

**Σχεδιασμός και Υλοποίηση των  
Μηχανισμών Διαχείρισης Ετικετών στο  
Περιβάλλον Ανάπτυξης Εφαρμογών  
Γεωγραφικών Δεδομένων Map Studio**

*Διπλωματική Εργασία που παρουσιάστηκε στο  
Πολυτεχνείο Κρήτης στα πλαίσια των απαιτήσεων  
απόκτησης Διπλώματος της Σχολής Ηλεκτρονικών  
Μηχανικών & Μηχανικών Ηλεκτρονικών  
Υπολογιστών*

**Ελευθέριος Τσεκούρας**

**Αύγουστος 1998**

# Περίληψη

Το αντικείμενο της εργασίας αυτής είναι η δημιουργία ενός ολοκληρωμένου μηχανισμού ανάπτυξης και διαχείρισης ετικετών που ενσωματώθηκε στο περιβάλλον Ανάπτυξης Εφαρμογών Γεωγραφικών Δεδομένων Map Studio.

Η χρήση των ετικετών προσφέρει στο δημιουργό του χάρτη τη δυνατότητα να παρουσιάσει στους αναγνώστες του διάφορες πληροφορίες σχετικά με τα αντικείμενα που απεικονίζονται στο χάρτη.

Το μοντέλο των ετικετών που υλοποιήθηκε στα πλαίσια της παρούσας διατριβής αναπτύχθηκε με βάση την προϋπάρχουσα ιεραρχία αντικειμένων της εφαρμογής Map Studio. Επιπλέον, χρησιμοποιήθηκαν διάφοροι τρόποι παρουσίασης και απεικόνισης των ετικετών στο χάρτη, ανάλογα με τον τύπο του αντικειμένου στο οποίο ανήκει η κάθε ετικέτα. Με αυτό τον τρόπο επιτυγχάνονται πιο καλαίσθητες παρουσιάσεις των χαρτών που κατασκευάζονται, ενώ γίνεται και πιο εύκολη η διασύνδεση των ετικετών με τα αντικείμενα στα οποία ανήκουν.

Επίσης, παρέχεται στο χρήστη η δυνατότητα μετακίνησης των ετικετών πάνω στο χάρτη σύμφωνα με τις επιθυμίες του, χωρίς όμως να επιτρέπεται η τοποθέτηση των ετικετών με τέτοιο τρόπο ώστε να καθίσταται προβληματική η αντιστοίχησή τους με τα αντικείμενα στα οποία ανήκουν.

Τέλος, με τη χρήση διάφορων αλγορίθμων που υλοποιήσαμε, είναι δυνατή η αυτόματη τοποθέτηση των ετικετών σε θέσεις στις οποίες δεν παρουσιάζονται επικαλύψεις μεταξύ τους. Με αυτό τον τρόπο επιτυγχάνεται η παρουσίαση αρκετά πιο καλαίσθητων χαρτών, χωρίς την άμεση παρέμβαση του χρήστη της εφαρμογής.

Το σύστημα υλοποιήθηκε σε πλατφόρμα Windows 95 με τη χρήση της Microsoft Visual C++.

# Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθηγητή κ. Χριστοδουλάκη Σταύρο για την επίβλεψη και την καθοδήγησή του για την ολοκλήρωση αυτής της διπλωματικής εργασίας, καθώς και για την ευκαιρία που μου προσέφερε να εργαστώ στο Εργαστήριο Διανεμημένων Πληροφοριακών Συστημάτων και Εφαρμογών, από όπου αποκόμισα πολύτιμες εμπειρίες.

Επίσης, θα ήθελα να ευχαριστήσω προκαταβολικά τους αναγνώστες της εργασίας κ. Τριανταφύλλου Παναγιώτη και κ. Δόλλα Απόστολο για το χρόνο που θα διαθέσουν για την ανάγνωσή της και τις παρατηρήσεις τους.

Τέλος, οφείλω να ευχαριστήσω ιδιαίτερα τους μεταπτυχιακούς φοιτητές Μουμουτζή Νεκτάριο και Αναστασιάδη Μανόλη για τη συνεργασία και τη βοήθεια που μου προσέφεραν στο σχεδιασμό της εργασίας, καθώς και για τις συμβουλές τους σε τεχνικά θέματα και στη δομή του κειμένου αυτού.

# Περιεχόμενα

<i>Περιεχόμενα</i> .....	4
<i>Κατάλογος Σχημάτων</i> .....	6
<i>Κατάλογος Πινάκων</i> .....	8
<i>1 Εισαγωγή</i> .....	9
1.1 ΓΕΝΙΚΑ.....	9
1.2 ΣΚΟΠΟΣ ΚΑΙ ΑΝΑΓΚΑΙΟΤΗΤΑ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ.....	10
1.3 ΔΟΜΗ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ.....	13
<i>2 Περιγραφή του Περιβάλλοντος Ανάπτυξης Εφαρμογών Γεωγραφικών Δεδομένων Map Studio</i> ... 15	
2.1 ΒΑΣΙΚΑ ΑΝΤΙΚΕΙΜΕΝΑ.....	15
2.2 ΚΛΑΣΕΙΣ ΓΕΝΙΚΗΣ ΧΡΗΣΗΣ.....	17
2.3 ΓΕΩΓΡΑΦΙΚΑ ΑΝΤΙΚΕΙΜΕΝΑ.....	18
2.4 ΕΠΙΠΕΔΟ.....	21
2.5 ΧΑΡΤΗΣ.....	25
2.5.1 Στιλ Σχεδιασμού.....	25
2.5.2 Δεσμευμένο Επίπεδο.....	26
2.6 ΔΟΜΕΣ ΥΠΟΣΤΗΡΙΞΗΣ.....	27
2.6.1 Δεξαμενή Αντικειμένων.....	28
2.6.2 Χωρικό Ευρετήριο (Τετραδικό Δέντρο).....	28
2.7 ΠΕΡΙΛΗΨΗ.....	31
<i>3 Περιγραφή Αλγορίθμων Επίλυσης του Προβλήματος Βέλτιστης Τοποθέτησης Ετικετών</i> ..... 32	
3.1 ΓΕΝΙΚΑ.....	32
3.2 ΠΟΛΥΠΛΟΚΟΤΗΤΑ ΤΟΥ ΠΒΤΕ.....	34
3.3 ΑΛΓΟΡΙΘΜΟΙ ΕΠΙΛΥΣΗΣ ΤΟΥ ΠΒΤΕ ΓΙΑ ΕΤΙΚΕΤΕΣ ΣΗΜΕΙΑΚΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ.. 35	
3.3.1 Αλγόριθμοι Εξαντλητικής Έρευνας (Exhaustive Search Algorithms).....	35
3.3.2 Άπληστοι Αλγόριθμοι (Greedy Algorithms).....	36
3.3.3 Αλγόριθμοι Καθόδου Σταδιακής Κλίσης (Discrete Gradient Descent Algorithms).....	36
3.3.4 Προσέγγιση της Καθόδου (Gradient) με τη χρήση Διανυσμάτων Επικάλυψης (Overlap Vectors).....	38
3.3.5 Μαθηματικός Προγραμματισμός Επίλυσης του ΠΒΤΕ.....	41
3.3.6 Ανακεφαλαίωση των Αλγορίθμων επίλυσης του ΠΒΤΕ για Σημειακά Αντικείμενα....	47
3.4 ΑΛΓΟΡΙΘΜΟΙ ΕΥΡΕΣΗΣ ΕΛΕΥΘΕΡΟΥ ΧΩΡΟΥ ΓΙΑ ΕΤΙΚΕΤΕΣ ΣΗΜΕΙΑΚΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ.....	47
3.4.1 Τυποποίηση του Προβλήματος Ελεύθερου Χώρου.....	49
3.4.2 Αλγόριθμος Εύρεσης Ελεύθερου Χώρου με Σημείο Αναφοράς.....	50
3.4.3 Αλγόριθμος Εύρεσης Ελεύθερου Χώρου με Μέγιστα Ελεύθερα Ορθογώνια.....	54
3.5 ΠΕΡΙΛΗΨΗ.....	55
<i>4 Υλοποίηση Αλγορίθμων</i> .....	57
4.1 ΓΕΝΙΚΑ.....	57
4.2 ΤΟ ΠΡΟΒΛΗΜΑ ΤΩΝ ΕΤΙΚΕΤΩΝ.....	57
4.3 Η ΜΕΘΟΔΟΣ ΤΗΣ ΠΡΟΣΟΜΟΙΩΣΗΣ ΑΝΟΠΤΗΣΗΣ.....	58
4.4 ΑΛΓΟΡΙΘΜΟΣ ΕΠΙΛΥΣΗΣ ΤΟΥ ΠΒΤΕ.....	60
4.4.1 Αρχικοποίηση.....	61

4.4.2 Αντικειμενική συνάρτηση .....	62
4.4.3 Τροποποίηση λύσεων .....	63
4.4.4 Πρόγραμμα Ανόπτησης.....	63
<b>4.5 ΕΤΙΚΕΤΕΣ ΓΡΑΜΜΙΚΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ .....</b>	<b>68</b>
4.5.1 Καμπύλες Bézier .....	69
4.5.2 Πολύωνυμα Bernstein .....	70
4.5.3 Αναλυτική Έκφραση Καμπύλης Bézier .....	71
4.5.4 Κλίση των Χαρακτήρων των Ετικετών.....	72
4.5.5 Επίλυση συγκρούσεων Ετικετών Γραμμικών Αντικειμένων .....	76
<b>4.6 ΠΕΡΙΛΗΨΗ .....</b>	<b>79</b>
<i>5 Μοντέλο Αντικειμένων Ετικετών .....</i>	<i>80</i>
<b>5.1 ΓΕΝΙΚΑ .....</b>	<b>80</b>
<b>5.2 ΜΟΝΤΕΛΟ ΕΤΙΚΕΤΩΝ .....</b>	<b>80</b>
5.2.1 Κλάση Ετικέτας (CLabel) .....	81
5.2.2 Ετικέτα Εικονίδιο (CIconLabel).....	83
5.2.3 Ετικέτες Κειμένου .....	85
<b>5.3 ΔΟΜΕΣ ΥΠΟΣΤΗΡΙΞΗΣ ΓΙΑ ΤΟ ΜΟΝΤΕΛΟ ΤΩΝ ΕΤΙΚΕΤΩΝ .....</b>	<b>89</b>
<b>5.4 ΤΡΟΠΟΠΟΙΗΣΕΙΣ ΣΤΙΣ ΥΠΑΡΧΟΥΣΕΣ ΔΟΜΕΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ .....</b>	<b>90</b>
<b>5.5 ΠΕΡΙΓΡΑΦΗ ΜΗΧΑΝΙΣΜΟΥ ΑΛΛΗΛΕΠΙΔΡΑΣΗΣ ΓΙΑ ΤΗ ΔΙΑΧΕΙΡΙΣΗ ΤΩΝ ΕΤΙΚΕΤΩΝ</b>	<b>92</b>
<b>5.4 ΠΕΡΙΛΗΨΗ .....</b>	<b>97</b>
<i>6 Ανακεφαλαίωση, Συνεισφορά και Μελλοντικές Επεκτάσεις.....</i>	<i>99</i>
<b>6.1 ΓΕΝΙΚΑ .....</b>	<b>99</b>
<b>6.2 ΑΝΑΚΕΦΑΛΑΙΩΣΗ .....</b>	<b>99</b>
<b>6.3 ΣΥΝΕΙΣΦΟΡΑ.....</b>	<b>101</b>
<b>6.4 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ .....</b>	<b>103</b>
<i>Παράρτημα Α .....</i>	<i>106</i>
<b>ΣΥΜΒΑΣΕΙΣ ΟΝΟΜΑΣΙΩΝ ΚΑΤΑ ΤΑ ΠΡΟΤΥΠΑ ΤΗΣ MFC (ΕΠΕΚΤΑΣΕΙΣ ΣΤΗΝ ΟΥΓΓΡΙΚΗ ΣΥΜΒΑΣΗ).....</b>	<b>106</b>
<i>Παράρτημα Β .....</i>	<i>108</i>
<b>ΟΙ ΒΑΣΙΚΕΣ ΚΛΑΣΕΙΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ MFC .....</b>	<b>108</b>
<i>Βιβλιογραφία.....</i>	<i>109</i>

# Κατάλογος Σχημάτων

Σχήμα 2.1 Οι κλάσεις αντικειμένων	16
Σχήμα 2.2 Η EINAI ιεραρχία των κλάσεων Γενικής Χρήσης	17
Σχήμα 2.3 Κλάσεις Γεωγραφικών Αντικειμένων	18
Σχήμα 2.4 Παράδειγμα Γεωγραφικών Αντικειμένων	20
Σχήμα 2.5 Η δομή του επιπέδου	21
Σχήμα 2.6 Γωνία διαδρομής με άξονα παράλληλο του $X$	22
Σχήμα 2.7 Παράδειγμα επιπέδου	23
Σχήμα 2.8 Το Δεσμευμένο Επίπεδο ως υπερσύνολο του Επιπέδου	26
Σχήμα 2.9 Η ιεραρχία κλάσεων Επιπέδων	27
Σχήμα 2.10 Τετραδικό Δέντρο μέγιστου βάθους 5 με Γεωμετρικά Αντικείμενα	30
Σχήμα 3.1 Οι οκτώ θέσεις μιας ετικέτας σε σχέση με το σημείο που περιγράφει. Οι καλύτερες θέσεις είναι αυτές με τον μικρότερο αριθμό.	33
Σχήμα 3.2 Παραδείγματα συνεχών μοντέλων τοποθέτησης ετικετών	34
Σχήμα 3.3 Παράδειγμα τοπικού ελάχιστου για τους Αλγόριθμους ΚΣΚ	38
Σχήμα 3.4 Πιθανές θέσεις τοποθέτησης ετικέτας γύρω από το σημείο στο οποίο ανήκει κατά τον Hirsch [1982]	38
Σχήμα 3.5 Ειδικές θέσεις τοποθέτησης ετικετών	39
Σχήμα 3.6 Διανύσματα επικάλυψης	39
Σχήμα 3.7 Το πρόβλημα του τοπικού ελάχιστου για τον Αλγόριθμο του Hirsch	41
Σχήμα 3.8 Περίπτωση τοπικού ελάχιστου για τον Αλγόριθμο του Zoraster	45
Σχήμα 3.9 Μία ασταθής κατάσταση για τον Αλγόριθμο του Zoraster	46
Σχήμα 3.10 Τοποθέτηση ετικέτας σημειακού αντικειμένου	48
Σχήμα 3.11 Ο ελεύθερος χώρος μεταξύ των ορθογωνίων $r$ και $q$	52
Σχήμα 4.1 Παράδειγμα ομαδοποίησης ετικετών κατά το δεύτερο στάδιο της αρχικοποίησης	62
Σχήμα 4.2 Παράδειγμα αναπαράστασης γραμμικής ετικέτας	69
Σχήμα 4.3 Τοποθέτηση γραμμικής ετικέτας με τη χρήση καμπύλης Bézier	70
Σχήμα 4.4 Τοποθέτηση γραμμικής ετικέτας χωρίς τη χρήση των πρώτων παραγώγων των καμπυλών Bézier	72
Σχήμα 4.5 Τοποθέτηση γραμμικής ετικέτας με ανάποδη και σωστή κλίση	74
Σχήμα 4.6 Παράδειγμα περιστροφής χαρακτήρα κατά γωνία $\theta$	75
Σχήμα 4.7 Παράδειγμα εφαρμογής του αλγορίθμου επίλυσης του ΠΒΤΕ για τις γραμμικές ετικέτες	78
Σχήμα 5.1 Η EINAI ιεραρχία των κλάσεων ετικετών	81
Σχήμα 5.2 Παράδειγμα ετικέτας και μεγενθυμένου ορθογωνίου	83

Σχήμα 5.3 Πιθανές τοποθετήσεις ετικετών εικονίδιο	85
Σχήμα 5.4 Ετικέτα κειμένου που ανήκει σε θέση	86
Σχήμα 5.5 Ετικέτες κειμένου που αντιστοιχίζονται με διαδρομές	88
Σχήμα 5.6 Ετικέτα περιοχής	89
Σχήμα 5.7 Κουτί διαλόγου για τη λειτουργία <i>Object – Properties</i>	92
Σχήμα 5.8 Κουτί διαλόγου μεταβολής του στυλ σχεδιασμού	93
Σχήμα 5.9 Κουτί διαλόγου για τη διαχείριση των ετικετών	93
Σχήμα 5.10 Φύλλο μάγου για την επιλογή του τύπου της ετικέτας	94
Σχήμα 5.11 Φύλλο μάγου για την εισαγωγή του κειμένου της ετικέτας	95
Σχήμα 5.12 Σελίδα διαχείρισης ετικετών με τοποθετημένη ετικέτα κειμένου	95
Σχήμα 5.13 Φύλλο μάγου για την εισαγωγή ετικέτας εικονίδιο	96
Σχήμα 5.14 Κουτί διαλόγου επιλογής αρχείου με εικονίδιο	96
Σχήμα 5.15 Φύλλο μάγου για την εισαγωγή ετικετών εικονίδιο με επιλεγμένο το εικονίδιο που επιθυμούμε	97
Σχήμα 5.16 Σελίδα διαχείρισης ετικετών με τοποθετημένη ετικέτα εικονίδιο	97
Σχήμα 6.1 Κυκλική τοποθέτηση εικονιδίων	104
Σχήμα 6.2 Γραμμική τοποθέτηση εικονιδίων	104
Σχήμα 6.3 Τοποθέτηση εικονιδίων σε πλέγμα	104

# Κατάλογος Πινάκων

Πίνακας 2.1 Ρουτίνες μέλη της κλάσης <i>CMapObj</i>	19
Πίνακας 2.2 Τα πεδία των Γεωγραφικών Αντικειμένων και η σημασία τους	19
Πίνακας 2.3 Τα πεδία της κλάσης <i>CLayer</i>	24
Πίνακας 2.4 Τα χαρακτηριστικά της κλάσης <i>CMSDoc</i>	25
Πίνακας 2.5 Πεδία της κλάσης <i>CDrawStyle</i>	26
Πίνακας 2.6 Τα πεδία της κλάσης <i>CLayerObjProperties</i>	27
Πίνακας 2.7 Τα πεδία της κλάσης <i>CBoundLayer</i>	27
Πίνακας 2.8 Πεδία της κλάσης <i>CLayerObjPool</i>	28
Πίνακας 2.9 Πεδία της κλάσης <i>CQTreeNode</i>	30
Πίνακας 2.10 Πεδία της κλάσης <i>CQuadTree</i>	31
Πίνακας 5.1 Τα πεδία της κλάσης <i>CLabel</i> και η σημασία τους	81
Πίνακας 5.2 Τα πεδία της κλάσης <i>COID</i>	82
Πίνακας 5.3 Ρουτίνες μέλη της κλάσης <i>COID</i>	82
Πίνακας 5.4 Οι ρουτίνες της κλάσης <i>CLabel</i>	83
Πίνακας 5.5 Οι ρουτίνες της κλάσης <i>CIconLabel</i>	84
Πίνακας 5.6 Οι ρουτίνες μέλη της κλάσης <i>CTextLabel</i>	85
Πίνακας 5.7 Οι ρουτίνες μέλη της κλάσης <i>CPartitionLabel</i>	86
Πίνακας 5.8 Οι ρουτίνες μέλη της κλάσης <i>CRouteLabel</i>	87
Πίνακας 5.9 Οι ρουτίνες μέλη της κλάσης <i>CRegionLabel</i>	89
Πίνακας 5.10 Πεδία της κλάσης <i>CRoute</i> για την υποστήριξη των καμπυλών <i>Bézier</i>	91
Πίνακας 5.11 Ρουτίνες της κλάσης <i>CRoute</i> για την υποστήριξη των καμπυλών <i>Bézier</i>	91
Πίνακας 5.12 Πεδίο της κλάσης <i>CLayerObjProperties</i> για την υποστήριξη των ετικετών	91
Πίνακας 5.13 Ρουτίνες της κλάσης <i>CLayerObjProperties</i> για την υποστήριξη των ετικετών	91



# Κεφάλαιο 1

## Εισαγωγή

### 1.1 Γενικά

Ο άνθρωπος ανέκαθεν επιζητούσε την εύρεση νέων μέσων, με τη χρησιμοποίηση των οποίων θα μπορούσε να αποτυπώσει τις σκέψεις και την εργασία του, τα οποία στη συνέχεια θα μπορούσε να επεξεργαστεί, οργανώσει και διαδώσει μέσω αυτών. Μία μέθοδος παρουσίασης των παραπάνω πληροφοριών είναι η χρήση εικόνων και σχεδίων. Παραδείγματα τέτοιων παρουσιάσεων είναι η ζωγραφική επικόνιση μίας περιοχής, ένα διάγραμμα παρουσίασης στατιστικών στοιχείων κ.α.. Γενικά, οι προαναφερόμενες παρουσιάσεις προσφέρουν ένα φιλικότερο και πιο άμεσο τρόπο ανάγνωσης. Στη σημερινή εποχή οι ηλεκτρονικοί υπολογιστές με τη ταχεία βελτίωση της ισχύος τους και τη συνεχή πτώση του κόστους τους αποτελούν το ιδανικό εργαλείο για την ανάπτυξη και εκμετάλλευση συστημάτων που καλύπτουν τις παραπάνω ανάγκες με τη χρήση γραφικών αντικειμένων.

Ένα από τα κυριότερα στοιχεία που είναι απαραίτητο να διαθέτει μία εφαρμογή ανάπτυξης γραφικών αντικειμένων είναι η τοποθέτηση κειμένου και ορισμένων γραφικών συμβόλων, τα οποία μας επιτρέπουν να δώσουμε στον αναγνώστη διάφορες πληροφορίες σχετικές με τα αντικείμενα που απεικονίζονται. Η εμφάνιση των προαναφερόμενων πληροφοριών επιτυγχάνεται με τη χρήση **ετικετών**. Οι ετικέτες αποτελούν ένα σύντομο και λιτό τρόπο παρουσίασης πληροφοριών, ο οποίος εμφανίζει ένα πλεονέκτημα: την εύκολη και άμεση διασύνδεση των πληροφοριών αυτών με το αντικείμενο στο οποίο ανήκουν.

Μία κατηγορία εφαρμογών ανάπτυξης γραφικών αντικείμενων, στα οποία η χρήση των ετικετών κρίνεται αναγκαία, είναι τα **Συστήματα Γεωγραφικών Δεδομένων – Σ.Γ.Δ. (Geographic Information Systems – G.I.S.)** και **Εφαρμογών Γεωγραφικών Δεδομένων – Ε.Γ.Δ.** Τα συγκεκριμένα συστήματα προσφέρουν την δυνατότητα κατασκευής χαρτών που περιγράφουν κάποιες περιοχές (π.χ. γεωφυσικούς σχηματισμούς μίας περιοχής, πολεοδομικές κατασκευές μίας πόλης κ.α.). Επιπλέον, μέσω της διαχείρισης βάσεων δεδομένων που περιέχουν, προσφέρουν διάφορες πληροφορίες που περιγράφουν τις ιδιότητες των γεωγραφικών στοιχείων, όπως τον πληθυσμό μίας πόλης, το είδος της καλλιέργειας μίας περιοχής κ.α.

## **1.2 Σκοπός και Αναγκαιότητα της Διπλωματικής Εργασίας**

Στο [Ανασ95] υλοποιήθηκε μία βασική πλατφόρμα ανάλυσης και παρουσίασης γεωγραφικών δεδομένων, η εφαρμογή **Map Studio**. Η εφαρμογή αυτή κατασκευάστηκε για το περιβάλλον των Microsoft Windows και εξασφαλίζει:

- Ευκολία χρήσης.
- Επεξεργασία και προβολή των δεδομένων σε περιβάλλον WYSIWYG<sup>1</sup>.
- Κατασκευή καλαίσθητων χαρτών.
- Επεκτασιμότητα.
- Συνδυασμός Διανυσματικών και raster Χωρικών Δεδομένων.

Στα πλαίσια αυτής της διπλωματικής εργασίας προσθέσαμε στο Map Studio τη δυνατότητα της διαχείρισης ετικετών. Κατασκευάστηκαν τα απαραίτητα εργαλεία έτσι ώστε να προσφέρεται η δυνατότητα εισαγωγής ετικετών κειμένου σε όλους τους τύπους χωρικών γεωγραφικών δεδομένων που υποστηρίζει το Map Studio (σημεία, γραμμές και περιοχές). Καθένας από τους παραπάνω τύπους γεωγραφικών αντικείμενων παρουσιάζει ορισμένες ιδιαιτερότητες ως προς τον τρόπο προβολής

---

<sup>1</sup> What You See Is What You Get

τους πάνω στο χάρτη. Σε αυτές τις ιδιαιτερότητες βασίστηκαν και οι μέθοδοι παρουσίασης των ετικετών, έτσι ώστε να είναι όσο το δυνατόν ευκολότερη η αντιστοίχιση κάθε ετικέτας με το γραφικό αντικείμενο του χάρτη στο οποίο ανήκει. Οι ετικέτες κατηγοριοποιούνται ως ετικέτες σημειακών αντικειμένων, γραμμικών αντικειμένων και περιοχών, ανάλογα με τον τύπο του αντικειμένου στο οποίο ανήκουν.

Πιο συγκεκριμένα, οι ετικέτες σημειακών αντικειμένων τοποθετούνται σε μία περιοχή γύρω από το σημείο στο οποίο ανήκουν κατά τέτοιο τρόπο ώστε μία πλευρά της ετικέτας να εφάπτεται πάντα με το συγκεκριμένο σημείο. Υπάρχουν δύο τρόποι τοποθέτησης των συγκεκριμένων ετικετών, η *διακριτή τοποθέτηση* κατά την οποία η ετικέτα τοποθετείται σε προκαθορισμένες θέσεις γύρω από το σημείο και η *συνεχής τοποθέτηση* στην οποία η ετικέτα μπορεί να τοποθετηθεί οπουδήποτε γύρω από το σημείο μέσα στα όρια μίας συγκεκριμένης περιοχής. Στη παρούσα διπλωματική εργασία επιλέξαμε να χρησιμοποιήσουμε το συνεχές μοντέλο τοποθέτησης, το οποίο προσφέρει στον χρήστη της εφαρμογής περισσότερες δυνατές θέσεις τοποθέτησης των ετικετών, ενώ, σύμφωνα με μετρήσεις που πραγματοποιήθηκαν στο [Μουμ97], επιφέρει καλύτερα αποτελέσματα στον αλγόριθμο αυτόματης τοποθέτησης των ετικετών, στον οποίο θα αναφερθούμε στη συνέχεια.

Οι ετικέτες γραμμικών αντικειμένων τοποθετούνται κατά τέτοιο τρόπο ώστε να ακολουθούν πάντα την καμπυλότητα της γραμμής στην οποία ανήκουν. Η τοποθέτηση των συγκεκριμένων ετικετών μπορεί να γίνει είτε κατά μήκος της γραμμής στην οποία ανήκουν, είτε κατά μήκος μίας καμπύλης που υπολογίζεται από τη συγκεκριμένη γραμμή. Στη παρούσα διπλωματική εργασία επιλέξαμε να χρησιμοποιήσουμε το μοντέλο των καμπυλών Bézier ως οδηγούς τοποθέτησης των ετικετών γραμμικών αντικειμένων.

Τέλος, οι ετικέτες που ανήκουν σε περιοχές τοποθετούνται στο κέντρο της κάθε περιοχής, έτσι ώστε να είναι ευδιάκριτη η αντιστοίχιση κάθε ετικέτας με την περιοχή στην οποία ανήκει.

Επιπλέον, ο κατασκευαστής του χάρτη μπορεί να επανατοποθετήσει τις ετικέτες σημειακών και γραμμικών αντικειμένων, μέσα στα όρια που θέσαμε στην προηγούμενη παράγραφο. Με αυτό τον τρόπο δίνονται στο χρήστη της εφαρμογής

μεγαλύτερα περιθώρια αξιοποίησης του μοντέλου ετικετών που υλοποιήσαμε και η δυνατότητα να κατασκευάσει πιο καλαίσθητους χάρτες σύμφωνα με τις επιθυμίες του. Ακόμη, οι ετικέτες μπορούν να τροποποιηθούν σύμφωνα με τις επιλογές του χρήστη τόσο κατά το μέγεθός τους όσο και κατά το χρώμα τους, μέσω ενός συνόλου επιλογών που προσφέρονται στο χρήστη.

Τέλος, παρέχεται η δυνατότητα εισαγωγής γραφικών συμβόλων (εικονιδίων) στα χωρικά δεδομένα εκτός από ετικέτες που περιέχουν κείμενο. Με τη χρήση των εικονιδίων μπορούμε να αποτυπώσουμε στο χάρτη πληροφορίες που δεν είναι δυνατόν να αποδοθούν από τις ετικέτες κειμένου, όπως για παράδειγμα τις παρεχόμενες υπηρεσίες ενός ξενοδοχείου, τους τύπους των τουριστικών αξιοθεάτων μίας περιοχής κ.α.

Ένα από τα κυριότερα προβλήματα που παρουσιάζεται σε εφαρμογές που χρησιμοποιούν ετικέτες είναι η αποφυγή επικαλύψεων που παρουσιάζονται μεταξύ των ετικετών και των υπόλοιπων γραφικών αντικειμένων του χάρτη (συμπεριλαμβανομένων και των υπόλοιπων ετικετών), ώστε να έχουμε το καλύτερο δυνατό, οπτικό αποτέλεσμα. Το παραπάνω πρόβλημα είναι γνωστό ως **Πρόβλημα Βέλτιστης Τοποθέτησης Ετικετών – ΠΒΤΕ**.

Η ανάλυση της πολυπλοκότητας του ΠΒΤΕ που εξετάζεται στο [Chms95] αποκαλύπτει ότι πρόκειται για NP-Hard πρόβλημα. Συνεπώς, καταλήγουμε στο συμπέρασμα ότι δεν μπορεί να κατασκευαστεί αλγόριθμος, ο οποίος θα δίνει βέλτιστη λύση στο ΠΒΤΕ σε αποδεκτά χρονικά πλαίσια. Άρα η καλύτερη προσέγγιση της λύσης του ΠΒΤΕ μπορεί να επιτευχθεί με τη χρήση κατάλληλων ευριστικών μεθόδων, οι οποίες έχουν ως στόχο την εύρεση αποδεκτών λύσεων στο πρόβλημα που εξετάζουμε σε σύντομα, σχετικά, χρονικά περιθώρια. Στα πλαίσια της παρούσας διπλωματικής εργασίας προχωρήσαμε στην υλοποίηση τέτοιων αλγορίθμων τόσο για τις ετικέτες σημειακών αντικειμένων όσο και για τις ετικέτες γραμμικών αντικειμένων, οι οποίοι επιτυγχάνουν την αυτόματη τοποθέτηση των ετικετών σε θέσεις στις οποίες παρουσιάζουν την ελάχιστη δυνατή (ή μηδαμινή) επικάλυψη με άλλα γραφικά αντικείμενα.

Για τις ετικέτες σημειακών αντικειμένων αναπτύξαμε έναν αλγόριθμο που στηρίζεται στη μέθοδο **Προσομοίωσης Ανόπτησης** που παρουσιάστηκε στο

[Chms95]. Αυτή η ευριστική μέθοδος δεν προσπαθεί να πετύχει μια βέλτιστη λύση στο ΠΒΤΕ. Σκοπός της είναι με συνεχείς βελτιώσεις να δώσει ένα καλαίσθητο αποτέλεσμα στην τελική παρουσίαση του χάρτη.

Για τις ετικέτες γραμμικών αντικειμένων λόγω του γεγονότος ότι παρουσιάζουν σημαντικές διαφορές από τις ετικέτες σημειακών αντικειμένων, ο προηγούμενος αλγόριθμος ήταν αδύνατο να χρησιμοποιηθεί. Συνεπώς, υλοποιήσαμε έναν νέο αλγόριθμο επίλυσης του ΠΒΤΕ για τις συγκεκριμένες ετικέτες λαμβάνοντας υπόψιν τις ιδιαιτερότητές τους.

### **1.3 Δομή της Διπλωματικής Εργασίας**

Στο Κεφάλαιο 2 θα παρουσιάσουμε μία σύντομη περίληψη του περιβάλλοντος ανάπτυξης εφαρμογών γεωγραφικών δεδομένων Map Studio στην οποία προσθέσαμε την υλοποίηση των ετικετών και τους αλγόριθμους επίλυσης του ΠΒΤΕ. Επιπλέον, Θα αναφερθούμε στις κλάσεις που χρησιμοποιήθηκαν για την υλοποίηση των αντικειμένων της εφαρμογής.

Στο Κεφάλαιο 3 της εργασίας θα προχωρήσουμε σε μία περιληπτική περιγραφή αλγορίθμων που έχουν παρουσιαστεί σε προηγούμενες δημοσιεύσεις σχετικά με την επίλυση του ΠΒΤΕ και που ορισμένοι από αυτούς χρησιμοποιήθηκαν εν μέρει στην παρούσα διπλωματική εργασία.

Στο Κεφάλαιο 4 θα αναφερθούμε διεξοδικά στους αλγόριθμους που υλοποιήσαμε για την επίλυση του ΠΒΤΕ τόσο για τις Ετικέτες των Σημειακών Αντικειμένων όσο και για τις ετικέτες των γραμμικών αντικειμένων, καθώς και στην υλοποίηση των καμπυλών Bézier που χρησιμοποιούνται για την εξομάλυνση των πολυγωνικών γραμμών που αποτελούν τα γραμμικά αντικείμενα του χάρτη και την τοποθέτηση των ετικετών πάνω σε αυτές.

Στο Κεφάλαιο 5 θα προχωρήσουμε στην αναλυτική περιγραφή του Μοντέλου Αντικειμένων Ετικετών, στη σύνδεσή του με την υπάρχουσα ιεραρχία κλάσεων της εφαρμογής Map Studio, καθώς και στις αλλαγές οι οποίες ήταν αναγκαίο να γίνουν στις δομές και τα αντικείμενα της συγκεκριμένης εφαρμογής.

Κλείνοντας, στο Κεφάλαιο 6 παρουσιάζουμε μια ανακεφαλαίωση όσων υλοποιήθηκαν στη διάρκεια της διπλωματικής εργασίας και παρουσιάζουμε τα σχέδια για μελλοντική επέκτασή της.

