παρουσιάζονται στην προτεινόμενη μέθοδο, σε αντίθεση με ορισμένα θετικά τους στοιχεία. Το γεγονός αποδεικνύει πως η τεχνική είναι ορθά δομημένη σε θεωρητικό επίπεδο και ικανοποιητικά εφαρμόσιμη.

## ΚΕΦΑΛΑΙΟ 6: ΣΥΜΠΕΡΑΣΜΑΤΑ

Τα τελευταία χρόνια, ερευνητικό ζητούμενο για τους χαρτογράφους είναι η δημιουργία ‘καθολικών συστημάτων γενίκευσης’, τα οποία εκτιμάται πως αποτελούν τη βέλτιστη λύση στο πεδίο της γενίκευσης ψηφιακών γραμμών. Επιδίωξη είναι η προσομοίωση μέσω αυτών, της αντίστοιχης χειροκίνητης διαδικασίας. Οι έρευνες εστιάζουν στην ‘αντιγραφή’ του τρόπου, με τον οποίο ο άνθρωπος αντιλαμβάνεται και, κατά συνέπεια, αναλύει μια γραμμική οντότητα και στην αυτοματοποίησή του μέσω ηλεκτρονικών υπολογιστών.

Πάνω σε αυτό το πλαίσιο, οι χαρτογράφοι έχουν καταλήξει, μεταξύ άλλων, σε δύο συμπεράσματα. Το πρώτο είναι πως κατά τη χειροκίνητη γενίκευση, ο άνθρωπος τείνει να εστιάζει περισσότερο (και τελικά να επιλέγει) σε θέσεις όπου παρατηρείται μεγάλη μεταβολή στην κλίση της γραμμής. Οι θέσεις αυτές (κρίσιμα σημεία όπως ονομάζονται) είναι ιδιαίτερου ενδιαφέροντος και παίζουν σημαντικό ρόλο στην ανάλυση και επεξεργασία των ψηφιακών γραμμών. Επιπρόσθετα, κατά την περάτωση της χειροκίνητης γενίκευσης, ο άνθρωπος αναλύει τη γραμμή, εντοπίζει τμήματά της με κοινά χαρακτηριστικά, τα οποία και μετασχηματίζει κατάλληλα. Έτσι, και η φιλοσοφία των καθολικών συστημάτων βασίζεται στο τρίπτυχο ‘κατάτμηση-χαρακτηρισμός-γενίκευση’. Με άλλα λόγια, η λειτουργία τους στηρίζεται στην κατάτμηση της γραμμής σε ομοιογενή τμήματα, το χαρακτηρισμό τους μέσω ποσοτικών ή και ποιοτικών κριτηρίων και βάσει αυτών, τη γενίκευση τους μέσω κατάλληλων τελεστών. Με την παρούσα διατριβή επιχειρείται μια θεωρητική και πρακτική προσέγγιση των δύο παραπάνω ζητούμενων.

Η έννοια των κρίσιμων σημείων, όπως παρουσιάστηκε από τον ψυχολόγο Attneave (1954), με σκοπό τον χαρακτηρισμό, σε επίπεδο οπτικής αντίληψης, των γραφικών αναπαραστάσεων των αντικειμένων και, στη συνέχεια, υιοθετήθηκε από τους χαρτογράφους, αποτελεί σημαντικής σημασίας αντικείμενο μελέτης στον τομέα της γενίκευσης. Η έννοια αυτή είναι σημαντική για τους χαρτογράφους, καθώς χρησιμοποιείται, έμμεσα, ως μία μορφή ‘οδηγίας’ για την αξιολόγηση των αλγορίθμων απλοποίησης, αλλά και άμεσα, αφού οι περισσότερες διαδικασίες απλοποίησης εξαρτώνται, ως ένα βαθμό, από τη διατήρηση ή την απαλοιφή των κρίσιμων σημείων.

Ο προτεινόμενος δείκτης LR αποτελεί μία νέα μέθοδο εντοπισμού των κρίσιμων σημείων γραμμικών οντοτήτων. Είναι προσαρμοσμένη για την κάλυψη των αναγκών της ψηφιακής αναπαράστασης των γραμμών διανυσματικής (vector) δομής, σε αντίθεση με την μέθοδο που αναπτύχθηκε από τον Thapa (1988a), ο οποίος επικεντρώνει το ενδιαφέρον του σε ψηφιακές γραμμές κανονικοποιημένης (raster) δομής. Ο εντοπισμός κρίσιμων σημείων ολοκληρώνεται, είτε μέσω εμπειρικών διαδικασιών, είτε μέσω των αλγορίθμων απλοποίησης. Στην μεν πρώτη περίπτωση, όμως, είναι αδύνατη η αυτοματοποίηση της διαδικασίας και η απεξάρτησή της από τον υποκειμενικό ανθρώπινο παράγοντα, ενώ η δεύτερη ελλοχεύει αρκετούς κινδύνους, αφού η δομή και η λειτουργία των αλγορίθμων είναι σχεδιασμένη για διαφορετικούς σκοπούς. Αν και το θεωρητικό υπόβαθρο μιας διαδικασίας εντοπισμού κρίσιμων σημείων δε διαφέρει σημαντικά από αυτό, μιας αντίστοιχης,



επιλογής σημείων που διαμορφώνουν μια γενικευμένη γραμμή, η αυτοματοποίησή τους διαφέρει σημαντικά.

Η λειτουργία του προτεινόμενου δείκτη LR, στηρίζεται στην εξέταση της γραμμής σε τοπικό επίπεδο, στα όρια της εκάστοτε κορυφής. Η μέθοδος προσεγγίζει την ιδέα της 'περιοχής εξέτασης', όπως αυτή παρουσιάζεται σε ανάλογους αλγόριθμους εντοπισμού κρίσιμων σημείων που προέρχονται από το πεδίο της επεξεργασίας εικόνας. Η 'περιοχή εξέτασης' προσδιορίζεται με την εφαρμογή κύκλου προκαθορισμένης ακτίνας, με κέντρο κάθε κορυφή της ψηφιακής γραμμής. Έτσι, ο υπολογισμός της καμπυλότητας δεν πραγματοποιείται άμεσα, με τη μέτρηση της γωνιακότητας, αλλά, έμμεσα, μέσω της γεωμετρικής ανάλυσης της γραμμής, με κριτήριο το μήκος.

Το μήκος αποτελεί ένα στοιχείο της γραμμής που επηρεάζεται, τόσο από τη μορφολογία της, όσο και από τη μονάδα μέτρησής του (Buttenfield 1985). Στην προτεινόμενη μεθοδολογία, εκτιμάται πως θέτοντας σταθερό το δεύτερο, μπορεί να αξιολογηθεί το πρώτο, να υπολογισθεί η καμπυλότητα και τελικά, βάση των αρχών του Attneave, να εντοπισθούν τα κρίσιμα σημεία. Το γεγονός ότι ο δείκτης LR είναι αδιάστατο μέγεθος, το μετατρέπει σε ένα σταθερό μέτρο, το οποίο, συγχρόνως, είναι, σε υψηλό βαθμό ευαίσθητο στις μεταβολές της καμπυλότητας, αφού το ανάπτυγμα L αυξάνει και η χορδή S ελαττώνεται ανάλογα με αυτές.

Η εξέταση κάθε κορυφής ανεξάρτητα, έχει ως αποτέλεσμα την αξιολόγηση της τοπικής σημαντικότητας της εκάστοτε θέσης και, στη συνέχεια, την ένταξή της στο σύνολο, μέσω της επιλογής τιμών του δείκτη μεγαλύτερων ενός προκαθορισμένου, για όλη τη γραμμή, ορίου. Ως ένα βαθμό, η διαδικασία μιμείται την αντίστοιχη χειροκίνητη, κατά την οποία η σημαντικότητα κάθε θέσης εξετάζεται ταυτόχρονα σε τοπικό και ολικό επίπεδο. Το γεγονός αυτό, είναι ιδιαίτερα σημαντικό στην ανάλυση χαρτογραφικών απεικονίσεων, όπου ο βαθμός λεπτομέρειας που πρέπει να αποδοθεί, διαφέρει ανάλογα με την τοπική μορφολογία της γραμμής. Την άποψη αυτή στηρίζουν σε μελέτες τους αρκετοί χαρτογράφοι, όπως, ο Dutton (1999), οι Visvalingam και Whyatt (1990), η Buttenfield (1989) ή ο Thapa (1988).

Σε πρακτικό επίπεδο, κατά τη διαδικασία εφαρμογής της προτεινόμενης μεθόδου, απαιτείται ο ορισμός από τον χρήστη δύο παραμέτρων: της ακτίνας (R) του κύκλου εφαρμογής και του ορίου επιλογής (κατώτατο όριο επιλογής του δείκτη LR). Οι δεσμεύσεις αυτές, όμως, γίνεται προσπάθεια να συσχετισθούν, τόσο με τα δεδομένα της γραμμής, όσο και με τη μορφολογία της. Η ακτίνα τίθεται, κάθε φορά, ανάλογη του μέσου βήματος ψηφιοποίησης της γραμμής, ενώ η αντιστοίχιση τιμών του δείκτη LR με το απεικονιζόμενο σχήμα, βοηθά στον προκαθορισμό του ορίου επιλογής, το οποίο τίθεται ίσο με 1,04, μια τιμή που αποκλείει από τον εντοπισμό όλες τις κυρτώσεις που χαρακτηρίζονται από λόγο βάσης ως προς ύψος μεγαλύτερο από 11:1. Έτσι, σε ικανοποιητικό βαθμό, επιτυγχάνεται η 'σταθεροποίηση' των παραμέτρων, ώστε η προτεινόμενη μεθοδολογία να αποκτήσει πιο αντικειμενικό χαρακτήρα, απεξαρτημένο από την υποκειμενική κρίση του ανθρώπινου παράγοντα.

Τα αποτελέσματα που προκύπτουν από την εφαρμογή του δείκτη μπορούν να θεωρηθούν αρκετά ικανοποιητικά, σε ποσοτικό (σύγκριση με μελέτες Marino και Thapa), αλλά και σε ποιοτικό (οπτικά αποδεκτή απεικόνιση της ακτογραμμής της

νήσου Περιστέρα) επίπεδο. Ικανοποιητικά μπορούν, επίσης, να θεωρηθούν τα αποτελέσματα της εμπειρικής έρευνας που πραγματοποιήθηκε και σκοπό είχε την αξιολόγηση του δείκτη σε σύγκριση με τη μέθοδο των οφθαλμικών καταγραφών, αφού παρατηρείται υψηλή συγκέντρωση προσηλώσεων των υποκειμένων σε θέσεις γύρω από τα κρίσιμα σημεία που έχουν εντοπισθεί μέσω του δείκτη ALR. Ωστόσο, η μη επιθυμητή μεταβολή του δείκτη LR ανάλογα με την πολυπλοκότητα της γραμμής, όπως αναφέρεται σε προηγούμενες ενότητες, είναι ένα θέμα που χρήζει διεξοδικότερης μελέτης. Επίσης, ο δείκτης LR δεν εντοπίζει τις επιμηκυμένες, διευρυμένες καμπύλες με σταθερή αλλαγή της κλίσης. Η τελευταία αυτή επισήμανση δεν είναι μια ανεπάρκεια της μεθόδου. Αυτά τα 'σιωπηλά' σημεία που απαρτίζουν τις παραπάνω καμπύλες μορφές, μπορεί να είναι σημαντικά στη διαδικασία απλοποίησης των γραμμών, εντούτοις, δεν θεωρούνται ως κρίσιμα σημεία και επομένως, δεν εντοπίζονται από την προτεινόμενη μέθοδο. Όσον αφορά τη συγκεκριμένη μέθοδο, δεν σχεδιάστηκε προκειμένου να εκτελεστεί για να αντιμετωπίσει τη διαδικασία της απλοποίησης γραμμών, αλλά για να εντοπίζει κρίσιμα σημεία, η λειτουργία της μπορεί για αυτόν το λόγο να θεωρηθεί ως πολύ ικανοποιητική.

Με άξονα το εννοιολογικό πλαίσιο της δημιουργίας καθολικών συστημάτων γενίκευσης που βασίζονται στο τρίπτυχο 'κατάτμηση-χαρακτηρισμός-γενίκευση', σχεδιάζεται και υλοποιείται μία τεχνική γενίκευσης ψηφιακών χαρτογραφικών γραμμών, αποσκοπώντας να συμβάλλει στη νέα ερευνητική τάση πάνω στο αντικείμενο. Η τεχνική ενσωματώνει μια σειρά επιμέρους διεργασιών επεξεργασίας γραμμικών οντοτήτων, οι οποίες, γενικά, μπορούν να διαχωρισθούν σε τρία στάδια.

Το πρώτο αφορά την κατάτμηση των γραμμών σε ομοιογενή τμήματα. Η μεθοδολογία που εισάγεται, στηρίζεται στην ιδέα του κυλιόμενου δίσκου του Perkal (1966a), η οποία προσαρμόζεται στην ψηφιακή απεικόνιση δεδομένων. Για την ολοκλήρωσή της χρησιμοποιείται ως κριτήριο η ευαναγνωσιμότητα των γραμμών, η οποία ορίζεται μέσω χαρτογραφικών κανόνων που στηρίζονται στην οπτική αντίληψη. Με τον τρόπο αυτό, η διαδικασία λαμβάνει αντικειμενικό χαρακτήρα και διασφαλίζεται η δυνατότητα εφαρμογής της σε κάθε γραμμή και για οποιοδήποτε σενάριο γενίκευσης.

Στο δεύτερο στάδιο επεξεργασίας λαμβάνει χώρα η κατηγοριοποίηση και ο χαρακτηρισμός των τμημάτων που αποκόπτονται κατά μήκος της γραμμής. Ανάλογα με τον τρόπο δημιουργίας και τη μορφή τους, τα τμήματα διακρίνονται σε τέσσερις κατηγορίες. Στην πρώτη εντάσσονται τμήματα που διαμορφώνονται από μία κύρτωση ή κοίλωση, στη δεύτερη εκείνα που έχουν τη μορφή επαλληλίας δύο ή περισσοτέρων κυρτώσεων και κοιλώσεων, στην τρίτη τμήματα που πλησιάζουν μεταξύ τους απόσταση που δημιουργεί πρόβλημα ευαναγνωσιμότητας και στην τέταρτη ομαλά τμήματα. Είναι εμφανές, πως οι τέσσερις κατηγορίες καλύπτουν χαρακτηριστικούς τύπους σχηματισμών στους οποίους μπορεί να αποδομηθεί μια γραμμική οντότητα. Παράλληλα, ο εντοπισμός (όπως και οι μετέπειτα διεργασίες γενίκευσης) τμημάτων που ανήκουν στην τρίτη κατηγορία μπορεί να θεωρηθεί καινοτόμος, αφού μολονότι αποτελεί ερευνητικό ζητούμενο στη διεθνή βιβλιογραφία (γνωστό ως 'bottleneck'), δε συναντάται αντίστοιχη μεθοδολογία διαχείρισης.

Στο τρίτο στάδιο επεξεργασίας ολοκληρώνεται η γενίκευση των γραμμών. Κάθε τμήμα γενικεύεται ως ανεξάρτητη οντότητα, μέσω ενός ή συνδυασμού τελεστών, ανάλογα με την κατηγορία στην οποία ανήκει, τη γεωμετρία και τις ποσοτικές ιδιότητές του. Έτσι, διασφαλίζεται, ως ένα βαθμό, η αντικειμενικότητα της διαδικασίας, περιορίζοντας σημαντικά την παρέμβαση του χρήστη. Κάτι τέτοιο επιτυγχάνεται στο ακέραιο στην περίπτωση των τμημάτων σύγκλισης, δεδομένου ότι οι μετασχηματισμοί γενίκευσης εφαρμόζονται βάσει πληροφοριών/δεσμεύσεων που προκύπτουν από την ανάλυση των ιδιοτήτων της 'περιοχής σύγκλισης'. Ομοίως, και ο αλγόριθμος 'Δεσμευμένης Διόγκωσης' υλοποιείται βάσει παραμέτρων που απορρέουν από την ανάλυση της γεωμετρίας στη 'θέση Κ' (βλ. κεφ. 4.1.2.2.1). Για κάθε κατηγορία ορίζονται κατάλληλοι χαρτογραφικοί μετασχηματισμοί των τμημάτων (μεγέθυνση για την πρώτη, αφαίρεση/συγχώνευση και μεγέθυνση για τη δεύτερη, διεύρυνση για την τρίτη και απλοποίηση για την τέταρτη), οι οποίοι περατώνονται μέσω τελεστών γενίκευσης (διόγκωση, συμπίεση, εξομάλυνση, απλοποίηση καμπών και διεύρυνση) και χαρτογραφικών 'εργαλείων' (εντοπισμός καμπών, ομοπαράλληλος μετασχηματισμός) που δημιουργούνται. Η αυτοτελής επεξεργασία των τμημάτων συντελεί στον δυνατόν εγκυρότερο μετασχηματισμό τους, ώστε να αποδοθούν στην παράγωγη κλίμακα, αφού αυτός πραγματοποιείται σε τοπικό επίπεδο, με βάση τα ιδιαίτερα χαρακτηριστικά τους, χωρίς να επηρεάζεται από την υπόλοιπη γραμμή. Παράλληλα, είναι σημαντικό πως η επεξεργασία πραγματοποιείται έχοντας ως στόχο την επίτευξη του κριτηρίου ευαναγνωσιμότητας, το οποίο έχει εκ των προτέρων ορισθεί και το οποίο βασίζεται σε κανόνες της οπτικής αντίληψης. Το γεγονός αποτελεί, αφενός ένα μέτρο εκτίμησης για την εγκυρότητα ή μη του παράγωγου προϊόντος και αφετέρου, μία 'κοινή συνθήκη' για την επεξεργασία του συνόλου της γραμμής, απεξαρτώντας την από την υποκειμενικότητα του χρήστη. Σε πρακτικό επίπεδο, η μη μετάθεση των άκρων των τμημάτων κατά τους μετασχηματισμούς που υφίστανται συντελεί στην ορθή ένταξή τους στο σύνολο της γραμμής, γεγονός αναγκαίο για τη δημιουργία ενός αποδεκτού τελικού προϊόντος. Όσον αφορά την οικονομία χρόνου, είναι θετικό πως υπολογίζονται, κάθε φορά, τα γεωμετρικά και ποσοτικά χαρακτηριστικά που είναι απαραίτητα για τη γενίκευση του εκάστοτε τμήματος ανάλογα με την κατηγορία στην οποία ανήκει και όχι για το σύνολο της γραμμής.

Πλεονέκτημα του μοντέλου μπορεί να θεωρηθεί πως οι αλγόριθμοι εφαρμογής των τελεστών γενίκευσης που εισάγονται, σχεδιάζονται με γνώμονα την υλοποίησή τους σε γραμμές που αναπαριστούν φυσικές οντότητες (ακτογραμμές, ποτάμια κτλ.). Το γεγονός αυτό, τους μετατρέπει σε πιο αξιόπιστους, αφού η επιτυχής λειτουργία στις πιο πολύπλοκες και ακανόνιστου σχήματος γραμμές, δημιουργεί εχέγγυα για την αποτελεσματικότητά τους σε οποιαδήποτε γραμμή. Παράλληλα, αποτελεί καινοτομία, δεδομένου πως το ζήτημα της γενίκευσης τέτοιου είδους γραμμών δεν έχει αντιμετωπισθεί ευρέως στη διεθνή έρευνα.

Δεδομένου πως η μεθοδολογία που παρουσιάζεται είναι αρκετά πολύπλοκη, τόσο στο θεωρητικό σχεδιασμό, όσο και στην υλοποίησή της, τα αποτελέσματα μίας εφαρμογής αποτελούσαν ζητούμενο της έρευνας. Τα συμπεράσματα από την πιλοτική εφαρμογή στην ακτογραμμή της νήσου Περιστερά για τα δύο σενάρια γενίκευσης (απεικόνιση σε χάρτες κλίμακας 1:1.000.000 και 1:500.000) θα μπορούσαν να χαρακτηρισθούν ως αρκετά ενθαρρυντικά. Το παράγωγο προϊόν

κρίνεται χαρτογραφικά και αισθητικά αποδεκτό, σύμφωνα με τα δεδομένα και τις απαιτήσεις που ορίζονται στο θεωρητικό υπόβαθρο του έργου. Και στις δύο περιπτώσεις, η βασική μορφή και ο χαρακτήρας της γραμμής διατηρείται, παρά τον μεγάλο βαθμό γενίκευσης. Η ομοιότητά τους με τις ακτογραμμές που προέρχονται από τη ψηφιοποίηση των αναλογικών χαρτών της ΓΥΣ αναδεικνύει το γεγονός, αφού η επεξεργασία των γραμμών πραγματοποιείται μέσω εργαλείων που λειτουργούν βάσει αιτιοκρατικών κανόνων και δεν έχουν τη δυνατότητα 'παρέμβασης' και 'διόρθωσης', που έχει ο χαρτογράφος κατά τη χειροκίνητη σχεδίαση ενός χάρτη. Παράλληλα, η μη εμφάνιση ατελειών που παρατηρούνται σε ακτογραμμές που έχουν δημιουργηθεί μέσω διαφορετικών τεχνικών γενίκευσης (τελεστών απλοποίησης), δημιουργεί θετικά συμπεράσματα όσον αφορά τη λειτουργία της μεθόδου. Ένα γενικό συμπέρασμα από την εφαρμογή στην ακτογραμμή της νήσου Περιστέρα, είναι πως έχουν δημιουργηθεί οι σωστές βάσεις στο σχεδιασμό της τεχνικής. Μέσω της μεθοδολογίας κατάτμησης εντοπίζονται, απομονώνονται και χαρακτηρίζονται αντιπροσωπευτικοί σχηματισμοί της γραμμής (μεγάλου ή μεσαίου μεγέθους κόλποι και χερσόνησοι). Μάλιστα οι οντότητες αυτές αποτελούν μέρη των δύο πρώτων κατηγοριών των τμημάτων (μονές ή επαλληλίες κυρτώσεων και κοιλώσεων). Δεδομένου πως έχουν χαρακτηριστεί ως 'προβληματικές' κατά τη διαδικασία κατάτμησης, η γενίκευσή τους δεν ολοκληρώνεται μέσω ενός απλού τελεστή απλοποίησης, αλλά μιας σειράς μετασχηματισμών που έχουν ορισθεί. Αυτό έχει ως αποτέλεσμα να διατηρείται το βασικό σχήμα τους στην παράγωγη κλίμακα. Επιπλέον, ορισμένες χαρακτηριστικές θέσεις αυτών (ακρωτήρια, βαθύτερο σημείων κόλπων) αποδίδονται αρκετά κοντά στις αρχικές, αφού στους μετασχηματισμούς 'εμπλέκονται' τα κρίσιμα σημεία των αρχικών γραμμών, γεγονός που βελτιώνει χαρτογραφικά το προϊόν. Σημαντικό είναι, επίσης, το γεγονός ότι εντοπίζεται και γενικεύεται αποτελεσματικά η θέση όπου εμφανίζεται στένωση μεταξύ των γραμμών (περιοχή σύγκλισης). Τέλος, οι ακτογραμμές θα μπορούσαν να χαρακτηρισθούν αισθητικά έγκυρες, αφού δεν εμφανίζονται απότομα σπασίματα ή ασυνέχειες, οι μικροί κυματισμοί έχουν απαλειφθεί σε όλο το μήκος τους και χαρακτηρίζονται από ομαλότητα.

Ως μειονέκτημα της μεθόδου θα μπορούσε να χαρακτηριστεί το γεγονός των δοκιμαστικών εφαρμογών του τελεστή εξομάλυνσης, ώστε να επιτευχθούν οι κατάλληλες συνθήκες για την γενίκευση των μονόπλευρων και των αμφίπλευρων τμημάτων. Αν και η τελική επιλογή του βαθμού εξομάλυνσης πραγματοποιείται λαμβάνοντας υπόψη τα ποσοτικά χαρακτηριστικά των τμημάτων (βάθος, εμβαδό, μήκος γραμμής βάσης κτλ) που υπολογίζονται από το σύστημα, η όλη διαδικασία ενέχει στοιχεία υποκειμενικότητας, ενώ παράλληλα αυξάνεται χρόνος περάτωσης της. Αποτελεί, λοιπόν, ζητούμενο μελλοντικής έρευνας η ενσωμάτωση των παραπάνω χαρακτηριστικών στη ρουτίνα υλοποίησης, ώστε να αυτοματοποιηθεί πλήρως η όλη επεξεργασία και να περιορισθεί η παρέμβαση του χρήστη.

Η μέθοδος που παρουσιάζεται θα μπορούσε να χαρακτηριστεί ως ένα 'καθολικό σύστημα γενίκευσης', υπό την έννοια ότι η επεξεργασία της γραμμής ολοκληρώνεται μέσω μιας σειράς διεργασιών που στηρίζονται στο τρίπτυχο 'κατάτμηση-χαρακτηρισμός-γενίκευση' και συγχρόνως, πραγματοποιείται σε περιβάλλον ενός λογισμικού πακέτου (ArcGIS® της εταιρίας ESRI). Το τελευταίο αποτελεί σημαντικό πλεονέκτημα του μοντέλου, αφού διευκολύνει τη διαχείριση

από τον χρήστη των ρουτινών επεξεργασίας και γενίκευσης των γραμμών μέσω των 'παράθυρων' που έχουν δημιουργηθεί, ενώ παράλληλα, επιτυγχάνεται η άμεση οπτικοποίηση των αποτελεσμάτων. Επιπλέον, παρέχεται η δυνατότητα το μοντέλο κατάτμησης/χαρακτηρισμού και οι τελεστές γενίκευσης που έχουν δημιουργηθεί να χρησιμοποιηθούν ανεξάρτητα, γεγονός που θα μπορούσε να συμβάλει στη δημιουργία ανεξάρτητων διαδραστικών 'εργαλείων' γενίκευσης ψηφιακών χαρτογραφικών γραμμών.

## **Βιβλιογραφία**

- Ansari, N. and Huang K., 1991. "Non-Parametric Dominant Points Detection." *Pattern Recognition*, 24(9): 849-862.
- Antoine, J.-P., Barache, D., Cesar Jr., R.M. and da Fontoura Costa, L., 1997. "Shape Characterization with the Wavelet Transform." *Signal Processing*, 62: 265-290.
- Attneave F., 1954. "Some Informational Aspects of Visual Perception." *Psychological Review*, 61(3): 183-193.
- Awh E, Belopolsky A V, Theeuwes J., 2012. "Top-down versus bottom-up attentional control: a failed theoretical dichotomy." *Trends in Cognitive Sciences* 16(8): 437-443
- Beard M.K., 1991. "Theory of the Cartographic Line Revisited/Implications for Automated Generalization", *Cartographica*, 25(4): 32-58.
- Brassel K.E. and Weibel R., 1988. "A Review and Conceptual Framework on Automated Map Generalization", *International Journal of Geographical Information Systems*, 2(3): 229-244.
- Buttenfield B., 1985. "Treatment of the Cartographic Line." *Cartographica*, 22(2): 1-26.
- Buttenfield B., 1989. "Scale-Dependence and Self-Similarity in Cartographic Lines", *Cartographica*, 26(1): 79-99.
- Castner H W and Eastman J.R., 1984. "Eye-movement parameters and perceived map complexity-I." *The American Cartographer*, 11(2): 107-117
- Cornic Ph., 1997. "Another Look at the Dominant Point Detection of Digital Curves." *Pattern Recognition Letters*, 18: 13-25.
- Coltekin A., Heil B., Garlandini S., Fabrikant S., 2009. "Evaluating the Effectiveness of Interactive Map Interface Designs: A Case Study Integrating Usability Metrics with Eye-movement Analysis." *Cartography and Geographic Information Science*, 36(1): 5-17
- Cronin, T.M. 1999. "A Boundary Code to Support Dominant Point Detection." *Pattern Recognition Letters*, 20: 617-634.
- Douglas D.H. and Peucker Th. P., 1973. "Algorithms for the Reduction of the Number of Points Required to represent a Digitized Line or its Caricature", *The Canadian Cartographer*, 10(2): 112-122.
- Dutton G., 1999. "Scale, Sinuosity and Point Selection in Digital Line Generalization" *Cartography and Geographic Information Science*, 26(1): 33-53.
- Freeman H., 1978. "Shape Description Via the Use of Critical Points." *Pattern Recognition*, 10: 159-166.
- Freeman H. and Davis L., 1977. "A Corner-finding Algorithm for Chain-coded Curves." *Transaction on Computers*, 26: 297-303.

- Fritsch E., 1997. Représentations de la géométrie et des contraintes cartographiques pour la généralisation du linéaire routier. PhD dissertation, Université de Marne La Vallée.
- Greene M.R. and Wolfe J.M., 2011. "Global image properties do not guide visual search." *Journal of Vision*, 11(6): 1-9
- Hoffman, D., and Richards, W.A. 1982. "Representing Smooth Plane Curves for Recognition: Implications for Figure-Ground Reversal." *In Proceedings of the National Conference on Artificial Intelligence: 5-8.*
- Jenks G.F., 1981. "Lines, Computers, and Human Frailties." *Annals of the Association of American Geographers*, 71(1): 1-10.
- Jones C.B. and Abraham I.M., 1987. "Line Generalization in a Global Cartographic Database." *Cartographica*, 24(3): 32-45.
- Krassanakis V, Filippakopoulou V., Nakos B., 2011. "An Application of Eye Tracking Methodology in Cartographic Research." *Proceedings of the EyeTrackBehavior (Tobii)*, Frankfurt
- Lecordix F., Plazanet C. And Lagrange J. Ph., 1997. "A Platform for Research in Generalization: Application to Caricature". *Geoinformatica*, 1(2): 161-182.
- Li Z., 1995. "An Examination of Algorithms for the Detection of Critical Points on Digital Cartographic Lines." *The Cartographic Journal*, 32(2): 121-125.
- Li Z., 2007. "Algorithmic Foundation of Multi-Scale Spatial Representation." *Taylor & Francis Group*, Boca Raton: 280.
- Lloyd R., 1997. "Visual Search Processes Used in Map Reading." *Cartographica*, 34(1): 11-32.
- Lowe D., 1988. "Organization of Smooth Image Curves at Multiple Scales." *Proceedings of the International Conference on Computer Vision*, Florida, USA, 558-567.
- Mandelbrot, B.B. 1982. *The Fractal Geometry of Nature*, New York: Freeman & Co.
- Marino J.S., 1979. "Identification of Characteristic Points Along Naturally Occurring Lines / An Empirical Study." *The Canadian Cartographer*, 16(1): 70-80.
- McMaster R.B., 1986. "A Statistical Analysis of Mathematical Measures for Linear Simplification." *The American Cartographer*, 13(2): 103-116.
- McMaster R.B., 1987. "Automated Line Generalization", *Cartographica*, 24(2): 74-111.
- McMaster R.B. and K.S. Shea, 1992. "Generalization in Digital Cartography", (Resource Publications in Geography). Washington, D.C., Association of American Geographers.
- Michaelidou E., Filippakopoulou V., Nakos B., Petropoulou A., 2005. "Designing Point Map Symbols: The effect of preattentive attribute of shape". *Proceedings of the 22<sup>th</sup> International Cartographic Association Conference*, A Coruna.

- Mustière S., 2005. "Cartographic Generalization of Roads in a Local Adaptive Approach: A Knowledge Acquisition Problem", *International Journal of Geographic Information Science*, 19(8-9): 937-955.
- Νάκος Β., 2006. Αναλυτική Χαρτογραφία. Σχολή Αγρονόμων & Τοπογράφων Μηχανικών, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα.
- Nakos B., and V. Mitropoulos, 2005. "Critical Points Detection Using the Length Ratio (LR) for Line Generalization". *Cartographica*, 40(3): 35-51.
- Ooms K., De Maeyer P., Fack V., Van Assche E., Witlox F., 2012. "Interpreting maps through the eyes of expert and novice users." *International Journal of Geographical Information Science*, 26(10): 1773-1788.
- Pei S.C. and Lin C.N., 1992. "The Detection of Dominant Points on Digital Curves by Scale- filtering." *Pattern Recognition*, 25(11): 1307-1314.
- PerkalJ., 1966a. "On the Length of Empirical Curves", *Discussion Paper 10*. Department of Geography, University of Michigan, Ann Arbor, Michigan.
- Perkal, J., 1966b. "An Attempt at Objective Generalization." *Discussion Paper 10*. Department of Geography, University of Michigan, Ann Arbor, Michigan.
- Plazanet C., Affholder J.G and Fritsch E., 1995. "The Importance of Geometric Modeling in Linear Feature Generalization", *Cartography and Geographic Information Systems*, 22(4): 291-305.
- Qian H., Zhang M., Wu F., 2016. "A New Simplification Approach Based on the Oblique-Dividing-Curve Method for Contour Lines.", *International Journal of Geo-Information*, 5(9): 153.
- Ray B. and Ray K., 1992. "An Algorithm for Detection of Dominant Points and Polygonal Approximation of Digitized Curves." *Patterns Recognition Letters*, 13(12): 849-856.
- Rattarangsi A. and Chin R.T., 1992. "Scale-Based Detection of Corners of Planar Curves." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(4): 430-449.
- Rosenfeld A. and Johnston E., 1973. "Angle Detection of Digital Curves." *Transactions on Computers*, 22: 875-878.
- Rouleau B., 1984. "Theory of Cartographic Expression and Design", *In Basic Cartography for Students and Technicians*, (Vol. 1), International Cartographic Association, Hampshire, 81-111.
- Ruas A. and Duchêne C., 2007, "A Prototype Generalisation System Based on the Multi-Agent System Paradigm". *In Generalisation of Geographic Information: Cartographic Modelling and Applications* (Mackaness W.A., A. Ruas, and L.T.Sarjakiski, Eds.), Elsevier Ltd., Amsterdam: 269-284.
- Russell, S.J. and Norvig P., 2003. "Artificial Intelligence: A Modern Approach" (Prentice Hall Series in Artificial Intelligence). 2nd ed., Upper Saddle River, NJ: Prentice Hall.
- Spakov O. and Miniotas D., 2007. "Visualization of Eye Gaze Data using Heat Maps." *Electronics and electrical engineering*, 2(74): 55-58.



