



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΠΟΛΙΤΙΚΩΝ ΜΗΧΑΝΙΚΩΝ

ΤΟΜΕΑΣ ΜΕΤΑΦΟΡΩΝ ΚΑΙ ΣΥΓΚΟΙΝΩΝΙΑΚΗΣ ΥΠΟΔΟΜΗΣ

ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΔΡΟΜΟΛΟΓΗΣΗΣ ΣΥΝΕΡΓΕΙΩΝ ΚΑΤΑ ΤΗΝ ΕΚΤΕΛΕΣΗ ΣΠΟΡΑΔΙΚΩΝ ΕΡΓΑΣΙΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

ΦΟΙΒΟΥ ΓΚΕΛΛΑ

ΕΠΙΒΛΕΠΩΝ: ΚΩΝΣΤΑΝΤΙΝΟΣ ΓΚΙΟΤΣΑΛΙΤΗΣ

ΕΠΙΚΟΥΡΟΣ ΚΑΘΗΓΗΤΗΣ Ε.Μ.Π.

ΕΡΓΑΣΤΗΡΙΟ ΣΙΔΗΡΟΔΡΟΜΙΚΗΣ ΤΕΧΝΙΚΗΣ ΚΑΙ ΜΕΤΑΦΟΡΩΝ

ΑΘΗΝΑ, ΙΟΥΛΙΟΣ 2025

ΠΕΡΙΛΗΨΗ

Η παρούσα διπλωματική εργασία πραγματεύεται ένα από τα σημαντικότερα προβλήματα του τομέα των μεταφορών και της επιχειρησιακής έρευνας: τη βέλτιστη δρομολόγηση οχημάτων. Στόχος είναι η μελέτη και υλοποίηση ενός αλγορίθμου που να καλύπτει τις ανάγκες μιας μικρομεσαίας τεχνικής επιχείρησης, η οποία καθημερινά καλείται να διαχειριστεί επισκευές βλαβών σε τυχαία και διασκορπισμένα σημεία εντός της πόλης της Αθήνας. Το πρόβλημα αντιμετωπίζεται με μεθοδολογίες επιχειρησιακής έρευνας και μαθηματικό προγραμματισμό, ενώ αξιολογούνται και συγκρίνονται διαφορετικά εργαλεία και τεχνικές επίλυσης, όπως οι βιβλιοθήκες PuLP και Google OR-Tools.

Η εργασία χωρίζεται σε τρία βασικά μέρη: τη δημιουργία τυχαίων σημείων βλαβών βάσει ρεαλιστικών περιορισμών, την επιλογή και εφαρμογή του καταλληλότερου αλγορίθμου για την επίλυση του προβλήματος δρομολόγησης, και τέλος, τη διαμόρφωση ενός συστήματος που υπολογίζει καθημερινά τον αριθμό και τις διαδρομές των συνεργείων. Τα σημεία διακρίνονται ανάλογα με την προτεραιότητα, τη γεωγραφική περιοχή και το είδος της βλάβης, ενώ το τελικό πρόγραμμα στοχεύει στη μέγιστη δυνατή απόδοση εντός του ωραρίου εργασίας.

Μέσα από εκτενή πειραματική ανάλυση, αξιολογούνται οι δύο προσεγγίσεις ως προς την ακρίβεια και την ταχύτητα επίλυσης, με τα αποτελέσματα να δείχνουν ότι το PuLP προσφέρει πιο βέλτιστες λύσεις για μικρό πλήθος σημείων, ενώ το OR-Tools υπερτερεί σε μεγαλύτερης κλίμακας προβλήματα λόγω ταχύτερης απόκρισης. Ο τελικός αλγόριθμος, βασισμένος στο OR-Tools, εφαρμόζεται σε πέντε διαδοχικές ημερήσιες επαναλήψεις και αποδίδει ένα αποδοτικό πλάνο δρομολόγησης, προσφέροντας πρακτικά οφέλη στην καθημερινή λειτουργία της επιχείρησης.

Η εργασία καταλήγει σε συγκεκριμένες προτάσεις για μελλοντική έρευνα, όπως η ενσωμάτωση δυναμικών δεδομένων και τεχνητής νοημοσύνης για τη συνεχή βελτίωση της δρομολόγησης και την ανάπτυξη εργαλείων παρακολούθησης σε πραγματικό χρόνο. Η προσέγγιση που παρουσιάζεται αποδεικνύει πως ακόμα και μικρές επιχειρήσεις μπορούν να επωφεληθούν από την υιοθέτηση τεχνικών βελτιστοποίησης, με απτά αποτελέσματα στην απόδοση και το κόστος λειτουργίας.

ΕΥΧΑΡΙΣΤΙΕΣ

Με την ολοκλήρωση της διπλωματικής μου εργασίας, αισθάνομαι την ανάγκη να εκφράσω την ευγνωμοσύνη μου προς όλους όσοι συνέβαλαν με τον τρόπο τους σε αυτή τη διαδρομή και στην επιτυχή περάτωση των προπτυχιακών μου σπουδών στη Σχολή Πολιτικών Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου.

Θα ήθελα πρωτίστως να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου, κ. Κωνσταντίνο Γκιοτσαλίτη, Επίκουρο Καθηγητή της Σχολής, για την εμπιστοσύνη που μου έδειξε αναθέτοντάς μου τη διπλωματική αυτή εργασία, καθώς και για τη συνεχή καθοδήγηση και υποστήριξη του καθ' όλη τη διάρκεια της εκπόνησής της.

Ευχαριστώ επίσης ιδιαίτερα τον υποψήφιο διδάκτορα του ΕΜΠ, Γιάννη Καμπούρη, για τον πολύτιμο χρόνο του, τη βοήθειά του και τις γνώσεις που μου μετέδωσε.

Τέλος, ένα μεγάλο ευχαριστώ στην οικογένειά μου, για τη διαρκή στήριξη, κατανόηση και ενθάρρυνση που μου προσέφεραν όχι μόνο κατά την περίοδο της διπλωματικής μου εργασίας, αλλά σε όλη τη διάρκεια των σπουδών μου.

Γκέλλας Φοίβος

Ιούλιος 2025

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1. Εισαγωγή	5
2. Βιβλιογραφική Ανασκόπηση	7
2.1 Γενικά Στοιχεία για τη Δρομολόγηση Οχημάτων	7
2.2 Πρόβλημα του Πλανώδιου Πωλητή – TSP	8
2.3 Πρόβλημα Δρομολόγησης Οχημάτων – VRP	10
2.4 Συνήθεις Παραλλαγές του Προβλήματος Δρομολόγησης Οχημάτων	12
2.5 Αλγόριθμος επίλυσης	14
2.6 Εξέλιξη και Μεθοδολογικές Προσεγγίσεις του Προβλήματος Δρομολόγησης Οχημάτων	16
2.7 Πίνακας Μελετών και Προσεγγίσεων για το Πρόβλημα Δρομολόγησης Οχημάτων	17
2.8 Πίνακας με του πιο Δημοφιλείς Επιλυτές (Solvers) για Προβλήματα Δρομολόγησης Οχημάτων	18
3. Πρόβλημα	19
3.1 Παρουσίαση του Προβλήματος	19
3.2 Ανάλυση Μεθοδολογίας	21
3.2.1 Δημιουργία Δεδομένων του Προβλήματος	21
3.2.2 Μαθηματική Διατύπωση του Βασικού Κώδικα Σύγκρισης	21
3.2.3 Υπολογιστικό Περιβάλλον και Τεχνική Υλοποίηση	24
3.2.4 Επισκόπηση της Εφαρμοζόμενης Μεθοδολογίας	24
3.2.5 Υλοποίηση Αλγόριθμου και Σύγκριση Επιλυτών	25

4. Επίλυση του Προβλήματος	27
4.1 Σύγκριση Επιλυτών	27
4.2 Μαθηματικό Μοντέλο για Μεγλύτερα Δίκτυα	35
4.3 Εφαρμογή Τελικού Αλγορίθμου	38
5. Αποτελέσματα - Συμπεράσματα	45
6. Επίλογος - Προτάσεις για Μελλοντική Έρευνα	47
Βιβλιογραφία	49
Παράρτημα 1	51
Παράρτημα 2	67

1. Εισαγωγή

Τις τελευταίες δεκαετίες, με τη ραγδαία εξέλιξη της τεχνολογίας, γίνεται όλο και περισσότερο αναγκαία η διαχείριση της εφοδιαστικής αλυσίδας στις επιχειρήσεις, ώστε να παραμείνουν ανταγωνιστικές στον τομέα τους. Παρότι οι μεγάλες επιχειρήσεις έχουν εντοπίσει αυτή την αναγκαιότητα και τα οφέλη που παρέχει, οι μικρές και μεσαίες επιχειρήσεις υστερούν σε αυτόν τον τομέα και στην αξιολόγηση του οφέλους που μπορούν να έχουν, όσον αφορά την καλύτερη ποιότητα των υπηρεσιών, τη μείωση του κόστους και την αύξηση της αποδοτικότητας [1], [2], [3], [20].

Το πρόβλημα της βέλτιστης δρομολόγησης είναι πρακτικό και προσφέρει άμεση και εμφανή βελτίωση στη λειτουργία κάθε εταιρείας. Οι μεγάλες εταιρείες, που καλούνται να διαχειριστούν έναν μεγάλο όγκο μεταφορών, είτε διαθέτουν ειδικό τμήμα που ασχολείται με τη μελέτη και την εξέλιξη λύσεων του προβλήματος είτε έχουν υιοθετήσει και εφαρμόσει λύσεις και αλγορίθμους που έχουν ήδη μελετηθεί. Οι μικρομεσαίες επιχειρήσεις, όμως, συνήθως χρησιμοποιούν πρακτικές - εμπειρικές μεθόδους κατά τη διεξαγωγή των καθημερινών τους δραστηριοτήτων. Με αυτόν τον τρόπο, αν και η λειτουργία τους δεν παρουσιάζει ουσιαστικά προβλήματα, δεν επιτυγχάνουν το βέλτιστο δυνατό ποσοτικό, ποιοτικό και οικονομικό αποτέλεσμα [2], [3], [4], [6].

Η επιχειρησιακή έρευνα αποτελεί ένα από τα πιο αποτελεσματικά εργαλεία στη λήψη αποφάσεων σε πολύπλοκα επιχειρησιακά περιβάλλοντα. Με την εφαρμογή της σε προβλήματα όπως αυτό της δρομολόγησης, επιτρέπει τη μοντελοποίηση ρεαλιστικών περιορισμών και την αναζήτηση της βέλτιστης λύσης με μαθηματική ακρίβεια. Στην περίπτωση της διαχείρισης συνεργείων τεχνικής εταιρείας, η επιχειρησιακή έρευνα βοηθά στην εκτίμηση και βελτιστοποίηση του χρόνου απόκρισης, της χρήσης πόρων και της γεωγραφικής κατανομής των εργασιών [4], [6], [7], [8], [12], [41].

Το πρόβλημα της βελτιστοποίησης της δρομολόγησης οχημάτων έχει μελετηθεί εκτενώς τις τελευταίες δεκαετίες, με αποτέλεσμα να υπάρχουν αλγόριθμοι επίλυσης του προβλήματος στην αγορά, για διαφόρων τύπων προβλήματα που ανταποκρίνονται στις ανάγκες κάθε εταιρείας. Είναι ένα πρόβλημα που αφορά κατά κύριο λόγο τις εταιρείες διανομής, αλλά μπορεί να εφαρμοστεί σε κάθε εταιρεία που έχει να διαχειριστεί καθημερινά έναν στόλο οχημάτων [4], [17], [29], [30], [34], [35], [39].

Η ραγδαία ανάπτυξη των υπολογιστικών δυνατοτήτων και των αλγορίθμων βελτιστοποίησης, όπως οι ευρετικοί και μεταευρετικοί αλγόριθμοι, έχει καταστήσει εφικτή την επίλυση προβλημάτων που παλαιότερα θεωρούνταν μη επίλυσιμα. Η εφαρμογή τους σε μικρομεσαίες επιχειρήσεις δεν απαιτεί πλέον εξειδικευμένες γνώσεις ή ακριβές υποδομές, γεγονός που καθιστά την τεχνολογία προσιτή και χρήσιμη για κάθε επιχειρηματική δραστηριότητα που εμπλέκει τη μεταφορά ή την παροχή υπηρεσιών στο πεδίο [16], [17], [23], [24], [40].

Είναι ένα πρόβλημα που ξεκινάει από την απλοϊκή του μορφή, που αφορά παραδόσεις από ένα σημείο σε πολλαπλά σημεία, και μπορεί να γίνει τόσο σύνθετο όσο οι ανάγκες της κάθε εταιρείας. Η επίλυση, αλλά και η εξέλιξη του προβλήματος, γίνεται με τεχνικές επιχειρησιακής έρευνας και χρήση μαθηματικού προγραμματισμού και στοχεύει στη μείωση του κόστους μεταφοράς και στην αύξηση της παραγωγικότητας και της ποιότητας των μεταφορών, άρα και σε οικονομικά οφέλη για τις επιχειρήσεις αυτές [4], [6], [8], [12], [14], [41].

Η διπλωματική αυτή εργασία φιλοδοξεί να γεφυρώσει το χάσμα μεταξύ θεωρίας και πράξης, παρουσιάζοντας ένα ολοκληρωμένο εργαλείο που μπορεί να χρησιμοποιηθεί άμεσα από μικρομεσαίες επιχειρήσεις. Μέσα από τη χρήση πραγματικών δεδομένων και την αξιολόγηση διαφόρων αλγορίθμων, προτείνεται μια λύση εφαρμόσιμη, αποδοτική και προσαρμοσμένη στις ανάγκες μιας τυπικής τεχνικής εταιρείας. Η εργασία δεν μένει σε θεωρητικό επίπεδο, αλλά προχωρά σε υλοποίηση λειτουργικού συστήματος, το οποίο μπορεί να προσαρμόζεται δυναμικά στην καθημερινή επιχειρησιακή πραγματικότητα [13], [14], [20], [25], [28].

Στόχος της διπλωματικής είναι η παρουσίαση ενός προβλήματος δρομολόγησης οχημάτων σε μία μικρομεσαία τεχνική εταιρεία, που ασχολείται με την επίλυση προβλημάτων σε έναν τομέα αρμοδιότητας (παραδείγματος χάριν, βλάβες στα δίκτυα Οργανισμών Κοινής Ωφέλειας), και πώς μπορεί, με μαθηματικό τρόπο, να γίνει η βελτιστοποίηση του προγράμματος των συνεργειών της.

Η φύση του προβλήματος που εξετάζεται στην παρούσα εργασία είναι δυναμική. Οι βλάβες εμφανίζονται σε διαφορετικά σημεία κάθε μέρα, γεγονός που καθιστά απαραίτητη τη συνεχή προσαρμογή του προγράμματος δρομολόγησης. Η μεταβλητότητα αυτή ενισχύει τη σημασία των μεθόδων που βασίζονται σε στοχαστικές τεχνικές και επαναληπτικούς αλγορίθμους, ικανούς να προσφέρουν ποιοτικές λύσεις εντός χρονικών περιορισμών. Ένα σύστημα που είναι σε θέση να αναπροσαρμόζει το πρόγραμμα καθημερινά, με βάση τα νεότερα δεδομένα, προσφέρει στην επιχείρηση ευελιξία, συνέπεια και επιχειρησιακή συνέχεια [24], [26], [27], [31], [33], [38].

2. Βιβλιογραφική Ανασκόπηση

2.1 Γενικά Στοιχεία για τη Δρομολόγηση Οχημάτων

Ο κλάδος των μεταφορών αποτελεί κρίσιμο σημείο στην ανάπτυξη της οικονομίας. Με την πρόοδο της τεχνολογίας και την παγκοσμιοποίηση, το πρόβλημα της δρομολόγησης οχημάτων (Vehicle Routing Problem - VRP), τόσο σε τοπικό όσο και σε παγκόσμιο επίπεδο, έχει έρθει στο προσκήνιο και αποτελεί έναν από τους σημαντικότερους τομείς έρευνας και εξέλιξης [4], [6]. Αυτό συμβαίνει διότι αποσκοπεί στο να προσφέρει μαθηματικές λύσεις σε καθημερινά επιχειρησιακά προβλήματα, οδηγώντας σε καλύτερα οικονομικά αποτελέσματα για τις επιχειρήσεις .

Μια πειραματική μελέτη [1] εξετάζει τη χρήση λύσεων διαχείρισης της εφοδιαστικής αλυσίδας με τη βοήθεια τεχνητής νοημοσύνης, στο πλαίσιο της Βιομηχανίας 5.0. Η ανάλυση δεδομένων που αφορούσαν αποθέματα προϊόντων, λεπτομέρειες προμηθευτών, παραγγελίες πελατών και στοιχεία μεταφοράς, έδειξε σημαντική εξοικονόμηση κόστους στη μεταφορά logistics (10%), βελτιώσεις στην αποδοτικότητα κόστους προμηθευτών (20%) και σημαντικές μειώσεις στην πλεονάζουσα αποθήκευση (10%).

Το πρόβλημα της δρομολόγησης οχημάτων (VRP) αφορά τον υπολογισμό της βέλτιστης διαδρομής διανομής προϊόντων που πρέπει να εκτελούνται από ένα σύνολο οχημάτων, ώστε να ικανοποιηθεί η ζήτηση, στοχεύοντας στην ελαχιστοποίηση του κόστους. Απαραίτητη προϋπόθεση είναι ο στόλος των οχημάτων να ξεκινά και να καταλήγει στο ίδιο σημείο (αμαξοστάσιο - αποθήκη) [4], [6], [10].

Το πρόβλημα της δρομολόγησης οχημάτων (VRP) αποτελεί γενικευμένη μορφή του προβλήματος του πλανόδιου πωλητή (Travelling Salesman Problem – TSP), το οποίο αποσκοπεί στην εύρεση της βέλτιστης διαδρομής που πρέπει να ακολουθήσει ένας πωλητής ώστε να επισκεφθεί κάθε σημείο σε ένα σύνολο και να επιστρέψει στην αφετηρία του. Η γενίκευση αυτή έχει ως αποτέλεσμα το πρόβλημα της δρομολόγησης οχημάτων να εξελίσσεται με τρόπο που όχι μόνο καλύπτει προβλήματα υπαρχουσών καταστάσεων, αλλά και καθορίζει στρατηγικές για μελλοντικές επιχειρησιακές δραστηριότητες (π.χ. δημιουργία νέων αποθηκευτικών εγκαταστάσεων) [4], [10], [22].

Όπως προαναφέρθηκε, το πρόβλημα της δρομολόγησης οχημάτων μπορεί να είναι τόσο απλό ή τόσο σύνθετο όσο απαιτείται, προκειμένου να καλύψει τις ανάγκες κάθε επιχείρησης. Αυτή η ευελιξία το καθιστά ένα δυναμικό και πολυδιάστατο πρόβλημα, το οποίο όμως βασίζεται σε ένα σαφές θεωρητικό υπόβαθρο [4], [6], [8], [12].

2.2 Πρόβλημα του Πλανώδιου Πωλητή - TSP

Η πρώτη συστηματική μελέτη του προβλήματος του Πλανώδιου Πωλητή (Travelling Salesman Problem - TSP) έγινε τη δεκαετία του 1930, αλλά καθιερώθηκε ακαδημαϊκά τη δεκαετία του 1950, κυρίως μέσω της εργασίας του Merrill Flood (1956). Ωστόσο, η επίσημη επιστημονική διατύπωση και η συστηματική μαθηματική μελέτη του προβλήματος αποδίδεται στους Dantzig, Fulkerson και Johnson το 1954, όταν ανέπτυξαν μια προσέγγιση μέσω γραμμικού προγραμματισμού [4], [10], [22].

Το πρόβλημα του Πλανώδιου Πωλητή αφορά έναν πωλητή που ξεκινά από μια αφετηρία A, επισκέπτεται ένα σύνολο πόλεων μία και μόνο φορά την καθεμία και επιστρέφει στην αφετηρία, εκτελώνοντας τη συντομότερη δυνατή διαδρομή.

Το πρόβλημα του Πλανώδιου Πωλητή μπορεί να μοντελοποιηθεί ως εξής:

Η αντικειμενική συνάρτηση που πρέπει να ελαχιστοποιηθεί:

$$\min \sum_{i \in N} \sum_{j \in N \setminus \{i\}} c_{ij} x_{ij} \quad (2.2.1)$$

Ορίζουμε τις μεταβλητές απόφασης x_{ij} :

$$x_{ij} = \begin{cases} 1, & \text{εάν το τόξο } (i, j) \text{ χρησιμοποιείται} \\ 0 & \text{στην αντίθετη περίπτωση} \end{cases} \quad (2.2.2)$$

Οι περιορισμοί που διασφαλίζουν τη διατήρηση της ροής, ότι δηλαδή σε κάθε πελάτη θα γίνει επίσκεψη μόνο 1 φορά είναι οι:

$$\sum_{j \in N \setminus \{i\}} x_{ij} = 1 \quad \forall i \in N \quad (2.2.3)$$

$$\sum_{i \in N \setminus \{j\}} x_{ij} = 1 \quad \forall j \in N \quad (2.2.4)$$

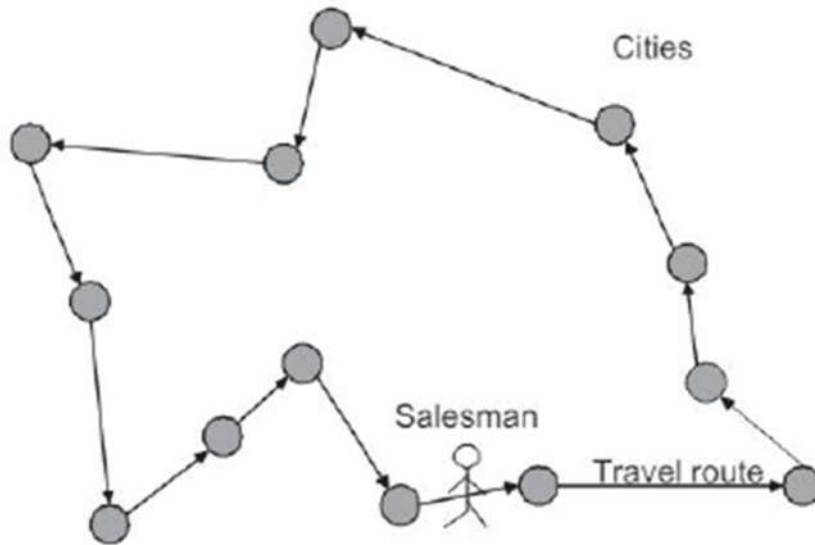
Όπου, η εξίσωση (2.2.3) εξασφαλίζει ότι κατά την επίσκεψη στην πόλη i , ο πωλητής πρέπει να επισκεφτεί μόνο μια πόλη μετά, η εξίσωση (2.2.4) εξασφαλίζει ότι ο πωλητής προέρχεται από μια μόνο πόλη.

Υπάρχουν διάφοροι τρόποι να μοντελοποιήσουμε τους περιορισμούς που διασφαλίζουν ότι δεν θα δημιουργηθούν υποδιαδρομές. Ένας από αυτούς είναι η μεθοδολογία Miller-Tucker-Zemlin (MTZ) που περιλαμβάνει τις παρακάτω εξισώσεις [10], [22]:

$$(n - 1)x_{ij} + u_i - u_j \leq n - 2 \quad \forall i, j \in \{2, 3, \dots, n\} \mid i \neq j \quad (2.2.5)$$

$$1 \leq u_i \leq n - 1 \quad \forall i \in \{2, 3, \dots, n\} \quad (2.2.6)$$

$$u_i \in \mathbb{Z} \quad \forall i \in \{2, 3, \dots, n\} \quad (2.2.7)$$



Εικόνα 1 - TSP

2.3 Πρόβλημα Δρομολόγησης Οχημάτων - VRP

Το βασικό πρόβλημα δρομολόγησης οχημάτων (Vehicle Routing Problem - VRP) αναφέρεται σε ένα σύνολο οχημάτων που ξεκινούν και καταλήγουν στο ίδιο σημείο και διέρχονται από ένα σύνολο σημείων με συγκεκριμένη ζήτηση (δηλαδή πελάτες της εταιρείας). Στη βασική του μορφή, το πρόβλημα δεν περιλαμβάνει περιορισμούς, όπως χρόνοι παράδοσης, προτεραιότητες κ.λπ [4], [6], [10].

Ο στόχος είναι η εύρεση της αποδοτικότερης διαδρομής που θα εκτελέσει κάθε όχημα, έτσι ώστε το σύνολο των οχημάτων να καλύψει τις ανάγκες των πελατών, διανύοντας τη μικρότερη δυνατή απόσταση. Κάθε όχημα έχει συγκεκριμένη χωρητικότητα και κάθε σημείο έχει καθορισμένη ζήτηση. Συνεπώς, το ζητούμενο είναι η εύρεση των βέλτιστων διαδρομών, ώστε ο ελάχιστος αριθμός οχημάτων να εκτελέσει τις απαιτούμενες διαδρομές με τη μικρότερη δυνατή συνολική απόσταση, εντός ορισμένου χρονικού πλαισίου, καλύπτοντας πλήρως τις απαιτήσεις των πελατών [4], [11], [13], [17], [24].