## Κεφάλαιο 2

# Περιγραφή του Περιβάλλοντος Ανάπτυξης Εφαρμογών Γεωγραφικών Δεδομένων Map Studio

### **2.1 Βασικά Αντικείμενα**

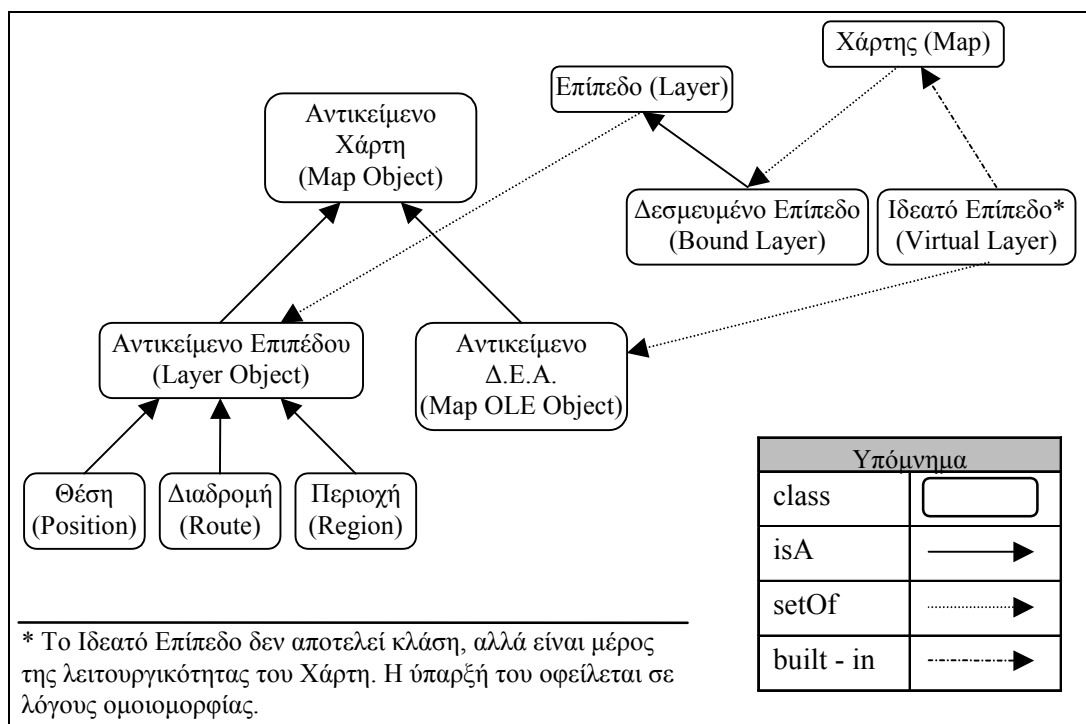
Στο Κεφάλαιο αυτό θα αναφερθούμε περιληπτικά στο περιβάλλον ανάπτυξης εφαρμογών γεωγραφικών δεδομένων Map Studio που υλοποιήθηκε στο [Ανασ95] και πιο συγκεκριμένα στο μοντέλο αντικειμένων χρήστη και στις δομές υποστήριξής τους. Η αναφορά αυτή είναι αναγκαία για να μπορέσει αρχικά ο αναγνώστης να κατανοήσει τις δομές αυτές και τη χρησιμότητά τους, οπότε στη συνέχεια να γίνουν πιο κατανοητές οι δομές που προστέθηκαν για την υλοποίηση των ετικετών.

Τα βασικά αντικείμενα της εφαρμογής είναι:

- Ο **Χάρτης** που είναι ο βασικός τύπος εγγράφου και αποτελείται από ένα σύνολο επιπέδων πληροφορίας.
- Το **Επίπεδο** που περιέχει ένα σύνολο από Αντικείμενα Επιπέδου, τα οποία συνήθως περιέχουν ομοειδή πληροφορία (π.χ. Επίπεδο οδικού δικτύου).

- Το **Αντικείμενο Χάρτη** που ορίζει τη βασική λειτουργικότητα για τα αντικείμενα που μπορεί να περιέχονται σε ένα Χάρτη (ή στα Επίπεδά του).
- Το **Αντικείμενο Επιπέδου** που είναι το Αντικείμενο Χάρτη που περιέχεται σε ένα Επίπεδο και αναπαριστά την γεωγραφική πληροφορία.
- Το **Αντικείμενο Δ.Ε.Α. (OLE)** που αποτελεί την ειδική κατηγορία Αντικειμένων Χάρτη που αναπαριστά τη μη-γεωγραφική πληροφορία του (π.χ. πολυμέσα).

Στο Σχήμα 2.1 που ακολουθεί, παρουσιάζονται περιληπτικά τα αντικείμενα που προαναφέραμε και την ιεραρχική δομή που υπάρχει μεταξύ τους. Στις ενότητες που ακολουθούν θα γίνει μια πιο εκτενής αναφορά στη λειτουργικότητα και τα χαρακτηριστικά τους.

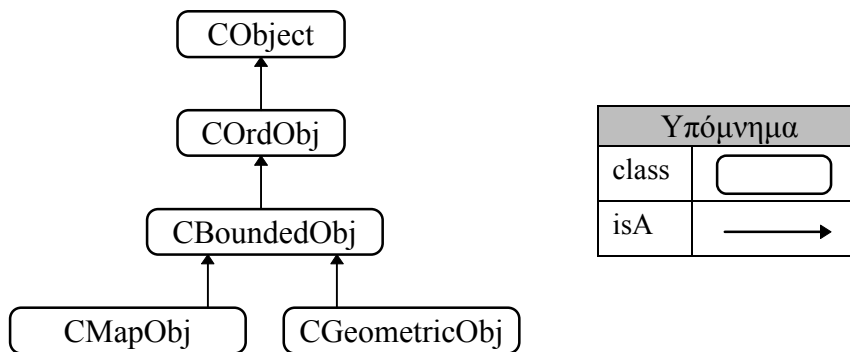


Σχήμα 2.1 - Οι κλάσεις αντικειμένων



## 2.2 Κλάσεις Γενικής Χρήσης

Σε αυτή την ενότητα θα παρουσιάσουμε τις κλάσεις γενικής χρήσης που υλοποιήθηκαν με σκοπό να καθοριστούν ορισμένες κοινές, βασικές λειτουργίες που χρησιμοποιούνται για την ικανοποίηση των συνθετότερων απαιτήσεων του χρήστη. Στο Σχήμα 2.2 παρουσιάζεται η EINAI ιεραρχία αυτών των βασικών κλάσεων.



Σχήμα 2.2 - Η EINAI ιεραρχία των κλάσεων Γενικής Χρήσης

Η κλάση **CObject** είναι η βασική κλάση που χρησιμοποιείται από την MFC και συνεπώς είναι η βασική κλάση της ιεραρχίας.

Η αμέσως επόμενη κλάση στην ιεραρχία είναι η κλάση **COrdObj** και είναι η υλοποίηση του προτύπου του διατεταγμένου αντικειμένου. Τα στιγμιότυπα αυτής της κλάσης (και των κλάσεων που απορρέουν από αυτή) δίνουν τη δυνατότητα σε δομές συλλογών (π.χ. πίνακες, λίστες) να διατάσσουν τα στοιχεία τους με τρόπο που ορίζεται από τα ίδια τα αντικείμενα.

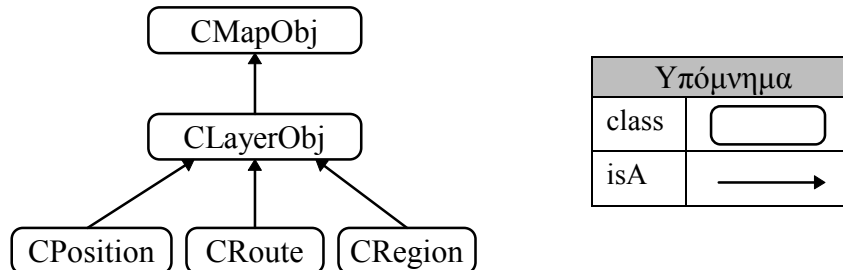
Η κλάση **CObject** αποτελεί τη μεταφορά του προτύπου του διδιάστατου αντικειμένου που περιορίζεται από ένα περιγεγραμμένο ορθογώνιο.

Η κλάση **CMapObj** είναι η βασική κλάση από την οποία απορρέουν τα (γεωγραφικά και μη) αντικείμενα της εφαρμογής. Τα αντικείμενα αυτά θα αναλυθούν στη επόμενη ενότητα.

Τέλος, η κλάση **CGeometricObj** αποτελεί τη βασική κλάση μίας ιεραρχίας γεωγραφικών σχημάτων που χρησιμοποιούνται για να υποβοηθήσουν ορισμένες λειτουργίες της εφαρμογής.

## 2.3 Γεωγραφικά Αντικείμενα

Η βάση της ιεραρχίας των γεωγραφικών (και μη) αντικειμένων της εφαρμογής είναι η κλάση CMapObj, όπως φαίνεται και στο Σχήμα 2.3.



Σχήμα 2.3 - Κλάσεις Γεωγραφικών Αντικειμένων

Το πεδίο της συγκεκριμένης κλάσης **m\_oid** προσδιορίζει μοναδικά το κάθε αντικείμενο (αναγνωριστικό) και χρησιμοποιείται για αυτό το λόγο ως πρωτεύον κλειδί στη βάση δεδομένων που περιέχεται. Ο Πίνακας 2.1 που ακολουθεί, παρουσιάζει τις ρουτίνες μέλη της κλάσης CMapObj:

Ρουτίνα(-ες)	Λειτουργία
<b>GetObjID()</b>	Ανακτά την τιμή του αναγνωριστικού
<b>SetObjID(OID)</b>	Καθορίζει την τιμή του αναγνωριστικού
<b>GetClassName()</b>	Επιστρέφει το όνομα της κλάσης
<b>Clone()</b>	Κατασκευάζει ένα ακριβές αντίγραφο του αντικειμένου (με το ίδιο αναγνωριστικό)
<b>Compare(other)</b>	Καθορίζει τον τρόπο διάταξης σε κλάσεις συλλογών - (Αύξουσα διάταξη σύμφωνα με το πρωτεύον κλειδί)
<b>Serialize(archive)</b>	Αποθηκεύει ή ανακτά τα δεδομένα του αντικειμένου (το ID) προς – από το δοθέν αποθηκευτικό μέσο
<b>OnOpen(view)</b>	Καθορίζει την συμπεριφορά του αντικειμένου όταν πατηθεί το δεξί πλήκτρο του mouse πάνω στο αντικείμενο στην δοθείσα view
<b>GetBoundRectEx(dc)</b>	Ανακτά το περιγεγραμμένο ορθογώνιο του σχήματος σε “λογικές” συντεταγμένες λαμβάνοντας υπ’ όψη το συντελεστή μεγέθυνσης / σμίκρυνσης που έχει επιλεγθεί στο δοθέν Πρότυπο Συσκευής καθώς και άλλα χαρακτηριστικά του

<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
<b>GetObjID()</b>	Ανακτά την τιμή του αναγνωριστικού
<b>Draw(dc, gdiobjects)</b>	Σχεδιάζει το αντικείμενο στο δοθέν Πρότυπο Συσκευής <sup>2</sup> (Device Context - CDC) χρησιμοποιώντας τα δοθέντα χαρακτηριστικά σχεδίασης

Πίνακας 2.1 - Ρουτίνες μέλη της κλάσης CMapObj

Η κλάση CMapObj περιλαμβάνει τόσο λειτουργίες γενικής φύσεως όσο και λειτουργίες που σχετίζονται με τη σχεδίαση του αντικειμένου σε ένα πρότυπο συσκευής.

Επόμενη στην ιεραρχία είναι η κλάση **CLayerObj** που αποτελεί τη μεταφορά του προτύπου Γεωγραφικού Αντικειμένου. Το πεδίο **m\_pLayer** είναι ένας δείκτης στο Επίπεδο στο οποίο ανήκει το αντικείμενο τύπου CLayerObj. Μέσω αυτού του πεδίου μπορεί το αντικείμενο να προσπελάσει δυναμικά το Επίπεδο στο οποίο ανήκει.

Τα αντικείμενα των υποκλάσεων της CLayerObj είναι τα Αντικείμενα Χρήστη (User Items) της εφαρμογής. Τα αντικείμενα αυτά χρησιμοποιούνται για την αναπαράσταση της γεωγραφικής πληροφορίας στο Χάρτη και τα Επίπεδα που τον απαρτίζουν. Οι κλάσεις αυτές είναι η **CPosition** (Θέση), **CRoute** (Διαδρομή) και **CRegion** (Περιοχή). Ο Πίνακας 2.2 δείχνει τα πεδία που ορίζονται σε αυτές τις κλάσεις και η σημασία τους:

<b>Κλάση</b>	<b>Πεδία</b>	<b>Σημασία</b>	<b>Πεδία Σύνδεσης</b>
<b>CPosition</b>	<i>m_ptPos</i>	Οι συντεταγμένες του σημείου που αντιστοιχεί στην “Θέση”	<i>m_idFirstRoute</i>
<b>CRoute</b>	<i>m_idFromPos</i> <i>m_idToPos</i> <i>m_nInnerPoints</i> <i>m_pInnerPoints</i>	Η Θέση εκκίνησης της διαδρομής Η Θέση τερματισμού της διαδρομής Το πλήθος των “εσωτερικών” σημείων της Οι συντεταγμένες των “εσωτερικών” σημείων της	<i>m_idLeftRgn</i> <i>m_idRightRgn</i> <i>m_idFromPosNextRoute</i> <i>m_idToPosNextRoute</i>
<b>CRegion</b>			<i>m_idFirstRoute</i>

Πίνακας 2.2 - Τα πεδία των Γεωγραφικών Αντικειμένων και η σημασία τους

<sup>2</sup> Το Πρότυπο Συσκευής αποτελεί τη μεταφορά του προτύπου της συσκευής εμφάνισης. Μπορεί να αναφέρεται στην οθόνη του υπολογιστή, στον εκτυπωτή, ή σε οποιαδήποτε άλλη συσκευή εξόδου, συμπεριλαμβανομένων και των Windows Metafiles. Τα Windows Metafiles αποτελούν μηχανισμό αποθήκευσης των γραφικών λειτουργιών που χρησιμοποιήθηκαν για τον σχεδιασμό μιας εικόνας σε ειδικά αρχεία.

Τα πεδία σύνδεσης διατηρούν συγκεκριμένες πληροφορίες που απαιτούνται για τη κατασκευή του γραφήματος και θα τις αναλύσουμε στην επόμενη ενότητα.

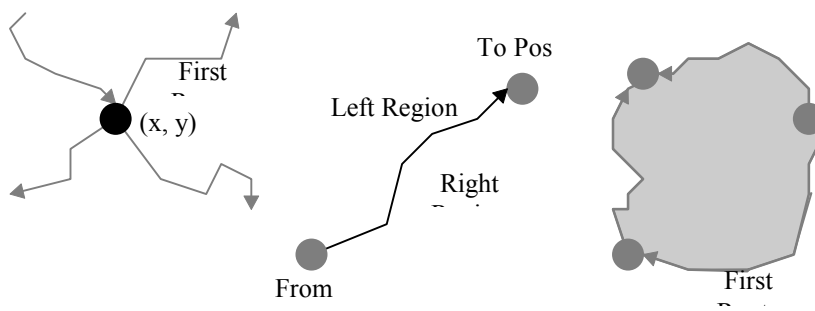
Η **CPosition (Θέση)** συμβολίζει έναν κόμβο σε ένα γράφημα. Το πεδίο **m\_ptPos** προσδιορίζει τις συντεταγμένες του σημείου στο επίπεδο.

Η **CRoute (Διαδρομή)** συμβολίζει μια κατευθυνόμενη ακμή σε ένα γράφημα, η οποία συνδέει δύο κόμβους (Θέσεις). Τα πεδία **m\_idFromPos** και **m\_idToPos** είναι οι αναγνωριστικές τιμές του αρχικού και τελικού κόμβου, αντίστοιχα, ενώ τα πεδία **m\_nInnerPoints** και **m\_pInnerPoints** περιέχουν το πλήθος των ενδιάμεσων σημείων και τις συντεταγμένες τους αντίστοιχα. Τα ενδιάμεσα αυτά σημεία, τα οποία δεν είναι Θέσεις, ορίζουν το σχήμα της Διαδρομής.

Τέλος, η **CRegion (Περιοχή)** ορίζεται ως:

*το κομμάτι του επιπέδου που περικλείεται από ένα κλειστό μονοπάτι διαδοχικών Διαδρομών που αποτελεί ένα απλό πολύγωνο<sup>3</sup>.*

Η Περιοχή δεν έχει δικά της πεδία πληροφορίας, γιατί οι απαραίτητες πληροφορίες για την αναπαράστασή της στο γράφημα περιέχονται στις Διαδρομές που την ορίζουν.



Σχήμα 2.4 - Παράδειγμα Γεωγραφικών Αντικειμένων

<sup>3</sup> Ένα πολύγωνο καλείται *απλό* αν δεν υπάρχει ζεύγος μη διαδοχικών πλευρών του που να τέμνονται. Ένα απλό πολύγωνο τμηματοποιεί το χώρο σε δύο ξένες μεταξύ τους περιοχές: το *εσωτερικό* (φραγμένο) και το *εξωτερικό* (μη φραγμένο) [Prep85-σελ.18]. Το πολύγωνο αποτελεί το *περίβλημα* της *εσωτερικής* περιοχής. Αυτή η *εσωτερική* περιοχή είναι αυτό που ορίζεται ως Περιοχή στο Map Studio.

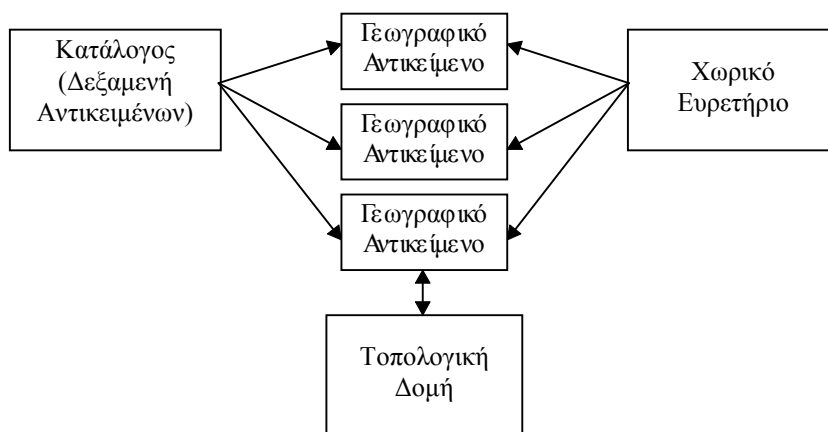
## 2.4 Επίπεδο

Το **Επίπεδο (Layer)** περιέχει ένα σύνολο Γεωγραφικών Αντικειμένων. Τα αντικείμενα αυτά τοποθετούνται σε έναν **Κατάλογο (Δεξαμενή Αντικειμένων - Object Pool)**, ο οποίος ευθύνεται για την αποθήκευση και ανάκτησή τους από το αποθηκευτικό μέσο. Επιπλέον, στο Επίπεδο έχει οριστεί ακόμη μία δομή, το **Χωρικό Ευρετήριο (Spatial Index)**, η οποία επιταχύνει την εύρεση των αντικειμένων με βάση χωρικά στοιχεία, όπως η επιλογή των αντικειμένων που περικλείονται ή τέμνονται από ένα δοσμένο ορθογώνιο. Οι παραπάνω δομές θα αναλυθούν λεπτομερέστερα στην ενότητα 2.6.

Η τοπολογική δομή του Επιπέδου προσδιορίζεται με δύο (λογικά) γραφήματα τα οποία συσχετίζουν:

- Το πρώτο, Θέσεις του Επιπέδου με τις Διαδρομές που τις έχουν ως αφετηρία ή τέλος τους.
- Το δεύτερο, Περιοχές του Επιπέδου με τις Διαδρομές που τις ορίζουν.

Στο επόμενο Σχήμα απεικονίζεται η δομή του Επιπέδου:

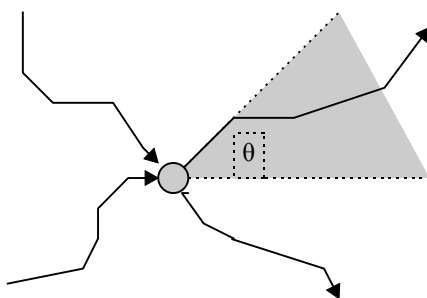


Σχήμα 2.5 - Η δομή του επιπέδου

Τα Πεδία Σύνδεσης των Γεωγραφικών Αντικειμένων, που αναφέραμε στην προηγούμενη ενότητα, διαμορφώνουν την τοπολογική δομή του Επιπέδου. Πιο

συγκεκριμένα, οι Διαδρομές δαιτηρούνται σε μια απλά-συνδεδεμένη κυκλική λίστα, μια για κάθε Θέση, διατεταγμένη κατά τη γωνία που σχηματίζεται από:

- το ευθύγραμμο τμήμα της Διαδρομής που εφάπτεται με τη συγκεκριμένη Θέση και
- την παράλληλη προς τον άξονα  $X$ , ευθεία που διέρχεται από τη Θέση αυτή.



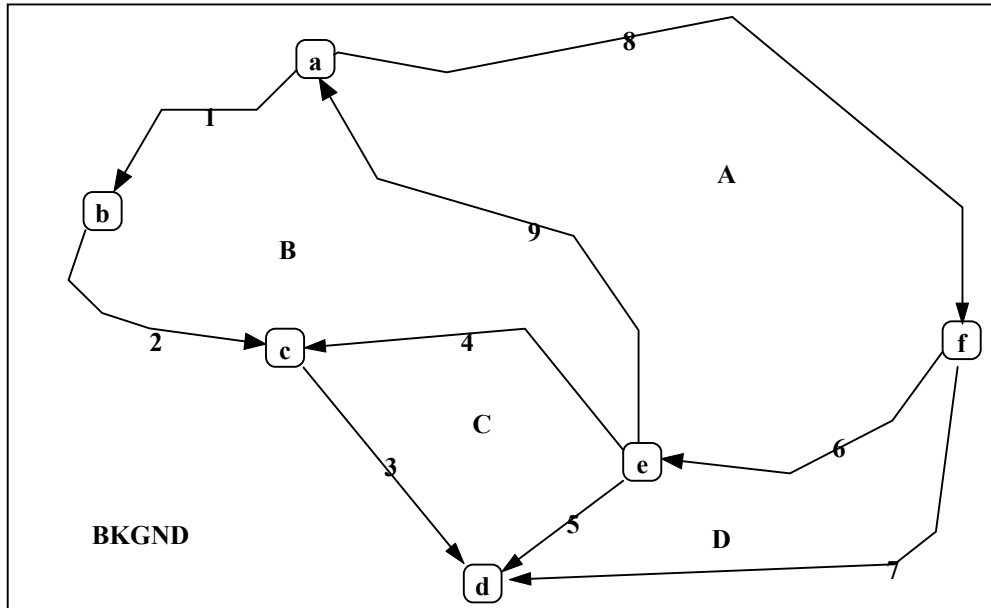
Σχήμα 2.6 - Γωνία διαδρομής με άξονα παράλληλο του  $X$

Η κυκλική αυτή λίστα αποτελείται από τα πεδία **m\_idFirstRoute** της κλάσης CPosition και τα πεδία **m\_idFromPosNextRoute** και **m\_idToPosNextRoute** της κλάσης CRoute.

Αντίστοιχα, το σύνολο των Διαδρομών που σχηματίζουν μια Περιοχή αποτελείται από τις Διαδρομές που έχουν σε ένα από τα πεδία **m\_idLeftRgn** και **m\_idRightRgn** ως τιμή, την τιμή του αναγνωριστικού της συγκεκριμένης Περιοχής. Οι Διαδρομές αυτές είναι διαδοχικά συνδεδεμένες λόγω των λιστών που υπάρχουν στην αρχική και τελική Θέση της καθεμιάς. Ως αρχικό αντικείμενο σε αυτή τη λίστα ορίζεται η Διαδρομή της οποίας η τιμή του αναγνωριστικού της υπάρχει ως τιμή στο πεδίο **m\_idFirstRoute** της κλάσης CRegion της εν λόγω Περιοχής.

Στη συνέχεια, παραθέτουμε ένα παράδειγμα Επιπέδου, έτσι ώστε να γίνουν πιο κατανοητά όσα προαναφέραμε. Τα Γεωγραφικά Αντικείμενα που τοποθετήθηκαν στο Επίπεδο του παραδείγματος, αναγράφονται στους ακόλουθους Πίνακες, όπου για χάρη συντομογραφίας τα ονόματα των πεδίων των κλάσεων έχουν αντικατασταθεί ως εξής:

<i>m_oid</i>		id
<i>m_idFirstRoute</i>		head
<i>m_idFromPos</i>		from
<i>m_idToPos</i>		to
<i>m_idLeftRgn</i>		left
<i>m_idRightRgn</i>		right
<i>m_idFromPosNextRoute</i>		nFrom
<i>m_idToPosNextRoute</i>		nTo



Σχήμα 2.7 - Παράδειγμα επιπέδου

Θέσεις (CPositions)	
id	head
a	1
b	1
c	4
d	7
e	9
f	8

Περιοχές (Cpositions)	
id	head
BKGND	-
A	8
B	9
C	4
D	6

<b>Διαδρομές (CRoutes)</b>						
<b>id</b>	<b>from</b>	<b>to</b>	<b>left</b>	<b>right</b>	<b>nFrom</b>	<b>nTo</b>
1	a	b	B	BKGND	9	2
2	b	c	B	BKGND	1	3
3	c	d	C	BKGND	4	7
4	e	c	C	B	5	2
5	e	d	D	C	6	3
6	f	e	D	A	7	9
7	f	d	BKGND	D	8	5
8	a	f	BKGND	A	1	6
9	e	a	B	A	4	8

Το Επίπεδο μορφοποιείται στην κλάση **CLayer** που έχει παραχθεί από την κλάση **CObject** της **MFC**, και περιέχει τα παρακάτω πεδία:

<b>Πεδίο</b>	<b>Σημασία</b>
<b>m_idCurrent</b>	Η τρέχουσα τιμή του αναγνωριστικού που χρησιμοποιείται από την Γεννήτρια Μοναδικών Αναγνωριστικών του επιπέδου
<b>m_pDocument</b>	Δείκτης στο έγγραφο (document) που ανήκει το Επίπεδο
<b>m_strName</b>	Το όνομα του Επιπέδου
<b>m_strFileName</b>	Όνομα αρχείου για Επίπεδα που δεν ανήκουν σε Χάρτη
<b>m_obPool</b>	Ο Κατάλογος (Δεξαμενή Αντικειμένων) των Γεωγραφικών Αντικειμένων
<b>m_obQuadTree</b>	Το Χωρικό Ευρετήριο

Πίνακας 2.3 - Τα πεδία της κλάσης **CLayer**

Η κλάση **CLayerDoc** έχει ως βασική της κλάση, εκτός από τη **CDocument**, τη **CMSDoc**, χρησιμοποιώντας το μηχανισμό Πολλαπλής Κληρονομικότητας (Multiple Inheritance) της **C++**. Η κλάση **CMSDoc** έχει κατασκευαστεί για να οριστούν μέσω αυτής τα βασικά χαρακτηριστικά και η λειτουργικότητα που πρέπει να έχει ένα έγγραφο της εφαρμογής **Map Studio**. Τα χαρακτηριστικά αυτά δίνονται στον ακόλουθο πίνακα:

<b>Πεδίο</b>	<b>Σημασία</b>
<b>m_size</b>	Οι διαστάσεις του “κόσμου” που αναπαριστά το Επίπεδο, σε “λογικές” συντεταγμένες (pixels)
<b>m_options</b>	Επιλογές του χρήστη όσον αφορά την εμφάνιση των δεδομένων κ.ά.
<b>m_strImageFile</b>	Όνομα αρχείου εικόνας (Windows Bitmap) που θα χρησιμοποιηθεί ως βάση για την κατασκευή των αντικειμένων του επιπέδου
<b>m_pImage</b>	Αντικείμενο της κλάσης <b>CBMIImage</b> που αναλαμβάνει την διαχείριση του αρχείου εικόνας που ορίζει το πεδίο <b>m_strImageFile</b> και την εμφάνισή της στο παράθυρο της εφαρμογής
<b>m_listUndo, m_listRedo</b>	Λίστες “Εντολών”, που χρησιμοποιούνται για την υλοποίηση του Μοντέλου Λειτουργιών και για την υποστήριξη του μηχανισμού



Πεδίο	Σημασία
	Εξουδετέρωσης και Επαναφοράς τους (Undo/Redo).

Πίνακας 2.4 - Τα χαρακτηριστικά της κλάσης *CMSDoc*

## 2.5 Χάρτης

Ο **Χάρτης** ορίζεται ως “η σχηματική αναπαράσταση ενός συνόλου επιπέδων από τα οποία αποτελείται”.

Το Επίπεδο, όπως το ορίσαμε στην προηγούμενη ενότητα, περιέχει πληροφορία σχετικά με τη γεωμετρία και την τοπολογία των Γεωγραφικών Αντικειμένων του. Για να μπορέσουμε να κατασκευάσουμε ένα Χάρτη από Επίπεδα, σύμφωνα με τον παραπάνω ορισμό, χρειάζεται να εισάγουμε επιπρόσθετη πληροφορία σχετικά με τον τρόπο αναπαράστασης των δεδομένων. Για αυτόν το λόγο είναι απαραίτητο να μοντελοποιήσουμε την πληροφορία αυτή.

### 2.5.1 Στιλ Σχεδιασμού

Το **Στιλ Σχεδιασμού (Drawing Style)** αποτελείται από μια σειρά παραμέτρων, οι οποίες καθορίζουν τα παρουσιαστικά χαρακτηριστικά (presentation characteristics) των Γεωγραφικών Αντικειμένων που περιέχονται στα Επίπεδα ενός Χάρτη. Οι παράμετροι αυτές είναι:

- Χρώμα, Πάχος και είδος γραμμής
- Στιλ, Χρώμα, Υφή (Texture) και Μοτίβο (Pattern) του εσωτερικού ενός πολυγώνου (fill)
- Γραμματοσειρά (font) και Χρώμα κειμένου

Το Στιλ Σχεδιασμού υλοποιείται στο Map Studio από την κλάση **CDrawStyle**. Περιέχει ως πεδία της τα χαρακτηριστικά που ορίσαμε παραπάνω και δίνονται στον παρακάτω πίνακα, τα οποία καθοδηγούν την κατασκευή των επιθυμητών Πόρων<sup>4</sup> των

<sup>4</sup> Οι Πόροι (Resources) του λειτουργικού συστήματος των Microsoft Windows είναι δομές που χρησιμοποιούνται για τον σχεδιασμό σε ένα Πρότυπο Συσκευής (Device Context), τις οποίες διαχειρίζεται το τμήμα GDI.

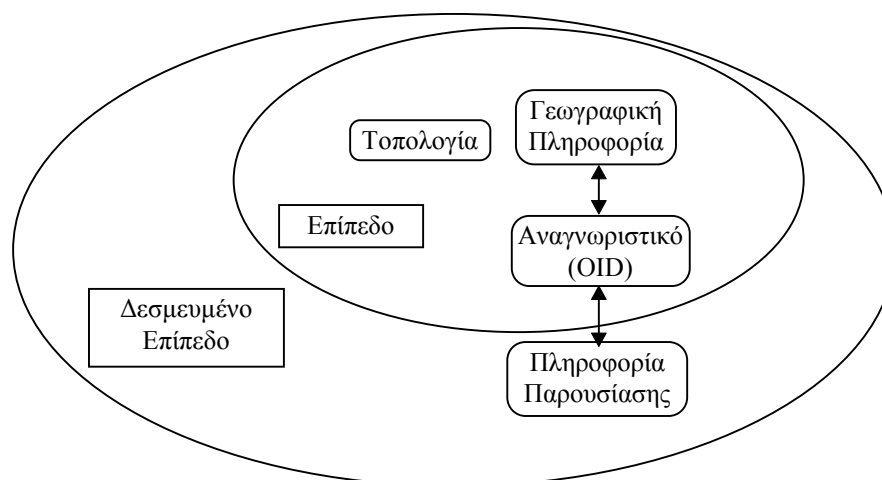
Windows που θα απαιτηθούν για το σχεδιασμό των Γεωγραφικών Αντικειμένων στην οθόνη. Οι Πόροι αυτοί είναι: Γραμματοσειρές (Fonts), Μολύβια (Pens), Πινέλα (Brushes) και Εικόνες (Bitmaps).

Πεδίο	Σημασία
<b>m_strName</b>	Συμβολοσειρά με το όνομα του Στιλ
<b>m_lfFont</b>	Δομή των Microsoft Windows που περιέχει πληροφορία για την Γραμματοσειρά που θα χρησιμοποιηθεί για την εμφάνιση κειμένου
<b>m_crFont</b>	Χρώμα (τιμή R-G-B) για το κείμενο
<b>m_lgpnPen</b>	Δομή των Microsoft Windows που περιέχει πληροφορία για το Μολύβι που θα χρησιμοποιηθεί για τον σχεδιασμό γραμμών
<b>m_nFillStyle</b>	Τύπος "γεμίσματος" του εσωτερικού των κλειστών πολυγώνων (DS_FILL_NONE / DS_FILL_SOLID / DS_FILL_TEXTURE)
<b>m_crFill</b>	Χρώμα (τιμή R-G-B) για το εσωτερικό (αν έχει επιλεγθεί συμπαγές χρώμα με το πεδίο m_nFillStyle)
<b>m_nPattern</b>	Τύπος Μοτίβου (γραμμοσκίασης) (NONE, HS_BDIAGONAL, HS_CROSS, HS_DIAGCROSS, HS_FDIAGONAL, HS_HORIZONTAL, HS_VERTICAL)
<b>m_crPattern</b>	Χρώμα (τιμή R-G-B) για το Μοτίβο
<b>m_nTexture</b>	Αναγνωριστικό του Bitmap που χρησιμοποιείται για την Υφή (Texture)

Πίνακας 2.5 - Πεδία της κλάσης CDrawStyle

## 2.5.2 Δεσμευμένο Επίπεδο

Το **Δεσμευμένο Επίπεδο** είναι ένας τύπος Επιπέδου, που εκτός της γεωγραφικής πληροφορίας που περιέχουν τα (απλά) Επίπεδα, διατηρεί επιπλέον πληροφορία για το παρουσιαστικό των Γεωγραφικών του Αντικειμένων (Σχήμα 2.8).