- Steinke T.R., 1985. "Eye movements studies in cartography and related fields." *Cartographica*, 24(2): 40-73.
- the C. and Chin R.T., 1989. "On the Detection of Dominant Points on Digital Curves." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8): 859-872.
- Thapa, K. 1987. "Detection of Critical Points: The First Step to Automatic Line Generalization." PhD Dissertation, Report No. 379. Department of Geodetic Science and Surveying, The Ohio State Univ., Columbus, Ohio.
- Thapa, K. 1988a. "Automatic Line Generalization in Raster Data Using Zero-Crossings." *Photogrammetric Engineering and Remote Sensing*, 54(4): 511-517.
- Thapa, K. 1988b. "Critical Points Detection and Automatic Line Generalization in Raster Data Using Zero-Crossings." *The Cartographic Journal*, 25(1): 58-68.
- Töpfer, F. and Pillewizer, W. 1966. "The Principles of Selection." *The Cartographic Journal*, 3(1): 10-16.
- Visvalingam M. and Whyatt J.D., 1990. "The Douglas-Peucker Algorithm for Line Simplification: Re-evaluation through Visualization." *Computer Graphics Forum*, 9: 213-228.
- Visvalingam M. and Whyatt J.D., 1993. "Line Generalisation by Repeated Elimination of Points." *The Cartographic Journal*, 30(1): 46-51.
- Voßkühler A., Nordmeier V., Kuchinke L., Jacobs A.M., 2008. "OGAMA (Open Gaze and Mouse Analyzer): open-source software designed to analyze eye and mouse movements in slideshow study designs." *Behavior Research Methods*, 40(4): 1150-1162.
- Wang Z. and Müller J.-C., 1998. "Line Generalization Based on Analysis of Shape Characteristics." *Cartography and Geographic Information Systems*, 25(1): 3-15.
- White E.R., 1985. "Assessment of Line-Generalization Algorithms Using Characteristic Points." *The American Cartographer*, 12(1): 17-27.
- Wood C., 1995. "Perceptual responses to line simplification in a task-oriented map analysis experiment." *Cartographica*, 32(1): 22-32.
- Young L.R. and Sheena D., 1975. "Survey of eye movement recording methods." *Behavior Research Methods & Instrumentation*, 7(5): 397-429.
- Zelinsky G.J., 1996. "Using Eye Saccades to Assess the Selectivity of Search Movements." *Vision Research*, 36(14): 2177-2187.



## Παράρτημα

### Π3.1 Πρόγραμμα εντοπισμού κρίσιμων σημείων μέσω του δείκτη αναλογίας μήκους LR

```

# Anoigmaarxeiwn
f = open('C:file.txt')
# Eisagogisimeiwngammis
x=[]
y=[]
id=[]
for line in f:
    x.append(float(line[:-1].split(' ')[0]))
    y.append(float(line[:-1].split(' ')[1]))
    id.append(int(line[:-1].split(' ')[2]))
import math
# Ypologismosapostasewnmetaxisimeiwngammis
Sbp=[]
for i in range(len(id)-1):
    Sbps=math.sqrt(((x[i+1]-x[i])**2)+((y[i+1]-y[i])**2))
    Sbp.append(Sbps)
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

if len(id)==3: # Efarmogi se bend me 3 simeia
    CPnum=id[0]
elif len(id)==4: # Efarmogi se bend me 4 simeia
    for i in range(len(id)): # Ypologismos LR1
        if i in range(0,1):
            llr1=1
        elif i in range(1,len(id)-1):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            llr1=l1/s1
        elif i in range(len(id)-1,len(id)):
            llr1=1
        LR1.append(llr1)
        alr=LR1[i] # Ypologismos ALR(==LR1)
        ALR.append(alr)
        if ALR[i]==max(ALR): # Entopismos ID
            simeioupuantistoixeistomegisto ALR
            CPnum=id[i-1]
elif len(id)==5 or len(id)==6: # Efarmogi se bend me 5 'h 6 simeia -----
    for i in range(len(id)): # Ypologismos LR1, LR2
        if i in range(0,2):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            llr2=l2/s2
        elif i in range(2,len(id)-2):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
            llr2=l2/s2
        elif i in range (len(id)-2,len(id)):

```

```

        llr1=1
        l2=Sbp[i-1]+Sbp[i-2]
        s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
        llr2=l2/s2
    LR1.append(llr1)
    LR2.append(llr2)
    alr=(LR1[i]+LR2[i])/2 # Υπολογισμος ALR
    ALR.append(alr)
    if ALR[i]==max(ALR): # Entopismos ID
simeioupuantistoixeistomegisto ALR
        CPnum=id[i-1]
elif len(id)==7 or len(id)==8: # Efarmogi se bend me 7 h' 8 simeia -----
    for i in range(len(id)): # Υπολογισμος LR1, LR2, LR3
        if i in range(0,3):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i])**2)+((y[i+3]-y[i])**2))
            llr3=l3/s3
        elif i in range(3,len(id)-3):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i-3])**2)+((y[i+3]-y[i-3])**2))
            llr3=l3/s3
        elif i in range(len(id)-3,len(id)):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((x[i-3]-x[i])**2)+((y[i-3]-y[i])**2))
            llr3=l3/s3
    LR1.append(llr1)
    LR2.append(llr2)
    LR3.append(llr3)
    alr=(LR1[i]+LR2[i]+LR3[i])/3 # Υπολογισμος ALR
    ALR.append(alr)
    if ALR[i]==max(ALR): # Entopismos ID
simeioupuantistoixeistomegisto ALR

        CPnum=id[i-1]
elif len(id)>=9:
    for i in range(len(id)): # Efarmogi se bend me toulaxiston 9 simeia
        if i in range(0,4): # Υπολογισμος LR1, LR2, LR3, LR4
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i])**2)+((y[i+3]-y[i])**2))
            llr3=l3/s3

```

```

l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
s4=math.sqrt(((x[i+4]-x[i])**2)+((y[i+4]-y[i])**2))
l1r4=l4/s4
elif i in range(4,len(id)-4):
    l1=Sbp[i-1]+Sbp[i]
    s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
    l1r1=l1/s1
    l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
    s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
    l1r2=l2/s2
    l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
    s3=math.sqrt(((x[i+3]-x[i-3])**2)+((y[i+3]-y[i-3])**2))
    l1r3=l3/s3
    l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
    s4=math.sqrt(((x[i+4]-x[i-4])**2)+((y[i+4]-y[i-4])**2))
    l1r4=l4/s4
elif i in range(len(id)-4,len(id)):
    l1r1=1
    l2=Sbp[i-1]+Sbp[i-2]
    s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
    l1r2=l2/s2
    l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
    s3=math.sqrt(((x[i-3]-x[i])**2)+((y[i-3]-y[i])**2))
    l1r3=l3/s3
    l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
    s4=math.sqrt(((x[i-4]-x[i])**2)+((y[i-4]-y[i])**2))
    l1r4=l4/s4
LR1.append(l1r1)
LR2.append(l1r2)
LR3.append(l1r3)
LR4.append(l1r4)
alr=(LR1[i]+LR2[i]+LR3[i]+LR4[i])/4 # Υπολογισμος ALR
ALR.append(alr)
if ALR[i]==max(ALR): # Entopismos ID
simeiou pou antistoixei stomegisto ALR
    CPnum=id[i-1]

# ----- Syntetagmenes kai id Critical Point -----
CPx=x[CPnum]
CPy=y[CPnum]
CPid=id[CPnum]

# Exagwgiarxeiou
fout=open('C:CP.txt','w')
fout.write((str(CPx))+ ' '+ (str(CPy))+ ' '+ (str(CPid)))
fout.close()

```

### Π3.2 Πρόγραμμα τελεστή εξομάλυνσης Gauss (ανοιχτές γραμμές)

```
# Anoigmaarxeiou
f = open('C:file.txt')
# Eisagwgisimeiw narxikisgrammis
inix=[]
iniy=[]
iniid=[]
for line in f:
    inix.append(float(line[:-1].split(' ')[0]))
    iniy.append(float(line[:-1].split(' ')[1]))
    iniid.append(int(line[:-1].split(' ')[2]))
import math

# Ypologismosapodektwntimwn sigma
sigmaAccept=[]
sig=0
for i in range((len(iniid)-1)/2):
    sig=sig+0.25
    sigmaAccept.append(sig)
sigmaMax=float(len(iniid)-1)/8
sigmaAccept.append(sigmaMax)
sigma=[]
for i in range(0,1):
    sigma.append(1)

# YpologismosX,Y Gauss AkraiwnSimeiw narxisGrammis (arximexri +4sigma)
# Dimiourgiakatoptrikwnsimeiw n apo to prwtosimeio
mirrorAx=[]
mirrorAy=[]
for i in range (0, int(4*sigma[0])):
    mAx=2*inix[0]-inix[i+1]
    mAy=2*iniy[0]-iniy[i+1]
    mirrorAx.append(mAx)
    mirrorAy.append(mAy)

mirrorAx.reverse()#Antistrofipinaka - diorthwsiseirassimeiw n
mirrorAy.reverse()
# Enwsikatoptrikwnsimeiw n me arxikigrammi
arxix=mirrorAx+inix
arxiy=mirrorAy+iniy

# Ypologismos x(i-k)*g(k) giakathesimeio
xika=[]
for i in range(int(4*sigma[0]), int(8*sigma[0])):
    for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
        xikae=arxix[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2)/(2*(sigma[0]**2)))))
        xika.append(xikae)

# YpologismosXGauss
xGa=[]
xGaassist=[]
for j in range(1,(len(xika)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0]+1)):
        xGaassist.append(xika[i])
    sum(xGaassist)
```

```
xGa.append(sum(xGaassist))
xika[0:int(8*sigma[0]+1)]=[]
xGaassist[:]=[]

# Ypologismos  $y(i-k)*g(k)$  giakathesimeio
yika=[]
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        yikae=arxiy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yika.append(yikae)

# YpologismosYGauss
yGa=[]
yGaassist=[]
for j in range(1,(len(yika)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0])+1):
        yGaassist.append(yika[i])
    sum(yGaassist)
    yGa.append(sum(yGaassist))
    yika[0:int(8*sigma[0]+1)]=[]
    yGaassist[:]=[]

# YpologismosX,Y Gauss MeswnSimeiwNGrammis (+4sigma+1 apo tin arximexri -
4sigma+1 apo to peras)
# Ypologismos  $x(i-k)*g(k)$  giakathesimeio
xikm=[]
for i in range (int(4*sigma[0]), len(iniid)-int(4*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        xikme=inix[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        xikm.append(xikme)

# YpologismosXGauss
xGm=[]
xGmassist=[]
for j in range(1,(len(xikm)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0])+1):
        xGmassist.append(xikm[i])
    sum(xGmassist)
    xGm.append(sum(xGmassist))
    xikm[0:int(8*sigma[0]+1)]=[]
    xGmassist[:]=[]

# Ypologismos  $y(i-k)*g(k)$  giakathesimeio
yikm=[]
for i in range (int(4*sigma[0]), len(iniid)-int(4*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        yikme=iniy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yikm.append(yikme)

# YpologismosYGauss
yGm=[]
yGmassist=[]
```

```
for j in range(1, (len(yikm)/int(8*sigma[0]+1))+1):
    for i in range(0, int(8*sigma[0]+1)):
        yGmassist.append(yikm[i])
    sum(yGmassist)
    yGm.append(sum(yGmassist))
    yikm[0:int(8*sigma[0]+1)]=[]
    yGmassist[:]=[]

# Ypologismos X,Y Gauss akraiwnsimeiwneperatosgrammis (-4sigma mexriperas)
# Dimiourgiakatoptrikwnsimeiwne apo to teleutaiosimeio
mirrorPx=[]
mirrorPy=[]

# Dimiourgiaantistrofwnpinakwn x kai y
xrev=[]
yrev=[]
for i in range (len(iniid)):
    xrev.append(inix[i])
    yrev.append(iniy[i])
xrev.reverse()
yrev.reverse()

for i in range (0, int(4*sigma[0])):
    mPx=2*xrev[0]-xrev[i+1]
    mPy=2*yrev[0]-yrev[i+1]
    mirrorPx.append(mPx)
    mirrorPy.append(mPy)

# Enwsikatoptrikwnsimeiwne me arxikigrammi (orthiseirasimeiwne)
perasx=inix+mirrorPx
perasy=iniy+mirrorPy

# Antisxtrofipinakwnperasx kai perasy (epexergasia se antistrofouspinakes)
perasx.reverse()
perasy.reverse()

# Ypologismos x(i-k)*g(k) giakathesimeio
xikp=[]
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0]+1)):
        xikpe=perasx[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2)/(2*(sigma[0]**2)))))
        xikp.append(xikpe)

# YpologismosXGauss
xGp=[]
xGpassist=[]
for j in range(1, (len(xikp)/int(8*sigma[0]+1))+1):
    for i in range(0, int(8*sigma[0]+1)):
        xGpassist.append(xikp[i])
    sum(xGpassist)
    xGp.append(sum(xGpassist))
    xikp[0:int(8*sigma[0]+1)]=[]
    xGpassist[:]=[]

# Ypologismos y(i-k)*g(k) giakathesimeio
yikp=[]
```



```
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        yikpe=perasy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yikp.append(yikpe)

# YpologismosYGauss
yGp=[]
yGpassist=[]
for j in range(1,(len(yikp)/int(8*sigma[0])+1)+1):
    for i in range(0,int(8*sigma[0])+1):
        yGpassist.append(yikp[i])
    sum(yGpassist)
    yGp.append(sum(yGpassist))
    yikp[0:int(8*sigma[0])+1]=[]
    yGpassist[:]=[]

# AntistrofipinakwnxGp kai yGp - diorthwsiseirassimeiw
xGp.reverse()
yGp.reverse()
# EnwsipinakwnxGa-xGm-xGp / yGa-yGm-yGp
xu=xGa+xGm+xGp
yu=yGa+yGm+yGp
id=iniid

# ----- X kai Y Gauss -----
x=[]
y=[]

# Diorthwsix,y apo sfalmastroggylopoihsisvarous
# Ypologismosathroismatosepimerousvarwn
gsk=0
for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
    gsk=gsk+(1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))

# Diorthwsisyntetagmenwn
for i in range(len(id)):
    xx=xu[i]/gsk
    yy=yu[i]/gsk
    x.append(xx)
    y.append(yy)

# Exagwgiarxeiouapodektwntimwn sigma
gout=open('C:SigmaAccept.txt', 'w')
for i in range(len(sigmaAccept)-1): # (stoixeiaektostousigmaMax)
    gout.write(str(sigmaAccept[i])+' '+'\n')

gout.write('sigmaMax:'+str(sigmaMax))
gout.close()

# Exagwgiarxeiou
fout=open('C:ExomGauss.txt', 'w')
for i in range (len(id)):
    fout.write(str(x[i])+' '+'(str(y[i]))+' '+'(str(id[i]))+''\n')

fout.close()
```

### Π3.2Κ Πρόγραμμα τελεστή εξομάλυνσης Gauss (κλειστές γραμμές)

```
# Anoigmaarxeiou
f = open('C:file.txt')

# Eisagwgisimeiw narxikisgrammis
inix=[]
iniy=[]
iniid=[]
for line in f:
    inix.append(float(line[:-1].split(' ')[0]))
    iniy.append(float(line[:-1].split(' ')[1]))
    iniid.append(int(line[:-1].split(' ')[2]))

import math

# Ypologismosapodektwntimwn sigma
sigmaAccept=[]
sig=0
for i in range((len(iniid)-1)/2):
    sig=sig+0.25
    sigmaAccept.append(sig)

sigmaMax=float(len(iniid)-1)/8
sigmaAccept.append(sigmaMax)

sigma=[]
for i in range(0,1):
    sigma.append(1)

# ===== YpologismosXGauss =====
# ----- Ypologismos x(i-k)*g(k) giakathesimeio -----
xikm=[]

# Simeia l mexri n-4
for i in range(0, len(iniid)-int(4*sigma[0])):
    for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
        xikme=inix[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        xikm.append(xikme)

# Simeia apo n-3 mexri n
inixassist=inix+inix
for i in range(len(iniid)-int(4*sigma[0]), len(iniid)):
    for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
        xikme=inixassist[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        xikm.append(xikme)

# ----- YpologismosXGauss -----
xGm=[]
xGmassist=[]
for j in range(1, (len(xikm)/int(8*sigma[0])+1)+1):
    for i in range(0, int(8*sigma[0])+1):
        xGmassist.append(xikm[i])
    sum(xGmassist)
```

```

xGm.append(sum(xGmassist))
xikm[0:int(8*sigma[0]+1)]=[]
xGmassist[:]=[]

# ===== YpologismosYGauss =====
# ----- Ypologismos y(i-k)*g(k) giakathesimeio -----
yikm=[]

# Simeia 1 mexri n-4
for i in range(0, len(iniid)-int(4*sigma[0])):
    for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
        yikme=iniy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yikm.append(yikme)

# Simeia apo n-3 mexri n
iniyassist=iniy+iniy
for i in range(len(iniid)-int(4*sigma[0]), len(iniid)):
    for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
        yikme=iniyassist[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yikm.append(yikme)

# ----- YpologismosYGauss -----
yGm=[]
yGmassist=[]
for j in range(1, (len(yikm)/int(8*sigma[0]+1))+1):
    for i in range(0, int(8*sigma[0])+1):
        yGmassist.append(yikm[i])
    sum(yGmassist)
    yGm.append(sum(yGmassist))
    yikm[0:int(8*sigma[0]+1)]=[]
    yGmassist[:]=[]

# ===== Telika X kai Y Gauss =====
x=[]
y=[]
id=iniid

# ----- Diorthwsix,y apo sfalmastroggylopoihsisvarous -----
# Ypologismosathroismatosepimerousvarwn
gsk=0
for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
    gsk=gsk+(1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))

# Diorthwsisyntetagmenwn
for i in range(len(id)):
    xx=xGm[i]/gsk
    yy=yGm[i]/gsk
    x.append(xx)
    y.append(yy)

# -----
# Exagwggiarxeiouapodektwntimwn sigma
gout=open('C:SigmaAcceptClosedLine.txt', 'w')

```

```
for i in range(len(sigmaAccept)-1): # (στοιχειαektostousigmaMax)
    gout.write(str(sigmaAccept[i])+' '+'\n')

gout.write('sigmaMax:'+str(sigmaMax))
gout.close()

# Exagwgiarxeiou
fout=open('C:ExomGaussClosedLine.txt', 'w')
for i in range (len(id)):
    fout.write(str(x[i])+' '+'+(str(y[i]))+' '+'+(str(id[i]))+'\n')

fout.close()

f.close()
```

### Π3.3 Πρόγραμμα εντοπισμού καμπών

```
# Anoigmaarxeiwn
f = open('C:file.txt')

# Eisagwgisimeiwngammis
x=[]
y=[]
id=[]
for line in f:
    x.append(float(line[:-1].split(' ')[0]))
    y.append(float(line[:-1].split(' ')[1]))
    id.append(int(line[:-1].split(' ')[2]))

# ===== Entopismos Bends =====

# Ypologismosapostasewnmetaxitwnsimeiwntis grammis
distx=[]
disty=[]
distid=[]
for i in range(len(id)-1):
    dx=x[i+1]-x[i]
    dy=y[i+1]-y[i]
    distx.append(dx)
    disty.append(dy)
    distid.append(i)

import math

# Ypologismosgwniasdiefthinsiseftthgrammwntmimatwngrammis
w=[]
for i in range (len(distid)):
    if distx[i]>0 and disty[i]>0:
        ww=
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i]))))* (200/math.pi))
    elifdistx[i]>0 and disty[i]<0:
        ww= 200-
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i]))))* (200/math.pi))
    elifdistx[i]<0 and disty[i]<0:
        ww=
200+((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i]))))* (200/math.pi))
    elifdistx[i]<0 and disty[i]>0:
        ww= 400-
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i]))))* (200/math.pi))
    elifdistx[i]==0 and disty[i]>0:
        ww=0
    elifdistx[i]==0 and disty[i]<0:
        ww=200
    elifdistx[i]>0 and disty[i]==0:
        ww=100
    elifdistx[i]<0 and disty[i]==0:
        ww=300
    w.append(ww)

# Elegxosgiapithaniafairesi h' prosthesi 400g
wf=[]
for i in range(len(w)):
    if w[i]>400:
```

```
        wwf=w[i]-400
elif w[i]<0:
        wwf=w[i]+400
else:
        wwf=w[i]
wf.append(wwf)

# Metatropi grad se rad
wfr=[]
for i in range(len(wf)):
        wwfr=(math.pi*wf[i])/200
        wfr.append(wwfr)

# Ypologismosgwniasthlasissimeiwn 2 ews n-1
slo=[]
for i in range(len(wfr)-1):
        slop=wfr[i+1]-wfr[i]
        slo.append(slop)

# Diorthwsigwniwnthlasis (megaliterwntou pi) / Telikesgwniesthlasis
sl=[]
for i in range(len(slo)):
        if (math.pi)<slo[i]<(math.pi*2):
                ssl=slo[i]-(math.pi*2)
        elif -(math.pi*2)<slo[i]<-(math.pi):
                ssl=slo[i]+(math.pi*2)
        else:
                ssl=slo[i]
        sl.append(ssl)

# Entopismos Inflection Points
# (entopizontaidiadoxikasimeia me antithetiklisi / katagrafontai kai ta 2)
infpx=[]
infpy=[]
infpid=[]
for i in range (len(sl)-1):
        if sl[i]<0 and sl[i+1]>0 or sl[i]>0 and sl[i+1]<0:
                infpx.append(x[i+1])
                infpx.append(x[i+2])
                infpy.append(y[i+1])
                infpy.append(y[i+2])
                infpid.append(id[i+1])
                infpid.append(id[i+2])

# Entopismosmesou (mip) metaxi inflection points kai dimiourgiapinaka me ip-
mip-ip
ipmipx=[]
ipmipy=[]
ipmipid=[]
for i in range(0, len(infpid)-1, 2):
        ipmipxx=(infpx[i]+infpx[i+1])/2
        ipmipyy=(infpy[i]+infpy[i+1])/2
        ipmipidd=(float(infpid[i])+float(infpid[i+1]))/2
        ipmipx.append(infpx[i])
        ipmipx.append(ipmipxx)
        ipmipx.append(infpx[i+1])
        ipmipy.append(infpy[i])
        ipmipy.append(ipmipyy)
```

```
    ipmipy.append(infpy[i+1])
    ipmipid.append(infpid[i])
    ipmipid.append(ipmipidd)
    ipmipid.append(infpid[i+1])

# Afairesidiplwnkatagrafwn (periptwseis bends triwnsimeiw)
ipmipxassist=[]
ipmipyassist=[]
ipmipidassist=[]
for i in range(1,len(ipmipid)):
    if ipmipid[i]!=ipmipid[i-1]:
        ipmipxassist.append(ipmipx[i])
        ipmipyassist.append(ipmipy[i])
        ipmipidassist.append(ipmipid[i])

# Prosthikitouprwtousimeioustouspinakesipmipassist
ipmipxassist.insert(0, ipmipx[0])
ipmipyassist.insert(0, ipmipy[0])
ipmipidassist.insert(0, ipmipid[0])

# Entopismos non-inflection points grammis
noninfpx=[]
noninfpy=[]
noninfpid=[]
for i in range(len(id)):
    if id[i] not in ipmipidassist:
        noninfpx.append(x[i])
        noninfpy.append(y[i])
        noninfpid.append(id[i])

# ----- Dimiourgiatelikisgrammis (me nip, ip kai mip) -----
# Enwsipinakwnipmipassisst-noninfp
xvm=ipmipxassist+noninfpx
yvm=ipmipyassist+noninfpy
idvm=ipmipidassist+noninfpid

# Telikoipinakesgrammis
vmx=[]
vmy=[]
vmid=[]

# Dimiourgiataxinomimenoupinakaidvm
idvm.sort()
idvmsort=idvm

# Epanadimiourgiapinakaidvm
idvm=ipmipidassist+noninfpid

# Taxinomisipinakwnx,y me vasi to id
for i in range(len(idvmsort)):
    for j in range(len(idvm)):
        if idvmsort[i]==idvm[j]:
            vmx.append(xvm[j])
            vmy.append(yvm[j])

# Dimiourgiatelikoupinaka id
vmid=idvmsort
```

```
# ===== Idiotites Bends =====

# Ypologismos 'i' twnmipstintelikigrammi
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# ----- YpologismosMikousGrammisVasis Bends -----
LBLbend=[]
# Prwto bend
bl=math.sqrt(((vmx[0]-vmx[iip[0]])**2)+((vmy[0]-vmy[iip[0]])**2))
LBLbend.append(bl)

# Endiamesa bends
for i in range(len(iip)-1):
    bl=math.sqrt(((vmx[iip[i]]-vmx[iip[i+1]])**2)+((vmy[iip[i]]-
vmy[iip[i+1]])**2))
    LBLbend.append(bl)

# Teleutaio bend
bl=math.sqrt(((vmx[iip[len(iip)-1]]-vmx[len(vmid)-1])**2)+((vmy[iip[len(iip)-
1]]-vmy[len(vmid)-1])**2))
LBLbend.append(bl)

# ----- YpologismosEmbadou Bends -----
Areabend=[]
# Prwto bend
area=[]
for i in range(0, iip[0]+1):
    if i==0:
        ar=0.5*(vmy[0]*(vmx[iip[0]]-vmx[i+1]))
    elif i>0 and i<iip[0]:
        ar=0.5*(vmy[i]*(vmx[i-1]-vmx[i+1]))
    elif i==iip[0]:
        ar=0.5*(vmy[iip[0]]*(vmx[i-1]-vmx[0]))
    area.append(ar)

areasum=math.fabs(sum(area))
Areabend.append(areasum)

# Endiamesa bends
area=[]
for j in range(len(iip)-1):
    for i in range(iip[j], iip[j+1]+1):
        if i==iip[j]:
            ar=0.5*(vmy[iip[j]]*(vmx[iip[j+1]]-vmx[i+1]))
        elif i>iip[j] and i<iip[j+1]:
            ar=0.5*(vmy[i]*(vmx[i-1]-vmx[i+1]))
        elif i==iip[j+1]:
            ar=0.5*(vmy[iip[j+1]]*(vmx[i-1]-vmx[iip[j]]))
        area.append(ar)
    areasum=math.fabs(sum(area))
    area[:]=[]
    Areabend.append(areasum)

# Teleutaio bend
```



```
area=[]
for i in range(iip[len(iip)-1], len(vmid)):
    if i==iip[len(iip)-1]:
        ar=0.5*(vmy[iip[len(iip)-1]]*(vmx[len(vmid)-1]-vmx[i+1]))
    elif i>iip[len(iip)-1] and i<len(vmid)-1:
        ar=0.5*(vmy[i]*(vmx[i-1]-vmx[i+1]))
    elif i==len(vmid)-1:
        ar=0.5*(vmy[len(vmid)-1]*(vmx[i-1]-vmx[iip[len(iip)-1]]))
    area.append(ar)

areasum=math.fabs(sum(area))
Areabend.append(areasum)

# ----- Entopismos Koryfwn Bends (ALR method) -----
# Ypologismos apostasewn metaxisimeiwvn
Sbp=[]
for i in range(len(vmid)-1):
    Sbps=math.sqrt(((vmx[i+1]-vmx[i])**2)+((vmy[i+1]-vmy[i])**2))
    Sbp.append(Sbps)

# Koryfes bends
CPxbend=[]
CPybend=[]
CPidbend=[]

# Prwto bend
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

if iip[0]+1==3: # bend 3 simeiwvn
    CPnum=vmid[0]
elif iip[0]+1==4: # bend 4 simeiwvn
    for i in range(0, iip[0]+1):
        if i in range(0,1):
            llr1=1
        elif i in range(1,iip[0]+1-1):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-1])**2))
            llr1=l1/s1
        elif i in range(iip[0]+1-1,iip[0]+1):
            llr1=1
        LR1.append(llr1)
        for t in range(len(LR1)):
            alr=LR1[t]
        ALR.append(alr)
        for v in range(len(ALR)):
            if ALR[v]==max(ALR):
                CPnum=v
elif iip[0]+1==5 or iip[0]+1==6: # bend 5 'h 6 simeiwvn
    for i in range(0, iip[0]+1):
        if i in range(0,2):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
```

```

        llr2=l2/s2
    elifi in range(2,iip[0]+1-2):
        l1=Sbp[i-1]+Sbp[i]
        s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))
        llr1=l1/s1
        l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))
        llr2=l2/s2
    elifi in range(iip[0]+1-2,iip[0]+1):
        llr1=1
        l2=Sbp[i-1]+Sbp[i-2]
        s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
        llr2=l2/s2
    LR1.append(llr1)
    LR2.append(llr2)
    for t in range(len(LR1)):
        alr=(LR1[t]+LR2[t])/2
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v
elifiip[0]+1==7 or iip[0]+1==8: # bend 7 h' 8 simeiw
    for i in range(0, iip[0]+1):
        if i in range(0,3):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
        elifi in range(3,iip[0]+1-3):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))
            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))
            llr3=l3/s3
        elifi in range(iip[0]+1-3,iip[0]+1):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
    LR1.append(llr1)
    LR2.append(llr2)
    LR3.append(llr3)
    for t in range(len(LR1)):

```

```

        alr=(LR1[t]+LR2[t]+LR3[t])/3
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v
elif iip[0]+1>=9: # bend tou laxiston 9 simeiw
    for i in range(0, iip[0]+1):
        if i in range(0,4):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i])**2)+((vmy[i+4]-vmy[i])**2))
            llr4=l4/s4
        elif i in range(4,iip[0]+1-4):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))
            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))
            llr3=l3/s3
            l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i-4])**2)+((vmy[i+4]-vmy[i-
4])**2))
            llr4=l4/s4
        elif i in range(iip[0]+1-4,iip[0]+1):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
            s4=math.sqrt(((vmx[i-4]-vmx[i])**2)+((vmy[i-4]-vmy[i])**2))
            llr4=l4/s4
    LR1.append(llr1)
    LR2.append(llr2)
    LR3.append(llr3)
    LR4.append(llr4)
    for t in range(len(LR1)):
        alr=(LR1[t]+LR2[t]+LR3[t]+LR4[t])/4
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v

```

```
CPxbend.append(vmx[CPnum])
CPybend.append(vmy[CPnum])
CPidbend.append(vmid[CPnum])

# Endiamesa bends
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

# Ypologismosarithmousimeiwkathe bend
n=[]
for t in range(len(iip)-1):
    for i in range(iip[0], iip[1]+1):
        nn=(iip[1]+1)-(iip[0])
        n.append(nn)
        iip[0:1]=[]

# Epanadimiourgiaiip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# Ypologismos LR1, LR2, LR3, LR4
for t in range(len(n)): #(epanalipseis oses to plithostwnendiameswn bends)
    if n[0]==3: # bend 3 simeiw
        for i in range(iip[0], iip[1]+1):
            if i==iip[0]:
                llr1=1 #(afthaireti timi, elaxisti)
            elif i==iip[0]+1:
                llr1=1.5 # (afthairetitimi, megalyteri apo
ypoloipestou bend)
            elif i==iip[1]:
                llr1=1 #(afthaireti timi, elaxisti)
                LR1.append(llr1)
            iip[0:1]=[]
            n[0:1]=[]
        elif n[0]==4: # bend 4 simeiw
            for i in range(iip[0], iip[1]+1):
                if i in range(iip[0], iip[0]+1):
                    llr1=1
                elif i in range(iip[0]+1, iip[1]+1-1):
                    l1=Sbp[i-1]+Sbp[i]
                    s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))
                    llr1=l1/s1
                elif i in range(iip[1]+1-1, iip[1]+1):
                    llr1=1
                LR1.append(llr1)
            iip[0:1]=[]
            n[0:1]=[]
        elif n[0]==5 or n[0]==6: # bend 5 'h 6 simeiw
            for i in range(iip[0], iip[1]+1):
                if i in range(iip[0], iip[0]+2):
                    llr1=1
                    l2=Sbp[i]+Sbp[i+1]
```

```

                s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-
vmy[i])**2))
                l1r2=l2/s2
            elif i in range(iip[0]+2, iip[1]+1-2):
                l1=Sbp[i-1]+Sbp[i]
                s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))
                l1r1=l1/s1
                l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
                s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-
vmy[i-2])**2))
                l1r2=l2/s2
            elif i in range(iip[1]+1-2, iip[1]+1):
                l1r1=1
                l2=Sbp[i-1]+Sbp[i-2]
                s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-
vmy[i])**2))
                l1r2=l2/s2
                LR1.append(l1r1)
                LR2.append(l1r2)
            iip[0:1]=[
            n[0:1]=[
            elif n[0]==7 or n[0]==8: # bend 7 h' 8 simeiw
            for i in range(iip[0], iip[1]+1):
                if i in range(iip[0], iip[0]+3):
                    l1r1=1
                    l2=Sbp[i]+Sbp[i+1]
                    s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-
vmy[i])**2))
                    l1r2=l2/s2
                    l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
                    s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-
vmy[i])**2))
                    l1r3=l3/s3
                elif i in range(iip[0]+3, iip[1]+1-3):
                    l1=Sbp[i-1]+Sbp[i]
                    s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))
                    l1r1=l1/s1
                    l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
                    s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-
vmy[i-2])**2))
                    l1r2=l2/s2
                    l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
                    s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-
vmy[i-3])**2))
                    l1r3=l3/s3
                elif i in range(iip[1]+1-3, iip[1]+1):
                    l1r1=1
                    l2=Sbp[i-1]+Sbp[i-2]
                    s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-
vmy[i])**2))
                    l1r2=l2/s2
                    l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
                    s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-
vmy[i])**2))
                    l1r3=l3/s3
                LR1.append(l1r1)