Το VRP είναι ένα NP-Hard πρόβλημα (Non-deterministic Polynomial time – NP), δηλαδή ένα πρόβλημα μη πολυωνυμικής δυσκολίας, γεγονός που σημαίνει πως η πολυπλοκότητά του αυξάνεται εκθετικά σε σχέση με το μέγεθος του προβλήματος. Όταν ο αριθμός των σημείων είναι μικρός, μπορούν να εφαρμοστούν “ακριβείς” αλγόριθμοι (exact algorithms). Ωστόσο, για προβλήματα με δεδομένα πραγματικού κόσμου, χρησιμοποιούνται ευρετικοί (heuristic) αλγόριθμοι, προσαρμοσμένοι κάθε φορά στις ιδιαιτερότητες του προβλήματος. Σε τέτοιες περιπτώσεις συχνά απαιτείται απλοποίηση των δεδομένων, ώστε να προκύψει μία προσεγγιστική αλλά ικανοποιητική λύση [6], [15], [17], [23], [24], [40].

Το πρόβλημα μπορεί να διατυπωθεί μαθηματικά ως εξής [4], [6], [8], [10], [22]:

Το πρόβλημα της δρομολόγησης οχημάτων μπορεί να οριστεί ως ένα πλήρες κατευθυνόμενο γράφημα $G = (V, A)$. Με τα σημεία V ορίζονται κορυφές, με το σημείο 0 να ορίζεται ως αποθήκη και τα υπόλοιπα σημεία ως οι υπόλοιποι πελάτες. Για το A έχουμε $A = \{(i, j) : i, j \in V, i \neq j\}$, που το κάθε τόξο (i, j) ορίζει τη διαδρομή που ακολουθεί το όχημα για να πάει στο σημείο j από το σημείο i . Το γράφημα είναι κατευθυνόμενο, κάτι που σημαίνει ότι $(i, j) \neq (j, i)$. Κάθε διαδρομή έχει συγκεκριμένο κόστος c_{ij} , που ορίζεται ανάλογα με τις ανάγκες του προβλήματος και μπορεί να εξαρτάται από τα χιλιόμετρα, τα καύσιμα, το χρόνο διέλευσης του οχήματος ή οποιαδήποτε άλλη μεταβλητή μπορεί να μεταφραστεί ως ποσότητα κόστους.

Ως διαδρομή ορίζεται ως η ακολουθία $(i_0, i_1, i_2, \dots, i_s, i_{s+1})$ με το $i_0 = i_{s+1}$ να ορίζει την αποθήκη που είναι το σημείο έναρξης και λήξης των διαδρομών και το σετ $S = \{i_1, i_2, \dots, i_s\}$ να ορίζει τους πελάτες που εξυπηρετούνται από τις διαδρομές r . Το συνολικό κόστος μιας διαδρομής είναι το σύνολο του κόστους των επι μέρους διαδρομών που ορίζονται από τα τόξα (i, j)

$$c(r) = \sum_{p=0}^s c_{ip,ip+1} \quad (2.3.1)$$

Στόχος του VRP είναι να καθοριστεί ένα σύνολο διαδρομών ελάχιστου κόστους και μεγέθους $< |K|$, όπου ως K ορίζεται το σύνολο των διαθέσιμων οχημάτων, οι οποίες θα ικανοποιούν τις απαιτήσεις όλων των πελατών.

Αν θεωρήσουμε ότι $S \subseteq V \setminus \{0\}$, για κατευθυνόμενους γράφους ορίζουμε τα εντός-τόξων και εκτός-τόξων ως $\delta^-(S) = \{(i, j) \in A : i \notin S, j \in S\}$ και $\delta^+(S) = \{(i, j) \in A : i \in S, j \notin S\}$.

Ορίζουμε τις μεταβλητές απόφασης x_{ij} :

$$x_{ij} = \begin{cases} 1 & \text{εάν το } (i, j) \text{ είναι μέρος της λύσης} \\ 0 & \text{στην αντίθετη περίπτωση} \end{cases}$$

Η αντικειμενική συνάρτηση που πρέπει να ελαχιστοποιηθεί:

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} \quad (2.3.2)$$

$$\sum_i x_{ij} = 1 \quad \forall i \in V \quad (2.3.3)$$

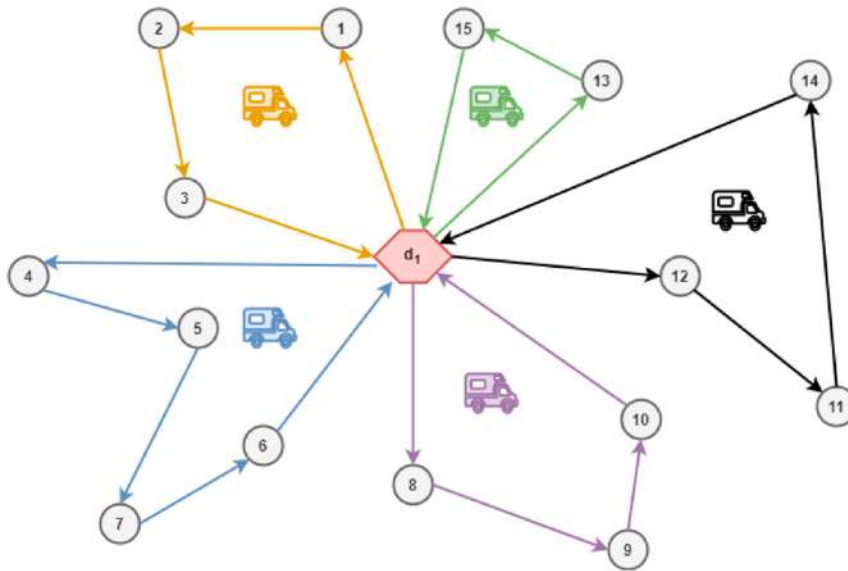
$$\sum_j x_{ij} = 1 \quad \forall j \in V \quad (2.3.4)$$

$$\sum_j x_{0j} \leq |K| \quad (2.3.5)$$

$$\sum_{(i,j)} x_{ij} \geq r(S) \quad \forall S \subseteq V \quad (2.3.6)$$

Οι περιορισμοί (2.3.3) και (2.3.4) διασφαλίζουν τη διατήρηση της ροής, ότι δηλαδή σε κάθε πελάτη θα γίνει επίσκεψη μόνο 1 φορά. Για κάθε κόμβο i , η λύση πρέπει να περιλαμβάνει μόνο ένα τόξο που θα μπαίνει και ένα τόξο που θα βγαίνει. Ο περιορισμός (2.3.5) εισάγει τον αριθμό των οχημάτων και ο περιορισμός (2.3.6) εισάγει τον περιορισμό χωρητικότητας αλλά και τον περιορισμό ότι δεν θα δημιουργηθούν υποδιαδρομές.

Για την τελική λύση του VRP, δηλαδή, θα πρέπει το κάθε όχημα να ξεκινάει και να καταλήγει στο σημείο 0 που ορίσαμε ως αποθήκη. Η διέλευση από κάθε σημείο $i \in V/\{0\}$ θα πρέπει να είναι μια και μοναδική. Και το τελικό σύνολο των διαδρομών θα πρέπει να μην είναι μεγαλύτερο από τον αριθμό των διαθέσιμων οχημάτων.



Εικόνα 2 VRP – CVRP

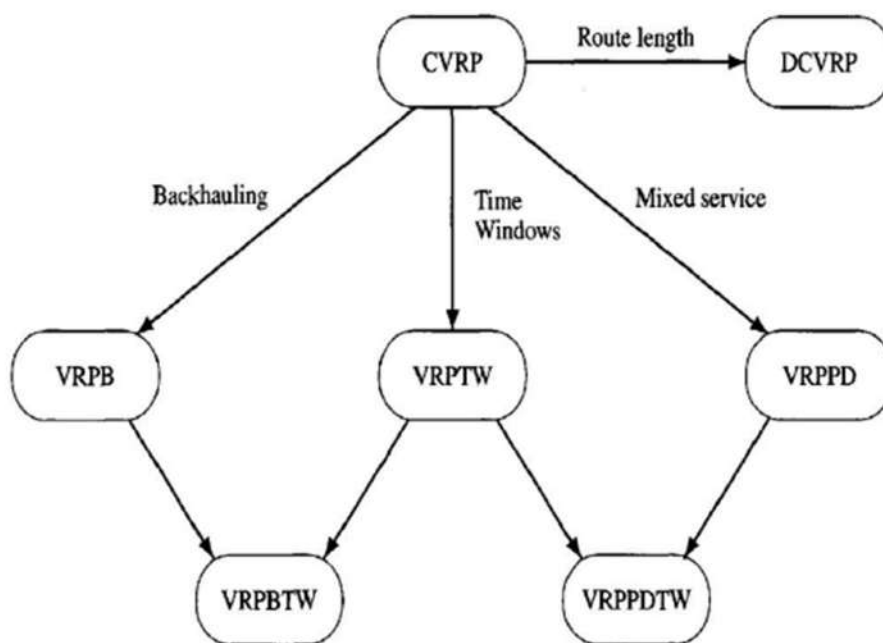
2.4 Συνήθεις Παραλλαγές του Προβλήματος Δρομολόγησης Οχημάτων

Ξεκινώντας από τη βασική μορφή του VRP, όπως αυτή έχει διατυπωθεί, έχουν προκύψει παραλλαγές που αντανακλούν τον βαθμό πολυπλοκότητας κάθε προβλήματος στην εφοδιαστική αλυσίδα. Η πρόοδος της τεχνολογίας, σε συνδυασμό με την αύξηση των απαιτήσεων — τόσο χρονικών όσο και ποιοτικών και ποσοτικών — έχει οδηγήσει στην εξέλιξη του προβλήματος με την προσθήκη περιορισμών στο αρχικό του πλαίσιο [4], [6], [29], [32], [35].

Ως αποτέλεσμα, η μελέτη της διαχείρισης της εφοδιαστικής αλυσίδας έχει μετατραπεί από ένα απλό βελτιστοποιητικό πρόβλημα σε έναν ολόκληρο επιστημονικό και επιχειρησιακό τομέα, ο οποίος επηρεάζει άμεσα τη λειτουργία και την αποδοτικότητα χιλιάδων επιχειρήσεων [2], [3], [20], [36].

Κάποιες βασικές παραλλαγές του VRP ανάλογα με τους περιορισμούς παρουσιάζονται στον παρακάτω πίνακα:

Ακρώνυμο	Επεξήγηση
CVRP	Πρόβλημα δρομολόγησης με περιορισμένη χωρητικότητα
DCVRP	Πρόβλημα δρομολόγησης με περιορισμένη χωρητικότητα και απόσταση
VRPTW	Πρόβλημα δρομολόγησης με χρονικούς περιορισμούς
VRPB	Πρόβλημα δρομολόγησης με επιστροφές
VRPPD	Πρόβλημα δρομολόγησης με παραλαβή και παράδοση
VRPBTW	Πρόβλημα δρομολόγησης με backhands και χρονικούς περιορισμούς
VRPPDTW	Πρόβλημα δρομολόγησης με παραλαβή και παράδοση και μηχανικούς περιορισμούς για διανομή εμπορευμάτων



Εικόνα 3 Παραλλαγές VRP και οι σχέσεις μεταξύ τους (πηγή: Toth και Vigo 2002)

Η πιο συνήθης παραλλαγή του προβλήματος δρομολόγησης οχημάτων (VRP) είναι το CVRP (Capacitated Vehicle Routing Problem – Πρόβλημα Δρομολόγησης με Περιορισμένη Χωρητικότητα). Η βασική διαφορά τους είναι η εισαγωγή ενός επιπλέον περιορισμού: της συγκεκριμένης χωρητικότητας του κάθε οχήματος. Αυτό καθιστά το CVRP ιδιαίτερα χρήσιμο για επιχειρήσεις που μεταφέρουν αγαθά, καθώς αντανακλά πιο ρεαλιστικά τις επιχειρησιακές συνθήκες.

Το πρόβλημα του CVRP μπορεί να οριστεί ως εξής: Μια εταιρεία μεταφορών διαθέτει έναν στόλο K οχημάτων και καλείται να εξυπηρετήσει τις ανάγκες ενός πλήθους πελατών V . Κάθε όχημα έχει περιορισμένη χωρητικότητα Q , ενώ κάθε πελάτης έχει ζήτηση d_i για κάποιο αγαθό. Ο στόχος είναι να προσδιοριστεί η βέλτιστη διαδρομή για κάθε όχημα, έτσι ώστε:

- κάθε πελάτης να εξυπηρετείται μία και μόνο φορά,
- κάθε όχημα να ξεκινά και να καταλήγει στην ίδια αφετηρία (αποθήκη),
- το φορτίο του κάθε οχήματος να μην υπερβαίνει τη χωρητικότητά του,
- και το συνολικό κόστος (συχνά σε όρους απόστασης ή χρόνου) να είναι ελάχιστο.

Με τον τρόπο αυτό, επιδιώκεται να χρησιμοποιηθεί ο ελάχιστος δυνατός αριθμός οχημάτων και να ελαχιστοποιηθεί η συνολική απόσταση ή το κόστος μεταφοράς, ενώ καλύπτονται πλήρως οι ανάγκες όλων των πελατών [4], [11], [17], [29].

2.5 Αλγόριθμοι Επίλυσης

Η αύξηση και διαφοροποίηση των παραμέτρων σε κάθε πρόβλημα δρομολόγησης έχει οδηγήσει την ερευνητική κοινότητα στην ανάπτυξη «ακριβών» αλγορίθμων (exact algorithms), οι οποίοι στοχεύουν στον εντοπισμό της συνολικά βέλτιστης λύσης μέσω εξαντλητικού ελέγχου όλων των πιθανών διαδρομών. Οι ακριβείς αλγόριθμοι έχουν τη δυνατότητα να εγγυηθούν τη βέλτιστη λύση ενός προβλήματος. Ωστόσο, η υπολογιστική πολυπλοκότητά τους συχνά τους καθιστά απαγορευτικούς για εφαρμογή σε προβλήματα πραγματικής κλίμακας [6], [10], [22], [41].

Για την αντιμετώπιση προβλημάτων που αντανακλούν πραγματικές επιχειρησιακές ανάγκες, προτιμώνται ευρετικοί αλγόριθμοι (heuristic algorithms). Οι ευρετικοί αλγόριθμοι στοχεύουν στην εύρεση αποδεκτών λύσεων σε σύντομο χρονικό διάστημα, ακόμα και όταν το πρόβλημα δεν είναι εφικτό να λυθεί επακριβώς με κλασικές μεθόδους. Αν και δεν εγγυώνται τη συνολικά βέλτιστη λύση, έχουν τη δυνατότητα να προσεγγίσουν την πραγματικότητα και να παρέχουν λειτουργικές και άμεσα εφαρμόσιμες λύσεις σε πραγματικά σενάρια [17], [23], [29], [40].

Οι ευρετικοί αλγόριθμοι δεν καταλήγουν απαραίτητα στη «μία και μοναδική βέλτιστη λύση», αλλά σε μια λύση επαρκώς καλή για τις ανάγκες των επιχειρήσεων. Αυτός είναι και ο λόγος για τον οποίο προτιμώνται, καθώς οι ακριβείς αλγόριθμοι — όπως οι Branch-and-Bound (B&B) ή Branch-and-Cut (BCP) — απαιτούν μεγάλο υπολογιστικό χρόνο και πόρους, γεγονός που τους καθιστά μη οικονομικά αποδοτικούς σε καθημερινή χρήση [6], [17], [23], [40].

Αντίθετα, οι ευρετικοί αλγόριθμοι μπορούν να ενσωματώσουν τους περιορισμούς κάθε προβλήματος και να δώσουν άμεσα εφαρμόσιμες λύσεις, καλύπτοντας λειτουργικές ανάγκες της καθημερινότητας σε τομείς όπως η εφοδιαστική, οι μεταφορές και η διανομή προϊόντων [17], [23], [24], [40].

Οι ευρετικές τεχνικές μπορούν να διακριθούν σε τρεις βασικές κατηγορίες:

1. **Εποικοδομητικά ευρετικά (Constructive Heuristics):**

Αυτές οι τεχνικές ξεκινούν από κενά δεδομένα και, μέσω επαναληπτικής διαδικασίας, προσθέτουν στοιχεία στη λύση έως ότου ολοκληρωθεί μία εφικτή λύση. Η διαδικασία τερματίζεται όταν δεν είναι πλέον δυνατή η προσθήκη νέων στοιχείων χωρίς παραβίαση των περιορισμών του προβλήματος.

2. **Ευρετικά βελτίωσης (Improvement Heuristics):**

Ξεκινούν από μια αρχική (συνήθως αποδεκτή) λύση και επιχειρούν να τη βελτιώσουν σταδιακά μέσω τοπικών ή παγκόσμιων τροποποιήσεων. Στόχος είναι η αύξηση της ποιότητας της λύσης χωρίς να απαιτείται επανασχεδίαση από το μηδέν.

3. **Μετά-ευρετικές τεχνικές (Metaheuristics):**

Σύμφωνα με τους Sørensen και Glover (2013), οι μετά-ευρετικές τεχνικές αποτελούν αλγοριθμικά πλαίσια υψηλού επιπέδου που δεν έχουν σχεδιαστεί για συγκεκριμένα προβλήματα, αλλά προσφέρουν ένα σύνολο στρατηγικών και κατευθυντήριων γραμμών για την ανάπτυξη ευρετικών αλγορίθμων βελτιστοποίησης.

Οι μετά-ευρετικές τεχνικές διακρίνονται σε δύο βασικές υποκατηγορίες:

1. Τεχνικές βασισμένες σε μία λύση (Single-Solution Based Methods):

Αυτές οι τεχνικές εξετάζουν και βελτιώνουν μία μόνο λύση κάθε φορά, με στόχο τη σταδιακή προσέγγιση του βέλτιστου.

- Περιορισμένης αναζήτησης (Tabu Search)
- Προσομοίωση ανόπτησης (Simulated Annealing)
- Επαναλαμβανόμενη τοπική αναζήτηση (Iterated Local Search)
- Αναζήτηση μεγάλης γειτονιάς (Large Neighborhood Search)

2. Τεχνικές βασισμένες σε πληθυσμό (Population-Based Methods):

Σε αυτές τις μεθόδους διατηρείται και εξελίσσεται ένας πληθυσμός λύσεων, με στόχο τη διερεύνηση πολλαπλών περιοχών του χώρου αναζήτησης ταυτόχρονα.

- Γενετικοί αλγόριθμοι (Genetic Algorithms)
- Αλγόριθμοι βελτιστοποίησης αποικίας μυρμηγκιών (Ant Colony Optimization)
- Μεμετικοί αλγόριθμοι (Memetic Algorithms)

2.6 Εξέλιξη και Μεθοδολογικές Προσεγγίσεις του Προβλήματος Δρομολόγησης Οχημάτων

Η μελέτη του προβλήματος δρομολόγησης οχημάτων (Vehicle Routing Problem - VRP) ξεκινάει από τις αρχές της δεκαετίας του 1950, με την εργασία των Dantzig et al. (1954), οι οποίοι εισήγαγαν το πρόβλημα του πλανόδιου πωλητή (TSP) και το αντιμετώπισαν μέσω γραμμικού προγραμματισμού. Η μεθοδολογική τους προσέγγιση, με χρήση του Simplex και των Cutting Planes, αποτέλεσε τη βάση για τις μελλοντικές διατυπώσεις του VRP και των παραλλαγών του. Αργότερα, οι Lenstra και Rinnooy Kan (1981) ανέδειξαν τη θεωρητική δυσκολία του προβλήματος, αποδεικνύοντας ότι πολλές μορφές του VRP είναι NP-hard, γεγονός που καθορίζει τα όρια των δυνατών ακριβών λύσεων [6], [10].

Σημαντική πρόοδος σημειώθηκε με την εργασία των Toth και Vigo (2002), οι οποίοι συγκέντρωσαν και ανέλυσαν εκτενώς τόσο το βασικό πρόβλημα VRP όσο και παραλλαγές του, όπως το πρόβλημα χωρητικότητας (CVRP), το VRP με χρονικά παράθυρα (VRPTW) και το VRP με παραλαβές και παραδόσεις (VRPPD). Χρησιμοποιώντας μεθόδους ακέραιου γραμμικού προγραμματισμού και τεχνικές όπως Branch & Bound και Branch & Cut, έθεσαν τις βάσεις για σύγχρονες ακριβείς λύσεις. Η εργασία των Kara και Bektas (2005) εμπλούτισε περαιτέρω το πεδίο, εισάγοντας το VRP με ελάχιστο φορτίο ανά διαδρομή, μια προσέγγιση που αποτυπώνει καλύτερα επιχειρησιακές απαιτήσεις, μέσω μοντέλων μικτού ακέραιου προγραμματισμού [4], [11], [22].

Στο πεδίο των ευρετικών μεθόδων, οι Pisinger και Ropke (2007) εισήγαγαν την τεχνική Adaptive Large Neighborhood Search (ALNS), μια γενικευμένη ευρετική ικανή να προσαρμόζεται σε πολλές παραλλαγές του VRP, παρέχοντας υψηλής ποιότητας λύσεις για μεγάλης κλίμακας προβλήματα. Παράλληλα, οι Cormen et al. (2009) και Kleinberg & Tardos (2005), αν και δεν επικεντρώθηκαν στο VRP ειδικά, συνέβαλαν σημαντικά μέσω της παρουσίασης θεμελιωδών αλγορίθμων όπως greedy και dynamic programming, που αξιοποιούνται ευρέως σε λύσεις VRP [8], [12], [17], [23], [24].

Τέλος, πιο πρόσφατες εργασίες έχουν επικεντρωθεί σε εξειδικευμένες εφαρμογές και πρακτικά ζητήματα. Ο Stenger (2012) μελέτησε προβλήματα VRP με μη γραμμικά κόστη και τη χρήση υπεργολάβων, προτείνοντας υβριδικούς αλγορίθμους. Οι Desaulniers et al. (2016) εισήγαγαν μοντέλα set-partitioning ειδικά προσαρμοσμένα για στόλους ηλεκτρικών οχημάτων, ενώ ο Gkiotsalitis (2022) ασχολήθηκε με τη βελτιστοποίηση δρομολογίων σε δημόσιες συγκοινωνίες με χρήση προσομοίωσης και ακέραιου προγραμματισμού. Τέλος, οι Sahin και Yaman (2023) εξέτασαν το VRP υπό περιορισμούς πρόσβασης, προτείνοντας νέους περιορισμούς τύπου path elimination για την εξασφάλιση ρεαλιστικότερων λύσεων [5], [7], [9], [24], [29], [41].

2.7 Πίνακας Μελετών και Προσεγγίσεων για το Πρόβλημα Δρομολόγησης Οχημάτων

Ο Πίνακας που ακολουθεί συνοψίζει βασικές μελέτες και προσεγγίσεις που έχουν διαμορφώσει την επιστημονική κατανόηση και την υπολογιστική επίλυση του προβλήματος δρομολόγησης οχημάτων (Vehicle Routing Problem - VRP). Περιλαμβάνει τόσο κλασικές όσο και πιο πρόσφατες εργασίες, καλύπτοντας μια ευρεία γκάμα παραλλαγών του προβλήματος, μεθόδων διατύπωσης και τεχνικών επίλυσης — από ακριβείς αλγορίθμους έως ευρετικές και υβριδικές προσεγγίσεις.

Συγγραφείς / Έτος	Πρόβλημα	Μαθηματική Διατύπωση	Μέθοδος Επίλυσης
Dantzig et al. (1954)	TSP, προάγγελος του VRP	Γραμμικός προγραμματισμός	Simplex – Cutting Planes
Lenstra & Rinnooy Kan (1981)	Πολυπλοκότητα VRP και σχετικών προβλημάτων	Θεωρία υπολογιστικής πολυπλοκότητας	Απόδειξη NP-hardness
Toth & Vigo (2002)	Βασικό VRP και παραλλαγές του (CVRP, VRPTW, VRPPD)	Ακέραιος γραμμικός προγραμματισμός	Branch & Bound, Branch & Cut
Kara & Bektas (2005)	VRP με ελάχιστο φορτίο ανά διαδρομή	Μικτό ακέραιο προγραμματισμό	Exact solution με περιορισμούς φορτίου
Pisinger & Ropke (2007)	Γενική ευρετική για VRP	Ευρετική προσέγγιση	Adaptive Large Neighborhood Search (ALNS)
Cormen et al. (2009), Kleinberg & Tardos (2005)	Θεμελιώδεις αλγόριθμοι	-	Χρήση σε VRP: greedy, dynamic programming κ.ά.
Stenger (2012)	VRP με μη γραμμικό κόστος και subcontractors	Συνδυαστικό μοντέλο	Υβριδικός αλγόριθμος δρομολόγησης
Desaulniers et al. (2016)	Set-partitioning μοντέλα για ηλεκτρικά οχήματα	Set-partitioning model	Branch-price-and-cut
Gkiotsalitis (2022)	Βελτιστοποίηση δρομολογίων σε δημόσιες συγκοινωνίες	Integer programming / Simulation	Ευρετικοί και ακριβείς αλγόριθμοι
Sahin & Yaman (2023)	VRP με περιορισμούς πρόσβασης	Μικτό ακέραιο προγραμματισμό	Εξειδικευμένοι περιορισμοί τύπου path elimination

2.8 Πίνακας με του πιο Δημοφιλείς Επιλυτές (Solvers) για Προβλήματα Δρομολόγησης Οχημάτων

Ο παρακάτω πίνακας παρουσιάζει τους πιο διαδεδομένους επιλυτές (solvers) που χρησιμοποιούνται για την επίλυση προβλημάτων δρομολόγησης οχημάτων (VRP) και των παραλλαγών τους. Περιλαμβάνει τόσο εμπορικά όσο και ανοικτού κώδικα εργαλεία, με διαφορετικές μεθοδολογικές προσεγγίσεις — από ακριβείς αλγορίθμους γραμμικού και ακέραιου προγραμματισμού έως ευρετικές και μετα-ευρετικές τεχνικές. Η επιλογή του κατάλληλου επιλυτή εξαρτάται από τη φύση του προβλήματος, τις απαιτήσεις σε ακρίβεια και τον διαθέσιμο υπολογιστικό χρόνο.