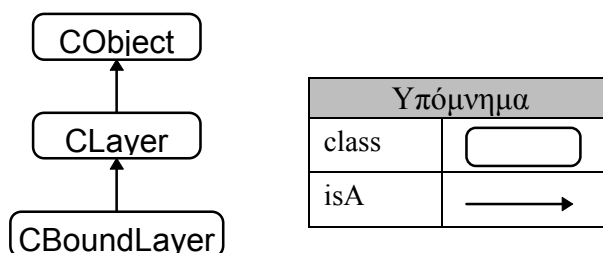
Σχήμα 2.8 - Το Δεσμευμένο Επίπεδο ως υπερσύνολο του Επιπέδου

Η κλάση **CLayerObjProperties** (ΕΙΝΑΙ υποκλάση της CObject) περιέχει αυτή την επιπλέον πληροφορία (Πίνακας 2.6). Η αντιστοίχιση των αντικειμένων της παραπάνω κλάσης με τα Γεωγραφικά Αντικείμενα του Επιπέδου γίνεται μέσω ενός Hash Table, χρησιμοποιώντας τα αναγνωριστικά των αντικειμένων.

Πεδίο	Σημασία
<b>m_nDrawStyle</b>	Αναγνωριστικό του επιθυμητού Στιλ Παρουσίασης
<b>m_szCaption</b>	Μια συμβολοσειρά που χρησιμοποιείται ως "λεζάντα" στο αντικείμενο

Πίνακας 2.6 - Τα πεδία της κλάσης CLayerObjProperties

Η κλάση που υλοποιεί το Δεσμευμένο Επίπεδο είναι η **CBoundLayer** και είναι υποκλάση της CLayer (Σχήμα 2.9). Ο Πίνακας 2.7 παρουσιάζει τα επιπρόσθετα πεδία που ορίζονται στη CBoundLayer.



Σχήμα 2.9 - Η ιεραρχία κλάσεων Επιπέδων

Πεδίο	Σημασία
<b>m_wAttr</b>	Μια ψηφιολέξη που καθορίζει τα χαρακτηριστικά του Επιπέδου, όπως αν θα εμφανίζεται στον Χάρτη ( <i>LA_HIDDEN</i> ) ή αν επιτρέπεται να μεταβληθεί ( <i>LA_LOCKED</i> )
<b>m_nDrawStyle</b>	Αναγνωριστικό του τυπικού (default) Στιλ Παρουσίασης. Χρησιμοποιείται κατά την κατασκευή νέων αντικειμένων
<b>m_mapObProperties</b>	Το Hash Table, που αντιστοιχίζει αναγνωριστικά αντικειμένων (OIDs) σε αντικείμενα της κλάσης CLayerObjProperties

Πίνακας 2.7 - Τα πεδία της κλάσης CBoundLayer

## 2.6 Δομές Υποστήριξης

Όπως προαναφέραμε, το Επίπεδο εκτός από τα Γεωγραφικά Αντικείμενα περιέχει επιπλέον και ορισμένες δομές, που χρησιμοποιούνται για να πετύχουμε όσο δυνατόν ταχύτερη διαχείριση, αποθήκευση και ανάκτηση των αντικειμένων του Επιπέδου. Σε αυτή την ενότητα θα αναφερθούμε σε αυτές τις **Δομές Υποστήριξης**.

## 2.6.1 Δεξαμενή Αντικειμένων

Η Δεξαμενή Αντικειμένων αποτελεί τη βασική δομή διαχείρισης των αντικειμένων του Map Studio. Αποτελείται από τρία Στατικά Hash Tables, ένα για κάθε τύπο Γεωγραφικών Αντικειμένων (Θέσεις, Διαδρομές και Περιοχές). Για την ταξινόμηση των αντικειμένων χρησιμοποιούνται τα αναγνωριστικά τους (Object ID). Ο παραπάνω διαχωρισμός των Αντικειμένων ανά τύπο, σκοπεύει στην επιτάχυνση της ανάκτησης αντικειμένων με βάση τον τύπο τους.

Τα Hash Tables υλοποιούνται στο σύστημά μας από τη κλάση **CMapDWordToOb**. Βρίσκονται στην κύρια μνήμη (όπως εξάλλου και τα Γεωγραφικά Αντικείμενα) και χρησιμοποιούν την τεχνική *Αλυσίδας (Chaining)* για την επίλυση των συγκρούσεων (collision resolution).

Για τη Δεξαμενή Αντικειμένων έχουν υλοποιηθεί λειτουργίες Εισαγωγής, Διαγραφής, Αναζήτησης και Διάσχισης, ενώ επιπλέον είναι υπεύθυνη για την αποθήκευση και ανάκτηση των αντικειμένων προς και από τα αποθηκευτικά μέσα.

Η κλάση που υλοποιεί την Δεξαμενή Αντικειμένων στην εφαρμογή είναι η **CLayerObjPool**. Ο Πίνακας 2.8 περιγράφει τα βασικά πεδία αυτής της κλάσης.

Πεδίο	Σημασία
<i>m_tblPositions</i>	Hash Table για τις Θέσεις
<i>m_tblRoutes</i>	Hash Table για τις Διαδρομές
<i>m_tblRegions</i>	Hash Table για τις Περιοχές

Πίνακας 2.8 - Πεδία της κλάσης *CLayerObjPool*

## 2.6.2 Χωρικό Ευρετήριο (Τετραδικό Δέντρο)

Στην εφαρμογή Map Studio ο χρήστης έχει τη δυνατότητα να θέτει ερωτήσεις με βάση χωρικά στοιχεία, όπως η επιλογή των αντικειμένων που περικλείονται από ένα ορθογώνιο. Για να επιτευχθεί ταχύτερη απόκριση σε αυτές τις απαιτήσεις του συστήματος, χρησιμοποιείται το **Τετραδικό Δέντρο (Quad Tree)** που επιτυγχάνει αποδοτικότερο χωρικό ευρετηριασμό στη διδιάστατη γεωγραφική πληροφορία.

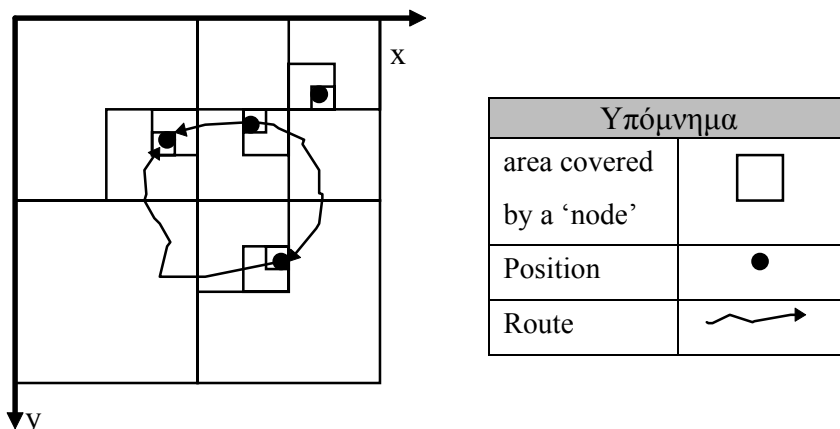
Ο όρος Τετραδικό Δέντρο περιγράφει μια τάξη ιεραρχικών δομών δεδομένων που βασίζονται στην κοινή αρχή της αναδρομικής αποσύνθεσης του χώρου [Same90].

Υπάρχουν διάφορες κατηγορίες Τετραδικών Δέντρων που κατατάσσονται σύμφωνα με:

- τον τύπο των δεδομένων για την αναπαράσταση των οποίων χρησιμοποιούνται
- την αρχή στην οποία βασίζεται η διαδικασία αποσύνθεσης
- τη σταθερότητα (ή όχι) της ανάλυσής τους (resolution)

Ο τύπος του Τετραδικού Δέντρου που χρησιμοποιείται στην εφαρμογή μας είναι μια παραλλαγή του Τετραδικού Δέντρου Περιοχών (Regional Quad Tree), που ονομάζεται MX-CIF Tree [Kede82] και πρόκειται για μια επέκταση του δυαδικού δέντρου αναζήτησης στο διδιάστατο χώρο.

Σύμφωνα με τον κανόνα που ακολουθείται, ο χώρος διαιρείται σε τέσσερα ισομεγέθη τεταρτημόρια, καθένα από τα οποία είναι με τη σειρά του ένα νέο Τετραδικό Δέντρο. Έτσι κάθε κόμβος του δέντρου μπορεί να έχει μέχρι και τέσσερα παιδιά-κόμβους (για να πετύχουμε καλύτερη διαχείριση μνήμης, κατασκευάζονται μόνο οι κόμβοι που περιέχουν αντικείμενα). Ένα αντικείμενο, τοποθετημένο στο χώρο που αναλύουμε, θεωρούμε ότι βρίσκεται σε έναν κόμβο, αν το συγκεκριμένο αντικείμενο περικλείεται εξ' ολοκλήρου στον κόμβο αλλά ταυτόχρονα δεν περικλείεται εξ' ολοκλήρου σε κάποιο από τα παιδιά του. Κάθε κόμβος κατέχει μια λίστα που περιέχει τα αναγνωριστικά των αντικειμένων που πληρούν τον παραπάνω κανόνα για αυτόν. Στο Σχήμα 2.10 παρουσιάζεται ένα παράδειγμα Τετραδικού Δέντρου.



Σχήμα 2.10 - Τετραδικό Δέντρο μέγιστου βάθους 5 με Γεωμετρικά Αντικείμενα

Το Τετραδικό Δέντρο κατασκευάζεται στην κύρια μνήμη από τη “ρίζα” προς τα “φύλλα” κατά την εισαγωγή Γεωγραφικών Αντικειμένων σε αυτό. Σε αυτό το σημείο αποφασίζεται σε ποιόν κόμβο θα τοποθετηθεί το νέο αντικείμενο σύμφωνα με τον κανόνα που περιγράψαμε στην προηγούμενη παράγραφο.

Το Τετραδικό Δέντρο υλοποιείται στην εφαρμογή από την κλάση **CQTreeNode** (υποκλάση της CObject), τα αντικείμενα της οποίας είναι οι κόμβοι του δέντρου και την κλάση **CQuadTree** (υποκλάση της CQTreeNode) που αντιπροσωπεύει τη “ρίζα” του δέντρου. Οι Πίνακες που ακολουθούν, αναγράφουν τα πεδία των προαναφερόμενων κλάσεων.

Πεδίο	Σημασία
<b>m_iDepth</b>	Το βάθος του συγκεκριμένου κόμβου
<b>m_rcArea</b>	Το ορθογώνιο παραλληλόγραμμο που αντιπροσωπεύει την περιοχή του χώρου που καταλαμβάνει ο κόμβος αυτός
<b>m_listObjIDs</b>	Η λίστα των αναγνωριστικών των Γεωγραφικών Αντικειμένων που έχουν τοποθετηθεί στον συγκεκριμένο κόμβο
<b>m_pChildren</b>	Πίνακας τεσσάρων θέσεων με δείκτες στους κόμβους – παιδιά

Πίνακας 2.9 - Πεδία της κλάσης CQTreeNode

<b>Πεδίο</b>	<b>Σημασία</b>
<b><i>m_nObjects</i></b>	Το πλήθος των αντικειμένων που βρίσκονται στο Τετραδικό Δέντρο
<b><i>m_pLayer</i></b>	Δείκτης στο επίπεδο στο οποίο ανήκει το Τετραδικό Δέντρο. Χρειάζεται για την πρόσβαση στα Γεωγραφικά Αντικείμενα (μια και γνωρίζουμε μόνο το αναγνωριστικό)

*Πίνακας 2.10 - Πεδία της κλάσης CQuadTree*

Η κλάση CQuadTree είναι εφοδιασμένη με λειτουργίες Εισαγωγής, Διαγραφής και Απάντησης σε χωρικές ερωτήσεις. Η διάσχιση, αντί του συμβατικού τρόπου (ρουτίνες), επιτυγχάνεται με ειδικές κλάσεις που ονομάζονται Iterators [Stro91] και υλοποιούν μη - αναδρομική, pre - order ή post - order διάσχιση του δέντρου.

## **2.7 Περίληψη**

Σε αυτό το Κεφάλαιο περιγράψαμε το περιβάλλον ανάπτυξης γεωγραφικών δεδομένων Map Studio. Αρχικά, αναφερθήκαμε στα βασικά αντικείμενα της εφαρμογής και στην ιεραρχία των κλάσεων που υλοποιούν τα γεωγραφικά αντικείμενά της. Στη συνέχεια, περιγράψαμε τις έννοιες του Επιπέδου και του Χάρτη μέσω των οποίων γίνεται η αναπαράσταση των γεωγραφικών δεδομένων. Τέλος, παρουσιάσαμε τους μηχανισμούς διαχείρισης αναγνωριστικών και χωρικού ευρετηριασμού που έχουν ενσωματωθεί στη διαδικασία του Επιπέδου, οι οποίοι χρησιμοποιούνται για τη διαχείριση και αποθήκευση των δεδομένων.

## Κεφάλαιο 3

# Περιγραφή Αλγορίθμων Επίλυσης του Προβλήματος Βέλτιστης Τοποθέτησης Ετικετών

### 3.1 Γενικά

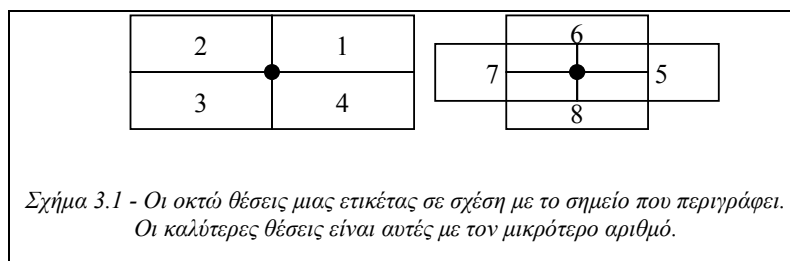
Όπως έχουμε ήδη αναφέρει στο Κεφάλαιο 1, το **Πρόβλημα Βέλτιστης Τοποθέτησης Ετικετών – ΠΒΤΕ** αφορά την εξεύρεση της καλύτερης τοποθέτησης ενός συνόλου ετικετών, ώστε να αποφευχθούν επικαλύψεις. Στο Κεφάλαιο αυτό θα περιγράψουμε ορισμένους αλγόριθμους που έχουν παρουσιαστεί και αναλυθεί σε προηγούμενες δημοσιεύσεις σχετικά με την επίλυση του ΠΒΤΕ. Η ανάλυση που ακολουθεί κρίνεται απαραίτητη για να μπορέσει ο αναγνώστης να κατανοήσει το συγκεκριμένο πρόβλημα σε βάθος, τόσο σχετικά με την πολυπλοκότητά του όσο σχετικά με τις διάφορες πτυχές του που πρέπει να αντιμετωπιστούν συνολικά.

Το ΠΒΤΕ μπορεί να θεωρηθεί ως ένα συνδυαστικό πρόβλημα που αποτελείται από δύο υποπροβλήματα: το **Χώρο Αναζήτησης (*Search Space*)** και την **Αντικειμενική Συνάρτηση (*Objective Function*)**. Σε κάθε ετικέτα αντιστοιχεί ένας συγκεκριμένος χώρος μέσα στον οποίο μπορεί να τοποθετηθεί. Αυτός ο χώρος αποτελεί το χώρο αναζήτησης κάθε ετικέτας και οριοθετεί όλες τις πιθανές θέσεις τοποθέτησης της ετικέτας. Η αντικειμενική συνάρτηση αντιστοιχίζει ένα συγκεκριμένο στοιχείο του χώρου αναζήτησης που θέλουμε να βελτιστοποιήσουμε με μία τιμή, η οποία χαρακτηρίζει την ποιότητα της εκάστοτε τοποθέτησης. Ορισμένοι παράγοντες που

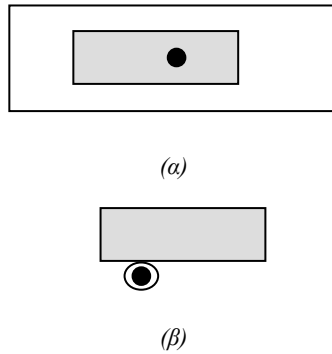


επηρεάζουν την ποιότητα των τοποθετήσεων και μπορούμε να συμπεριλάβουμε στην αντικειμενική συνάρτηση είναι οι εξής:

- Το συνολικό εμβαδόν του χώρου επικάλυψης των ετικετών με άλλα γραφικά αντικείμενα (συμπεριλαμβανομένων και των υπόλοιπων ετικετών).
- Ο αριθμός των αντικειμένων που δεν έχουν κάποια ετικέτα.
- Η, από πρότινος, αξιολόγηση των πιθανών θέσεων τοποθέτησης των ετικετών έτσι, ώστε ορισμένες από αυτές να προτιμούνται από τις υπόλοιπες. Στο Σχήμα 3.1 παρουσιάζεται μια τέτοια αξιολόγηση για ετικέτες που ανήκουν σε σημειακά αντικείμενα.



Το παραπάνω Σχήμα, επιπλέον, αποτελεί ένα παράδειγμα διακριτού μοντέλου τοποθέτησης ετικετών, σύμφωνα με το οποίο η κάθε ετικέτα έχει περιορισμένο αριθμό θέσεων τοποθέτησης γύρω από το σημείο αναφοράς της. Αντίστοιχα, το συνεχές μοντέλο τοποθέτησης ετικετών επιτρέπει στην ετικέτα να τοποθετηθεί σε πολύ περισσότερες θέσεις γύρω από την ετικέτα. Στο Σχήμα 3.2 παρουσιάζονται δύο μοντέλα συνεχούς τοποθέτησης ετικετών. Στο Σχήμα 3.2(α) παρουσιάζεται το μοντέλο, το οποίο και τελικά υλοποιήσαμε, κατά το οποίο η ετικέτα μπορεί να τοποθετηθεί οπουδήποτε μέσα στο μεγάλο ορθογώνιο με κέντρο το σημείο στο οποίο ανήκει η ετικέτα. Κατά το μοντέλο του Σχήματος 3.2(β) η ετικέτα μπορεί να τοποθετηθεί οπουδήποτε γύρω από το σημείο στο οποίο ανήκει, αρκεί να εφάπτεται πάντα με τον κύκλο που έχει ως κέντρο το συγκεκριμένο σημείο.



Σχήμα 3.2 - Παραδείγματα συνεχών μοντέλων τοποθέτησης ετικετών

Είναι προφανές ότι τα συνεχή μοντέλα δίνοντάς μας τη δυνατότητα να έχουμε πολύ περισσότερες διαφορετικές θέσεις για κάθε σημειακή ετικέτα, αυξάνουν τις πιθανότητες τοποθέτησης τελικά μεγαλύτερου αριθμού ετικετών χωρίς επικαλύψεις.

### 3.2 Πολυπλοκότητα του ΠΒΤΕ

Στην Εισαγωγή αναφέραμε ότι η πολυπλοκότητα του ΠΒΤΕ είναι NP-Hard. Η απόδειξη επιτυγχάνεται με τον εξής συλλογισμό: για κάθε χάρτη στον οποίο παρουσιάζονται επικαλύψεις ετικετών μπορούμε να θέσουμε το εξής ερώτημα: υπάρχει κάποια τοποθέτηση των ετικετών που να παρουσιάζει μηδενικό κόστος, δηλαδή μία τοποθέτηση στην οποία δεν υπάρχουν επικαλύψεις ετικετών είτε μεταξύ τους είτε με άλλα γραφικά αντικείμενα;

Το παραπάνω ερώτημα καθορίζει ένα πρόβλημα που είναι γνωστό με την ονομασία **Πρόβλημα Αποδεκτής Τοποθέτησης Ετικετών – ΠΑΤΕ (Admissible-Labeling Problem)**. Η πολυπλοκότητα του προαναφερόμενου προβλήματος έχει αποδειχτεί από τους Kato και Imai [1988], Marks και Shieber [1991], Formann και Wagner [1991] ότι είναι NP-Hard. Είναι φανερό ότι οποιοσδήποτε αλγόριθμος επίλυσης του ΠΒΤΕ μπορεί να χρησιμοποιηθεί για την εύρεση λύσης σε ένα ΠΑΤΕ, δηλαδή να βρεθεί μία τοποθέτηση ετικετών χωρίς επικαλύψεις. Συνεπώς το ΠΒΤΕ είναι τουλάχιστον το ίδιο δύσκολο να επιλυθεί όσο και το ΠΑΤΕ. Οπότε, καταλήξαμε στο συμπέρασμα ότι το ΠΒΤΕ είναι πρόβλημα πολυπλοκότητας NP-Hard.

### **3.3 Αλγόριθμοι Επίλυσης του ΠΒΤΕ για Ετικέτες Σημειακών Αντικειμένων**

Σε αυτή την ενότητα θα περιγράψουμε ορισμένους αλγόριθμους επίλυσης του ΠΒΤΕ για Ετικέτες Σημειακών Αντικειμένων που παρουσιάστηκαν στο παρελθόν και θα αναφερθούμε στα μειονεκτήματα και τα πλεονεκτήματα που παρουσιάζουν. Οι αλγόριθμοι αυτοί συνοψίζονται στο [Chms95]. Το σύνολο των αλγόριθμων αυτών ανήκουν στην κατηγορία των **Αλγορίθμων Πλήρους Έρευνας (Complete Search Algorithms)**, οι οποίοι στην προσπάθειά τους να επιλύσουν το ΠΒΤΕ, αναζητούν τη βέλτιστη λύση του εκάστοτε προβλήματος, σπαταλώντας με αυτό τον τρόπο υπερβολική υπολογιστική ισχύ χωρίς να προσφέρουν ικανοποιητικές λύσεις σε αποδεκτά χρονικά περιθώρια για λογικό πλήθος ετικετών. Οι Αλγόριθμοι Πλήρους Έρευνας μπορούν να ταξινομηθούν σε δύο κατηγορίες: τους **Αλγόριθμους Εξαντλητικής Έρευνας (Exhaustive Search Algorithms)** που πραγματοποιούν εξονυχιστική έρευνα στο σύνολο των ετικετών και τους **Αλγόριθμους Τοπικής Έρευνας (Local Search Algorithms)** οι οποίοι ομαδοποιούν τις ετικέτες σύμφωνα με ορισμένα κριτήρια και εκτελούν έρευνα σε κάθε ομάδα ετικετών χωριστά.

#### **3.3.1 Αλγόριθμοι Εξαντλητικής Έρευνας (Exhaustive Search Algorithms)**

Οι Αλγόριθμοι Εξαντλητικής Έρευνας ακολουθούν της εξής τακτική στην προσπάθειά τους να επιλύσουν το ΠΒΤΕ: ο συνολικός χώρος τοποθέτησης κάθε ετικέτας υποδιαιρείται σε επιμέρους χώρους τοποθέτησης. Στη συνέχεια, ελέγχονται σειριακά οι χώροι αυτοί και η εκάστοτε ετικέτα τοποθετείται σε μία θέση στην οποία δεν παρουσιάζεται καμία σύγκρουση. Αν κάποια ετικέτα δεν έχει ελεύθερη θέση τοποθέτησης, τότε επιστρέφουμε (backtracking) στην προηγούμενη ετικέτα και την τοποθετούμε σε μία διαφορετική, ελεύθερη θέση με σκοπό να ελευθερωθεί κάποια θέση τοποθέτησης για την τελευταία ετικέτα. Ο αλγόριθμος συνεχίζει την εκτέλεσή του μέχρι να επιτευχθεί ένα αποδεκτό αποτέλεσμα συνολικής τοποθέτησης ή μέχρι να εξαντληθεί ο συνολικός χώρος αναζήτησης.

Ο παραπάνω αλγόριθμος έχει χρησιμοποιηθεί ως βάση για πολλούς αλγόριθμους τοποθέτησης ετικετών που έχουν παρουσιαστεί. Παρόλο που οι συγκεκριμένοι αλγόριθμοι δίνουν αποδεκτά αποτελέσματα σε σχετικά μικρό χρόνο εκτέλεσης για μικρό αριθμό ετικετών, το γεγονός ότι ο χώρος αναζήτησης αυξάνεται εκθετικά, τους καθιστά ασύμφορους για να χρησιμοποιηθούν ως γενική λύση για το ΠΒΤΕ.

### **3.3.2 Άπληστοι Αλγόριθμοι (Greedy Algorithms)**

Οι Άπληστοι Αλγόριθμοι προσφέρουν μια πιο πρακτική προσέγγιση στην επίλυση του ΠΒΤΕ αποφεύγοντας την επιστροφή (backtracking) σε προηγούμενες ετικέτες, αν δεν βρεθεί αποδεκτή θέση τοποθέτησης για κάποια από αυτές. Έτσι, με τον παραπάνω περιορισμό της έρευνας που εκτελείται, οι Άπληστοι Αλγόριθμοι προσφέρουν αρκετά καλαίσθητα αποτελέσματα με μικρότερο υπολογιστικό κόστος σε σύγκριση με τους Αλγόριθμους Εξαντλητικής Έρευνας. Οι Άπληστοι Αλγόριθμοι στην περίπτωση που μία ετικέτα δεν μπορεί να τοποθετηθεί σε κάποια θέση, είτε δεν τοποθετείται [Langran and Poiker 1986] είτε τοποθετείται ακόμα και αν επικαλύπτεται με κάποιο άλλο αντικείμενο.

Οι Άπληστοι Αλγόριθμοι δίνουν αποδεκτά αποτελέσματα για ρεαλιστικούς αριθμούς ετικετών σε αρκετά μικρότερους χρόνους εκτέλεσης από ό,τι οι Αλγόριθμοι Εξαντλητικής Έρευνας, όμως η έλλειψη επανατοποθέτησης των ετικετών σε περίπτωση επικαλύψεων επηρεάζει αρνητικά την ποιότητα των λύσεων που βρίσκονται.

### **3.3.3 Αλγόριθμοι Καθόδου Σταδιακής Κλίσης (Discrete Gradient Descent Algorithms)**

Η ποιότητα των λύσεων που προσφέρουν οι Άπληστοι Αλγόριθμοι μπορεί να βελτιωθεί δραματικά, αν οι ετικέτες επανατοποθετηθούν σε τοπικό επίπεδο. Οι Αλγόριθμοι Καθόδου Σταδιακής Κλίσης (ΑΚΣΚ σε συντομογραφία) υλοποιήθηκαν για να προσφέρουν το παραπάνω αποτέλεσμα. Μία μέθοδος Καθόδου Κλίσης καθορίζει μία ομάδα λειτουργιών που επιτρέπουν σε μία ή περισσότερες ετικέτες να μπορούν να επανατοποθετηθούν ταυτόχρονα σε περίπτωση επικαλύψεων. Η βάση των μεθόδων αυτών είναι η επιλογή από το σύνολο των λειτουργιών, της

κατάλληλης, η οποία αποδίδει την πιο άμεση βελτίωση. Με τη συνεχή χρησιμοποίηση της συγκεκριμένης λειτουργίας καταλήγουμε σε μία συνολική τοποθέτηση των ετικετών σημαντικά ανώτερη από την αρχική.

Στη συνέχεια θα αναφέρουμε ένα παράδειγμα Αλγόριθμου ΚΣΚ. Έστω ότι η μέθοδος περιλαμβάνει λειτουργίες, οι οποίες μετακινούν μία ετικέτα τυχαία από μία θέση τοποθέτησης σε μία άλλη. Ο αλγόριθμος είναι ο εξής:

Τοποθέτησε όλες τις ετικέτες σε κάποια θέση τοποθέτησης τυχαία

Επανάλαβε έως ότου καμία βελτίωση να μην είναι δυνατή

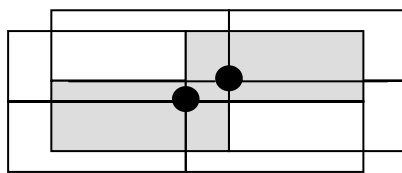
Για κάθε ετικέτα θεώρησε ότι επανατοποθετείται σε όλες τις πιθανές θέσης

Για κάθε πιθανή επανατοποθέτηση υπολόγισε την αλλαγή στην Αντικειμενική Συνάρτηση, η οποία θα σημειωνόταν αν η ετικέτα μετακινιόταν στη συγκεκριμένη θέση

Τοποθέτησε κάθε ετικέτα στη θέση στην οποία παρατηρείται η μεγαλύτερη βελτίωση

Πρακτικά ο ΑΚΣΚ υπολογίζει αρχικά ένα πίνακα από κόστη που σχετίζονται με κάθε επανατοποθέτηση ετικέτας. Μετά από κάθε τοποθέτηση ετικέτας επαναυπολογίζονται μόνο τα κόστη των θέσεων τοποθέτησής της που συγκρούονται με τις ήδη τοποθετημένες ετικέτες.

Το κυριότερο μειονέκτημα των ΑΚΣΚ είναι η ανικανότητά τους να ξεπεράσουν το τοπικό ελάχιστο (*local minima*) της αντικειμενικής συνάρτησης. Στο Σχήμα 3.3 δίνεται ένα παράδειγμα τοπικού ελάχιστου.

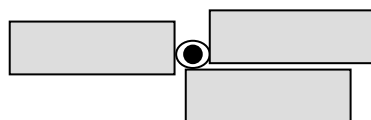


Σχήμα 3.3 – Παράδειγμα τοπικού ελάχιστου για τους Αλγόριθμους ΚΣΚ

Στο παραπάνω παράδειγμα η σύγκρουση μπορεί να επιλυθεί με την τοποθέτηση της ετικέτας του κάτω σημείου στην κάτω αριστερά θέση τοποθέτησης και της πάνω ετικέτας στην πάνω αριστερά θέση αντίστοιχα. Όμως, αν μετακινήσουμε μόνο μία από τις δύο ετικέτες, η τιμή της αντικειμενικής συνάρτησης δεν βελτιώνεται και επειδή ο ΑΚΣΚ δέχεται μόνο αλλαγές που επιφέρουν άμεση βελτίωση στην τιμή της αντικειμενικής συνάρτησης, ο αλγόριθμος δεν μπορεί να μας οδηγήσει στη λύση που προαναφέραμε. Αν δώσουμε την δυνατότητα στον αλγόριθμο να μετακινεί τις ετικέτες χωρίς να ελέγχει την τιμή της αντικειμενικής συνάρτησης, είναι πιθανόν να οδηγήσει σε λύση το παράδειγμα του παραπάνω σχήματος, αλλά γενικά δεν επιλύει το πρόβλημα. Για να αποφύγουμε τις επιπτώσεις του τοπικού ελάχιστου είναι απαραίτητο να χρησιμοποιηθούν πιο περίπλοκοι Αλγόριθμοι ΚΣΚ.

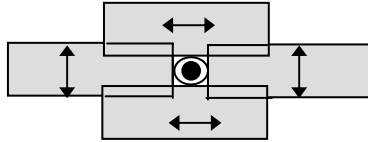
### 3.3.4 Προσέγγιση της Καθόδου (Gradient) με τη χρήση Διανυσμάτων Επικάλυψης (Overlap Vectors)

Ο Hirsch [1982] παρουσίασε μία τροποποιημένη μέθοδο Καθόδου Κλίσης για την επίλυση του ΠΒΤΕ. Στον αλγόριθμο αυτό η κάθε ετικέτα τοποθετείται σύμφωνα με το συνεχές μοντέλο τοποθέτησης, εφαπτόμενη πάντα σε έναν κύκλο με κέντρο το σημείο στο οποίο ανήκει, όπως φαίνεται στο Σχήμα 3.4:



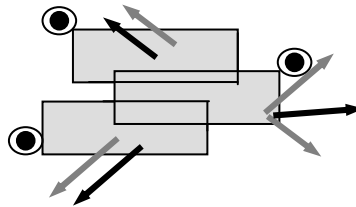
Σχήμα 3.4 – Πιθανές θέσεις τοποθέτησης ετικέτας γύρω από το σημείο στο οποίο ανήκει κατά τον Hirsch [1982]

Το υψηλότερο, χαμηλότερο, πιο αριστερό η πιο δεξιό σημείο του κύκλου θεωρούνται ειδικές θέσεις τοποθέτησης, στα οποία η ετικέτα επιτρέπεται να κυλάει, όπως παρουσιάζεται στο Σχήμα 3.5 κατά μήκος του σημείου επαφής.



Σχήμα 3.5 – Ειδικές θέσεις τοποθέτησης ετικετών

Αρχικά όλες οι ετικέτες τοποθετούνται στη δεξιά ειδική θέση. Στη συνέχεια, κάθε ετικέτα εξετάζεται για πιθανές συγκρούσεις με άλλες ετικέτες και επικαλύψεις με άλλα σημεία. Για κάθε επικάλυψη υπολογίζεται ένα *διάνυσμα επικάλυψης* με βάση τις διαστάσεις της περιοχής επικάλυψης. Έπειτα, υπολογίζεται το άθροισμα των διανυσμάτων επικάλυψης για κάθε ετικέτα. Ο προηγούμενος υπολογισμός κατασκευάζει ένα νέο διάνυσμα, το οποίο ονομάζεται *αθροιστικό διάνυσμα* και μας αποκαλύπτει τη διεύθυνση προς την οποία θα πρέπει να μετακινηθεί η ετικέτα έτσι ώστε να επιλυθούν οι συγκρούσεις. Στο Σχήμα 3.6 που ακολουθεί περιγράφεται ένα παράδειγμα σύγκρουσης ετικετών, όπου τα διανύσματα επικάλυψης παρουσιάζονται με γκριζό χρώμα, ενώ τα αθροιστικά διανύσματα με μαύρο. Παρατηρούμε ότι για τις ετικέτες που σχετίζονται μόνο με μία σύγκρουση τα δύο διανύσματα συμπίπτουν.