```

```

LR2.append(llr2)
LR3.append(llr3)
iip[0:1]=[
n[0:1]=[
elifn[0]>=9: # bend toulaxiston 9 simeiwn
for i in range(iip[0], iip[1]+1):
if i in range(iip[0], iip[0]+4):
llr1=1
l2=Sbp[i]+Sbp[i+1]
s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-
vmy[i])**2))

llr2=l2/s2
l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-
vmy[i])**2))

llr3=l3/s3
l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
s4=math.sqrt(((vmx[i+4]-vmx[i])**2)+((vmy[i+4]-
vmy[i])**2))

llr4=l4/s4
elifi in range(iip[0]+4, iip[1]+1-4):
l1=Sbp[i-1]+Sbp[i]
s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))

llr1=l1/s1
l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-
vmy[i-2])**2))

llr2=l2/s2
l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-
vmy[i-3])**2))

llr3=l3/s3
l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
s4=math.sqrt(((vmx[i+4]-vmx[i-4])**2)+((vmy[i+4]-
vmy[i-4])**2))

llr4=l4/s4
elifi in range(iip[1]+1-4, iip[1]+1):
llr1=1
l2=Sbp[i-1]+Sbp[i-2]
s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-
vmy[i])**2))

llr2=l2/s2
l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-
vmy[i])**2))

llr3=l3/s3
l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
s4=math.sqrt(((vmx[i-4]-vmx[i])**2)+((vmy[i-4]-
vmy[i])**2))

llr4=l4/s4
LR1.append(llr1)
LR2.append(llr2)
LR3.append(llr3)
LR4.append(llr4)
iip[0:1]=[
n[0:1]=[

```

```
# Epanadimiourgia iip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# Epanypologismos arithmousimeiw nkathe bend
n=[]
for t in range(len(iip)-1):
    for i in range(iip[0], iip[1]+1):
        nn=(iip[1]+1)-(iip[0])
    n.append(nn)
    iip[0:1]=[]

# Ypologismos ALR
for t in range(len(n)):
    if n[t]==3: # bend 3 simeiw n
        for i in range(0, n[t]):
            alr=LR1[i]
            ALR.append(alr)
        LR1[0:n[t]]=[]
    elif n[t]==4: # bend 4 simeiw n
        for i in range(0, n[t]):
            alr=LR1[i]
            ALR.append(alr)
        LR1[0:n[t]]=[]
    elif n[t]==5 or n[t]==6: # bend 5 'h 6 simeiw n
        for i in range(0, n[t]):
            alr=(LR1[i]+LR2[i])/2
            ALR.append(alr)
        LR1[0:n[t]]=[]
        LR2[0:n[t]]=[]
    elif n[t]==7 or n[t]==8: # bend 7 h' 8 simeiw n
        for i in range(0, n[t]):
            alr=(LR1[i]+LR2[i]+LR3[i])/3
            ALR.append(alr)
        LR1[0:n[t]]=[]
        LR2[0:n[t]]=[]
        LR3[0:n[t]]=[]
    elif n[t]>=9: # bend tou laxiston 9 simeiw n
        for i in range(0, n[t]):
            alr=(LR1[i]+LR2[i]+LR3[i]+LR4[i])/4
            ALR.append(alr)
        LR1[0:n[t]]=[]
        LR2[0:n[t]]=[]
        LR3[0:n[t]]=[]
        LR4[0:n[t]]=[]

# Entopismos 'i' koryfwn
CPnum=[] #('i' apo tin arxitou bend)
ALRassist=[]
for t in range(len(n)):
    for j in range(0, n[t]):
        ALRassist.append(ALR[j])
    for i in range(len(ALRassist)):
        if ALRassist[i]==max(ALRassist):
            CPnum.append(i)
    ALRassist[:]=[]
```

```
ALR[0:n[t]]=[]

# Epanadimiourgiaiip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# Ypologismosx,y,idkoryfwn bends
# (koryfi == x,y,id['i' touarxikousimeioukathe bend + CPnum])
for t in range(len(n)):
    CPxbend.append(vmx[iip[0]+CPnum[0]])
    CPybend.append(vmy[iip[0]+CPnum[0]])
    CPidbend.append(vmid[iip[0]+CPnum[0]])
    iip[0:1]=[]
    CPnum[0:1]=[]

# Epanadimiourgiaiip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# Teleutaio bend
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

if len(vmid)-(iip[len(iip)-1])==3: # bend 3 simeiw
    CPnum=1
elif len(vmid)-(iip[len(iip)-1])==4: # bend 4 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+1):
            l1r1=1
        elif i in range(iip[len(iip)-1]+1, len(vmid)-1):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-1])**2))
            l1r1=l1/s1
        elif i in range(len(vmid)-1, len(vmid)):
            l1r1=1
        LR1.append(l1r1)
        for t in range(len(LR1)):
            alr=LR1[t]
        ALR.append(alr)
        for v in range(len(ALR)):
            if ALR[v]==max(ALR):
                CPnum=v
elif len(vmid)-(iip[len(iip)-1])==5 or len(vmid)-(iip[len(iip)-1])==6: # bend
5 'h 6 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+2):
            l1r1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            l1r2=l2/s2
```



```

elifi in range(iip[len(iip)-1]+2, len(vmid)-2):
    l1=Sbp[i-1]+Sbp[i]
    s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))

    llr1=l1/s1
    l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
    s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))

    llr2=l2/s2
elifi in range(len(vmid)-2, len(vmid)):
    llr1=1
    l2=Sbp[i-1]+Sbp[i-2]
    s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
    llr2=l2/s2
LR1.append(llr1)
LR2.append(llr2)
for t in range(len(LR1)):
    alr=(LR1[t]+LR2[t])/2
ALR.append(alr)
for v in range(len(ALR)):
    if ALR[v]==max(ALR):
        CPnum=v
elif len(vmid)-(iip[len(iip)-1])==7 or len(vmid)-(iip[len(iip)-1])==8: # bend
7 h' 8 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+3):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
        elifi in range(iip[len(iip)-1]+3, len(vmid)-3):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))

            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))

            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))

            llr3=l3/s3
        elifi in range(len(vmid)-3, len(vmid)):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
    LR1.append(llr1)
    LR2.append(llr2)
    LR3.append(llr3)
    for t in range(len(LR1)):

```

```

        alr=(LR1[t]+LR2[t]+LR3[t])/3
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v
elif len(vmid)-(iip[len(iip)-1])>=9: # bend toulaxiston 9 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+4):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i])**2)+((vmy[i+4]-vmy[i])**2))
            llr4=l4/s4
        elif i in range(iip[len(iip)-1]+4, len(vmid)-4):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))
            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))
            llr3=l3/s3
            l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i-4])**2)+((vmy[i+4]-vmy[i-
4])**2))
            llr4=l4/s4
        elif i in range(len(vmid)-4, len(vmid)):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
            s4=math.sqrt(((vmx[i-4]-vmx[i])**2)+((vmy[i-4]-vmy[i])**2))
            llr4=l4/s4
    LR1.append(llr1)
    LR2.append(llr2)
    LR3.append(llr3)
    LR4.append(llr4)
    for t in range(len(LR1)):
        alr=(LR1[t]+LR2[t]+LR3[t]+LR4[t])/4
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v

```

```
CPxbend.append(vmx[iip[len(iip)-1]+CPnum]) #(koryfi == x,y,id['i'  
touarxikousimeiououteleutaiau bend + CPnum])  
CPybend.append(vmy[iip[len(iip)-1]+CPnum])  
CPidbend.append(vmid[iip[len(iip)-1]+CPnum])  
  
# YpologismosVathous Bends (apostasikoryfis apo grammivasis)  
Wbend=[]  
# Ptwto bend  
d=(math.fabs(((-(vmy[iip[0]]-vmy[0])/(vmx[iip[0]]-  
vmx[0]))) *CPxbend[0])+CPybend[0]+((vmy[iip[0]]-vmy[0])/(vmx[iip[0]]-  
vmx[0])) *vmx[0])-vmy[0]))/(math.sqrt(((-(vmy[iip[0]]-vmy[0])/(vmx[iip[0]]-  
vmx[0])) **2)+1))  
Wbend.append(d)  
  
# Endiamesa bends  
for i in range(len(iip)-1):  
    d=(math.fabs(((-(vmy[iip[i+1]]-vmy[iip[i]])/(vmx[iip[i+1]]-  
vmx[iip[i]]))) *CPxbend[i+1])+CPybend[i+1]+((vmy[iip[i+1]]-  
vmy[iip[i]])/(vmx[iip[i+1]]-vmx[iip[i])) *vmx[iip[i]])-  
vmy[iip[i]]))/(math.sqrt(((-(vmy[iip[i+1]]-vmy[iip[i]])/(vmx[iip[i+1]]-  
vmx[iip[i]]))) **2)+1))  
    Wbend.append(d)  
  
# Teleutaio bend  
d=(math.fabs(((-(vmy[len(vmid)-1]-vmy[iip[len(iip)-1]])/(vmx[len(vmid)-1]-  
vmx[iip[len(iip)-1]]))) *CPxbend[len(CPidbend)-1])+CPybend[len(CPidbend)-  
1]+((vmy[len(vmid)-1]-vmy[iip[len(iip)-1]])/(vmx[len(vmid)-1]-  
vmx[iip[len(iip)-1]])) *vmx[iip[len(iip)-1]])-vmy[iip[len(iip)-  
1]]))/(math.sqrt(((-(vmy[len(vmid)-1]-vmy[iip[len(iip)-1]])/(vmx[len(vmid)-  
1]-vmx[iip[len(iip)-1]]))) **2)+1))  
Wbend.append(d)  
  
# ----- ProsdiorismosKampylotitas Bends -----  
Curvbend=[]  
  
# Euresigwniwnthlasispouantistoixounstiskoryfes tis grammis  
CurvCP=[]  
for i in range(len(id)):  
    for j in range(len(CPidbend)):  
        if CPidbend[j]==id[i]: # antisoixisikoryfwn bends  
stinarxikigrammi  
        CurvCP.append(sl[i-1]) # euresigwniasthlasispouantistoixei  
se kathekoryfi  
  
for i in range(len(CurvCP)): # prosdiorismoskampylotitas me kritirio to  
prosimo tis gwniasthlasis tis koryfis  
    if CurvCP[i]>0:  
        Curvbend.append(1)  
    elifCurvCP[i]<0:  
        Curvbend.append(-1)  
  
# ----- YpologismosArithmouSimeiwN Bends -----  
NumPbend=[]  
NumPbend.append(iip[0]+1) # ArithmossimeiwNprwtou bend  
for i in range(len(n)):  
    NumPbend.append(n[i])  
  
NumPbend.append(len(vmid)-(iip[len(iip)-1])) # ArithmossimeiwNteleutaiauobend
```

```
# ExagwgiArxeiou
fout=open('C:VM.txt', 'w')
for i in range(len(CPidbend)):
    fout.write('Bend'+str(i+1)+' '+'CPx:'+str(CPxbend[i])+
'+CPy:'+str(CPybend[i])+ '+'CPid:'+str(CPidbend[i])+
'+LengthBL:'+str(LBLbend[i])+ '+'Area:'+str(Areabend[i])+
'+Depth:'+str(Wbend[i])+ '+'Curvature:'+str(Curvbend[i])+
'+NumOfPoints:'+str(NumPbend[i]))+'\n')

fout.close()
```

### Π3.4 Πρόγραμμα Ομοπαράλληλου Μετασχηματισμού

```
# Anoigmaarxeiwn
f = open('C:Gauss.txt')
g = open('C:ICPointsA.txt')
k = open('C:ICPointsT.txt')

# Eisagwgisimeiwneoxomalimenisgrammis
xg=[]
yg=[]
idg=[]
for line in f:
    xg.append(float(line[:-1].split(' ')[0]))
    yg.append(float(line[:-1].split(' ')[1]))
    idg.append(int(line[:-1].split(' ')[2]))

# Eisagogi Inflection kai Critical points arxikisgrammis
xa=[]
ya=[]
ida=[]
for line in g:
    xa.append(float(line[:-1].split(' ')[0]))
    ya.append(float(line[:-1].split(' ')[1]))
    ida.append(int(line[:-1].split(' ')[2]))

# Eisagogi Inflection kai Critical points exomalimenisgrammis
xt=[]
yt=[]
idt=[]
for line in k:
    xt.append(float(line[:-1].split(' ')[0]))
    yt.append(float(line[:-1].split(' ')[1]))
    idt.append(int(line[:-1].split(' ')[2]))

# Yplogismossintelestwna,b,c,d,e,fafinikoumetasximatismou
a=((yt[1]*xa[0])-(yt[2]*xa[0])+(yt[2]*xa[1])-(yt[0]*xa[1])+(yt[0]*xa[2])-(yt[1]*xa[2]))/((xt[0]*yt[1])-(xt[0]*yt[2])-(yt[0]*xt[1])+(yt[0]*xt[2])+(xt[1]*yt[2])-(xt[2]*yt[1]))
b=((xt[2]*xa[0])-(xt[1]*xa[0])+(xt[0]*xa[1])-(xt[2]*xa[1])+(xt[1]*xa[2])-(xt[0]*xa[2]))/((xt[0]*yt[1])-(xt[0]*yt[2])-(yt[0]*xt[1])+(yt[0]*xt[2])+(xt[1]*yt[2])-(xt[2]*yt[1]))
c=((xt[1]*yt[2]*xa[0])-(yt[1]*xt[2]*xa[0])+(xt[2]*yt[0]*xa[1])-(yt[2]*xt[0]*xa[1])+(xt[0]*yt[1]*xa[2])-(xt[1]*yt[0]*xa[2]))/((xt[0]*yt[1])-(xt[0]*yt[2])-(yt[0]*xt[1])+(yt[0]*xt[2])+(xt[1]*yt[2])-(xt[2]*yt[1]))
d=((yt[1]*ya[0])-(yt[2]*ya[0])+(yt[2]*ya[1])-(yt[0]*ya[1])+(yt[0]*ya[2])-(yt[1]*ya[2]))/((xt[0]*yt[1])-(xt[0]*yt[2])-(yt[0]*xt[1])+(yt[0]*xt[2])+(xt[1]*yt[2])-(xt[2]*yt[1]))
e=((xt[2]*ya[0])-(xt[1]*ya[0])+(xt[0]*ya[1])-(xt[2]*ya[1])+(xt[1]*ya[2])-(xt[0]*ya[2]))/((xt[0]*yt[1])-(xt[0]*yt[2])-(yt[0]*xt[1])+(yt[0]*xt[2])+(xt[1]*yt[2])-(xt[2]*yt[1]))
f=((xt[1]*yt[2]*ya[0])-(yt[1]*xt[2]*ya[0])+(xt[2]*yt[0]*ya[1])-(xt[0]*yt[2]*ya[1])+(xt[0]*yt[1]*ya[2])-(xt[1]*yt[0]*ya[2]))/((xt[0]*yt[1])-(xt[0]*yt[2])-(yt[0]*xt[1])+(yt[0]*xt[2])+(xt[1]*yt[2])-(xt[2]*yt[1]))

# Ypologismosnewsintetagmenwn apo afinikometasximatismo
# (affinkosefarmozetaistasimeia 2-->n-1 / stoprwto kai to teleutaioeisagontai oi syntetagmenestwnantisoixwnsimeiwne tis grammis Gauss
xaff=[]
```

```
yaaff=[]
idaaff=[]

xaff.append(xg[0])
yaaff.append(yg[0])
idaaff.append(idg[0])

for i in range(1, len(idg)-1):
    xaffine=(a*xg[i])+(b*yg[i])+c
    yaaffine=(d*xg[i])+(e*yg[i])+f
    xaff.append(xaffine)
    yaaff.append(yaaffine)
    idaaff.append(i+1)

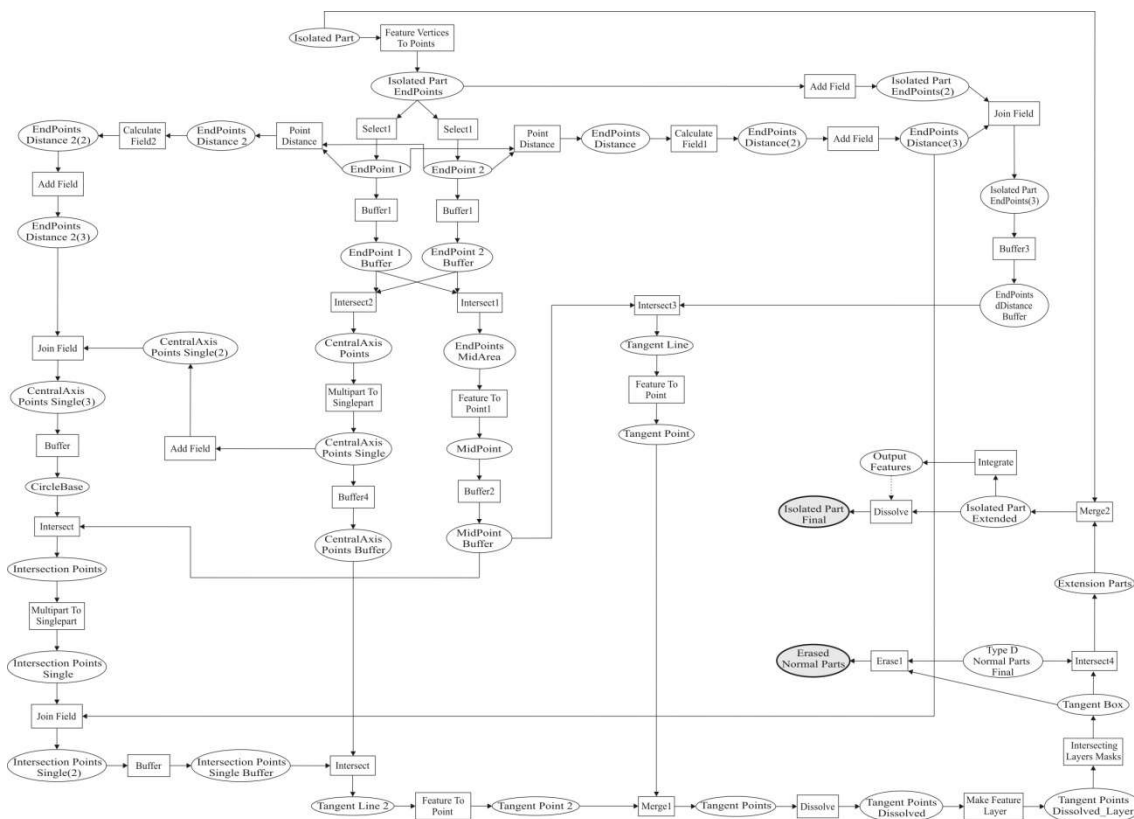
xaff.append(xg[len(idg)-1])
yaaff.append(yg[len(idg)-1])
idaaff.append(idg[len(idg)-1])

# Exagwgiarxeiou
fout=open('C:af.txt', 'w')
for i in range (len(idaaff)):
    fout.write(str(xaff[i])+' '+str(yaaff[i])+' '+str(idaaff[i]))+'\n')

fout.close()
```

**Π4.1.1 Διεύρυνση μονόπλευρων €-μη-κυρτών τμημάτων**

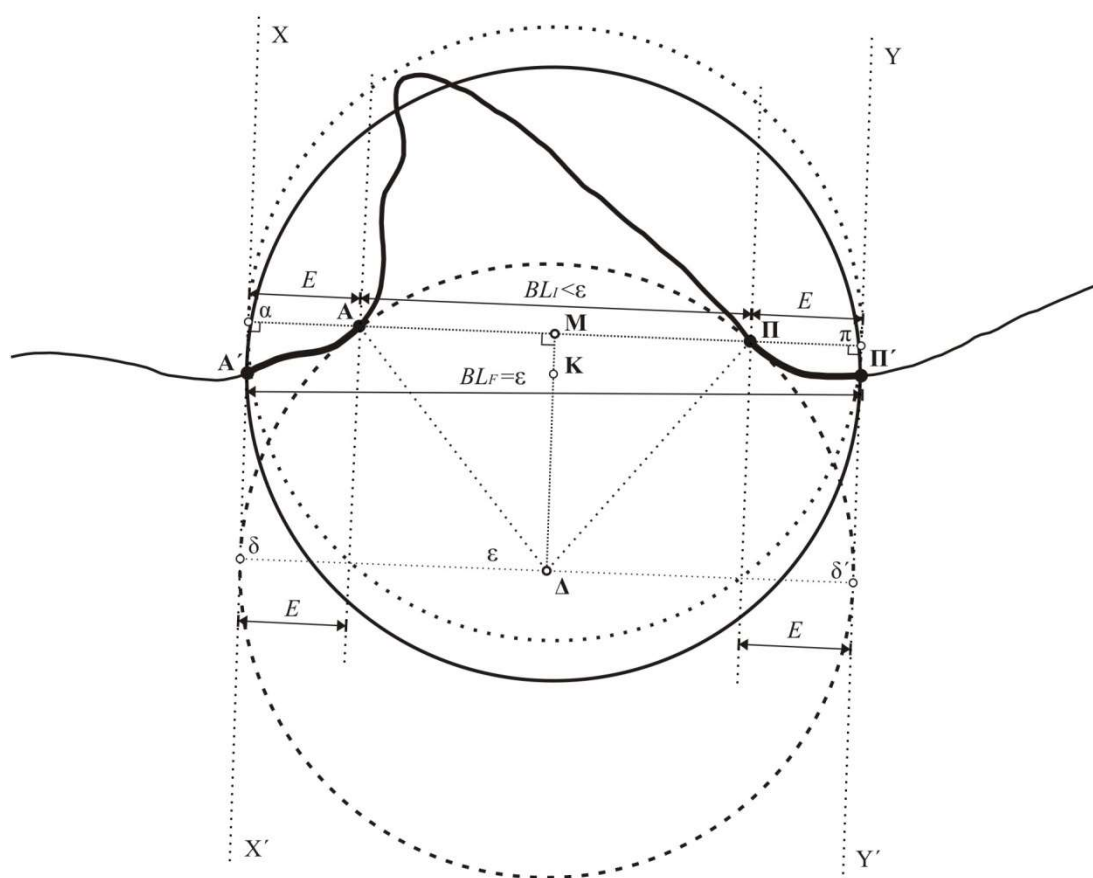
Η διεύρυνση των μονόπλευρων €-μη-κυρτών τμημάτων υλοποιείται στο περιβάλλον του λογισμικού πακέτου ArcGIS®, με τη βοήθεια του εργαλείου ModelBuilder. Στην ενότητα που ακολουθεί περιγράφονται τα βασικά του στάδια του μοντέλου εφαρμογής (Σχήμα Π1). Η περιγραφή σχετίζεται με το θεωρητικό πλαίσιο της μεθόδου διεύρυνσης, όπως αυτή αναλύεται στο κεφάλαιο 4.1.1. Για την καλύτερη κατανόηση της λειτουργίας του μοντέλου παρατίθεται το Σχήμα 4.1 του κεφαλαίου 4.1.1 με τις απαραίτητες τροποποιήσεις (Σχήμα Π2).



Σχήμα Π1: Μοντέλο διεύρυνσης μονόπλευρου €-μη-κυρτού τμήματος.

Το εκάστοτε μονόπλευρο €-μη-κυρτό τμήμα (*Isolated Part*) διασπάται (*Feature Vertices To Points*) στα σημεία που το δομούν, εντοπίζονται η αρχή και το πέρας του (Α και Π), τα οποία και διαχωρίζονται (*Select1*) σε ανεξάρτητες οντότητες (*EndPoint1* και *EndPoint2*). Στη συνέχεια, ανευρίσκονται ποσοτικά και γεωμετρικά χαρακτηριστικά του €-τμήματος, τα οποία χρησιμοποιούνται στις μετέπειτα διεργασίες διεύρυνσης. Πιο συγκεκριμένα, υπολογίζεται (*PointDistance*) η απόσταση μεταξύ των δύο άκρων Α και Π (*EndPointsDistance*), δηλαδή το μήκος  $BL_i$  της γραμμής βάσης του, το μήκος  $E = \frac{\epsilon - BL_i}{2}$  (*Calculate Field1, End Points Distance(2)*), όπως και το μήκος  $\frac{BL_i}{2}$  (*Calculate Field2, End Points Distance2(2)*). Παράλληλα, εντοπίζονται το μέσο Μ της γραμμής βάσης ΑΠ και το σημείο Δ, όπου βρίσκεται το κέντρο του δίσκου του Perkal όταν ‘δημιουργεί’ το €-

τμήμα (διακεκομμένος κύκλος στο Σχήμα Π2). Γύρω από το κάθε άκρο Α, Π του  $\epsilon$ -τμήματος δημιουργείται (*Buffer1*) ζώνη επιρροής (*EndPoint1 Buffer* και *EndPoint2 Buffer*), με εύρος ίσο με  $\frac{\epsilon}{2}$ . Οι δύο ζώνες επιρροής επικαλύπτονται (*Intersect1*) σχηματίζοντας χωρίο (*End Points Mid Area*), του οποίου το κέντρο αποτελεί το μέσο Μ της γραμμής βάσης (*Mid Point*) και εντοπίζεται (*Feature To Point1*). Συγχρόνως, οι δύο ζώνες επιρροής τέμνονται (*Intersect2*) σε δύο σημεία (*Central Axis Points*). Το ένα εξ αυτών, ανάλογα με την καμπυλότητα του  $\epsilon$ -τμήματος, αποτελεί το ζητούμενο κέντρο Δ του δίσκου του Perkal, ενώ το έτερο, το συμμετρικό του ως προς τη γραμμή βάσης. Τα δύο σημεία και το μέσο Μ είναι συνευθειακά και προσδιορίζουν τον άξονα κύλισης ΔΜ του δίσκου (βλ. κεφ. 4.1.1).



Σχήμα Π2: Διεύρυνση μονόπλευρου  $\epsilon$ -μη-κυρτού τμήματος.

Αφού υπολογισθούν τα ποσοτικά και γεωμετρικά χαρακτηριστικά του  $\epsilon$ -τμήματος, υλοποιείται διαδικασία, μέσω της οποίας προσομοιώνεται η κύλιση του δίσκου του Perkal κατά μήκος του άξονα ΔΜ. Ζητούμενο είναι ο εντοπισμός σημείων που ορίζουν τις ευθείες ΧΧ' και ΥΥ', από την τομή των οποίων με τη γραμμή εντοπίζονται τα σημεία Α' και Π'. Αρχικά, με κέντρο το μέσο Μ της γραμμής βάσης δημιουργείται ζώνη επιρροής (*Buffer2, Mid Point Buffer*) εύρους ίσου με  $\frac{\epsilon}{2}$ , η οποία αντιστοιχεί στο δίσκο του Perkal στη συγκεκριμένη θέση (εστιγμένος κύκλος στο Σχήμα Π2). Παράλληλα,



γύρω από την αρχή και το πέρας του  $\epsilon$ -τμήματος, φέρεται ζώνη επιρροής μεγέθους  $E$  (*Buffer3, End Points Distance Buffer*). Τα σημεία (*TangentPoint*) όπου οι τρεις ζώνες επιρροής τέμνονται (*Intersect3*) εντοπίζονται και αντιστοιχούν στις θέσεις που θα βρίσκονταν τα άκρα του αρχικού  $\epsilon$ -τμήματος, αν η γραμμή βάσης του επεκτείνονταν κατά μήκος  $E$  εξωτερικά του καθενός από αυτά, ώστε το μήκος της να είναι  $BL=\epsilon$  (α και π στο Σχήμα Π12). Στη συνέχεια, με κέντρο το σημείο  $\Delta$  δημιουργείται ζώνη επιρροής (*Buffer4, Central Axis Points Buffer*) εύρους ίσου με  $\frac{\epsilon}{2}$ , η οποία προσομοιώνει το δίσκο

του Perkal που 'δημιουργεί' το  $\epsilon$ -τμήμα. Μετά από μια διαδικασία εφαρμογής διαδοχικών ζωνών επιρροής και κατατμήσεων, εντοπίζονται πάνω σε αυτόν τα δύο και μοναδικά σημεία (*TangentPoint 2*), που απέχουν απόσταση  $\frac{\epsilon}{2}$  από το σημείο  $\Delta$  και

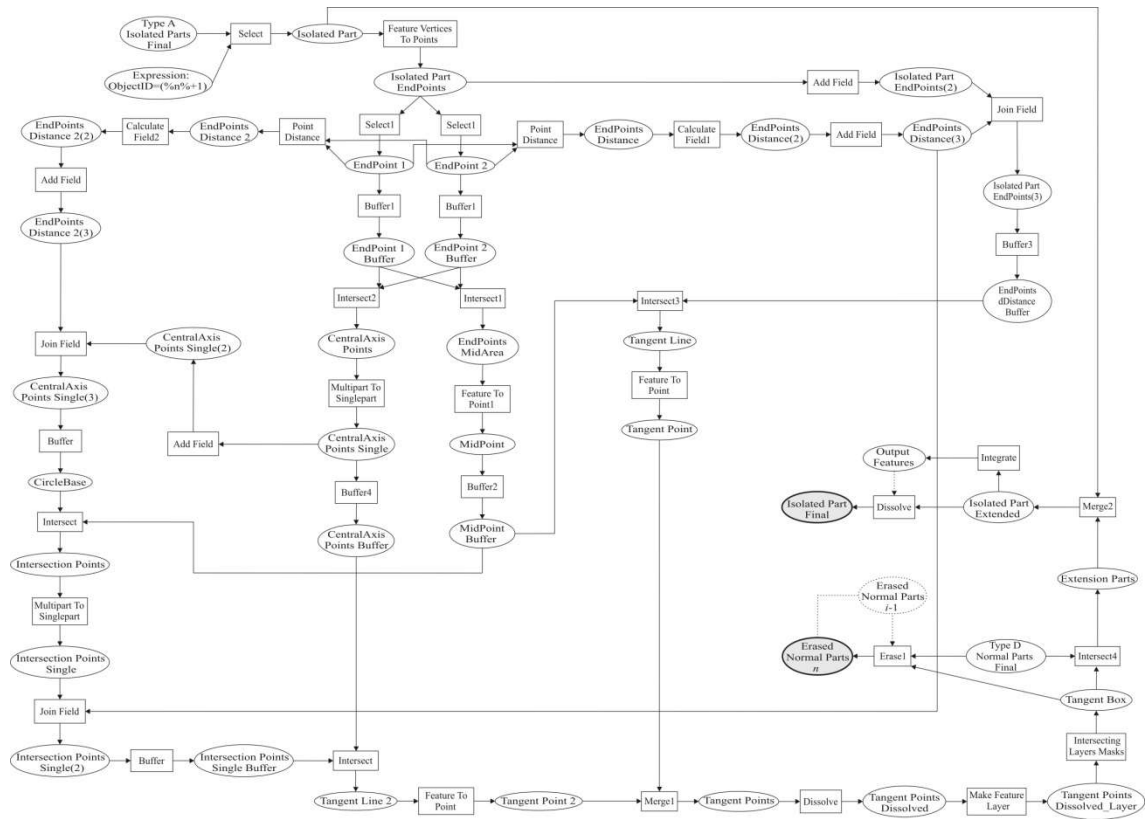
συγχρόνως, το κάθε ένα οριζόντια απόσταση  $E$  από το εγγύτερο άκρο του  $\epsilon$ -τμήματος (δ και δ' στο Σχήμα Π2).

Βάσει των ιδιοτήτων τους, τα σημεία  $\alpha$ ,  $\delta$  και  $\pi$ ,  $\delta'$  ορίζουν τις ευθείες  $XX'$  και  $YY'$ , αντίστοιχα. Τα τέσσερα σημεία συνενώνονται (*Merge1, TangentPoints*) και αποτελούν τα άκρα χωρίου (*TangentBox*) που δημιουργείται (*IntersectingLayerMasks*). Θεωρητικά, τα ζητούμενα σημεία  $A'$  και  $P'$  είναι τα σημεία τομής της γραμμής με το περίγραμμα του χωρίου. Στην πράξη, τα μέρη των γειτονικών  $\epsilon$ -τμημάτων που βρίσκονται εντός του χωρίου (*ExtensionParts*) αποκόπτονται (*Intersect4*) και συνενώνονται (*Merge2*) με το αρχικό μονόπλευρο  $\epsilon$ -μη-κυρτό τμήμα, διαμορφώνοντας τη διευρυμένη εκδοχή του (*IsolatedPartFinal*). Σύμφωνα με τις διεργασίες δημιουργίας τους, τα μονόπλευρα  $\epsilon$ -μη-κυρτά τμήματα γειτνιάζουν με  $\epsilon$ -κυρτά τμήματα, των οποίων το ανάπτυγμα είναι μεγαλύτερο από μια τιμή κατωφλίου που ορίζεται από το όριο διακριτικής ικανότητας ( $D$ ) και μια τιμή ανοχής ( $T$ ) στην κλίμακα του χάρτη (βλ. κεφ. 2.4.2). Τα μέρη των γειτονικών  $\epsilon$ -τμημάτων που αποκόπτονται κατά τη διαδικασία διεύρυνσης έχουν μήκος που προσεγγίζει την οριζόντια απόσταση  $E = \frac{\epsilon - BL'}{2}$ , μήκος μικρότερο του

ανάπτυγματος των  $\epsilon$ -κυρτών τμημάτων. Έτσι, αποκλείεται η περίπτωση αποκοπής μέρους  $\epsilon$ -τμημάτων πέραν των παρακείμενων  $\epsilon$ -κυρτών (*TypeDNormalPartsFinal*) και για το λόγο αυτό, μόνο αυτά εξετάζονται. Τα μέρη των  $\epsilon$ -κυρτών τμημάτων που βρίσκονται εκτός του χωρίου απομονώνονται (*Erase1*), δημιουργώντας τα τελικά  $\epsilon$ -κυρτά τμήματα (*ErasedNormalParts*). Αξίζει να σημειωθεί πως τα μονόπλευρα  $\epsilon$ -τμήματα είναι δυνατό να γειτνιάζουν και με τμήματα σύγκλισης. Το φαινόμενο συναντάται όταν κατά τη γενίκευση των τελευταίων, κάποιο ή κάποια εξ αυτών διευρύνονται, προσαρτώντας εξολοκλήρου το άμεσα γειτονικό τους  $\epsilon$ -τμήμα, καθώς και μέρος μονόπλευρου που έπεται (βλ. κεφ. 4.3.2). Λόγω της σημαντικότητας και των ιδιοτήτων που παρουσιάζει η επεξεργασία των τμημάτων σύγκλισης, επιλέγεται η διατήρηση του σχήματος τους. Σε μια τέτοια περίπτωση λοιπόν, η διεύρυνση των μονόπλευρων  $\epsilon$ -τμημάτων λαμβάνει χώρα μόνο από την πλευρά του έτερου γειτονικού  $\epsilon$ -κυρτού τμήματος. Για το λόγο αυτό, κατά την υλοποίησή της δεν πραγματοποιείται έλεγχος για τον εντοπισμό εντός του χωρίου (*TangentBox*) των τμημάτων σύγκλισης. Η μονομερής διεύρυνση των μονόπλευρων  $\epsilon$ -τμημάτων έχει ως αποτέλεσμα η γραμμή

βάσης τους να παραμένει μικρότερη της τιμής  $\epsilon$  με την ολοκλήρωση της διαδικασίας. Το πρόβλημα αποτελεί ζητούμενο μελλοντικής έρευνας, αφού στην πράξη, το φαινόμενο που περιγράφεται παρατηρείται σπάνια σε χαρτογραφικές γραμμές που αναπαριστούν φυσικές οντότητες.