Solver / Βιβλιοθήκη	Υποστηριζόμενα Προβλήματα	Τύπος Μεθόδου	Γλώσσα / Υλοποίηση
PuLP + MTZ (Miller–Tucker–Zemlin)	TSP, CVRP	Ακέραιος γραμμικός προγραμματισμός με MTZ περιορισμούς	Python (PuLP + CBC/GLPK)
Google OR-Tools	TSP, VRP, CVRP, VRPTW	Ευρετικοί + exact (Routing Solver)	C++, Python, Java
Gurobi	TSP, VRP, CVRP, και γενικά MIP	Ακριβής – εμπορικός λύτης MIP	Python, C++, Java, κ.ά.
CPLEX	TSP, VRP, CVRP	Ακριβής – εμπορικός λύτης MIP/LP/CP	Python, Java, C++, C#
SCIP	TSP, CVRP	Ακριβής – MIP/CP hybrid	C, διαθέσιμο μέσω Python API
Concorde TSP Solver	TSP (μόνο)	Ακριβής – cutting plane method	C (με υποστήριξη για TSPLIB instances)
LKH (Lin–Kernighan–Helsgaun)	TSP, ATSP, CVRP (με τροποποιήσεις)	Ευρετικός – Iterated local search	C
jsprit	VRP, CVRP, VRPTW	Ευρετικός – metaheuristics	Java
HGSADC (Hybrid Genetic Search)	VRP, CVRP	Μετα-ευρετικός – genetic + local search	C++
VRPH	VRP παραλλαγές	Ευρετικός – Large Neighborhood Search	C++

3. ΠΡΟΒΛΗΜΑ

3.1 Παρουσίαση του Προβλήματος

Η παρούσα εργασία αναφέρεται σε μια μικρομεσαία τεχνική εταιρεία που ασχολείται με έργα επισκευής βλαβών σε δίκτυα κοινής ωφέλειας στην πόλη των Αθηνών. Τα σημεία αυτά είναι τυχαία και διασκορπισμένα σε διάφορα σημεία.

Η εταιρεία μπορεί να χρησιμοποιεί καθημερινά από ένα έως τρία συνεργεία επισκευής, τα οποία δρουν ανεξάρτητα. Η αρχή και το τέλος της καθημερινής τους διαδρομής είναι η αμαξοστάσιο της εταιρείας (Μέγαρο Μουσικής). Ο στόχος της παρούσας διπλωματικής εργασίας είναι η δημιουργία ενός προγράμματος, το οποίο θα ορίζει την καθημερινή βέλτιστη διαδρομή των συνεργείων, ώστε να έχουν τη μέγιστη δυνατή απόδοση, εντός των περιορισμών που υπάρχουν για την εύρυθμη εκτέλεση των αναγκών του έργου. Οι ώρες εργασίας των συνεργείων ορίζονται από 7:00 – 15:00 κάθε μέρα.

Για τις βλάβες υπάρχει καθημερινή προσθήκη και αφαίρεση σημείων (αντιστοιχούν στις δοθείσες από την υπηρεσία και τις εκτελεσμένες εντολές). Με τη λήψη των νέων εντολών, οι βλάβες αυτές κατηγοριοποιούνται με βάση την προτεραιότητα που πρέπει να λάβουν, την περιοχή που βρίσκονται στον χάρτη, καθώς και το είδος της εργασίας που πρέπει να εκτελεστεί. Κάθε σημείο αντιστοιχεί σε μία και μοναδική εργασία.

Συνεπώς, έχουμε σημεία A1 υψηλής προτεραιότητας, που ορίζουν βλάβες οι οποίες έχουν ως αποτέλεσμα τη δημιουργία προβλημάτων στην εύρυθμη λειτουργία των αναγκών της καθημερινότητας και πρέπει να αποκατασταθούν άμεσα. Ο χρόνος για την αντιμετώπισή τους μπορεί να είναι 20, 40 ή 75 λεπτά. Είναι τα σημεία με τη μέγιστη προτεραιότητα και πρέπει να εντάσσονται πάντα στο πρόγραμμα.

Ως σημεία A2 ορίζονται οι βλάβες μέτριας προτεραιότητας, και ο χρόνος αντιμετώπισής τους είναι 40 ή 75 λεπτά, με την αναλογία να είναι 70%-30% μεταξύ τους. Τέλος, ως σημεία A3 ορίζονται οι βλάβες χαμηλής προτεραιότητας και έχουν χρόνο αντιμετώπισης 20 λεπτά.

Για την κατηγοριοποίηση με βάση την περιοχή που βρίσκονται στον χάρτη, ορίζουμε περιοχές αυξημένης προτεραιότητας, οι οποίες έχουν μεγάλη κίνηση οχημάτων και πεζών και αυτό έχει ως αποτέλεσμα την αύξηση του κινδύνου ύπαρξης ατυχήματος.

Στη συγκεκριμένη περίπτωση, ορίζουμε ως κρίσιμη περιοχή B1 την περιοχή γύρω από το κέντρο της Αθήνας (κύκλος ακτίνας 2,5 χιλιομέτρων γύρω από την Πλατεία Συντάγματος) και ως περιοχή B2 το υπόλοιπο τμήμα του τομέα αρμοδιότητας. Ορίζουμε σαν μέγιστο χρόνο εργασίας στην κρίσιμη ζώνη τις 3 ώρες.

Για την κατηγοριοποίηση με βάση το είδος της εργασίας, διαχωρίζουμε τις εργασίες σε κατηγορίες Γ1, Γ2 και Γ3. Ως Γ1 ορίζονται οι εκτεταμένες βλάβες που χρειάζονται μεγαλύτερο χρόνο αποκατάστασης, τον οποίο ορίζουμε στα 75 λεπτά. Ως Γ2

ορίζονται βλάβες μεσαίου μεγέθους που χρειάζονται χρόνο αποκατάστασης 40 λεπτά, ενώ ως Γ3 ορίζονται οι μικρές βλάβες που χρειάζονται 20 λεπτά.

Τα μεγέθη είναι ορισμένα εμπειρικά και αφορούν βλάβες που μπορούν να αποκατασταθούν άμεσα και δεν χρειάζονται μεγάλη κινητοποίηση συνεργείων και μηχανημάτων (π.χ. κομμένο καλώδιο ΔΕΗ, ΟΤΕ ή διαρροή σε φρεάτια ΕΥΔΑΠ). Βλάβες που απαιτούν συνδρομή μηχανημάτων χρειάζονται διαφορετικού τύπου οργάνωση και δεν περιλαμβάνονται στην παρούσα εργασία.

Για τις ανάγκες της εργασίας θα πρέπει να γίνουν κάποιες παραδοχές, οι οποίες όμως έχουν βάση την πραγματικότητα. Έτσι, θεωρούμε ότι το σύνολο των σημείων που θα υπάρχουν στον χάρτη κάθε χρονική στιγμή δεν θα ξεπερνά τα 90, ενώ τα σημεία Α2 δεν θα ξεπερνούν τα 45.

Οι νέες εντολές που θα λαμβάνονται καθημερινά θα δίνονται από πρόγραμμα (randomizer). Το πρόγραμμα θα δίνει καθημερινά 4 σημεία Α1, 12–20 σημεία Α2 και 10 σημεία Α3.

Το τελικό πρόγραμμα κάθε συνεργείου θέλουμε να έχει συνολική διάρκεια 7 ωρών καθαρής εργασίας. Θεωρούμε πως ο χρόνος μετάβασης από σημείο σε σημείο συμπεριλαμβάνεται στον καθορισμένο χρόνο των κατηγοριών Γ1, Γ2 και Γ3.

Για την επιλογή σημείων ξεκινάμε από τα Α1.

Οι περιοχές Β1 και Β2 έχουν την ίδια προτεραιότητα λόγω της περιορισμένης έκτασης του προβλήματος σε σημεία και οχήματα.

Το πρόγραμμα θα εκτελείται κατόπιν εντολής, αφού προηγηθεί η αφαίρεση και η προσθήκη σημείων, και το αποτέλεσμα θα είναι το πρόγραμμα του κάθε συνεργείου για την επόμενη ημέρα.

3.2 Ανάλυση Μεθοδολογίας

3.2.1 Δημιουργία Δεδομένων του Προβλήματος

Το πρόβλημα που μελετάται αφορά τη βέλτιστη κατανομή και δρομολόγηση συνεργείων μιας τεχνικής εταιρείας για την αποκατάσταση βλαβών σε τμήμα της πόλης της Αθήνας. Για τη δημιουργία των εισερχόμενων δεδομένων, αναπτύχθηκε ένα πρόγραμμα παραγωγής τυχαίων σημείων, το οποίο λειτουργεί με βάση ρεαλιστικές παραδοχές και περιορισμούς.

Οι παραδοχές βασίζονται σε εμπειρικά δεδομένα και περιλαμβάνουν:

- Την κατηγοριοποίηση των σημείων ανάλογα με την προτεραιότητα (A1, A2, A3)
- Την τοποθέτηση σημείων σε κρίσιμες ή μη περιοχές (B1, B2)
- Την εκτίμηση χρόνου αποκατάστασης κάθε σημείου (20, 40 ή 75 λεπτά)
- Το πλήθος των εργασιών ανά ημέρα, το οποίο διαμορφώνεται ως εξής: 4 σημεία A1, 12–20 σημεία A2, 10 σημεία A3
- Την ελάχιστη και μέγιστη χρονική χωρητικότητα ενός συνεργείου (330 έως 420 λεπτά)

Το πρόγραμμα randomizer δημιουργεί ημερήσια σύνολα εργασιών, με χωρική κατανομή και επιμέρους χαρακτηριστικά που εξυπηρετούν την πολυπλοκότητα του προβλήματος, καθιστώντας την προσομοίωση πιο κοντά σε πραγματικά δεδομένα.

3.2.2 Μαθηματική Διατύπωση Αρχικού Μοντέλου

Για την επίλυση του προβλήματος δρομολόγησης χρησιμοποιήθηκε ως βάση το μαθηματικό μοντέλο [5] και το οποίο περιγράφεται στη συνέχεια:

Σύνολα

$V = \{0, 1, \dots, n\}$: Σύνολο σημείων (0 = αμαξοστάσιο)

$A = \{(i, j) \in V \times V \mid i \neq j\}$: Σύνολο κατευθυνόμενων τόξων

K : Μέγιστος αριθμός διαθέσιμων οχημάτων (ορίζεται ως 3)

Παράμετροι

c_{ij} : Ευκλείδεια απόσταση μεταξύ σημείων i και j

d_i : Χρόνος εξυπηρέτησης (σε λεπτά) για το σημείο i (με $d_0 = 0$)

$C = 420$: Μέγιστη χωρητικότητα (χρονική) κάθε οχήματος

Μεταβλητές Απόφασης

$x_{ij} \in \{0, 1\}$: 1 αν το όχημα επισκέπτεται το j αμέσως μετά το i , 0 αλλιώς

$u_i \in \mathbb{R}^+$: Συγκεντρωμένος χρόνος εξυπηρέτησης έως το σημείο i (για MTZ υποπεριορισμούς)

Συνάρτηση Στόχου

Ελαχιστοποίηση της συνολικής απόστασης μετακίνησης:

Minimize:

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} \quad (3.2.2.1)$$

Περιορισμοί

1. Κάθε σημείο εξυπηρετείται ακριβώς μία φορά:

$$\sum_i x_{ij} = 1 \quad \forall i \in V \quad (3.2.2.2)$$

$$\sum_j x_{ij} = 1 \quad \forall j \in V \quad (3.2.2.3)$$

2. Περιορισμός πλήθους οχημάτων:

$$\sum_j x_{0j} \leq |K| \quad (3.2.2.4)$$

3. Ισορροπία εισερχόμενων/εξερχόμενων οχημάτων:

$$\sum_i x_{i0} = \sum_j x_{0j} \quad \forall i, j \in V \quad i \neq 0 \quad j \neq 0 \quad (3.2.2.5)$$

4. MTZ υποπεριορισμοί για αποφυγή υποκύκλων:

$$u_i - u_j + C * x_{ij} \leq C - d_j \quad \forall i \neq j, \quad i \neq 0, \quad j \neq 0 \quad (3.2.2.6)$$

$$d_i \leq u_i \leq C \quad \forall i \neq 0 \quad (3.2.2.7)$$

V: Το σύνολο των κόμβων (σημείων) — περιλαμβάνει την αμαξοστάσιο (κόμβος 0) και όλους τους πελάτες.

c_{ij} : Το κόστος ή η απόσταση μετακίνησης από το σημείο i στο σημείο j .

x_{ij} : Δυαδική μεταβλητή:

- $x_{ij} = 1$: Αν υπάρχει μετακίνηση από i σε j
- $x_{ij} = 0$: Διαφορετικά

x_{0j} : Αν ένα όχημα ξεκινήσει από την αμαξοστάσιο και πάει στον κόμβο j

$|K|$: Πλήθος διαθέσιμων οχημάτων

u_i : Βοηθητική μεταβλητή που εκφράζει τη σωρευτική ζήτηση (ή θέση) στο δρομολόγιο στο σημείο i

d_j : Ζήτηση του κόμβου j

C : Χωρητικότητα οχήματος

3.2.3 Υπολογιστικό Περιβάλλον και Τεχνική Υλοποίηση

Η υλοποίηση πραγματοποιήθηκε στη γλώσσα Python. Όλα τα πειράματα πραγματοποιήθηκαν σε συμβατικό υπολογιστή με 8 GB RAM και επεξεργαστή Intel 11th GEN Intel(R) Core(TM) i5-11300H @ 3.10GHz χρησιμοποιώντας το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) PyCharm. Για τη διαχείριση, επεξεργασία και οπτικοποίηση των δεδομένων αξιοποιήθηκαν οι βιβλιοθήκες NumPy, Pandas και Matplotlib [21].

Η αρχιτεκτονική της εφαρμογής περιλαμβάνει:

- Εργαλείο παραγωγής ημερήσιων δεδομένων (randomizer)
- Αλγορίθμους επίλυσης (PuLP + MTZ, OR-Tools)
- Υποσύστημα απεικόνισης αποτελεσμάτων (plotting διαδρομών, αναφορά χρόνων/φορτίου)

Το τελικό αποτέλεσμα είναι ένα σύστημα που εκτελεί τη διαδικασία προσομοίωσης και επίλυσης καθημερινών δρομολογίων για τεχνικά συνεργεία, ενσωματώνοντας χρονικά, τοπικά και επιχειρησιακά χαρακτηριστικά.

3.2.4 Επισκόπηση της Εφαρμοζόμενης Μεθοδολογίας

Ο στόχος της διπλωματικής είναι η δημιουργία ενός προγράμματος που θα καθορίζει το πλήθος των συνεργείων καθώς και την πορεία που θα εκτελέσουν αυτά μέσα στην ημέρα, ανάλογα με τον όγκο των προβλημάτων που καλούνται να διαχειριστούν. Με αυτόν τον τρόπο, η εργασία διαχωρίζεται σε τρία τμήματα. Πρώτα, πρέπει να δημιουργηθεί ένα πρόγραμμα που παράγει τυχαία σημεία με βάση τους περιορισμούς που θέτουμε για το συγκεκριμένο πρόβλημα (χωρικούς, ποσοτικούς και χρονικούς). Το δεύτερο τμήμα είναι να επιλέξουμε τον αλγόριθμο με τον οποίο θα καταστεί δυνατή η επίλυση του προβλήματος. Θα κάνουμε πειραματική σύγκριση της επίλυσης του μαθηματικού μοντέλου με χρήση OR-Tools και PuLP με MTZ περιορισμούς. Και το τρίτο τμήμα είναι η δημιουργία ενός αλγορίθμου που θα καθορίζει το πλήθος των συνεργείων αναλόγως με τον φόρτο σημείων και θα καθορίζει την πορεία που θα ακολουθήσει κάθε όχημα.

Το Google OR-Tools είναι ένα λογισμικό ανοικτού κώδικα για βελτιστοποίηση, το οποίο αναπτύχθηκε από την Google. Υποστηρίζει διάφορους τύπους αλγορίθμων όπως γραμμικός προγραμματισμός, ακέραιος προγραμματισμός, καθώς και προβλήματα δρομολόγησης και χρονοπρογραμματισμού. Το OR-Tools αξιοποιεί εξελιγμένους αλγορίθμους αναζήτησης και υποστηρίζει πολλαπλά backends όπως το SCIP, CBC, CP-SAT. Είναι ιδιαίτερα κατάλληλο για μεγάλα και πολύπλοκα προβλήματα με περιορισμούς. Έχει υψηλή απόδοση και υποστηρίζει σύνθετους περιορισμούς και προσαρμοσμένες στρατηγικές αναζήτησης [13].

Το PuLP είναι μια βιβλιοθήκη Python για μοντελοποίηση γραμμικών προβλημάτων βελτιστοποίησης. Η φόρμουλα MTZ (Miller–Tucker–Zemlin) χρησιμοποιείται κυρίως για προβλήματα TSP και VRP, για την αποφυγή υποδιαδρομών. Η μέθοδος MTZ είναι απλή στην κατανόηση και την υλοποίηση, αλλά μπορεί να είναι λιγότερο αποδοτική για μεγάλα προβλήματα. Το PuLP υποστηρίζει πολλαπλούς solvers όπως CBC, CPLEX και Gurobi. Είναι εύκολο στην εκμάθηση και χρήση και είναι κατάλληλο για ακαδημαϊκή χρήση και μικρά προβλήματα. Όμως, έχει περιορισμένες δυνατότητες για περίπλοκους περιορισμούς και δεν είναι αποδοτικό σε μεγάλης κλίμακας προβλήματα [14], [22].

Η σύγκριση των αλγορίθμων καθώς και η τελική επίλυση γίνεται με το πρόγραμμα PyCharm. Το PyCharm είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης λογισμικού (IDE) που έχει σχεδιαστεί ειδικά για τη γλώσσα προγραμματισμού Python. Αναπτύχθηκε από την JetBrains και προσφέρει ένα ευρύ φάσμα εργαλείων που διευκολύνουν τον προγραμματιστή, όπως αυτόματη συμπλήρωση κώδικα, έλεγχο συντακτικού, ενσωματωμένο debugger και διαχείριση έργων. Ένα από τα κύρια πλεονεκτήματά του είναι η ευκολία στη χρήση και η υποστήριξη για σύγχρονες τεχνολογίες όπως Django, Flask και επιστημονικές βιβλιοθήκες (NumPy, Pandas κ.ά.). Επιπλέον, διαθέτει εργαλεία για testing, version control (π.χ. Git) και υποστήριξη βάσεων δεδομένων. Ωστόσο, τα αρνητικά του περιλαμβάνουν την υψηλή κατανάλωση πόρων συστήματος, που μπορεί να επηρεάσει την απόδοση σε παλαιότερους υπολογιστές, καθώς και το γεγονός ότι η πλήρης έκδοση (Professional) είναι επί πληρωμή, περιορίζοντας τις δυνατότητες στην δωρεάν έκδοση (Community). Παρ' όλα αυτά, παραμένει ένα από τα πιο δημοφιλή IDEs για Python, ιδιαίτερα σε επαγγελματικό περιβάλλον [21].

3.2.5 Υλοποίηση Αλγορίθμου και Σύγκριση Επιλυτών

Για την επίλυση του προβλήματος του περιορισμένου χρονικά οχήματος (CVRP με χρονικούς περιορισμούς), υλοποιήθηκε το μαθηματικό μοντέλο που περιγράφηκε στην προηγούμενη ενότητα, χρησιμοποιώντας δύο διαφορετικές προσεγγίσεις: (α) με τη βιβλιοθήκη PuLP σε συνδυασμό με MTZ υποπεριορισμούς για την αποφυγή υποκύκλων, και (β) με τη χρήση του Google OR-Tools, ενός ισχυρού solver βελτιστοποίησης για προβλήματα δρομολόγησης. Και στις δύο περιπτώσεις χρησιμοποιήθηκαν τα ίδια σύνολα δεδομένων (τοποθεσίες πελατών, αποστάσεις, χρόνοι εξυπηρέτησης), με μέγιστο αριθμό διαθέσιμων οχημάτων $K=3$ και χρονική χωρητικότητα $C=420$ λεπτά ανά όχημα.

Η υλοποίηση με PuLP βασίστηκε σε γραμμικό μοντέλο με χρήση δυαδικών μεταβλητών για τις μεταβάσεις μεταξύ σημείων και συνεχών μεταβλητών για τον σωρευτικό χρόνο εξυπηρέτησης. Η υλοποίηση PuLP επιλύει το μοντέλο χρησιμοποιώντας αλγόριθμο Branch-and-Bound. Αντίθετα, η προσέγγιση με OR-Tools αξιοποίησε το εξειδικευμένο module VRP (Vehicle Routing Problem) με ενσωματωμένους μηχανισμούς για την επιβολή χρονικών παραθύρων, χωρητικότητας και ελαχιστοποίησης απόστασης. Ο solver της Google διαχειρίζεται εσωτερικά την αποφυγή κύκλων και προσφέρει βελτιστοποιημένες ευρετικές για μεγαλύτερα προβλήματα [14], [22].

Η συγκριτική αξιολόγηση των δύο επιλυτών ανέδειξε τα πλεονεκτήματα και τα όρια της κάθε προσέγγισης. Η λύση με PuLP είναι πιο ακριβής, αλλά αποδεικνύεται υπολογιστικά πιο αργή, ειδικά καθώς αυξάνεται ο αριθμός των σημείων. Η υλοποίηση με OR-Tools, από την άλλη, προσφέρει πολύ ταχύτερους χρόνους επίλυσης και μπορεί να επιλύσει μεγαλύτερα δίκτυα, αν και είναι λιγότερο ευέλικτη στην προσθήκη σύνθετων περιορισμών. Συνολικά, για μικρότερα προβλήματα με ανάγκη εύρεση βέλτιστης λύσης, το μοντέλο PuLP+MTZ είναι κατάλληλο. Για μεγαλύτερα ή επιχειρησιακά σενάρια που απαιτούν ταχύτητα και σταθερότητα, το OR-Tools υπερέχει σημαντικά.

Χαρακτηριστικό	PuLP + MTZ	OR-Tools
Τύπος εργαλείου	Βιβλιοθήκη για LP/IP	Πλήρες εργαλείο operations research
Υποστήριξη TSP	Ναι, με MTZ περιορισμούς	Ναι, με ειδικό routing solver
Απόδοση σε μεγάλα προβλήματα	Μέτρια	Πολύ Καλή
Εξαρτήσεις	Python + LP Solver (π.χ. CBC)	Περιλαμβάνει δικό του μεταερευτικό solver

Πίνακας 1 : Συγκριτικός Πίνακας μεταξύ PuLP και OR-Tools

4. Επίλυση Προβλήματος

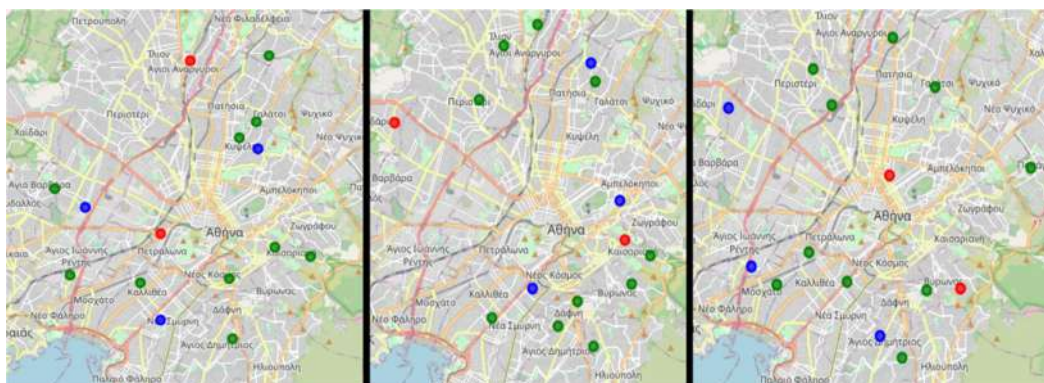
4.1 Σύγκριση Επιλυτών

Προκειμένου να συγκριθούν τα αποτελέσματα των αλγορίθμων PuLP και OR-Tools, εκτελέστηκαν από 3 επαναλήψεις για κάθε ένα, για περιπτώσεις προβλημάτων με 15, 18, 20 και 23 σημεία. Τα σημεία αυτά έχουν την ίδια βαρύτητα και η επίλυση γίνεται μόνο έχοντας μέγιστη χωρητικότητα οχήματος 7 εργάσιμες ώρες (420 λεπτά). Το κάθε σημείο έχει ως demand τον χρόνο αποκατάστασης που ορίζεται τυχαία για κάθε σημείο και είναι 75 λεπτά, 40 λεπτά και 20 λεπτά, τον χρόνο δηλαδή που χρειάζεται για αποκατάσταση της βλάβης. Ο χρόνος ορίστηκε μετά από παραδοχές. Ως κριτήριο τερματισμού του επιλύτη OR-Tools ορίστηκε ο υπολογιστικός χρόνος των 180 δευτερολέπτων ανά επανάληψη, παράγοντας λύσεις κοντά στο βέλτιστο [18]. Ο κώδικας PuLP δεν έχει χρονικό περιορισμό.