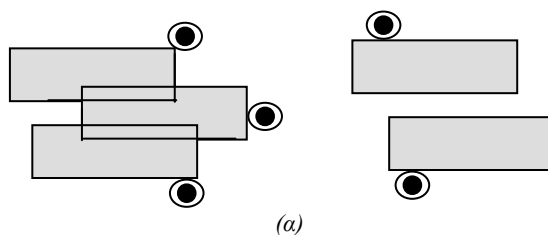
Σχήμα 3.6 – Διανύσματα επικάλυψης

Αφού τα αθροιστικά διανύσματα έχουν υπολογιστεί, ο αλγόριθμος μετακινεί κάθε ετικέτα προς την κατεύθυνση του διανύσματος που της αντιστοιχεί σε μία προσπάθεια να μειώσει τον αριθμό των επικαλύψεων. Η μετακίνηση αυτή μπορεί να γίνει με δύο τρόπους, είτε *σταδιακά* (*incremental movement*) γύρω από τον κύκλο τοποθέτησης, είτε *απόλυτα* (*absolute movement*) στη σημείο τοποθέτησης του κύκλου το οποίο υποδεικνύεται από το αθροιστικό διάνυσμα. Η απόλυτη μετακίνηση τοποθετεί την ετικέτα στη θέση που υποδεικνύεται από το αθροιστικό διάνυσμα, ενώ η σταδιακή μετακίνηση χρησιμοποιεί ορισμένους ευριστικούς κανόνες που μετακινούν την ετικέτα προς τη διεύθυνση του αθροιστικού διανύσματος. Μπορούμε να χρησιμοποιήσουμε μία από τις δύο προαναφερόμενες λειτουργίες κάθε γύρο εκτέλεσης του αλγορίθμου, επιτυγχάνοντας με αυτόν τον τρόπο τη μετακίνηση

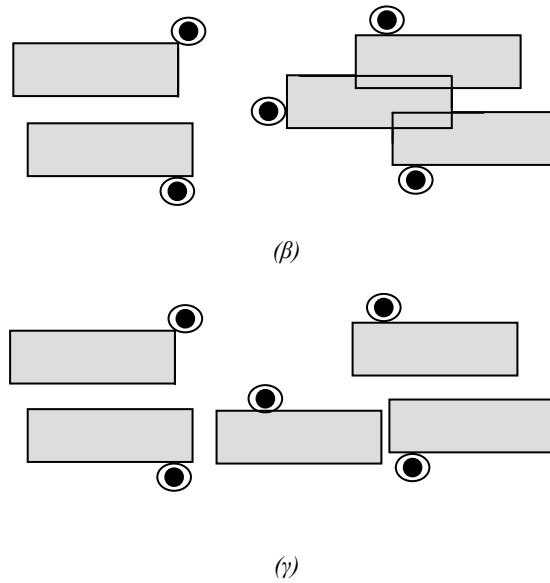
πολλών ετικετών ταυτόχρονα. Ο Hirsch προτείνει την εναλλακτική χρήση των δύο λειτουργιών με έμφαση, όμως, στη σταδιακή μετακίνηση.

Βασικό πρόβλημα του συγκεκριμένου αλγόριθμου είναι οι πιθανές λανθασμένες ενδείξεις επανατοποθέτησης των ετικετών, καθώς τα διανύσματα επικάλυψης προσφέρουν μόνο μία προσέγγιση της μετακίνησης κάθε ετικέτας που πρέπει να εκτελεστεί. Εάν, για παράδειγμα, μία ετικέτα επικαλύπτεται από μεγάλο αριθμό ετικετών, το μέγεθος του υπολογισμένου αθροιστικού διανύσματος μπορεί να είναι υπερβολικά μεγάλο με τελικό αποτέλεσμα να απαιτούνται άσκοποι, μεγάλοι υπολογισμοί κατά την σταδιακή μετακίνηση. Χρειάζεται να τονίσουμε ότι τα παραπάνω διανύσματα έχουν δύο βαθμούς ελευθερίας, ενώ η μετακίνηση των ετικετών περιορίζεται γύρω από τον κύκλο τοποθέτησης. Αυτό έχει ως αποτέλεσμα ότι ακόμα και σε περιπτώσεις στις οποίες το αθροιστικό διάνυσμα έχει τη σωστή διεύθυνση μετακίνησης για την επίλυση της σύγκρουσης, αν το μέτρο του είναι υπερβολικά μεγάλο, η νέα θέση της ετικέτας είναι κάπως αυθαίρετη.

Επιπλέον, όπως και ο Αλγόριθμος ΚΣΚ έτσι και ο Αλγόριθμος του Hirsch δεν μπορεί να αντιμετωπίσει με απόλυτη επιτυχία το τοπικό ελάχιστο. Το πρόβλημα του τοπικού ελάχιστου για τον Αλγόριθμο του Hirsch οφείλεται στις μεθόδους μετακίνησης των ετικετών που έχουμε ήδη αναφέρει. Στο Σχήμα 3.7 δίνεται ένα παράδειγμα στο οποίο το πρόβλημα του Τοπικού Ελάχιστου.







Σχήμα 3.7 – Το πρόβλημα του τοπικού ελάχιστου για τον Αλγόριθμο του Hirsch

Ο αλγόριθμος στο παραπάνω πρόβλημα ταλαντεύεται μεταξύ των καταστάσεων που φαίνονται στο Σχήμα 3.7(α) και στο Σχήμα 3.7(β), αδυνατώντας να καταλήξει στη σωστή λύση (Σχήμα 3.7(γ)). Αν χρησιμοποιηθεί η σταδιακή μετακίνηση η ετικέτα του κεντρικού σημείου μετακινείται ελάχιστα προς τα πάνω και προς τα κάτω, χωρίς όμως να αποφεύγεται η σύγκρουσή της με τις δύο άλλες ετικέτες. Αν, αντίθετα, χρησιμοποιηθεί η απόλυτη μετακίνηση, η οριζόντια συνιστώσα του αθροιστικού διανύσματος υπερισχύει της κάθετης με αποτέλεσμα η ετικέτα να μετακινείται μεταξύ της αριστερής και της δεξιάς θέσης τοποθέτησης, χωρίς να μπορεί να βρει τη σωστή θέση.

### 3.3.5 Μαθηματικός Προγραμματισμός Επίλυσης του ΠΒΤΕ

Στη συνέχεια θα αναφερθούμε σε έναν αλγόριθμο που παρουσιάστηκε από τον Zoraster [1986, 1990], στον οποίο χρησιμοποιείται μαθηματικές προγραμματιστικές τεχνικές για την επίλυση του ΠΒΤΕ. Κατά τον Αλγόριθμο του Zoraster το ΠΒΤΕ αρχικά εκφράζεται ως ένα **0-1 Ακέραιο Προγραμματιστικό Πρόβλημα -ΜΕΑΠΠ (0-1 Integer Programming Problem –ZOLP)**.

- Έστω ότι ο χάρτης που εξετάζουμε έχει  $K$  ετικέτες και  $N_k$  πιθανές θέσεις τοποθέτησης για κάθε ετικέτα. Κάθε θέση τοποθέτησης αναπαριστάται από μία μεταβλητή  $X_{i,k}$ ,  $1 \leq i \leq N_k$ , και  $1 \leq k \leq K$ .

- Οι μεταβλητές  $X_{i,k}$  έχουν τιμή 0 ή 1, σημειώνοντας με αυτόν τον τρόπο την απουσία ή την παρουσία, αντίστοιχα, μίας ετικέτας στη συγκεκριμένη θέση.
- Κάθε σημείο μπορεί να έχει μόνο μία ετικέτα, δηλαδή:  $\sum_{i=1}^{N_k} X_{i,k} = 1$  για  $1 \leq k \leq K$  (περιορισμοί ετικετών).
- Αν έχουμε  $M$  επικαλύψεις ανά ζεύγη θέσεων τοποθέτησης, τότε για να αποφύγουμε μία πιθανή σύγκρουση ετικετών θα πρέπει να ισχύει:  $X_{r_m, s_m} + X_{r'_m, s'_m} \leq 1$  για κάθε πιθανή επικάλυψη,  $1 \leq m \leq M$  (περιορισμοί επικαλύψεων).
- Η αντικειμενική συνάρτηση είναι η εξής:  $\sum_{k=1}^K \sum_{i=1}^{N_k} W_{i,k} X_{i,k}$ , όπου  $W_{i,k}$  είναι οι συντελεστές αντικειμενικής συνάρτησης που υποδεικνύουν κάποιες πιθανές προτιμήσεις για ορισμένες θέσεις τοποθέτησης.

Αρχικά ο Αλγόριθμος του Zoraster συμπεριλαμβάνει τον περιορισμό επικαλύψεων στην Αντικειμενική Συνάρτηση, η οποία μετασχηματίζεται ως εξής:

- Ελαχιστοποίησε τη συνάρτηση  $\sum_{k=1}^K \sum_{i=1}^{N_k} W_{i,k} X_{i,k} + \sum_{m=1}^M (X_{r_m, s_m} + X_{r'_m, s'_m} - 1) d_m$
- Ακόμα ισχύει ο περιορισμός ετικετών.

Σε αυτή την τροποποιημένη αντικειμενική συνάρτηση οι όροι  $d_m \geq 0$  είναι πολλαπλασιαστές Lagrange<sup>5</sup>, ένας για κάθε περιορισμό επικαλύψεων. Πρέπει να σημειωθεί ότι για ένα δεδομένο σύνολο πολλαπλασιαστών Lagrange, η ελάχιστη τιμή της αντικειμενικής συνάρτησης μπορεί εύκολα να υπολογιστεί, αν επιλέξουμε τη μεταβλητή  $X_{i,k}$  με το μικρότερο συντελεστή  $W_{i,k}$  για κάθε σημείο.

Στη συνέχεια θα παραθέσουμε το βασικό Αλγόριθμο του Zoraster:

---

<sup>5</sup> Η μέθοδος των πολλαπλασιαστών Lagrange χρησιμοποιείται για την εύρεση ακροτάτων συναρτήσεων υπο συνθήκες. Στη συγκεκριμένη περίπτωση ζητάμε να ελαχιστοποιήσουμε τον πρώτο όρο της αντικειμενικής συνάρτησης με την προϋπόθεση ότι  $X_{r_m, s_m} + X_{r'_m, s'_m} - 1 = 0$ .

- (1) Υπολόγισε και αποθήκευσε τους συντελεστές της αντικειμενικής συνάρτησης  $W_{i,k}$  για κάθε θέση τοποθέτησης
- (2) Δημιούργησε μία Συνολική Τοποθέτηση Ετικετών (ΣΤΕ) επιλέγοντας για κάθε ετικέτα τη θέση τοποθέτησης με τον μικρότερο συντελεστή  $W_{i,k}$
- (3) Αρχικοποίησε το Ενεργό Σύνολο Περιορισμών (ΕΣΠ) ως μηδενικό σύνολο
- (4) Επανάλαβε για 40 φορές ή μέχρι να βρεθεί μία λύση χωρίς επικαλύψεις ετικετών:
  - (α) Εξακρίβωσε όλους τους περιορισμούς επικαλύψεων τους οποίους η ΣΤΕ παραβαίνει και πρόσθεσε τους νέους Περιορισμούς στο ΕΣΠ. (Ο πολλαπλασιαστής Lagrange για κάθε νέο Περιορισμό είναι αρχικά ίσος με μηδέν, οπότε η εισαγωγή του νέου περιορισμού στο ΕΣΠ δεν επηρεάζει τους συντελεστές της αντικειμενικής συνάρτησης).
  - (β) Δημιούργησε ένα τοπικό αντίγραφο ΣΤΕ' του ΣΤΕ
  - (γ) Επανάλαβε για  $x$  φορές, όπου  $x$  είναι ο αριθμός των επαναλήψεων που απαιτούνται για να βρεθεί μία πιθανή λύση σύμφωνα με το υπάρχον ΕΣΠ, συν 100 επιπλέον επαναλήψεις αν η πιθανή λύση βρέθηκε στις πρώτες 400 επαναλήψεις:
    - (i) Ενημέρωσε το ΣΤΕ' επιλέγοντας για κάθε ετικέτα τη θέση τοποθέτησης με το μικρότερο συντελεστή αντικειμενικής συνάρτησης
    - (ii) Αντέγραψε το ΣΤΕ' στο ΣΤΕ αν είναι καλύτερο

(iii) Αν κάποιος περιορισμός του ΕΣΠ είναι υπερβολικά περιοριστικός, δηλαδή και οι δύο θέσεις τοποθέτησης που ελέγχει είναι κατειλημμένες, ο αντίστοιχος πολλαπλασιαστής Lagrange αυξάνεται, οπότε αυξάνονται και οι συντελεστές αντικειμενικής συνάρτησης για τις δύο θέσεις τοποθέτησης

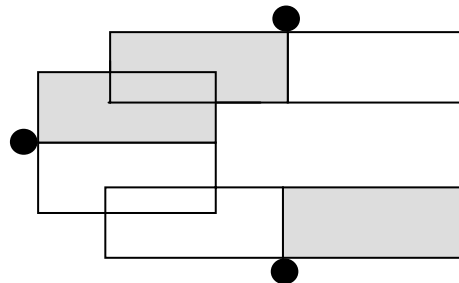
(iv) Αν κάποιος περιορισμός στο ΕΣΠ είναι ελάχιστα περιοριστικός, δηλαδή και οι δύο θέσεις τοποθέτησης που ελέγχει δεν είναι κατειλημμένες, ο αντίστοιχος πολλαπλασιαστής Lagrange μειώνεται, οπότε μειώνονται και οι συντελεστές αντικειμενικής συνάρτησης για τις δύο θέσεις τοποθέτησης

(5) Επέστρεψε το ΣΤΕ

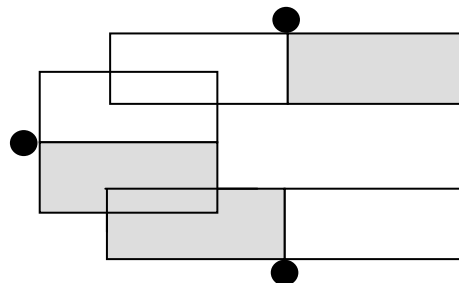
Αν ο αλγόριθμος εφαρμοζόταν ακριβώς, όπως περιγράφηκε στην προηγούμενη παράγραφο, τότε τα αποτελέσματα που θα είχαμε δεν θα ήταν ικανοποιητικά. Ο αλγόριθμος παρουσιάζει δύο κύρια μειονεκτήματα: είναι αρκετά ευαίσθητος στο τοπικό ελάχιστο και έχει την τάση να πέφτει σε μία άσκοπη, παλινδρομική συμπεριφορά. Για να αποφευχθούν τα δύο μειονεκτήματα που προαναφέραμε, είναι απαραίτητο να υλοποιηθούν ορισμένες αλλαγές στο βασικό αλγόριθμο. Μία τροποποίηση είναι η εξής: αν ένας καθορισμένος αριθμός επαναλήψεων εκτελεστεί, χωρίς να παρατηρηθεί βελτίωση στα αποτελέσματα που εξετάζονται, τότε θεωρούμε ότι ο βρισκόμαστε σε μία περιοχή τοπικού ελάχιστου της αντικειμενικής συνάρτησης. Με τη μείωση των αλλαγών των τιμών των συντελεστών αντικειμενικής συνάρτησης που γίνονται στα βήματα 4(γ)(iii) και 4(γ)(iv) ο αλγόριθμος είναι ικανός να αναγνωρίσει περιπτώσεις τοπικού ελάχιστου.

Ακόμα και αν μεταβάλλουμε τον αλγόριθμο σύμφωνα με τις παραπάνω τροποποιήσεις, ο αλγόριθμος ακόμα τείνει να παλινδρομεί μεταξύ περιπτώσεων

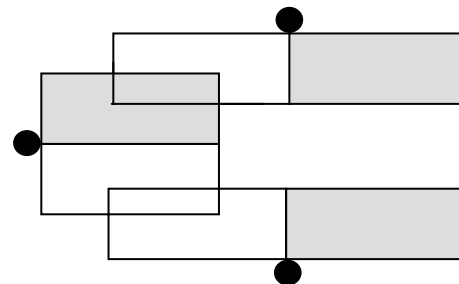
τοπικού ελάχιστου, στις οποίες συνεχώς επαναυπολογίζει μία συγκεκριμένη σειρά τοποθετήσεων. Στο Σχήμα 3.8 παρουσιάζουμε ένα παράδειγμα τοπικού ελάχιστου, το οποίο ο Αλγόριθμος του Zoraster αδυνατεί να ξεπεράσει.



(α)



(β)



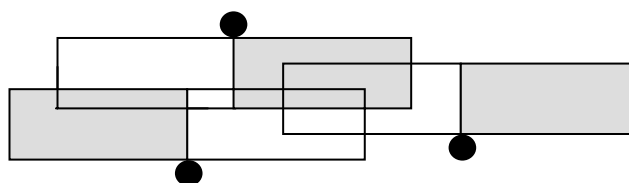
(γ)

Σχήμα 3.8 – Περίπτωση τοπικού ελάχιστου για τον Αλγόριθμο του Zoraster

Όπως παρατηρούμε, τα σημεία ανά δύο έχουν θέσεις τοποθέτησης ετικετών που συγκρούονται. Στις περιπτώσεις των Σχημάτων 3.8(α) και 3.8(β) ένα τέτοιο ζεύγος είναι κατειλημμένο από ετικέτες, οπότε οι συντελεστές Αντικειμενικής Συνάρτησης που τους αντιστοιχούν θα αρχίσουν να αυξάνονται σταδιακά. Αυτή η αύξηση με το χρόνο θα κάνει τις συγκεκριμένες θέσεις λιγότερο ελκυστικές, οπότε είναι πολύ πιθανόν και οι δύο ετικέτες να μετακινηθούν ταυτόχρονα. Έπειτα οι δύο θέσεις θα

μειώνουν κενές, οπότε οι τιμές των συντελεστών αντικειμενικής συνάρτησής τους θα μειώνονται με το πέρασμα του χρόνου. Αυτό θα έχει ως αποτέλεσμα οι δύο θέσεις να γίνονται σταδιακά πιο ελκυστικές, οπότε αυξάνονται οι πιθανότητες οι ετικέτες να επανατοποθετηθούν στις δύο συγκεκριμένες θέσεις ταυτόχρονα. Η σταθερή κατάσταση του Σχήματος 3.8(γ) ποτέ δε θα βρεθεί. Μία λύση στο συγκεκριμένο πρόβλημα είναι να επιλέξουμε μόνο μία από τις δύο θέσεις τοποθέτησης, της οποίας θα μεταβάλουμε το συντελεστή αντικειμενικής συνάρτησης. Η επιλογή του συντελεστή γίνεται εξετάζοντας αν ο αλγόριθμος βρίσκεται σε άρτιο ή περιττό αριθμό επαναλήψεων.

Μία ακόμα πιο δύσκολη περίπτωση είναι αυτή που παρουσιάζεται στο Σχήμα 3.9.



Σχήμα 3.9 – Μία ασταθής κατάσταση για τον Αλγόριθμο του Zoraster

Οι δύο θέσεις τοποθέτησης της αριστερής και δεξιάς ετικέτας που επικαλύπτονται δημιουργούν έναν ελάχιστο περιοριστικό περιορισμό, με τελικό αποτέλεσμα οι δύο ετικέτες να τοποθετηθούν στις συγκεκριμένες θέσεις, παρόλο που πάντα θα συγκρούονται με την πάνω ετικέτα. Αν οι θέσεις τοποθέτησης που επικαλύπτονται μεταξύ τους είναι ακόμη περισσότερες, το πρόβλημα γίνεται εντονότερο, καθώς οι ετικέτες τείνουν να μεταφέρονται σε περιοχές, στις οποίες ήδη υπάρχει μεγάλος αριθμός επικαλύψεων. Μία τροποποίηση του αλγόριθμου που περιορίζει το παραπάνω πρόβλημα είναι αυτή που παρατίθεται στη συνέχεια: αν δεν έχει βρεθεί μία ικανοποιητική λύση μετά από 400 επαναλήψεις, οι συντελεστές αντικειμενικής συνάρτησης των θέσεων τοποθέτησης που επικαλύπτονται ανά τρεις ή περισσότερες μηδενίζονται και καθλώνονται σε αυτή την τιμή. Αν ύστερα από 600 επαναλήψεις ακόμη δεν έχει βρεθεί μία αποδεκτή λύση, η τελευταία, μη αποδεκτή λύση επιστρέφεται στο αρχικό επίπεδο του αλγόριθμου. Αυτή η κίνηση ισοδυναμεί με εξάλειψη θέσεων τοποθέτησης από τις πυκνές περιοχές του χάρτη.

### **3.3.6 Ανακεφαλαίωση των Αλγορίθμων επίλυσης του ΠΒΤΕ για Σημειακά Αντικείμενα**

Όπως παρατηρήσαμε, κάθε Αλγόριθμος Τοπικής Έρευνας που περιγράψαμε μπορεί να παγιδευτεί σε περιπτώσεις τοπικού ελάχιστου του χώρου αναζήτησης. Τα προβλήματα που παρατηρούνται στους συγκεκριμένους αλγόριθμους μπορούμε να τα κατηγοριοποιήσουμε σε δύο ομάδες.

Πρώτον, υπάρχουν διάφοροι συνδυασμοί τοποθετήσεων ετικετών στους οποίους οι διάφοροι αλγόριθμοι παγιδεύονται στο τοπικό ελάχιστο. Καθώς ο αριθμός και η πυκνότητα τοποθέτησης των σημείων αυξάνονται, μεγαλώνουν ταυτόχρονα και οι πιθανότητες ο κάθε αλγόριθμος να βρεθεί αντιμέτωπος με μία τέτοια κατάσταση με τελικό αποτέλεσμα η απόδοσή του να πέσει κατακόρυφα.

Δεύτερον, οι διάφορες λειτουργίες που εκτελούν οι αλγόριθμοι για την τοποθέτηση των ετικετών δεν τους επιτρέπουν να ξεφύγουν από τις περιπτώσεις αυτές από τη στιγμή που θα παγιδευτούν σε αυτές. Τα δύο παραπάνω προβλήματα έχει παρατηρηθεί ότι αντιμετωπίζονται αρκετά επιτυχώς από στοχαστικές μεθόδους, κάτι που οφείλεται στη απρόβλεπτη συμπεριφορά που επιδεικνύουν, η οποία τους επιτρέπει να αποδεσμεύονται από καταστάσεις τοπικού ελάχιστου.

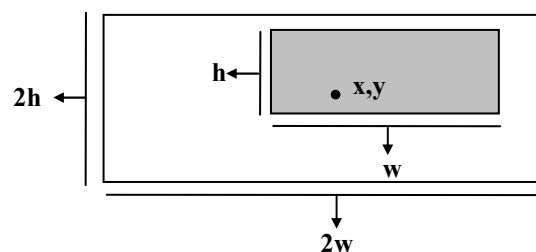
Στα πλαίσια της παρούσας διπλωματικής εργασίας δεν επιλέξαμε κάποιον Αλγόριθμο Εξαντλητικής Έρευνας ή Τοπικής Έρευνας, οι οποίοι παρουσιάζουν σημαντικά προβλήματα στην τοποθέτηση των ετικετών σημειακών αντικειμένων και μεγάλο υπολογιστικό κόστος εκτέλεσης, όπως έχουμε ήδη αναφέρει. Για την επίλυση του ΠΒΤΕ επιλέξαμε την ευριστική μέθοδο της **Προσομοίωσης Ανόπτησης**, η οποία είναι εμπνευσμένη από τη στατιστική μηχανική. Η συγκεκριμένη μέθοδος θα αναλυθεί διεξοδικά στο Κεφάλαιο 4.

## **3.4 Αλγόριθμοι Εύρεσης Ελεύθερου Χώρου για Ετικέτες Σημειακών Αντικειμένων**

Στην προηγούμενη ενότητα αναφερθήκαμε σε διάφορους αλγόριθμους τοποθέτησης ετικετών που περιγράφουν σημειακά γραφικά αντικείμενα. Σε αυτή την

ενότητα θα αναφερθούμε σε ορισμένους αλγόριθμους, οι οποίοι μας δίνουν τη δυνατότητα να αυξήσουμε τον αριθμό των δυνατών θέσεων τοποθέτησης των ετικετών, κάτι που με τη σειρά του αυξάνει τις πιθανότητες να τοποθετηθούν περισσότερες ετικέτες χωρίς επικαλύψεις.

Ο χώρος τοποθέτησης μίας ετικέτας μήκους  $w$  και ύψους  $h$  που ανήκει στο σημείο με συντεταγμένες  $\langle x, y \rangle$  μπορεί να τοποθετηθεί οπουδήποτε μέσα στο ορθογώνιο  $r = \langle x - w, y - h, x + w, y + h \rangle$ , όπως παρουσιάζεται στο Σχήμα 3.10. Το ορθογώνιο ονομάζεται **μεγεθυμένο ορθογώνιο (Augmented Rectangle)**.



Σχήμα 3.10 – Τοποθέτηση ετικέτας σημειακού αντικειμένου

Όταν μία ετικέτα επικαλύπτεται από άλλες ετικέτες ή άλλα γραφικά αντικείμενα θα πρέπει να ελεγχθεί το ορθογώνιο τοποθέτησής της έτσι, ώστε να αφαιρεθούν τα τμήματα που επισκιάζονται από άλλες ετικέτες και στη συνέχεια να επανατοποθετηθεί η ετικέτα σε μία από τις ελεύθερες περιοχές που απομένουν, καθώς ο αλγόριθμος επίλυσης του ΠΒΤΕ εκτελείται.

Ο προσδιορισμός του ελεύθερου χώρου μπορεί να γίνει με δύο διαφορετικούς τρόπους:

- 1) Με τη χρήση ενός σημείου αναφοράς για κάθε θέση τοποθέτησης της ετικέτας και υπολογισμό του γεωμετρικού τύπου των σημείων των οποίων οι αντίστοιχες θέσεις δεν παρουσιάζουν επικαλύψεις.
- 2) Με υπολογισμό του ελεύθερου χώρου τοποθέτησης ως ένα σύνολο ορθογωνίων μέσα στα οποία μπορεί να τοποθετηθεί η ετικέτα χωρίς να παρουσιάζει επικαλύψεις.



Είναι φανερό ότι η πρώτη μέθοδος εξαρτάται από τις διαστάσεις της εκάστοτε ετικέτας, ενώ η δεύτερη είναι ανεξάρτητη από αυτές. Στην ενότητα αυτή θα περιγράψουμε δύο αλγόριθμους υπολογισμού του ελεύθερου χώρου, ένα για κάθε μέθοδο.

### 3.4.1 Τυποποίηση του Προβλήματος Ελεύθερου Χώρου

Αρχικά, είναι αναγκαίο να δώσουμε μια γενική περιγραφή του προβλήματος της εύρεσης ελεύθερου χώρου έτσι ώστε στη συνέχεια να μελετήσουμε την επίλυση του συγκεκριμένου προβλήματος για την τοποθέτηση ετικετών σημειακών αντικειμένων.

Αν θεωρήσουμε μια περιοχή  $R$  ενός διδιάστατου χώρου και ένα σύνολο  $S = \{r_i | i \in I\}$  αντικειμένων στο χώρο αυτό, το πρόβλημα εύρεσης του ελεύθερου χώρου είναι ο προσδιορισμός της περιοχής  $r$  του  $R$  για την οποία ισχύουν οι ακόλουθες συνθήκες:

$$1) \quad r \cap r_i = \emptyset, \forall i \in I$$

$$2) \quad r \cup \left( \bigcup_{i \in I} r_i \right) = R$$

Αν ο χώρος  $R$  δεν περιέχει εξολοκλήρου τις περιοχές του συνόλου  $S$ , τότε είναι απαραίτητο να τροποποιήσουμε το σύνολο  $S$  ώστε να αποτελείται από την τομή των στοιχείων του με το  $R$ . Η τομή αυτή είναι το σύνολο  $S' = \{r'_i = r_i \cap R | i \in I\}$ .

Μία χρήσιμη πρόταση που θα χρησιμοποιήσουμε στην παρουσίαση των αλγόριθμων υπολογισμού του ελεύθερου χώρου είναι η εξής:

Δύο ορθογώνια  $r_1 = \langle a_1, b_1, c_1, d_1 \rangle$  και  $r_2 = \langle a_2, b_2, c_2, d_2 \rangle$  τέμνονται, αν και μόνο αν ισχύουν οι ακόλουθες συνθήκες:

- 1) Το αριστερό όριο του πρώτου είναι μικρότερο ή ίσο με το δεξί όριο του δεύτερου.

- 2) Το πάνω όριο του πρώτου είναι μικρότερο ή ίσο με το κάτω όριο του δεύτερου.
- 3) Το δεξί όριο του πρώτου είναι μεγαλύτερο ή ίσο με το αριστερό όριο του δεύτερου.
- 4) Το κάτω όριο του πρώτου είναι μεγαλύτερο ή ίσο με το πάνω όριο του δεύτερου.

Δηλαδή:

$$r_1 \cap r_2 \neq \emptyset \Leftrightarrow \begin{cases} a_1 \leq c_2 \\ b_1 \leq d_2 \\ c_1 \geq a_2 \\ d_1 \geq b_2 \end{cases}$$

Η απόδειξη της παραπάνω πρότασης αναφέρεται στο [Μουμ97].

### 3.4.2 Αλγόριθμος Εύρεσης Ελεύθερου Χώρου με Σημείο Αναφοράς

Σε αυτή την ενότητα θα αναφερθούμε στον υπολογισμό του ελεύθερου χώρου με σημείο αναφοράς το κέντρο της ετικέτας. Το γενικό πρόβλημα στην περίπτωση αυτή τροποποιείται ως εξής:

- 1) Ο χώρος  $R$  είναι το σύνολο των σημείων αναφοράς, δηλαδή το ορθογώνιο  $r_c = \langle x, y, x + w, y + h \rangle$ .
- 2) Για κάθε ορθογώνιο  $r_i = \langle a_i, b_i, c_i, d_i \rangle, i \in I$  η τομή του με το χώρο των σημείων αναφοράς υπολογίζεται αν το επεκτείνουμε αριστερά κατά  $w$  και προς τα κάτω κατά  $h$  και έπειτα πάρουμε την τομή του επεκταμένου αυτού ορθογωνίου με το  $r_c$ , δηλαδή είναι  $r_{c,i} = \langle x, y, x + w, y + h \rangle \cap \langle a_i - w, b_i - h, c_i, d_i \rangle \forall i \in I$ .

Οπότε ο αλγόριθμος υπολογισμού του ελεύθερου χώρου με σημείο αναφοράς είναι ο ακόλουθος:

Για κάθε ορθογώνιο  $r_i = \langle a_i, b_i, c_i, d_i \rangle, i \in I$

Υπολόγισε το ορθογώνιο  $r_{c,i} = \langle a_i - w, b_i - h, c_i, d_i \rangle$

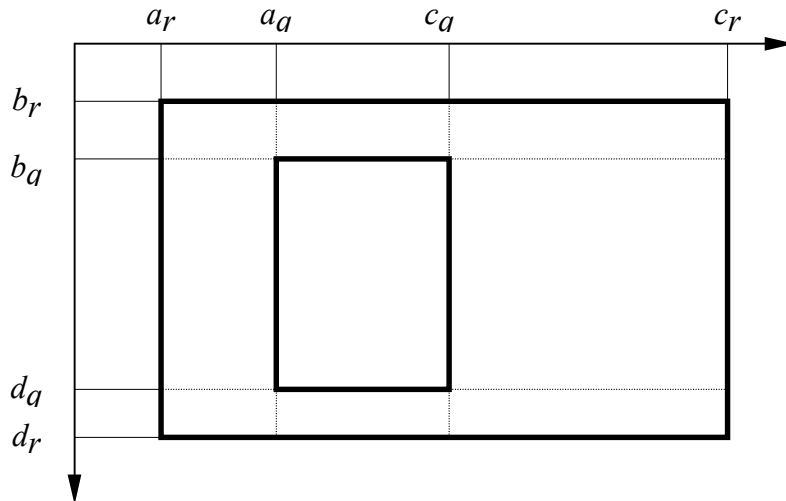
Υπολόγισε το  $\langle x, y, x + w, y + h \rangle - \bigcup_{i \in I} r_{c,i}$

Στη συνέχεια θα πρέπει να κατασκευάσουμε ένα αλγόριθμο υπολογισμού του  $\langle x, y, x + w, y + h \rangle - \bigcup_{i \in I} r_{c,i}$ .

Ένας τρόπος να προχωρήσουμε στο υπολογισμό αυτό είναι η εύρεση του συνόλου  $R$  των ορθογωνίων που αποτελούν τον ελεύθερο χώρο. Για να προχωρήσουμε στην υλοποίηση του αλγόριθμου είναι αναγκαίο να διευκρινίσουμε ότι αν το ορθογώνιο  $r = \langle a_r, b_r, c_r, d_r \rangle$  τέμνεται με το ορθογώνιο  $q = \langle a_q, b_q, c_q, d_q \rangle$ , τότε ο χώρος του  $r$  που δεν ανήκει στο  $q$  είναι το σύνολο των ορθογωνίων:

$$\langle a_r, b_r, c_r, b_q \rangle, \langle a_r, d_q, c_r, d_r \rangle, \langle a_r, b_r, a_q, d_r \rangle, \langle c_q, b_r, c_r, d_r \rangle$$

Υπάρχει η περίπτωση κάποια από τα παραπάνω ορθογώνια να είναι κενά, όταν κάποιες από τις γωνίες των δύο ορθογωνίων ταυτίζονται, με αποτέλεσμα το σύνολο να αποτελείται από λιγότερα από τέσσερα ορθογώνια. Στο Σχήμα 3.11 παρουσιάζεται η περίπτωση στην οποία το ορθογώνιο  $q$  περιέχεται πλήρως μέσα στο ορθογώνιο  $r$ . Τα τέσσερα ορθογώνια που προσδιορίζουν τον ελεύθερο χώρο, οριοθετούνται από τις πλευρές του  $r$  και τις διακεκομμένες γραμμές.



Σχήμα 3.11 – Ο ελεύθερος χώρος μεταξύ των ορθογωνίων  $r$  και  $q$

Συνεπώς, το δεύτερο μέρος του αλγόριθμου έχει ως εξής:

(1) Θέσσε  $R = \{r_c\}$

(2) Για κάθε  $r_{c,i} = \langle a_i, b_i, c_i, d_i \rangle, i \in I$  επανέλαβε τα εξής:

(α) Έστω  $R_1 = \{r \in R \mid r \cap r_{c,i} \neq \emptyset\}$

(β) Θέσσε  $R_2 = \emptyset$

(γ) Για κάθε  $r = \langle a, b, c, d \rangle \in R_1$

(i) Θέσσε

$$R_2 = R_2 \cup \left\{ \langle a, b, c, b_i \rangle, \langle a, b, a_i, d \rangle, \langle a, d_i, c, d \rangle, \langle c_i, b, c, d \rangle \right\}$$

(δ) Τέλος

(ε) Θέσσε  $R = (R - R_1) \cup R_2$

(3) Τέλος

Στη γραμμή 2(γ)(i) αφαιρούμε από το σύνολο  $R_2$  τα κενά ορθογώνια που πιθανώς υπάρχουν, όπως αναφέραμε στην προηγούμενη παράγραφο.