Το μοντέλο που περιγράφεται στο Σχήμα Π1 αναφέρεται στη διαδικασία διεύρυνσης ενός μονόπλευρου  $\epsilon$ -μη-κυρτού τμήματος. Στην πλειοψηφία των εφαρμογών όμως, κατά μήκος των χαρτογραφικών γραμμών αποκόπτονται περισσότερα του ενός μονόπλευρα  $\epsilon$ -τμήματα. Για την αποφυγή εκτέλεσης του μοντέλου ξεχωριστά σε κάθε ένα από αυτά, έχει δημιουργηθεί μια τροποποιημένη εκδοχή του, κατά την οποία η διαδικασία υλοποιείται διαδοχικά, σε όλα τα μονόπλευρα  $\epsilon$ -τμήματα της γραμμής. Οι διεργασίες που ακολουθούνται είναι ίδιες με εκείνες που παρουσιάζονται στο Σχήμα Π1. Σε κάθε επανάληψη εκτέλεσης του μοντέλου ολοκληρώνεται η διεύρυνση του εκάστοτε  $\epsilon$ -τμήματος, το οποίο επιλέγεται (*Select*) βάσει του αύξοντα αριθμού του (*ObjectID=%n%+1*), από το σύνολο ( $n$ ) των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων (*Type A Isolated Parts Final*). Η μόνη διαφοροποίηση έγκειται στη δημιουργία των τελικών  $\epsilon$ -κυρτών τμημάτων (*Erased Normal Parts*). Κατά την πρώτη επανάληψη ( $i=1$ ), εξετάζονται τα αρχικά  $\epsilon$ -κυρτά τμήματα (*Type D Normal Parts Final*), όπως ακριβώς συμβαίνει στην περίπτωση της μεμονωμένης εκτέλεσης του μοντέλου. Στις επόμενες, εξετάζονται τα  $\epsilon$ -κυρτά τμήματα που κάθε φορά προκύπτουν από την αμέσως προηγούμενη επανάληψη (*Erased Normal Parts i-1*). Έτσι, από τα  $\epsilon$ -κυρτά τμήματα αφαιρούνται διαδοχικά, τα μέρη που συνενώνονται με το εκάστοτε μονόπλευρο  $\epsilon$ -τμήμα. Τα τελικά  $\epsilon$ -κυρτά τμήματα (*Erased Normal Parts n*) είναι εκείνα που έχουν διατηρηθεί μετά τη διεύρυνση όλων των μονόπλευρων  $\epsilon$ -μη-κυρτών τμημάτων. Το τροποποιημένο μοντέλο παρουσιάζεται στο Σχήμα Π3.



Σχήμα Π3: Μοντέλο διεύρυνσης μονόπλευρου ε-μη-κυρτού τμήματος με επαναλήψεις.

## Π4.1.2 Πρόγραμμα γενίκευσης μονόπλευρων €-μη-κυρτών τμημάτων

### Προεπεξεργασία

```
import arcgisscripting
gp = arcgisscripting.create(9.3)

# ===== EisagwgiDedomenwn =====
# ----- EisagwgiArxikisGrammis -----
f = open('C:LinePart_Points.txt')
# Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in f:
    NumLines=NumLines+1

f = open('C:LinePart_Points.txt')
# Anoigmaarxeiou

# Eisagwgisimeiwngammis (inix, iniy, iniid)
inix=[]
iniy=[]
iniid=[]
mtr=0
for line in f:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        iniid.append(int(line[:-1].split(' ')[0]))
        inix.append(float(line[:-1].split(' ')[1]))
        iniy.append(float(line[:-1].split(' ')[2]))

# ----- Ypologismosapodektwntimwn 'sigma' -----
sigmaAccept=[]
sig=0
for i in range((len(iniid)-1)/2):
    sig=sig+0.25
    sigmaAccept.append(sig)

sigmaMax=float(len(iniid)-1)/8
sigmaAccept.append(sigmaMax)

# ----- Eisagwgi 'sigma' -----
m = open('C:Sigma.txt')
sigma=[]
for line in m:
    sigma.append(float(line[:]))

# -----
import math

# =====
# ===== ExomalinsiArxikisGrammis =====
# =====

# YpologismosX,Y Gauss AkraiwnSimeiwngArxisGrammis (arximexri +4sigma)
# Dimiourgiakatoptrikwnsimeiwng apo to prwtosimeio
mirrorAx=[]
mirrorAy=[]
for i in range (0, int(4*sigma[0])):
    mAx=2*inix[0]-inix[i+1]
```

```
mAy=2*inix[0]-inix[i+1]
mirrorAx.append(mAx)
mirrorAy.append(mAy)

mirrorAx.reverse()
# Antistrofipinaka - diorthwsiseirassimeiw
mirrorAy.reverse()

# Enwsikatoptrikwnsimeiw me arxikigrammi
arxix=mirrorAx+inix
arxiy=mirrorAy+inix

# Ypologismos x(i-k)*g(k) giakathesimeio
xika=[]
for i in range(int(4*sigma[0]), int(8*sigma[0])):
    for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
        xikae=arxix[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        xika.append(xikae)

# YpologismosXGauss
xGa=[]
xGaassist=[]
for j in range(1,(len(xika)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0]+1)):
        xGaassist.append(xika[i])
    sum(xGaassist)
    xGa.append(sum(xGaassist))
    xika[0:int(8*sigma[0]+1)]=[]
    xGaassist[:]=[]

# Ypologismos y(i-k)*g(k) giakathesimeio
yika=[]
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        yikae=arxiy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yika.append(yikae)

# YpologismosYGauss
yGa=[]
yGaassist=[]
for j in range(1,(len(yika)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0]+1)):
        yGaassist.append(yika[i])
    sum(yGaassist)
    yGa.append(sum(yGaassist))
    yika[0:int(8*sigma[0]+1)]=[]
    yGaassist[:]=[]

# ===== YpologismosX,Y Gauss MeswnSimeiwGrammaris (+4sigma+1 apo tin
arximexri -4sigma+1 apo to peras) =====
# Ypologismos x(i-k)*g(k) giakathesimeio
xikm=[]
for i in range (int(4*sigma[0]), len(iniid)-int(4*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
```

```
        xikme=inix[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2))))))
        xikm.append(xikme)

# YpologismosXGauss
xGm=[]
xGmassist=[]
for j in range(1,(len(xikm)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0]+1)):
        xGmassist.append(xikm[i])
    sum(xGmassist)
# Athroismatwntimwn -->XGausstousimeiou
xGm.append(sum(xGmassist))
xikm[0:int(8*sigma[0]+1)]=[]
xGmassist[:]=[]

# Ypologismos y(i-k)*g(k) giakathesimeio
yikm=[]
for i in range (int(4*sigma[0]), len(iniid)-int(4*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0]+1)):
        yikme=iniy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2))))))
        yikm.append(yikme)

# YpologismosYGauss
yGm=[]
yGmassist=[]
for j in range(1,(len(yikm)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0]+1)):
        yGmassist.append(yikm[i])
    sum(yGmassist)
yGm.append(sum(yGmassist))
yikm[0:int(8*sigma[0]+1)]=[]
yGmassist[:]=[]

# YpologismosX,Y Gauss akraiwnsimeiwntoperatosgrammis (-4sigma mexriperas)
# Dimiourgiakatoptrikwnsimeiwnt apo to teleutaiosimeio
mirrorPx=[]
mirrorPy=[]

# Dimiourgiaantistrofwnpinakwn x kai y
xrev=[]
yrev=[]
for i in range (len(iniid)):
    xrev.append(inix[i])
    yrev.append(iniy[i])

xrev.reverse()
yrev.reverse()

for i in range (0, int(4*sigma[0])):
    mPx=2*xrev[0]-xrev[i+1]
    mPy=2*yrev[0]-yrev[i+1]
    mirrorPx.append(mPx)
    mirrorPy.append(mPy)
```

```
# Enwsikatoptrikwnsimeiw n me arxikigrammi (orthiseirasimeiw n)
perasx=inix+mirrorPx
perasy=iniy+mirrorPy

# Antisxtrofipinakwnperasx kai perasy (epexergasia se antistrofouspinakes)
perasx.reverse()
perasy.reverse()

# Ypologismos x(i-k)*g(k) giakathesimeio
xikp=[]
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        xikpe=perasx[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        xikp.append(xikpe)

# YpologismosXGauss
xGp=[]
xGpassist=[]
for j in range(1,(len(xikp)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0])+1):
        xGpassist.append(xikp[i])
    sum(xGpassist)
    xGp.append(sum(xGpassist))
    xikp[0:int(8*sigma[0]+1)]=[]
    xGpassist[:]=[]

# Ypologismos y(i-k)*g(k) giakathesimeio
yikp=[]
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        yikpe=perasy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yikp.append(yikpe)

# YpologismosYGauss
yGp=[]
yGpassist=[]
for j in range(1,(len(yikp)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0])+1):
        yGpassist.append(yikp[i])
    sum(yGpassist)
    yGp.append(sum(yGpassist))
    yikp[0:int(8*sigma[0]+1)]=[]
    yGpassist[:]=[]

# AntistrofipinakwnxGp kai yGp - diorthwsiseirassimeiw n
xGp.reverse()
yGp.reverse()

# EnwsipinakwnxGa-xGm-xGp / yGa-yGm-yGp
xu=xGa+xGm+xGp
yu=yGa+yGm+yGp
id=iniid

# ===== X kai Y Gauss =====
```

```

x=[]
y=[]

# Diorthwsix,y apo sfalmastroggylopoihsisvarous
# Ypologismosathroismatosepimerousvarwn
gsk=0
for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
    gsk=gsk+(1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2))))

# Diorthwsisyntetagmenwn
for i in range(len(id)):
    xx=xu[i]/gsk
    yy=yu[i]/gsk
    x.append(xx)
    y.append(yy)

# =====
# ===== Arithmos Bends stiGrammi Gauss =====
# =====

# ----- Entopismos Bends -----
# Ypologismosapostasewnmetaxitwnsimeiwntis grammis
distx=[]
disty=[]
distid=[]
for i in range(len(id)-1):
    dx=x[i+1]-x[i]
    dy=y[i+1]-y[i]
    distx.append(dx)
    disty.append(dy)
    distid.append(i)

# Ypologismosgwniasdiefthinsiseftthigrammwntmimatwngrammis
w=[]
for i in range (len(distid)):
    if distx[i]>0 and disty[i]>0:
        ww=
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
        elifdistx[i]>0 and disty[i]<0:
            ww= 200-
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
            elifdistx[i]<0 and disty[i]<0:
                ww=
200+((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
                elifdistx[i]<0 and disty[i]>0:
                    ww= 400-
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
                    elifdistx[i]==0 and disty[i]>0:
                        ww=0
                    elifdistx[i]==0 and disty[i]<0:
                        ww=200
                    elifdistx[i]>0 and disty[i]==0:
                        ww=100
                    elifdistx[i]<0 and disty[i]==0:
                        ww=300
    w.append(ww)

```



```
# Elegxosgiapithaniafairesi h' prosthesi 400g
wf=[]
for i in range(len(w)):
    if w[i]>400:
        wwf=w[i]-400
    elif w[i]<0:
        wwf=w[i]+400
    else:
        wwf=w[i]
    wf.append(wwf)

# Metatropi grad se rad
wfr=[]
for i in range(len(wf)):
    wwfr=(math.pi*wf[i])/200
    wfr.append(wwfr)

# Ypologismosgwniasthlasissimeiwn 2 ews n-1
slo=[]
for i in range(len(wfr)-1):
    slop=wfr[i+1]-wfr[i]
    slo.append(slop)

# Diorthwsigwniwnthlasis (megaliterwntou pi) / Telikesgwniesthlasis
sl=[]
for i in range(len(slo)):
    if (math.pi)<slo[i]<(math.pi*2):
        ssl=slo[i]-(math.pi*2)
    elif -(math.pi*2)<slo[i]<-(math.pi):
        ssl=slo[i]+(math.pi*2)
    else:
        ssl=slo[i]
    sl.append(ssl)

# Entopismos Inflection Points
#(entopizontai diadoxikasimeia me antithetiklisi/katagrafontai kai ta 2**)
infpid=[]
for i in range(len(sl)-1):
    if sl[i]<0 and sl[i+1]>0 or sl[i]>0 and sl[i+1]<0:
        infpid.append(id[i+1])
        infpid.append(id[i+2])

# ----- YpologismosArithmou Bends -----
if len(infpid)>0:
    NumOfBends=(len(infpid)/2)+1 # (NumOfBends=(arithmosIP/2 (logw**))+1
elif len(infpid)==0:
    NumOfBends=1

# ===== ExagwgiArxeiwn =====
# ----- Apodektes Times 'sigma' -----
gout=open('C:SigmaAccept.txt', 'w')
for i in range(len(sigmaAccept)-1): # (stoixeiaektostousigmaMax)
    gout.write(str(sigmaAccept[i])+' '+'\n')

gout.write('sigmaMax:'+str(sigmaMax))
gout.close()

# ----- Arithmos Bends -----
```

```
fout=open('C:NumOfBends.txt', 'w')
fout.write('NumOfBends:'+ ' '+str(NumOfBends)+'\n')
fout.close()

# ----- Grammi Gauss -----
mout=open('C:SmoothPart.txt', 'w')
mout.write(str('Polyline')+'\n')
for i in range(len(id)):
    mout.write(str(id[i])+ ' '+str(x[i])+ ' '+str(y[i]))+'\n')

mout.write(str('END'))
mout.close()

m.close()
f.close()
```

### Διόγκωση με δεσμεύσεις

```
import arcgisscripting
gp = arcgisscripting.create(9.3)

# ===== EisagwgiDedomenwn =====
# ----- EisagwgiGrammis Gauss -----
f = open('C:SmoothPart.txt')
# Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in f:
    NumLines=NumLines+1

f = open('C:SmoothPart.txt')
# Anoigmaarxeiou

# Eisagogisimeiwngrammis (x,y,id)
x=[]
y=[]
id=[]
mtr=0
for line in f:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        id.append(int(line[:-1].split(' ')[0]))
        x.append(float(line[:-1].split(' ')[1]))
        y.append(float(line[:-1].split(' ')[2]))

# ----- EisagwgiArxikisGrammis -----
g = open('C:LinePart_Points.txt')
# Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in g:
    NumLines=NumLines+1

g = open('C:LinePart_Points.txt') # Anoigmaarxeiou

# Eisagogisimeiwngrammis (inix, iniy, iniid)
inix=[]
iniy=[]
iniid=[]
mtr=0
```

```

for line in g:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        iniid.append(int(line[:-1].split(' ')[0]))
        inix.append(float(line[:-1].split(' ')[1]))
        iniy.append(float(line[:-1].split(' ')[2]))

# ----- Eisagwgi e -----
m = open('C:e.txt')
e=[]
for line in m:
    e.append(float(line[:]))

import math

# =====
# ===== EntopismosKoryfisGrammis Gauss mesw ALR =====
# ===== EntopismosAntistoixisKoryfisstinArxikiGrammi =====
# =====

# Ypologismosapostasewnmetaxisimeiw
Sbp=[]
for i in range(len(id)-1):
    Sbps=math.sqrt(((x[i+1]-x[i])**2)+((y[i+1]-y[i])**2))
    Sbp.append(Sbps)

# Ypologismos LR1, LR2, LR3, LR4, ALR, maxALR kai entopismosKoryfis
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

if len(id)==3: # ----- Efarmogi se bend me 3 simeia -----
    CPnum=id[0]
elif len(id)==4: # ----- Efarmogi se bend me 4 simeia -----
    for i in range(len(id)): # Ypologismos LR1
        if i in range(0,1):
            llr1=1
        elif i in range(1,len(id)-1):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            llr1=l1/s1
        elif i in range(len(id)-1,len(id)):
            llr1=1
        LR1.append(llr1)
        alr=LR1[i] # Ypologismos ALR(==LR1)
        ALR.append(alr)
        if ALR[i]==max(ALR): # Entopismos ID
            simeioupuantistoixeistomegisto ALR
            CPnum=id[i-1]
elif len(id)==5 or len(id)==6: # ----- Efarmogi se bend 5 'h 6 simeia -----
    for i in range(len(id)): # Ypologismos LR1, LR2
        if i in range(0,2):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            llr2=l2/s2

```

```

elif i in range(2, len(id)-2):
    l1=Sbp[i-1]+Sbp[i]
    s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
    llr1=l1/s1
    l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
    s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
    llr2=l2/s2
elif i in range(len(id)-2, len(id)):
    llr1=1
    l2=Sbp[i-1]+Sbp[i-2]
    s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
    llr2=l2/s2
LR1.append(llr1)
LR2.append(llr2)
alr=(LR1[i]+LR2[i])/2 # Υπολογισμος ALR
ALR.append(alr)
if ALR[i]==max(ALR): # Entopismos ID
simeioupuantistoixeistomegisto ALR
    CPnum=id[i-1]
elif len(id)==7 or len(id)==8: # ----- Efarmogi se bend me 7 h' 8 simeia-----
    for i in range(len(id)): # Υπολογισμος LR1, LR2, LR3
        if i in range(0,3):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i])**2)+((y[i+3]-y[i])**2))
            llr3=l3/s3
        elif i in range(3, len(id)-3):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i-3])**2)+((y[i+3]-y[i-3])**2))
            llr3=l3/s3
        elif i in range(len(id)-3, len(id)):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((x[i-3]-x[i])**2)+((y[i-3]-y[i])**2))
            llr3=l3/s3
        LR1.append(llr1)
        LR2.append(llr2)
        LR3.append(llr3)
        alr=(LR1[i]+LR2[i]+LR3[i])/3 # Υπολογισμος ALR
        ALR.append(alr)
        if ALR[i]==max(ALR): # Entopismos ID
simeioupuantistoixeistomegisto ALR
            CPnum=id[i-1]
elif len(id)>=9:
    for i in range(len(id)): # Efarmogi se bend me toulaxiston 9 simeia
        if i in range(0,4): # Υπολογισμος LR1, LR2, LR3, LR4

```

```

        l1r1=1
        l2=Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
        l1r2=l2/s2
        l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
        s3=math.sqrt(((x[i+3]-x[i])**2)+((y[i+3]-y[i])**2))
        l1r3=l3/s3
        l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
        s4=math.sqrt(((x[i+4]-x[i])**2)+((y[i+4]-y[i])**2))
        l1r4=l4/s4
    elifi in range(4,len(id)-4):
        l1=Sbp[i-1]+Sbp[i]
        s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
        l1r1=l1/s1
        l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
        l1r2=l2/s2
        l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
        s3=math.sqrt(((x[i+3]-x[i-3])**2)+((y[i+3]-y[i-3])**2))
        l1r3=l3/s3
        l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
        s4=math.sqrt(((x[i+4]-x[i-4])**2)+((y[i+4]-y[i-4])**2))
        l1r4=l4/s4
    elifi in range(len(id)-4,len(id)):
        l1r1=1
        l2=Sbp[i-1]+Sbp[i-2]
        s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
        l1r2=l2/s2
        l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
        s3=math.sqrt(((x[i-3]-x[i])**2)+((y[i-3]-y[i])**2))
        l1r3=l3/s3
        l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
        s4=math.sqrt(((x[i-4]-x[i])**2)+((y[i-4]-y[i])**2))
        l1r4=l4/s4
    LR1.append(l1r1)
    LR2.append(l1r2)
    LR3.append(l1r3)
    LR4.append(l1r4)
    alr=(LR1[i]+LR2[i]+LR3[i]+LR4[i])/4 # Υπολογισμος ALR
    ALR.append(alr)
    if ALR[i]==max(ALR): # Entopismos ID
simeiou pou antistoixei stomegisto ALR
        CPnum=id[i-1]

# Syntetagmenes kai id Koryfisgrammis Gauss
korx=x[CPnum]
kory=y[CPnum]
korid=id[CPnum]

# -----
# Dimiourgia arxeiou me Inflection Points kai Koryfigrammis Gauss
xp=[]
yp=[]
idp=[]
xp.append(x[0])
xp.append(korx)
xp.append(x[len(id)-1])

```

```
yp.append(y[0])
yp.append(kory)
yp.append(y[len(id)-1])
idp.append(1)
idp.append(2)
idp.append(3)

# Dimiourgiaarxeiou me Inflection Points kai KoryfiArxikisgrammis
xpa=[]
ypa=[]
xpa.append(inix[0])
xpa.append(inix[CPnum]) # Entopismoskoryfis
xpa.append(inix[len(id)-1])
ypa.append(iniy[0])
ypa.append(iniy[CPnum])
ypa.append(iniy[len(id)-1])
# -----

# =====
# ===== DiogwsiGrammis =====
# =====

# ----- Ypologismos syntelestwn u kai v -----
# Ypologismos dx kai dy gia ta simeia 2 to n-1
dx=[]
dy=[]
for i in range(1, len(id)-1):
    ddx=x[i+1]-x[i-1]
    ddy=y[i+1]-y[i-1]
    dx.append(ddx)
    dy.append(ddy)

# Ypologismos gnias diefthinsiseftheias n-1---n+1 giakathesimeio
w=[]
for i in range(len(dx)):
    if dx[i]>0 and dy[i]>0:
        ww=(math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
    elif dx[i]>0 and dy[i]<0:
        ww= 200-
        ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
    elif dx[i]<0 and dy[i]<0:
        ww=
        200+((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
    elif dx[i]<0 and dy[i]>0:
        ww= 400-
        ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
    elif dx[i]==0 and dy[i]>0:
        ww=0
    elif dx[i]==0 and dy[i]<0:
        ww=200
    elif dx[i]>0 and dy[i]==0:
        ww=100
    elif dx[i]<0 and dy[i]==0:
        ww=300
    w.append(ww)

# Ypologismos diefthinsismetatopisisshmeiwn (kathetistin w)
```

```
wm=[]
for i in range(len(w)):
    wwm=w[i]-100
    wm.append(wwm)

# Elegxosgiapithaniafairesi h' prothesi 400g
wmf=[]
for i in range(len(wm)):
    if wm[i]>400:
        wwmf=wm[i]-400
    elifwm[i]<0:
        wwmf=wm[i]+400
    else:
        wwmf=wm[i]
    wmf.append(wwmf)

# Metatropi grad se rad
wmfr=[]
for i in range(len(wmf)):
    wwmfr=(math.pi*wmf[i])/200
    wmfr.append(wwmfr)

# Ypologismossyntelestwn u kai v
u=[]
for i in range(len(wmfr)):
    uu=math.fabs(math.sin(wmfr[i]))
    u.append(uu)

v=[]
for i in range(len(wmfr)):
    vv=math.fabs(math.cos(wmfr[i]))
    v.append(vv)

# ----- Ypologismos sense -----
sensex=[]
sensey=[]
for i in range(len(wmfr)):
    if wmfr[i]>=0 and wmfr[i]<(math.pi)/2:
        sensx=1
        sensey=1
    elifwmfr[i]>=(math.pi)/2 and wmfr[i]<math.pi:
        sensx=1
        sensey=-1
    elifwmfr[i]>=math.pi and wmfr[i]<(3*math.pi)/2:
        sensx=-1
        sensey=-1
    elifwmfr[i]>=(3*math.pi)/2 and wmfr[i]<2*math.pi:
        sensx=-1
        sensey=1
    sensex.append(sensx)
    sensey.append(sensy)

# ----- Ypologismos S/Sn -----
# Ypologismos S simeiw n 1 wsKoryfi
# Ypologismosapostasewnmetaxisimeiw n
Sbp1K=[]
for i in range (0, korid-1):
    Sbps1K=math.sqrt(((x[i+1]-x[i])**2)+((y[i+1]-y[i])**2))
```

```
Sbp1K.append(Sbps1K)

# Ypologismos S kathesimeiou
S1K=[]
for i in range(len(Sbp1K)):
    if i==0:
        sslk=0+Sbp1K[i]
    else:
        sslk=sslk+Sbp1K[i]
    S1K.append(sslk)

# Ypologismos Sn apo 1 wsKoryfi
Sn1K=sum(Sbp1K, 0)

# Ypologismos S simeiw n wsKoryfi
# Ypologismosapostasewnmetaxisimeiw n
SbpNK=[]
for i in reversed(range(korid-1, len(id)-1)):
    SbpsNK=math.sqrt(((x[i]-x[i+1])**2)+((y[i]-y[i+1])**2))
    SbpNK.append(SbpsNK)

# Ypologismos S kathesimeiou
SNK=[]
for i in range(len(SbpNK)):
    if i==0:
        sskn=0+SbpNK[i]
    else:
        sskn=sskn+SbpNK[i]
    SNK.append(sskn)

# Ypologismos Sn apo n wsKoryfi
SnNK=sum(SbpNK, 0)

# Ypologismos S/Sn (S1K/Sn1K) apo 2 wsKoryfi
Sf1K=[]
for i in range(len(S1K)):
    sf1k=S1K[i]/Sn1K
    Sf1K.append(sf1k)

# Ypologismos S/Sn (SNK/SnNK) apo n-1 wsKoryfi
SfNK=[]
for i in range(len(SNK)):
    sfnk=SNK[i]/SnNK
    SfNK.append(sfnk)

# ----- Ypologismos Inflating -----
# Ypologismoskentroukyklou C (metaxiarxis-peratos)
xc=(x[0]+x[len(id)-1])/2
yc=(y[0]+y[len(id)-1])/2

# Ypologismosapostasiskentroukyklou C kai CP grammis Gauss
CCP=math.sqrt(((xc-x[korid-1])**2)+((yc-y[korid-1])**2))

# Ypologismos Inflating analoga an CP ektos, panw h' ektoskyklou
if CCP>e[0]/2:
    Inflating=math.sqrt(((x[korid-1]-xpa[1])**2)+((y[korid-1]-ypa[1])**2))
# Ypologismos Inflating (apostasi CP apo CParxiko)
elif CCP==e[0]/2:
```



```

    Inflating=math.sqrt(((x[korid-1]-xpa[1])**2)+((y[korid-1]-ypa[1])**2))
# Ypologismos Inflating (apostasi CP apo CParxiko)
elif CCP<e[0]/2:
    l=-((x[korid]-x[korid-2])/(y[korid]-y[korid-2])) #
Ypologismos syntelestidiefthisis (l) eftheiasmetatopisiskoryfis
    D=((-2*xc)-(2*(1**2)*(x[korid-1]))+(2*1*y[korid-1])-(2*yc*1)**2)-
(4*((1**2)+1)*((xc**2)+(yc**2)-((e[0]**2)/4)+((1**2)*(x[korid-1]**2))-
(2*1*x[korid-1]*y[korid-1])+(y[korid-1]**2)+(2*yc*1*x[korid-1])-
(2*yc*y[korid-1]))) # Ypologismos Diakrinousas D
    xt1=(((2*xc)+(2*(1**2)*x[korid-1])-(2*1*y[korid-
1])+(2*yc*1))+(math.sqrt(D)))/((2*(1**2))+2) # Ypologismos pithanwn lisewn
t1,t2
    yt1=(1*xt1)-(1*x[korid-1])+y[korid-1]
    xt2=(((2*xc)+(2*(1**2)*x[korid-1])-(2*1*y[korid-1])+(2*yc*1))-
(math.sqrt(D)))/((2*(1**2))+2)
    yt2=(1*xt2)-(1*x[korid-1])+y[korid-1]
    ap1=math.sqrt(((x[korid-1]-xt1)**2)+((y[korid-1]-yt1)**2)) #
Ypologismos apostasewn CP-t1 (ap1) kai CP-t2 (ap2)
    ap2=math.sqrt(((x[korid-1]-xt2)**2)+((y[korid-1]-yt2)**2))
    if ap1>ap2: # Entopismos apodektis lysis systimatos (to
xtpouapexeimikroteri apostasi apo to CP)
        xtomi=xt2
        ytomi=yt2
    elif ap1<ap2:
        xtomi=xt1
        ytomi=yt1
    Inflating=math.sqrt(((x[korid-1]-xtomi)**2)+((y[korid-1]-ytomi)**2)) #
Ypologismos Inflating (apostasi CP apo tomi)

# ----- Ypologismos Xnew, Ynew Diogwmenis Grammis -----
# Ypologismos Xnew apo 1 ws Koryfi
X1K=[]
for i in range (0, korid):
    if i==0:
        x1k=x[i]
    else:
        x1k=x[i]+(Inflating*sensex[i-1]*u[i-1]*Sf1K[i-1])
    X1K.append(x1k)

# Ypologismos Ynew apo 1 ws Koryfi
Y1K=[]
for i in range (0, korid):
    if i==0:
        y1k=y[i]
    else:
        y1k=y[i]+(Inflating*sensey[i-1]*v[i-1]*Sf1K[i-1])
    Y1K.append(y1k)

# Dimiourgiav oithitikwn pinakwn pou perilambanoun mono ta simeia N ws K
xNKassist=[]
for i in reversed(range(korid-1, len(id))):
    xNKassist.append(x[i])

yNKassist=[]
for i in reversed(range(korid-1, len(id))):
    yNKassist.append(y[i])

```

```
# Antistrofiarxeiwnsensex, sensey, u kai v
gianaxrisimopoihthounstonypologismoX,Ytwn N ws K
sensex.reverse()
sensey.reverse()
u.reverse()
v.reverse()

# YpologismosXnew apo N wsKoryfi
XNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        xnk=xNKassist[i]
    else:
        xnk=xNKassist[i]+(Inflating*sensex[i-1]*u[i-1]*SfNK[i-1])
    XNKnew.append(xnk)

# Antistrofi XNK se XKN
XNKnew.reverse()
# Diagرافي lou stoixeiou
XNKnew.remove(XNKnew[0])

# YpologismosYnew apo N wsKoryfi
YNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        ynk=yNKassist[i]
    else:
        ynk=yNKassist[i]+(Inflating*sensey[i-1]*v[i-1]*SfNK[i-1])
    YNKnew.append(ynk)

# Antistrofi YNK se YKN
YNKnew.reverse()
# Diagرافي lou stoixeiou (dil. tou Y tou K pouyparxei kai sto YlK)
YNKnew.remove(YNKnew[0])

# Enwsipinakwn ---- X kai Y DiogwmenisGrammis ----
Xbal=[]
Xbal=XlK+XNKnew

Ybal=[]
Ybal=YlK+YNKnew

# ----- Elegxos an exeigineidiogwsi h' siriknwsigrammis -----
# TelikaX,Ygrammis Balloon meta ton elegxo
Xfinal=[]
Yfinal=[]

# Ypologismosapostasiskentroukyklou C kai CP diogwmenisgrammis
CCPbal=math.sqrt(((xc-Xbal[korid-1])**2)+((yc-Ybal[korid-1])**2))

# An CCP<CCPbalexeigineidiogwsi
# An CCP>CCPbalexeigineisiriknwsii --
>xrisimopoietaiantithetidiefthinsimetatopisis kai xanatrexei to programma
if CCP<CCPbal:
    Xfinal=Xbal
    Yfinal=Ybal
elif CCP>CCPbal:
    wm=[]
```

```
for i in range(len(w)):
    wwm=w[i]+100
    wm.append(wwm)
wmf=[]
for i in range(len(wm)):
    if wm[i]>400:
        wwmf=wm[i]-400
    elifwm[i]<0:
        wwmf=wm[i]+400
    else:
        wwmf=wm[i]
    wmf.append(wwmf)
wmfr=[]
for i in range(len(wmf)):
    wwmfr=(math.pi*wmf[i])/200
    wmfr.append(wwmfr)
u=[]
for i in range(len(wmfr)):
    uu=math.fabs(math.sin(wmfr[i]))
    u.append(uu)
v=[]
for i in range(len(wmfr)):
    vv=math.fabs(math.cos(wmfr[i]))
    v.append(vv)
sensex=[]
sensey=[]
for i in range(len(wmfr)):
    if wmfr[i]>=0 and wmfr[i]<(math.pi)/2:
        sensx=1
        sensy=1
    elifwmfr[i]>=(math.pi)/2 and wmfr[i]<math.pi:
        sensx=1
        sensy=-1
    elifwmfr[i]>=math.pi and wmfr[i]<(3*math.pi)/2:
        sensx=-1
        sensy=-1
    elifwmfr[i]>=(3*math.pi)/2 and wmfr[i]<2*math.pi:
        sensx=-1
        sensy=1
    sensex.append(sensx)
    sensey.append(sensy)
Sbp1K=[]
for i in range (0, korid-1):
    Sbps1K=math.sqrt(((x[i+1]-x[i])**2)+(y[i+1]-y[i])**2))
    Sbp1K.append(Sbps1K)
S1K=[]
for i in range(len(Sbp1K)):
    if i==0:
        ss1k=0+Sbp1K[i]
    else:
        ss1k=ss1k+Sbp1K[i]
    S1K.append(ss1k)
Sn1K=sum(Sbp1K, 0)
SbpNK=[]
for i in reversed(range(korid-1, len(id)-1)):
    SbpsNK=math.sqrt(((x[i]-x[i+1])**2)+(y[i]-y[i+1])**2))
    SbpNK.append(SbpsNK)
SNK=[]
```

```

for i in range(len(SbpNK)):
    if i==0:
        sskn=0+SbpNK[i]
    else:
        sskn=sskn+SbpNK[i]
    SNK.append(sskn)
SnNK=sum(SbpNK, 0)
Sf1K=[]
for i in range(len(S1K)):
    sf1k=S1K[i]/Sn1K
    Sf1K.append(sf1k)
SfNK=[]
for i in range(len(SNK)):
    sfnk=SNK[i]/SnNK
    SfNK.append(sfnk)
xc=(x[0]+x[len(id)-1])/2
yc=(y[0]+y[len(id)-1])/2
CCP=math.sqrt(((xc-x[korid-1])**2)+((yc-y[korid-1])**2))
if CCP>e[0]/2:
    Inflating=math.sqrt(((x[korid-1]-xpa[1])**2)+((y[korid-1]-
ypa[1])**2))
elif CCP==e[0]/2:
    Inflating=math.sqrt(((x[korid-1]-xpa[1])**2)+((y[korid-1]-
ypa[1])**2))
elif CCP<e[0]/2:
    l=-((x[korid]-x[korid-2])/(y[korid]-y[korid-2]))
    D=(((-2*xc)-(2*(1**2)*(x[korid-1]))+(2*1*y[korid-1])-(
2*yc*1)**2)-(4*((1**2)+1)*((xc**2)+(yc**2)-(e[0]**2)/4)+((1**2)*(x[korid-
1]**2))-(2*1*x[korid-1]*y[korid-1])+(y[korid-1]**2)+(2*yc*1*x[korid-1])-(
2*yc*y[korid-1])))
    xt1=((2*xc)+(2*(1**2)*x[korid-1])-(2*1*y[korid-
1])+(2*yc*1)+(math.sqrt(D)))/((2*(1**2))+2)
    yt1=(1*xt1)-(1*x[korid-1])+y[korid-1]
    xt2=((2*xc)+(2*(1**2)*x[korid-1])-(2*1*y[korid-1])+(2*yc*1))-
(math.sqrt(D))/((2*(1**2))+2)
    yt2=(1*xt2)-(1*x[korid-1])+y[korid-1]
    ap1=math.sqrt(((x[korid-1]-xt1)**2)+((y[korid-1]-yt1)**2))
    ap2=math.sqrt(((x[korid-1]-xt2)**2)+((y[korid-1]-yt2)**2))
    if ap1>ap2:
        xtomi=xt2
        ytomi=yt2
    elif ap1<ap2:
        xtomi=xt1
        ytomi=yt1
    Inflating=math.sqrt(((x[korid-1]-xtomi)**2)+((y[korid-1]-
ytomi)**2))
X1K=[]
for i in range(0, korid):
    if i==0:
        x1k=x[i]
    else:
        x1k=x[i]+(Inflating*sensex[i-1]*u[i-1]*Sf1K[i-1])
    X1K.append(x1k)
Y1K=[]
for i in range(0, korid):
    if i==0:
        y1k=y[i]
    else:

```

```

        ylk=y[i]+(Inflating*sensey[i-1]*v[i-1]*Sf1K[i-1])
    Y1K.append(ylk)
    xNKassist=[]
    for i in reversed(range(korid-1, len(id))):
        xNKassist.append(x[i])
    yNKassist=[]
    for i in reversed(range(korid-1, len(id))):
        yNKassist.append(y[i])
    sensex.reverse()
    sensey.reverse()
    u.reverse()
    v.reverse()
    XNKnew=[]
    for i in range(len(xNKassist)):
        if i==0:
            xnk=xNKassist[i]
        else:
            xnk=xNKassist[i]+(Inflating*sensex[i-1]*u[i-1]*SfNK[i-1])
        XNKnew.append(xnk)
    XNKnew.reverse()
    XNKnew.remove(XNKnew[0])
    YNKnew=[]
    for i in range(len(xNKassist)):
        if i==0:
            ynk=yNKassist[i]
        else:
            ynk=yNKassist[i]+(Inflating*sensey[i-1]*v[i-1]*SfNK[i-1])
        YNKnew.append(ynk)
    YNKnew.reverse()
    YNKnew.remove(YNKnew[0])
    Xbal=[]
    Xbal=X1K+XNKnew
    Ybal=[]
    Ybal=Y1K+YNKnew
    Xfinal=Xbal
    Yfinal=Ybal

# =====
# ===== Telika X kai Y =====
# ===== ElegxosgiaefarmogiAfinikouMetasximatismou =====
# =====

# - Efarmozetaiotan: CCP>=e/2 kai otan CCP<e/2 kai CCParx>e/2 (X,Ytelika -->
X,Y Affine)
# - Den Efarmozetaiotan CCP<e/2 kai CCParx<e/2 (X,Ytelika --> X,Y Balloon)

Xf=[]
Yf=[]
IDf=[]
if CCP>=e[0]/2:
    a=((Yfinal[korid-1]*xpa[0])-(Yfinal[len(id)-1]*xpa[0])+(Yfinal[len(id)-
1]*xpa[1])-(Yfinal[0]*xpa[1])+(Yfinal[0]*xpa[2])-(Yfinal[korid-
1]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(Xfinal[0]*Yfinal[len(id)-1])-(
Yfinal[0]*Xfinal[korid-1])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-
1]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[korid-1])) #
Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
    b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[korid-
1]*xpa[0])+(Xfinal[0]*xpa[1])-(Xfinal[len(id)-1]*xpa[1])+(Xfinal[korid-

```

```

1]*xpa[2])-(Xfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1]))+(Xfinal[korid-1]*Yfinal[len(id)-1]))-
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    c=((Xfinal[korid-1]*Yfinal[len(id)-1]*xpa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-1]*Yfinal[0]*xpa[1])-(
Yfinal[len(id)-1]*Xfinal[0]*xpa[1]))+(Xfinal[0]*Yfinal[korid-1]*xpa[2])-(
(Xfinal[korid-1]*Yfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1]))+(Xfinal[korid-1]*Yfinal[len(id)-1]))-
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    d=((Yfinal[korid-1]*ypa[0])-(Yfinal[len(id)-1]*ypa[0]))+(Yfinal[len(id)-
1]*ypa[1])-(Yfinal[0]*ypa[1]))+(Yfinal[0]*ypa[2])-(Yfinal[korid-
1]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(Xfinal[0]*Yfinal[len(id)-1]))-
(Yfinal[0]*Xfinal[korid-1]))+(Yfinal[0]*Xfinal[len(id)-1]))+(Xfinal[korid-
1]*Yfinal[len(id)-1]))-(Xfinal[len(id)-1]*Yfinal[korid-1]))
    ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[korid-
1]*ypa[0]))+(Xfinal[0]*ypa[1])-(Xfinal[len(id)-1]*ypa[1]))+(Xfinal[korid-
1]*ypa[2])-(Xfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1]))+(Xfinal[korid-1]*Yfinal[len(id)-1]))-
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    f=((Xfinal[korid-1]*Yfinal[len(id)-1]*ypa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*ypa[0]))+(Xfinal[len(id)-1]*Yfinal[0]*ypa[1])-(
(Xfinal[0]*Yfinal[len(id)-1]*ypa[1]))+(Xfinal[0]*Yfinal[korid-1]*ypa[2]))-
(Xfinal[korid-1]*Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1]))+(Xfinal[korid-1]*Yfinal[len(id)-1]))-
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    Xf.append(x[0]) # Eisagwgiprw tousimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos
    Yf.append(y[0])
    IDf.append(id[0])
    for i in range(1, len(id)-1):
        xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c #
Ypologismosnewsintetagmenwn apo afinikometasximatismotwnsimeiw n 2-->n-1
        yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
        Xf.append(xaffine)
        Yf.append(yaffine)
        IDf.append(i+1)
    Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi Gauss /
den efarmozetaiafinikosmetasximatismos
    Yf.append(y[len(id)-1])
    IDf.append(id[len(id)-1])
elif CCP<e[0]/2:
    CCParx=math.sqrt(((xc-xpa[1])**2)+((yc-ypa[1])**2)) #
Ypologismosapostasiskentroukyklou C kai CP arxikisgrammis
    if CCParx>e[0]/2:
        a=((Yfinal[korid-1]*xpa[0])-(Yfinal[len(id)-
1]*xpa[0]))+(Yfinal[len(id)-1]*xpa[1])-(Yfinal[0]*xpa[1]))+(Yfinal[0]*xpa[2])-(
(Yfinal[korid-1]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1]))+(Xfinal[korid-1]*Yfinal[len(id)-1]))-
(Xfinal[len(id)-1]*Yfinal[korid-1])) #
Ypologismosintelestwna,b,c,d,e,fafinikometasximatismou
        b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[korid-
1]*xpa[0]))+(Xfinal[0]*xpa[1])-(Xfinal[len(id)-1]*xpa[1]))+(Xfinal[korid-
1]*xpa[2])-(Xfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(

```

```

(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
    c=((Xfinal[korid-1]*Yfinal[len(id)-1]*xpa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-1]*Yfinal[0]*xpa[1])-(
Yfinal[len(id)-1]*Xfinal[0]*xpa[1])+(Xfinal[0]*Yfinal[korid-1]*xpa[2])-(
Xfinal[korid-1]*Yfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
    d=((Yfinal[korid-1]*ypa[0])-(Yfinal[len(id)-
1]*ypa[0])+(Yfinal[len(id)-1]*ypa[1])-(Yfinal[0]*ypa[1])+(Yfinal[0]*ypa[2])-(
Yfinal[korid-1]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
    ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[korid-
1]*ypa[0])+(Xfinal[0]*ypa[1])-(Xfinal[len(id)-1]*ypa[1])+(Xfinal[korid-
1]*ypa[2])-(Xfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
    f=((Xfinal[korid-1]*Yfinal[len(id)-1]*ypa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*ypa[0])+(Xfinal[len(id)-1]*Yfinal[0]*ypa[1])-(
Xfinal[0]*Yfinal[len(id)-1]*ypa[1])+(Xfinal[0]*Yfinal[korid-1]*ypa[2])-(
Xfinal[korid-1]*Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
    Xf.append(x[0]) # Eisagwgiprw tousimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos
    Yf.append(y[0])
    IDf.append(id[0])
    for i in range(1, len(id)-1):
        xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c #
Ypologismosnewsintetagmenwn apo afinikometasximatismotwnsimeiw n 2-->n-1
        yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
        Xf.append(xaffine)
        Yf.append(yaffine)
        IDf.append(i+1)
    Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi
Gauss / den efarmozetaiafinikosmetasximatismos
    Yf.append(y[len(id)-1])
    IDf.append(id[len(id)-1])
elifCCParx<e[0]/2:
    Xf=Xfinal
    Yf=Yfinal
    IDf=id

# ===== ExagwgiArxeiou =====
fout=open('C:GeneralizedTypeAPart.txt', 'w')
fout.write(str('Polyline')+'\n')
for i in range(len(id)):
    fout.write(str(IDf[i])+' '+str(Xf[i])+' '+str(Yf[i])+'\n')

fout.write(str('END'))
fout.close()

```

```
g.close()
m.close()
```

### Διόγκωση χωρίς δεσμεύσεις

```
import arcgisscripting
gp = arcgisscripting.create(9.3)

# ===== EisagwgiDedomenwn =====
# ----- EisagwgiGrammis Gauss -----
f = open('C:SmoothPart.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in f:
    NumLines=NumLines+1

f = open('C:SmoothPart.txt') # Anoigmaarxeiou

# Eisagogisimeiwngammis (x,y,id)
x=[]
y=[]
id=[]
mtr=0
for line in f:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        id.append(int(line[:-1].split(' ')[0]))
        x.append(float(line[:-1].split(' ')[1]))
        y.append(float(line[:-1].split(' ')[2]))

# ----- EisagwgiArxikisGrammis -----
g = open('C:LinePart_Points.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in g:
    NumLines=NumLines+1

g = open('C:LinePart_Points.txt') # Anoigmaarxeiou

# Eisagogisimeiwngammis (inix,iniy,iniid)
inix=[]
iniy=[]
iniid=[]
mtr=0
for line in g:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        iniid.append(int(line[:-1].split(' ')[0]))
        inix.append(float(line[:-1].split(' ')[1]))
        iniy.append(float(line[:-1].split(' ')[2]))

# ----- Eisagwgi e -----
m = open('C:e.txt')
e=[]
for line in m:
    e.append(float(line[:]))

# ----- EisagwgiSyntelestiDiogwsis (Inflating) -----
o = open('C:Inflating.txt')
Inflating=[]
```



```
for line in o:
    Inflating.append(float(line[:]))

import math

# =====
# ===== EntopismosKoryfisGrammis Gauss mesw ALR =====
# ===== EntopismosAntistoixisKoryfisstinArxikiGrammi =====
# =====

# Ypologismosapostasewnmetaxisimeiw
Sbp=[]
for i in range(len(id)-1):
    Sbps=math.sqrt(((x[i+1]-x[i])**2)+((y[i+1]-y[i])**2))
    Sbp.append(Sbps)

# Ypologismos LR1, LR2, LR3, LR4, ALR, maxALR kai entopismosKoryfis
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

if len(id)==3: # ----- Efarmogi se bend me 3 simeia -----
    CPnum=id[0]
elif len(id)==4: # ----- Efarmogi se bend me 4 simeia -----
    for i in range(len(id)): # Ypologismos LR1
        if i in range(0,1):
            llr1=1
        elif i in range(1,len(id)-1):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            llr1=l1/s1
        elif i in range(len(id)-1,len(id)):
            llr1=1
        LR1.append(llr1)
        alr=LR1[i] # Ypologismos ALR(==LR1)
        ALR.append(alr)
        if ALR[i]==max(ALR): # Entopismos ID
            simeioupuantistoixeistomegisto ALR
            CPnum=id[i-1]
elif len(id)==5 or len(id)==6: # ----- Efarmogi se bend 5 'h 6 simeia -----
    for i in range(len(id)): # Ypologismos LR1, LR2
        if i in range(0,2):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            llr2=l2/s2
        elif i in range(2,len(id)-2):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
            llr2=l2/s2
        elif i in range(len(id)-2,len(id)):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
```

```

        s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
        l1r2=l2/s2
    LR1.append(l1r1)
    LR2.append(l1r2)
    alr=(LR1[i]+LR2[i])/2 # Υπολογισμος ALR
    ALR.append(alr)
    if ALR[i]==max(ALR): # Entopismos ID
simeioupuantistoixeistomegisto ALR
        CPnum=id[i-1]
elif len(id)==7 or len(id)==8: # ----- Efarmogi se bend me 7 h' 8 simeia -----
    for i in range(len(id)): # Υπολογισμος LR1, LR2, LR3
        if i in range(0,3):
            l1r1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            l1r2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i])**2)+((y[i+3]-y[i])**2))
            l1r3=l3/s3
        elif i in range(3,len(id)-3):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((x[i+1]-x[i-1])**2)+((y[i+1]-y[i-1])**2))
            l1r1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i-2])**2)+((y[i+2]-y[i-2])**2))
            l1r2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i-3])**2)+((y[i+3]-y[i-3])**2))
            l1r3=l3/s3
        elif i in range(len(id)-3,len(id)):
            l1r1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((x[i-2]-x[i])**2)+((y[i-2]-y[i])**2))
            l1r2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((x[i-3]-x[i])**2)+((y[i-3]-y[i])**2))
            l1r3=l3/s3
    LR1.append(l1r1)
    LR2.append(l1r2)
    LR3.append(l1r3)
    alr=(LR1[i]+LR2[i]+LR3[i])/3 # Υπολογισμος ALR
    ALR.append(alr)
    if ALR[i]==max(ALR): # Entopismos ID
simeioupuantistoixeistomegisto ALR
        CPnum=id[i-1]
elif len(id)>=9:
    for i in range(len(id)): # Efarmogi se bend me toulaxiston 9 simeia
        if i in range(0,4): # Υπολογισμος LR1, LR2, LR3, LR4
            l1r1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((x[i+2]-x[i])**2)+((y[i+2]-y[i])**2))
            l1r2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((x[i+3]-x[i])**2)+((y[i+3]-y[i])**2))
            l1r3=l3/s3
            l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((x[i+4]-x[i])**2)+((y[i+4]-y[i])**2))
            l1r4=l4/s4

```

```

elifi in range(4, len(id)-4):
    l1=Sbp[i-1]+Sbp[i]
    s1=math.sqrt((x[i+1]-x[i-1])**2)+(y[i+1]-y[i-1])**2)
    l1r1=l1/s1
    l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
    s2=math.sqrt((x[i+2]-x[i-2])**2)+(y[i+2]-y[i-2])**2)
    l1r2=l2/s2
    l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
    s3=math.sqrt((x[i+3]-x[i-3])**2)+(y[i+3]-y[i-3])**2)
    l1r3=l3/s3
    l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
    s4=math.sqrt((x[i+4]-x[i-4])**2)+(y[i+4]-y[i-4])**2)
    l1r4=l4/s4
elifi in range(len(id)-4, len(id)):
    l1r1=1
    l2=Sbp[i-1]+Sbp[i-2]
    s2=math.sqrt((x[i-2]-x[i])**2)+(y[i-2]-y[i])**2)
    l1r2=l2/s2
    l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
    s3=math.sqrt((x[i-3]-x[i])**2)+(y[i-3]-y[i])**2)
    l1r3=l3/s3
    l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
    s4=math.sqrt((x[i-4]-x[i])**2)+(y[i-4]-y[i])**2)
    l1r4=l4/s4
LR1.append(l1r1)
LR2.append(l1r2)
LR3.append(l1r3)
LR4.append(l1r4)
alr=(LR1[i]+LR2[i]+LR3[i]+LR4[i])/4 # Υπολογισμος ALR
ALR.append(alr)
if ALR[i]==max(ALR): # Entopismos ID
simeioupuantistoiixeistomegisto ALR
    CPnum=id[i-1]

# Syntetagmenes kai id Koryfisgrammis Gauss
korx=x[CPnum]
kory=y[CPnum]
korid=id[CPnum]

# -----
# Dimiourgiaarxeiou me Inflection Points kai Koryfigrammis Gauss
xp=[]
yp=[]
idp=[]
xp.append(x[0])
xp.append(korx)
xp.append(x[len(id)-1])
yp.append(y[0])
yp.append(kory)
yp.append(y[len(id)-1])
idp.append(1)
idp.append(2)
idp.append(3)

# Dimiourgiaarxeiou me Inflection Points kai KoryfiArxikisgrammis
xpa=[]
ypa=[]

```

```
xpa.append(inix[0])
xpa.append(inix[CPnum]) # Entopismoskoryfis
xpa.append(inix[len(id)-1])
ypa.append(iniy[0])
ypa.append(iniy[CPnum])
ypa.append(iniy[len(id)-1])
# -----

# ----- YpologismosXarakteristikwn Bend -----
# Ypologismoskentroukyklou C (metaxiarxis-peratos)
xc=(x[0]+x[len(id)-1])/2
yc=(y[0]+y[len(id)-1])/2

# Ypologismosapostasiskentroukyklou C kai CP grammis Gauss
CCP=math.sqrt(((xc-x[korid-1])**2)+((yc-y[korid-1])**2))

# Ypologismosapostasiskentroukyklou C kai CP arxikisgrammis
CCParx=math.sqrt(((xc-xpa[1])**2)+((yc-ypa[1])**2))
# -----

# =====
# ===== DiogwsiGrammis =====
# =====

# ----- Ypologismossyntelestwn u kai v -----
# Ypologismos dx kai dygia ta simeia 2 to n-1
dx=[]
dy=[]
for i in range(1, len(id)-1):
    ddx=x[i+1]-x[i-1]
    ddy=y[i+1]-y[i-1]
    dx.append(ddx)
    dy.append(ddy)

# Ypologismosgniasdiefthinsiseftheias n-1---n+1 giakathesimeio
w=[]
for i in range(len(dx)):
    if dx[i]>0 and dy[i]>0:

        ww=((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
        elif dx[i]>0 and dy[i]<0:
            ww= 200-
            ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
            elif dx[i]<0 and dy[i]<0:
                ww=
                200+((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
                elif dx[i]<0 and dy[i]>0:
                    ww= 400-
                    ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
                    elif dx[i]==0 and dy[i]>0:
                        ww=0
                    elif dx[i]==0 and dy[i]<0:
                        ww=200
                    elif dx[i]>0 and dy[i]==0:
                        ww=100
                    elif dx[i]<0 and dy[i]==0:
                        ww=300
                    w.append(ww)
```

```
# Ypologismos diefthinsismetatopisissheimiwn (kathetistin w)
wm=[]
for i in range(len(w)):
    wwm=w[i]-100
    wm.append(wwm)

# Elegxos giapithaniafaireisi h' prosthesi 400g
wmf=[]
for i in range(len(wm)):
    if wm[i]>400:
        wwmf=wm[i]-400
    elif wm[i]<0:
        wwmf=wm[i]+400
    else:
        wwmf=wm[i]
    wmf.append(wwmf)

# Metatropi grad se rad
wmfr=[]
for i in range(len(wmf)):
    wwmfr=(math.pi*wmf[i])/200
    wmfr.append(wwmfr)

# Ypologismos syntelestwn u kai v
u=[]
for i in range(len(wmfr)):
    uu=math.fabs(math.sin(wmfr[i]))
    u.append(uu)

v=[]
for i in range(len(wmfr)):
    vv=math.fabs(math.cos(wmfr[i]))
    v.append(vv)

# ----- Ypologismos sense -----
sensex=[]
sensey=[]
for i in range(len(wmfr)):
    if wmfr[i]>=0 and wmfr[i]<(math.pi)/2:
        sensx=1
        sensey=1
    elif wmfr[i]>=(math.pi)/2 and wmfr[i]<math.pi:
        sensx=1
        sensey=-1
    elif wmfr[i]>=math.pi and wmfr[i]<(3*math.pi)/2:
        sensx=-1
        sensey=-1
    elif wmfr[i]>=(3*math.pi)/2 and wmfr[i]<2*math.pi:
        sensx=-1
        sensey=1
    sensex.append(sensx)
    sensey.append(sensey)

# ----- Ypologismos S/Sn -----
# Ypologismos S simeiw n 1 ws Koryfi
# Ypologismos apostasewn metaxisimeiw n
Sbp1K=[]
```

```
for i in range (0, korid-1):
    Sbps1K=math.sqrt(((x[i+1]-x[i])**2)+(y[i+1]-y[i])**2))
    Sbp1K.append(Sbps1K)

# Ypologismos S kathesimeiou
S1K=[]
for i in range(len(Sbp1K)):
    if i==0:
        sslk=0+Sbp1K[i]
    else:
        sslk=sslk+Sbp1K[i]
    S1K.append(sslk)

# Ypologismos Sn apo 1 wsKoryfi
Sn1K=sum(Sbp1K, 0)

# Ypologismos S simeiw n wsKoryfi
# Ypologismosapostasewnmetaxisimeiw n
SbpNK=[]
for i in reversed(range(korid-1, len(id)-1)):
    SbpsNK=math.sqrt(((x[i]-x[i+1])**2)+(y[i]-y[i+1])**2))
    SbpNK.append(SbpsNK)

# Ypologismos S kathesimeiou
SNK=[]
for i in range(len(SbpNK)):
    if i==0:
        sskn=0+SbpNK[i]
    else:
        sskn=sskn+SbpNK[i]
    SNK.append(sskn)

# Ypologismos Sn apo n wsKoryfi
SnNK=sum(SbpNK, 0)

# Ypologismos S/Sn (S1K/Sn1K) apo 2 wsKoryfi
Sf1K=[]
for i in range(len(S1K)):
    sf1k=S1K[i]/Sn1K
    Sf1K.append(sf1k)

# Ypologismos S/Sn (SNK/SnNK) apo n-1 wsKoryfi
SfNK=[]
for i in range(len(SNK)):
    sfnk=SNK[i]/SnNK
    SfNK.append(sfnk)

# ----- YpologismosXnew, YnewDiogwmenisGrammis -----
# YpologismosXnew apo 1 wsKoryfi
X1K=[]
for i in range (0, korid):
    if i==0:
        x1k=x[i]
    else:
        x1k=x[i]+(Inflating[0]*senssex[i-1]*u[i-1]*Sf1K[i-1])
    X1K.append(x1k)

# YpologismosYnew apo 1 wsKoryfi
```

```
Y1K=[]
for i in range (0, korid):
    if i==0:
        y1k=y[i]
    else:
        y1k=y[i]+(Inflating[0]*sensey[i-1]*v[i-1]*Sf1K[i-1])
    Y1K.append(y1k)

# Dimiourgia voithitikwn pinakwn pou perilamvanoun mono ta simeia N ws K
xNKassist=[]
for i in reversed(range(korid-1, len(id))):
    xNKassist.append(x[i])

yNKassist=[]
for i in reversed(range(korid-1, len(id))):
    yNKassist.append(y[i])

# Antistrofiarxeiwn sensex, sensey, u kai v
gianaxrisimopoihthoun ston ypologismo X,Y twn N ws K
sensex.reverse()
sensey.reverse()
u.reverse()
v.reverse()

# Ypologismos Xnew apo N ws Koryfi
XNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        xnk=xNKassist[i]
    else:
        xnk=xNKassist[i]+(Inflating[0]*sensex[i-1]*u[i-1]*SfNK[i-1])
    XNKnew.append(xnk)

# Antistrofi XNK se XKN
XNKnew.reverse()
# Diagrafi lou stoixeiou (dil. tou X tou K pou parxei kai sto X1K)
XNKnew.remove(XNKnew[0])

# Ypologismos Ynew apo N ws Koryfi
YNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        ynk=yNKassist[i]
    else:
        ynk=yNKassist[i]+(Inflating[0]*sensey[i-1]*v[i-1]*SfNK[i-1])
    YNKnew.append(ynk)

# Antistrofi YNK se YKN
YNKnew.reverse()
# Diagrafi lou stoixeiou (dil. tou Y tou K pou parxei kai sto Y1K)
YNKnew.remove(YNKnew[0])

# Enwsipinakwn ---- X kai Y Diogwmenis Grammis ----
Xbal=[]
Xbal=X1K+XNKnew

Ybal=[]
Ybal=Y1K+YNKnew
```

```
# ----- Elegxos an exeigineidiogwsi h' siriknwsigrammis -----
# TelikaX,Ygrammis Balloon meta ton elegxo
Xfinal=[]
Yfinal=[]

# Ypologismosapostasiskentroukyklou C kai CP diogwmenisgrammis
CCPbal=math.sqrt(((xc-Xbal[korid-1])**2)+(yc-Ybal[korid-1])**2)

# An CCP<CCPbalexexeigineidiogwsi
# An CCP>CCPbalexexeigineisiriknws --
>xrisimopoietaiantitithetidiefthinsimetatopisis kai xanatrexei to programma
if CCP<CCPbal:
    Xfinal=Xbal
    Yfinal=Ybal
elif CCP>CCPbal:
    wm=[]
    for i in range(len(w)):
        wwm=w[i]+100
        wm.append(wwm)
    wmf=[]
    for i in range(len(wm)):
        if wm[i]>400:
            wwmf=wm[i]-400
        elifwm[i]<0:
            wwmf=wm[i]+400
        else:
            wwmf=wm[i]
        wmf.append(wwmf)
    wmfr=[]
    for i in range(len(wmf)):
        wwmfr=(math.pi*wmf[i])/200
        wmfr.append(wwmfr)
    u=[]
    for i in range(len(wmfr)):
        uu=math.fabs(math.sin(wmfr[i]))
        u.append(uu)
    v=[]
    for i in range(len(wmfr)):
        vv=math.fabs(math.cos(wmfr[i]))
        v.append(vv)
    sensex=[]
    sensey=[]
    for i in range(len(wmfr)):
        if wmfr[i]>=0 and wmfr[i]<(math.pi)/2:
            sensx=1
            sensey=1
        elifwmfr[i]>=(math.pi)/2 and wmfr[i]<math.pi:
            sensx=1
            sensey=-1
        elifwmfr[i]>=math.pi and wmfr[i]<(3*math.pi)/2:
            sensx=-1
            sensey=-1
        elifwmfr[i]>=(3*math.pi)/2 and wmfr[i]<2*math.pi:
            sensx=-1
            sensey=1
        sensex.append(sensx)
        sensey.append(sensy)
```



```
Sbp1K=[]
for i in range (0, korid-1):
    Sbps1K=math.sqrt(((x[i+1]-x[i])**2)+(y[i+1]-y[i])**2))
    Sbp1K.append(Sbps1K)
S1K=[]
for i in range(len(Sbp1K)):
    if i==0:
        sslk=0+Sbp1K[i]
    else:
        sslk=sslk+Sbp1K[i]
    S1K.append(sslk)
Sn1K=sum(Sbp1K, 0)
SbpNK=[]
for i in reversed(range(korid-1, len(id)-1)):
    SbpsNK=math.sqrt(((x[i]-x[i+1])**2)+(y[i]-y[i+1])**2))
    SbpNK.append(SbpsNK)
SNK=[]
for i in range(len(SbpNK)):
    if i==0:
        sskn=0+SbpNK[i]
    else:
        sskn=sskn+SbpNK[i]
    SNK.append(sskn)
SnNK=sum(SbpNK, 0)
Sf1K=[]
for i in range(len(S1K)):
    sf1k=S1K[i]/Sn1K
    Sf1K.append(sf1k)
SfNK=[]
for i in range(len(SNK)):
    sfnk=SNK[i]/SnNK
    SfNK.append(sfnk)
X1K=[]
for i in range (0, korid):
    if i==0:
        x1k=x[i]
    else:
        x1k=x[i]+(Inflating[0]*sensex[i-1]*u[i-1]*Sf1K[i-1])
    X1K.append(x1k)
Y1K=[]
for i in range (0, korid):
    if i==0:
        y1k=y[i]
    else:
        y1k=y[i]+(Inflating[0]*sensey[i-1]*v[i-1]*Sf1K[i-1])
    Y1K.append(y1k)
xNKassist=[]
for i in reversed(range(korid-1, len(id))):
    xNKassist.append(x[i])
yNKassist=[]
for i in reversed(range(korid-1, len(id))):
    yNKassist.append(y[i])
sensex.reverse()
sensey.reverse()
u.reverse()
v.reverse()
XNKnew=[]
for i in range(len(xNKassist)):
```

```

        if i==0:
            xnk=xNKassist[i]
        else:
            xnk=xNKassist[i]+(Inflating[0]*sensex[i-1]*u[i-1]*SfNK[i-1])
        XNKnew.append(xnk)
    XNKnew.reverse()
    XNKnew.remove(XNKnew[0])
    YNKnew=[]
    for i in range(len(xNKassist)):
        if i==0:
            ynk=yNKassist[i]
        else:
            ynk=yNKassist[i]+(Inflating[0]*sensesy[i-1]*v[i-1]*SfNK[i-1])
        YNKnew.append(ynk)
    YNKnew.reverse()
    YNKnew.remove(YNKnew[0])
    Xbal=[]
    Xbal=X1K+XNKnew
    Ybal=[]
    Ybal=Y1K+YNKnew
    Xfinal=Xbal
    Yfinal=Ybal

# =====
# ===== Telika X kai Y =====
# ===== EfarmogiAfinikouMetasximatismou =====
# =====

# Periptwseis:
# - otan CCP>=e/2 kai otan CCP<e/2 kai CcParx>e/2 affinikoswsteCPballoon --
>CParxkis
# - otan CCP<e/2 kai CcParx<e/2 affinikoswsteCPballoon --
>tomieftheiasmetatopisisCPgauss - kyklou

Xf=[]
Yf=[]
IDf=[]
if CCP>=e[0]/2: # affinikoswsteCPballoon -->CParxkis
    a=((Yfinal[korid-1]*xpa[0])-(Yfinal[len(id)-1]*xpa[0])+(Yfinal[len(id)-1]*xpa[1])-(Yfinal[0]*xpa[1])+(Yfinal[0]*xpa[2])-(Yfinal[korid-1]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-1])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[korid-1])) #
    Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
    b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[korid-1]*xpa[0])+(Xfinal[0]*xpa[1])-(Xfinal[len(id)-1]*xpa[1])+(Xfinal[korid-1]*xpa[2])-(Xfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-1])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[korid-1]))
    c=((Xfinal[korid-1]*Yfinal[len(id)-1]*xpa[0])-(Yfinal[korid-1]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-1]*Yfinal[0]*xpa[1])-(Yfinal[len(id)-1]*Xfinal[0]*xpa[1])+(Xfinal[0]*Yfinal[korid-1]*xpa[2])-(Xfinal[korid-1]*Yfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-1])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[korid-1]))