Παρακάτω παρουσιάζονται οι χάρτες και τα δεδομένα για κάθε επανάληψη.

➤ 15 ΣΗΜΕΙΑ

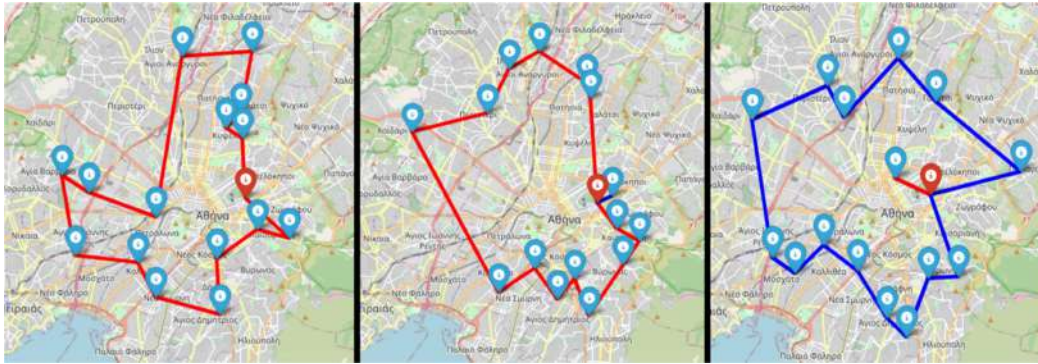
ΑΡΧΙΚΑ ΣΗΜΕΙΑ



ΑΠΟΤΕΛΕΣΜΑΤΑ

ΑΡΙΘΜΟΣ ΣΗΜΕΙΩΝ	ΕΠΑΝΑΛΗΨΗ	ΜΕΘΟΔΟΣ	ΑΡΙΘΜΟΣ ΟΧΗΜΑΤΩΝ	ΣΗΜΕΙΑ	ΑΠΟΣΤΑΣΗ	CAPACITY	CPU TIME (SEC)
15	1	PULP	1	15	36,94	395	14.99
		ORTOOLS	1	15	38,94	395	180
	2	PULP	2	14	32,66	410	4,92
				1	1,32	40	
		ORTOOLS	2	14	37,07	410	180
				1	1,32	40	
	3	PULP	2	14	36,89	395	1.46
				1	3,08	40	
		ORTOOLS	2	14	36,89	395	180
				1	3,08	40	

ΧΑΡΤΗΣ ΔΙΑΔΡΟΜΩΝ ΜΕ PuLP



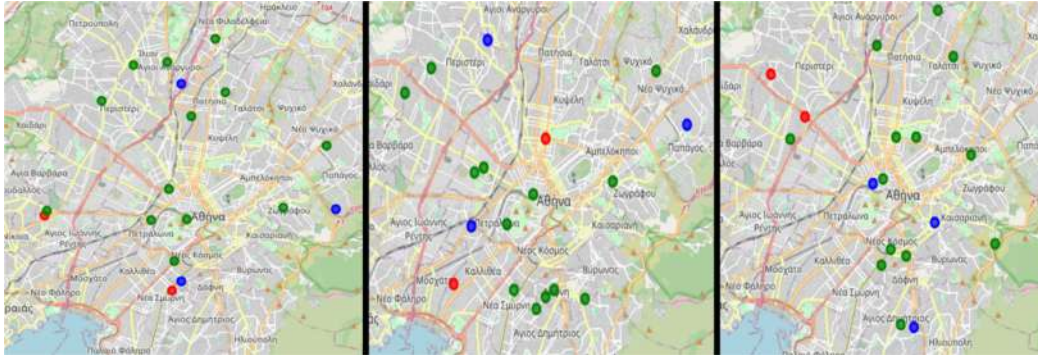
ΧΑΡΤΗΣ ΔΙΑΔΡΟΜΩΝ ΜΕ ORTOOLS



Παρατηρείται ότι τα αποτελέσματα του αλγορίθμου PuLP είναι καλύτερα στην πρώτη και δεύτερη δοκιμή, ενώ στην τρίτη δοκιμή και οι δύο μέθοδοι (PuLP και OR-Tools) καταλήγουν στην ίδια λύση. Σε όλες τις περιπτώσεις, ο αριθμός των χρησιμοποιούμενων οχημάτων καθώς και το συνολικό φορτίο ανά διαδρομή παραμένουν σταθερά. Συγκεκριμένα, στην πρώτη δοκιμή ο PuLP αλγόριθμος παρέχει λύση κατά 5,14% καλύτερη ως προς τη συνολική απόσταση, ενώ στη δεύτερη το ποσοστό βελτίωσης ανέρχεται σε 11,9%

➤ 18 ΣΗΜΕΙΑ

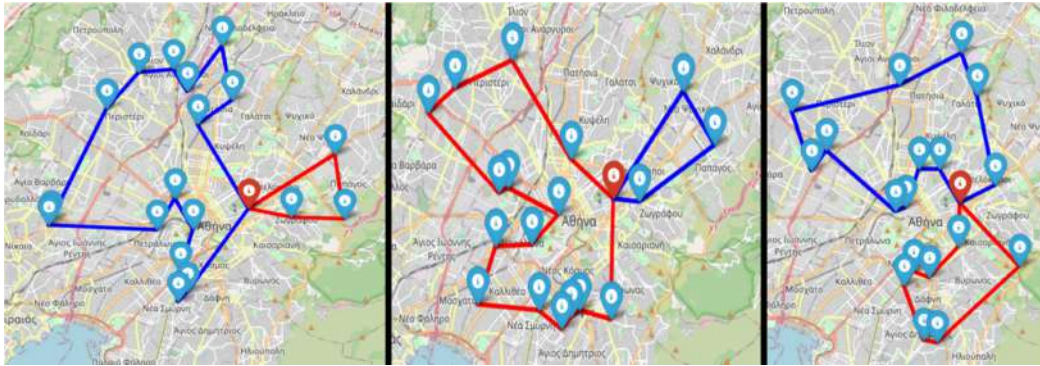
ΑΡΧΙΚΑ ΣΗΜΕΙΑ



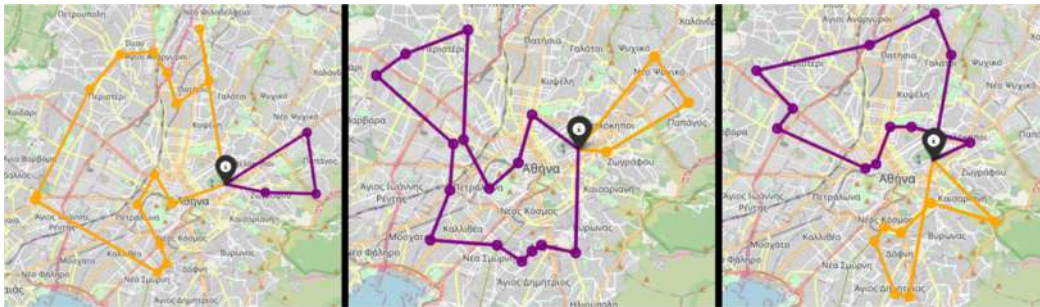
ΑΠΟΤΕΛΕΣΜΑΤΑ

ΑΡΙΘΜΟΣ ΣΗΜΕΙΩΝ	ΕΠΑΝΑΛΗΨΗ	ΜΕΘΟΔΟΣ	ΑΡΙΘΜΟΣ ΟΧΗΜΑΤΩΝ	ΣΗΜΕΙΑ	ΑΠΟΣΤΑΣΗ	CAPACITY	CPU TIME (SEC)
18	1	PULP	2	3	10,13	80	540.23
				15	33,09	395	
		ORTOOLS	2	3	10,13	80	180
				15	34,49	395	
	2	PULP	2	3	10,59	115	82.36
				15	29,25	415	
		ORTOOLS	2	3	10,59	115	180
				15	30,79	415	
	3	PULP	2	7	14,38	180	134.35
				11	23,93	315	
		ORTOOLS	2	7	16,13	180	180
				11	25,09	315	

ΧΑΡΤΕΣ ΔΙΑΔΡΟΜΩΝ ΜΕ PuLP



ΧΑΡΤΕΣ ΔΙΑΔΡΟΜΩΝ ΜΕ ORTOOLS

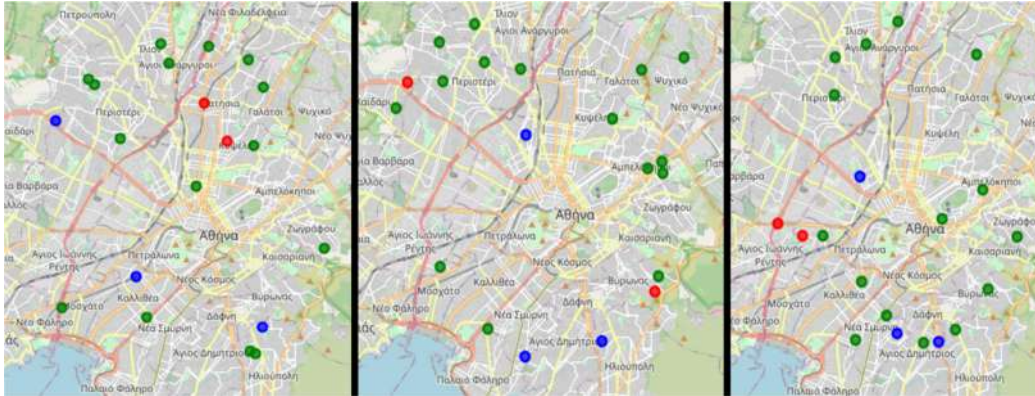


Παρατηρείται ότι και στις τρεις επαναλήψεις, τόσο ο αριθμός των χρησιμοποιούμενων οχημάτων όσο και το συνολικό φορτίο ανά διαδρομή παραμένουν σταθερά. Ωστόσο, η συνολική απόσταση που διανύουν τα οχήματα είναι μικρότερη στις λύσεις που προκύπτουν από τον αλγόριθμο PuLP. Συγκεκριμένα, η συνολική απόσταση στις επαναλήψεις είναι κατά 3,14%, 3,72% και 7,06% μικρότερη αντίστοιχα, συγκριτικά με τα αποτελέσματα του αλγορίθμου OR-Tools.

Παρά τα βελτιωμένα αποτελέσματα ως προς την απόσταση, παρατηρείται σημαντική αύξηση στον υπολογιστικό χρόνο, ιδίως σε σύγκριση με την περίπτωση των 15 σημείων. Επιπλέον, καταγράφηκαν μεγάλες διακυμάνσεις στον χρόνο εκτέλεσης μεταξύ των επαναλήψεων· χαρακτηριστικά, στην πρώτη επανάληψη το πρόγραμμα χρειάστηκε περίπου 9 λεπτά για να παραγάγει αποτέλεσμα.

➤ 20 ΣΗΜΕΙΑ

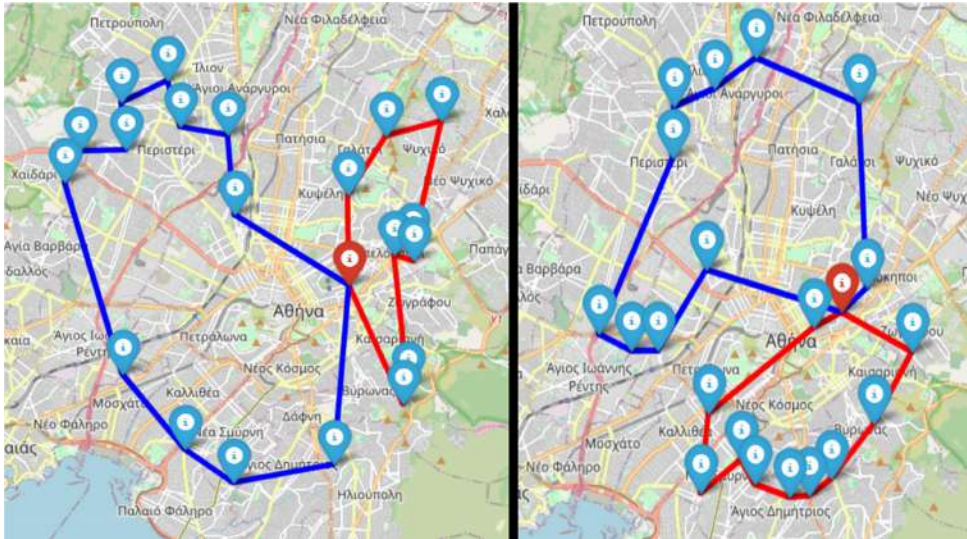
ΑΡΧΙΚΑ ΣΗΜΕΙΑ



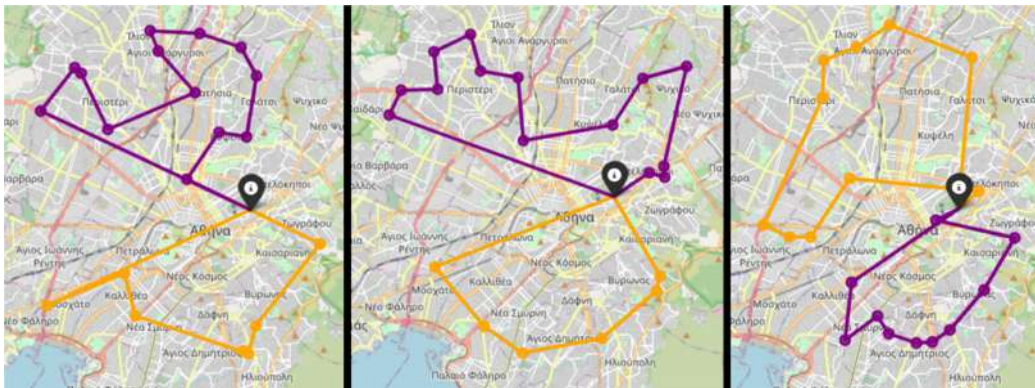
ΑΠΟΤΕΛΕΣΜΑΤΑ

ΑΡΙΘΜΟΣ ΣΗΜΕΙΩΝ	ΕΠΑΝΑΛΗΨΗ	ΜΕΘΟΔΟΣ	ΑΡΙΘΜΟΣ ΟΧΗΜΑΤΩΝ	ΣΗΜΕΙΑ	ΑΠΟΣΤΑΣΗ	CAPACITY	CPU TIME (SEC)
20	1	PULP					697.34
		ORTOOLS	2	7	22,28	180	180
	2	PULP	2	13	26,88	335	383.17
				8	17,25	215	
		ORTOOLS	2	12	29,43	390	180
				6	18,26	250	
	3	PULP	2	14	28,97	355	317.85
				10	16	240	
		ORTOOLS	2	10	23,84	330	180
				10	17,46	240	
				10	24,97	330	

ΧΑΡΤΕΣ ΔΙΑΔΡΟΜΩΝ ΜΕ PuLP



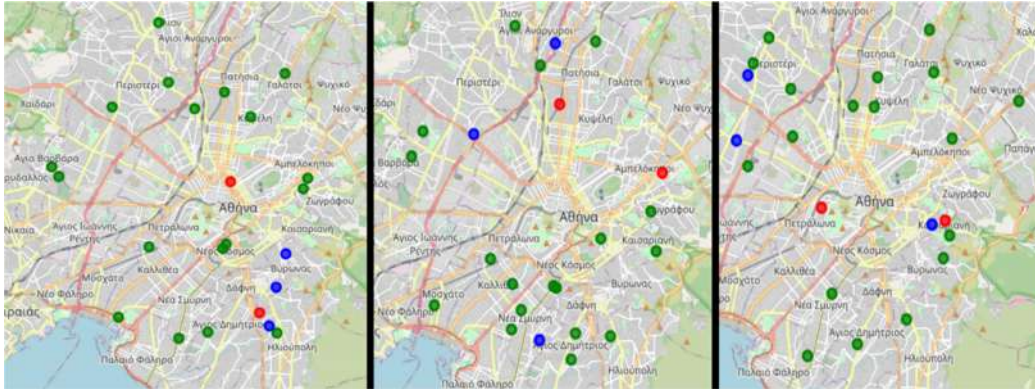
ΧΑΡΤΕΣ ΔΙΑΔΡΟΜΩΝ ΜΕ ORTOOLS



Στο συγκεκριμένο παράδειγμα παρατηρείται αδυναμία επίλυσης της πρώτης επανάληψης, καθώς η εκτέλεση διακόπηκε μετά από 11,6 λεπτά χωρίς να παραχθεί λύση. Επιπλέον, στη δεύτερη επανάληψη διαπιστώνεται ότι τα δύο οχήματα ακολουθούν διαφορετικές διαδρομές, με την κάθε διαδρομή να εξυπηρετεί διαφορετικό όγκο ζήτησης. Η λύση που προκύπτει από τον αλγόριθμο PuLP εμφανίζεται βελτιωμένη, με τη συνολική απόσταση να είναι κατά 1,2% μικρότερη. Στην τρίτη επανάληψη, αν και οι διαδρομές εξυπηρετούν τα ίδια σημεία, η λύση του PuLP είναι και πάλι καλύτερη, με τη συνολική απόσταση να είναι κατά 6,1% μικρότερη.

➤ 23 ΣΗΜΕΙΑ

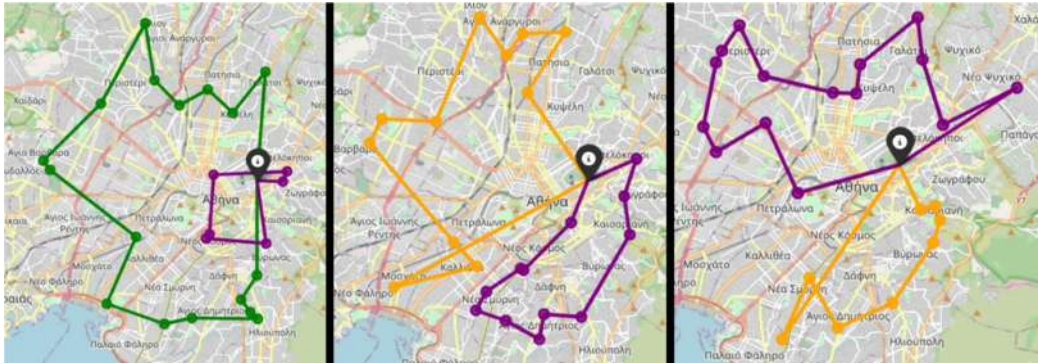
ΑΡΧΙΚΑ ΣΗΜΕΙΑ



ΑΠΟΤΕΛΕΣΜΑΤΑ

ΑΡΙΘΜΟΣ ΣΗΜΕΙΩΝ	ΕΠΑΝΑΛΗΨΗ	ΜΕΘΟΔΟΣ	ΑΡΙΘΜΟΣ ΟΧΗΜΑΤΩΝ	ΣΗΜΕΙΑ	ΑΠΟΣΤΑΣΗ	CAPACITY	CPU TIME (SEC)
23	1	PULP					677.23
		ORTOOLS	2	6 17	10,71 35,67	140 415	180
	2	PULP					644.76
		ORTOOLS	2	11 12	30,92 17,38	315 315	180
	3	PULP					727.03
		ORTOOLS	2	8 15	17,6 29,86	200 395	180

ΧΑΡΤΕΣ ΔΙΑΔΡΟΜΩΝ ΜΕ ORTOOLS



Στο συγκεκριμένο παράδειγμα παρατηρείται αδυναμία εξαγωγής αποτελέσματος από τον αλγόριθμο PuLP εντός 10 λεπτών, καθώς η επίλυση δεν ολοκληρώθηκε στο προκαθορισμένο χρονικό όριο. Αντιθέτως, ο αλγόριθμος OR-Tools κατάφερε να παράγει λύση εντός του μέγιστου χρόνου αναζήτησης των 3 λεπτών που είχε οριστεί.

➤ Συμπέρασμα

Καταλήγουμε στο συμπέρασμα ότι η υλοποίηση με τον αλγόριθμο PuLP αποδίδει καλύτερα αποτελέσματα σε προβλήματα CVRP με μικρό πλήθος σημείων. Συγκεκριμένα, για προβλήματα έως και 15 σημεία, οι λύσεις που παράγονται είναι όχι μόνο ποιοτικά ανώτερες αλλά και υπολογιστικά ταχύτερες. Στην περιοχή των 15 έως 20 σημείων, ο PuLP συνεχίζει να δίνει καλύτερες λύσεις ως προς την απόσταση, ωστόσο παρατηρείται σημαντική αύξηση στον χρόνο υπολογισμού.

Η αύξηση του αριθμού των σημείων, η μεταξύ τους γεωμετρία και η ζήτηση (demand) που αντιστοιχεί σε κάθε σημείο συμβάλλουν στη δυσκολία εύρεσης λύσης. Στο πρόβλημα με 20 σημεία καταγράφηκε αδυναμία επίλυσης εντός του χρονικού ορίου των περίπου 10 λεπτών. Για προβλήματα με περισσότερα από 20 σημεία, παρατηρείται γενικευμένη αδυναμία εξαγωγής λύσης εντός αποδεκτού χρόνου.

Αντιθέτως, η υλοποίηση με OR-Tools παρουσιάζει σταθερή απόδοση, καθώς καταφέρνει να παράγει λύση σε κάθε περίπτωση εντός του προκαθορισμένου ορίου των 3 λεπτών. Αν και οι λύσεις αυτές είναι σε ορισμένες περιπτώσεις υποδεέστερες σε σχέση με εκείνες του PuLP, παραμένουν αποδεκτές, με μικρές αποκλίσεις που έχουν αναλυθεί παραπάνω.

Συμπερασματικά, για προβλήματα CVRP με έως και 20 σημεία, προτείνεται η χρήση του PuLP με στόχο την εύρεση της βέλτιστης λύσης. Για προβλήματα με μεγαλύτερο πλήθος σημείων, ενδείκνυται η χρήση του OR-Tools, καθώς εξασφαλίζει παραγωγή λύσης εντός εύλογου χρονικού διαστήματος, έστω και με μικρή θυσία στην ακρίβεια.

4.2 Μαθηματικό Μοντέλο για Μεγαλύτερα Δίκτυα

Με βάση τη συνολική ανάλυση, την παραγωγή δεδομένων και τη συγκριτική αξιολόγηση επιλυτών, διαμορφώθηκε ένα ενισχυμένο μαθηματικό μοντέλο που ενσωματώνει πρόσθετες επιχειρησιακές απαιτήσεις και λειτουργικούς περιορισμούς. Το τελικό μοντέλο σχεδιάστηκε ώστε να αντικατοπτρίζει με μεγαλύτερη ακρίβεια τις ρεαλιστικές συνθήκες λειτουργίας τεχνικών συνεργείων, ενσωματώνοντας κρίσιμες παραμέτρους όπως σταθμισμένο κόστος μετακίνησης ανάλογα με τη σπουδαιότητα του σημείου, ποινές μη εξυπηρέτησης, χρονικές δεσμεύσεις ανά ζώνη (όπως η B1), και περιορισμούς στο ελάχιστο φορτίο ενεργών δρομολογίων. Το προτεινόμενο μοντέλο αναπτύσσεται ως πρόβλημα μικτού ακέραιου γραμμικού προγραμματισμού και στοχεύει στην ελαχιστοποίηση του συνολικού κόστους εξυπηρέτησης, λαμβάνοντας υπόψη τόσο τη χωρική κατανομή των σημείων όσο και τη χρονική και λειτουργική αποδοτικότητα των διαθέσιμων συνεργείων. Ακολουθεί η πλήρης μαθηματική διατύπωση του τελικού μοντέλου [4], [6], [11], [14], [22].