Στη συνέχεια θα προχωρήσουμε στην μελέτη της πολυπλοκότητας του παραπάνω αλγόριθμου ως προς το πλήθος  $n = |I|$  των ορθογωνίων που επικαλύπτουν κάθε ετικέτα. Ο υπολογισμός του πρώτου μέρους του αλγόριθμου είναι φανερό ότι απαιτεί γραμμικό χρόνο ως προς το  $n$ . Το μεγαλύτερο μέρος του χρόνου εκτέλεσης του δεύτερου βήματος του αλγόριθμου που αναφέρεται στον υπολογισμό του ελεύθερου χώρου καταναλώνεται από το βρόγχο των γραμμών 2-3, ο οποίος θα εκτελεστεί  $n$  φορές. Ο υπολογισμός του πλήθους των ορθογωνίων του συνόλου  $R$  που τέμνονται με το ορθογώνιο  $r_{c,i}$  απαιτεί λογαριθμικό χρόνο εκτέλεσης ως προς  $n$  με τη χρήση μιας κατάλληλης δομής ευριτηριασμού, όπως το τετραδικό δέντρο (Quad Tree<sup>6</sup>). Ο χρόνος εκτέλεσης της γραμμής 2(ε) πάλι με τη χρήση κατάλληλων δομών μπορεί να είναι σταθερός, οπότε να μην επηρεάζει την πολυπλοκότητα του αλγόριθμου. Η ανακύκλωση των γραμμών 2(γ)-2(δ) απαιτεί γραμμικό χρόνο εκτέλεσης ως προς το πλήθος των ορθογωνίων του συνόλου  $R_I$  που βρέθηκαν στη γραμμή 2(α). Συνήθως το πλήθος αυτό είναι ανάλογο του  $n$ , συνεπώς η εκτέλεση του βρόχου των γραμμών 2-3 απαιτεί χρόνο  $O(n^2)$ . Αν δεχτούμε όμως ότι το πλήθος των ορθογωνίων του συνόλου  $R_I$  δεν υπερβαίνει μία σταθερά  $c$ , τότε το κόστος της ανακύκλωσης γίνεται  $O(nc \log n)$  δηλαδή  $O(n \log n)$ .

Το κυριότερο μειονέκτημα του παραπάνω αλγόριθμου είναι ότι η αναπαράσταση του ελεύθερου χώρου ως ένα σύνολο ορθογωνίων που επικαλύπτονται μεταξύ τους, κάτι που είναι πιθανό να επηρεάζει αρνητικά τα αποτελέσματα του αλγόριθμου επίλυσης του ΠΒΤΕ. Θα ήταν προτιμότερο αν η παραπάνω αναπαράσταση του ελεύθερου χώρου γινόταν ως ένα σύνολο μη επικαλυπτόμενων ορθογωνίων, κάτι που επιτρέπει στον αλγόριθμο επίλυσης του ΠΒΤΕ να διαλέξει τυχαία μία νέα θέση της ετικέτας εξασφαλίζοντας ίδια πιθανότητα επιλογής για κάθε θέση. Στην περίπτωση των μη επικαλυπτόμενων ορθογωνίων αυτό εξασφαλίζεται με την αντιστοίχιση σε κάθε ορθογώνιο μίας πιθανότητας επιλογής ίσης με το εμβαδόν του.

---

<sup>6</sup> Το τετραδικό δέντρο μία ιεραρχική δομή που βασίζεται στην αρχή της αναδομικής αποσύνθεσης του χώρου. Στην υποενότητα 2.6.2 παρουσιάστηκε μία λεπτομερής περιγραφή του.

### 3.4.3 Αλγόριθμος Εύρεσης Ελεύθερου Χώρου με Μέγιστα Ελεύθερα Ορθογώνια

Στην ενότητα αυτή θα περιγράψουμε τον αλγόριθμο υπολογισμού του ελεύθερου χώρου με τη χρήση μεγίστων ελεύθερων ορθογώνιων. Το γενικό πρόβλημα στην περίπτωση αυτή τροποποιείται ως εξής:

- 1) Ο χώρος  $R$  είναι το ορθογώνιο  $\langle left_R, top_R, right_R, bottom_R \rangle$ .
- 2) Οι περιοχές  $r_i, i \in S$  που επικαλύπτουν το χώρο  $R$  είναι ορθογώνια και τα συμβολίζουμε ως  $r_i = \langle left_i, top_i, right_i, bottom_i \rangle$ .
- 3) Το τελικό αποτέλεσμα εκφράζεται ως ένα σύνολο  $S$  μεγίστων ελεύθερων ορθογώνιων.

Ένα ορθογώνιο  $r'$  ονομάζεται *Ελεύθερο Ορθογώνιο* αν και μόνο αν ισχύουν οι παρακάτω ιδιότητες:

- 1)  $r' \subseteq R$
- 2)  $\forall r_i, i \in I, r' \cap r_i = \emptyset$

Επιπλέον, ένα ορθογώνιο  $r'$  ονομάζεται *Μέγιστο Ελεύθερο Ορθογώνιο* αν και μόνο αν ισχύουν οι παρακάτω ιδιότητες:

- 1) Το  $r'$  είναι ελεύθερο ορθογώνιο.
- 2) Οποιοδήποτε ορθογώνιο περιέχει εξολοκλήρου το  $r'$  δεν είναι ελεύθερο.

Αν εκφράσουμε τον ελεύθερο χώρο ως ένα σύνολο μεγίστων ελεύθερων ορθογώνιων, τότε είναι προφανές ότι μπορούμε να τοποθετήσουμε την ετικέτα σε οποιοδήποτε ορθογώνιο του συνόλου αυτού το οποίο έχει διαστάσεις μεγαλύτερες από την ετικέτα.

Στη συνέχεια θα περιγράψουμε τον συγκεκριμένο αλγόριθμο.

$$(1) \quad \Theta \acute{\epsilon} \sigma \epsilon \quad R = \{r_c\}$$

$$(2) \quad \Gamma \iota \alpha \quad \kappa \acute{\alpha} \theta \epsilon \quad r_{c,i} = \langle a_i, b_i, c_i, d_i \rangle, i \in I$$

$$(\alpha) \quad \text{\texttt{E}} \sigma \tau \omega \quad R_1 = \{r \in R \mid r \cap r_{c,i} \neq \emptyset\}$$

$$(\beta) \quad \Theta \acute{\epsilon} \sigma \epsilon \quad R_2 = \emptyset$$

$$(\gamma) \quad \Gamma \iota \alpha \quad \kappa \acute{\alpha} \theta \epsilon \quad r = \langle a, b, c, d \rangle \in R_1$$

$$(i) \quad \Theta \acute{\epsilon} \sigma \epsilon \quad R_2 = R_2 \cup \{\langle a, b, c, b_i \rangle, \langle a, b, a_i, d \rangle, \langle a, d_i, c, d \rangle, \langle c_i, b, c, d \rangle\}$$

$$(\delta) \quad \tau \acute{\epsilon} \lambda \omicron \varsigma$$

$$(\epsilon) \quad \Theta \acute{\epsilon} \sigma \epsilon \quad R = (R - R_1) \cup R_2$$

(3) Τέλος

Ο συγκεκριμένος αλγόριθμος είναι αυτός που υλοποιήσαμε στα πλαίσια της παρούσας διπλωματικής εργασίας, λόγω της ευκολίας που παρουσιάζει τόσο στη χρήση όσο και στην υλοποίησή της και του μικρού υπολογιστικού κόστους που απαιτεί. Ο αλγόριθμος εύρεσης των ελεύθερων ορθογωνίων είναι ακριβώς το πρώτο μέρος του αλγορίθμου για την εύρεση του ελεύθερου χώρου με σημεία αναφοράς με τη διαφορά ότι ελέγχουμε το σύνολο των αποτελεσμάτων έτσι ώστε να απορρίπτουμε τα ορθογώνια που περιέχονται εξολοκλήρου σε άλλα ορθογώνια του αποτελέσματος. Ο αλγόριθμος αυτός έχει ήδη αποδειχθεί στην προηγούμενη υποενότητα ότι παρουσιάζει πολυπλοκότητα  $O(n \log n)$ .

### 3.5 Περίληψη

Στο Κεφάλαιο αυτό αναφερθήκαμε σε ορισμένους Αλγόριθμους Πλήρους Έρευνας σχετικά με την εύρεση λύσης στο ΠΒΤΕ που είχαν αναφερθεί στο παρελθόν σε δημοσιεύσεις. Περιγράψαμε τις μεθόδους προσέγγισης του προβλήματος που

χρησιμοποιούν, τη λογική πάνω στην οποία βασίζονται και αναλύσαμε τα προβλήματα τοπικού ελάχιστου στα οποία εγκλωβίζονται, χωρίς να μπορούν να βρουν αποδεκτές λύσεις. Στην παραπάνω ανάλυση βασιστήκαμε για την τελική επιλογή του αλγόριθμου επίλυσης του ΠΒΤΕ που περιγράφεται στο επόμενο Κεφάλαιο , ο οποίος βασίζεται σε ευριστικές μεθόδους και παρουσιάζει σημαντικά πλεονεκτήματα σε σύγκριση με τους παραπάνω αλγόριθμους, πλεονεκτήματα στα οποία έχουμε ήδη αναφερθεί.

Επιπλέον, αναλύσαμε το Πρόβλημα Εύρεσης Ελεύθερου Χώρου για σημειακές ετικέτες. Αναφερθήκαμε στην πολυπλοκότητα των αλγορίθμων που επιλύουν το πρόβλημα αυτή, η οποία κρίνεται ικανοποιητική ( $O(n \log n)$ ) και στην αναγκαιότητα χρησιμοποίησης των συγκεκριμένων αλγορίθμων για το συνεχές μοντέλο τοποθέτησης σημειακών ετικετών, καθώς αυξάνουν σημαντικά τον αριθμό των δυνατών θέσεων τοποθέτησης των ετικετών.



## **Κεφάλαιο 4**

# **Υλοποίηση Αλγορίθμων**

### **4.1 Γενικά**

Στο Κεφάλαιο αυτό, αρχικά, θα αναφερθούμε στο ΠΒΤΕ και στις μεθόδους που επιλέξαμε για την επίλυση του συγκεκριμένου προβλήματος. Αφού στο προηγούμενο Κεφάλαιο καταλήξαμε στο συμπέρασμα ότι οι αλγόριθμοι που είχαν έως τώρα προταθεί για την επίλυση του ΠΒΤΕ για ετικέτες σημειακών αντικειμένων παρουσιάζουν πολλά προβλήματα και συνεπώς είναι ανεπαρκής, θα παρουσιάσουμε την ευριστική μέθοδο Προσομοίωσης Ανόπτησης που είναι εμπνευσμένη από τη στατιστική μηχανική.

Στη συνέχεια ακολουθεί μία αναλυτική περιγραφή του τρόπου απεικόνισης των ετικετών γραμμικών αντικειμένων, η οποία βασίζεται στις καμπύλες Bézier. Τέλος, θα παρουσιάσουμε τον αλγόριθμο επίλυσης του ΠΒΤΕ για τις ετικέτες γραμμικών αντικειμένων που αναπτύξαμε και ενσωματώσαμε στον υπόλοιπο αλγόριθμο. Ο λόγος που προσφύγαμε σε διαφορετικό αλγόριθμο για τις γραμμικές ετικέτες είναι ο εντελώς διαφορετικός τρόπος απεικόνισής τους σε σύγκριση με τις σημειακές ετικέτες.

### **4.2 Το πρόβλημα των ετικετών**

Κατά την τοποθέτηση των ετικετών είναι απαραίτητο να μην υπάρχουν επικαλύψεις μεταξύ τους. Συνεπώς δημιουργείται το πρόβλημα εύρεσης της βέλτιστης θέσης για κάθε ετικέτα έτσι ώστε να μην καλύπτει τις γειτονικές της. Η

πολυπλοκότητα του συγκεκριμένου προβλήματος αυξάνεται εκθετικά με το πλήθος των αντικειμένων, κάτι που οφείλεται στο γεγονός ότι η τοποθέτηση μιας ετικέτας μπορεί να έχει συνέπειες σε όλες τις υπόλοιπες λόγω των επικαλύψεων.

Το ΠΒΤΕ θεωρείται ένα συνδυαστικό πρόβλημα βελτιστοποίησης, οπότε πρέπει να οριστεί ο χώρος αναζήτησης και η αντικειμενική συνάρτηση που πρέπει να βελτιστοποιήσουμε.

Για τις ετικέτες σημειακών αντικειμένων, όπως έχουμε ήδη αναφέρει στο Κεφάλαιο 3, ο χώρος αναζήτησης είναι ένα ορθογώνιο με κέντρο το σημείο και διαστάσεις  $(2x, 2y)$ , όπου  $(x, y)$  οι διαστάσεις του ορθογωνίου που περικλείει την ετικέτα.

Το δεύτερο χαρακτηριστικό του ΠΒΤΕ είναι η αντικειμενική συνάρτηση. Η συνάρτηση αυτή αντιστοιχίζει κάθε πιθανή θέση μιας ετικέτας με έναν αριθμό που εκφράζει την ποιότητα αυτής της θέσης σε σχέση με τις υπόλοιπες δυνατές θέσεις.

### **4.3 Η μέθοδος της Προσομοίωσης Ανόπτησης**

Κάθε ευριστική μέθοδος είναι μάλλον αδύνατον να βρει μια ολική βέλτιστη λύση. Συνήθως μια μέθοδος αναζήτησης με συνεχή βελτίωση στην τιμή της αντικειμενικής συνάρτησης βρίσκει μόνο τοπικές βέλτιστες λύσεις, χωρίς να είναι δυνατόν να επιτευχθεί περαιτέρω βελτίωση χωρίς προηγούμενη χειροτέρευση. Μια τεχνική για να περιορίσουμε το πρόβλημα είναι να ξεκινήσουμε με διαφορετικές αρχικές λύσεις, αυξάνοντας με αυτόν τον τρόπο την πιθανότητα εύρεσης πιο αποδεκτών τοπικών βέλτιστων λύσεων. Μια εναλλακτική τεχνική είναι να επιτρέψουμε μια προσωρινή χειροτέρευση στη λύση.

Η μέθοδος της **Προσομοίωσης Ανόπτησης** (συντομογραφικά **ΠΑ**) στηρίζεται σε αυτή ακριβώς την τεχνική. Η ιδέα που ακολουθείται είναι η εξής: Έστω ένα σύστημα που αποτελείται από ένα μεγάλο πλήθος σωματιδίων σταθερού όγκου σε μια θερμοκρασία  $T$ . Εφόσον τα σωματίδια κινούνται, το σύστημά μας μπορεί να βρίσκεται σε διάφορες καταστάσεις ενέργειας. Η πιθανότητα το σύστημα να βρίσκεται στην κατάσταση ενέργειας  $E$  δίνεται από την κατανομή Boltzmann:

$$f(E) = \frac{e^{-\frac{E}{K_B \cdot T}}}{z(T)}$$

όπου  $z(T)$  είναι ένας συντελεστής κανονικοποίησης και  $K_B$  η σταθερά του Boltzmann.

Η διαδικασία που υλοποιεί τον αλγόριθμο ΠΑ είναι η εξής:

Προσομοίωση Ανόπτωσης

Βρες μια αρχική λύση  $\Lambda$

Προσδιόρισε μια αρχική θερμοκρασία  $T$

Προσδιόρισε έναν συντελεστή επανάληψης  $\epsilon$

Όσο δεν ικανοποιείται η συνθήκη τερματισμού

Επανάλαβε  $\epsilon$  φορές

Τροποποίησε τυχαία τη  $\Lambda$  και θέσε  $\Lambda'$  τη νέα λύση

$\Delta E = \text{κόστος}(\Lambda') - \text{κόστος}(\Lambda)$

Έστω  $x$  τυχαίος αριθμός στο διάστημα  $[0,1]$

Αν  $\Delta E < 0$  ή  $x < e^{-\frac{E}{K_B \cdot T}}$  τότε

$\Lambda = \Lambda'$

τέλος

τέλος

Άλλαξε τα  $T$  και  $\epsilon$

τέλος

Λ είναι η λύση που βρέθηκε

τέλος

Η θερμοκρασία συνήθως μειώνεται σύμφωνα με τη σχέση  $T=\gamma T$ , όπου  $0<\gamma<1$ , ενώ ο συντελεστής επανάληψης  $\varepsilon$  αυξάνεται σύμφωνα με τη σχέση  $\varepsilon=\alpha\varepsilon$ , όπου  $1<\alpha<2$ . Ειδικότερα αυτό το σχήμα τροποποίησης της θερμοκρασίας και του συντελεστή επανάληψης ονομάζεται **γεωμετρική ανόπτηση**. Ο αλγόριθμος τερματίζεται, όταν η λύση δεν μεταβάλλεται για κάποιο προκαθορισμένο αριθμό συνεχόμενων θερμοκρασιών. Το μεγάλο πλεονέκτημα του συγκεκριμένου αλγόριθμου ΠΑ είναι η απλή υλοποίησή του.

## 4.4 Αλγόριθμος Επίλυσης του ΠΒΤΕ

Στο [Chms95] γίνεται μια πολύ καλή περιγραφή του ΠΒΤΕ για σημειακά αντικείμενα. Η πολυπλοκότητα του προβλήματος αποδεικνύεται ότι είναι NP-Hard. Στο σύγγραμμα αυτό γίνεται ακόμη μια ανασκόπηση των κυριότερων αλγόριθμων επίλυσης του ΠΒΤΕ και προτείνονται δύο νέες μέθοδοι. Από τα αποτελέσματα της πειραματικής μελέτης που περιγράφονται, φτάνουμε στο συμπέρασμα ότι η πιο αποδοτική μέθοδος είναι η ΠΑ.

Στη συνέχεια δίνουμε το γενικό αλγόριθμο που προτείνεται στο [Chms95]:

Για κάθε αντικείμενο

    Τοποθέτησε την ετικέτα του σε οποιαδήποτε διαθέσιμη θέση

τέλος

Όσο ο ρυθμός βελτίωσης της λύσης είναι μεγαλύτερος από ένα κατώφλι

    Μείωσε τη θερμοκρασία  $T$  σύμφωνα με το πρόγραμμα ψύξης

    Διάλεξε μια ετικέτα και βάλ' την σε μια νέα θέση

Υπολόγισε τη μεταβολή  $\Delta E$  στην αντικειμενική συνάρτηση

Αν προκύπτει χειροτέρευση στην αντικειμενική συνάρτηση

Ακύρωσε τη νέα τοποθέτηση με πιθανότητα  $P = 1 - e^{-\Delta E/T}$

Τέλος

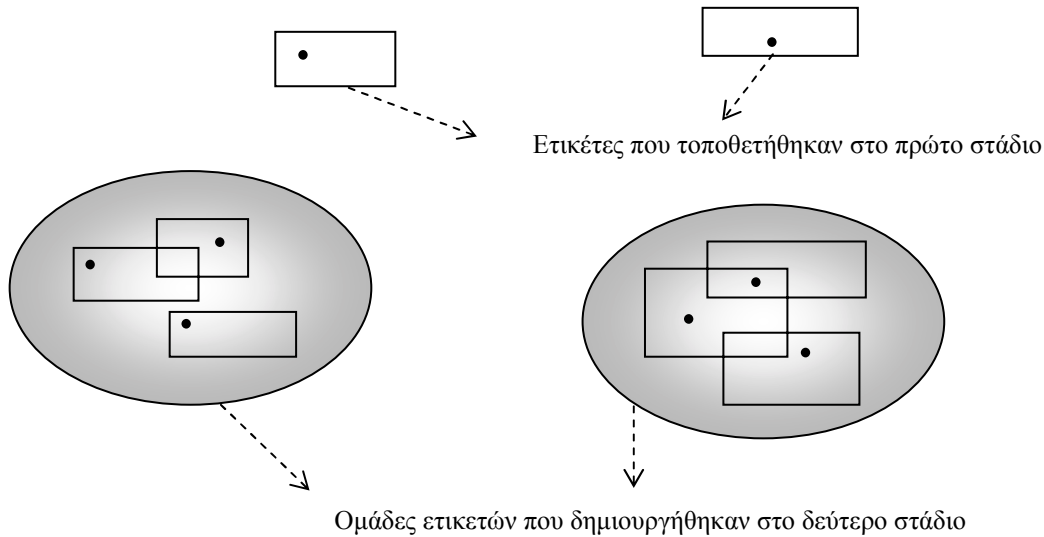
Τέλος

Όπως φαίνεται από τον παραπάνω αλγόριθμο και σύμφωνα με τα όσα προαναφέραμε στην προηγούμενη ενότητα, η επίλυση του ΠΒΤΕ αποτελείται από τέσσερα τμήματα: **την αρχικοποίηση, την αντικειμενική συνάρτηση, τη μέθοδο τροποποίησης των λύσεων και το πρόγραμμα ανόπτησης.** Στη συνέχεια θα περιγράψουμε τα τέσσερα αυτά τμήματα χωριστά.

#### 4.4.1 Αρχικοποίηση

Στο [Chms95] αναφέρεται ότι ακόμα και αν αρχικά γίνει μια τοποθέτηση των ετικετών με κάποιον άλλο αλγόριθμο επίλυσης του ΠΒΤΕ, δεν επέρχεται σημαντική βελτίωση στο τελικό αποτέλεσμα, ενώ η υλοποίηση καθίσταται πιο πολύπλοκη. Όμως, είναι πιθανό κάποιες ετικέτες να μπορούν να τοποθετηθούν εξαρχής σε θέσεις τέτοιες ώστε να μην επηρεάζουν καμία άλλη. Συνεπώς αφαιρώντας από την είσοδο του αλγορίθμου τις ετικέτες που μπορούν να τοποθετηθούν χωρίς να επικαλύπτουν άλλες ετικέτες, ο αλγόριθμος τερματίζει ταχύτερα.

Μια ακόμη βελτίωση που χρησιμοποιούμε είναι η ομαδοποίηση των ετικετών που δεν έχουν αφαιρεθεί από το πρώτο μέρος της αρχικοποίησης που μόλις περιγράψαμε. Οι ετικέτες που αποτελούν μια ομάδα κατά το στάδιο αυτό, δεν επηρεάζουν σε καμία περίπτωση την τοποθέτηση ετικετών που ανήκουν σε κάποια άλλη ομάδα ή τις ετικέτες που έχουν ήδη τοποθετηθεί από το πρώτο στάδιο της αρχικοποίησης. Ένα παράδειγμα ομαδοποίησης δίνεται στο Σχήμα 4.1 που ακολουθεί.



Σχήμα 4.1 - Παράδειγμα ομαδοποίησης ετικετών κατά το δεύτερο στάδιο της αρχικοποίησης

Η ομαδοποίηση των ετικετών προφανώς επιταχύνει το συνολικό χρόνο εκτέλεσης του αλγορίθμου, αφού δημιουργούνται επιμέρους ΠΒΤΕ μικρότερου μεγέθους που αθροιστικά έχουν μικρότερο χρόνο επίλυσης από το αρχικό ΠΒΤΕ.

#### 4.4.2 Αντικειμενική συνάρτηση

Η αντικειμενική συνάρτηση επηρεάζει κατά μεγάλο ποσοστό την απόδοση του αλγορίθμου τόσο από πλευράς ταχύτητας εκτέλεσης όσο και από πλευράς ποιότητας του τελικού αποτελέσματος. Επειδή η επίλυση του ΠΒΤΕ με τη μέθοδο που επιλέξαμε, στηρίζεται σε στατιστικά δεδομένα, τα οποία συλλέγονται ύστερα από ένα μεγάλο πλήθος δοκιμών, η αντικειμενική συνάρτηση για να δώσει ικανοποιητικά αποτελέσματα πρέπει να είναι τέτοια ώστε ο υπολογισμός το  $\Delta E$  να επιτυγχάνεται όσο το δυνατόν γρηγορότερα.

Μια αρκετά αποδοτική αντικειμενική συνάρτηση, την οποία και τελικά επιλέξαμε να χρησιμοποιήσουμε στην υλοποίηση του αλγορίθμου, είναι η *διαφορά του εμβαδού των επικαλύψεων που έχουμε με τη νέα θέση της ετικέτας μείον το εμβαδόν των επικαλύψεων με που είχαμε με την παλιά θέση της ετικέτας*.

### 4.4.3 Τροποποίηση λύσεων

Μια καλή τακτική είναι αντί να επιλέγουμε τυχαία μια ετικέτα για επανατοποθέτηση είναι να επιλέγουμε μια ετικέτα που εμφανίζει επικαλύψεις με άλλες. Με αυτόν τον τρόπο επικεντρώνουμε το ενδιαφέρον μας μόνο στις περιοχές του χάρτη που παρουσιάζουν προβλήματα επικάλυψης ετικετών και κατά συνέπεια βελτιώνεται ο χρόνος σύγκλισης του αλγόριθμου.

### 4.4.4 Πρόγραμμα Ανόπτησης

Το ΠΑ που περιγράφεται στο [Chms95] και που τελικά χρησιμοποιήσαμε είναι το εξής:

Η θερμοκρασία εκκίνησης επιλέγεται έτσι ώστε όταν  $\Delta E = 1$ , να ισχύει  $P=2/3$ . Η εξίσωση  $P = 1 - e^{-\Delta E/T}$  με τις προαναφερόμενες τιμές μετατρέπεται στη  $\frac{2}{3} = 1 - e^{-\frac{1}{T}}$ , οπότε λύνοντας την, βρίσκουμε ότι  $T=0,91$ . Σε κάθε θερμοκρασία επιχειρείται  $20n$  φορές να επανατοποθετηθούν ετικέτες, όπου  $n$  το πλήθος των ετικετών. Στη συνέχεια η θερμοκρασία μειώνεται κατά 10%. Επιπρόσθετα, η θερμοκρασία μειώνεται, αν γίνουν  $5n$  επιτυχείς επανατοποθετήσεις, ακόμα και αν δεν έχουν εξαντληθεί οι  $20n$  προσπάθειες. Η παραπάνω διαδικασία επαναλαμβάνεται για το πολύ 50 διαφορετικές θερμοκρασίες. Όμως ο αλγόριθμος τερματίζεται και σε μια ακόμη περίπτωση, αν δηλαδή εξαντληθούν οι  $20n$  προσπάθειες χωρίς να έχει επιτευχθεί έστω και μία επανατοποθέτηση.

Η χρησιμοποίηση του παραπάνω αλγόριθμου αντί κάποιου άλλου, ανάλογου κρίθηκε ως η καταλληλότερη, επειδή η ποιότητα των λύσεων που μας προσφέρει σε σχέση με την απόδοσή του θεωρήθηκε η καλύτερη σε σύγκριση με άλλα, πιο χρονοβόρα προγράμματα ανόπτησης που τελικά ελάχιστα βελτιώνουν την ποιότητα της τελικής λύσης, όπως παρουσιάστηκε στο [Μουμ97].

Στη συνέχεια παρουσιάζουμε το γενικό περίγραμμα του αλγόριθμου λύσης του ΠΒΤΕ, σύμφωνα με όσα έχουμε έως τώρα αναφέρει σχετικά με τα τέσσερα τμήματα που τον αποτελούν.

/\* Αρχικοποίηση \*/

Για κάθε αντικείμενο

Σημείωσε ως "ΜΗ ΕΛΕΓΜΕΝΟ" και "ΜΗ ΤΟΠΟΘΕΤΗΜΕΝΟ"

Τέλος

Φτιάξε μια κενή στοίβα "ΠΡΟΒΛΗΜΑΤΑ"

Όσο υπάρχει αντικείμενο "ΜΗ ΕΛΕΓΜΕΝΟ"

Διάλεξε ένα "ΜΗ ΕΛΕΓΜΕΝΟ" αντικείμενο

Φτιάξε μια στοίβα  $\Sigma$  που περιέχει μόνο το αντικείμενο αυτό

Φτιάξε μια κενή στοίβα  $\Sigma_1$

Όσο η στοίβα  $\Sigma$  έχει στοιχεία

Βγάλε από την  $\Sigma$  το πρώτο αντικείμενο, έστω  $\alpha$

Σημείωσε το  $\alpha$  ως "ΕΛΕΓΜΕΝΟ"

Έστω  $\Sigma_A$  το σύνολο των αντικειμένων που είναι "ΜΗ ΤΟΠΟΘΕΤΗΜΕΝΑ" και οι ετικέτες τους επικαλύπτονται με το  $\alpha$

Για κάθε αντικείμενο  $\beta$  το  $\Sigma_A$

Ακύρωσε τις θέσεις της ετικέτας του  $\alpha$  που επικαλύπτονται με θέσεις της ετικέτας του  $\beta$

Αν το  $\beta$  είναι "ΜΗ ΕΛΕΓΜΕΝΟ"

Βάλε το  $\beta$  στη  $\Sigma$

Τέλος



Τέλος

Αν υπάρχουν μη ακυρωμένες θέσεις για την ετικέτας του α

Βρες την καλύτερη από τις θέσεις αυτές και τοποθέτησε την ετικέτας

Σημείωσε το α ως "ΤΟΠΟΘΕΤΗΜΕΝΟ"

Για κάθε "ΕΛΕΓΜΕΝΟ" αντικείμενο β του ΣΑ που καμία θέση της ετικέτας του δεν επικαλύπτεται με την ετικέτα του α

Βάλε το β στην Σ

Τέλος

Αλλιώς

Βάλε το α στη Σ1 (εκτός κι αν ήδη είναι μέσα)

Τέλος

Τέλος

Αδειασε τη Σ2

Για κάθε αντικείμενο α του Σ1

Αν το α είναι "ΜΗ ΤΟΠΟΘΕΤΗΜΕΝΟ"

Βάλε το α στη Σ2

Διάλεξε μια τυχαία θέση για την ετικέτα του α

Τέλος

Τέλος

Βάλε τη Σ2 στην στοίβα ΠΡΟΒΛΗΜΑΤΑ

Τέλος

/\* Κύριο μέρος αλγορίθμου\*/

Όσο η στοίβα ΠΡΟΒΛΗΜΑΤΑ έχει στοιχεία

Βγάλε από τη στοίβα ΠΡΟΒΛΗΜΑΤΑ το πρώτο στοιχείο Π

N = πλήθος στοιχείων του Π

T = 0.91

ΠΛΗΘΟΣ\_ΘΕΡΜΟΚΡΑΣΙΩΝ = 1

ΕΓΙΝΕ\_ΑΛΛΑΓΗ = Αληθές

Όσο ( ΠΛΗΘΟΣ\_ΘΕΡΜΟΚΡΑΣΙΩΝ <= 50 ) ΚΑΙ ΕΓΙΝΕ\_ΑΛΛΑΓΗ

AP\_ΒΗΜΑΤΟΣ = 0

AP\_ΕΠΑΝΑΤΟΠΟΘΕΤΗΣΕΩΝ = 0

Όσο ( AP\_ΒΗΜΑΤΟΣ <= 20\*N ) ΚΑΙ ( AP\_ΕΠΑΝΑΤΟΠΟΘΕΤΗΣΕΩΝ <= 5\*N )

AP\_ΒΗΜΑΤΟΣ = AP\_ΒΗΜΑΤΟΣ + 1

Διάλεξε ένα αντικείμενο α από το Π

Διάλεξε μια νέα θέση για την ετικέτα του

Υπολόγισε τη μεταβολή ΔΕ στην αντικειμενική συνάρτηση

Αν προκύπτει χειροτέρευση στην αντικειμενική συνάρτηση

Ακύρωσε τη νέα τοποθέτηση με πιθανότητα  
 $P = 1 - e^{-\Delta E/T}$

Τέλος

Αν η νέα τοποθέτηση έχει γίνει δεκτή

AP\_ΕΠΑΝΑΤΟΠΟΘΕΤΗΣΕΩΝ = AP\_ΕΠΑΝΑΤΟΠΟΘΕΤΗΣΕΩΝ + 1

Τέλος

Τέλος

ΕΓΙΝΕ\_ΑΛΛΑΓΗ = ( AP\_ΕΠΑΝΑΤΟΠΟΘΕΤΗΣΕΩΝ > 0 )

Αν ΕΓΙΝΕ\_ΑΛΛΑΓΗ

ΠΛΗΘΟΣ\_ΘΕΡΜΟΚΡΑΣΙΩΝ = ΠΛΗΘΟΣ\_ΘΕΡΜΟΚΡΑΣΙΩΝ + 1

T = T\*0.9

Τέλος

Τέλος

Τέλος

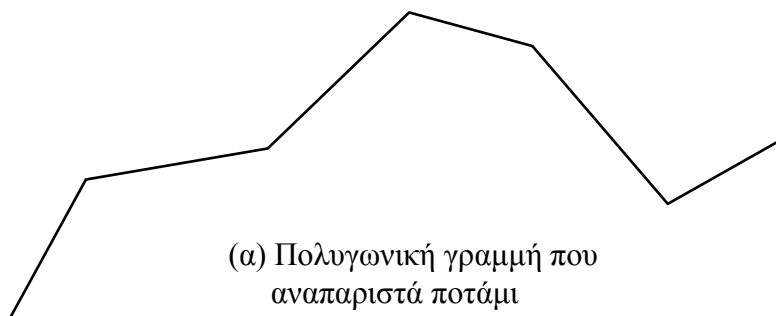
Όπως έχουμε ήδη αναφέρει, κατά τη διάρκεια της αρχικοποίησης τοποθετούνται οι ετικέτες που δεν επηρεάζουν την τοποθέτηση όλων των υπόλοιπων ετικετών. Επιπλέον, στη στοίβα Σ2 δημιουργούνται οι ομάδες των ετικετών, τα μέλη των οποίων δεν έχουν σχέση με τις υπόλοιπες ετικέτες του χάρτη. Στη συνέχεια οι ομάδες αυτές τοποθετούνται στη στοίβα ΠΡΟΒΛΗΜΑΤΑ και κατόπιν τροφοδοτούνται στο κύριο μέρος του αλγόριθμου που επιλύει τα επιμέρους ΠΒΤΕ.

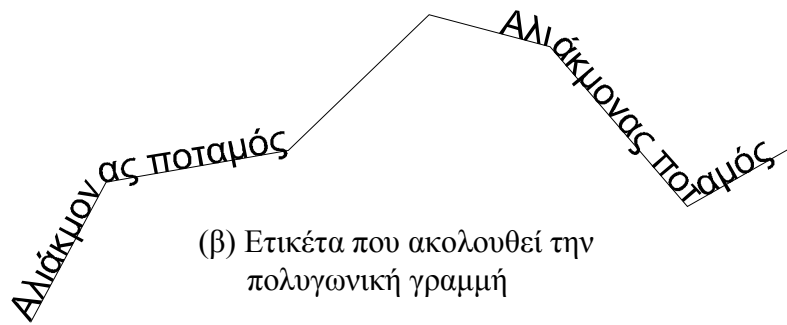
## 4.5 Ετικέτες Γραμμικών Αντικειμένων

Στην ενότητα αυτή θα εξετάσουμε τον τρόπο απεικόνισης των ετικετών που σχετίζονται με τα γραμμικά αντικείμενα του Χάρτη. Τέτοιου είδους αντικείμενα είναι οι δρόμοι σε έναν πολεοδομικό χάρτη, οι ποταμοί σε ένα γεωφυσικό χάρτη κ.ά.

Όπως έχουμε ήδη αναφέρει η απεικόνιση των συγκεκριμένων ετικετών παρουσιάζει σημαντικές διαφορές από τις απεικονίσεις των υπόλοιπων ετικετών. Οι σημειακές ετικέτες τοποθετούνται γύρω από το σημείο στο οποίο ανήκουν, όμως κάποιο παρόμοιο σημείο δεν υπάρχει στις γραμμές.