```

```

d=((Yfinal[korid-1]*ypa[0])-(Yfinal[len(id)-1]*ypa[0])+(Yfinal[len(id)-
1]*ypa[1])-(Yfinal[0]*ypa[1])+(Yfinal[0]*ypa[2])-(Yfinal[korid-
1]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(Xfinal[0]*Yfinal[len(id)-1])-(
Yfinal[0]*Xfinal[korid-1])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-
1]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[korid-1]))
ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[korid-
1]*ypa[0])+(Xfinal[0]*ypa[1])-(Xfinal[len(id)-1]*ypa[1])+(Xfinal[korid-
1]*ypa[2])-(Xfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
f=((Xfinal[korid-1]*Yfinal[len(id)-1]*ypa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*ypa[0])+(Xfinal[len(id)-1]*Yfinal[0]*ypa[1])-(
Xfinal[0]*Yfinal[len(id)-1]*ypa[1])+(Xfinal[0]*Yfinal[korid-1]*ypa[2])-(
Xfinal[korid-1]*Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
Xf.append(x[0]) # Eisagwgiprwtoousimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos
Yf.append(y[0])
IDf.append(id[0])
for i in range(1, len(id)-1):
    xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c #
Ypologismosnewsintetagmenwn apo afinikometasximatismotwnsimeiw n 2-->n-1
    yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
    Xf.append(xaffine)
    Yf.append(yaffine)
    IDf.append(i+1)
Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi Gauss /
den efarmozetaiafinikosmetasximatismos
Yf.append(y[len(id)-1])
IDf.append(id[len(id)-1])
elif CCP<e[0]/2: # Ypologismostomiskyklou-eftheiasmetatopisistouCPgauss
l=-((x[korid]-x[korid-2])/(y[korid]-y[korid-2])) #
Ypologismossyntelestidiefthinsis (l) eftheiasmetatopisiskoryfis
D=(((-2*xc)-(2*(1**2)*(x[korid-1]))+(2*1*y[korid-1])-(2*yc*1)**2)-
(4*((1**2)+1)*((xc**2)+(yc**2)-((e[0]**2)/4)+((1**2)*(x[korid-1]**2))-
(2*1*x[korid-1]*y[korid-1])+(y[korid-1]**2)+(2*yc*1*x[korid-1])-
(2*yc*y[korid-1]))) # YpologismosDiakrinousas D
xt1=((2*xc)+(2*(1**2)*x[korid-1])-(2*1*y[korid-
1])+(2*yc*1)+(math.sqrt(D)))/((2*(1**2))+2) # Ypologismospithanwnlisewn
t1,t2
yt1=(1*xt1)-(1*x[korid-1])+y[korid-1]
xt2=((2*xc)+(2*(1**2)*x[korid-1])-(2*1*y[korid-1])+(2*yc*1))-
(math.sqrt(D))/((2*(1**2))+2)
yt2=(1*xt2)-(1*x[korid-1])+y[korid-1]
ap1=math.sqrt(((x[korid-1]-xt1)**2)+((y[korid-1]-yt1)**2)) #
Ypologismosapostasewn CP-t1 (ap1) kai CP-t2 (ap2)
ap2=math.sqrt(((x[korid-1]-xt2)**2)+((y[korid-1]-yt2)**2))
if ap1>ap2: # Entopismosapodektis lysis systimatos (to
xtpouapexeimikroteriapostasi apo to CP)
    xtomi=xt2
    ytomi=yt2
elif ap1<ap2:
    xtomi=xt1
    ytomi=yt1
if CCParx>e[0]/2: # affinikoswsteCPballoon -->CParxkis

```

```

a=((Yfinal[korid-1]*xpa[0])-(Yfinal[len(id)-
1]*xpa[0])+(Yfinal[len(id)-1]*xpa[1])-(Yfinal[0]*xpa[1])+(Yfinal[0]*xpa[2])-(
Yfinal[korid-1]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1])) #
Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[korid-
1]*xpa[0])+(Xfinal[0]*xpa[1])-(Xfinal[len(id)-1]*xpa[1])+(Xfinal[korid-
1]*xpa[2])-(Xfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
c=((Xfinal[korid-1]*Yfinal[len(id)-1]*xpa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-1]*Yfinal[0]*xpa[1])-(
Yfinal[len(id)-1]*Xfinal[0]*xpa[1])+(Xfinal[0]*Yfinal[korid-1]*xpa[2])-(
Xfinal[korid-1]*Yfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
d=((Yfinal[korid-1]*ypa[0])-(Yfinal[len(id)-
1]*ypa[0])+(Yfinal[len(id)-1]*ypa[1])-(Yfinal[0]*ypa[1])+(Yfinal[0]*ypa[2])-(
Yfinal[korid-1]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[korid-
1]*ypa[0])+(Xfinal[0]*ypa[1])-(Xfinal[len(id)-1]*ypa[1])+(Xfinal[korid-
1]*ypa[2])-(Xfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
f=((Xfinal[korid-1]*Yfinal[len(id)-1]*ypa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*ypa[0])+(Xfinal[len(id)-1]*Yfinal[0]*ypa[1])-(
Xfinal[0]*Yfinal[len(id)-1]*ypa[1])+(Xfinal[0]*Yfinal[korid-1]*ypa[2])-(
Xfinal[korid-1]*Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-(
Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
Xfinal[len(id)-1]*Yfinal[korid-1]))
Xf.append(x[0]) # Eisagwgiprwtoousimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos
Yf.append(y[0])
IDf.append(id[0])
for i in range(1, len(id)-1):
    xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c #
Ypologismosnewsintetagmenwn apo afinikometasximatismotwnsimeiw n 2-->n-1
    yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
    Xf.append(xaffine)
    Yf.append(yaffine)
    IDf.append(i+1)
Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi
Gauss / den efarmozetaiafinikosmetasximatismos
Yf.append(y[len(id)-1])
IDf.append(id[len(id)-1])
elifCCParx<e[0]/2: # afinikoswsteCPballoon --
>tomieftheiasmetatopisisCPgauss - kyklou
    a=((Yfinal[korid-1]*xpa[0])-(Yfinal[len(id)-
1]*xpa[0])+(Yfinal[len(id)-1]*xtomi)-(Yfinal[0]*xtomi)+(Yfinal[0]*xpa[2])-(

```

```

(Yfinal[korid-1]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
(Xfinal[len(id)-1]*Yfinal[korid-1])) #
Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
    b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[korid-
1]*xpa[0])+(Xfinal[0]*xtomi)-(Xfinal[len(id)-1]*xtomi)+(Xfinal[korid-
1]*xpa[2])-(Xfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    c=((Xfinal[korid-1]*Yfinal[len(id)-1]*xpa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-1]*Yfinal[0]*xtomi)-
(Yfinal[len(id)-1]*Xfinal[0]*xtomi)+(Xfinal[0]*Yfinal[korid-1]*xpa[2])-(
(Xfinal[korid-1]*Yfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[korid-1])-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    d=((Yfinal[korid-1]*ypa[0])-(Yfinal[len(id)-
1]*ypa[0])+(Yfinal[len(id)-1]*ytomi)-(Yfinal[0]*ytomi)+(Yfinal[0]*ypa[2])-(
(Yfinal[korid-1]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[korid-
1]*ypa[0])+(Xfinal[0]*ytomi)-(Xfinal[len(id)-1]*ytomi)+(Xfinal[korid-
1]*ypa[2])-(Xfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    f=((Xfinal[korid-1]*Yfinal[len(id)-1]*ypa[0])-(Yfinal[korid-
1]*Xfinal[len(id)-1]*ypa[0])+(Xfinal[len(id)-1]*Yfinal[0]*ytomi)-
(Xfinal[0]*Yfinal[len(id)-1]*ytomi)+(Xfinal[0]*Yfinal[korid-1]*ypa[2])-(
(Xfinal[korid-1]*Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[korid-1])-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[korid-
1]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[korid-1]*Yfinal[len(id)-1])-(
(Xfinal[len(id)-1]*Yfinal[korid-1]))
    Xf.append(x[0]) # Eisagwgiprwtoousimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos
    Yf.append(y[0])
    IDf.append(id[0])
    for i in range(1, len(id)-1):
        xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c #
Ypologismosnewsintetagmenwn apo afinikometasximatismotwnsimeiw n 2-->n-1
        yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
        Xf.append(xaffine)
        Yf.append(yaffine)
        IDf.append(i+1)
    Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi
Gauss / den efarmozetaiafinikosmetasximatismos
    Yf.append(y[len(id)-1])
    IDf.append(id[len(id)-1])

# === YpologismosXarakteristikwnTimwnSyntelestiDiogwsis (Inflating) ==
# (Times Inflating pouypologizontaistin 'ypodesmeuseis' ekdoxitouprogrammatos
(ArcGISBalloonALRnew). Oi times apotelounkatwfli kata tidiadikasiadokimwn)
if CCP>=e[0]/2:

```

```
dCParxCPgauss=math.fabs(math.sqrt(((xpa[1]-x[korid-1])**2)+((ypa[1]-y[korid-1])**2))) # apostasiCPgauss apo CParxiko
elif CCP<e[0]/2:
    dtomiCPgauss=math.fabs(math.sqrt(((xtomi-x[korid-1])**2)+((ytomi-y[korid-1])**2))) # apostasiCPgauss apo tin tomikyklou-eftheiasmetatopisistouCPgauss

# ===== ExagwgiArxeiwn =====
# ----- Elaxistes Times SyntelestiDiogwsis (Inflating) -----
gout=open('C:InflatingLowestValue.txt', 'w')
if CCP>=e[0]/2:
    gout.write('CCP>=e/2'+ ' ... '+'Distance CParx-CPgauss: '+'
'+str(dCParxCPgauss))+'\n')
elif CCP<e[0]/2:
    gout.write('CCP<e/2'+ ' ... '+'Distance CPtomi-CPgauss: '+'
'+str(dtomiCPgauss))+'\n')

gout.close()
# ----- GenikeumeniGrammi -----
fout=open('C:GeneralizedTypeAPartfInf.txt', 'w')
fout.write(str('Polyline')+'\n')
for i in range(len(id)):
    fout.write(str(IDf[i])+ ' '+'(str(Xf[i]))'+ ' '+'(str(Yf[i]))'+ '\n')

fout.write(str('END'))
fout.close()
g.close()
m.close()
o.close()
```

## Π4.2. Πρόγραμμα γενίκευσης αμφίπλευρων ε-μη-κυρτών τμημάτων

### Προεπεξεργασία

```

import arcgisscripting
gp = arcgisscripting.create(9.3)

# ===== EisagwgiDedomenwn =====
# ----- EisagwgiArxikisGrammis -----
f = open('C:LinePart_Points.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammw (lines) arxeiou
for line in f:
    NumLines=NumLines+1

f = open('C:LinePart_Points.txt') # Anoigmaarxeiou

# Eisagwgisimeiwngammis (inix, iniy, iniid)
inix=[]
iniy=[]
iniid=[]
mtr=0
for line in f:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        iniid.append(int(line[:-1].split(' ')[0]))
        inix.append(float(line[:-1].split(' ')[1]))
        iniy.append(float(line[:-1].split(' ')[2]))

# ----- Ypologismosapodektwntimwn 'sigma' -----
sigmaAccept=[]
sig=0
for i in range((len(iniid)-1)/2):
    sig=sig+0.25
    sigmaAccept.append(sig)

sigmaMax=float(len(iniid)-1)/8
sigmaAccept.append(sigmaMax)

# ----- Eisagwgi 'sigma' -----
m = open('C:Sigma.txt')
sigma=[]
for line in m:
    sigma.append(float(line[:]))

import math

# =====
# ===== ExomalinsiArxikisGrammis =====
# =====

# YpologismosX,Y Gauss AkraiwnSimeiwArxisGrammis (arximexri +4sigma)
# Dimiourgiakatoptrikwnsimeiw apo to prwtosimeio
mirrorAx=[]
mirrorAy=[]
for i in range (0, int(4*sigma[0])):
    mAx=2*inix[0]-inix[i+1]
    MAy=2*iniy[0]-iniy[i+1]
    mirrorAx.append(mAx)
    mirrorAy.append(mAy)

```

```
mirrorAx.reverse() # Antistrofipinaka - diorthwsiseirassimeiw  
mirrorAy.reverse()  
  
# Enwsikatoptrikwnsimeiw me arxikigrammi  
arxix=mirrorAx+inix  
arxiy=mirrorAy+iniy  
  
# Ypologismos x(i-k)*g(k) giakathesimeio  
xika=[]  
for i in range(int(4*sigma[0]), int(8*sigma[0])):  
    for k in range(-int(4*sigma[0]), int(4*sigma[0])+1): # (k=-4s-->4s)  
        xikae=arxix[i-  
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-  
(float(k)**2))/(2*(sigma[0]**2))))))  
        xika.append(xikae)  
  
# YpologismosXGauss  
xGa=[]  
xGaassist=[]  
for j in range(1,(len(xika)/int(8*sigma[0]+1))+1):  
    for i in range(0,int(8*sigma[0]+1)):  
        xGaassist.append(xika[i])  
    sum(xGaassist)  
    xGa.append(sum(xGaassist))  
    xika[0:int(8*sigma[0]+1)]=[]  
    xGaassist[:]=[]  
  
# Ypologismos y(i-k)*g(k) giakathesimeio  
yika=[]  
for i in range (int(4*sigma[0]), int(8*sigma[0])):  
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):  
        yikae=arxiy[i-  
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-  
(float(k)**2))/(2*(sigma[0]**2))))))  
        yika.append(yikae)  
  
# YpologismosYGauss  
yGa=[]  
yGaassist=[]  
for j in range(1,(len(yika)/int(8*sigma[0]+1))+1):  
    for i in range(0,int(8*sigma[0]+1)):  
        yGaassist.append(yika[i])  
    sum(yGaassist)  
    yGa.append(sum(yGaassist))  
    yika[0:int(8*sigma[0]+1)]=[]  
    yGaassist[:]=[]  
  
# ===== YpologismosX,Y Gauss MeswnSimeiwNGrammis (+4sigma+1 apo tin  
arximexri -4sigma+1 apo to peras) =====  
# Ypologismos x(i-k)*g(k) giakathesimeio  
xikm=[]  
for i in range (int(4*sigma[0]), len(iniid)-int(4*sigma[0])):  
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):  
        xikme=inix[i-  
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-  
(float(k)**2))/(2*(sigma[0]**2))))))  
        xikm.append(xikme)
```



```
# YpologismosXGauss
xGm=[]
xGmassist=[]
for j in range(1,(len(xikm)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0]+1)):
        xGmassist.append(xikm[i])
    sum(xGmassist)
    xGm.append(sum(xGmassist))
    xikm[0:int(8*sigma[0]+1)]=[]
    xGmassist[:]=[] # Adeiasmavoithitikoupinaka

# Ypologismos y(i-k)*g(k) giakathesimeio
yikm=[]
for i in range (int(4*sigma[0]), len(iniid)-int(4*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0]+1)):
        yikme=iniy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yikm.append(yikme)

# YpologismosYGauss
yGm=[]
yGmassist=[]
for j in range(1,(len(yikm)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0]+1)):
        yGmassist.append(yikm[i])
    sum(yGmassist)
    yGm.append(sum(yGmassist))
    yikm[0:int(8*sigma[0]+1)]=[]
    yGmassist[:]=[]

# YpologismosX,Y Gauss akraiwnsimeiw nperatosgrammis (-4sigma mexriperas)
# Dimiourgiakatoptrikw nsimeiw n apo to teleutaiosimeio
mirrorPx=[]
mirrorPy=[]

# Dimiourgiaantistrofwnpinakwn x kai y
xrev=[]
yrev=[]
for i in range (len(iniid)):
    xrev.append(inix[i])
    yrev.append(iniy[i])

xrev.reverse()
yrev.reverse()

for i in range (0, int(4*sigma[0])):
    mPx=2*xrev[0]-xrev[i+1]
    mPy=2*yrev[0]-yrev[i+1]
    mirrorPx.append(mPx)
    mirrorPy.append(mPy)

# Enwsikatoptrikw nsimeiw n me arxikigrammi (orthiseirasimeiw n)
perasx=inix+mirrorPx
perasy=iny+mirrorPy

# Antisxtrofipinakwnperasx kai perasy (epexergasia se antistrofouspinakes)
```

```
perasx.reverse()
perasy.reverse()

# Υπολογισμος x(i-k)*g(k) giakathesimeio
xikp=[]
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        xikpe=perasx[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        xikp.append(xikpe)

# ΥπολογισμοςXGauss
xGp=[]
xGpassist=[]
for j in range(1,(len(xikp)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0])+1):
        xGpassist.append(xikp[i])
    sum(xGpassist)
    xGp.append(sum(xGpassist))
    xikp[0:int(8*sigma[0]+1)]=[]
    xGpassist[:]=[]

# Υπολογισμος y(i-k)*g(k) giakathesimeio
yikp=[]
for i in range (int(4*sigma[0]), int(8*sigma[0])):
    for k in range (-int(4*sigma[0]), int(4*sigma[0])+1):
        yikpe=perasy[i-
k]*((1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2)))))
        yikp.append(yikpe)

# ΥπολογισμοςYGauss
yGp=[]
yGpassist=[]
for j in range(1,(len(yikp)/int(8*sigma[0]+1))+1):
    for i in range(0,int(8*sigma[0])+1):
        yGpassist.append(yikp[i])
    sum(yGpassist)
    yGp.append(sum(yGpassist))
    yikp[0:int(8*sigma[0]+1)]=[]
    yGpassist[:]=[]

# AntistrofipinakwnxGp kai yGp - diorthwsiseirassimeiw
xGp.reverse()
yGp.reverse()

# EnwsipinakwnxGa-xGm-xGp / yGa-yGm-yGp
xu=xGa+xGm+xGp
yu=yGa+yGm+yGp
id=iniid

# ===== X kai Y Gauss =====
x=[]
y=[]

# Diorthwsix,y apo sfalmastroggylopoihsisvarous
# Υπολογισμοςathroismatosepimerousvarwn
```

```
gsk=0
for k in range(-int(4*sigma[0]), int(4*sigma[0])+1):
    gsk=gsk+(1/(sigma[0]*(math.sqrt(2*math.pi))))*(math.exp((-
(float(k)**2))/(2*(sigma[0]**2))))

# Diorthwsis syntetagmenwn
for i in range(len(id)):
    xx=xu[i]/gsk
    yy=yu[i]/gsk
    x.append(xx)
    y.append(yy)

# =====
# ===== Entopismos Bends =====
# =====

# Ypologismos apostasewn metaxi twnsimeiwntis grammis
distx=[]
disty=[]
distid=[]
for i in range(len(id)-1):
    dx=x[i+1]-x[i]
    dy=y[i+1]-y[i]
    distx.append(dx)
    disty.append(dy)
    distid.append(i)

# Ypologismos gnias diefthinsise fthi grammwntimiatwn grammis
w=[]
for i in range(len(distid)):
    if distx[i]>0 and disty[i]>0:
        ww=
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
        elif distx[i]>0 and disty[i]<0:
            ww= 200-
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
            elif distx[i]<0 and disty[i]<0:
                ww=
200+((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
                elif distx[i]<0 and disty[i]>0:
                    ww= 400-
((math.atan((math.fabs(distx[i]))/(math.fabs(disty[i])))))*(200/math.pi))
                    elif distx[i]==0 and disty[i]>0:
                        ww=0
                    elif distx[i]==0 and disty[i]<0:
                        ww=200
                    elif distx[i]>0 and disty[i]==0:
                        ww=100
                    elif distx[i]<0 and disty[i]==0:
                        ww=300
                    w.append(ww)

# Elegxos giapithania fairesi h' prosthesi 400g
wf=[]
for i in range(len(w)):
    if w[i]>400:
        wwf=w[i]-400
    elif w[i]<0:
```

```
        wwf=w[i]+400
    else:
        wwf=w[i]
    wf.append(wwf)

# Metatropi grad se rad
wfr=[]
for i in range(len(wf)):
    wwfr=(math.pi*wf[i])/200
    wfr.append(wwfr)

# Ypologismosgwniasthlasissimeiwn 2 ews n-1
slo=[]
for i in range(len(wfr)-1):
    slop=wfr[i+1]-wfr[i]
    slo.append(slop)

# Diorthwsigwniwnthlasis (megaliterwntou pi) / Telikesgwniesthlasis
sl=[]
for i in range(len(slo)):
    if (math.pi)<slo[i]<(math.pi*2):
        ssl=slo[i]-(math.pi*2)
    elif -(math.pi*2)<slo[i]<-(math.pi):
        ssl=slo[i]+(math.pi*2)
    else:
        ssl=slo[i]
    sl.append(ssl)

# Entopismos Inflection Points
# (entopizontaidiadoxikasimeia me antithetiklisi / katagrafontai kai ta 2)
infpx=[]
infpy=[]
infpid=[]
for i in range (len(sl)-1):
    if sl[i]<0 and sl[i+1]>0 or sl[i]>0 and sl[i+1]<0:
        infpx.append(x[i+1])
        infpx.append(x[i+2])
        infpy.append(y[i+1])
        infpy.append(y[i+2])
        infpid.append(id[i+1])
        infpid.append(id[i+2])

# Entopismosmesou (mip) metaxi inflection points kai dimiourgiapinaka me ip-
mip-ip
ipmipx=[]
ipmipy=[]
ipmipid=[]
for i in range(0, len(infpid)-1, 2):
    ipmipxx=(infpx[i]+infpx[i+1])/2
    ipmipyy=(infpy[i]+infpy[i+1])/2
    ipmipidd=(float(infpid[i])+float(infpid[i+1]))/2
    ipmipx.append(infpx[i])
    ipmipx.append(ipmipxx)
    ipmipx.append(infpx[i+1])
    ipmipy.append(infpy[i])
    ipmipy.append(ipmipyy)
    ipmipy.append(infpy[i+1])
    ipmipid.append(infpid[i])
```

```
ipmipid.append(ipmipidd)
ipmipid.append(infpid[i+1])

# Afairesidiplwnkatagrafwn (periptwseis bends triwnsimeiw)
ipmipxassist=[]
ipmipyassist=[]
ipmipidassist=[]
for i in range(1,len(ipmipid)):
    if ipmipid[i]!=ipmipid[i-1]:
        ipmipxassist.append(ipmipx[i])
        ipmipyassist.append(ipmipy[i])
        ipmipidassist.append(ipmipid[i])

# Prosthikitouprwtousimeioustouspinakesipmipassist
ipmipxassist.insert(0, ipmipx[0])
ipmipyassist.insert(0, ipmipy[0])
ipmipidassist.insert(0, ipmipid[0])

# Entopismos non-inflection points grammis
noninfpix=[]
noninfpixy=[]
noninfpid=[]
for i in range(len(id)):
    if id[i] not in ipmipidassist:
        noninfpix.append(x[i])
        noninfpixy.append(y[i])
        noninfpid.append(id[i])

# ===== Dimiourgiatelikisgrammis (me nip, ip kai mip) =====
# Enwsipinakwnipmipassisst-noninfp
xvm=ipmipxassist+noninfpix
yvm=ipmipyassist+noninfpixy
idvm=ipmipidassist+noninfpid

# Telikoipinakesgrammis
vmx=[]
vmy=[]
vmid=[]

# Dimiourgiataxinomimenoupinakaidvm
idvm.sort()
idvmsort=idvm

# Epanadimiourgiapinakaidvm
idvm=ipmipidassist+noninfpid

# Taxinomisipinakwnx,y me vasi to id
for i in range(len(idvmsort)):
    for j in range(len(idvm)):
        if idvmsort[i]==idvm[j]: # Antistoixisitaxinomimenwn kai
atxinomitwn id
            vmx.append(xvm[j]) # Dimiourgiapinakwnx,y
            vmy.append(yvm[j]) # (ta
stoixeiatawnataxinomitwnx,yeisagontaistousneouspinakesstistheseistwntaxinomime
nwn)

# Dimiourgiatelikoupinaka id
vmid=idvmsort
```

```
# =====
# ===== Idiotites Bends =====
# =====

# Υπολογισμος 'i' twnmipstintelikigrammi
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# ===== ΥπολογισμοςMikousGrammisVasis Bends =====
LBLbend=[]
# Prwto bend
bl=math.sqrt(((vmx[0]-vmx[iip[0]])**2)+((vmy[0]-vmy[iip[0]])**2))
LBLbend.append(bl)

# Endiamesa bends
for i in range(len(iip)-1):
    bl=math.sqrt(((vmx[iip[i]]-vmx[iip[i+1]])**2)+((vmy[iip[i]]-
vmy[iip[i+1]])**2))
    LBLbend.append(bl)

# Teleutaio bend
bl=math.sqrt(((vmx[iip[len(iip)-1]]-vmx[len(vmid)-1])**2)+((vmy[iip[len(iip)-
1]]-vmy[len(vmid)-1])**2))
LBLbend.append(bl)

# ===== ΥπολογισμοςEmbadou Bends =====
Areabend=[]
# Prwto bend
area=[]
for i in range(0, iip[0]+1):
    if i==0:
        ar=0.5*(vmy[0]*(vmx[iip[0]]-vmx[i+1]))
    elif i>0 and i<iip[0]:
        ar=0.5*(vmy[i]*(vmx[i-1]-vmx[i+1]))
    elif i==iip[0]:
        ar=0.5*(vmy[iip[0]]*(vmx[i-1]-vmx[0]))
    area.append(ar)

areasum=math.fabs(sum(area))
Areabend.append(areasum)

# Endiamesa bends
area=[]
for j in range(len(iip)-1):
    for i in range(iip[j], iip[j+1]+1):
        if i==iip[j]:
            ar=0.5*(vmy[iip[j]]*(vmx[iip[j+1]]-vmx[i+1]))
        elif i>iip[j] and i<iip[j+1]:
            ar=0.5*(vmy[i]*(vmx[i-1]-vmx[i+1]))
        elif i==iip[j+1]:
            ar=0.5*(vmy[iip[j+1]]*(vmx[i-1]-vmx[iip[j]]))
        area.append(ar)
    areasum=math.fabs(sum(area))
    area[:]=[]
```

```
Areabend.append(areasum)

# Teleutaio bend
area=[]
for i in range(iip[len(iip)-1], len(vmid)):
    if i==iip[len(iip)-1]:
        ar=0.5*(vmy[iip[len(iip)-1]]*(vmx[len(vmid)-1]-vmx[i+1]))
    elif i>iip[len(iip)-1] and i<len(vmid)-1:
        ar=0.5*(vmy[i]*(vmx[i-1]-vmx[i+1]))
    elif i==len(vmid)-1:
        ar=0.5*(vmy[len(vmid)-1]*(vmx[i-1]-vmx[iip[len(iip)-1]))
    area.append(ar)

areasum=math.fabs(sum(area))
Areabend.append(areasum)

# Entopismos Koryfwn Bends sti Grammi Gauss (ALR method)
# Ypologismos apostasewn metaxisimeiwvn
Sbp=[]
for i in range(len(vmid)-1):
    Sbps=math.sqrt(((vmx[i+1]-vmx[i])**2)+((vmy[i+1]-vmy[i])**2))
    Sbp.append(Sbps)

# Koryfes bends
CPxbend=[]
CPybend=[]
CPidbend=[]

# ----- Prwto bend -----
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

if iip[0]+1==3: # bend 3 simeiwvn
    CPnum=vmid[0]
elif iip[0]+1==4: # bend 4 simeiwvn
    for i in range(0, iip[0]+1):
        if i in range(0,1):
            llr1=1
        elif i in range(1,iip[0]+1-1):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-1])**2))
            llr1=l1/s1
        elif i in range(iip[0]+1-1,iip[0]+1):
            llr1=1
        LR1.append(llr1)
        for t in range(len(LR1)):
            alr=LR1[t]
        ALR.append(alr)
        for v in range(len(ALR)):
            if ALR[v]==max(ALR):
                CPnum=v
elif iip[0]+1==5 or iip[0]+1==6: # bend 5 'h 6 simeiwvn
    for i in range(0, iip[0]+1):
        if i in range(0,2):
```

```

        llr1=1
        l2=Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
        llr2=l2/s2
    elif i in range(2,iip[0]+1-2):
        l1=Sbp[i-1]+Sbp[i]
        s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))

        llr1=l1/s1
        l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))

        llr2=l2/s2
    elif i in range(iip[0]+1-2,iip[0]+1):
        llr1=1
        l2=Sbp[i-1]+Sbp[i-2]
        s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
        llr2=l2/s2
    LR1.append(llr1)
    LR2.append(llr2)
    for t in range(len(LR1)):
        alr=(LR1[t]+LR2[t])/2
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v
elif iip[0]+1==7 or iip[0]+1==8: # bend 7 h' 8 simeiw
    for i in range(0, iip[0]+1):
        if i in range(0,3):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
        elif i in range(3,iip[0]+1-3):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))

            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))

            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))

            llr3=l3/s3
        elif i in range(iip[0]+1-3,iip[0]+1):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
    LR1.append(llr1)

```



```

LR2.append(llr2)
LR3.append(llr3)
for t in range(len(LR1)):
    alr=(LR1[t]+LR2[t]+LR3[t])/3
ALR.append(alr)
for v in range(len(ALR)):
    if ALR[v]==max(ALR):
        CPnum=v
elif iip[0]+1>=9: # bend tou laxiston 9 simeiwon
    for i in range(0, iip[0]+1):
        if i in range(0,4):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i])**2)+((vmy[i+4]-vmy[i])**2))
            llr4=l4/s4
        elif i in range(4,iip[0]+1-4):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))
            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))
            llr3=l3/s3
            l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i-4])**2)+((vmy[i+4]-vmy[i-
4])**2))
            llr4=l4/s4
        elif i in range(iip[0]+1-4,iip[0]+1):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
            s4=math.sqrt(((vmx[i-4]-vmx[i])**2)+((vmy[i-4]-vmy[i])**2))
            llr4=l4/s4
LR1.append(llr1)
LR2.append(llr2)
LR3.append(llr3)
LR4.append(llr4)
for t in range(len(LR1)):
    alr=(LR1[t]+LR2[t]+LR3[t]+LR4[t])/4
ALR.append(alr)
for v in range(len(ALR)):

```

```

        if ALR[v]==max(ALR):
            CPnum=v

CPxbend.append(vmx[CPnum])
CPybend.append(vmy[CPnum])
CPidbend.append(vmid[CPnum])

# ----- Endiamesa bends -----
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

# Ypologismosarithmousimeiwkathe bend
n=[]
for t in range(len(iip)-1):
    for i in range(iip[0], iip[1]+1):
        nn=(iip[1]+1)-(iip[0])
        n.append(nn)
        iip[0:1]=[]

# Epanadimiourgiaiip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# Ypologismos LR1, LR2, LR3, LR4
for t in range(len(n)):
    if n[0]==3: # bend 3 simeiw
        for i in range(iip[0], iip[1]+1):
            if i==iip[0]:
                llr1=1 # (afthaireti timi, elaxisti)
            elif i==iip[0]+1:
                llr1=1.5
            elif i==iip[1]:
                llr1=1
            LR1.append(llr1)
        iip[0:1]=[]
        n[0:1]=[]
    elif n[0]==4: # bend 4 simeiw
        for i in range(iip[0], iip[1]+1):
            if i in range(iip[0], iip[0]+1):
                llr1=1
            elif i in range(iip[0]+1, iip[1]+1-1):
                ll=Sbp[i-1]+Sbp[i]
                s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))
                llr1=ll/s1
            elif i in range(iip[1]+1-1, iip[1]+1):
                llr1=1
            LR1.append(llr1)
        iip[0:1]=[]
        n[0:1]=[]
    elif n[0]==5 or n[0]==6: # bend 5 'h 6 simeiw
        for i in range(iip[0], iip[1]+1):
            if i in range(iip[0], iip[0]+2):

```

```

        llr1=1
        l2=Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-
vmy[i])**2))
        llr2=l2/s2
    elifi in range(iip[0]+2, iip[1]+1-2):
        l1=Sbp[i-1]+Sbp[i]
        s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))
        llr1=l1/s1
        l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-
vmy[i-2])**2))
        llr2=l2/s2
    elifi in range(iip[1]+1-2, iip[1]+1):
        llr1=1
        l2=Sbp[i-1]+Sbp[i-2]
        s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-
vmy[i])**2))
        llr2=l2/s2
        LR1.append(llr1)
        LR2.append(llr2)
    iip[0:1]=[ ]
    n[0:1]=[ ]
    elifn[0]==7 or n[0]==8: # bend 7 h' 8 simeiw
        for i in range(iip[0], iip[1]+1):
            if i in range(iip[0], iip[0]+3):
                llr1=1
                l2=Sbp[i]+Sbp[i+1]
                s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-
vmy[i])**2))
                llr2=l2/s2
                l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
                s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-
vmy[i])**2))
                llr3=l3/s3
            elifi in range(iip[0]+3, iip[1]+1-3):
                l1=Sbp[i-1]+Sbp[i]
                s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))
                llr1=l1/s1
                l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
                s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-
vmy[i-2])**2))
                llr2=l2/s2
                l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
                s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-
vmy[i-3])**2))
                llr3=l3/s3
            elifi in range(iip[1]+1-3, iip[1]+1):
                llr1=1
                l2=Sbp[i-1]+Sbp[i-2]
                s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-
vmy[i])**2))
                llr2=l2/s2
                l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
                s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-
vmy[i])**2))

```

```

        l1r3=13/s3
        LR1.append(l1r1)
        LR2.append(l1r2)
        LR3.append(l1r3)
    iip[0:1]=[
    n[0:1]=[
elif n[0]>=9: # bend tou laxiston 9 simeiw n
    for i in range(iip[0], iip[1]+1):
        if i in range(iip[0], iip[0]+4):
            l1r1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-
vmy[i])**2))

            l1r2=12/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-
vmy[i])**2))

            l1r3=13/s3
            l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i])**2)+((vmy[i+4]-
vmy[i])**2))

            l1r4=14/s4
        elif i in range(iip[0]+4, iip[1]+1-4):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-
vmy[i-1])**2))

            l1r1=11/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-
vmy[i-2])**2))

            l1r2=12/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-
vmy[i-3])**2))

            l1r3=13/s3
            l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i-4])**2)+((vmy[i+4]-
vmy[i-4])**2))

            l1r4=14/s4
        elif i in range(iip[1]+1-4, iip[1]+1):
            l1r1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-
vmy[i])**2))

            l1r2=12/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-
vmy[i])**2))

            l1r3=13/s3
            l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
            s4=math.sqrt(((vmx[i-4]-vmx[i])**2)+((vmy[i-4]-
vmy[i])**2))

            l1r4=14/s4
        LR1.append(l1r1)
        LR2.append(l1r2)
        LR3.append(l1r3)
        LR4.append(l1r4)