Σύνολα

$V = \{0, 1, \dots, n\}$: Σύνολο σημείων (0 = αμαξοστάσιο)

K : Μέγιστος αριθμός διαθέσιμων οχημάτων (ορίζεται ως 3)

$B1 \in V \setminus \{0\}$: Σύνολο σημείων που ανήκουν στην ζώνη B1

Παράμετροι

d_{ij} : Ευκλείδεια απόσταση μεταξύ σημείων i και j

w_j : Συντελεστής βαρύτητας σημείου j

$$w_j = \begin{cases} 1 & \text{αν } j \in A_1 \\ 0,75 & \text{αν } j \in A_2 \\ 0,5 & \text{αν } j \in A_3 \end{cases}$$

$c_{ij} = \frac{d_{ij}}{w_j}$: Σταθμισμένο κόστος μετάβασης

q_j : Απαίτηση (σε λεπτά) του σημείου j – χρόνος

$Q = 420$: Μέγιστη χωρητικότητα οχήματος

$Q_{min} = 330$: Ελάχιστο φορτίο για έγκυρη διαδρομή

$T_{max}^{B1} = 180$: Μέγιστος συνολικός χρόνος εξυπηρέτησης ζώνης B1 ανά όχημα

P_j : Ποινή μη εξυπηρέτησης

$$P_j = \begin{cases} 100000 & \text{αν } j \in A_1 \\ 50000 & \text{αν } j \in A_2 \\ 10000 & \text{αν } j \in A_3 \end{cases}$$

Μεταβλητές Απόφασης

$x_{ijk} \in \{0,1\}$: 1 αν το όχημα k πάει από σημείο i στο j

$\psi_{jk} \in \{0,1\}$: 1 αν το σημείο j εξυπηρετείται από το όχημα k

$I_{jk} \in R^+$: σωρευτικό φορτίο στο όχημα k μετά την επίσκεψη του j

$z_k \in \{0,1\}$: 1 αν το όχημα k χρησιμοποιείται

Αντικειμενική Συνάρτηση Στόχου – Objective Function

$$\min \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} c_{ij} * x_{ijk} + \sum_{j \in V \setminus \{0\}} P_j * (1 - \sum_{k \in K} \psi_{jk}) \quad (4.2.1)$$

Όπου P_j : Ποινή για μη εξυπηρέτηση του σημείου j (ανάλογα με A1, A2, A3)

Περιορισμοί

1. Κάθε σημείο εξυπηρετείται το πολύ μία φορά:

$$\sum_{k \in K} \psi_{jk} \leq 1 \quad \forall j \in V \setminus \{0\} \quad (4.2.2)$$

2. Ροή διαδρομής ανά όχημα (εξασφάλιση κύκλου)

$$\sum_{j \in V} x_{0jk} = 1, \quad \sum_{i \in V} x_{i0k} = 1 \quad \forall k \in K \quad (4.2.3)$$

$$\sum_{i \in V} x_{ij} = \psi_{jk}, \quad \sum_{j \in V} x_{ijk} = \psi_{ik} \quad \forall i, j \in V \setminus \{0\}, \forall k \in K \quad (4.2.4)$$

3. Περιορισμός χωρητικότητας:

$$I_{jk} \geq I_{ik} + q_j - Q * (1 - x_{ijk}) \quad \forall i, j \in V, \forall k \in K \quad (4.2.5)$$

$$q_j * \psi_{jk} \leq I_{jk} \leq Q * \psi_{jk} \quad \forall j \in V, \forall k \in K \quad (4.2.6)$$

4. Ελάχιστο φορτίο ενεργών διαδρομών:

$$\sum_{j \in V \setminus \{0\}} q_j * \psi_{jk} \geq Q_{min} * z_k \quad \forall k \in K \quad (4.2.7)$$

Όπου $z_k = 1$ αν χρησιμοποιείται το όχημα k

5. Χρονικός περιορισμός σημείων στη ζώνη B1

$$\sum_{j \in B1} q_j * \psi_{jk} \leq T_{max}^{B1} * z_k \quad \forall k \in K \quad (4.2.8)$$

4.3 Εφαρμογή Τελικού Αλγόριθμου

Δεδομένης της πολυπλοκότητας του προβλήματος και του πλήθους των σημείων που καλείται να διαχειριστεί το σύστημα σε ημερήσια βάση, η επιλογή του αλγορίθμου OR-Tools κρίθηκε καταλληλότερη για την τελική υλοποίηση. Η βιβλιοθήκη αυτή προσφέρει εξελιγμένες μεταερευνητικές τεχνικές, όπως το Guided Local Search και το Large Neighborhood Search, οι οποίες αποδίδουν αποτελέσματα υψηλής ποιότητας σε αποδεκτούς χρόνους εκτέλεσης, ακόμη και σε προβλήματα μεγάλης κλίμακας. Το πλεονέκτημα της OR-Tools έναντι ακριβών μεθόδων, όπως αυτές που βασίζονται σε γραμμικό προγραμματισμό (π.χ. PuLP), έγκειται στην ικανότητά της να κλιμακώνεται αποδοτικά, διατηρώντας χαμηλή υπολογιστική απαίτηση ανεξαρτήτως αύξησης των σημείων.

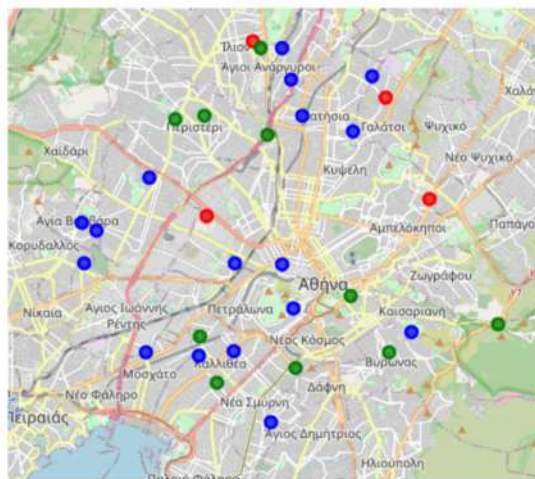
Η εφαρμογή σχεδιάστηκε έτσι ώστε να λειτουργεί επαναληπτικά, προσομοιώνοντας μια πλήρη εργασιακή εβδομάδα. Η καθημερινή δημιουργία τυχαίων σημείων εξυπηρέτησης και η εκτέλεση του αλγορίθμου σε πέντε διαδοχικές επαναλήψεις επιτρέπει τη μελέτη της συμπεριφοράς του συστήματος σε διαφορετικά λειτουργικά σενάρια. Μέσω αυτής της διαδικασίας διαπιστώνεται κατά πόσο το σύστημα μπορεί να διαχειριστεί την εισροή νέων δεδομένων, να μεταφέρει μη εξυπηρετηθέντα σημεία στην επόμενη ημέρα και να τηρεί τους περιορισμούς χωρητικότητας και προτεραιότητας με συνέπεια. Η επαναληπτική προσέγγιση προσδίδει στον κώδικα χαρακτηριστικά δυναμικής λειτουργίας, αναδεικνύοντας την καταλληλότητά του για καθημερινή επιχειρησιακή χρήση.

Τα αποτελέσματα των επαναλήψεων επιβεβαιώνουν την αποτελεσματικότητα της υλοποίησης. Σε όλες τις περιπτώσεις, η δρομολόγηση επιτεύχθηκε με τρόπο συμβατό προς τους τεθέντες περιορισμούς, ενώ η προτεραιοποίηση των σημείων υψηλής σημασίας διατηρήθηκε. Παρατηρήθηκε ότι η στοχαστική φύση του αλγορίθμου οδηγεί σε διαφοροποιημένες, αλλά ισοδύναμα αποδοτικές λύσεις. Επίσης, η μεταφορά σημείων χαμηλής προτεραιότητας στην επόμενη ημέρα αποδεικνύει τη δυνατότητα του συστήματος να διαχειρίζεται ασύμμετρη ζήτηση χωρίς απώλεια δεδομένων ή παραβίαση επιχειρησιακών κανόνων. Συνεπώς, η εφαρμογή του τελικού αλγορίθμου αποδεικνύεται επαρκώς ευέλικτη, επεκτάσιμη και αξιόπιστη, καλύπτοντας τις απαιτήσεις ενός ρεαλιστικού σεναρίου χρήσης.

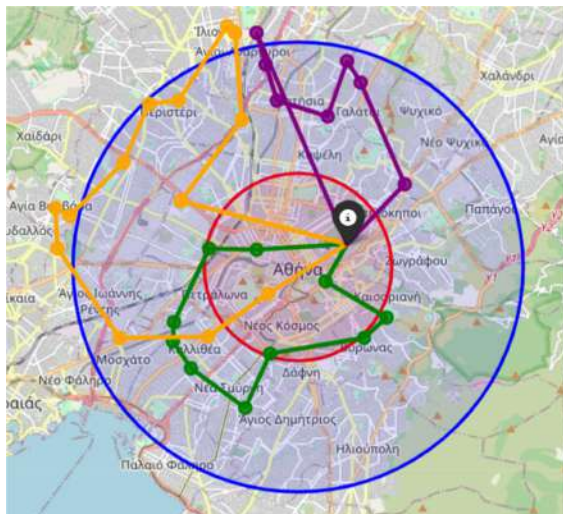
➤ 1^η ΗΜΕΡΑ

Δημιουργήθηκαν συνολικά 31 τυχαία σημεία. Παρατίθενται ο πίνακας με τις σχετικές πληροφορίες των σημείων, καθώς και ο χάρτης που απεικονίζει τη γεωγραφική τους κατανομή.

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	OPEN
2	37,97967	23,7222932	A2	B1	75	2025-05-23	OPEN
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	OPEN
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	OPEN
5	37,9534817	23,7615485	A2	B2	75	2025-05-23	OPEN
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	OPEN
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	OPEN
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	OPEN
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	OPEN
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	OPEN
11	38,031627	23,7220714	A2	B2	40	2025-05-23	OPEN
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	OPEN
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	OPEN
14	37,941795	23,7188034	A2	B2	40	2025-05-23	OPEN
15	37,9575984	23,696655	A2	B2	40	2025-05-23	OPEN
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	OPEN
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	OPEN
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	OPEN
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	OPEN
20	37,9585606	23,754938	A3	B2	20	2025-05-23	OPEN
21	37,95473	23,7262241	A3	B2	20	2025-05-23	OPEN
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	OPEN
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	OPEN
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	OPEN
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	OPEN
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	OPEN
27	38,0316	23,7158291	A3	B2	20	2025-05-23	OPEN
28	38,014468	23,6895591	A3	B2	20	2025-05-23	OPEN
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	OPEN
30	38,0240869	23,725119	A2	B2	40	2025-05-23	OPEN
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	OPEN



Με τη χρήση του αλγορίθμου εξάγονται οι διαδρομές των οχημάτων, καθώς και ο τελικός πίνακας σημείων, ο οποίος περιλαμβάνει πληροφορίες για τα εξυπηρετηθέντα και τα μη εξυπηρετηθέντα σημεία.



A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9534817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	OPEN
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED

Παρατηρείται ότι χρησιμοποιήθηκαν τρία οχήματα για την κάλυψη των απαιτήσεων της συγκεκριμένης περίπτωσης. Επιπλέον, το σημείο 19 δεν εξυπηρετήθηκε από καμία από τις διαδρομές και παρέμεινε σε κατάσταση OPEN, προκειμένου να παραμείνει διαθέσιμο για εξυπηρέτηση σε επόμενη ημέρα.

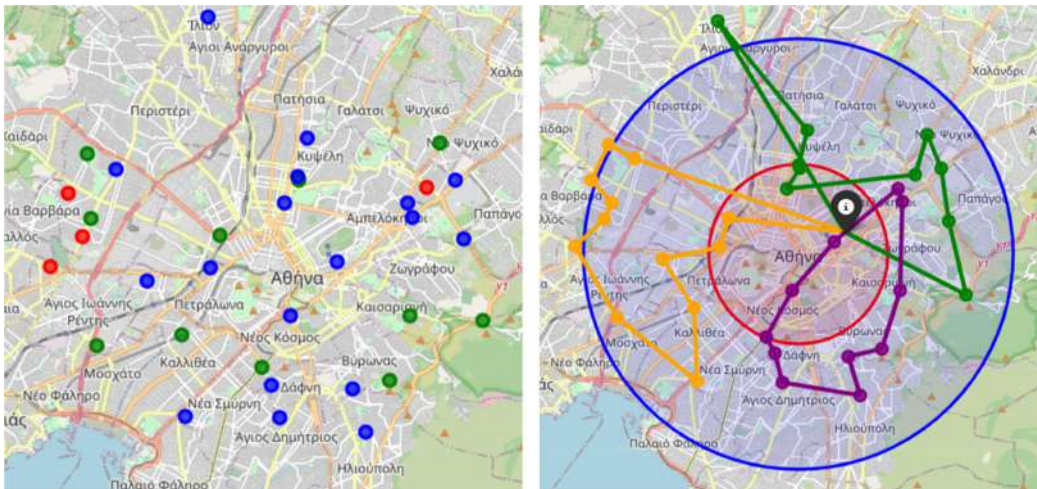
➤ 2^η ΗΜΕΡΑ

Τα σημεία που δημιουργήθηκαν είναι 32. Έτσι έχω:

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,02471	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,97983	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,01147	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,96348	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,97213	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,96917	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,00046	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,96246	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,03163	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,01548	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,01547	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,9418	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9576	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,01976	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,03309	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,98766	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,96537	23,7881108	A3	B2	20	2025-05-23	OPEN
20	37,95856	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,95845	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,97981	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,98972	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,95874	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,01077	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,01447	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,95117	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,02409	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,99537	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,03381	23,7091981	A2	B2	75	2025-05-24	OPEN
33	37,94003	23,7544334	A2	B2	75	2025-05-24	OPEN
34	37,98427	23,6731588	A1	B2	20	2025-05-24	OPEN
35	37,99953	23,682598	A2	B2	75	2025-05-24	OPEN
36	37,99692	23,7803971	A2	B2	75	2025-05-24	OPEN
37	37,98824	23,6755466	A3	B2	20	2025-05-24	OPEN
38	37,97435	23,6916898	A2	B2	75	2025-05-24	OPEN
39	37,96647	23,7327596	A2	B1	40	2025-05-24	OPEN
40	37,95057	23,7275189	A2	B2	40	2025-05-24	OPEN
41	38,00533	23,7757093	A3	B2	20	2025-05-24	OPEN
42	37,96195	23,7015338	A3	B2	20	2025-05-24	OPEN
43	37,98864	23,7677957	A2	B2	40	2025-05-24	OPEN
44	37,94342	23,7299333	A2	B2	40	2025-05-24	OPEN
45	37,95182	23,7613104	A3	B2	20	2025-05-24	OPEN
46	38,00286	23,674653	A3	B2	20	2025-05-24	OPEN
47	37,94376	23,7027373	A2	B2	40	2025-05-24	OPEN
48	37,96635	23,7671193	A3	B2	20	2025-05-24	OPEN
49	37,95976	23,6772071	A3	B2	20	2025-05-24	OPEN
50	37,98456	23,7124659	A3	B1	20	2025-05-24	OPEN
51	37,99525	23,7720967	A1	B2	40	2025-05-24	OPEN
52	37,97851	23,7462318	A2	B1	40	2025-05-24	OPEN
53	37,99703	23,7351688	A3	B1	20	2025-05-24	OPEN
54	37,95468	23,7247726	A3	B2	20	2025-05-24	OPEN
55	37,99191	23,7663477	A2	B2	40	2025-05-24	OPEN
56	37,99769	23,7350731	A2	B1	40	2025-05-24	OPEN
57	37,9917	23,7312294	A2	B1	40	2025-05-24	OPEN
58	37,94984	23,7506832	A2	B2	40	2025-05-24	OPEN
59	37,97744	23,6636983	A1	B2	20	2025-05-24	OPEN
60	37,99391	23,6690323	A1	B2	40	2025-05-24	OPEN
61	37,97718	23,7099784	A2	B1	40	2025-05-24	OPEN
62	37,98371	23,7825302	A2	B2	40	2025-05-24	OPEN
63	38,00654	23,7375275	A2	B2	40	2025-05-24	OPEN

Παρατηρείται ότι τα σημεία που έχουν ήδη εξυπηρετηθεί παραμένουν στον πίνακα δεδομένων, προκειμένου να είναι διαθέσιμα για μελλοντική αναφορά ή έλεγχο. Το σημείο 19, το οποίο δεν εξυπηρετήθηκε κατά την προηγούμενη ημέρα, διατηρείται σε κατάσταση ανοιχτή (OPEN), ώστε να ενταχθεί σε νέο πρόγραμμα δρομολόγησης. Δεδομένου ότι πρόκειται για τη δεύτερη ημέρα παρατηρείται και η αντίστοιχη αλλαγή στην ημερομηνία των νέων σημείων.

Παρατίθεται ο χάρτης με τα σημεία προς εξυπηρέτηση καθώς και οι τελικές διαδρομές.

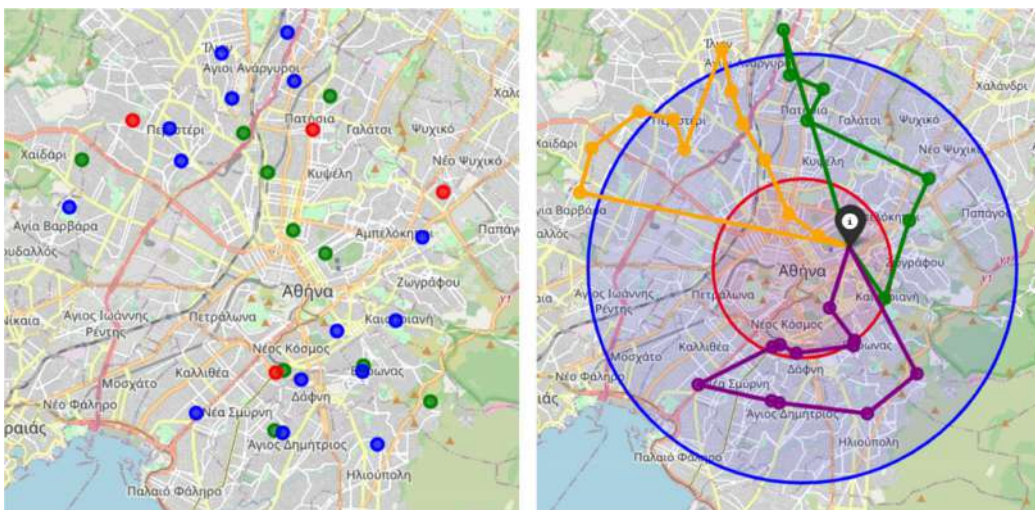


Παρατηρείται ότι όλα τα σημεία έχουν εξυπηρετηθεί επιτυχώς. Κατά συνέπεια, ο πίνακας στο τέλος της δεύτερης ημέρας ενημερώνεται ώστε να αντικατοπτρίζει την ολοκλήρωση της εξυπηρέτησης για κάθε σημείο.

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9634817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	CLOSED
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,0338068	23,7091981	A2	B2	75	2025-05-24	CLOSED
33	37,9400333	23,7544334	A2	B2	75	2025-05-24	CLOSED
34	37,9842731	23,6731588	A1	B2	20	2025-05-24	CLOSED
35	37,9995306	23,682598	A2	B2	75	2025-05-24	CLOSED
36	37,9969159	23,7803971	A2	B2	75	2025-05-24	CLOSED
37	37,9882392	23,6755466	A3	B2	20	2025-05-24	CLOSED
38	37,9743537	23,6916898	A2	B2	75	2025-05-24	CLOSED
39	37,9664718	23,7327596	A2	B1	40	2025-05-24	CLOSED
40	37,9505651	23,7275189	A2	B2	40	2025-05-24	CLOSED
41	38,0053307	23,7757093	A3	B2	20	2025-05-24	CLOSED
42	37,9619506	23,7015338	A3	B2	20	2025-05-24	CLOSED
43	37,9886403	23,7677957	A2	B2	40	2025-05-24	CLOSED
44	37,943421	23,7299333	A2	B2	40	2025-05-24	CLOSED
45	37,9518225	23,7613104	A3	B2	20	2025-05-24	CLOSED
46	38,002855	23,674653	A3	B2	20	2025-05-24	CLOSED
47	37,9437648	23,7027373	A2	B2	40	2025-05-24	CLOSED
48	37,96635	23,7671193	A3	B2	20	2025-05-24	CLOSED
49	37,9597588	23,6772071	A3	B2	20	2025-05-24	CLOSED
50	37,9845642	23,7124659	A3	B1	20	2025-05-24	CLOSED
51	37,9952524	23,7720967	A1	B2	40	2025-05-24	CLOSED
52	37,9785149	23,7462318	A2	B1	40	2025-05-24	CLOSED
53	37,9970325	23,7351688	A3	B1	20	2025-05-24	CLOSED
54	37,9546785	23,7247726	A3	B2	20	2025-05-24	CLOSED
55	37,99191	23,7663477	A2	B2	40	2025-05-24	CLOSED
56	37,9976946	23,7350731	A2	B1	40	2025-05-24	CLOSED
57	37,9917038	23,7312294	A2	B1	40	2025-05-24	CLOSED
58	37,9498422	23,7506832	A2	B2	40	2025-05-24	CLOSED
59	37,9774362	23,6636983	A1	B2	20	2025-05-24	CLOSED
60	37,9939115	23,6690323	A1	B2	40	2025-05-24	CLOSED
61	37,977176	23,7099784	A2	B1	40	2025-05-24	CLOSED
62	37,9837138	23,7825302	A2	B2	40	2025-05-24	CLOSED
63	38,0065378	23,7375275	A2	B2	40	2025-05-24	CLOSED

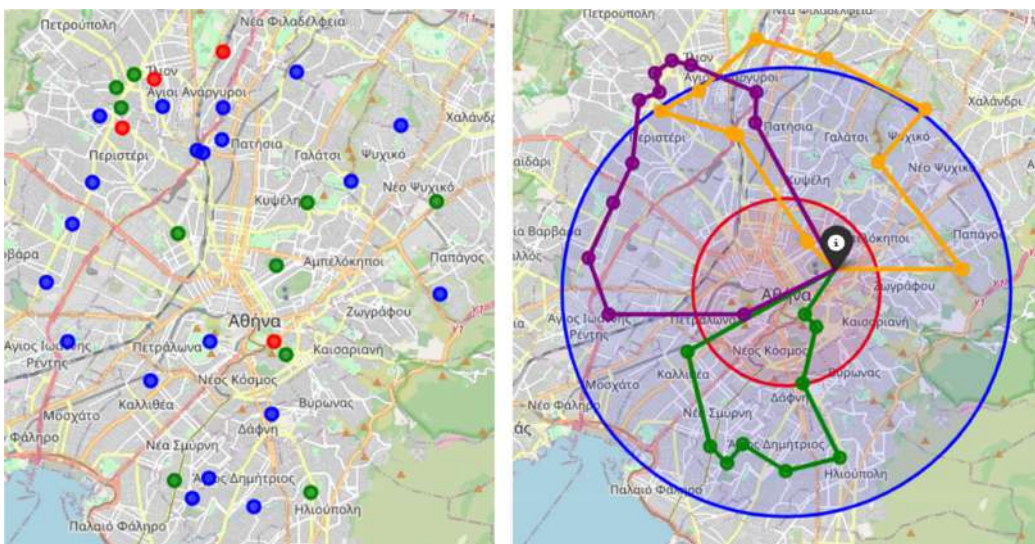
➤ 3^η ΗΜΕΡΑ

Στην τρίτη ημέρα δημιουργήθηκαν συνολικά 29 σημεία (βλ. πίνακα στο Παράρτημα). Ο χάρτης που ακολουθεί απεικονίζει τη γεωγραφική κατανομή των σημείων καθώς και τις διαδρομές των οχημάτων. Ο τελικός πίνακας της τρίτης ημέρας παρατίθεται επίσης στο Παράρτημα.