Ένας τρόπος τοποθέτησης των γραμμικών ετικετών είναι αυτός που ακολουθεί την πολυγωνική γραμμή που ορίζει το γραμμικό αντικείμενο. Στο Σχήμα 4.2(α) αναπαριστάται ένα γραμμικό αντικείμενο και στο Σχήμα 4.2(β) η ετικέτα τοποθετημένη σύμφωνα με τον τρόπο που προαναφέραμε. Παρατηρούμε ότι ο συγκεκριμένος τρόπος παρουσίασης της ετικέτας έχει ένα σοβαρό μειονέκτημα: οι ανωμαλίες που παρουσιάζονται σε συγκεκριμένα σημεία της γραμμής λόγω των γωνιών που σχηματίζουν τα ευθύγραμμα τμήματα που αποτελούν τη Διαδρομή στα σημεία αυτά δημιουργούν ένα άσχημο αισθητικά αποτέλεσμα όσον αφορά την ομαλότητα της ετικέτας. Συνεπώς παρουσιάζεται η ανάγκη να δημιουργήσουμε μια δεύτερη πολυγωνική γραμμή από το γραμμικό αντικείμενο, η οποία θα παρουσιάζει μεγαλύτερη καμπυλότητα, καθιστώντας την έτσι καταλληλότερη για την τοποθέτηση της ετικέτας πάνω σε αυτή.

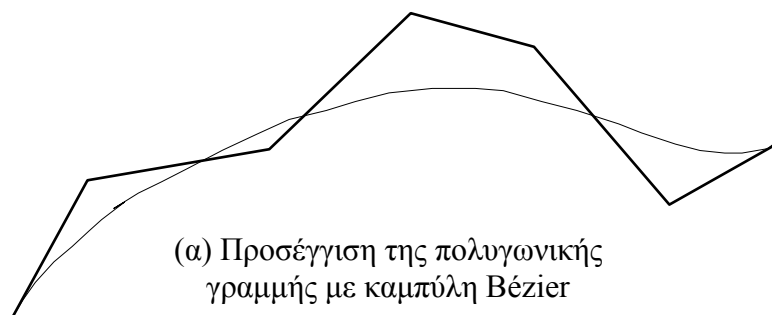


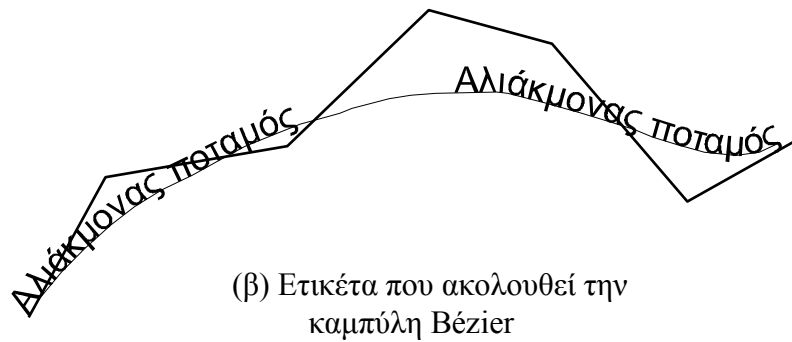


Σχήμα 4.2 - Παράδειγμα αναπαράστασης γραμμικής ετικέτας

#### 4.5.1 Καμπύλες Bézier

Οι καμπύλες που προαναφέραμε και οι οποίες έχουν την ιδιότητα της εξομάλυνσης που επιθυμούμε είναι οι **καμπύλες Bézier**. Στο Σχήμα 4.3(α) που ακολουθεί, παρουσιάζεται η καμπύλη Bézier που σχηματίζεται από την πολυγωνική γραμμή του Σχήματος 4.2(α) που περιγράφει το ποτάμι του παραδείγματος. Είναι προφανής η εξομάλυνση που επιτυγχάνεται συγκρινόμενη με τις ανωμαλίες που παρουσίαζε η γραμμή από την οποία προήρθε. Στο Σχήμα 4.3(β) δίνεται η ίδια ετικέτα τοποθετημένη πάνω στην καμπύλη Bézier που κατασκευάστηκε. Είναι φανερό ότι η νέα τοποθέτηση της ετικέτας παρουσιάζει αρκετά πιο βελτιωμένο αισθητικά αποτέλεσμα από αυτή του σχήματος 4.2(β).





Σχήμα 4.3 - Τοποθέτηση γραμμικής ετικέτας με τη χρήση καμπύλης Bézier

Συνεπώς, είναι φανερό, πλέον, ότι οι καμπύλες Bézier λόγω της ιδιότητας τους να εξομαλύνουν τις πολυγωνικές γραμμές από τις οποίες ορίζονται, καθίστανται ιδανικές για οδηγούς στην τοποθέτηση γραμμικών ετικετών.

Στη συνέχεια θα προχωρήσουμε στην αναλυτική περιγραφή των καμπυλών Bézier με βάση τα πολυώνυμα Bernstein. Με βάση τη μελέτη αυτή, προχωρήσαμε στην υλοποίηση ενός μοντέλου τοποθέτησης των γραμμικών ετικετών, καθώς και του αλγόριθμου αυτόματης τοποθέτησης των συγκεκριμένων ετικετών, ο οποίος ενσωματώθηκε στο γενικότερο πλαίσιο του ΠΒΤΕ.

#### 4.5.2 Πολυώνυμα Bernstein

Τα πολυώνυμα Bernstein τάξεως  $n$  ορίζονται ως εξής:

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

για  $i = 0, 1, \dots, n$ , όπου:

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}$$

Ορίζονται, συνεπώς  $n+1$  πολυώνυμα Bernstein τάξεως  $n$ . Για ευκολία θέτουμε επιπλέον  $B_{i,n}(t) = 0 \forall t \in [0,1]$ , αν  $i < 0$  ή  $i > n$ .

Τα πολυώνυμα Bernstein τάξεως  $n$  μπορούν να οριστούν αναδρομικά από δύο πολυώνυμα Bernstein τάξεως  $n - 1$  ως εξής:

$$B_{k,n}(t) = (1 - t)B_{k,n-1}(t) + tB_{k-1,n-1}(t)$$

Η απόδειξη της εξίσωσης αυτής δίνεται στη συνέχεια:

### 4.5.3 Αναλυτική Έκφραση Καμπύλης Bézier

Αν μας δοθεί ένα σύνολο  $n + 1$  σημείων  $\{P_0, P_1, \dots, P_n\}$ , μπορούμε να ορίσουμε μια καμπύλη Bézier τάξεως  $n$  σύμφωνα με την επόμενη παραμετρική εξίσωση:

$$P(t) = \sum_{i=0}^n P_i B_{i,n}(t), t \in [0,1]$$

όπου  $B_{i,n}(t)$  είναι τα πολυώνυμα Bernstein τάξεως  $n$  που περιγράψαμε στην προηγούμενη ενότητα.

Αν  $P_i = \langle x_i, y_i \rangle, i = 0, 1, 2, \dots, n$ , όπου  $x_i, y_i$  οι συντεταγμένες του σημείου  $P_i$ , τότε η προηγούμενη εξίσωση μπορεί να γραφτεί ως ένα ζεύγος εξισώσεων που δίνουν την τεταγμένη και την τετμημένη κάθε σημείου της καμπύλης ως εξής:

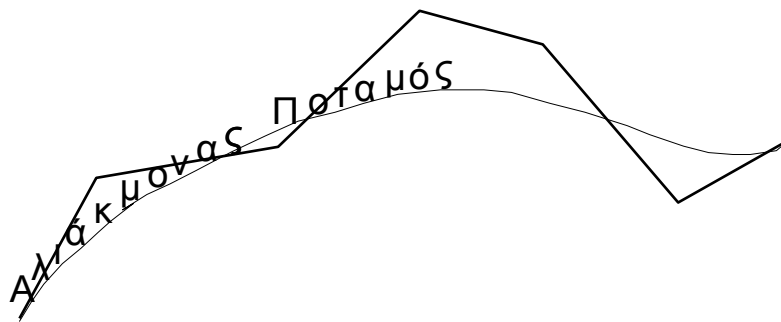
$$x(t) = \sum_{i=0}^n x_i B_{i,n}(t), t \in [0,1]$$

$$y(t) = \sum_{i=0}^n y_i B_{i,n}(t), t \in [0,1]$$

Στην υλοποίηση των καμπυλών Bézier στην εφαρμογή μας θέσαμε ότι ο αριθμός των σημείων που θα ορίζουν κάθε καμπύλη είναι 64, μέγεθος που κρίνεται ικανοποιητικό για κάθε μήκους γραμμικό αντικείμενο και μέγεθος ετικέτας.

#### 4.5.4 Κλίση των Χαρακτήρων των Ετικετών

Μια επιπλέον σημαντική ιδιότητα των καμπυλών Βέζιερ που μπορούμε να εκμεταλλευτούμε για να προσδώσουμε στις γραμμικές ετικέτες καλύτερο οπτικό αποτέλεσμα είναι η πρώτη παράγωγός τους. Με τη χρήση της πρώτης παραγώγου μπορούμε να δώσουμε κλίση σε κάθε χαρακτήρα της ετικέτας που τοποθετείται πάνω στην καμπύλη, παίρνοντας την τοποθέτηση της ετικέτας όπως φαίνεται στο Σχήμα 4.3(β). Αν δεν χρησιμοποιούσαμε το παραπάνω χαρακτηριστικό των καμπυλών Βέζιερ το αποτέλεσμα που θα είχαμε θα ήταν αυτό που φαίνεται στο Σχήμα 4.4, το οποίο είναι φανερά κατώτερο σε σύγκριση με αυτό του Σχήματος 4.3(β).



Σχήμα 4.4 - Τοποθέτηση γραμμικής ετικέτας χωρίς τη χρήση των πρώτων παραγώγων των καμπυλών Βέζιερ

Παραγωγίζοντας τις εξισώσεις που δίνουν τις συντεταγμένες κάθε σημείου  $P_i$  της καμπύλης Βέζιερ, δημιουργούνται οι τύποι που δίνουν την πρώτη παράγωγο της καμπύλης στο συγκεκριμένο σημείο:

$$a_x(t) = \frac{d}{dt} x(t) = n \cdot \sum_{i=0}^{n-1} ((x_{i+1} - x_i) \cdot B_{i,n-1}(t)), t \in [0,1]$$
$$a_y(t) = \frac{d}{dt} y(t) = n \cdot \sum_{i=0}^{n-1} ((y_{i+1} - y_i) \cdot B_{i,n-1}(t)), t \in [0,1]$$

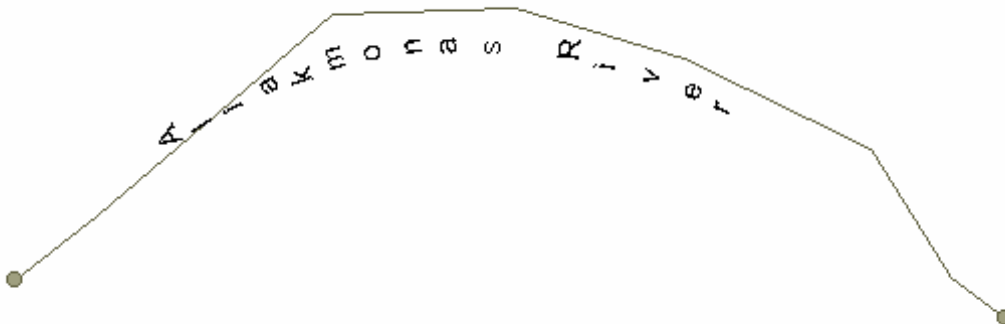
Με βάση τους παραπάνω τύπους μπορούμε να υπολογίσουμε την κλίση της καμπύλης Βέζιερ σε κάθε σημείο της και συνεπώς την κλίση κάθε γράμματος της ετικέτας που τοποθετείται πάνω στην καμπύλη. Η κλίση εκφρασμένη ως γωνία της



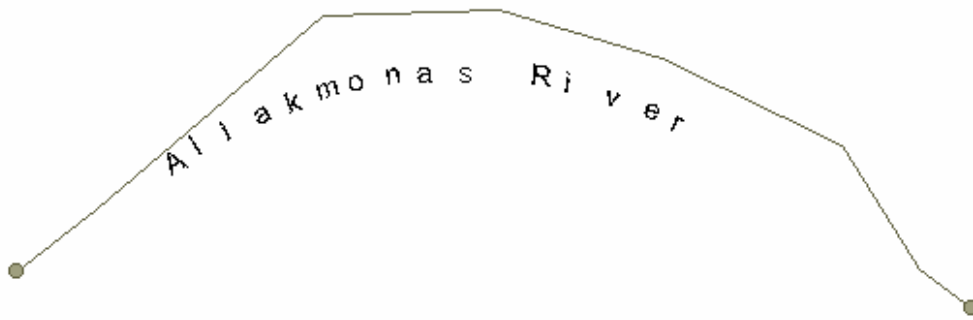
εφαπτομένης με μια οριζόντια ευθεία που διέρχεται από το σημείο επαφής δίνεται από τον τύπο:

$$a = \begin{cases} \frac{\pi}{2}, a_y = 0 \\ \arctan\left(\frac{a_x}{a_y}\right), a_y \neq 0 \end{cases}$$

Όμως η συγκεκριμένη γωνία μπορεί να δεχθεί τιμές από  $0$  έως  $2\pi$ , κάτι που μπορεί να επιφέρει το μη αποδεκτό αποτέλεσμα τοποθέτησης της ετικέτας που παρουσιάζεται στο Σχήμα 4.5(α). Όπως φαίνεται, χαρακτηριστικά, η κλίση της καμπύλης Bézier σε ορισμένα σημεία της είναι τέτοια ώστε τα γράμματα της ετικέτας έχουν τοποθετηθεί με ανάποδη φορά.



(α)



(β)

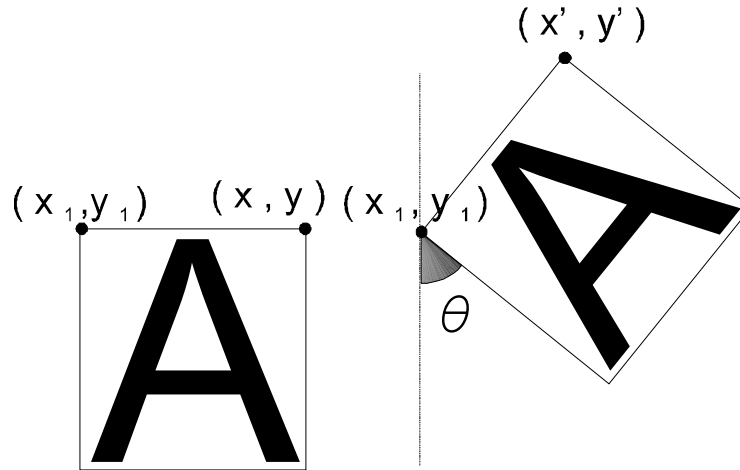
Σχήμα 4.5 - Τοποθέτηση γραμμικής ετικέτας με ανάποδη και σωστή κλίση

Συνεπώς, ο προηγούμενος τύπος που δίνει την κλίση στα διάφορα σημεία της καμπύλης είναι αναγκαίο να τροποποιηθεί με τέτοιο τρόπο ώστε να αποκλείεται ένα εύρος τιμών που θα δημιουργήσει το αποτέλεσμα του Σχήματος 4.5(α).

Το παραπάνω πρόβλημα λύνεται συμπληρώνοντας τον προηγούμενο τύπο με τις εξής συνθήκες που αποκόπτουν ορισμένες τιμές που επιφέρουν το συγκεκριμένο άσχημο οπτικά αποτέλεσμα:

$$a = a - \frac{\pi}{2}, 0 \leq a \leq \pi$$

Επιπλέον, κρίθηκε αναγκαία η εύρεση μιας μεθόδου η οποία θα δίνει με ακρίβεια το περιγεγραμμένο ορθογώνιο των χαρακτήρων που απαρτίζουν τις γραμμικές ετικέτες. Το προαναφερθέν ορθογώνιο είναι απαραίτητο για την υλοποίηση του αλγορίθμου που επιλύει το ΠΒΤΕ για τις γραμμικές ετικέτες συγκεκριμένα.



Σχήμα 4.6 – Παράδειγμα περιστροφής χαρακτήρα κατά γωνία  $\theta$

Στο Σχήμα 4.6 δίνεται ως παράδειγμα ένας χαρακτήρας ο οποίος στη συνέχεια περιστρέφεται κατά γωνία  $\theta$ . Η περιστροφή έχει ως κέντρο της την πάνω αριστερά γωνία του περιγεγραμμένου ορθογώνιου του χαρακτήρα με συντεταγμένες έστω  $(x_1, y_1)$ . Αν  $(x, y)$  είναι οι συντεταγμένες της πάνω δεξιά γωνίας του ορθογώνιου πριν την περιστροφή, οι νέες της συντεταγμένες  $(x', y')$  μετά την περιστροφή δίνονται από τον παρακάτω τύπο:

$$[x' \quad y' \quad 1] = [x \quad y \quad 1] \cdot \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ x_1(1 - \cos \theta) + y_1 \sin \theta & y_1(1 - \cos \theta) - x_1 \sin \theta & 1 \end{bmatrix}$$

Επιλύοντας τον παραπάνω τύπο ως προς  $x'$  και  $y'$  βρίσκουμε τελικά την τετμημένη και την τεταγμένη του σημείου που περιστράφηκε:

$$\begin{aligned} x' &= x \cos \theta - y \sin \theta + x_1(1 - \cos \theta) + y_1 \sin \theta \\ y' &= x \sin \theta + y \cos \theta + y_1(1 - \cos \theta) - x_1 \sin \theta \end{aligned}$$

Ο ίδιος τύπος ισχύει για όλα τα σημεία του ορθογώνιου, συνεπώς μπορούμε να καθορίσουμε εύκολα τις νέες συντεταγμένες των κορυφών του περιγεγραμμένου ορθογώνιου μετά την περιστροφή του.

Επιπλέον, το Μεγενθυμένο Ορθογώνιο για τις γραμμικές ετικέτες ορίζεται ως το ελάχιστο ορθογώνιο που περικλείει όλα τα περιγεγραμμένα ορθογώνια των χαρακτήρων που απαρτίζουν την ετικέτα.

#### 4.5.5 Επίλυση συγκρούσεων Ετικετών Γραμμικών Αντικειμένων

Οι ετικέτες γραμμικών αντικειμένων, παρουσιάζουν τεράστιες διαφορές σε σύγκριση με τις σημειακές ετικέτες. Ως γνωστόν, οι σημειακές ετικέτες μπορούν να τοποθετηθούν γύρω από ένα σταθερό σημείο με τον τρόπο που περιγράφεται στο Κεφάλαιο 3. Αντίθετα, οι γραμμικές ετικέτες, τοποθετούνται πάνω στην καμπύλη Bézier που κατασκευάζεται βασιζόμενη στο γραμμικό αντικείμενο στο οποίο ανήκει.

Συνεπώς, ο αλγόριθμος επίλυσης του ΠΒΤΕ για τις ετικέτες σημειακών αντικειμένων που αναλύσαμε στις προηγούμενες παραγράφους δεν είναι δυνατόν να χρησιμοποιηθεί για την τοποθέτηση των γραμμικών ετικετών. Οπότε, είναι αναγκαίο να χρησιμοποιήσουμε έναν εντελώς διαφορετικό αλγόριθμο για την επίλυση του προβλήματος όσον αφορά τις συγκεκριμένου είδους ετικέτες.

Ο αλγόριθμος που τελικά υλοποιήσαμε για την επίλυση του ΠΒΤΕ για τις γραμμικές ετικέτες είναι ο εξής:

Αρχικά ο αλγόριθμος ελέγχει για πιθανές συγκρούσεις μεταξύ της γραμμικής ετικέτας με τις υπόλοιπες ετικέτες εξετάζοντας τα μεγεθυμένα ορθογώνια, οι οποίες τοποθετούνται σε μια στοίβα A. Στη συνέχεια, εξετάζει αν οι πιθανές συγκρούσεις υπάρχουν πραγματικά ελέγχοντας αν τα περιγεγραμμένα ορθογώνια των χαρακτήρων που απαρτίζουν τη γραμμική ετικέτα επικαλύπτονται με τα περιγεγραμμένα ορθογώνια των ετικετών της στοίβας A, οπότε και ακυρώνει τα σημεία τοποθέτησης της καμπύλης Bézier στα οποία έχουν τοποθετηθεί οι χαρακτήρες της γραμμικής ετικέτας που παρουσιάζουν επικάλυψη.

Έπειτα, είτε έχει παρουσιαστεί επικάλυψη στο προηγούμενο βήμα είτε όχι, ακυρώνει όλες τις υπόλοιπες θέσεις τοποθέτησης της καμπύλης Bézier που παρουσιάζουν επικάλυψη με οποιαδήποτε ετικέτα.

Στο τελευταίο στάδιο ο αλγόριθμος προσπαθεί να βρει τη βέλτιστη θέση τοποθέτησης στην καμπύλη Bézier για τη γραμμική ετικέτα ελέγχοντας όλες τις σειρές διαδοχικών, μη ακυρομένων θέσεων τοποθέτησης στην καμπύλη Bézier που έχουν δημιουργηθεί στα προηγούμενα βήματα.

Στη συνέχεια παρουσιάζουμε το γενικό περίγραμμα του αλγόριθμου επίλυσης του ΠΒΤΕ για τις ετικέτες γραμμικών αντικειμένων:

Για κάθε γραμμική ετικέτα

Διάλεξε μια γραμμική ετικέτα, έστω  $\delta$

Έλεγχξε με ποια μεγεθυμένα ορθογώνια άλλων ετικετών επικαλύπτεται το μεγενθυμένο ορθογώνιο της ετικέτας  $\delta$

Τοποθέτησε τις ετικέτες αυτές στη στοίβα A

Τέλος

Όσο η στοίβα A έχει στοιχεία

Βγάλε από την A μια ετικέτα, έστω  $\alpha$

Ακύρωσε τις θέσεις τοποθέτησης πάνω στην καμπύλη Bézier στις οποίες παρουσιάζονται επικαλύψεις, ελέγχοντας για επικάλυψη των περιγεγραμμένων ορθογωνίων των χαρακτήρων που αποτελούν τη  $\delta$  με ορθογώνιο που περικλείει την  $\alpha$

Τέλος

Τέλος

Ακύρωσε όσα από τα υπόλοιπα σημεία της καμπύλης Bézier επικαλύπτονται με κάποια ετικέτα

Ψάξε όλες τις σειρές συνεχόμενων, μη ακυρομένων θέσεων τοποθέτησης στην καμπύλη Bézier

Αν βρεις σειρά που μπορεί να τοποθετηθεί η  $\delta$  ολόκληρη

Τοποθέτησε τη  $\delta$  στη συγκεκριμένη σειρά

Τέλος

Αλλιώς

Τοποθέτησε τη δ στη μεγαλύτερη δυνατή σειρά

Τέλος

Τέλος

Τέλος

Στο Σχήμα 4.7 που ακολουθεί παρουσιάζεται ένα παράδειγμα εφαρμογής του αλγορίθμου που μόλις περιγράψαμε.



Σχήμα 4.7 – Παράδειγμα εφαρμογής του αλγορίθμου επίλυσης του ΠΒΤΕ για τις γραμμικές ετικέτες

## 4.6 Περίληψη

Στο Κεφάλαιο αυτό δώσαμε στον αναγνώστη μία πλήρη ανάλυση του ΠΒΤΕ και αναλύσαμε τις διάφορες πτυχές που το αποτελούν. Στη συνέχεια, προχωρήσαμε στην διατύπωση της λύσης που προτείναμε και υλοποιήσαμε στα πλαίσια της διπλωματικής μας εργασίας για το συγκεκριμένο πρόβλημα με τη χρήση της μεθόδου Προσομοίωσης Ανόπτησης για τις σημειακές ετικέτες.

Έπειτα, αφού παρουσιάσαμε τα πλεονεκτήματα της χρήσης των καμπυλών Bézier στην απεικόνιση των ετικετών γραμμικών αντικειμένων, αναλύσαμε το πλήρες μαθηματικό μοντέλο που τελικά χρησιμοποιείται στην παρουσίαση των συγκεκριμένων ετικετών.

Τέλος, αφού εξηγήσαμε τις σημαντικές διαφορές που παρουσιάζουν στους τρόπους απεικόνισής τους οι γραμμικές ετικέτες σε σύγκριση με τις σημειακές και πώς αυτές εμποδίζουν τη χρησιμοποίηση κοινού αλγορίθμου επίλυσης του ΠΒΤΕ και για τα δύο είδη ετικετών, περιγράψαμε τον αλγόριθμο επίλυσης του προβλήματος που χρησιμοποιείται για τις γραμμικές ετικέτες ειδικά.

## **Κεφάλαιο 5**

# **Μοντέλο Αντικειμένων Ετικετών**

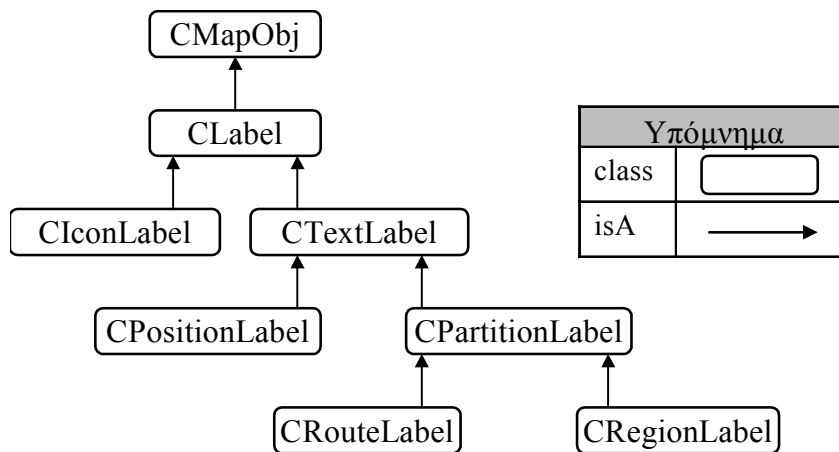
### **5.1 Γενικά**

Στο Κεφάλαιο αυτό θα αναφερθούμε στο Μοντέλο των Αντικειμένων Ετικετών και πιο συγκεκριμένα στη χρησιμότητά τους, στη σύνδεση του Μοντέλου στην υπάρχουσα ιεραρχία κλάσεων της εφαρμογής Map Studio καθώς και στις προσθήκες που χρειάστηκε να γίνουν έτσι ώστε οι δομές της εφαρμογής να υποστηρίζουν το νέο μοντέλο. Τέλος, θα αναφερθούμε στο μηχανισμό αλληλεπίδρασης εφαρμογής-χρήστη που κατασκευάστηκε για τη διαχείριση των ετικετών.

### **5.2 Μοντέλο Ετικετών**

Η ιεραρχία ΕΙΝΑΙ των κλάσεων που υλοποιούν τις ετικέτες στην εφαρμογή Map Studio παρουσιάζεται στο Σχήμα 5.1.





Σχήμα 5.1 - Η EINAI ιεραρχία των κλάσεων ετικετών

Η κλάση CMapObj αποτελεί τη βασική κλάση των γεωγραφικών και μη αντικειμένων της εφαρμογής, όπως έχουμε ήδη αναφέρει στο προηγούμενο Κεφάλαιο 3.

### 5.2.1 Κλάση Ετικέτας (CLabel)

Η κλάση CLabel (EINAI CMapObj) είναι η βάση της ιεραρχίας των Ετικετών. Σκοπός της συγκεκριμένης κλάσης είναι να ορίσει τα χαρακτηριστικά και τα κοινά γνωρίσματα που διέπουν όλα τα είδη των ετικετών.

Ο Πίνακας 5.1 που ακολουθεί, περιγράφει τα πεδία που ορίζονται στην κλάση CLabel.

Πεδίο	Σημασία
<b><i>m_idAttachedTo</i></b>	Επιστρέφει ένα δείκτη στο αντικείμενο με το οποίο συνδέεται η ετικέτα.
<b><i>m_byChecked</i></b>	Σημαία που δείχνει αν η ετικέτα έχει «ελεγχθεί».
<b><i>m_byPlaced</i></b>	Σημαία που δείχνει αν η ετικέτα έχει «τοποθετηθεί».
<b><i>m_byVisible</i></b>	Σημαία που δείχνει αν η ετικέτα είναι ορατή.
<b><i>m_rcAugmented</i></b>	Το ορθογώνιο μέσα στο οποίο μπορεί να τοποθετηθεί η ετικέτα.
<b><i>m_listFreeRects</i></b>	Η λίστα που περιέχει τα ελεύθερα ορθογώνια τοποθέτησης της ετικέτας.

Πίνακας 5.1 - Τα πεδία της κλάσης CLabel και η σημασία τους

Όπως φαίνεται και από την EINAI ιεραρχία, οι ετικέτες είναι Αντικείμενα Χάρτη και όχι Επιπέδου, όπως είναι τα αντικείμενα (θέσεις, διαδρομές, περιοχές) με τα οποία αντιστοιχίζονται. Αυτό σημαίνει ότι οι ετικέτες τοποθετούνται συνολικά στο

χάρτη και όχι σε κάθε επίπεδο του χωριστά. Αυτό εξάλλου επιθυμούμε, αφού αυτό που ενδιαφέρει το χρήστη είναι το συνολικό αποτέλεσμα της τοποθέτησης των ετικετών πάνω στο χάρτη.

Το πεδίο **m\_idAttachedTo** είναι η σύνδεση της ετικέτας με το Αντικείμενο Χρήστη με το οποίο αντιστοιχίζεται. Το πεδίο αυτό είναι αντικείμενο κλάσης **COID**, η οποία ανασύρει το συγκεκριμένο αντικείμενο από το Hash Table του επιπέδου όπου ανήκει χρησιμοποιώντας το αναγνωριστικό του. Μέσω αυτού του πεδίου έχουμε πρόσβαση στα χαρακτηριστικά και τις πληροφορίες που είναι απαραίτητες για τη σωστή τοποθέτηση και σχεδιασμό της ετικέτας. Τα πεδία και οι ρουτίνες της COID περιγράφονται στους επόμενους πίνακες.

Πεδίο	Σημασία
<b>m_oid</b>	Το αναγνωριστικό του αντικειμένου με το οποίο συνδέεται η ετικέτα.
<b>m_pLayer</b>	Δείκτης στο Επίπεδο στο οποίο ανήκει το παραπάνω αντικείμενο.

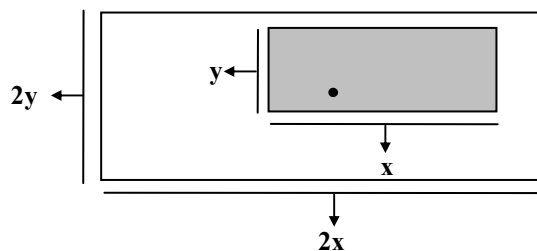
Πίνακας 5.2 - Τα πεδία της κλάσης COID

Ρουτίνα(-ες)	Λειτουργία
<b>FromLayerObj(pObj)</b>	Αποθηκεύει το αναγνωριστικό και το Επίπεδο του αντικειμένου στα πεδία της κλάσης.
<b>operator CLayerObj* ()</b>	Επιστρέφει ένα δείκτη στο αντικείμενο με το οποίο συνδέεται η ετικέτα.

Πίνακας 5.3 - Ρουτίνες μέλη της κλάσης COID

Τα πεδία **m\_byChecked**, **m\_byPlaced** και **m\_listFreeRects** χρησιμοποιούνται από τον αλγόριθμο επίλυσης του ΠΒΤΕ. Τα πεδία **m\_byChecked** και **m\_byPlaced** καθορίζουν αν μία ετικέτα είναι «ελεγμένη» ή αντίστοιχα «τοποθετημένη» στον αλγόριθμο Προσομοίωσης Ανόπτησης που περιγράφεται στην υποενότητα 4.4.4. Το πεδίο **m\_listFreeRects** είναι μία λίστα, στην οποία έχουν αποθηκευθεί τα ορθογώνια που καθορίζουν τον ελεύθερο χώρο τοποθέτησης της ετικέτας σύμφωνα με τον αλγόριθμο της υποενότητας 3.4.3.

Το πεδίο **m\_rcAugmented** περιέχει το ορθογώνιο μέσα στο οποίο μπορεί να τοποθετηθεί η ετικέτα, δηλαδή καθορίζει όλες τις επιτρεπτές θέσεις για την τοποθέτηση της ετικέτας. Το ορθογώνιο αυτό ονομάζεται **Μεγεθυμμένο Ορθογώνιο** (Augmented Rectangle). Για τις ετικέτες τύπου **CIconLabel** και **CPositionLabel** το ορθογώνιο αυτό έχει διαστάσεις (2x,2y), όπου (x,y) οι διαστάσεις του ορθογωνίου που περικλείει την ετικέτα, όπως φαίνεται και στο Σχήμα 5.2.



Σχήμα 5.2 - Παράδειγμα ετικέτας και μεγενθυμένου ορθογωνίου

Στο Πίνακα που ακολουθεί αναφέρονται οι ρουτίνες που περιέχονται στην κλάση CLabel και δίνεται μια περιληπτική περιγραφή των λειτουργιών που υλοποιούν.

<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
<b>Attach(pObj)</b>	Συνδέει την ετικέτα με το Αντικείμενο Χρήστη.
<b>GetAttachedObj()</b>	Επιστρέφει ένα δείκτη στο Αντικείμενο Χρήστη, με το οποίο συνδέεται η ετικέτα.
<b>CalcAugmentedRect()</b>	Υπολογίζει τις διαστάσεις του Μεγενθυμένου Ορθογωνίου.
<b>SetAugmentedRect(rect)</b>	Θέτει το Μεγενθυμένο Ορθογώνιο ίσο με το ορθογώνιο rect.
<b>GetAugmentedRect()</b>	Επιστρέφει το Μεγενθυμένο Ορθογώνιο.
<b>SetLabelRect(rect)</b>	Θέτει το ορθογώνιο που περικλείει την ετικέτα ίσο με rect.
<b>GetLabelRect()</b>	Επιστρέφει το ορθογώνιο που περικλείει την ετικέτα.
<b>PlaceLabel()</b>	Ελέγχει αν η ετικέτα μπορεί να τοποθετηθεί σε κάποιο από τα Ελεύθερα Ορθογώνια Τοποθέτησης και αν κάτι τέτοιο ισχύει, την τοποθετεί τυχαία μέσα σε αυτό.
<b>PlaceLabelRandom()</b>	Τοποθετεί την ετικέτα τυχαία μέσα στα όρια του Μεγενθυμένου Ορθογωνίου.