```

```
iip[0:1]=[]
n[0:1]=[]

# Epanadimiourgia iip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# Epanypologismos arithmousimeiwnekathe bend
n=[]
for t in range(len(iip)-1):
    for i in range(iip[0], iip[1]+1):
        nn=(iip[1]+1)-(iip[0])
    n.append(nn)
    iip[0:1]=[]

# Ypologismos ALR
for t in range(len(n)): #(epanalipseis oses to plithostwn endiameswn bends)
    if n[t]==3: # bend 3 simeiwnekathe
        for i in range(0, n[t]):
            alr=LR1[i]
            ALR.append(alr)
        LR1[0:n[t]]=[]
    elif n[t]==4: # bend 4 simeiwnekathe
        for i in range(0, n[t]):
            alr=LR1[i]
            ALR.append(alr)
        LR1[0:n[t]]=[]
    elif n[t]==5 or n[t]==6: # bend 5 'h 6 simeiwnekathe
        for i in range(0, n[t]):
            alr=(LR1[i]+LR2[i])/2
            ALR.append(alr)
        LR1[0:n[t]]=[]
        LR2[0:n[t]]=[]
    elif n[t]==7 or n[t]==8: # bend 7 'h 8 simeiwnekathe
        for i in range(0, n[t]):
            alr=(LR1[i]+LR2[i]+LR3[i])/3
            ALR.append(alr)
        LR1[0:n[t]]=[]
        LR2[0:n[t]]=[]
        LR3[0:n[t]]=[]
    elif n[t]>=9: # bend tou laxiston 9 simeiwnekathe
        for i in range(0, n[t]):
            alr=(LR1[i]+LR2[i]+LR3[i]+LR4[i])/4
            ALR.append(alr)
        LR1[0:n[t]]=[]
        LR2[0:n[t]]=[]
        LR3[0:n[t]]=[]
        LR4[0:n[t]]=[]

# Entopismos 'i' koryfwn (arithmisientostou bend)
CPnum=[] #('i' apo tin arxithou bend)
ALRassist=[]
for t in range(len(n)):
    for j in range(0, n[t]):
        ALRassist.append(ALR[j])
    for i in range(len(ALRassist)):
```

```
        if ALRassist[i]==max(ALRassist):
            CPnum.append(i)
    ALRassist[:]=[]
    ALR[0:n[t]]=[]

# Epanadimiourgiaiip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# Ypologismosx,y,idkoryfwn bends
# (koryfi == x,y,id['i' touarxikousimeioukathe bend + CPnum])
for t in range(len(n)):
    CPxbend.append(vmx[iip[0]+CPnum[0]])
    CPybend.append(vmy[iip[0]+CPnum[0]])
    CPidbend.append(vmid[iip[0]+CPnum[0]])
    iip[0:1]=[]
    CPnum[0:1]=[]

# Epanadimiourgiaiip
iip=[]
for i in range(len(vmid)):
    if type(vmid[i])==float:
        iip.append(i)

# ----- Teleutaio bend -----
LR1=[]
LR2=[]
LR3=[]
LR4=[]
ALR=[]

if len(vmid)-(iip[len(iip)-1])==3: # bend 3 simeiw
    CPnum=1
elif len(vmid)-(iip[len(iip)-1])==4: # bend 4 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+1):
            llr1=1
        elif i in range(iip[len(iip)-1]+1, len(vmid)-1):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-1])**2))
            llr1=l1/s1
        elif i in range(len(vmid)-1, len(vmid)):
            llr1=1
    LR1.append(llr1)
    for t in range(len(LR1)):
        alr=LR1[t]
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v
elif len(vmid)-(iip[len(iip)-1])==5 or len(vmid)-(iip[len(iip)-1])==6: # bend
5 'h 6 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+2):
            llr1=1
```

```

        l2=Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
        llr2=l2/s2
    elif i in range(iip[len(iip)-1]+2, len(vmid)-2):
        l1=Sbp[i-1]+Sbp[i]
        s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))

        llr1=l1/s1
        l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
        s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))

        llr2=l2/s2
    elif i in range(len(vmid)-2, len(vmid)):
        llr1=1
        l2=Sbp[i-1]+Sbp[i-2]
        s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
        llr2=l2/s2
    LR1.append(llr1)
    LR2.append(llr2)
    for t in range(len(LR1)):
        alr=(LR1[t]+LR2[t])/2
    ALR.append(alr)
    for v in range(len(ALR)):
        if ALR[v]==max(ALR):
            CPnum=v
elif len(vmid)-(iip[len(iip)-1])==7 or len(vmid)-(iip[len(iip)-1])==8: # bend
7 h' 8 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+3):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
        elif i in range(iip[len(iip)-1]+3, len(vmid)-3):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))

            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))

            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))

            llr3=l3/s3
        elif i in range(len(vmid)-3, len(vmid)):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
    LR1.append(llr1)

```

```

LR2.append(llr2)
LR3.append(llr3)
for t in range(len(LR1)):
    alr=(LR1[t]+LR2[t]+LR3[t])/3
ALR.append(alr)
for v in range(len(ALR)):
    if ALR[v]==max(ALR):
        CPnum=v
elif len(vmid)-(iip[len(iip)-1])>=9: # bend toulaxiston 9 simeiw
    for i in range(iip[len(iip)-1], len(vmid)):
        if i in range(iip[len(iip)-1], iip[len(iip)-1]+4):
            llr1=1
            l2=Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i])**2)+((vmy[i+2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i])**2)+((vmy[i+3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i])**2)+((vmy[i+4]-vmy[i])**2))
            llr4=l4/s4
        elif i in range(iip[len(iip)-1]+4, len(vmid)-4):
            l1=Sbp[i-1]+Sbp[i]
            s1=math.sqrt(((vmx[i+1]-vmx[i-1])**2)+((vmy[i+1]-vmy[i-
1])**2))
            llr1=l1/s1
            l2=Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]
            s2=math.sqrt(((vmx[i+2]-vmx[i-2])**2)+((vmy[i+2]-vmy[i-
2])**2))
            llr2=l2/s2
            l3=Sbp[i-3]+Sbp[i-2]+Sbp[i-1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]
            s3=math.sqrt(((vmx[i+3]-vmx[i-3])**2)+((vmy[i+3]-vmy[i-
3])**2))
            llr3=l3/s3
            l4=Sbp[i-4]+Sbp[i-3]+Sbp[i-2]+Sbp[i-
1]+Sbp[i]+Sbp[i+1]+Sbp[i+2]+Sbp[i+3]
            s4=math.sqrt(((vmx[i+4]-vmx[i-4])**2)+((vmy[i+4]-vmy[i-
4])**2))
            llr4=l4/s4
        elif i in range(len(vmid)-4, len(vmid)):
            llr1=1
            l2=Sbp[i-1]+Sbp[i-2]
            s2=math.sqrt(((vmx[i-2]-vmx[i])**2)+((vmy[i-2]-vmy[i])**2))
            llr2=l2/s2
            l3=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]
            s3=math.sqrt(((vmx[i-3]-vmx[i])**2)+((vmy[i-3]-vmy[i])**2))
            llr3=l3/s3
            l4=Sbp[i-1]+Sbp[i-2]+Sbp[i-3]+Sbp[i-4]
            s4=math.sqrt(((vmx[i-4]-vmx[i])**2)+((vmy[i-4]-vmy[i])**2))
            llr4=l4/s4
        LR1.append(llr1)
        LR2.append(llr2)
        LR3.append(llr3)
        LR4.append(llr4)
        for t in range(len(LR1)):
            alr=(LR1[t]+LR2[t]+LR3[t]+LR4[t])/4
        ALR.append(alr)

```



```

        for v in range(len(ALR)):
            if ALR[v]==max(ALR):
                CPnum=v

CPxbend.append(vmx[iip[len(iip)-1]+CPnum]) #(koryfi == x,y,id['i'
touarxikousimeiououteleutaiau bend + CPnum])
CPybend.append(vmy[iip[len(iip)-1]+CPnum])
CPidbend.append(vmid[iip[len(iip)-1]+CPnum])

# ===== EntopismosKoryfwn Bends stinArxikiGrammi =====
iniCPxbend=[]
iniCPybend=[]
iniCPidbend=[]
for i in range(len(CPidbend)):
    iniCPxbend.append(inix[CPidbend[i]-1]) # antistoixisi id koryfwn bends
grammis Gauss me simeiastinarxikigrammi
    iniCPybend.append(iniy[CPidbend[i]-1])
    iniCPidbend.append(iniid[CPidbend[i]-1])

# ===== YpologismosVathous Bends (apostasikoryfis apo grammivasis)
Wbend=[]
# Ptwto bend
d=(math.fabs(((-(vmy[iip[0]]-vmy[0])/(vmx[iip[0]]-
vmx[0]))) *CPxbend[0])+CPybend[0]+((vmy[iip[0]]-vmy[0])/(vmx[iip[0]]-
vmx[0])) *vmx[0]) -vmy[0]))/(math.sqrt(((-(vmy[iip[0]]-vmy[0])/(vmx[iip[0]]-
vmx[0])) **2)+1))
Wbend.append(d)

# Endiamesa bends
for i in range(len(iip)-1):
    d=(math.fabs(((-(vmy[iip[i+1]]-vmy[iip[i]])/(vmx[iip[i+1]]-
vmx[iip[i]]))) *CPxbend[i+1])+CPybend[i+1]+((vmy[iip[i+1]]-
vmy[iip[i]])/(vmx[iip[i+1]]-vmx[iip[i])) *vmx[iip[i]])-
vmy[iip[i]]))/(math.sqrt(((-(vmy[iip[i+1]]-vmy[iip[i]])/(vmx[iip[i+1]]-
vmx[iip[i]]))) **2)+1))
    Wbend.append(d)

# Teleutaio bend
d=(math.fabs(((-(vmy[len(vmid)-1]-vmy[iip[len(iip)-1]])/(vmx[len(vmid)-1]-
vmx[iip[len(iip)-1]]))) *CPxbend[len(CPidbend)-1])+CPybend[len(CPidbend)-
1]+((vmy[len(vmid)-1]-vmy[iip[len(iip)-1]])/(vmx[len(vmid)-1]-
vmx[iip[len(iip)-1]])) *vmx[iip[len(iip)-1]])-vmy[iip[len(iip)-
1]]))/(math.sqrt(((-(vmy[len(vmid)-1]-vmy[iip[len(iip)-1]])/(vmx[len(vmid)-
1]-vmx[iip[len(iip)-1]]))) **2)+1))
Wbend.append(d)

# ===== ProsdiorismosKampylotitas Bends =====
Curvbend=[]

# Euresigwniwnthlasispouantistoixounstiskoryfes tis grammis
CurvCP=[]
for i in range(len(id)):
    for j in range(len(CPidbend)):
        if CPidbend[j]==id[i]:
            CurvCP.append(sl[i-1]) # euresigwniasthlasispouantistoixei
se kathekoryfi

for i in range(len(CurvCP)): # prosdiorismoskampylotitas

```

```
    if CurvCP[i]>0:
        Curvbend.append(1)
    elifCurvCP[i]<0:
        Curvbend.append(-1)

# ===== YpologismosArithmouSimeiwN Bends =====
NumPbend=[]
NumPbend.append(iip[0]+1) # ArithmossimeiwNprwtou bend
for i in range(len(n)):
    NumPbend.append(n[i])

NumPbend.append(len(vmid)-(iip[len(iip)-1])) # ArithmossimeiwNteleutaioubend

# =====
# ===== ExagwgiArxeiwn =====
# =====

# ----- Apodektes Times 'sigma' -----
gout=open('C:SigmaAccept.txt', 'w')
for i in range(len(sigmaAccept)-1): # (stoixeiaektostousigmaMax)
    gout.write(str(sigmaAccept[i])+' '+'\n')

gout.write('sigmaMax:'+str(sigmaMax))
gout.close()

# ----- Bends kai Idiotites -----
fout=open('C:Bends and Attributes.txt', 'w')
for i in range(len(CPidbend)):
    fout.write('Bend'+str(i+1)+' '+' '+'LengthBL:'+(str(LBLbend[i]))+'
'+ 'Area:'+(str(Areabend[i]))+' '+'Depth:'+(str(Wbend[i]))+'
'+ 'Curvature:'+(str(Curvbend[i]))+' '+'NumOfPoints:'+(str(NumPbend[i]))+' '\n')

fout.close()

# ----- Grammi Gauss me Bends -----
rout=open('C:SmoothPartBends.txt', 'w')
rout.write(str('Polyline')+'\n')
for i in range(len(vmid)):
    rout.write(str(vmid[i])+' '+'+(str(vmx[i]))+' '+'+(str(vmy[i]))+' '\n')

rout.write(str('END'))
rout.close()

# ----- Koryfes Bends stiGrammi Gauss -----
uout=open('C:BendsCP.txt', 'w')
uout.write(str('Point')+'\n')
for i in range(len(CPidbend)):
    uout.write(str(CPidbend[i])+' '+'+(str(CPxbend[i]))+'
'+(str(CPybend[i]))+' '\n')

uout.write(str('END'))
uout.close()

# ----- Koryfes Bends stinArxikiGrammi -----
oout=open('C:iniBendsCP.txt', 'w')
oout.write(str('Point')+'\n')
for i in range(len(CPidbend)):
```

```
        out.write(str(iniCPidbend[i])+ ' '(str(iniCPxbend[i]))+'
'+(str(iniCPybend[i]))+'\n')

out.write(str('END'))
out.close()

# ----- Kampyloptita Bends Grammis Gauss -----
tout=open('C:CurvBend.txt', 'w')
for i in range(len(CPidbend)):
    tout.write(str(Curvbend[i])+'\n')

tout.close()

# ----- Grammi Gauss -----
mout=open('C:SmoothPart.txt', 'w')
mout.write(str('Polyline')+'\n')
for i in range(len(id)):
    mout.write(str(id[i])+ ' '(str(x[i]))+' '(str(y[i]))+'\n')

mout.write(str('END'))
mout.close()

m.close()
f.close()
```

### Διόγκωση με δεσμεύσεις

```
import arcgisscripting
gp = arcgisscripting.create(9.3)

# ===== EisagwgiDedomenwn =====
# ----- EisagwgiGrammis Gauss Tmimatos me Bends -----
f = open('C:SmoothPartBends.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in f:
    NumLines=NumLines+1

f = open('C:SmoothPartBends.txt') # Anoigmaarxeiou

# Eisagogisimeiwngrammis SPB(x,y,id)
SPBx=[]
SPBy=[]
SPBid=[]
mtr=0
for line in f:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        SPBid.append(float(line[:-1].split(' ')[0]))
        SPBx.append(float(line[:-1].split(' ')[1]))
        SPBy.append(float(line[:-1].split(' ')[2]))

# Metatropitwn float id se integer
for i in range(len(SPBid)):
    if float(SPBid[i])!=int(SPBid[i]): # Epilogi float stoiceiwn (endiamesa
inflection points)
        SPBid[i]=SPBid[i]
    else: # Epilogi int stoiceiwn...
        SPBid[i]=int(SPBid[i]) # ...kai metatropi se integer
```

```
# ----- EisagwgiKoryfwn Bends Grammis Gauss -----
g = open('C:BendsCP.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in g:
    NumLines=NumLines+1

g = open('C:BendsCP.txt') # Anoigmaarxeiou

# EisagwgikoryfwnSPB(korx, kory, korid)
SPBkorx=[]
SPBkory=[]
SPBkorid=[]
mtr=0
for line in g:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        SPBkorid.append(int(line[:-1].split(' ')[0]))
        SPBkorx.append(float(line[:-1].split(' ')[1]))
        SPBkory.append(float(line[:-1].split(' ')[2]))

# ----- EisagwgiKoryfwn Bends ArxikisGrammis -----
o = open('C:iniBendsCP.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in o:
    NumLines=NumLines+1

o = open('C:iniBendsCP.txt') # Anoigmaarxeiou

# EisagwgikoryfwnSPB(inikorx, inikory, inikorid)
SPBinikorx=[]
SPBinikory=[]
SPBinikorid=[]
mtr=0
for line in o:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        SPBinikorid.append(int(line[:-1].split(' ')[0]))
        SPBinikorx.append(float(line[:-1].split(' ')[1]))
        SPBinikory.append(float(line[:-1].split(' ')[2]))

# ----- Eisagwgi e -----
m = open('C:Table\e.txt')
e=[]
for line in m:
    e.append(float(line[:]))

# ----- Eisagwgi ID Exetazomenou Bend -----
h = open('C:BendID.txt')
BendID=[]
for line in h:
    BendID.append(int(line[:]))

# ===== ApomonwsiExetazomenou Bend =====
# 'i' exetazomenou bend
SelB=BendID[0]-1

# Ypologismos 'i' tw'n inflection points stigrammi Gauss
```

```
iip=[]
for i in range(len(SPBid)):
    if type(SPBid[i])==float:
        iip.append(i)

# 'i' inflection point se periptwsimonou bend
if iip==[]:
    iip.append(len(SPBid)-1)
else:
    iip==iip

# Dimiourgiapinakwn me x,y,idexetazomenou bend
x=[]
y=[]
id=[]
if SelB==0: # Prwto bend
    for i in range(0, iip[SelB]+1):
        x.append(SPBx[i])
        y.append(SPBy[i])
        id.append(SPBid[i])
elif SelB>=1 and SelB<len(SPBkorid)-1: # Endiamesa bends
    for i in range(iip[SelB-1], iip[SelB]+1):
        x.append(SPBx[i])
        y.append(SPBy[i])
        id.append(SPBid[i])
elif SelB==len(SPBkorid)-1: # Teleutaio bend
    for i in range(iip[SelB-1], len(SPBid)):
        x.append(SPBx[i])
        y.append(SPBy[i])
        id.append(SPBid[i])

# Dimiourgiapinakwn me x,y,idkoryfisexetazomenou bend kor(x,y,id)
korx=SPBkorx[SelB] # epilogikoryfispouantistoixeistoexetazomeno bend
kory=SPBkory[SelB]
korid=SPBkorid[SelB]

# Dimiourgiapinakwn me x,y,idkoryfisexetazomenou bend
stinarxikigrammiinikor(x,y,id)
inikorx=SPBinikorx[SelB] # epilogikoryfispouantistoixeistoexetazomeno bend
inikory=SPBinikory[SelB]
inikorid=SPBinikorid[SelB]

# -----
# Dimiourgiapinakwn me Inflection Points kai Koryfiexetazomenou bend
xp=[]
yp=[]
xp.append(x[0])
xp.append(korx)
xp.append(x[len(id)-1])
yp.append(y[0])
yp.append(kory)
yp.append(y[len(id)-1])

# Dimiourgiapinakwn me Inflection Points kai Koryfiexetazomenou bend
stinarxikigrammi
xpa=[]
ypa=[]
xpa.append(x[0])
```

```
xpa.append(inikorx)
xpa.append(x[len(id)-1])
ypa.append(y[0])
ypa.append(inikory)
ypa.append(y[len(id)-1])
# -----

# ----- EisagwgiProsimouKampylofitas Bends -----
t = open('C:CurvBend.txt')
CurvatureBend=[]
for line in t:
    CurvatureBend.append(int(line[:]))

# ProximoKampylofitasexetazomenou bend
CurvBend=CurvatureBend[SelB] #
epilogiproximoukampylofitaspouantistoiixeistoexetazomeno bend

# -----
import math
# -----

# =====
# ===== Diogwsi Bend =====
# =====

# ----- Ypologismossyntelestwn u kai v -----
# Ypologismos dx kai dygia ta simeia 2 to n-1
dx=[]
dy=[]
for i in range(1, len(id)-1):
    ddx=x[i+1]-x[i-1]
    ddy=y[i+1]-y[i-1]
    dx.append(ddx)
    dy.append(ddy)

# Ypologismosgniasdiefthinsiseftheias n-1---n+1 giakathesimeio
w=[]
for i in range(len(dx)):
    if dx[i]>0 and dy[i]>0:
        ww=((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
        elif dx[i]>0 and dy[i]<0:
            ww= 200-
            ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
            elif dx[i]<0 and dy[i]<0:
                ww=
                200+((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
                elif dx[i]<0 and dy[i]>0:
                    ww= 400-
                    ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
                    elif dx[i]==0 and dy[i]>0:
                        ww=0
                    elif dx[i]==0 and dy[i]<0:
                        ww=200
                    elif dx[i]>0 and dy[i]==0:
                        ww=100
                    elif dx[i]<0 and dy[i]==0:
                        ww=300
```

```
w.append(ww)

# Ypologismos diefthinsismetatopisisshmeiwn (kathetistin w)
wm=[]
for i in range(len(w)): # exetasiperiptwsewn:
    if CurvBend==1: # to prosimokampyloitas tis grammiseinai +
        wwm=w[i]-100
    elif CurvBend==-1: # to prosimokampyloitas tis grammiseinai -
        wwm=w[i]+100
    wm.append(wwm)

# Elegxos giapithaniafaireisi h' prosthesi 400g
wmf=[]
for i in range(len(wm)):
    if wm[i]>400:
        wwmf=wm[i]-400
    elif wm[i]<0:
        wwmf=wm[i]+400
    else:
        wwmf=wm[i]
    wmf.append(wwmf)

# Metatropi grad se rad
wmfr=[]
for i in range(len(wmf)):
    wwmfr=(math.pi*wmf[i])/200
    wmfr.append(wwmfr)

# Ypologismos syntelestwn u kai v
u=[]
for i in range(len(wmfr)):
    uu=math.sin(wmfr[i])
    u.append(uu)

v=[]
for i in range(len(wmfr)):
    vv=math.cos(wmfr[i])
    v.append(vv)

# ----- Ypologismos S/Sn -----
# Ypologismos S simeiwon 1 ws Koryfi
# Ypologismos apostasewn metaxisimeiwon
Sbp1K=[]
for i in range(0, int(korid-(id[0]-0.5))): # (id[int(korid-(id[0]-
0.5))]==korid & int(korid-(id[0]-0.5))==korid-1)
    Sbps1K=math.sqrt(((x[i+1]-x[i])**2)+(y[i+1]-y[i])**2)
    Sbp1K.append(Sbps1K)

# Ypologismos S kathesimeiou
S1K=[]
for i in range(len(Sbp1K)):
    if i==0:
        sslk=0+Sbp1K[i]
    else:
        sslk=sslk+Sbp1K[i]
    S1K.append(sslk)

# Ypologismos Sn apo 1 ws Koryfi
```

```

Sn1K=sum(Sbp1K, 0)

# Ypologismos S simeiw n wsKoryfi
# Ypologismosapostasewnmetaxisimeiw n
SbpNK=[]
for i in reversed(range(int(korid-(id[0]-0.5)), len(id)-1)):
    SbpsNK=math.sqrt(((x[i]-x[i+1])**2)+(y[i]-y[i+1])**2)
    SbpNK.append(SbpsNK)

# Ypologismos S kathesimeiou
SNK=[]
for i in range(len(SbpNK)):
    if i==0:
        sskn=0+SbpNK[i]
    else:
        sskn=sskn+SbpNK[i]
    SNK.append(sskn)

# Ypologismos Sn apo n wsKoryfi
SnNK=sum(SbpNK, 0)

# Ypologismos S/Sn (S1K/Sn1K) apo 2 wsKoryfi
Sf1K=[]
for i in range(len(S1K)):
    sf1k=S1K[i]/Sn1K
    Sf1K.append(sf1k)

# Ypologismos S/Sn (SNK/SnNK) apo n-1 wsKoryfi
SfNK=[]
for i in range(len(SNK)):
    sfnk=SNK[i]/SnNK
    SfNK.append(sfnk)

# ----- Ypologismos Inflating -----
# Ypologismoskentroukyklou C (metaxiarxis-peratos)
xc=(x[0]+x[len(id)-1])/2
yc=(y[0]+y[len(id)-1])/2

# Ypologismosapostasiskentroukyklou C kai CP grammis Gauss
CCP=math.sqrt(((xc-korx)**2)+(yc-kory)**2)

# Ypologismos Inflating analoga an CP ektos, panw h' ektoskyklou
if CCP>e[0]/2:
    Inflating=math.sqrt(((korx-xpa[1])**2)+(kory-ypa[1])**2) #
Ypologismos Inflating (apostasi CP apo CParxiko)
elif CCP==e[0]/2:
    Inflating=math.sqrt(((korx-xpa[1])**2)+(kory-ypa[1])**2) #
Ypologismos Inflating (apostasi CP apo CParxiko)
elif CCP<e[0]/2:
    l=-((x[int(korid-(id[0]-0.5))+1]-x[int(korid-(id[0]-0.5))-1])/
(y[int(korid-(id[0]-0.5))+1]-y[int(korid-(id[0]-0.5))-1])) #
Ypologismossyntelestidiefthinsis (l) eftheiasmetatopisiskoryfis
    D=(((-2*xc)-(2*(1**2)*(korx)+(2*1*kory)-(2*yc*1)**2)-
(4*((1**2)+1)*((xc**2)+(yc**2)-(e[0]**2)/4)+(1**2)*(korx**2))-
(2*1*korx*kory)+(kory**2)+(2*yc*1*korx)-(2*yc*kory))) #
YpologismosDiakrinousas D

```



```

        xt1=((2*xc)+(2*(1**2)*korx)-
(2*1*kory)+(2*yc*1))+(math.sqrt(D))/((2*(1**2))+2) #
Ypologismospithanwnnlisewn t1,t2
        yt1=(1*xt1)-(1*korx)+kory
        xt2=((2*xc)+(2*(1**2)*korx)-(2*1*kory)+(2*yc*1))-
(math.sqrt(D))/((2*(1**2))+2)
        yt2=(1*xt2)-(1*korx)+kory
        ap1=math.sqrt(((korx-xt1)**2)+((kory-yt1)**2)) # Ypologismosapostasewn
CP-t1 (ap1) kai CP-t2 (ap2)
        ap2=math.sqrt(((korx-xt2)**2)+((kory-yt2)**2))
        if ap1>ap2: # Entopismosapodektis lysis systimatos (to
xtpouapexeimikroteriapostasi apo to CP)
            xtomi=xt2
            ytomi=yt2
        elif ap1<ap2:
            xtomi=xt1
            ytomi=yt1
        Inflating=math.sqrt(((korx-xtomi)**2)+((kory-ytomi)**2)) # Ypologismos
Inflating (apostasi CP apo tomi)

# ----- YpologismosXnew, Ynew Balloon -----
# YpologismosXnew apo 1 wsKoryfi
X1K=[]
for i in range(0, int(korid-(id[0]-0.5))+1):
    if i==0:
        x1k=x[i]
    else:
        x1k=x[i]+(Inflating*u[i-1]*Sf1K[i-1])
    X1K.append(x1k)

# YpologismosYnew apo 1 wsKoryfi
Y1K=[]
for i in range(0, int(korid-(id[0]-0.5))+1):
    if i==0:
        y1k=y[i]
    else:
        y1k=y[i]+(Inflating*v[i-1]*Sf1K[i-1])
    Y1K.append(y1k)

# Dimiourgiavoithitikwnpinakwnpouperilamvanoun mono ta simeia N ws K
xNKassist=[]
for i in reversed(range(int(korid-(id[0]-0.5)), len(id))):
    xNKassist.append(x[i])

yNKassist=[]
for i in reversed(range(int(korid-(id[0]-0.5)), len(id))):
    yNKassist.append(y[i])

# Antistrofiarxeiwn u kai v gianaxrisimopoihthounstonypologismoX,Ytwn N ws K
u.reverse()
v.reverse()

# YpologismosXnew apo N wsKoryfi
XNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        xnk=xNKassist[i]
    else:

```

```
xnk=xNKassist[i]+(Inflating*u[i-1]*SfNK[i-1])
XNKnew.append(xnk)

# Antistrofi XNK se XKN
XNKnew.reverse()
# Diagrafi lou stoixeiou
XNKnew.remove(XNKnew[0])

# YpologismosYnew apo N wsKoryfi
YNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        ynk=yNKassist[i]
    else:
        ynk=yNKassist[i]+(Inflating*v[i-1]*SfNK[i-1])
    YNKnew.append(ynk)

# Antistrofi YNK se YKN
YNKnew.reverse()
# Diagrafi lou stoixeiou (dil. tou Y tou K pouyparxei kai sto Y1K)
YNKnew.remove(YNKnew[0])

# ===== X kai Y Diogwmenou Bend =====
# Enwsipinakwn
Xfinal=[]
Xfinal=X1K+XNKnew

Yfinal=[]
Yfinal=Y1K+YNKnew

# =====
# ===== AfinikosMetasximatismos =====
# ===== Telika X kai Y =====
# =====

Xf=[]
Yf=[]
IDf=[]

# Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
a=(Yfinal[int(korid-(id[0]-0.5))]*xpa[0])-(Yfinal[len(id)-1]*xpa[0])+(Yfinal[len(id)-1]*xpa[1])-(Yfinal[0]*xpa[1])+(Yfinal[0]*xpa[2])-(Yfinal[int(korid-(id[0]-0.5))]*xpa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-0.5))])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
b=(Xfinal[len(id)-1]*xpa[0])-(Xfinal[int(korid-(id[0]-0.5))]*xpa[0])+(Xfinal[0]*xpa[1])-(Xfinal[len(id)-1]*xpa[1])+(Xfinal[int(korid-(id[0]-0.5))]*xpa[2])-(Xfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-0.5))])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
c=(Xfinal[int(korid-(id[0]-0.5))]*Yfinal[len(id)-1]*xpa[0])-(Yfinal[int(korid-(id[0]-0.5))]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-1]*Yfinal[0]*xpa[1])-(Yfinal[len(id)-1]*Xfinal[0]*xpa[1])+(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))]*xpa[2])-(Xfinal[int(korid-(id[0]-0.5))])
```

```
0.5))]*Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))])*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
d=(Yfinal[int(korid-(id[0]-0.5))])*ypa[0])-(Yfinal[len(id)-
1]*ypa[0])+(Yfinal[len(id)-1]*ypa[1])-(Yfinal[0]*ypa[1])+(Yfinal[0]*ypa[2])-(
(Yfinal[int(korid-(id[0]-0.5))])*ypa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-
0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))])*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[int(korid-(id[0]-
0.5))])*ypa[0])+(Xfinal[0]*ypa[1])-(Xfinal[len(id)-
1]*ypa[1])+(Xfinal[int(korid-(id[0]-0.5))])*ypa[2])-(
(Xfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))])*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
f=((Xfinal[int(korid-(id[0]-0.5))])*Yfinal[len(id)-1]*ypa[0])-(
(Yfinal[int(korid-(id[0]-0.5))])*Xfinal[len(id)-1]*ypa[0])+(Xfinal[len(id)-
1]*Yfinal[0]*ypa[1])-(Xfinal[0]*Yfinal[len(id)-
1]*ypa[1])+(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])*ypa[2])-(
(Xfinal[int(korid-(id[0]-
0.5))])*Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))])*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))

# Prwtosimeio (Eisagwgisimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos)
Xf.append(x[0])
Yf.append(y[0])
IDf.append(id[0])

# Endiamesasimeia (Ypologismosnewsintetagmenwn apo
afinikometasximatismotwnsimeiw 2-->n-1)
for i in range(1, len(id)-1):
    xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c
    yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
    Xf.append(xaffine)
    Yf.append(yaffine)
    IDf.append(id[i])

# Teleutaiosimeio (Eisagwgiteleutaiousimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos)
Xf.append(x[len(id)-1])
Yf.append(y[len(id)-1])
IDf.append(id[len(id)-1])

# Exagwgiarxeiou
fout=open('C:GeneralizedTypeBPart.txt', 'w')
fout.write(str('Polyline')+'\n')
for i in range(len(id)):
    fout.write(str(IDf[i])+' '+str(Xf[i])+' '+str(Yf[i])+'\n')

fout.write(str('END'))
fout.close()

g.close()
```

```
m.close()
h.close()
o.close()
t.close()
```