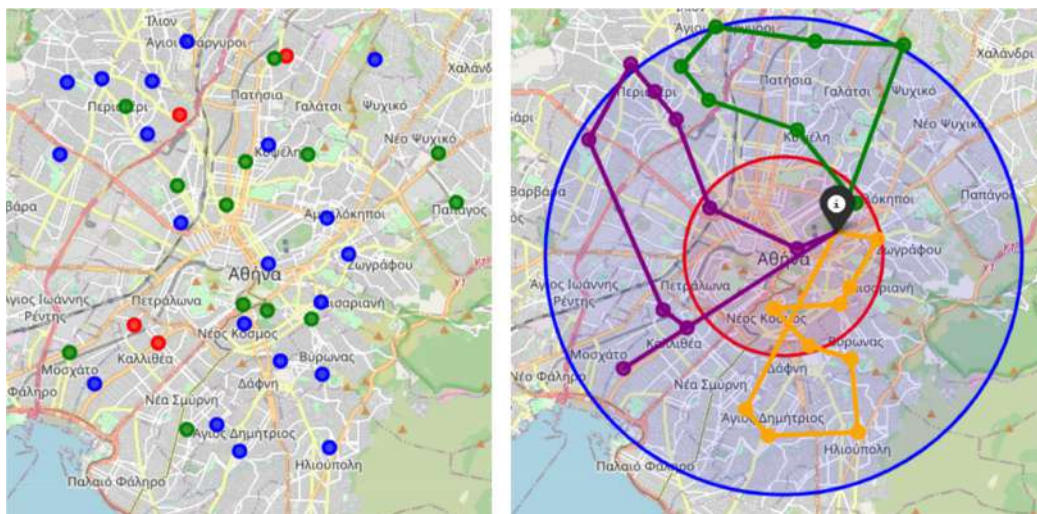
➤ 4^η ΗΜΕΡΑ

Στην τέταρτη ημέρα δημιουργήθηκαν συνολικά 34 σημεία (βλ. πίνακα στο Παράρτημα). Ο χάρτης που ακολουθεί απεικονίζει τη γεωγραφική θέση των σημείων καθώς και τις διαδρομές που ακολούθησαν τα οχήματα για την εξυπηρέτησή τους. Ο τελικός πίνακας της τέταρτης ημέρας παρατίθεται επίσης στο Παράρτημα.



➤ 5^η ΗΜΕΡΑ

Στην πέμπτη ημέρα δημιουργήθηκαν συνολικά 34 σημεία (βλ. πίνακα στο Παράρτημα). Ο χάρτης που ακολουθεί απεικονίζει τη γεωγραφική θέση των σημείων καθώς και τις διαδρομές που ακολουθήθηκαν για την εξυπηρέτησή τους. Ο τελικός πίνακας της πέμπτης ημέρας παρατίθεται επίσης στο Παράρτημα.



Ο ακόλουθος πίνακας συνοψίζει τα δεδομένα που προέκυψαν από τις πέντε επαναληπτικές εφαρμογές του αλγορίθμου, παρέχοντας συγκεντρωτική εικόνα των ημερήσιων αποτελεσμάτων.

ΗΜΕΡΑ	ΑΡΙΘΜΟΣ ΝΕΩΝ ΣΗΜΕΙΩΝ	ΑΡΙΘΜΟΣ ΟΡΕΝ ΣΗΜΕΙΩΝ	ΟΧΗΜΑΤΑ	ΣΗΜΕΙΑ ΠΟΥ ΕΞΥΠΗΡΕΤΗΘΗΚΑΝ	CAPACITY	ΑΠΟΣΤΑΣΗ ΠΟΥ ΔΙΑΝΥΘΗΚΕ (Km)	ΠΟΣΟΣΤΟ ΕΞΥΠΗΡΕΤΗΣΗΣ
1	31	31	3	30	1245	49,76	96,77
2	32	33	3	33	1235	54,25	100,00
3	29	29	3	29	1185	49,40	100,00
4	34	34	3	31	1250	49,95	91,18
5	34	37	3	25	1240	43,85	67,57

5. Αποτελέσματα - Συμπεράσματα

Ο παρών κώδικας υλοποιεί έναν αλγόριθμο επίλυσης του προβλήματος δρομολόγησης οχημάτων (Vehicle Routing Problem – VRP), ο οποίος εφαρμόζεται σε γεωγραφικά σημεία εντός των ζωνών B1 και B2 της Αθήνας. Ο κύριος στόχος του συστήματος είναι η βέλτιστη κατανομή των σημείων σε έως και τρία οχήματα, με σεβασμό στους περιορισμούς χωρητικότητας κάθε οχήματος, καθώς και στους χρονικούς περιορισμούς που έχουν τεθεί για την εξυπηρέτηση των επιμέρους σημείων.

Τα δεδομένα εισάγονται από αρχείο Excel και περιλαμβάνουν τη γεωγραφική θέση κάθε σημείου, την ομάδα εργασίας στην οποία ανήκει, τις χρονικές απαιτήσεις εξυπηρέτησης, την περιοχή, τον απαιτούμενο χρόνο παραμονής στο σημείο, καθώς και την κατάσταση (εξυπηρετημένο ή μη). Ο αλγόριθμος επεξεργάζεται τα δεδομένα με σκοπό την ελαχιστοποίηση της συνολικής απόστασης που θα διανύσουν τα οχήματα, τηρώντας παράλληλα όλους τους λειτουργικούς περιορισμούς. Τα αποτελέσματα αποδίδονται σε διαδραστικούς χάρτες HTML, όπου απεικονίζονται τόσο τα σημεία όσο και οι διαδρομές που καλούνται να ακολουθήσουν τα οχήματα. Επιπλέον, παράγονται πίνακες Excel ανά διαδρομή, οι οποίοι περιλαμβάνουν το σύνολο των εξυπηρετημένων σημείων, ταξινομημένα χρονικά και γεωγραφικά.

Η λειτουργία του αλγορίθμου, όπως προκύπτει από τη δοκιμαστική εφαρμογή του σε επαναλαμβανόμενα ημερήσια σενάρια, αποδεικνύεται πλήρως ορθή και αξιόπιστη. Η δρομολόγηση πραγματοποιείται με ακρίβεια και σύμφωνα με τους προκαθορισμένους περιορισμούς. Σημεία που δεν καθίσταται εφικτό να εξυπηρετηθούν εντός της ίδιας ημέρας διατηρούνται σε κατάσταση OPEN και μεταφέρονται αυτόματα στην επόμενη ημέρα για να ενταχθούν σε νέο πρόγραμμα δρομολόγησης. Στο πλαίσιο της πέμπτης επανάληψης, παρατηρείται ότι παραμένουν 12 σημεία σε κατάσταση OPEN, εκ των οποίων τα έντεκα ανήκουν στην κατηγορία A3 (χαμηλής προτεραιότητας) και ένα στην κατηγορία A2 (μεσαίας προτεραιότητας). Η εν λόγω κατανομή επιβεβαιώνει την ορθότητα του μηχανισμού ιεράρχησης και προτεραιοποίησης των σημείων, ο οποίος δίνει έμφαση στην εξυπηρέτηση των υψηλότερης προτεραιότητας εργασιών.

Η συνολική εικόνα που προκύπτει καταδεικνύει ότι το σύστημα είναι ιδιαίτερα κατάλληλο για χρήση από μικρές και μεσαίες επιχειρήσεις, οι οποίες καλούνται να διαχειριστούν καθημερινά έναν δυναμικό και ενδεχομένως τυχαίο όγκο σημείων εξυπηρέτησης. Ο κώδικας αποδεικνύεται εύχρηστος, αποδοτικός και ταχύς στην εκτέλεσή του, παρέχοντας με ακρίβεια και σαφήνεια τα επιθυμητά αποτελέσματα, δηλαδή το πρόγραμμα δρομολόγησης ανά όχημα καθώς και τον αντίστοιχο χάρτη με τις διαδρομές που πρέπει να ακολουθηθούν.

Η παραγωγή των καθημερινών δεδομένων επιτυγχάνεται μέσω υποστηρικτικού κώδικα, ο οποίος δημιουργεί τυχαία γεωγραφικά σημεία σε ακτίνα έως έξι χιλιομέτρων από το κέντρο της Αθήνας, με σημείο αναφοράς το Σύνταγμα. Τα σημεία αυτά ταξινομούνται σε δύο γεωγραφικές ζώνες (B1 και B2) και αντιστοιχίζονται σε τρεις κατηγορίες εργασιών (A1, A2 και A3). Κάθε σημείο περιγράφεται με πλήρες σύνολο μεταδεδομένων, τα οποία καταγράφονται σε αρχείο Excel. Η απεικόνιση των αποτελεσμάτων επιτυγχάνεται με χρήση της βιβλιοθήκης Folium, με δυναμική ενημέρωση του χάρτη και οπτική διαφοροποίηση των σημείων ανάλογα με την ομάδα εργασίας στην οποία ανήκουν.

6. Επίλογος - Προτάσεις για Μελλοντική Έρευνα

Ο αλγόριθμος που αναπτύχθηκε στο πλαίσιο της παρούσας εργασίας συνιστά μια λειτουργική και αποδοτική λύση για την αντιμετώπιση του προβλήματος δρομολόγησης οχημάτων (VRP) σε πραγματικά δεδομένα. Παρέχει άμεσα εφαρμόσιμα αποτελέσματα, με δυνατότητα παραμετροποίησης και προσαρμογής στις ανάγκες μικρών και μεσαίων επιχειρήσεων. Ταυτόχρονα, η αρχιτεκτονική του είναι επεκτάσιμη, επιτρέποντας την ενσωμάτωση επιπλέον λειτουργιών που θα ενισχύσουν την ακρίβεια, την υπολογιστική αποδοτικότητα και τη διαχειριστική ευελιξία του συστήματος.

Η υλοποίηση βασίζεται σε μια στοχαστική μεταερευνητική προσέγγιση, συγκεκριμένα στον αλγόριθμο Guided Local Search της βιβλιοθήκης OR-Tools, ο οποίος εκκινεί κάθε φορά από διαφορετικό τυχαίο σημείο εντός του χώρου λύσεων. Καθώς το σημείο εκκίνησης μεταβάλλεται μεταξύ επαναλήψεων, ο αλγόριθμος παράγει διαφορετικές, αλλά ισοδύναμες από πλευράς κόστους, λύσεις. Η ιδιότητα αυτή εξασφαλίζει τη διερεύνηση ευρύτερου φάσματος λύσεων και συμβάλλει στην αποφυγή παγίδευσης σε τοπικά βέλτιστα. Ωστόσο, για τη συστηματική προσέγγιση της πραγματικά βέλτιστης λύσης, προτείνονται ορισμένες ενισχύσεις της διαδικασίας. Μεταξύ αυτών περιλαμβάνονται η εκτέλεση πολλαπλών ανεξάρτητων επαναλήψεων με διαφορετικούς αρχικούς σπόρους (seeds) και επιλογή της καλύτερης λύσης, η αύξηση του χρονικού ορίου εκτέλεσης, η χρήση εναλλακτικών τεχνικών όπως το Large Neighborhood Search για βαθύτερη εξερεύνηση του χώρου λύσεων, καθώς και η αξιοποίηση ήδη υπολογισμένων πινάκων αποστάσεων ή η αποθήκευση καλών λύσεων ως αρχικές (initial solutions) σε επόμενες εκτελέσεις.

Η στοχαστικότητα του αλγορίθμου, αντί να θεωρείται περιορισμός, συνιστά πλεονέκτημα. Επιτρέπει τη δυναμική προσαρμογή του συστήματος σε μεταβαλλόμενα δεδομένα—όπως η εισαγωγή νέων σημείων εξυπηρέτησης—χωρίς την ανάγκη πλήρους επανεπίλυσης. Παράλληλα, διευκολύνει την εύρεση εναλλακτικών δρομολογίων σε διαδοχικές εκτελέσεις, αυξάνοντας την ευρωστία του μοντέλου. Ως εκ τούτου, ο παρών αλγόριθμος αποτελεί αξιόπιστη βάση για καθημερινές επιχειρησιακές εφαρμογές, με υψηλή ποιότητα λύσεων ακόμη και όταν αυτές παρουσιάζουν μικρές διαφορές μεταξύ επαναλήψεων.

Για τη μελλοντική ενίσχυση του συστήματος, προτείνεται η ενσωμάτωση προηγμένων τεχνικών βελτιστοποίησης που θα αξιοποιούν δεδομένα σε πραγματικό χρόνο. Η ενσωμάτωση δυναμικών μοντέλων ανακατανομής διαδρομών, λαμβάνοντας υπόψη κυκλοφοριακές συνθήκες και μεταβαλλόμενη ζήτηση, θα μπορούσε να μειώσει σημαντικά τον χρόνο εξυπηρέτησης και να αυξήσει την επιχειρησιακή αποδοτικότητα. Παράλληλα, η εφαρμογή μεθόδων Μηχανικής Μάθησης δύναται να οδηγήσει στην αυτόματη αναγνώριση περιοχών αυξημένης σημασίας, είτε λόγω

χρονικών περιορισμών είτε λόγω συστηματικά υψηλής ζήτησης. Μέσω της ανάλυσης ιστορικών δεδομένων, το σύστημα θα μπορεί να προβλέπει σημεία που απαιτούν ενισχυμένη εξυπηρέτηση ή προτεραιοποίηση.

Ιδιαίτερη αξία θα είχε η ανάπτυξη εφαρμογής παρακολούθησης σε πραγματικό χρόνο, η οποία θα επιτρέπει την εποπτεία των διαδρομών, θα παρέχει ενημερώσεις για καθυστερήσεις ή απρόβλεπτα συμβάντα και θα συγκεντρώνει αναλυτικά στατιστικά δεδομένα για κάθε όχημα και διαδρομή. Η ύπαρξη βάσης δεδομένων με δείκτες απόδοσης, όπως το συνολικό κόστος, η μέση διανυθείσα απόσταση ανά όχημα και το ποσοστό εξυπηρετούμενων σημείων, θα προσφέρει τη δυνατότητα αντικειμενικής αξιολόγησης της απόδοσης του συστήματος και θα υποστηρίζει μελλοντικές παρεμβάσεις βελτιστοποίησης.

Επιπλέον, η παραμετροποίηση του συστήματος μέσω εξωτερικών αρχείων ή βάσεων δεδομένων, στα οποία θα ορίζονται κρίσιμες παράμετροι όπως χρονικά όρια, χωρητικότητες οχημάτων και κριτήρια κόστους, θα καταστήσει το μοντέλο πιο ευέλικτο και φιλικό προς τον τελικό χρήστη. Σε προβλήματα μεγαλύτερης κλίμακας, μπορεί να αξιοποιηθεί η έννοια των κρίσιμων περιοχών με διαφοροποιημένη βαρύτητα, έτσι ώστε η κάθε περιοχή να βαθμολογείται ανάλογα με τη σημασία της και να ενσωματώνεται στον στρατηγικό σχεδιασμό των διαδρομών.

Συνοψίζοντας, ο παρών κώδικας παρέχει μια αξιόπιστη και ευέλικτη πλατφόρμα δρομολόγησης, η οποία μπορεί να επεκταθεί με σκοπό να υποστηρίξει ακόμη πιο πολύπλοκα επιχειρησιακά σενάρια. Η ενσωμάτωση τεχνικών βελτιστοποίησης, ανάλυσης δεδομένων και διαδραστικών εφαρμογών εποπτείας συνθέτει ένα ισχυρό πλαίσιο για τη μετεξέλιξη του συστήματος σε ένα σύγχρονο, έξυπνο και πλήρως προσαρμοστικό εργαλείο διαχείρισης μεταφορών και πόρων.

Βιβλιογραφία

- [1] https://www.bio-conferences.org/articles/bioconf/pdf/2024/05/bioconf_rtbs2024_01093.pdf
- [2] Chopra, S., & Meindl, P. (2016). Supply Chain Management: Strategy, Planning, and Operation.
- [3] Mentzer, J. T. (2001). Supply Chain Management. Sage Publications.
- [4] Toth, Paolo, και Daniele Vigo. The Vehicle Routing Problem. Επιμέλεια Paolo Toth και Daniele Vigo. Society for Industrial and Applied Mathematics, 2002.
- [5] Konstantinos Gkiotsalitis, Public Transport Optimization, 2022.
- [6] Lenstra, J. K., και A. H. G. Rinnooy Kan. ‘Complexity of Vehicle Routing and Scheduling Problems’, 1981.
- [7] Stenger, A. (2012). The Prize-Collecting Vehicle Routing Problem with Non-Linear Cost – Integration of Subcontractors into Route Design of Small Package Shippers. Proceedings of the 1st International Conference on Operations Research and Enterprise Systems (ICORES 2012), pp. 265-273.
- [8] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3rd ed.). MIT Press.
- [9] Sahin M.K. & Yaman H. (2023) “The Vehicle Routing Problem with Access Restrictions”, Optimization Online, 29 σ.
- [10] Dantzig, G. B., Fulkerson, D. R., & Johnson, S. M. (1954). "Solution of a large-scale traveling-salesman problem". Journal of the Operations Research Society of America, 2(4), 393–410.
- [11] Kara İ. & Bektaş T. (2005) “Minimal Load Constrained Vehicle Routing Problems”, LNCS 3514, 188-195.
- [12] Kleinberg, J., & Tardos, É. (2005). *Algorithm Design*. Addison-Wesley.
- [13] Google Developers. (2024). OR-Tools Documentation. <https://developers.google.com/optimization>
- [14] Mitchell, S. D. (2020). PuLP: A Linear Programming Toolkit for Python. <https://coin-or.github.io/pulp/>
- [15] Vazirani, V. V. (2001). *Approximation Algorithms*. Springer.
- [16] Skiena, S. S. (2008). *The Algorithm Design Manual*. Springer.
- [17] Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. Computers & Operations Research, 34(8), 2403–2435.
- [18] <https://developers.google.com/optimization/routing/tsp>
- [19] https://developers.google.com/optimization/routing/routing_tasks
- [20] https://www.researchgate.net/publication/377359861_Supply_Chain_Optimization_in_Industry_50_An_Experimental_Investigation_Using_AI
- [21] JetBrains. (2024). PyCharm Documentation. <https://www.jetbrains.com/pycharm>
- [22] Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. Journal of the ACM, 7(4), 326–329.
- [23] Glover, F., & Kochenberger, G. A. (2003). Handbook of Metaheuristics.

- [24] Ropke, S., & Pisinger, D. (2006). An Adaptive Large Neighborhood Search heuristic for the pickup and delivery problem with time windows. *Transportation Science*.
- [25] Shahbazian, R. et al. (2023). Integrating Machine Learning into Vehicle Routing Problem: Methods and Applications. (IEEE Access, 2023).
- [26] Florio, A. M. et al. (2023). Recent Advances in Vehicle Routing with Stochastic Demands. (arXiv, 2023).
- [27] “A deep reinforcement learning approach for vehicle routing problems” (Elsevier, 2025)
- [28] Wouda, N. A. et al. (2023). PyVRP: a high-performance VRP solver package. (arXiv, 2023).
- [29] Liu, F. et al. (2023). Heuristics for Vehicle Routing Problem: A Survey and Recent Advances. (arXiv, 2023).
- [30] Bhandari, U. (2022). Solving Vehicle Routing Problem using Machine Learning based clustering and TSP cluster redistribution. (IJRR, 2022)
- [31] Yernar, A. & Turan, C. (2025). Recent developments in vehicle routing problem under time uncertainty: a comprehensive review. *Bulletin of EEI*.
- [32] The Vehicle-Routing Problem with Satellites Utilization. MDPI (2023)
- [33] Nazari, M. et al. (2018). Reinforcement Learning for Solving the Vehicle Routing Problem. (NeurIPS 2018)
- [34] A recent review of solution approaches for green vehicle routing (2016–2023). *Sci. Dir.*, 2024.
- [35] Sluijk, N. et al. (2023). Two-echelon vehicle routing problems: A literature review. (EJOR 2023).
- [36] MIT News (2021). Machine learning speeds up vehicle routing
- [37] Google OR-Tools Optimizing Routing and Scheduling Problems. Google Developers, 2024.
- [38] Ammon, S. et al. (2024). A Supervised Machine Learning Approach for the Vehicle Routing Problem. (Scitepress 2024)
- [39] “Recent Green VRP review (2016–2023)”, Yernar & Turan
- [40] Sørensen, K., & Glover, F. (2013). Metaheuristics. In M. Gendreau & J.-Y. Potvin (Eds.), *Handbook of Metaheuristics* (2nd ed., pp. 1–28). Springer.
- [41] Desaulniers, G., Desrosiers, J., & Solomon, M. M. (Eds.). (2016). *Column Generation*. Springer.

ΠΑΡΑΡΤΗΜΑ 1 – ΚΩΔΙΚΕΣ PYTHON

1.1 ΚΩΔΙΚΑΣ ΔΗΜΙΟΥΡΓΙΑΣ ΤΥΧΑΙΩΝ ΣΗΜΕΙΩΝ

```
import os
import random
import folium
import pandas as pd
from datetime import datetime
from geopy.geocoders import Nominatim
from geopy.exc import GeocoderTimedOut
import math
import time
import osmnx as ox
from openpyxl import load_workbook
from openpyxl.styles import PatternFill

EXCEL_FILE = 'points_data.xlsx'
MAP_FILE = 'map.html'
RUN_COUNT_FILE = 'run_count.txt'
MAX_TOTAL_POINTS = 90
MAX_A2_POINTS = 45
CENTER = (37.9838, 23.7275)
MAX_RADIUS_KM = 6
B1_RADIUS_KM = 2.5
G_TIMES = {'G1': 75, 'G2': 40, 'G3': 20}

geolocator = Nominatim(user_agent="point-generator")
G = ox.graph_from_point(CENTER, dist=MAX_RADIUS_KM*1000,
network_type='drive')

def load_or_create_excel():
    if os.path.exists(EXCEL_FILE):
        df = pd.read_excel(EXCEL_FILE)
        if 'Κατάσταση' not in df.columns:
            df['Κατάσταση'] = 'OPEN'
            df.to_excel(EXCEL_FILE, index=False)
        return df
    else:
        df = pd.DataFrame(columns=[
            'Α/Α', 'Πλάτος', 'Μήκος', 'Ομάδα Εργασίας',
            'Περιοχή', 'Χρόνος', 'Ημερομηνία', 'Κατάσταση'
        ])
        df.to_excel(EXCEL_FILE, index=False)
        return df

def load_or_create_run_count():
    if not os.path.exists(RUN_COUNT_FILE):
        with open(RUN_COUNT_FILE, 'w') as f:
            f.write('0')
    with open(RUN_COUNT_FILE, 'r') as f:
        return int(f.read().strip())

def increment_run_count():
    count = load_or_create_run_count() + 1
    with open(RUN_COUNT_FILE, 'w') as f:
        f.write(str(count))
    return count
```

```

def classify_region(lat, lon):
    dist = ox.distance.great_circle(lat, lon, CENTER[0], CENTER[1])
    if dist <= B1_RADIUS_KM * 1000:
        return 'B1'
    elif dist <= MAX_RADIUS_KM * 1000:
        return 'B2'
    else:
        return None

def is_accessible(lat, lon):
    try:
        location = geolocator.reverse((lat, lon), exactly_one=True,
        timeout=10)
        if location and location.raw.get("address", {}).get("road"):
            return True
    except GeocoderTimedOut:
        time.sleep(1)
        return is_accessible(lat, lon)
    return False

def generate_random_point():
    while True:
        node = random.choice(list(G.nodes))
        point = (G.nodes[node]['y'], G.nodes[node]['x'])
        if is_accessible(*point):
            region = classify_region(*point)
            if region:
                return point[0], point[1], region

def generate_points(existing_df, run_count):
    today = (datetime.today() + pd.Timedelta(days=run_count -
1)).strftime('%Y-%m-%d')
    open_df = existing_df[existing_df['Κατάσταση'] == 'OPEN']
    total_points = len(open_df)
    a2_count = len(open_df[open_df['Ομάδα Εργασίας'] == 'A2'])

    if total_points >= MAX_TOTAL_POINTS:
        return pd.DataFrame()

    a2_today = random.randint(12, min(20, MAX_A2_POINTS - a2_count))
    a2_g1 = int(a2_today * 0.3)
    a2_g2 = a2_today - a2_g1
    a3_today = min(10, MAX_TOTAL_POINTS - total_points - a2_today)
    a1_today = min(4, MAX_TOTAL_POINTS - total_points - a2_today -
a3_today)

    points = (['A1'] * a1_today) + (['A2'] * a2_today) + (['A3'] *
a3_today)
    random.shuffle(points)

    index = len(existing_df) + 1
    new_rows = []

    for group in points:
        if len(open_df) + len(new_rows) >= MAX_TOTAL_POINTS:
            break

        if group == 'A2':
            time_group = 'G1' if a2_g1 > 0 else 'G2'
            if time_group == 'G1':

```

```

        a2_g1 -= 1
    else:
        a2_g2 -= 1
    elif group == 'A3':
        time_group = 'G3'
    else:
        time_group = random.choice(['G1', 'G2', 'G3'])

    lat, lon, region = generate_random_point()

    new_rows.append({
        'A/A': index,
        'Πλάτος': lat,
        'Μήκος': lon,
        'Ομάδα Εργασίας': group,
        'Περιοχή': region,
        'Χρόνος': G_TIMES[time_group],
        'Ημερομηνία': today,
        'Κατάσταση': 'OPEN'
    })
    index += 1

    return pd.DataFrame(new_rows)

def update_excel(new_data):
    df = load_or_create_excel()
    updated_df = pd.concat([df, new_data], ignore_index=True)
    updated_df.to_excel(EXCEL_FILE, index=False)
    wb = load_workbook(EXCEL_FILE)
    ws = wb.active

    status_col_index = None
    for i, cell in enumerate(ws[1], 1):
        if cell.value == 'Κατάσταση':
            status_col_index = i
            break

    if status_col_index:
        gray_fill = PatternFill(start_color="D3D3D3",
                                end_color="D3D3D3", fill_type="solid")
        for row in ws.iter_rows(min_row=2, max_row=ws.max_row):
            if row[status_col_index - 1].value == "CLOSED":
                for cell in row:
                    cell.fill = gray_fill
    wb.save(EXCEL_FILE)

def update_map():
    df = pd.read_excel(EXCEL_FILE)
    df = df[df['Κατάσταση'] == 'OPEN']
    m = folium.Map(location=CENTER, zoom_start=13)
    color_map = {'A1': 'red', 'A2': 'blue', 'A3': 'green'}

    for _, row in df.iterrows():
        folium.CircleMarker(
            location=(row['Πλάτος'], row['Μήκος']),
            radius=5,
            popup=f"{row['Ομάδα Εργασίας']} - {row['Περιοχή']} - {row['Χρόνος']} λεπτά - {row['Ημερομηνία']}",
            color=color_map.get(row['Ομάδα Εργασίας'], 'gray'),
            fill=True,
            fill_opacity=0.7