Πίνακας 5.4 - Οι ρουτίνες της κλάσης CLabel

Είναι απαραίτητο να αναφέρουμε ότι το ορθογώνιο που περικλείει την ετικέτα αποθηκεύεται στο πεδίο **m\_rcBound** της κλάσης CBoundedObj.

## 5.2.2 Ετικέτα Εικονίδιο (CIconLabel)

Τα στιγμιότυπα της κλάσης **CIconLabel** υλοποιούν τις Ετικέτες Εικονίδια. Στην κλάση αυτή ορίζεται το πεδίο **m\_hIcon** που είναι ο διαχειριστής (handle<sup>7</sup>) του εικονιδίου που απεικονίζει η ετικέτα. Αν η ετικέτα έχει αντιστοιχηθεί με Διαδρομή, τότε το πεδίο **m\_iPoint** προσδιορίζει το «εσωτερικό σημείο» (inner point) της

<sup>7</sup>Ο διαχειριστής είναι ένας αριθμός (συνήθως μεγέθους 32 bit), ο οποίος παραπέμπει σε ένα αντικείμενο. Οι διαχειριστές στα Windows είναι παρόμοιοι με τους διαχειριστές στη C. Ένα πρόγραμμα δέχεται ένα διαχειριστή όταν καλεί μία συνάρτηση των Windows. Έπειτα, το πρόγραμμα χρησιμοποιεί το διαχειριστή σε άλλες συναρτήσεις των Windows για να προσπελάσει το συγκεκριμένο αντικείμενο.

Διαδρομής γύρω από το οποίο τοποθετείται η ετικέτα. Αν η ετικέτα έχει αντιστοιχηθεί με κάποιου άλλου είδους αντικείμενο, τότε η τιμή του `m_iPoint` μένει απροσδιόριστη.

Ο Πίνακας 5.5 που ακολουθεί, περιέχει τις ρουτίνες της συγκεκριμένης κλάσης.

<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
<b><i>GetIconHandle()</i></b>	Επιστρέφει τον διαχειριστή του εικονιδίου.
<b><i>SetIconHandle(hIcon)</i></b>	Θέτει το διαχειριστή του εικονιδίου.
<b><i>SetAnchorPoint(point)</i></b>	Θέτει το σημείο γύρω από το οποίο τοποθετείται η ετικέτα, το οποίο είναι «εσωτερικό σημείο» της Διαδρομής.

*Πίνακας 5.5 - Οι ρουτίνες της κλάσης CIconLabel*

Οι ετικέτες εικονίδιο μπορεί να ανήκουν είτε σε μία θέση είτε σε μία διαδρομή. Στο Σχήμα 5.3 δίνονται παραδείγματα τοποθέτησης ετικετών εικονίδιο τόσο σε θέση όσο και σε διαδρομή. Αν η ετικέτα εικονίδιο αντιστοιχίζεται σε θέση, τότε η τοποθέτησή της γύρω από τη θέση γίνεται όπως φαίνεται στο Σχήμα 5.3(α). Αν, όμως, η ετικέτα εικονίδιο αντιστοιχίζεται με διαδρομή, τότε η ετικέτα μπορεί να τοποθετηθεί σε οποιοδήποτε από τα εσωτερικά σημεία, συμπεριλαμβανομένων της αρχής και του τέλους, της διαδρομής, όπως φαίνεται και στο Σχήμα 5.3(β).



(α)



(β)

Σχήμα 5.3 – Πιθανές τοποθετήσεις ετικετών εικονίδιο

Και στις δύο περιπτώσεις για να απεικονιστεί η ετικέτα εικονίδιο ανακτούμε τις συντεταγμένες του ορθογωνίου που περικλείει την ετικέτα με τη χρησιμοποίηση της ρουτίνας `GetLabelRect()` και στη συνέχεια προβάλλουμε το εικονίδιο μέσα στο ορθογώνιο αυτό.

### 5.2.3 Ετικέτες Κειμένου

Η κλάση `CTextLabel` είναι η βασική κλάση στην υλοποίηση των ετικετών που περιέχουν κείμενο ως πληροφορία. Το πεδίο που ορίζεται σε αυτή την κλάση είναι το `m_szCaption` και είναι το αλφαριθμητικό που εμφανίζεται στην ετικέτα. Επιπλέον, η ετικέτα κειμένου είναι δυνατόν να περιέχει πληροφορίες, όπως το αναγνωριστικό του αντικειμένου με το οποίο συνδέεται η ετικέτα ή, αν πρόκειται για ετικέτα θέσης, το αναγνωριστικό της πρώτης διαδρομής που έχει ως αφετηρία ή τερματισμό τη θέση αυτή, σύμφωνα με όσα περιγράψαμε στην ενότητα 2.4.

Οι ρουτίνες μέλη που ορίζει η κλάση `CTextLabel` φαίνονται στον Πίνακα που ακολουθεί.

<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
<b><code>GetLabelText()</code></b>	Επιστρέφει το αλφαριθμητικό που απαρτίζει την ετικέτα.
<b><code>SetLabelText(szCaption)</code></b>	Θέτει το αλφαριθμητικό της ετικέτας.
<b><code>EvaluateToken()</code></b>	Εξετάζει το αλφαριθμητικό της ετικέτας και προσθέτει τα αναγνωριστικά της Θέσης ή / και της Διαδρομής, αν ζητηθούν.

Πίνακας 5.6 - Οι ρουτίνες μέλη της κλάσης `CTextLabel`

Η κλάση `CPositionLabel` που είναι υποκλάση της `CTextLabel` υλοποιεί τις ετικέτες κειμένου που ανήκουν σε θέσεις. Στο Σχήμα 5.4 που ακολουθεί

παρουσιάζεται ένα παράδειγμα τοποθέτησης μίας ετικέτας κειμένου που αντιστοιχίζεται με μία θέση.

## Position Label

Σχήμα 5.4 – Ετικέτα κειμένου που ανήκει σε θέση

Ο τρόπος παρουσίασης του συγκεκριμένου τύπου ετικέτας γίνεται με τον ίδιο ακριβώς τρόπο όπως και στην ετικέτα εικονίδιο, δηλαδή τυπώνεται το αλφαριθμητικό της ετικέτας μέσα στο ορθογώνιο που την περικλείει χρησιμοποιώντας τη ρουτίνα `GetLabelRect()`.

Για τη σωστότερη απεικόνιση των ετικετών κειμένου που αντιστοιχίζονται σε διαδρομές ή περιοχές χρειάζεται να προχωρήσουμε στον κατακερματισμό των αλφαριθμητικών τους στα στοιχεία που τα απαρτίζουν. Για να φέρουμε σε πέρας αυτή τη διαδικασία ορίσαμε τη κλάση **CPartitionLabel**. Μοναδικό πεδίο που ορίζεται στη συγκεκριμένη κλάση είναι το **m\_aBrokenLabel** που περιέχει τα στοιχεία του αλφαριθμητικού.

Ο Πίνακας 5.7 περιέχει τις συναρτήσεις που ορίζονται στη κλάση `CPartitionLabel`.

<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
<b>BreakLabel()</b>	Κατακερματίζει το αλφαριθμητικό της ετικέτας στα στοιχεία της.
<b>GetLabelChars()</b>	Επιστρέφει τα στοιχεία του αλφαριθμητικού.
<b>GetCountofChars()</b>	Επιστρέφει τον αριθμό των στοιχείων.

Πίνακας 5.7 - Οι ρουτίνες μέλη της κλάσης `CPartitionLabel`

Η υποκλάση της `CPartitionLabel`, **CRouteLabel** υλοποιεί τις ετικέτες κειμένου που αντιστοιχίζονται σε διαδρομές. Για την παρουσίαση των ετικετών διαδρομής χρησιμοποιούμε τις καμπύλες Bézier. Με τη χρησιμοποίηση αυτού του είδους των καμπυλών επιτυγχάνουμε πιο αισθητικά αποδεκτά αποτελέσματα, καθώς είναι πιο ομαλές από τις διαδρομές (πολυγωνικές γραμμές) από τις οποίες δημιουργούνται. Το πεδίο **m\_iStartingPoint** που ορίζεται στη συγκεκριμένη κλάση δηλώνει το σημείο της καμπύλης Bézier από το οποίο ξεκινάει η ετικέτα, ενώ το

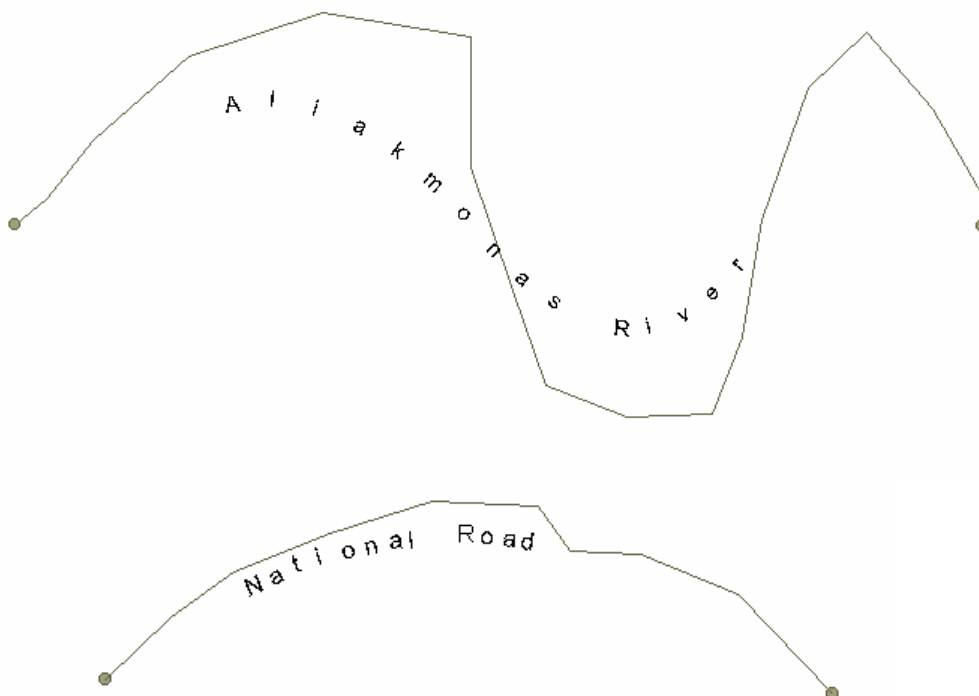
πεδίο **m\_iMaxStartingPoint** δηλώνει το τελευταίο σημείο της καμπύλης Bézier από το οποίο επιτρέπεται να ξεκινήσει η ετικέτα.

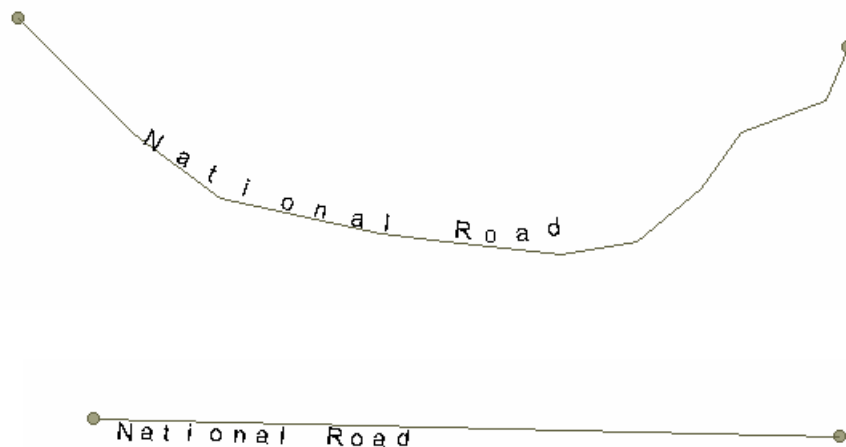
Στον Πίνακα που ακολουθεί παρουσιάζονται οι ρουτίνες μέλη της κλάσης CRouteLabel.

<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
<b>GetStartingPoint()</b>	Επιστρέφει το σημείο της καμπύλης Bézier από το οποίο ξεκινάει η ετικέτα.
<b>SetStartingPoint(point)</b>	Θέτει το σημείο της καμπύλης Bézier από το οποίο ξεκινάει η ετικέτα.
<b>CalcMaxStartingPoint()</b>	Υπολογίζει το τελευταίο σημείο της καμπύλης Bézier από το οποίο επιτρέπεται να ξεκινήσει η ετικέτα.
<b>RotateChar()</b>	Υπολογίζει τη γωνία περιστροφής ενός στοιχείου της ετικέτας που τοποθετείται σε ένα σημείο της καμπύλης Bézier σύμφωνα με την πρώτη παράγωγο της καμπύλης στο συγκεκριμένο σημείο.

Πίνακας 5.8 - Οι ρουτίνες μέλη της κλάσης CRouteLabel

Στο Σχήμα 5.5 που ακολουθεί παρουσιάζονται ετικέτες κειμένου που ανήκουν σε διαδρομές.





Σχήμα 5.5 – Ετικέτες κειμένου που αντιστοιχίζονται με διαδρομές

Η απεικόνιση των ετικετών διαδρομής επιτυγχάνεται με τον ακόλουθο τρόπο. Αρχικά, ελέγχουμε το σημείο της καμπύλης Βέζιερ που είναι η αφετηρία της ετικέτας μέσω του πεδίου `m_iStartingPoint`. Στη συνέχεια, τοποθετούμε τα στοιχεία του αλφαριθμητικού που περιέχονται στο πεδίο `m_aBrokenLabel` ξεκινώντας από το παραπάνω σημείο. Σε κάθε σημείο της καμπύλης Βέζιερ όπου τοποθετείται στοιχείο, ελέγχουμε την κλίση που του αντιστοιχεί και έπειτα περιστρέφουμε το στοιχείο που θέλουμε να τοποθετήσουμε σύμφωνα με αυτή.

Η τελευταία υποκλάση της `CPartitionLabel`, η `CRegionLabel` υλοποιεί τις ετικέτες περιοχής, δηλαδή τις ετικέτες που αντιστοιχίζονται σε περιοχές. Το πεδίο `m_iAverageWidth` είναι η μέση απόσταση μεταξύ των στοιχείων που απαρτίζουν την ετικέτα, ενώ στο πεδίο `m_pPlacePoints` αποθηκεύονται οι θέσεις στις οποίες τοποθετούνται τα στοιχεία αυτά. Η κεντρική θέση (η θέση τοποθέτησης του μεσαίου στοιχείου του αλφαριθμητικού) είναι το κέντρο της περιοχής, ενώ τα υπόλοιπα στοιχεία τοποθετούνται συμμετρικά αριστερά και δεξιά της.

Ο Πίνακας 5.9 περιγράφει τις ρουτίνες μέλη της κλάσης `CRegionLabel`.

<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
<b><code>CalcAverageWidth(dc)</code></b>	Υπολογίζει τη μέση απόσταση μεταξύ των στοιχείων της ετικέτας.
<b><code>CalcPlacePoints()</code></b>	Υπολογίζει τις θέσεις των στοιχείων της ετικέτας με βάση



<b>Ρουτίνα(-ες)</b>	<b>Λειτουργία</b>
	τη μέση απόσταση των στοιχείων και το κέντρο της Περιοχής.

Πίνακας 5.9 - Οι ρουτίνες μέλη της κλάσης CRegionLabel

Στο Σχήμα 5.6 δίνεται ένα παράδειγμα ετικέτας περιοχής.



Σχήμα 5.6 – Ετικέτα περιοχής

### 5.3 Δομές Υποστήριξης για το Μοντέλο των Ετικετών

Όπως αναφέραμε στο Κεφάλαιο 4 στο Map Studio έχουν ενσωματωθεί ορισμένες δομές υποστήριξης, οι οποίες χρησιμοποιούνται για τη διαχείριση, αποθήκευση και ανάκτηση αντικειμένων. Οι ίδιες δομές υποστήριξης χρησιμοποιήθηκαν και για την υποστήριξη των ετικετών.

Στην ενότητα 2.6 είδαμε ότι οι δομές υποστήριξης για το μοντέλο γεωγραφικών αντικειμένων του Map Studio (Θέσεις, Διαδρομές και Περιοχές) έχουν ενσωματωθεί στο αντικείμενο του επιπέδου. Όμως, όπως αναφέραμε στην υποενότητα 5.2.1, μας ενδιαφέρει η συνολική τοποθέτηση των ετικετών πάνω στο χάρτη και όχι σε κάθε επίπεδο χωριστά. Για αυτό το λόγο οι ετικέτες είναι Αντικείμενα Χάρτη, συνεπώς και οι δομές υποστήριξης για αυτές ήταν αναγκαίο να τοποθετηθούν στο αντικείμενο χάρτη.

Ως κύρια δομή διαχείρισης των ετικετών χρησιμοποιήσαμε ένα hash table που όπως ακριβώς και για τα άλλα γεωγραφικά αντικείμενα υλοποιείται από την κλάση CMapDWordToOb. Έτσι στο αντικείμενο χάρτη προσθέσαμε το πεδίο **m\_tblLabels** με τη χρήση του οποίου γίνεται ταξινόμηση των ετικετών σύμφωνα με τα αναγνωριστικά τους.

Ανάλογα, για τη γρήγορη επιλογή και ανεύρεση των ετικετών χρησιμοποιούμε τετραδικά δέντρα στο αντικείμενο χάρτη. Τα πεδία **m\_QTLabels** και **m\_QTAugmented** υλοποιούν τα τετραδικά δέντρα για τα ορθογώνια που περικλείουν τις ετικέτες και τα μεγενθυμένα ορθογώνια αντίστοιχα. Τα συγκεκριμένα δέντρα χρησιμοποιούνται από τον αλγόριθμο επίλυσης του ΠΒΤΕ για την απάντηση σε διάφορες χωρικές ερωτήσεις που είναι απαραίτητες για την εκτέλεσή του, όπως αν μία ετικέτα συγκρούεται με μία άλλη, αν μία ετικέτα βρίσκεται μέσα στο μεγενθυμένο ορθογώνιο κάποιας άλλης κ.α. Με τη χρήση του τετραδικού δέντρου επιταχύνονται οι αναζητήσεις των ετικετών και συνεπώς βελτιώνεται η απόδοση του αλγορίθμου. Επιπλέον, τα δέντρα αυτά ελέγχουν πάλι με τον ίδιο μηχανισμό ερωτήσεων-απαντήσεων, αν ο χρήστης επιθυμεί να μετακινήσει κάποια ετικέτα. Αν συμβαίνει μία τέτοια περίπτωση, τότε η ετικέτα μπορεί να επανατοποθετηθεί σύμφωνα με την επιθυμία του χρήστη μέσα στα όρια, βέβαια, που ισχύουν για κάθε τύπο ετικέτας. Στη συνέχεια, η νέα θέση της ετικέτας αποθηκεύεται στο τετραδικό δέντρο.

## **5.4 Τροποποιήσεις στις Υπάρχουσες Δομές της Εφαρμογής**

Στην ενότητα αυτή θα αναφερθούμε σε προθήκες πεδίων και ρουτινών στις υπάρχουσες δομές του Map Studio, έτσι ώστε να είναι δυνατή η ενσωμάτωση του νέου μοντέλου που κατασκευάσαμε στην εφαρμογή.

Στην κλάση CRoute προσθέσαμε τα παρακάτω πεδία και ρουτίνες, τα οποία αναλαμβάνουν την κατασκευή, διαχείριση και αποθήκευση των καμπυλών Βέζιερ που περιγράψαμε στο Κεφάλαιο 3.

Πεδία	Σημασία
<b>m_pBezierPoints</b>	Οι συντεταγμένες των σημείων της καμπύλης Bézier
<b>m_pDerivatives</b>	Το σύνολο των πρώτων παραγώγων για κάθε σημείο της Καμπύλης Bézier

Πίνακας 5.10 – Πεδία της κλάσης CRoute για την υποστήριξη των καμπυλών Bézier

Ρουτίνα (-ες)	Λειτουργία
<b>CreateBezier()</b>	Κατασκευάζει την καμπύλη Bézier της διαδρομής
<b>CalcBernsteinPol()</b>	Υπολογίζει ένα πολυώνυμο Bernstein
<b>CreateCombinations()</b>	Υπολογίζει τους συνδυασμούς $\binom{n}{i}$ που χρησιμοποιούνται από τα πολυώνυμα Bernstein
<b>CalcScope()</b>	Υπολογίζει την κλίση σε ένα σημείο της καμπύλης Bézier
<b>GetBezierPoints()</b>	Επιστρέφει τα σημεία που αποτελούν την καμπύλη Bézier
<b>GetDerivatives()</b>	Επιστρέφει τις πρώτους παραγώγους της καμπύλης Bézier

Πίνακας 5.11 – Ρουτίνες της κλάσης CRoute για την υποστήριξη των καμπυλών Bézier

Η κάθε καμπύλη Bézier δημιουργείται αυτόματα αμέσως μετά την κατασκευή της Διαδρομής στην οποία ανήκει.

Το πεδίο και οι ρουτίνες που ενσωματώσαμε στην κλάση CLayerObjProperties αναλαμβάνουν τη διαχείριση των ετικετών των αντικειμένων και είναι οι ακόλουθες:

Πεδία	Σημασία
<b>m_listLabels</b>	Η λίστα στην οποία αποθηκεύονται τα αναγνωριστικά των ετικετών που ανήκουν στο αντικείμενο

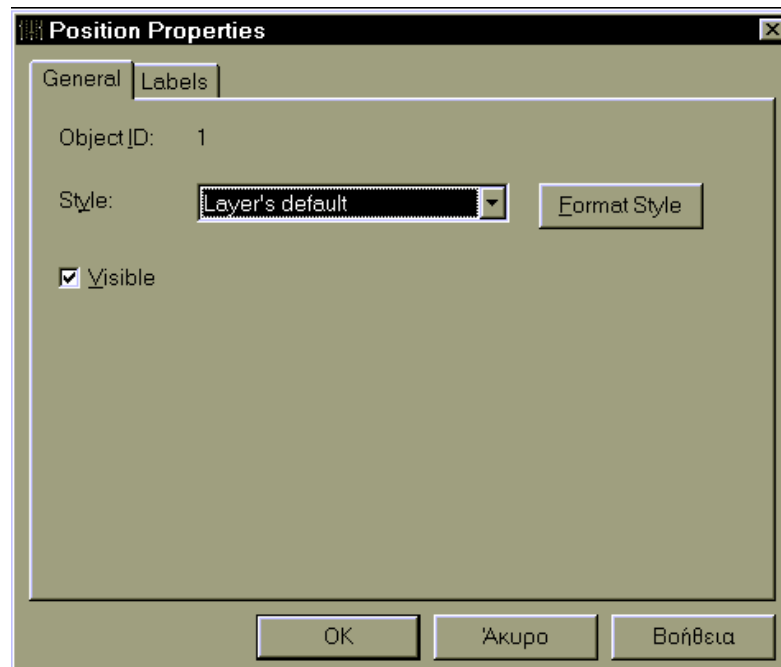
Πίνακας 5.12 – Πεδίο της κλάσης CLayerObjProperties για την υποστήριξη των ετικετών

Ρουτίνα (-ες)	Λειτουργία
<b>GetLabel()</b>	Επιστρέφει ένα δείκτη στη ζητούμενη ετικέτα
<b>GetLabelList()</b>	Επιστρέφει τη λίστα των ετικετών
<b>GetLabelText()</b>	Επιστρέφει το κείμενο μίας συγκεκριμένης ετικέτας
<b>GetCountOfLabels()</b>	Επιστρέφει τον αριθμό των ετικετών του αντικειμένου
<b>SetLabelText()</b>	Θέτει το κείμενο μίας ετικέτας
<b>IsLabelListEmpty()</b>	Εξετάζει αν το αντικείμενο έχει ετικέτες
<b>InsertInLabelList()</b>	Προσθέτει μία ετικέτα στη λίστα
<b>RemoveFromLabelList()</b>	Αφαιρεί μία ετικέτα από τη λίστα
<b>RemoveAllFromLabelList()</b>	Αφαιρεί όλες τις ετικέτες από τη λίστα

Πίνακας 5.13 – Ρουτίνες της κλάσης CLayerObjProperties για την υποστήριξη των ετικετών

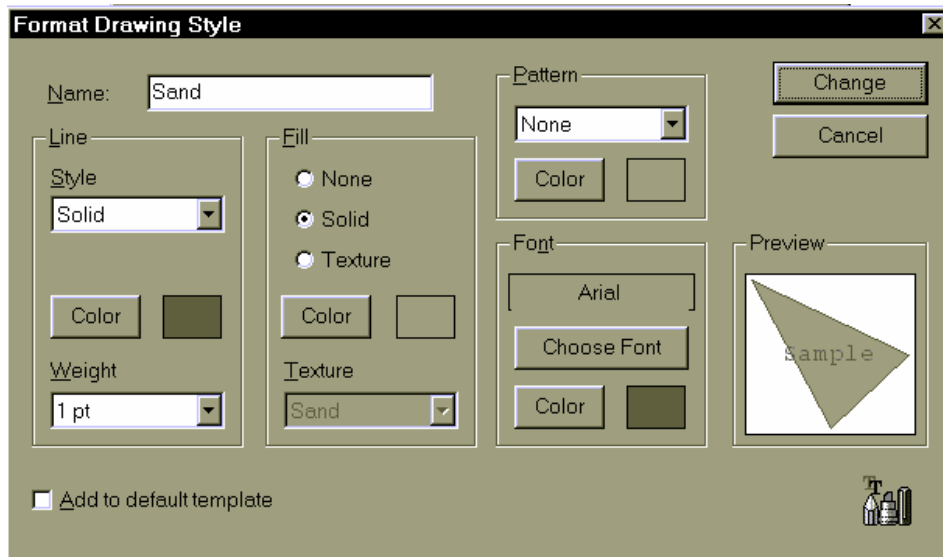
## 5.5 Περιγραφή Μηχανισμού Αλληλεπίδρασης για τη Διαχείριση των Ετικετών

Στην ενότητα αυτή θα αναφερθούμε στο μηχανισμό αλληλεπίδρασης με το χρήστη που υλοποιήθηκε για την υποστήριξη του μοντέλου των ετικετών. Ο συγκεκριμένος μηχανισμός επιτρέπει στο χρήστη να προσθέτει και να αφαιρεί ετικέτες από ένα γεωγραφικό αντικείμενο που έχει διαλέξει. Αφού έχει γίνει η επιλογή του αντικειμένου μέσω της λειτουργίας Properties εμφανίζεται το κουτί διαλόγου του Σχήματος 5.7.



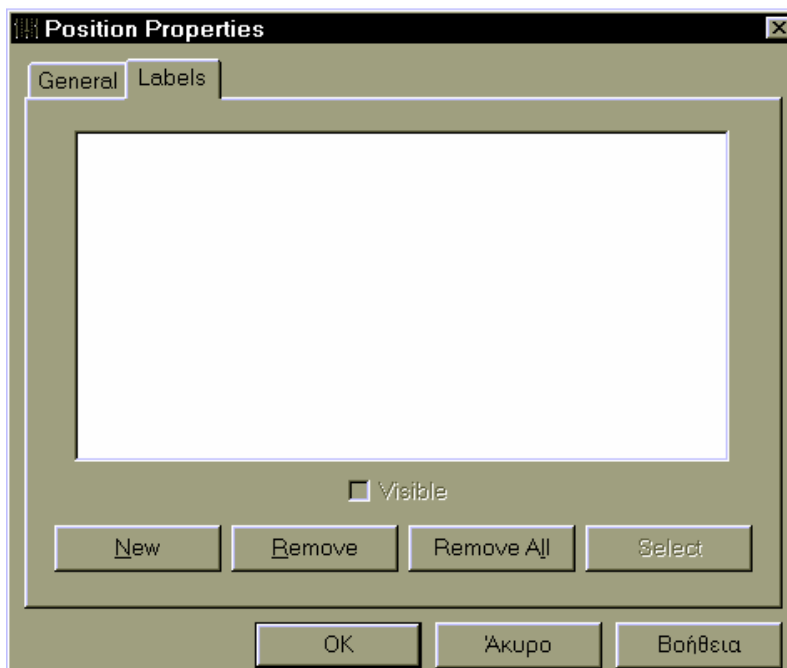
Σχήμα 5.7 – Κουτί διαλόγου για τη λειτουργία Object – Properties

Αν ο χρήστης πατήσει το κουμπί Format Style, τότε μπορεί να τροποποιήσει τις παραμέτρους του στιλ σχεδιασμού (χρώμα, γραμματοσειρά κ.α.), όπως παρουσιάστηκε στο [Ανας95], μέσω του κουτιού διαλόγου του Σχήματος 5.8.



Σχήμα 5.8 – Κουτί διαλόγου μεταβολής του στυλ σχεδιασμού

Αν ο χρήστης επιλέξει τη σελίδα Labels από το κουτί διαλόγου του Σχήματος 5.7, τότε εμφανίζεται το κουτί διαλόγου του Σχήματος 5.9, μέσω του οποίου ο χρήστης μπορεί να προσθέσει, αφαιρέσει και τροποποιήσει μία ετικέτα.



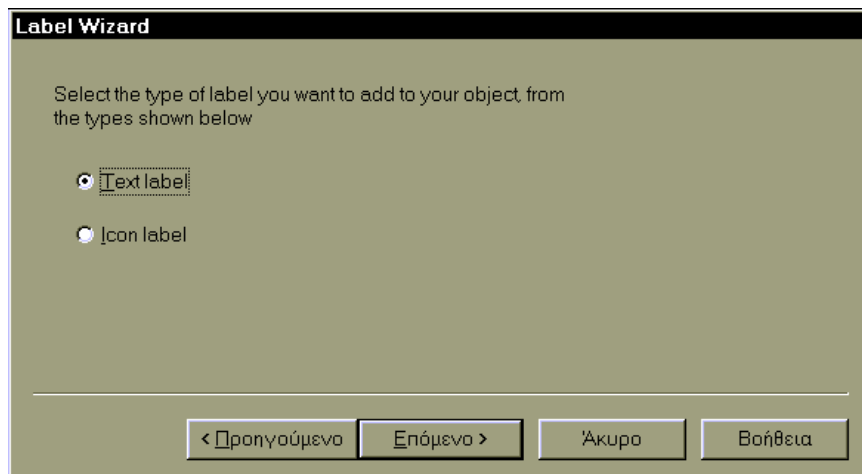
Σχήμα 5.9 – Κουτί διαλόγου για τη διαχείριση των ετικετών

Τα κουμπιά και επιλογή που παρουσιάζονται στη συγκεκριμένη σελίδα εκτελούν τις εξής λειτουργίες:

- **New** - Προσθέτει μία καινούργια ετικέτα στο αντικείμενο.

- **Remove** - Αφαιρεί την ετικέτα που έχει επιλέξει ο χρήστης.
- **Remove All** - Αφαιρεί όλες τις ετικέτες που ανήκουν στο επιλεγμένο αντικείμενο.
- **Select** - Μεταφέρει τον χρήστη στο παράθυρο του χάρτη, στο οποίο έχει επιλεγεί η ετικέτα που επιθυμούσε.
- **Visible** – Από αυτή την επιλογή ο χρήστης μπορεί να αποφασίσει αν η ετικέτα που επιθυμεί θα εμφανίζεται στο παράθυρο του χάρτη (ορατή) ή όχι (αόρατη), μέσω του πεδίου `m_byVisible` της κλάσης `CLabel`. Σε αυτό το σημείο είναι αναγκαίο να τονίσουμε ότι οι «αόρατες» ετικέτες δεν λαμβάνονται υπ' όψιν κατά την εκτέλεση του αλγόριθμου επίλυσης του ΠΒΤΕ.

Αν πατηθεί το κουμπί **New**, τότε μεταφερόμαστε στο πρώτο φύλλο ιδιοτήτων (property sheet) του μάγου (wizard<sup>8</sup>) ετικετών του Σχήματος 5.10. Σε αυτό το σημείο ο χρήστης μπορεί να επιλέξει αν θα τοποθετήσει ετικέτα κειμένου (text label) ή ετικέτα εικονίδιο (icon label).

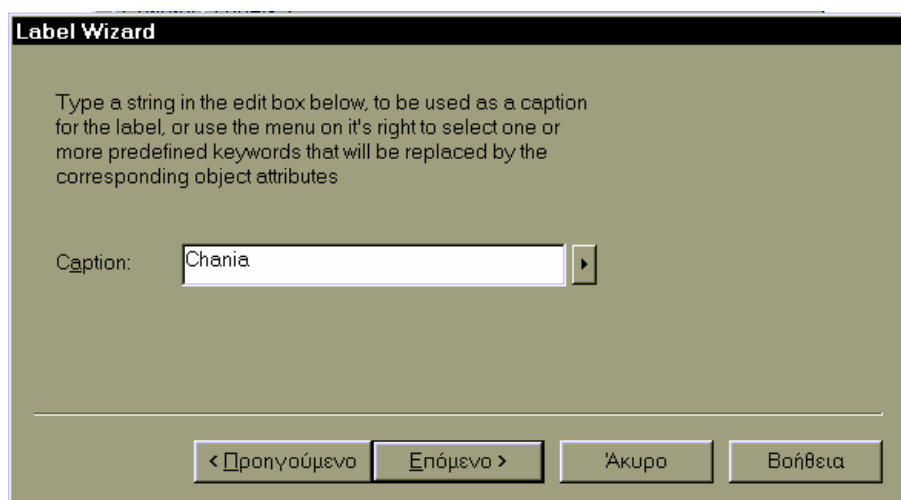


Σχήμα 5.10 – Φύλλο μάγου για την επιλογή του τύπου της ετικέτας

Αν επιλεγεί να τοποθετηθεί ετικέτα κειμένου, τότε μεταφερόμαστε στο επόμενο φύλλο, όπως παρουσιάζεται στο Σχήμα 5.11. Σε αυτό το φύλλο ο χρήστης μπορεί να πληκτρολογήσει το κείμενο που επιθυμεί να τοποθετηθεί ως ετικέτα στο αντικείμενο. Μέσω του μενού που εμφανίζεται, αν πατηθεί το δεξιό κουμπί δίπλα από το κουτί

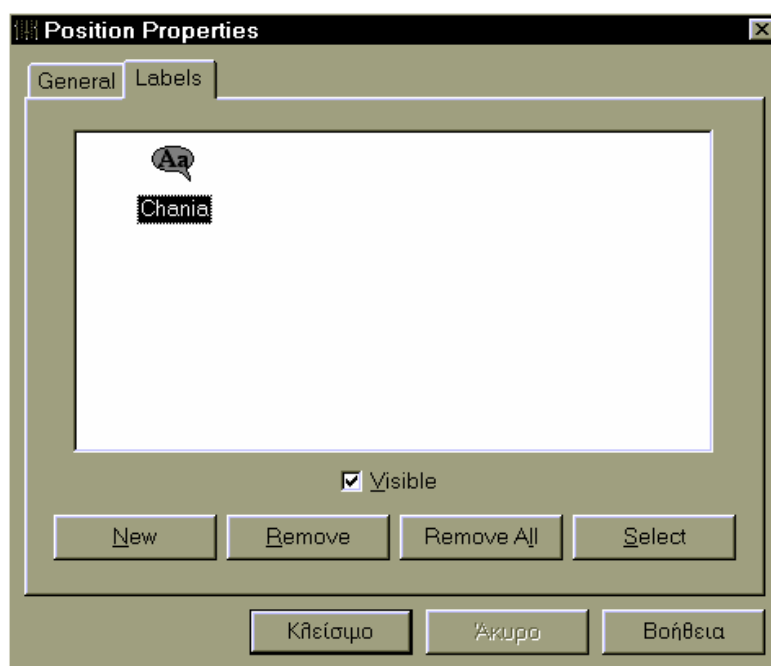
<sup>8</sup> Μάγος είναι ένα σύνθετο κουτί διαλόγου (dialog box) που αποτελείται από ένα σύνολο φύλλων ιδιοτήτων. Ένας μάγος δέχεται εντολές χρήστη μέσω των φύλλων του με μία συγκεκριμένη σειρά με σκοπό να εκτελέσει μία συγκεκριμένη εργασία.