### Διόγκωση χωρίς δεσμεύσεις

```
import arcgisscripting
gp = arcgisscripting.create(9.3)

# ===== EisagwgiDedomenwn =====
# ----- EisagwgiGrammis Gauss Tmimatos me Bends -----
f = open('C:SmoothPartBends.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwon (lines) arxeiou
for line in f:
    NumLines=NumLines+1

f = open('C:SmoothPartBends.txt') # Anoigmaarxeiou

# Eisagogisimeiwngammis SPB(x,y,id)
SPBx=[]
SPBy=[]
SPBid=[]
mtr=0
for line in f:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        SPBid.append(float(line[:-1].split(' ')[0]))
        SPBx.append(float(line[:-1].split(' ')[1]))
        SPBy.append(float(line[:-1].split(' ')[2]))

# Metatropitwn float id se integer
for i in range(len(SPBid)):
    if float(SPBid[i])!=int(SPBid[i]): # Epilogi float stoiceiwn (endiamesa
inflection points)
        SPBid[i]=SPBid[i]
    else: # Epilogi int stoiceiwn...
        SPBid[i]=int(SPBid[i]) # ...kai metatropi se integer

# ----- EisagwgiKoryfwn Bends Grammis Gauss -----
g = open('C:BendsCP.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwon (lines) arxeiou
for line in g:
    NumLines=NumLines+1

g = open('C:BendsCP.txt') # Anoigmaarxeiou

# EisagwgikoryfwnSPB(korx, kory, korid)
SPBkorx=[]
SPBkory=[]
SPBkorid=[]
mtr=0
for line in g:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        SPBkorid.append(int(line[:-1].split(' ')[0]))
        SPBkorx.append(float(line[:-1].split(' ')[1]))
        SPBkory.append(float(line[:-1].split(' ')[2]))
```

```
# ----- EisagwgiKoryfwn Bends ArxikisGrammis -----
o = open('C:iniBendsCP.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammw (lines) arxeiou
for line in o:
    NumLines=NumLines+1

o = open('C:iniBendsCP.txt') # Anoigmaarxeiou

# EisagwgiKoryfwnSPB(inikorx, inikory, inikorid)
SPBinikorx=[]
SPBinikory=[]
SPBinikorid=[]
mtr=0
for line in o:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        SPBinikorid.append(int(line[:-1].split(' ')[0]))
        SPBinikorx.append(float(line[:-1].split(' ')[1]))
        SPBinikory.append(float(line[:-1].split(' ')[2]))

# ----- Eisagwgi e -----
m = open('C:e.txt')
e=[]
for line in m:
    e.append(float(line[:]))

# ----- EisagwgiSyntelestiDiogwsis (Inflating) -----
z = open('C:Inflating.txt')
Inflating=[]
for line in z:
    Inflating.append(float(line[:]))

# ----- Eisagwgi ID Exetazomenou Bend -----
h = open('C:BendID.txt')
BendID=[]
for line in h:
    BendID.append(int(line[:]))

# ===== ApomonwsiExetazomenou Bend =====
# 'i' exetazomenou bend
SelB=BendID[0]-1

# Ypologismos 'i' tw n inflection points stigrammi Gauss
iip=[]
for i in range(len(SPBid)):
    if type(SPBid[i])==float:
        iip.append(i)

# 'i' inflection point se periptwsimonou bend
if iip==[]: # den entopizontai inflection points (dil. type()==float)
    iip.append(len(SPBid)-1) # tithetaiws 'i' auto touteleutaiousimeiou tou
bend
else:
    iip==iip

# Dimiourgiapinakwn me x,y,idexetazomenou bend
x=[]
```

```
y=[]
id=[]
if SelB==0: # Prwto bend
    for i in range(0, iip[SelB]+1):
        x.append(SPbx[i])
        y.append(SPby[i])
        id.append(SPbid[i])
elifSelB>=1 and SelB<len(SPbkorid)-1: # Endiamesa bends
    for i in range(iip[SelB-1], iip[SelB]+1):
        x.append(SPbx[i])
        y.append(SPby[i])
        id.append(SPbid[i])
elifSelB==len(SPbkorid)-1: # Teleutaio bend
    for i in range(iip[SelB-1], len(SPbid)):
        x.append(SPbx[i])
        y.append(SPby[i])
        id.append(SPbid[i])

# Dimiourgiapinakwn me x,y,idkoryfisexetazomenou bend kor(x,y,id)
korx=SPbkorx[SelB] # epilogikoryfispouantistoixeistoexetazomeno bend
kory=SPbkory[SelB]
korid=SPbkorid[SelB]

# Dimiourgiapinakwn me x,y,idkoryfisexetazomenou bend
stinarxikigrammiinikor(x,y,id)
inikorx=SPbinikorx[SelB] # epilogikoryfispouantistoixeistoexetazomeno bend
inikory=SPbinikory[SelB]
inikorid=SPbinikorid[SelB]

# -----
# Dimiourgiapinakwn me Inflection Points kai Koryfiexetazomenou bend
xp=[]
yp=[]
xp.append(x[0])
xp.append(korx)
xp.append(x[len(id)-1])
yp.append(y[0])
yp.append(kory)
yp.append(y[len(id)-1])

# Dimiourgiapinakwn me Inflection Points kai Koryfiexetazomenou bend
stinarxikigrammi
xpa=[]
ypa=[]
xpa.append(x[0])
xpa.append(inikorx)
xpa.append(x[len(id)-1])
ypa.append(y[0])
ypa.append(inikory)
ypa.append(y[len(id)-1])
# -----

# ===== EisagwgiProsimouKampylotitas Bends =====
t = open('C:CurvBend.txt')
CurvatureBend=[]
for line in t:
    CurvatureBend.append(int(line[:]))
```

```
# ProximoKampylotitasexetazomenou bend
CurvBend=CurvatureBend[SelB] #
epilogiprosimoukampylotitaspouantistoixeistoexetazomeno bend

# -----
import math
# -----

# ===== YpologismosXaraktiristikwn Bend =====
# Ypologismoskentroukyklou C (metaxiarxis-peratos)
xc=(x[0]+x[len(id)-1])/2
yc=(y[0]+y[len(id)-1])/2

# Ypologismosapostasiskentroukyklou C kai CP grammis Gauss
CCP=math.sqrt(((xc-korx)**2)+((yc-kory)**2))

# Ypologismosapostasiskentroukyklou C kai CP arxikisgrammis
CCParx=math.sqrt(((xc-xpa[1])**2)+((yc-ypa[1])**2))

# =====
# ===== Diogwsi Bend =====
# =====

# ----- Ypologismossyntelestwn u kai v -----
# Ypologismos dx kai dygia ta simeia 2 to n-1
dx=[]
dy=[]
for i in range(1, len(id)-1):
    ddx=x[i+1]-x[i-1]
    ddy=y[i+1]-y[i-1]
    dx.append(ddx)
    dy.append(ddy)

# Ypologismosgniasdiefthinsiseftheias n-1---n+1 giakathesimeio
w=[]
for i in range(len(dx)):
    if dx[i]>0 and dy[i]>0:
        ww=((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
        elif dx[i]>0 and dy[i]<0:
            ww= 200-
            ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
            elif dx[i]<0 and dy[i]<0:
                ww=
                200+((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
                elif dx[i]<0 and dy[i]>0:
                    ww= 400-
                    ((math.atan((math.fabs(dx[i]))/(math.fabs(dy[i]))))* (200/math.pi))
                    elif dx[i]==0 and dy[i]>0:
                        ww=0
                    elif dx[i]==0 and dy[i]<0:
                        ww=200
                    elif dx[i]>0 and dy[i]==0:
                        ww=100
                    elif dx[i]<0 and dy[i]==0:
                        ww=300
                    w.append(ww)
```

```
# Ypologismos diefthisis metatopisissheimiwn (kathetistin w)
wm=[]
for i in range(len(w)): # exetasiperiptwsewn:
    if CurvBend==1: # to prosimokampylotitas tis grammiseinai +
        wwm=w[i]-100
    elif CurvBend==-1: # to prosimokampylotitas tis grammiseinai -
        wwm=w[i]+100
    wm.append(wwm)

# Elegxos giapithania fairesi h' prosthesi 400g
wmf=[]
for i in range(len(wm)):
    if wm[i]>400:
        wwmf=wm[i]-400
    elif wm[i]<0:
        wwmf=wm[i]+400
    else:
        wwmf=wm[i]
    wmf.append(wwmf)

# Metatropi grad se rad
wmfr=[]
for i in range(len(wmf)):
    wwmfr=(math.pi*wmf[i])/200
    wmfr.append(wwmfr)

# Ypologismos syntelestwn u kai v
u=[]
for i in range(len(wmfr)):
    uu=math.sin(wmfr[i])
    u.append(uu)

v=[]
for i in range(len(wmfr)):
    vv=math.cos(wmfr[i])
    v.append(vv)

# ----- Ypologismos S/Sn -----
# Ypologismos S simeiw n 1 ws Koryfi
# Ypologismos apostasewn metaxisimeiw n
Sbp1K=[]
for i in range(0, int(korid-(id[0]-0.5))): # (id[int(korid-(id[0]-
0.5))]==korid & int(korid-(id[0]-0.5))==korid-1)
    Sbps1K=math.sqrt(((x[i+1]-x[i])**2)+(y[i+1]-y[i])**2)
    Sbp1K.append(Sbps1K)

# Ypologismos S kathesimeiou
S1K=[]
for i in range(len(Sbp1K)):
    if i==0:
        sslk=0+Sbp1K[i]
    else:
        sslk=sslk+Sbp1K[i]
    S1K.append(sslk)

# Ypologismos Sn apo 1 ws Koryfi
Sn1K=sum(Sbp1K, 0)
```



```
# Ypologismos S simeiw n wsKoryfi
# Ypologismosapostasewnmetaxisimeiw n
SbpNK=[]
for i in reversed(range(int(korid-(id[0]-0.5)), len(id)-1)):
    SbpsNK=math.sqrt(((x[i]-x[i+1])**2)+(y[i]-y[i+1])**2)
    SbpNK.append(SbpsNK)

# Ypologismos S kathesimeiou
SNK=[]
for i in range(len(SbpNK)):
    if i==0:
        sskn=0+SbpNK[i]
    else:
        sskn=sskn+SbpNK[i]
    SNK.append(sskn)

# Ypologismos Sn apo n wsKoryfi
SnNK=sum(SbpNK, 0)

# Ypologismos S/Sn (S1K/Sn1K) apo 2 wsKoryfi
Sf1K=[]
for i in range(len(S1K)):
    sf1k=S1K[i]/Sn1K
    Sf1K.append(sf1k)

# Ypologismos S/Sn (SNK/SnNK) apo n-1 wsKoryfi
SfNK=[]
for i in range(len(SNK)):
    sfnk=SNK[i]/SnNK
    SfNK.append(sfnk)

# ----- YpologismosXnew, Ynew Balloon -----
# YpologismosXnew apo 1 wsKoryfi
X1K=[]
for i in range(0, int(korid-(id[0]-0.5))+1):
    if i==0:
        x1k=x[i]
    else:
        x1k=x[i]+(Inflating[0]*u[i-1]*Sf1K[i-1])
    X1K.append(x1k)

# YpologismosYnew apo 1 wsKoryfi
Y1K=[]
for i in range(0, int(korid-(id[0]-0.5))+1):
    if i==0:
        y1k=y[i]
    else:
        y1k=y[i]+(Inflating[0]*v[i-1]*Sf1K[i-1])
    Y1K.append(y1k)

# Dimiourgiavoithitikwnpinakwnpouperilamvanoun mono ta simeia N ws K
xNKassist=[]
for i in reversed(range(int(korid-(id[0]-0.5)), len(id))):
    xNKassist.append(x[i])

yNKassist=[]
for i in reversed(range(int(korid-(id[0]-0.5)), len(id))):
    yNKassist.append(y[i])
```

```
# Antistrofiarxeiwn u kai v gianaxrisimopoihthounstonypologismoX,Ytwn N ws K
u.reverse()
v.reverse()

# YpologismosXnew apo N wsKoryfi
XNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        xnk=xNKassist[i]
    else:
        xnk=xNKassist[i]+(Inflating[0]*u[i-1]*SfNK[i-1])
    XNKnew.append(xnk)

# Antistrofi XNK se XKN
XNKnew.reverse()
# Diagrafi lou stoixeiou (dil. tou X tou K pouyparxei kai sto XlK)
XNKnew.remove(XNKnew[0])

# YpologismosYnew apo N wsKoryfi
YNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        ynk=yNKassist[i]
    else:
        ynk=yNKassist[i]+(Inflating[0]*v[i-1]*SfNK[i-1])
    YNKnew.append(ynk)

# Antistrofi YNK se YKN
YNKnew.reverse()
# Diagrafi lou stoixeiou (dil. tou Y tou K pouyparxei kai sto YlK)
YNKnew.remove(YNKnew[0])

# ===== X kai Y Diogwmenou Bend =====
# Enwsipinakwn
Xfinal=[]
Xfinal=XlK+XNKnew

Yfinal=[]
Yfinal=YlK+YNKnew

# =====
# ===== AfinikosMetasximatismos =====
# ===== Telika X kai Y =====
# =====

Xf=[]
Yf=[]
IDf=[]
if CCP>=e[0]/2: # afinikosmetasximatismoswsteCPballoon -->CParxikis
    a=((Yfinal[int(korid-(id[0]-0.5))] *xpa[0]) - (Yfinal[len(id) -
1] *xpa[0]) + (Yfinal[len(id) -1] *xpa[1]) - (Yfinal[0] *xpa[1]) + (Yfinal[0] *xpa[2]) -
(Yfinal[int(korid-(id[0]-0.5))] *xpa[2])) / ((Xfinal[0] *Yfinal[int(korid-(id[0]-
0.5))]) - (Xfinal[0] *Yfinal[len(id) -1]) - (Yfinal[0] *Xfinal[int(korid-(id[0]-
0.5))]) + (Yfinal[0] *Xfinal[len(id) -1]) + (Xfinal[int(korid-(id[0]-
0.5))]) *Yfinal[len(id) -1]) - (Xfinal[len(id) -1] *Yfinal[int(korid-(id[0]-0.5))]))
# Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
```

```

    b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[int(korid-(id[0]-
0.5))] *xpa[0]))+(Xfinal[0]*xpa[1])-(Xfinal[len(id)-
1]*xpa[1])+(Xfinal[int(korid-(id[0]-0.5))] *xpa[2])-(
(Xfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))] )-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))] ))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))] *Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))] ))
    c=((Xfinal[int(korid-(id[0]-0.5))] *Yfinal[len(id)-1]*xpa[0])-(
(Yfinal[int(korid-(id[0]-0.5))] *Xfinal[len(id)-1]*xpa[0]))+(Xfinal[len(id)-
1]*Yfinal[0]*xpa[1])-(Yfinal[len(id)-
1]*Xfinal[0]*xpa[1]))+(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))] *xpa[2])-(
(Xfinal[int(korid-(id[0]-
0.5))] *Yfinal[0]*xpa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))] )-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))] ))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))] *Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))] ))
    d=((Yfinal[int(korid-(id[0]-0.5))] *ypa[0])-(Yfinal[len(id)-
1]*ypa[0]))+(Yfinal[len(id)-1]*ypa[1])-(Yfinal[0]*ypa[1]))+(Yfinal[0]*ypa[2])-(
(Yfinal[int(korid-(id[0]-0.5))] *ypa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-
0.5))] )-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))] ))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))] *Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))] ))
    ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[int(korid-(id[0]-
0.5))] *ypa[0]))+(Xfinal[0]*ypa[1])-(Xfinal[len(id)-
1]*ypa[1]))+(Xfinal[int(korid-(id[0]-0.5))] *ypa[2])-(
(Xfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))] )-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))] ))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))] *Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))] ))
    f=((Xfinal[int(korid-(id[0]-0.5))] *Yfinal[len(id)-1]*ypa[0])-(
(Yfinal[int(korid-(id[0]-0.5))] *Xfinal[len(id)-1]*ypa[0]))+(Xfinal[len(id)-
1]*Yfinal[0]*ypa[1])-(Xfinal[0]*Yfinal[len(id)-
1]*ypa[1]))+(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))] *ypa[2])-(
(Xfinal[int(korid-(id[0]-
0.5))] *Yfinal[0]*ypa[2]))/((Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))] )-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))] ))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))] *Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))] ))
    Xf.append(x[0]) # Eisagwgiprwτουςimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos
    Yf.append(y[0])
    IDf.append(id[0])
    for i in range(1, len(id)-1): # Ypologismosnewsyntetagmenwn apo
afinikometasximatismotwnsimeiw n 2-->n-1
        xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c
        yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
        Xf.append(xaffine)
        Yf.append(yaffine)
        IDf.append(id[i])
    Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi Gauss /
den efarmozetaiafinikosmetasximatismos
    Yf.append(y[len(id)-1])
    IDf.append(id[len(id)-1])
elif CCP<e[0]/2: # Ypologismostomiskyklou - eftheiasmetatopisistouCPgauss
    l=-((x[int(korid-(id[0]-0.5))+1]-x[int(korid-(id[0]-0.5))-
1]))/(y[int(korid-(id[0]-0.5))+1]-y[int(korid-(id[0]-0.5))-1])) #
Ypologismossyntelestidiefthinsis (l) eftheiasmetatopisiskoryfis

```

```

D=(((-2*xc)-(2*(1**2)*(korx))+(2*1*kory)-(2*yc*1)**2)-
(4*((1**2)+1)*((xc**2)+(yc**2)-((e[0]**2)/4)+((1**2)*(korx**2)))-
(2*1*korx*kory)+(kory**2)+(2*yc*1*korx)-(2*yc*kory)) #
YpologismosDiakrinousas D
xt1=((2*xc)+(2*(1**2)*korx)-
(2*1*kory)+(2*yc*1))+(math.sqrt(D))/((2*(1**2))+2) #
Ypologismospithanwnlisewn t1,t2
yt1=(1*xt1)-(1*korx)+kory
xt2=((2*xc)+(2*(1**2)*korx)-(2*1*kory)+(2*yc*1))-
(math.sqrt(D))/((2*(1**2))+2)
yt2=(1*xt2)-(1*korx)+kory
ap1=math.sqrt(((korx-xt1)**2)+((kory-yt1)**2)) # Ypologismosapostasewn
CP-t1 (ap1) kai CP-t2 (ap2)
ap2=math.sqrt(((korx-xt2)**2)+((kory-yt2)**2))
if ap1>ap2: # Entopismosapodektis lysis systimatos (to
xtpouapexeimikroteriapostasi apo to CP)
xtomi=xt2
ytomi=yt2
elif ap1<ap2:
xtomi=xt1
ytomi=yt1
if CCParx>e[0]/2: # afinikosmetasximatismoswsteCPballoon -->CParxikis
a=((Yfinal[int(korid-(id[0]-0.5))] *xpa[0])-(Yfinal[len(id)-
1]*xpa[0])+(Yfinal[len(id)-1]*xpa[1])-(Yfinal[0]*xpa[1])+(Yfinal[0]*xpa[2])-(
Yfinal[int(korid-(id[0]-0.5))] *xpa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-
0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
# Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[int(korid-(id[0]-
0.5))]*xpa[0])+(Xfinal[0]*xpa[1])-(Xfinal[len(id)-
1]*xpa[1])+(Xfinal[int(korid-(id[0]-0.5))]*xpa[2])-(
(Xfinal[0]*xpa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
c=((Xfinal[int(korid-(id[0]-0.5))] *Yfinal[len(id)-1]*xpa[0])-(
Yfinal[int(korid-(id[0]-0.5))]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-
1]*Yfinal[0]*xpa[1])-(Yfinal[len(id)-
1]*Xfinal[0]*xpa[1])+(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))]*xpa[2])-(
(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[0]*xpa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
d=((Yfinal[int(korid-(id[0]-0.5))] *ypa[0])-(Yfinal[len(id)-
1]*ypa[0])+(Yfinal[len(id)-1]*ypa[1])-(Yfinal[0]*ypa[1])+(Yfinal[0]*ypa[2])-(
Yfinal[int(korid-(id[0]-0.5))]*ypa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-
0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[int(korid-(id[0]-
0.5))]*ypa[0])+(Xfinal[0]*ypa[1])-(Xfinal[len(id)-
1]*ypa[1])+(Xfinal[int(korid-(id[0]-0.5))]*ypa[2])-(
(Xfinal[0]*ypa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))])+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))

```

```

f=((Xfinal[int(korid-(id[0]-0.5))]*Yfinal[len(id)-1]*ypa[0])-
(Yfinal[int(korid-(id[0]-0.5))]*Xfinal[len(id)-1]*ypa[0])+(Xfinal[len(id)-
1]*Yfinal[0]*ypa[1])-(Xfinal[0]*Yfinal[len(id)-
1]*ypa[1])+(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))]*ypa[2])-
(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[0]*ypa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))]-
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
Xf.append(x[0]) # Eisagwgiprwtoousimeiou apo grammi Gauss / den
efarmozetaiafinikosmetasximatismos
Yf.append(y[0])
IDf.append(id[0])
for i in range(1, len(id)-1): # Ypologismosnewsyntetagmenwn apo
afinikometasximatismotwnsimeiw n 2-->n-1
xaffine=(a*Xfinal[i])+(b*Yfinal[i])+c
yaffine=(d*Xfinal[i])+(ee*Yfinal[i])+f
Xf.append(xaffine)
Yf.append(yaffine)
IDf.append(id[i])
Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi
Gauss / den efarmozetaiafinikosmetasximatismos
Yf.append(y[len(id)-1])
IDf.append(id[len(id)-1])
elifCCParx<e[0]/2: # afinikosmetasximatismoswsteCPballoon --
>tomieftheiasmetatopisisCPgauss - kykloou
a=((Yfinal[int(korid-(id[0]-0.5))]*xpa[0])-(Yfinal[len(id)-
1]*xpa[0])+(Yfinal[len(id)-1]*xtomi)-(Yfinal[0]*xtomi)+(Yfinal[0]*xpa[2])-
(Yfinal[int(korid-(id[0]-0.5))]*xpa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-
0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
# Ypologismossintelestwna,b,c,d,e,fafinikoumetasximatismou
b=((Xfinal[len(id)-1]*xpa[0])-(Xfinal[int(korid-(id[0]-
0.5))]*xpa[0])+(Xfinal[0]*xtomi)-(Xfinal[len(id)-1]*xtomi)+(Xfinal[int(korid-
(id[0]-0.5))]*xpa[2])-(Xfinal[0]*xpa[2]))/(Xfinal[0]*Yfinal[int(korid-
(id[0]-0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-
(id[0]-0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
c=((Xfinal[int(korid-(id[0]-0.5))]*Yfinal[len(id)-1]*xpa[0])-
(Yfinal[int(korid-(id[0]-0.5))]*Xfinal[len(id)-1]*xpa[0])+(Xfinal[len(id)-
1]*Yfinal[0]*xtomi)-(Yfinal[len(id)-
1]*Xfinal[0]*xtomi)+(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))]*xpa[2])-
(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[0]*xpa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-0.5))])-(
(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-
0.5))]))
d=((Yfinal[int(korid-(id[0]-0.5))]*ypa[0])-(Yfinal[len(id)-
1]*ypa[0])+(Yfinal[len(id)-1]*ytomi)-(Yfinal[0]*ytomi)+(Yfinal[0]*ypa[2])-
(Yfinal[int(korid-(id[0]-0.5))]*ypa[2]))/(Xfinal[0]*Yfinal[int(korid-(id[0]-
0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-(id[0]-
0.5))]))+(Yfinal[0]*Xfinal[len(id)-1])+(Xfinal[int(korid-(id[0]-
0.5))]*Yfinal[len(id)-1])-(Xfinal[len(id)-1]*Yfinal[int(korid-(id[0]-0.5))]))
ee=((Xfinal[len(id)-1]*ypa[0])-(Xfinal[int(korid-(id[0]-
0.5))]*ypa[0])+(Xfinal[0]*ytomi)-(Xfinal[len(id)-1]*ytomi)+(Xfinal[int(korid-
(id[0]-0.5))]*ypa[2])-(Xfinal[0]*ypa[2]))/(Xfinal[0]*Yfinal[int(korid-
(id[0]-0.5))])-(Xfinal[0]*Yfinal[len(id)-1])-(Yfinal[0]*Xfinal[int(korid-

```

```
(id[0]-0.5))] + (Yfinal[0]*Xfinal[len(id)-1]) + (Xfinal[int(korid-(id[0]-0.5))] * Yfinal[len(id)-1]) - (Xfinal[len(id)-1] * Yfinal[int(korid-(id[0]-0.5))]))
    f = ((Xfinal[int(korid-(id[0]-0.5))] * Yfinal[len(id)-1] * ypa[0]) - (Yfinal[int(korid-(id[0]-0.5))] * Xfinal[len(id)-1] * ypa[0]) + (Xfinal[len(id)-1] * Yfinal[0] * ytomi) - (Xfinal[0] * Yfinal[len(id)-1] * ytomi) + (Xfinal[0] * Yfinal[int(korid-(id[0]-0.5))] * ypa[2]) - (Xfinal[int(korid-(id[0]-0.5))] * Yfinal[0] * ypa[2])) / ((Xfinal[0] * Yfinal[int(korid-(id[0]-0.5))]) - (Xfinal[0] * Yfinal[len(id)-1]) - (Yfinal[0] * Xfinal[int(korid-(id[0]-0.5))]) + (Yfinal[0] * Xfinal[len(id)-1]) + (Xfinal[int(korid-(id[0]-0.5))] * Yfinal[len(id)-1]) - (Xfinal[len(id)-1] * Yfinal[int(korid-(id[0]-0.5))]))
    Xf.append(x[0]) # Eisagwgiprw tousimeiou apo grammi Gauss / den efarmozetaiafinikosmetasximatismos
    Yf.append(y[0])
    IDf.append(id[0])
    for i in range(1, len(id)-1): # Ypologismosnewsyntetagmenwn apo afinikometasximatismotwnsimeiw n 2-->n-1
        xaffine = (a*Xfinal[i]) + (b*Yfinal[i]) + c
        yaffine = (d*Xfinal[i]) + (ee*Yfinal[i]) + f
        Xf.append(xaffine)
        Yf.append(yaffine)
        IDf.append(id[i])
    Xf.append(x[len(id)-1]) # Eisagwgiteleutaiousimeiou apo grammi Gauss / den efarmozetaiafinikosmetasximatismos
    Yf.append(y[len(id)-1])
    IDf.append(id[len(id)-1])

# ===== YpologismosXarakteristikwnTimwnSyntelestiDiogwsis (Inflating)
# (Times Inflating pouypologizontaistin 'ypodesmeuseis' ekdoxitouprogrammatos (ArcGISBalloon22new2TypeB). Oi times apotelounkatwfli kata tidiadikasiadokimwn)
if CCP >= e[0]/2:
    dCParxCPgauss = math.sqrt(((korx-xpa[1])**2) + ((kory-ypa[1])**2)) # apostasiCPgauss apo CParxikis
elif CCP < e[0]/2:
    dtomiCPgauss = math.sqrt(((korx-xtomi)**2) + ((kory-ytomi)**2)) # apostasiCPgauss apo tin tomikyklou-eftheiasmetatopisistouCPgauss

# ===== ExagwgiArxeiwn =====
# ----- Elaxistes Times SyntelestiDiogwsis (Inflating) -----
gout = open('C:InflatingLowestValue.txt', 'w')
if CCP >= e[0]/2:
    gout.write('CCP >= e/2' + ' ... ' + ('Distance CParx-CPgauss:' + str(dCParxCPgauss)) + '\n')
elif CCP < e[0]/2:
    gout.write('CCP < e/2' + ' ... ' + ('Distance Cptomi-CPgauss:' + str(dtomiCPgauss)) + '\n')

gout.close()

# ----- GenikeumeniGrammi -----
fout = open('C:GeneralizedTypeBPartfInf.txt', 'w')
fout.write(str('Polyline') + '\n')
for i in range(len(id)):
    fout.write(str(IDf[i]) + ' ' + str(Xf[i]) + ' ' + str(Yf[i])) + '\n')

fout.write(str('END'))
fout.close()
```

g.close()  
m.close()  
h.close()  
o.close()  
t.close()  
z.close()

### Π4.3 Πρόγραμμα γενίκευσης τμημάτων σύγκλισης

```

import arcgisscripting
gp = arcgisscripting.create(9.3)

# ===== EisagwgiDedomenwn =====
# ----- EisagwgiGrammis -----
f = open('C:ConflictPart_Points.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in f:
    NumLines=NumLines+1

f = open('C:ConflictPart_Points.txt') # Anoigmaarxeiou

# Eisagogisimeiwngammis (x,y,id)
x=[]
y=[]
id=[]
mtr=0
for line in f:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        id.append(int(line[:-1].split(' ')[0]))
        x.append(float(line[:-1].split(' ')[1]))
        y.append(float(line[:-1].split(' ')[2]))

# ----- EisagwgiKoryfwn Conflict -----
g = open('C:Kor.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in g:
    NumLines=NumLines+1

g = open('C:Kor.txt') # Anoigmaarxeiou

# Eisagogikoryfwn (xk,yk,idk)
xk=[]
yk=[]
idk=[]
mtr=0
for line in g:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        idk.append(int(line[:-1].split(' ')[0]))
        xk.append(float(line[:-1].split(' ')[1]))
        yk.append(float(line[:-1].split(' ')[2]))

# ----- EisagogiKentrou Conflict -----
k = open('C:CC.txt') # Anoigmaarxeiou
NumLines=0 # Ypologismosarithmougrammwn (lines) arxeiou
for line in k:
    NumLines=NumLines+1

k = open('C:CC.txt') # Anoigmaarxeiou

# Eisagogikentrou (xcc,ycc, idcc)
xcc=[]
ycc=[]
idcc=[]

```



```

mtr=0
for line in k:
    mtr=mtr+1
    if mtr>1 and mtr<NumLines:
        idcc.append(int(line[:-1].split(' ')[0]))
        xcc.append(float(line[:-1].split(' ')[1]))
        ycc.append(float(line[:-1].split(' ')[2]))

# ----- Eisagwgi e -----
m = open('C:e.txt')
e=[]
for line in m:
    e.append(float(line[:]))

# -----
import math
# -----

# =====
# ===== SympiesiGrammis =====
# =====

# ----- Ypologismos Depress -----
# Ypologismoselaxistisapostasis (apostasimetaxikoryfwn)
minD=math.sqrt(((xk[1]-xk[0])**2)+((yk[1]-yk[0])**2))

Depress=(e[0]-minD)/2

# ----- Entopismoskorifisstasimeia tis grammis -----
korx=[]
kory=[]
korid=[]
# Exetasipoiia ek twnkoryfwnanikeistigrammi
for i in range(len(id)):
    if x[i]==xk[0] and y[i]==yk[0]:
        korx=x[i]
        kory=y[i]
        korid=id[i]

for i in range(len(id)):
    if x[i]==xk[1] and y[i]==yk[1]:
        korx=x[i]
        kory=y[i]
        korid=id[i]

# ----- Ypologismos sense -----
if korx>xcc[0] and kory>ycc[0]:
    sensex=1
    sensey=1
elifkorx<xcc[0] and kory>ycc[0]:
    sensex=-1
    sensey=1
elifkorx<xcc[0] and kory<ycc[0]:
    sensex=-1
    sensey=-1
elifkorx>xcc[0] and kory<ycc[0]:
    sensex=1
    sensey=-1

```

```
# ----- Ypologismos syntelestwn a kai b -----
# Ypologismos gnias diefthinsiskoryfwn
dxk=xk[1]-xk[0]
dyk=yk[1]-yk[0]
if dxk>0 and dyk>0:
    w=((math.atan((math.fabs(dxk))/(math.fabs(dyk))))*(200/math.pi))
elif dxk>0 and dyk<0:
    w=200-((math.atan((math.fabs(dxk))/(math.fabs(dyk))))*(200/math.pi))
elif dxk<0 and dyk<0:
    w=200+((math.atan((math.fabs(dxk))/(math.fabs(dyk))))*(200/math.pi))
elif dxk<0 and dyk>0:
    w=400-((math.atan((math.fabs(dxk))/(math.fabs(dyk))))*(200/math.pi))
elif dxk==0 and dyk>0:
    w=0
elif dxk==0 and dyk<0:
    w=200
elif dxk>0 and dyk==0:
    w=100
elif dxk<0 and dyk==0:
    w=300

# Elegxos giapithania fairesi h' prosthesi 400g
if w>400:
    w=w-400
elif w<0:
    w=w+400
else:
    w=w

# Metatropi grad se rad
wr=(math.pi*w)/200

# Ypologismos syntelestwn a kai b
a=math.fabs(math.sin(wr))
b=math.fabs(math.cos(wr))

# ----- Ypologismos S/Sn -----
# Ypologismos S simeiw n wsKoryfi
# Ypologismos apostasewn metaxisimeiw n
Sbp1K=[]
for i in range(0, korid-1):
    Sbps1K=math.sqrt(((x[i+1]-x[i])**2)+((y[i+1]-y[i])**2))
    Sbp1K.append(Sbps1K)

# Ypologismos S kathesimeiou
S1K=[]
for i in range(len(Sbp1K)):
    if i==0:
        sslk=0+Sbp1K[i]
    else:
        sslk=sslk+Sbp1K[i]
    S1K.append(sslk)

# Ypologismos Sn apo 1 wsKoryfi
Sn1K=sum(Sbp1K, 0)

# Ypologismos S simeiw n wsKoryfi
```

```
# Ypologismosapostasewnmetaxisimeiw  
SbpNK=[]  
for i in reversed(range(korid-1, len(id)-1)):  
    SbpsNK=math.sqrt(((x[i]-x[i+1])**2)+(y[i]-y[i+1])**2))  
    SbpNK.append(SbpsNK)  
  
# Ypologismos S kathesimeiou  
SNK=[]  
for i in range(len(SbpNK)):  
    if i==0:  
        sskn=0+SbpNK[i]  
    else:  
        sskn=sskn+SbpNK[i]  
    SNK.append(sskn)  
  
# Ypologismos Sn apo N wsKoryfi  
SnNK=sum(SbpNK, 0)  
  
# Ypologismos S/Sn (S1K/Sn1K) apo 2 wsKoryfi  
Sf1K=[]  
for i in range(len(S1K)):  
    sf1k=S1K[i]/Sn1K  
    Sf1K.append(sf1k)  
  
# Ypologismos S/Sn (SNK/SnNK) apo n-1 wsKoryfi  
SfNK=[]  
for i in range(len(SNK)):  
    sfnk=SNK[i]/SnNK  
    SfNK.append(sfnk)  
  
# ----- YpologismosX,YSympiesmenisGrammis -----  
# YpologismosXnew apo 1 wsKoryfi  
X1K=[]  
for i in range (0, korid):  
    if i==0:  
        x1k=x[i]  
    else:  
        x1k=x[i]+(Depress*sensex*a*Sf1K[i-1])  
    X1K.append(x1k)  
  
# YpologismosYnew apo 1 wsKoryfi  
Y1K=[]  
for i in range (0, korid):  
    if i==0:  
        y1k=y[i]  
    else:  
        y1k=y[i]+(Depress*sensey*b*Sf1K[i-1])  
    Y1K.append(y1k)  
  
# Dimiourgiavoithitikwnpinakwnpouperilamvanoun mono ta simeia N ws K  
xNKassist=[]  
for i in reversed(range(korid-1, len(id))):  
    xNKassist.append(x[i])  
  
yNKassist=[]  
for i in reversed(range(korid-1, len(id))):  
    yNKassist.append(y[i])
```

```
# Υπολογισμός Xnew από N wsKoryfi
XNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        xnk=xNKassist[i]
    else:
        xnk=xNKassist[i]+(Depress*sensex*a*SfNK[i-1])
    XNKnew.append(xnk)

# Αντιστροφή XNK σε XKN
XNKnew.reverse()
# Διαγραφή του στοιχείου (dil. του X του K που υπαρχε στο X1K)
XNKnew.remove(XNKnew[0])

# Υπολογισμός Ynew από N wsKoryfi
YNKnew=[]
for i in range(len(xNKassist)):
    if i==0:
        ynk=yNKassist[i]
    else:
        ynk=yNKassist[i]+(Depress*sensey*b*SfNK[i-1])
    YNKnew.append(ynk)

# Αντιστροφή YNK σε YKN
YNKnew.reverse()
# Διαγραφή του στοιχείου (dil. του Y του K που υπαρχε στο Y1K)
YNKnew.remove(YNKnew[0])

# ===== X και Y Συμπιέσεις Γραμμών =====
# Ενσωμάτωση
Xfinal=[]
Xfinal=X1K+XNKnew

Yfinal=[]
Yfinal=Y1K+YNKnew

# ===== Εξαγωγή αρχείου =====
fout=open('C:GeneralizedConflictPart.txt', 'w')
fout.write(str('Polyline')+'\n')
for i in range(len(id)):
    fout.write((str(id[i]))+' '+'+(str(Xfinal[i]))+' '+'+(str(Yfinal[i]))+'\n')

fout.write(str('END'))
fout.close()

f.close()
g.close()
k.close()
m.close()
```