```

```

        ).add_to(m)

    m.save(MAP_FILE)

def main():
    run_count = increment_run_count()
    existing_data = load_or_create_excel()
    new_data = generate_points(existing_data, run_count)
    if not new_data.empty:
        update_excel(new_data)
        print(f"☑ Προστέθηκαν {len(new_data)} νέα σημεία.")
    else:
        print("❌ Δεν προστέθηκαν νέα σημεία.")
    update_map()
    print("☐ Ο χάρτης ενημερώθηκε.")

if __name__ == "__main__":
    main()

```

1.2 ΚΩΔΙΚΑΣ ΣΥΓΚΡΙΣΗΣ PuLP

```
import pandas as pd
import numpy as np
from pulp import *
import folium

# Διαβάζει δεδομένα
df = pd.read_excel("points_data.xlsx")
open_points = df[df["Κατάσταση"] == "OPEN"].reset_index(drop=True)

# Depot + σημεία
yc = [37.9811] + open_points["Πλάτος"].tolist()
xc = [23.7497] + open_points["Μήκος"].tolist()
d = [0] + open_points["Χρόνος"].tolist()

n = len(open_points)
V = [i for i in range(n + 1)]
A = [(i, j) for i in V for j in V if i != j]
c = {(i, j): np.hypot(xc[i] - xc[j], yc[i] - yc[j]) for i in V for j
in V}
M = 1000000
for i in V:
    for j in V:
        if i == j:
            c[i, j] = M

K = 3
C = 420

# Μοντέλο CVRP
model = LpProblem("CVRP_formulation", LpMinimize)
x = LpVariable.dicts("x", ((i, j) for i in V for j in V),
cat='Binary')
u = LpVariable.dicts("u", (i for i in V), cat='Continuous')

model += lpSum(c[i, j] * x[i, j] for i in V for j in V)

for j in V:
    if j != 0:
        model += lpSum(x[i, j] for i in V) == 1
for i in V:
    if i != 0:
        model += lpSum(x[i, j] for j in V) == 1
model += lpSum(x[0, j] for j in V if j != 0) <= K
model += lpSum(x[i, 0] for i in V if i != 0) == lpSum(x[0, j] for j
in V if j != 0)

for i in V:
    for j in V:
        if i != j and i != 0 and j != 0:
            model += u[i] - u[j] + C * x[i, j] <= C - d[j]
for i in V:
    if i != 0:
        model += u[i] <= C
        model += d[i] <= u[i]

# Λύση
solver = PULP_CBC_CMD(msg=True)
```

```

model.solve(solver)

all_routes = []
visited_indices = set()

def find_next_node(current_node):
    for j in V:
        if current_node != j and x[current_node, j].varValue > 0.5:
            return j
    return None

for j in V:
    if x[0, j].varValue > 0.5:
        route = [0, j]
        current = j
        while True:
            next_node = find_next_node(current)
            if next_node is None or next_node == 0:
                break
            route.append(next_node)
            current = next_node
        route.append(0)
        all_routes.append(route)
        visited_indices.update(route[1:-1])

routes = all_routes

# Ανανεώνει Excel με CLOSED
updated_df = df.copy()
for i in visited_indices:
    updated_df.loc[open_points.index[i - 1], "Κατάσταση"] = "CLOSED"
updated_df.to_excel("updated_points_data.xlsx", index=False)

# Excel με διαδρομές
route_details = []
for i, route in enumerate(routes):
    for order, node in enumerate(route):
        route_details.append({
            "Vehicle": i + 1,
            "Stop Order": order,
            "Latitude": yc[node],
            "Longitude": xc[node],
            "Demand": d[node],
            "Original Index": node if node == 0 else
open_points.index[node - 1]
        })
pd.DataFrame(route_details).to_excel("vehicle_routes.xlsx",
index=False)

# Χάρτης
m = folium.Map(location=[yc[0], xc[0]], zoom_start=13)
folium.Marker([yc[0], xc[0]], popup="Depot",
icon=folium.Icon(color="red")).add_to(m)

colors = ['red', 'blue', 'green', 'purple', 'orange', 'darkred',
'lightred', 'beige', 'darkblue', 'darkgreen', 'cadetblue',
'darkpurple', 'white', 'pink', 'lightblue', 'lightgreen',
'gray', 'black', 'lightgray']

for i, route in enumerate(routes):
    coords = [[yc[node], xc[node]] for node in route]

```

```

        folium.PolyLine(coords, color=colors[i % len(colors)], weight=5,
                        tooltip=f"Vehicle {i+1}").add_to(m)
    for node in route[1:-1]:
        folium.Marker([yc[node], xc[node]],
                      popup=f"Stop {node}",
                      icon=folium.Icon(color='blue')).add_to(m)

m.save("cvrp_routes_map.html")
print("☑ Ολοκληρώθηκε. Δημιουργήθηκαν τα αρχεία:")
print("- updated_points_data.xlsx")
print("- vehicle_routes.xlsx")
print("- cvrp_routes_map.html")

```

1.3 ΚΩΔΙΚΑΣ ΣΥΓΚΡΙΣΗΣ ORTOOLS

```
import os
import pandas as pd
from datetime import timedelta
from geopy.distance import geodesic
import folium
from openpyxl import load_workbook
from openpyxl.styles import PatternFill
from ortools.constraint_solver import pywrapcp, routing_enums_pb2
import numpy as np

EXCEL_FILE = "points_data.xlsx"
RUN_COUNT_FILE = "run_count_routing.txt"
DEPOT = (37.9811, 23.7497)
ATHENS_CENTER = (37.9755, 23.7348)
MAX_CAPACITY = 420
MIN_USED = 0
NUM_VEHICLES = 3
COLOR_ZONE = {"B1": "red", "B2": "blue"}
colors = ["green", "orange", "purple"]

if not os.path.exists(RUN_COUNT_FILE):
    with open(RUN_COUNT_FILE, "w") as f:
        f.write("0")
with open(RUN_COUNT_FILE, "r") as f:
    run_count = int(f.read().strip()) + 1
with open(RUN_COUNT_FILE, "w") as f:
    f.write(str(run_count))

LOG_FILE = f"routing_solver_log_run{run_count}.txt"

# Διαβάζουμε δεδομένα
df = pd.read_excel(EXCEL_FILE)
df_open = df[df["Κατάσταση"] == "OPEN"].copy().reset_index(drop=True)
df_open["coords"] = list(zip(df_open["Πλάτος"], df_open["Μήκος"]))
df_open["assigned"] = False

# Βαρύτητα με βάση ομάδα εργασίας (ίση για A1, A2, A3)
weight_map = {"A1": 1.0, "A2": 1.0, "A3": 1.0}
df_open["weight"] = df_open["Ομάδα Εργασίας"].map(weight_map).fillna(0.1)

# Δημιουργία πίνακα αποστάσεων
all_points = [DEPOT] + df_open["coords"].tolist()
def compute_euclidean_matrix(locations):
    size = len(locations)
    matrix = np.zeros((size, size))
    for i in range(size):
        for j in range(size):
            if i != j:
                matrix[i][j] = geodesic(locations[i],
locations[j]).km
    return matrix
distance_matrix = compute_euclidean_matrix(all_points)
demands = [0] + df_open["Χρόνος"].tolist()

# OR-Tools Routing Model
manager = pywrapcp.RoutingIndexManager(len(distance_matrix),
NUM_VEHICLES, 0)
routing = pywrapcp.RoutingModel(manager)
```

```

def distance_callback(from_idx, to_idx):
    from_node = manager.IndexToNode(from_idx)
    to_node = manager.IndexToNode(to_idx)
    base_dist = distance_matrix[from_node][to_node]
    weight = df_open.iloc[to_node - 1]["weight"] if to_node > 0 else
1.0
    return int(base_dist * 1000 / weight)
transit_callback_idx =
routing.RegisterTransitCallback(distance_callback)
routing.SetArcCostEvaluatorOfAllVehicles(transit_callback_idx)

def demand_callback(from_idx):
    from_node = manager.IndexToNode(from_idx)
    return demands[from_node]
demand_callback_idx =
routing.RegisterUnaryTransitCallback(demand_callback)
routing.AddDimensionWithVehicleCapacity(
    demand_callback_idx, 0, [MAX_CAPACITY] * NUM_VEHICLES, True,
"Capacity")

# Ισοτίμες ποινές για όλα τα σημεία
for idx in range(len(df_open)):
    routing.AddDisjunction([manager.NodeToIndex(idx + 1)], 50000)

# Ρυθμίσεις επίλυσης
search_params = pywrapcp.DefaultRoutingSearchParameters()
search_params.first_solution_strategy =
routing_enums_pb2.FirstSolutionStrategy.PATH_CHEAPEST_ARC
search_params.local_search_metaheuristic =
routing_enums_pb2.LocalSearchMetaheuristic.GUIDED_LOCAL_SEARCH
search_params.time_limit.seconds = 180
search_params.log_search = True

with open(LOG_FILE, "w") as log_f:
    solution = routing.SolveWithParameters(search_params)
    if solution:
        log_f.write("\n--- ROUTING SOLUTION FOUND ---\n")
        log_f.write(str(solution))
    else:
        log_f.write("\n--- NO SOLUTION FOUND ---\n")

vehicle_routes = []
vehicle_distances = []
vehicle_loads = []

if solution:
    for vehicle_id in range(NUM_VEHICLES):
        index = routing.Start(vehicle_id)
        route = []
        route_distance = 0
        used = 0
        prev_index = None
        while not routing.IsEnd(index):
            node_index = manager.IndexToNode(index)
            if prev_index is not None:
                route_distance +=
distance_matrix[manager.IndexToNode(prev_index)][node_index]
                if node_index != 0:
                    route.append(node_index - 1)
                    used += demands[node_index]

```

```

        prev_index = index
        index = solution.Value(routing.NextVar(index))
    if used >= MIN_USED:
        for idx in route:
            df_open.at[idx, "assigned"] = True
            vehicle_routes.append(route)
            vehicle_distances.append(route_distance)
            vehicle_loads.append(used)

def optimize_subroute(route):
    if not route:
        return []
    locations = [DEPOT] + [df_open.loc[i, "coords"] for i in route] + [DEPOT]
    matrix = compute_euclidean_matrix(locations)
    tsp_manager = pywrapcp.RoutingIndexManager(len(matrix), 1, 0)
    tsp_routing = pywrapcp.RoutingModel(tsp_manager)

    def tsp_cb(i, j):
        if i < len(matrix) and j < len(matrix):
            return int(matrix[i][j] * 1000)
        else:
            return 1000000
    cb_idx = tsp_routing.RegisterTransitCallback(tsp_cb)
    tsp_routing.SetArcCostEvaluatorOfAllVehicles(cb_idx)

    tsp_params = pywrapcp.DefaultRoutingSearchParameters()
    tsp_params.first_solution_strategy =
routing_enums_pb2.FirstSolutionStrategy.PATH_CHEAPEST_ARC
    solution = tsp_routing.SolveWithParameters(tsp_params)

    ordered = []
    if solution:
        idx = tsp_routing.Start(0)
        while not tsp_routing.IsEnd(idx):
            node = tsp_manager.IndexToNode(idx)
            if node not in [0, len(locations) - 1]:
                ordered.append(route[node - 1])
            idx = solution.Value(tsp_routing.NextVar(idx))
    return ordered

vehicle_routes = [optimize_subroute(route) for route in
vehicle_routes]

for i, route in enumerate(vehicle_routes):
    df_out = df_open.loc[route].copy()
    df_out["A/A"] = range(1, len(df_out) + 1)
    df_out.insert(1, "Αρ. Σημείου", route)
    keep = ["A/A", "Αρ. Σημείου", "Πλάτος", "Μήκος", "Ομάδα
Εργασίας", "Περιοχή", "Χρόνος", "Ημερομηνία"]
    df_out[keep].to_excel(f"vehicle_{i+1}_route_run{run_count}.xlsx",
index=False)

wb = load_workbook(EXCEL_FILE)
ws = wb.active
for route in vehicle_routes:
    for idx in route:
        lat = df_open.loc[idx, "Πλάτος"]
        excel_row = df.index[df["Πλάτος"] == lat][0] + 2
        ws[f'H{excel_row}'] = "CLOSED"
        fill = PatternFill(start_color="DDDDDD", end_color="DDDDDD",

```

```

fill_type="solid")
    for col in range(1, ws.max_column + 1):
        ws.cell(row=excel_row, column=col).fill = fill
wb.save(EXCEL_FILE)

def create_map(vehicle_routes, df_open, filename):
    m = folium.Map(location=ATHENS_CENTER, zoom_start=12)
    folium.Circle(ATHENS_CENTER, radius=2500, color="red", fill=True,
fill_opacity=0.1).add_to(m)
    folium.Circle(ATHENS_CENTER, radius=6000, color="blue",
fill=True, fill_opacity=0.1).add_to(m)
    folium.Marker(DEPOT, icon=folium.Icon(color="black"),
popup="Depot").add_to(m)
    for i, route in enumerate(vehicle_routes):
        df_r = df_open.loc[route]
        coords = [DEPOT] + df_r["coords"].tolist() + [DEPOT]
        folium.PolyLine(coords, color=colors[i % 3],
weight=4).add_to(m)
        for _, row in df_r.iterrows():
            folium.CircleMarker(location=row["coords"], radius=5,
                                color=colors[i % 3], fill=True,
fill_opacity=0.7).add_to(m)
    m.save(filename)

create_map(vehicle_routes, df_open,
f"vehicle_routes_combined_run{run_count}.html")
for i, route in enumerate(vehicle_routes):
    create_map([route], df_open.loc[route],
f"vehicle_{i+1}_map_run{run_count}.html")

if vehicle_routes:
    summary_df = pd.DataFrame({
        "Οχήμα": [f"Vehicle {i+1}" for i in
range(len(vehicle_routes))],
        "Αριθμός Σημείων": [len(route) for route in vehicle_routes],
        "Συνολική Απόσταση (km)": vehicle_distances,
        "Συνολικό Φορτίο (min)": vehicle_loads
    })

summary_df.to_excel(f"vehicle_routes_summary_run{run_count}.xlsx",
index=False)

print(f"☒ Ολοκληρώθηκε η επανάληψη #{run_count}. Δημιουργήθηκαν
Excel, ενημερώθηκε το αρχείο points_data.xlsx, χάρτες και
στατιστικά.")

```

1.4 ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ ΑΣΚΗΣΗ ORTOOLS

```
import os
import pandas as pd
from datetime import timedelta
from geopy.distance import geodesic
import folium
from openpyxl import load_workbook
from openpyxl.styles import PatternFill
from ortools.constraint_solver import pywrapcp, routing_enums_pb2
import numpy as np

EXCEL_FILE = "points_data.xlsx"
RUN_COUNT_FILE = "run_count_routing.txt"
DEPOT = (37.9811, 23.7497)
ATHENS_CENTER = (37.9755, 23.7348)
MAX_CAPACITY = 420
MIN_USED = 330
NUM_VEHICLES = 3
COLOR_ZONE = {"B1": "red", "B2": "blue"}
colors = ["green", "orange", "purple"]

if not os.path.exists(RUN_COUNT_FILE):
    with open(RUN_COUNT_FILE, "w") as f:
        f.write("0")
with open(RUN_COUNT_FILE, "r") as f:
    run_count = int(f.read().strip()) + 1
with open(RUN_COUNT_FILE, "w") as f:
    f.write(str(run_count))

LOG_FILE = f"routing_solver_log_run{run_count}.txt"

# Διαβάζουμε δεδομένα
df = pd.read_excel(EXCEL_FILE)
df_open = df[df["Κατάσταση"] == "OPEN"].copy().reset_index(drop=True)
df_open["coords"] = list(zip(df_open["Πλάτος"], df_open["Μήκος"]))
df_open["assigned"] = False

# Βαρύτητα με βάση ομάδα εργασίας
weight_map = {"A1": 1.0, "A2": 0.75, "A3": 0.5}
df_open["weight"] = df_open["Ομάδα Εργασίας"].map(weight_map).fillna(0.1)

# Δημιουργία πίνακα αποστάσεων
all_points = [DEPOT] + df_open["coords"].tolist()
def compute_euclidean_matrix(locations):
    size = len(locations)
    matrix = np.zeros((size, size))
    for i in range(size):
        for j in range(size):
            if i != j:
                matrix[i][j] = geodesic(locations[i],
locations[j]).km
    return matrix

distance_matrix = compute_euclidean_matrix(all_points)
demands = [0] + df_open["Χρόνος"].tolist()

# OR-Tools Routing Model
```

```

manager = pywrapcp.RoutingIndexManager(len(distance_matrix),
NUM_VEHICLES, 0)
routing = pywrapcp.RoutingModel(manager)

def distance_callback(from_idx, to_idx):
    from_node = manager.IndexToNode(from_idx)
    to_node = manager.IndexToNode(to_idx)
    base_dist = distance_matrix[from_node][to_node]
    weight = df_open.iloc[to_node - 1]["weight"] if to_node > 0 else
1.0
    return int(base_dist * 1000 / weight)

transit_callback_idx =
routing.RegisterTransitCallback(distance_callback)
routing.SetArcCostEvaluatorOfAllVehicles(transit_callback_idx)

def demand_callback(from_idx):
    from_node = manager.IndexToNode(from_idx)
    return demands[from_node]

demand_callback_idx =
routing.RegisterUnaryTransitCallback(demand_callback)
routing.AddDimensionWithVehicleCapacity(
    demand_callback_idx, 0, [MAX_CAPACITY] * NUM_VEHICLES, True,
"Capacity")

# Προσθήκη περιορισμού συνολικού χρόνου εξυπηρέτησης για περιοχή B1
(max 180 λεπτά)
def b1_time_callback(from_idx):
    from_node = manager.IndexToNode(from_idx)
    if from_node == 0:
        return 0
    if df_open.iloc[from_node - 1]["Περιοχή"] == "B1":
        return demands[from_node]
    return 0

b1_time_callback_idx =
routing.RegisterUnaryTransitCallback(b1_time_callback)
routing.AddDimension(
    b1_time_callback_idx,
    0, # no slack
    180, # max B1 service time per vehicle
    True, # start cumul at zero
    "B1Time"
)

# Υποχρεωτική κάλυψη A1
for idx, row in df_open.iterrows():
    penalty = 100000 if row["Ομάδα Εργασίας"] == "A1" else 50000 if
row["Ομάδα Εργασίας"] == "A2" else 10000
    routing.AddDisjunction([manager.NodeToIndex(idx + 1)], penalty)

# Ρυθμίσεις επίλυσης
search_params = pywrapcp.DefaultRoutingSearchParameters()
search_params.first_solution_strategy =
routing_enums_pb2.FirstSolutionStrategy.PATH_CHEAPEST_ARC
search_params.local_search_metaheuristic =
routing_enums_pb2.LocalSearchMetaheuristic.GUIDED_LOCAL_SEARCH
search_params.time_limit.seconds = 180
search_params.log_search = True

```

```

with open(LOG_FILE, "w") as log_f:
    solution = routing.SolveWithParameters(search_params)
    if solution:
        log_f.write("\n--- ROUTING SOLUTION FOUND ---\n")
        log_f.write(str(solution))
    else:
        log_f.write("\n--- NO SOLUTION FOUND ---\n")

vehicle_routes = []
vehicle_distances = []
vehicle_loads = []

if solution:
    for vehicle_id in range(NUM_VEHICLES):
        index = routing.Start(vehicle_id)
        route = []
        route_distance = 0
        used = 0
        prev_index = None
        while not routing.IsEnd(index):
            node_index = manager.IndexToNode(index)
            if prev_index is not None:
                route_distance +=
distance_matrix[manager.IndexToNode(prev_index)][node_index]
            if node_index != 0:
                route.append(node_index - 1)
                used += demands[node_index]
            prev_index = index
            index = solution.Value(routing.NextVar(index))
        if used >= MIN_USED:
            for idx in route:
                df_open.at[idx, "assigned"] = True
            vehicle_routes.append(route)
            vehicle_distances.append(route_distance)
            vehicle_loads.append(used)

def optimize_subroute(route):
    if not route:
        return []
    locations = [DEPOT] + [df_open.loc[i, "coords"] for i in route] +
[DEPOT]
    matrix = compute_euclidean_matrix(locations)
    tsp_manager = pywrapcp.RoutingIndexManager(len(matrix), 1, 0)
    tsp_routing = pywrapcp.RoutingModel(tsp_manager)

    def tsp_cb(i, j):
        if i < len(matrix) and j < len(matrix):
            return int(matrix[i][j] * 1000)
        else:
            return 1000000
    cb_idx = tsp_routing.RegisterTransitCallback(tsp_cb)
    tsp_routing.SetArcCostEvaluatorOfAllVehicles(cb_idx)

    tsp_params = pywrapcp.DefaultRoutingSearchParameters()
    tsp_params.first_solution_strategy =
routing_enums_pb2.FirstSolutionStrategy.PATH_CHEAPEST_ARC
    solution = tsp_routing.SolveWithParameters(tsp_params)

    ordered = []
    if solution:
        idx = tsp_routing.Start(0)

```

```

        while not tsp_routing.IsEnd(idx):
            node = tsp_manager.IndexToNode(idx)
            if node not in [0, len(locations) - 1]:
                ordered.append(route[node - 1])
            idx = solution.Value(tsp_routing.NextVar(idx))
        return ordered

vehicle_routes = [optimize_subroute(route) for route in
vehicle_routes]

for i, route in enumerate(vehicle_routes):
    df_out = df_open.loc[route].copy()
    df_out["A/A"] = range(1, len(df_out) + 1)
    df_out.insert(1, "Αρ. Σημείου", route)
    keep = ["A/A", "Αρ. Σημείου", "Πλάτος", "Μήκος", "Ομάδα
Εργασίας", "Περιοχή", "Χρόνος", "Ημερομηνία"]
    df_out[keep].to_excel(f"vehicle_{i+1}_route_run{run_count}.xlsx",
index=False)

wb = load_workbook(EXCEL_FILE)
ws = wb.active
for route in vehicle_routes:
    for idx in route:
        lat = df_open.loc[idx, "Πλάτος"]
        excel_row = df.index[df["Πλάτος"] == lat][0] + 2
        ws[f'H{excel_row}'] = "CLOSED"
        fill = PatternFill(start_color="DDDDDD", end_color="DDDDDD",
fill_type="solid")
        for col in range(1, ws.max_column + 1):
            ws.cell(row=excel_row, column=col).fill = fill
wb.save(EXCEL_FILE)

def create_map(vehicle_routes, df_open, filename):
    m = folium.Map(location=ATHENS_CENTER, zoom_start=12)
    folium.Circle(ATHENS_CENTER, radius=2500, color="red", fill=True,
fill_opacity=0.1).add_to(m)
    folium.Circle(ATHENS_CENTER, radius=6000, color="blue",
fill=True, fill_opacity=0.1).add_to(m)
    folium.Marker(DEPOT, icon=folium.Icon(color="black"),
popup="Depot").add_to(m)
    for i, route in enumerate(vehicle_routes):
        df_r = df_open.loc[route]
        coords = [DEPOT] + df_r["coords"].tolist() + [DEPOT]
        folium.PolyLine(coords, color=colors[i % 3],
weight=4).add_to(m)
        for _, row in df_r.iterrows():
            folium.CircleMarker(location=row["coords"], radius=5,
                                color=colors[i % 3], fill=True,
fill_opacity=0.7).add_to(m)
    m.save(filename)

create_map(vehicle_routes, df_open,
f"vehicle_routes_combined_run{run_count}.html")
for i, route in enumerate(vehicle_routes):
    create_map([route], df_open.loc[route],
f"vehicle_{i+1}_map_run{run_count}.html")

if vehicle_routes:
    summary_df = pd.DataFrame({
        "Οχημα": [f"Vehicle {i+1}" for i in
range(len(vehicle_routes))],

```

```

        "Αριθμός Σημείων": [len(route) for route in vehicle_routes],
        "Συνολική Απόσταση (km)": vehicle_distances,
        "Συνολικό Φορτίο (min)": vehicle_loads
    })

summary_df.to_excel(f"vehicle_routes_summary_run{run_count}.xlsx",
index=False)

print(f"☒ Ολοκληρώθηκε η επανάληψη #{run_count}. Δημιουργήθηκαν
Excel, ενημερώθηκε το αρχείο points_data.xlsx, χάρτες και
στατιστικά.")