κειμένου, μπορούμε να εισάγουμε στο κείμενο πληροφορίες σχετικά με το αντικείμενο της ετικέτας, όπως έχουμε ήδη αναφέρει στην υποενότητα 5.2.3.



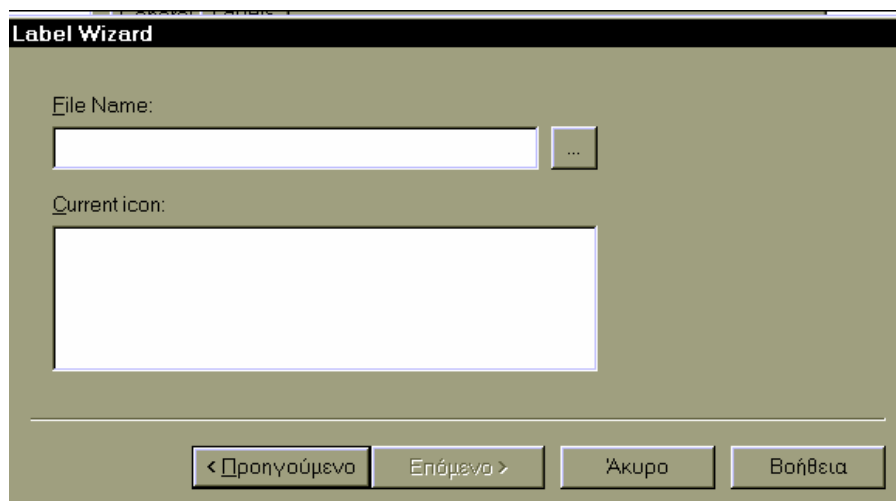
Σχήμα 5.11 – Φύλλο μάγου για την εισαγωγή του κειμένου της ετικέτας

Στη συνέχεια, αφού πατήσουμε το κουμπί Επόμενο, μεταφερόμαστε πίσω στη σελίδα διαχείρισης των ετικετών, όπου πλέον η ετικέτα που δημιουργήσαμε εμφανίζεται στο κουτί παρουσίασης, όπως φαίνεται στο Σχήμα 5.12. Οι ετικέτες κειμένου εμφανίζονται πάντα στο κουτί παρουσίασης με τη μορφή ενός συγκεκριμένου εικονιδίου με τοποθετημένο το κείμενο της ετικέτας στο κάτω μέρος του.



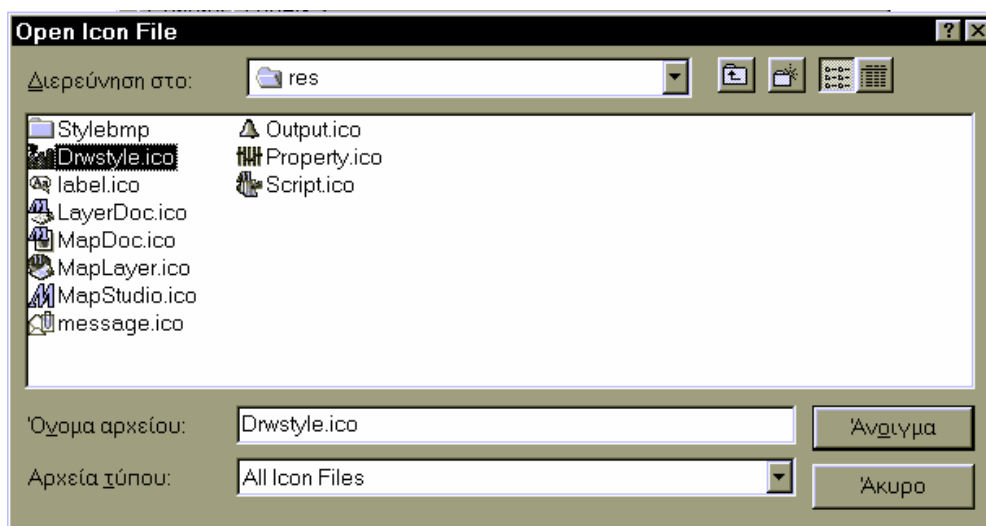
Σχήμα 5.12 – Σελίδα διαχείρισης ετικετών με τοποθετημένη ετικέτα κειμένου

Αν στο φύλλο του Σχήματος 5.10 επιλέγαμε την εισαγωγή ετικέτας εικονίδιο, τότε θα μεταφερόμασταν στο φύλλο του Σχήματος 5.13. το φύλλο αυτό αυτή είναι η σελίδα εισαγωγής των ετικετών εικονίδιο.



Σχήμα 5.13 – Φύλλο μάγου για την εισαγωγή ετικέτας εικονίδιο

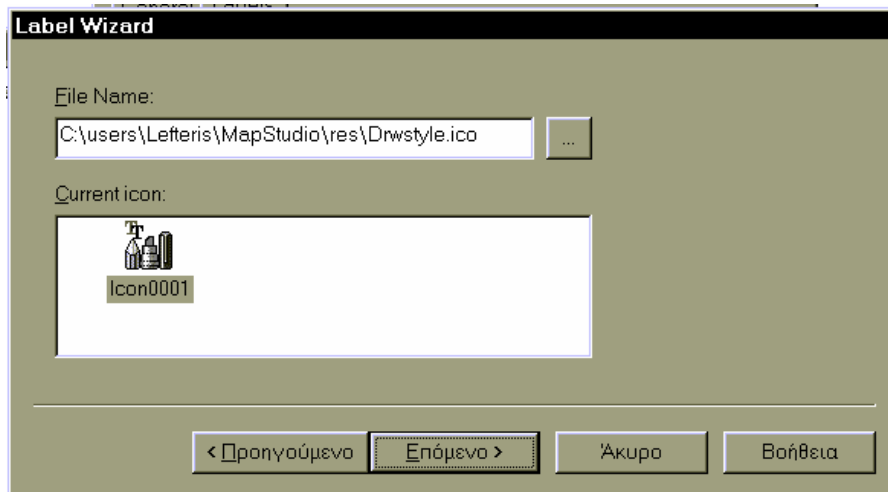
Πατώντας το κουμπί δεξιά του κουτιού κειμένου με την ονομασία File Name μεταφερόμαστε στο κουτί διαλόγου του Σχήματος 5.14, από το οποίο επιλέγουμε το αρχείο που περιέχει το εικονίδιο που θέλουμε να τοποθετήσουμε ως ετικέτα.



Σχήμα 5.14 – Κουτί διαλόγου επιλογής αρχείου με εικονίδιο

Αφού επιλέξουμε το αρχείο που επιθυμούμε, επιστρέφουμε στην προηγούμενη σελίδα, όπου πλέον στο κουτί κειμένου File Name εμφανίζεται το όνομα του αρχείου που επιλέξαμε και στο κουτί παρουσίασης τα εικονίδια που υπάρχουν μέσα στο αρχείο, όπως φαίνεται και στο Σχήμα 5.15.





Σχήμα 5.15 – Φύλλο μάγου για την εισαγωγή ετικετών εικονίδιο με επιλεγμένο το εικονίδιο που επιθυμούμε

Αφού επιλέξουμε το εικονίδιο που θέλουμε, πατώντας το κουμπί Επόμενο επιστρέφουμε στη σελίδα διαχείρισης ετικετών, όπου πλέον εμφανίζεται η νέα ετικέτα που κατασκευάσαμε στο κουτί εμφάνισης, όπως φαίνεται στο Σχήμα 5.16.



Σχήμα 5.16 – Σελίδα διαχείρισης ετικετών με τοποθετημένη ετικέτα εικονίδιο

## 5.4 Περίληψη

Στο Κεφάλαιο αυτό περιγράψαμε το μοντέλο αντικειμένων ετικετών που υλοποιήσαμε στα πλαίσια αυτής της διπλωματικής εργασίας και στη σύνδεσή του με

την προϋπάρχουσα ιεραρχία κλάσεων της εφαρμογής Map studio. Αναφερθήκαμε στα πεδία και τις ρουτίνες των κλάσεων των ετικετών, στις δομές υποστήριξης που ενσωματώθηκαν ειδικά για τις ετικέτες και στους τρόπους χρησιμοποίησής τους από τον αλγόριθμο επίλυσης του ΠΒΤΕ και για την απεικόνισή τους στο παράθυρο χάρτη. Επιπλέον, παρουσιάσαμε ορισμένα παραδείγματα τοποθέτησης των διάφορων τύπων ετικετών στο παράθυρο χρήστη.

Τέλος, αναφερθήκαμε στο μηχανισμό αλληλεπίδρασης εφαρμογής-χρήστη που σχεδιάσαμε για την υποστήριξη των λειτουργιών εισαγωγής, διαγραφής και τροποποίησης των ετικετών.

## **Κεφάλαιο 6**

# **Ανακεφαλαίωση, Συνεισφορά και Μελλοντικές Επεκτάσεις**

### **6.1 Γενικά**

Στο Κεφάλαιο αυτό θα παρουσιαστεί η τελική συνεισφορά της εργασίας καθώς και ορισμένα γενικά συμπεράσματα. Ο σχεδιασμός είναι τέτοιος ώστε να καθιστά υλοποιήσιμες ορισμένες μελλοντικές επεκτάσεις. Θα αναφερθούμε, συνεπώς, σε κάποιες προτάσεις για την μελλοντική επέκταση και εξέλιξή της.

### **6.2 Ανακεφαλαίωση**

Η εργασία αυτή αποσκοπούσε στο σχεδιασμό και την υλοποίηση ενός περιβάλλοντος διαχείρισης ετικετών για την εφαρμογή Map Studio [Ανας95]. Επιπλέον υλοποιήθηκε ένας μηχανισμός αυτόματης τοποθέτησης ετικετών στους χάρτες της εφαρμογής.

Αρχικά, προχωρήσαμε σε μία έρευνα σχετικά με τους αλγόριθμους που είχαν προταθεί στο παρελθόν για την εύρεση λύσης στο Πρόβλημα Βέλτιστης Τοποθέτησης Ετικετών για σημειακές ετικέτες, οι οποίοι όμως δεν ήταν ικανοί να προσφέρουν αποδεκτές λύσεις. Αναλύσαμε διεξοδικά τη φιλοσοφία πάνω στην οποία στηρίχθηκε ο καθένας από αυτούς και τονίσαμε τις διάφορες καταστάσεις που μπορούν να εμφανιστούν κατά την τοποθέτηση ετικετών, τις οποίες αδυνατούν να επιλύσουν.

Ένας άλλος παράγοντας που επηρεάζει σε σημαντικό βαθμό την αυτόματη τοποθέτηση των σημειακών ετικετών είναι ο ελεύθερος χώρος τοποθέτησης, ο οποίος προκύπτει αν από τον ολικό χώρο τοποθέτησης μίας ετικέτας αφαιρέσουμε το συνολικό εμβαδόν των επικαλύψεων που παρουσιάζει με τις γειτονικές ετικέτες. Μία σωστή μέθοδος αναπαράστασης του ελεύθερου χώρου αυξάνει αυτόματα τον αριθμό των δυνατών θέσεων τοποθέτησης των ετικετών, με συνέπεια να αυξάνονται ανάλογα οι πιθανότητες ο αλγόριθμος επίλυσης του ΠΒΤΕ να καταλήγει σε αποδεκτές λύσεις. Περιγράψαμε ορισμένους αλγόριθμους ανεύρεσης του ελεύθερου χώρου και καταλήξαμε στη χρησιμοποίηση του αλγόριθμου που χρησιμοποιεί τα μέγιστα ελεύθερα ορθογώνια. Τα ελεύθερα ορθογώνια που δημιουργούνται μετά την εκτέλεση του παραπάνω αλγόριθμου καθορίζουν τις περιοχές μέσα στις οποίες μπορεί να τοποθετηθεί η ετικέτα που ανήκουν χωρίς να παρουσιάζει επικαλύψεις με τις γειτονικές προς αυτήν ετικέτες.

Βασιζόμενοι στη σχετική έρευνα που αναφέραμε προηγουμένως, καταλήξαμε στο συμπέρασμα ότι ένας αλγόριθμος που στηρίζεται και χρησιμοποιεί ευριστικές μεθόδους θα πρόσφερε καλύτερα αποτελέσματα στην προσπάθεια επίλυσης του ΠΒΤΕ. Μία τέτοια είναι και η μέθοδος Προσομοίωσης Ανόπτησης, την οποία και υλοποιήσαμε. Για να αντιμετωπίσουμε σωστότερα το ΠΒΤΕ, διαιρέσαμε τον αλγόριθμο σε τέσσερα τμήματα. Η επιμέρους αντιμετώπιση των τμημάτων αυτών διευκόλυνε τη διαδικασία επίλυσης του προβλήματος, ενώ ταυτόχρονα βελτίωσε τόσο την ποιότητα των τελικών αποτελεσμάτων, όσο και το συνολικό χρόνο εκτέλεσης του αλγορίθμου.

Οι ετικέτες διαδρομής παρουσιάζουν αρκετές ιδιαιτερότητες ως προς τον τρόπο παρουσίασής τους. Η τοποθέτηση των ετικετών αυτών κατά μήκος της διαδρομής στην οποία ανήκουν δημιουργεί άσχημο αισθητικά αποτέλεσμα στα σημεία καμπυλότητας. Για να αποφύγουμε το παραπάνω πρόβλημα, επιλέξαμε τη χρήση των καμπυλών Bézier ως οδηγούς για την τοποθέτηση των συγκεκριμένων ετικετών. Οι καμπύλες αυτές είναι σχετικά εύκολο να υλοποιηθούν, ενώ μέσω της πρώτης παραγώγου για τις συντεταγμένες κάθε σημείου τοποθέτησης στη καμπύλη δώσαμε κλίση στους χαρακτήρες της ετικέτας, κάτι που βελτίωσε το οπτικό αποτέλεσμα.

Η προαναφερόμενη ιδιαιτερότητα των ετικετών διαδρομής δεν επέτρεψε τη χρησιμοποίηση της μεθόδου ΠΑ για την αυτόματη τοποθέτησή τους. Για να

επιλύσουμε το παραπάνω πρόβλημα τοποθέτησης αναπτύξαμε και υλοποιήσαμε ένα διαφορετικό αλγόριθμο εύρεσης της βέλτιστης θέσης τοποθέτησης για τις συγκεκριμένες ετικέτες, τον οποίο ενσωματώσαμε, τελικά, στη συνολική λύση του ΠΒΤΕ.

Όπως έχουμε ήδη αναφέρει αυτή η διπλωματική εργασία στηρίχτηκε στην εφαρμογή Map Studio, που είναι περιβάλλον ανάπτυξης εφαρμογών γεωγραφικών δεδομένων. Κρίθηκε απαραίτητο να αναφερθούμε στο μοντέλο αντικειμένων και τις δομές υποστηρίξις τους, πάνω στα οποία έχει υλοποιηθεί το Map Studio, έτσι ώστε να διευκολύνουμε τον αναγνώστη να καταλάβει σε μεγαλύτερο βαθμό τη βάση στην οποία στηρίχτηκε η δική μας εργασία. Σε αυτή την αναφορά γίνεται μία λεπτομερής περιγραφή των κλάσεων που υλοποιούν τα αντικείμενα της εφαρμογής, του τρόπου ομαδοποίησής τους σε επίπεδα και κατόπιν σε χάρτες, καθώς και των δομών υποστήριξης που αναλαμβάνουν τη διαχείρισή τους.

Την παραπάνω παρουσίαση του Map Studio ακολουθεί η ανάλυση του μοντέλου αντικειμένων ετικετών που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας. Παρουσιάσαμε την ιεραρχία των κλάσεων που αντιπροσωπεύουν τις ετικέτες και περιγράψαμε τα πεδία και τις ρουτίνες που περιέχονται σε αυτές. Επιπλέον, αναλύσαμε τους τρόπους με τους οποίους τα παραπάνω στοιχεία χρησιμοποιούνται για την παρουσίαση των ετικετών στο χρήστη μέσω του παράθυρου χάρτη της εφαρμογής και περιγράψαμε την υποστήριξη που παρέχεται από τις δομές της εφαρμογής για την παραπάνω παρουσίαση και για τον αλγόριθμο επίλυσης του ΠΒΤΕ. Τέλος, αναφερθήκαμε στο μηχανισμό αλληλεπίδρασης που κατασκευάστηκε για να προσφέρει στο χρήστη λειτουργίες όπως εισαγωγή, διαγραφή και διαχείριση ετικετών.

### **6.3 Συνεισφορά**

Στα πλαίσια της διατριβής αυτής υλοποιήθηκε ένα ολοκληρωμένο μοντέλο κατασκευής και παρουσίασης ετικετών. Εκμεταλλευόμενοι τους μηχανισμούς αλληλεπίδρασης των Microsoft Windows κατασκευάσαμε ένα φιλικό περιβάλλον εισαγωγής, διαγραφής και διαχείρισης των ιδιοτήτων των ετικετών μέσω της χρήσης

κουτιών διαλόγου. Ο μηχανισμός αυτός παρουσιάζει αρκετή ευκολία στη χρήση του και επιτρέπει το καθορισμό όλων των παραμέτρων στους οποίους μπορεί να επέμβει ο χρήστης, όπως χρώμα, γραμματοσειρά, μέγεθος γραμμάτων κ.α., με απλό, κατανοητό και γρήγορο τρόπο.

Επιπλέον, μέσω του περιβάλλοντος WYSIWYG παρέχεται η δυνατότητα επανατοποθέτησης των ετικετών από τον χρήστη μέσα στα όρια που έχουν τεθεί για κάθε τύπο ετικέτας. Η επιλογή της ετικέτας που επιθυμούμε να μετακινήσουμε γίνεται με ένα απλό πάτημα στη συγκεκριμένη ετικέτα στο παράθυρο χάρτη, όπου απεικονίζεται. Ο έλεγχος που δίνεται στο χρήστη είναι πλήρης αλλά αρκετά εύχρηστος στη λειτουργία του.

Αναπτύξαμε μηχανισμούς τοποθέτησης και παρουσίασης για κάθε τύπο ετικετών, οι οποίοι προσφέρουν ένα καλό αισθητικά αποτέλεσμα στη συνολική παρουσίαση των χαρτών της εφαρμογής. Ειδικότερα, για τις ετικέτες γραμμικών αντικειμένων η χρησιμοποίηση των καμπυλών Bézier ως οδηγό για την τοποθέτησή τους προσφέρει αρκετά πιο καλαίσθητα αποτελέσματα αν συγκριθεί με την τοποθέτηση κατά μήκος της γραμμής. Η ομαλοποίηση των ανωμαλιών της γραμμής από την οποία προέρχεται και η τοποθέτηση των χαρακτήρων που απαρτίζουν την ετικέτα με κλίση στα σημεία της καμπύλης είναι οι κυριότεροι λόγοι επιλογής τους. Επιπλέον, η υλοποίηση των καμπυλών Bézier αποτέλεσε τη βάση για την ανάπτυξη του αλγόριθμου επίλυσης του ΠΒΤΕ για τις συγκεκριμένες ετικέτες.

Το μοντέλο ετικετών που υλοποιήσαμε επιτρέπει τη δημιουργία τόσο ετικετών κειμένου όσο και ετικετών που απεικονίζουν εικονίδια. Οι ετικέτες εικονίδια μπορούν να τοποθετηθούν είτε γύρω από ένα σημείο είτε πάνω σε μία γραμμή στα εσωτερικά της σημεία. Και στις δύο περιπτώσεις ο χρήστης μπορεί να επανατοποθετήσει το εικονίδιο γύρω από το σημείο αναφοράς από το παράθυρο χρήστη με το μηχανισμό που περιγράψαμε προηγουμένως. Στη δεύτερη περίπτωση είναι δυνατή η μετακίνηση του εικονιδίου σε οποιοδήποτε εσωτερικό της σημείο.

Αναπτύξαμε και υλοποιήσαμε αλγορίθμους αυτόματης τοποθέτησης των ετικετών, διαδικασία η οποία κρίνεται αναγκαία για χάρτες εύλογου μεγέθους. Ο χρήστης δεν χρειάζεται να επανατοποθετεί τις ετικέτες που προσθέτει σε ένα χάρτη έτσι ώστε να μην παρουσιάζονται επικαλύψεις μεταξύ των γραφικών αντικειμένων, καθώς η

χρησιμοποίηση του παραπάνω μηχανισμού αναλαμβάνει την τοποθέτηση των ετικετών σε ελεύθερες θέσεις. Προσθέσαμε στον αλγόριθμο επίλυσης του ΠΒΤΕ ένα αρχικό βήμα τοποθέτησης, κατά το οποίο δεν εισάγουμε στο πρόγραμμα ανόπτησης ετικέτες οι οποίες μπορούν να τοποθετηθούν χωρίς να επικαλύπτουν άλλες ετικέτες, ενώ ομαδοποιούμε τις ετικέτες που είναι πιθανόν ή παρουσιάζουν, ήδη, επικαλύψεις. Οι ομάδες αυτές εξετάζονται στη συνέχεια από το πρόγραμμα ανόπτησης.

Τέλος, σχεδιάσαμε έναν εντελώς νέο αλγόριθμο αυτόματης τοποθέτησης των γραμμικών ετικετών, τον οποίο ενσωματώσαμε στη γενική λύση του ΠΒΤΕ. Ο αλγόριθμος αυτός στηρίζεται στις ιδιότητες των καμπυλών Bézier. Ως πρώτο βήμα του αλγορίθμου αυτού προχωράμε στην ακύρωση των σημείων που ορίζουν την καμπύλη και παρουσιάζουν επικαλύψεις από άλλες ετικέτες. Στη συνέχεια βρίσκουμε το μέγιστο συνεχόμενο πλήθος μη ακυρωμένων σημείων, το οποίο αποτελεί και τη βέλτιστη θέση τοποθέτησης της ετικέτας.

## 6.4 Μελλοντικές Επεκτάσεις

Στόχος της παρούσας εργασίας είναι να εξελιχθεί μελλοντικά έτσι ώστε να προστεθούν νέες δυνατότητες στην παρουσίαση των ετικετών.

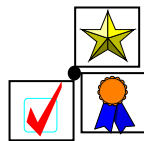
Στα σχέδια επεκτάσεων εντάσσεται η προσθήκη προχωρημένων χαρακτηριστικών παρουσίασης, οι οποίες θα εκμεταλλεύονται σε μεγαλύτερο βαθμό τις δυνατότητες παρουσίασης των Microsoft Windows. Ορισμένα από τα χαρακτηριστικά αυτά είναι τα εξής:

- **Σκίαση ετικετών.** Θα εμφανίζεται η σκιά των ετικετών στο χάρτη, κάτι που προσδίδει μία τριδιάστατη προοπτική.
- **Ετικέτες κειμένου πολλαπλών γραμμών.** Η κάθε ετικέτα κειμένου θα είναι δυνατόν να εμφανίζει τη συμβολοσειρά της σε περισσότερες από μία γραμμές.
- **Χρήση διαφορετικών στιλ σχεδιασμού για την ίδια ετικέτα.** Η σύνδεση μίας ετικέτας με διάφορα στιλ σχεδιασμού ταυτόχρονα, επιτρέπει π.χ. τη χρήση διαφορετικής γραμματοσειράς ή χρώματος για κάθε λέξη του κειμένου της.

- **Παρουσίαση επιπλέον πληροφορίας.** Ο χρήστης θα μπορεί πατώντας μία ετικέτα να λαμβάνει περισσότερη πληροφορία σχετικά με το αντικείμενο, όπως επεξηγηματικό κείμενο, προβολή video κ.α.
- **Κλίση.** Θα υποστηρίζεται η τοποθέτηση ετικετών σημειακών αντικειμένων με κλίση γύρω από το σημείο τοποθέτησης.

Τα εικονίδια, συνήθως, σε ένα χάρτη εμφανίζονται ως ένα σύνολο που αποδίδουν ένα ολοκληρωμένο ποσό πληροφορίας για ένα αντικείμενο. Ένα παράδειγμα τέτοιου συνόλου είναι η περιγραφή μέσω εικονιδίων των εγκαταστάσεων ενός ξενοδοχειακού συγκροτήματος (πισίνα, εστιατόριο, κ.τ.λ.). Συνεπώς, η αντιμετώπιση των ετικετών εικονίδιο ως σύνολα θα βοηθούσε στην παραγωγή πιο καλαίσθητων χαρτών. Ορισμένοι τρόποι ομαδοποίησης των ετικετών εικονίδιο είναι οι εξής:

- **Κυκλικά.** Τα εικονίδια τοποθετούνται κυκλικά γύρω από το σημείο.



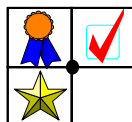
Σχήμα 6.1 – Κυκλική τοποθέτηση εικονιδίων

- **Γραμμικά.** Τα εικονίδια τοποθετούνται σε μία γραμμή που τέμνει το σημείο.



Σχήμα 6.2 – Γραμμική τοποθέτηση εικονιδίων

- **Σε πλέγμα.** Τα εικονίδια τοποθετούνται σε ένα πλέγμα που έχει ως κέντρο του το σημείο.



Σχήμα 6.3 – Τοποθέτηση εικονιδίων σε πλέγμα



Στη παρούσα εργασία υλοποιήθηκε ένα μοντέλο τοποθέτησης ετικετών διαδρομής βασισμένο στις καμπύλες Bézier. Μία πιθανή επέκταση του υπάρχοντος συστήματος τοποθέτησης αυτών των ετικετών είναι η υλοποίηση επιπλέον μοντέλων, τα οποία θα στηρίζονται σε διαφορετικές καμπύλες, όπως οι B-spline και Hermite, οι οποίες θα προσφέρουν εναλλακτικούς τρόπους τοποθέτησής τους.

Μία ακόμη επέκταση της εργασίας είναι η δυνατότητα επανατοποθέτησης των ετικετών περιοχής και η πιθανή ενσωμάτωση κάποιου αλγορίθμου σχετικά με αυτές στο μηχανισμό αυτόματης τοποθέτησης.

Τέλος, με τη χρησιμοποίηση ορισμένων νέων τύπων αρχείων εικόνων που υποστηρίζουν κίνηση (animation), θα μπορούσε να υλοποιηθεί ένας νέος τύπος ετικετών που θα εκμεταλλεύεται το συγκεκριμένο χαρακτηριστικό.

# Παράρτημα Α

*Συμβάσεις Ονομασιών κατά τα Πρότυπα της MFC  
(Επεκτάσεις στην Ουγγρική Σύμβαση)*

<b>Όνομα/Τύπος Μεταβλητής</b>	<b>Πρόθεμα</b>
Κλάση	C
Χαρακτηριστικό (Πεδίο) κλάσης	m_
ABORTPROC	abrtprc
ATOM	atm
BOOL	b, f
BYTE	by
BYTE FAR	lpby
char FAR	lpch
DLGPROC	dlgprc
DWORD	dw
DWORD FAR	lpdw
EDITWORDBREAKPROC	ewbprc
ENUMPROPPROC	enmprc
FONTENUMPROC	fntenmprc
GNOTIFYPROC	gnprc
GOBJENUMPROC	goenmprc
GRAYSTRINGPROC	gsprc
HACCEL	haccl
HBITMAP	hbm
HBRUSH	hbr
HCURSOR	hcur
HDC	hdc
HDRVR	hdrvr
HDWP	hdwp
HFILE	hf
HFONT	hfont
HGDIOBJ	hgdiobj
HGLOBAL	hglb
HHOOK	hhook

<b>Όνομα/Τύπος Μεταβλητής</b>	<b>Πρόθεμα</b>
HICON	hicon
HINSTANCE	hinst
HLOCAL	hloc
HMENU	hmenu
HMETAFILE	hmf
HMODULE	hmod
HOOKPROC	hkprc
HPALETTE	hpal
HPEN	hpen
HRGN	hrgn
HRSRC	hrsrc
HSTR	hstr
HTASK	htask
HWND	hwnd
int	i, n (short int), cb (πλήθος BYTES), cx (πλάτος), cy (ύψος)
LINEDDAPROC	lnddaprc
LNOTIFYPROC	lnprc
LONG	l
LPARAM	lParam
LPBYTE	lpby
LPCSTR	lpsz
LPINT	lpn
LPLONG	lpl
LPSTR	lpsz
LPVOID	lpv
LPWORD	lpw
LRESULT	lResult
MFENUMPROC	mfenmprc
NPSTR	npsz
PBYTE	npby
POINT FAR	lppt
PROPENUMPROC	prpenmprc
RECT FAR	lprc
RSRCHDLRPROC	rschldprc
TIMERPROC	tmprc
UINT	u (προαιρετικό)
WNDENUMPROC	wndenmprc
WNDPROC	wndprc
WORD	u, w
LPARAM	wParam

# Παράρτημα Β

## Οι βασικές κλάσεις της βιβλιοθήκης MFC

<b>Κλάση</b>	<b>Περιγραφή</b>
CObject	Η βασική κλάση για τις περισσότερες κλάσεις αντικειμένων της MFC.
CCommandTarget	Η βασική κλάση για όλες τις κλάσεις αντικειμένων, οι οποίες μπορούν να λαμβάνουν και να ανταποκρίνονται σε μηνύματα.
CWinApp	Περιέχει τον απαραίτητο κώδικα για την εκκίνηση, εκτέλεση και τερματισμό της εφαρμογής.
CDocument	Η βασική κλάση για τα έγγραφα (documents) κάθε εφαρμογής. Για κάθε τύπο εγγράφου που υποστηρίζει η εφαρμογή, παράγουμε μία κλάση από τη CDocument.
CWnd	Η βασική κλάση για τους διάφορους τύπους παραθύρων.
CFrameWnd	Η βασική κλάση για παράθυρα που, σημασιολογικά, αποτελούν το πλαίσιο για άλλους τύπους παραθύρων τα οποία και εμπεριέχουν ως «παιδιά» (child windows).
CView	Η βασική κλάση για τις εξειδικευμένες απόψεις (views) στα δεδομένα των εγγράφων της εφαρμογής. Οι απόψεις εμφανίζουν τα δεδομένα και ερμηνεύουν τις ενέργειες του χρήστη που έχουν σκοπό την επιλογή και τη μεταβολή των δεδομένων.
CDialog	Η βασική κλάση για κουτιά διαλόγου τα οποία ο χρήστης καλείται να εφοδιάσει με απαραίτητες πληροφορίες για την διεκπεραίωση κάποιων λειτουργιών της εφαρμογής.
CDocTemplate	Η βασική κλάση υποστήριξης προτύπων εγγράφων. Αντιστοιχίζει ένα τύπο CFrameWnd και ένα CView με ένα τύπο εγγράφου (CDocument).

# Βιβλιογραφία

- [Μουμ97] N. Μουμουτζής, «Σχεδιασμός Συστήματος Ανάπτυξης Αλληλεπιδραστικών Γεωγραφικών Εφαρμογών», Πολυτεχνείο Κρήτης, 1997
- [Ανας95] Μ. Αναστασιάδης, «Σχεδιασμός και Υλοποίηση ενός Ολοκληρωμένου Περιβάλλοντος Ανάπτυξης Εφαρμογών Γεωγραφικών Δεδομένων», Πολυτεχνείο Κρήτης, 1995
- [Cdre96] C. Cullens, M. Davidson, P. Robichaux, C. Corry, S. Potts, K. Gregory, «Using Visual C++ 4», Que Corporation, 1996, ISBN 1-7897-0401-3
- [Chms95] J. Christensen, J. Marks, S. Shieber, «An empirical Study of Algorithms for Point-Feature Label Placement», ACM Transactions on Graphics, Vol. 14, No. 3, July 1995, 203-232
- [Foda82] J. D. Foley, A. Van Dam, «Fundamentals of Interactive Computer Graphics», Addison-Wesley Publishing, 1982, ISBN 0-201-14468-9
- [Joy 96a] K. Joy, «The Bézier Curves», Graphical Notes, Computer Science Department, University of California, Davis, 1996
- [Joy 96b] K. Joy, «The Bernstein Polynomials», Graphical Notes, Computer Science Department, University of California, Davis, 1996
- [Kede82] G. Kedem, «The Quad-CIF Tree: a Data Structure for Hierarchical On-Line Algorithms», Proc. 19<sup>th</sup> Design Automation Conference, 1982
- [Krug96] D. J. Kruglinski, «Inside Visual C++», 3<sup>rd</sup> Edition, Microsoft Press, 1996, ISBN 1-55615-891-2
- [Petz96] C. Petzold, «Programming Windows 95», 4<sup>th</sup> Edition, Microsoft Press, 1996, ISBN 1-55615-676-6
- [Prep85] F. P. Preparata, M. I. Shamos, «Computational Geometry: An Introduction», Springer-Verlag Berlin Heidelberg, 1985
- [Same90] H. Samet, «Applications of Spatial Data Structures: Computer Graphics, Image Processing and G.I.S.», Addison-Wesley Publishing, 1990, ISBN 0-201-50300-X
- [Schi91] H. Schildt, «C++ The Complete Reference», Osborne-McGraw Hill, 1991, ISBN 0-07-881654-8
- [Stro91] B. Stroustrup, «The C++ Programming Language», 2<sup>nd</sup> Edition, Addison-Wesley Publishing, 1991, ISBN 0-201-53992-6
- [Watt92] A. Watt, M. Watt, «Advanced Animation and Rendering Techniques Theory and Practice», ACM Press, 1992, ISBN 0-201-54412-1