```

ΠΑΡΑΡΤΗΜΑ 2 – ΠΙΝΑΚΕΣ

ΕΠΑΝΑΛΗΨΕΩΝ

2.1 ΠΙΝΑΚΑΣ ΣΤΗΝ ΑΡΧΗ ΤΗΣ 3^{ΗΣ} ΕΠΑΝΑΛΗΨΗΣ

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9634817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	CLOSED
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,0338068	23,7091981	A2	B2	75	2025-05-24	CLOSED
33	37,9400333	23,7544334	A2	B2	75	2025-05-24	CLOSED
34	37,9842731	23,6731588	A1	B2	20	2025-05-24	CLOSED
35	37,9995306	23,682598	A2	B2	75	2025-05-24	CLOSED
36	37,9969159	23,7803971	A2	B2	75	2025-05-24	CLOSED
37	37,9882392	23,6755466	A3	B2	20	2025-05-24	CLOSED
38	37,9743537	23,6916898	A2	B2	75	2025-05-24	CLOSED
39	37,9664718	23,7327596	A2	B1	40	2025-05-24	CLOSED
40	37,9505651	23,7275189	A2	B2	40	2025-05-24	CLOSED
41	38,0053307	23,7757093	A3	B2	20	2025-05-24	CLOSED
42	37,9619506	23,7015338	A3	B2	20	2025-05-24	CLOSED
43	37,9886403	23,7677957	A2	B2	40	2025-05-24	CLOSED
44	37,943421	23,7299333	A2	B2	40	2025-05-24	CLOSED
45	37,9518225	23,7613104	A3	B2	20	2025-05-24	CLOSED
46	38,002855	23,674653	A3	B2	20	2025-05-24	CLOSED

47	37,9437648	23,7027373	A2	B2	40	2025-05-24	CLOSED
48	37,96635	23,7671193	A3	B2	20	2025-05-24	CLOSED
49	37,9597588	23,6772071	A3	B2	20	2025-05-24	CLOSED
50	37,9845642	23,7124659	A3	B1	20	2025-05-24	CLOSED
51	37,9952524	23,7720967	A1	B2	40	2025-05-24	CLOSED
52	37,9785149	23,7462318	A2	B1	40	2025-05-24	CLOSED
53	37,9970325	23,7351688	A3	B1	20	2025-05-24	CLOSED
54	37,9546785	23,7247726	A3	B2	20	2025-05-24	CLOSED
55	37,99191	23,7663477	A2	B2	40	2025-05-24	CLOSED
56	37,9976946	23,7350731	A2	B1	40	2025-05-24	CLOSED
57	37,9917038	23,7312294	A2	B1	40	2025-05-24	CLOSED
58	37,9498422	23,7506832	A2	B2	40	2025-05-24	CLOSED
59	37,9774362	23,6636983	A1	B2	20	2025-05-24	CLOSED
60	37,9939115	23,6690323	A1	B2	40	2025-05-24	CLOSED
61	37,977176	23,7099784	A2	B1	40	2025-05-24	CLOSED
62	37,9837138	23,7825302	A2	B2	40	2025-05-24	CLOSED
63	38,0065378	23,7375275	A2	B2	40	2025-05-24	CLOSED
64	38,0025799	23,722796	A3	B1	20	2025-05-25	OPEN
65	38,0204165	23,7416259	A3	B2	20	2025-05-25	OPEN
66	38,0125403	23,7362023	A1	B2	75	2025-05-25	OPEN
67	37,9836954	23,7399425	A3	B1	20	2025-05-25	OPEN
68	38,014754	23,6828747	A1	B2	75	2025-05-25	OPEN
69	37,9491741	23,7712022	A3	B2	20	2025-05-25	OPEN
70	37,9541015	23,7330198	A2	B2	75	2025-05-25	OPEN
71	38,0239282	23,73086	A2	B2	75	2025-05-25	OPEN
72	37,9574949	23,7514015	A3	B2	20	2025-05-25	OPEN
73	37,9679271	23,7611341	A2	B2	75	2025-05-25	OPEN
74	37,9981354	23,775022	A1	B2	75	2025-05-25	OPEN
75	37,9390389	23,7556215	A2	B2	75	2025-05-25	OPEN
76	37,9875477	23,7690288	A2	B2	40	2025-05-25	OPEN
77	38,020017	23,7121164	A2	B2	40	2025-05-25	OPEN
78	38,0057368	23,6675468	A3	B2	20	2025-05-25	OPEN
79	37,956039	23,7510721	A2	B2	40	2025-05-25	OPEN
80	37,9655982	23,7433831	A2	B1	40	2025-05-25	OPEN
81	38,035337	23,7285907	A2	B2	40	2025-05-25	OPEN
82	37,9464925	23,7017619	A2	B2	40	2025-05-25	OPEN
83	38,0117907	23,715673	A3	B2	20	2025-05-25	OPEN
84	37,9424264	23,7250789	A3	B2	20	2025-05-25	OPEN
85	38,0129063	23,6938307	A2	B2	40	2025-05-25	OPEN
86	37,9562468	23,7277496	A3	B2	20	2025-05-25	OPEN
87	37,9417008	23,7275471	A2	B2	40	2025-05-25	OPEN
88	37,994624	23,6640273	A2	B2	40	2025-05-25	OPEN
89	38,0053228	23,6972949	A2	B2	40	2025-05-25	OPEN
90	37,9891989	23,7306058	A3	B1	20	2025-05-25	OPEN
91	38,0305623	23,7093367	A2	B2	40	2025-05-25	OPEN
92	37,9558837	23,7253656	A1	B2	20	2025-05-25	OPEN

2.2 ΠΙΝΑΚΑΣ ΣΤΟ ΤΕΛΟΣ ΤΗΣ 3^{ΗΣ} ΕΠΑΝΑΛΗΨΗΣ

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9634817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	CLOSED
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,0338068	23,7091981	A2	B2	75	2025-05-24	CLOSED
33	37,9400333	23,7544334	A2	B2	75	2025-05-24	CLOSED
34	37,9842731	23,6731588	A1	B2	20	2025-05-24	CLOSED
35	37,9995306	23,682598	A2	B2	75	2025-05-24	CLOSED
36	37,9969159	23,7803971	A2	B2	75	2025-05-24	CLOSED
37	37,9882392	23,6755466	A3	B2	20	2025-05-24	CLOSED
38	37,9743537	23,6916898	A2	B2	75	2025-05-24	CLOSED
39	37,9664718	23,7327596	A2	B1	40	2025-05-24	CLOSED
40	37,9505651	23,7275189	A2	B2	40	2025-05-24	CLOSED
41	38,0053307	23,7757093	A3	B2	20	2025-05-24	CLOSED
42	37,9619506	23,7015338	A3	B2	20	2025-05-24	CLOSED
43	37,9886403	23,7677957	A2	B2	40	2025-05-24	CLOSED
44	37,943421	23,7299333	A2	B2	40	2025-05-24	CLOSED
45	37,9518225	23,7613104	A3	B2	20	2025-05-24	CLOSED
46	38,002855	23,674653	A3	B2	20	2025-05-24	CLOSED

47	37,9437648	23,7027373	A2	B2	40	2025-05-24	CLOSED
48	37,96635	23,7671193	A3	B2	20	2025-05-24	CLOSED
49	37,9597588	23,6772071	A3	B2	20	2025-05-24	CLOSED
50	37,9845642	23,7124659	A3	B1	20	2025-05-24	CLOSED
51	37,9952524	23,7720967	A1	B2	40	2025-05-24	CLOSED
52	37,9785149	23,7462318	A2	B1	40	2025-05-24	CLOSED
53	37,9970325	23,7351688	A3	B1	20	2025-05-24	CLOSED
54	37,9546785	23,7247726	A3	B2	20	2025-05-24	CLOSED
55	37,99191	23,7663477	A2	B2	40	2025-05-24	CLOSED
56	37,9976946	23,7350731	A2	B1	40	2025-05-24	CLOSED
57	37,9917038	23,7312294	A2	B1	40	2025-05-24	CLOSED
58	37,9498422	23,7506832	A2	B2	40	2025-05-24	CLOSED
59	37,9774362	23,6636983	A1	B2	20	2025-05-24	CLOSED
60	37,9939115	23,6690323	A1	B2	40	2025-05-24	CLOSED
61	37,977176	23,7099784	A2	B1	40	2025-05-24	CLOSED
62	37,9837138	23,7825302	A2	B2	40	2025-05-24	CLOSED
63	38,0065378	23,7375275	A2	B2	40	2025-05-24	CLOSED
64	38,0025799	23,722796	A3	B1	20	2025-05-25	CLOSED
65	38,0204165	23,7416259	A3	B2	20	2025-05-25	CLOSED
66	38,0125403	23,7362023	A1	B2	75	2025-05-25	CLOSED
67	37,9836954	23,7399425	A3	B1	20	2025-05-25	CLOSED
68	38,014754	23,6828747	A1	B2	75	2025-05-25	CLOSED
69	37,9491741	23,7712022	A3	B2	20	2025-05-25	CLOSED
70	37,9541015	23,7330198	A2	B2	75	2025-05-25	CLOSED
71	38,0239282	23,73086	A2	B2	75	2025-05-25	CLOSED
72	37,9574949	23,7514015	A3	B2	20	2025-05-25	CLOSED
73	37,9679271	23,7611341	A2	B2	75	2025-05-25	CLOSED
74	37,9981354	23,775022	A1	B2	75	2025-05-25	CLOSED
75	37,9390389	23,7556215	A2	B2	75	2025-05-25	CLOSED
76	37,9875477	23,7690288	A2	B2	40	2025-05-25	CLOSED
77	38,020017	23,7121164	A2	B2	40	2025-05-25	CLOSED
78	38,0057368	23,6675468	A3	B2	20	2025-05-25	CLOSED
79	37,956039	23,7510721	A2	B2	40	2025-05-25	CLOSED
80	37,9655982	23,7433831	A2	B1	40	2025-05-25	CLOSED
81	38,035337	23,7285907	A2	B2	40	2025-05-25	CLOSED
82	37,9464925	23,7017619	A2	B2	40	2025-05-25	CLOSED
83	38,0117907	23,715673	A3	B2	20	2025-05-25	CLOSED
84	37,9424264	23,7250789	A3	B2	20	2025-05-25	CLOSED
85	38,0129063	23,6938307	A2	B2	40	2025-05-25	CLOSED
86	37,9562468	23,7277496	A3	B2	20	2025-05-25	CLOSED
87	37,9417008	23,7275471	A2	B2	40	2025-05-25	CLOSED
88	37,994624	23,6640273	A2	B2	40	2025-05-25	CLOSED
89	38,0053228	23,6972949	A2	B2	40	2025-05-25	CLOSED
90	37,9891989	23,7306058	A3	B1	20	2025-05-25	CLOSED
91	38,0305623	23,7093367	A2	B2	40	2025-05-25	CLOSED
92	37,9558837	23,7253656	A1	B2	20	2025-05-25	CLOSED

2.3 ΠΙΝΑΚΑΣ ΣΤΗΝ ΑΡΧΗ ΤΗΣ 4^{ΗΣ} ΕΠΑΝΑΛΗΨΗΣ

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9634817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	CLOSED
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,0338068	23,7091981	A2	B2	75	2025-05-24	CLOSED
33	37,9400333	23,7544334	A2	B2	75	2025-05-24	CLOSED
34	37,9842731	23,6731588	A1	B2	20	2025-05-24	CLOSED
35	37,9995306	23,682598	A2	B2	75	2025-05-24	CLOSED
36	37,9969159	23,7803971	A2	B2	75	2025-05-24	CLOSED
37	37,9882392	23,6755466	A3	B2	20	2025-05-24	CLOSED
38	37,9743537	23,6916898	A2	B2	75	2025-05-24	CLOSED
39	37,9664718	23,7327596	A2	B1	40	2025-05-24	CLOSED
40	37,9505651	23,7275189	A2	B2	40	2025-05-24	CLOSED
41	38,0053307	23,7757093	A3	B2	20	2025-05-24	CLOSED
42	37,9619506	23,7015338	A3	B2	20	2025-05-24	CLOSED
43	37,9886403	23,7677957	A2	B2	40	2025-05-24	CLOSED
44	37,943421	23,7299333	A2	B2	40	2025-05-24	CLOSED
45	37,9518225	23,7613104	A3	B2	20	2025-05-24	CLOSED
46	38,002855	23,674653	A3	B2	20	2025-05-24	CLOSED

47	37,9437648	23,7027373	A2	B2	40	2025-05-24	CLOSED
48	37,96635	23,7671193	A3	B2	20	2025-05-24	CLOSED
49	37,9597588	23,6772071	A3	B2	20	2025-05-24	CLOSED
50	37,9845642	23,7124659	A3	B1	20	2025-05-24	CLOSED
51	37,9952524	23,7720967	A1	B2	40	2025-05-24	CLOSED
52	37,9785149	23,7462318	A2	B1	40	2025-05-24	CLOSED
53	37,9970325	23,7351688	A3	B1	20	2025-05-24	CLOSED
54	37,9546785	23,7247726	A3	B2	20	2025-05-24	CLOSED
55	37,99191	23,7663477	A2	B2	40	2025-05-24	CLOSED
56	37,9976946	23,7350731	A2	B1	40	2025-05-24	CLOSED
57	37,9917038	23,7312294	A2	B1	40	2025-05-24	CLOSED
58	37,9498422	23,7506832	A2	B2	40	2025-05-24	CLOSED
59	37,9774362	23,6636983	A1	B2	20	2025-05-24	CLOSED
60	37,9939115	23,6690323	A1	B2	40	2025-05-24	CLOSED
61	37,977176	23,7099784	A2	B1	40	2025-05-24	CLOSED
62	37,9837138	23,7825302	A2	B2	40	2025-05-24	CLOSED
63	38,0065378	23,7375275	A2	B2	40	2025-05-24	CLOSED
64	38,0025799	23,722796	A3	B1	20	2025-05-25	CLOSED
65	38,0204165	23,7416259	A3	B2	20	2025-05-25	CLOSED
66	38,0125403	23,7362023	A1	B2	75	2025-05-25	CLOSED
67	37,9836954	23,7399425	A3	B1	20	2025-05-25	CLOSED
68	38,014754	23,6828747	A1	B2	75	2025-05-25	CLOSED
69	37,9491741	23,7712022	A3	B2	20	2025-05-25	CLOSED
70	37,9541015	23,7330198	A2	B2	75	2025-05-25	CLOSED
71	38,0239282	23,73086	A2	B2	75	2025-05-25	CLOSED
72	37,9574949	23,7514015	A3	B2	20	2025-05-25	CLOSED
73	37,9679271	23,7611341	A2	B2	75	2025-05-25	CLOSED
74	37,9981354	23,775022	A1	B2	75	2025-05-25	CLOSED
75	37,9390389	23,7556215	A2	B2	75	2025-05-25	CLOSED
76	37,9875477	23,7690288	A2	B2	40	2025-05-25	CLOSED
77	38,020017	23,7121164	A2	B2	40	2025-05-25	CLOSED
78	38,0057368	23,6675468	A3	B2	20	2025-05-25	CLOSED
79	37,956039	23,7510721	A2	B2	40	2025-05-25	CLOSED
80	37,9655982	23,7433831	A2	B1	40	2025-05-25	CLOSED
81	38,035337	23,7285907	A2	B2	40	2025-05-25	CLOSED
82	37,9464925	23,7017619	A2	B2	40	2025-05-25	CLOSED
83	38,0117907	23,715673	A3	B2	20	2025-05-25	CLOSED
84	37,9424264	23,7250789	A3	B2	20	2025-05-25	CLOSED
85	38,0129063	23,6938307	A2	B2	40	2025-05-25	CLOSED
86	37,9562468	23,7277496	A3	B2	20	2025-05-25	CLOSED
87	37,9417008	23,7275471	A2	B2	40	2025-05-25	CLOSED
88	37,994624	23,6640273	A2	B2	40	2025-05-25	CLOSED
89	38,0053228	23,6972949	A2	B2	40	2025-05-25	CLOSED
90	37,9891989	23,7306058	A3	B1	20	2025-05-25	CLOSED
91	38,0305623	23,7093367	A2	B2	40	2025-05-25	CLOSED
92	37,9558837	23,7253656	A1	B2	20	2025-05-25	CLOSED
93	38,0234399	23,6961097	A3	B2	20	2025-05-26	OPEN

94	38,0361726	23,7257941	A1	B2	20	2025-05-26	OPEN
95	37,939023	23,721652	A2	B2	75	2025-05-26	OPEN
96	37,9537484	23,7397933	A2	B2	75	2025-05-26	OPEN
97	38,0193547	23,7770111	A2	B2	75	2025-05-26	OPEN
98	37,932473	23,7345134	A2	B2	75	2025-05-26	OPEN
99	37,9613433	23,7046758	A2	B2	75	2025-05-26	OPEN
100	38,0236393	23,7079981	A2	B2	75	2025-05-26	OPEN
101	38,0314804	23,7467804	A2	B2	40	2025-05-26	OPEN
102	38,0234299	23,7257183	A2	B2	40	2025-05-26	OPEN
103	37,9359108	23,7509022	A3	B2	20	2025-05-26	OPEN
104	38,0018121	23,7499768	A3	B2	20	2025-05-26	OPEN
105	37,9671167	23,7437072	A3	B1	20	2025-05-26	OPEN
106	37,9344239	23,7167525	A2	B2	40	2025-05-26	OPEN
107	37,9702844	23,6807088	A2	B2	40	2025-05-26	OPEN
108	38,0137826	23,718095	A2	B2	40	2025-05-26	OPEN
109	38,0190162	23,6964341	A1	B2	20	2025-05-26	OPEN
110	37,9969482	23,6819025	A2	B2	40	2025-05-26	OPEN
111	37,9700654	23,7403982	A1	B1	20	2025-05-26	OPEN
112	38,0161039	23,7251726	A2	B2	40	2025-05-26	OPEN
113	38,0281925	23,6948979	A3	B2	20	2025-05-26	OPEN
114	37,9701742	23,722012	A2	B1	40	2025-05-26	OPEN
115	37,9384481	23,7117177	A3	B2	20	2025-05-26	OPEN
116	38,0066144	23,7627436	A2	B2	40	2025-05-26	OPEN
117	38,0310486	23,6999457	A3	B2	20	2025-05-26	OPEN
118	37,9874526	23,7410488	A3	B1	20	2025-05-26	OPEN
119	38,0300582	23,705834	A1	B2	40	2025-05-26	OPEN
120	38,0065065	23,6878981	A2	B2	40	2025-05-26	OPEN
121	38,0216428	23,6899647	A2	B2	40	2025-05-26	OPEN
122	37,9836876	23,6746058	A2	B2	40	2025-05-26	OPEN
123	38,0021911	23,787535	A3	B2	20	2025-05-26	OPEN
124	37,9948363	23,7127156	A3	B1	20	2025-05-26	OPEN
125	38,0132396	23,7197681	A2	B2	40	2025-05-26	OPEN
126	37,9810042	23,7886499	A2	B2	40	2025-05-26	OPEN

2.4 ΠΙΝΑΚΑΣ ΣΤΟ ΤΕΛΟΣ ΤΗΣ 4^{ΗΣ} ΕΠΑΝΑΛΗΨΗΣ

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9634817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	CLOSED
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,0338068	23,7091981	A2	B2	75	2025-05-24	CLOSED
33	37,9400333	23,7544334	A2	B2	75	2025-05-24	CLOSED
34	37,9842731	23,6731588	A1	B2	20	2025-05-24	CLOSED
35	37,9995306	23,682598	A2	B2	75	2025-05-24	CLOSED
36	37,9969159	23,7803971	A2	B2	75	2025-05-24	CLOSED
37	37,9882392	23,6755466	A3	B2	20	2025-05-24	CLOSED
38	37,9743537	23,6916898	A2	B2	75	2025-05-24	CLOSED
39	37,9664718	23,7327596	A2	B1	40	2025-05-24	CLOSED
40	37,9505651	23,7275189	A2	B2	40	2025-05-24	CLOSED
41	38,0053307	23,7757093	A3	B2	20	2025-05-24	CLOSED
42	37,9619506	23,7015338	A3	B2	20	2025-05-24	CLOSED
43	37,9886403	23,7677957	A2	B2	40	2025-05-24	CLOSED
44	37,943421	23,7299333	A2	B2	40	2025-05-24	CLOSED
45	37,9518225	23,7613104	A3	B2	20	2025-05-24	CLOSED
46	38,002855	23,674653	A3	B2	20	2025-05-24	CLOSED

47	37,9437648	23,7027373	A2	B2	40	2025-05-24	CLOSED
48	37,96635	23,7671193	A3	B2	20	2025-05-24	CLOSED
49	37,9597588	23,6772071	A3	B2	20	2025-05-24	CLOSED
50	37,9845642	23,7124659	A3	B1	20	2025-05-24	CLOSED
51	37,9952524	23,7720967	A1	B2	40	2025-05-24	CLOSED
52	37,9785149	23,7462318	A2	B1	40	2025-05-24	CLOSED
53	37,9970325	23,7351688	A3	B1	20	2025-05-24	CLOSED
54	37,9546785	23,7247726	A3	B2	20	2025-05-24	CLOSED
55	37,99191	23,7663477	A2	B2	40	2025-05-24	CLOSED
56	37,9976946	23,7350731	A2	B1	40	2025-05-24	CLOSED
57	37,9917038	23,7312294	A2	B1	40	2025-05-24	CLOSED
58	37,9498422	23,7506832	A2	B2	40	2025-05-24	CLOSED
59	37,9774362	23,6636983	A1	B2	20	2025-05-24	CLOSED
60	37,9939115	23,6690323	A1	B2	40	2025-05-24	CLOSED
61	37,977176	23,7099784	A2	B1	40	2025-05-24	CLOSED
62	37,9837138	23,7825302	A2	B2	40	2025-05-24	CLOSED
63	38,0065378	23,7375275	A2	B2	40	2025-05-24	CLOSED
64	38,0025799	23,722796	A3	B1	20	2025-05-25	CLOSED
65	38,0204165	23,7416259	A3	B2	20	2025-05-25	CLOSED
66	38,0125403	23,7362023	A1	B2	75	2025-05-25	CLOSED
67	37,9836954	23,7399425	A3	B1	20	2025-05-25	CLOSED
68	38,014754	23,6828747	A1	B2	75	2025-05-25	CLOSED
69	37,9491741	23,7712022	A3	B2	20	2025-05-25	CLOSED
70	37,9541015	23,7330198	A2	B2	75	2025-05-25	CLOSED
71	38,0239282	23,73086	A2	B2	75	2025-05-25	CLOSED
72	37,9574949	23,7514015	A3	B2	20	2025-05-25	CLOSED
73	37,9679271	23,7611341	A2	B2	75	2025-05-25	CLOSED
74	37,9981354	23,775022	A1	B2	75	2025-05-25	CLOSED
75	37,9390389	23,7556215	A2	B2	75	2025-05-25	CLOSED
76	37,9875477	23,7690288	A2	B2	40	2025-05-25	CLOSED
77	38,020017	23,7121164	A2	B2	40	2025-05-25	CLOSED
78	38,0057368	23,6675468	A3	B2	20	2025-05-25	CLOSED
79	37,956039	23,7510721	A2	B2	40	2025-05-25	CLOSED
80	37,9655982	23,7433831	A2	B1	40	2025-05-25	CLOSED
81	38,035337	23,7285907	A2	B2	40	2025-05-25	CLOSED
82	37,9464925	23,7017619	A2	B2	40	2025-05-25	CLOSED
83	38,0117907	23,715673	A3	B2	20	2025-05-25	CLOSED
84	37,9424264	23,7250789	A3	B2	20	2025-05-25	CLOSED
85	38,0129063	23,6938307	A2	B2	40	2025-05-25	CLOSED
86	37,9562468	23,7277496	A3	B2	20	2025-05-25	CLOSED
87	37,9417008	23,7275471	A2	B2	40	2025-05-25	CLOSED
88	37,994624	23,6640273	A2	B2	40	2025-05-25	CLOSED
89	38,0053228	23,6972949	A2	B2	40	2025-05-25	CLOSED
90	37,9891989	23,7306058	A3	B1	20	2025-05-25	CLOSED
91	38,0305623	23,7093367	A2	B2	40	2025-05-25	CLOSED
92	37,9558837	23,7253656	A1	B2	20	2025-05-25	CLOSED
93	38,0234399	23,6961097	A3	B2	20	2025-05-26	CLOSED

94	38,0361726	23,7257941	A1	B2	20	2025-05-26	CLOSED
95	37,939023	23,721652	A2	B2	75	2025-05-26	CLOSED
96	37,9537484	23,7397933	A2	B2	75	2025-05-26	CLOSED
97	38,0193547	23,7770111	A2	B2	75	2025-05-26	CLOSED
98	37,932473	23,7345134	A2	B2	75	2025-05-26	CLOSED
99	37,9613433	23,7046758	A2	B2	75	2025-05-26	CLOSED
100	38,0236393	23,7079981	A2	B2	75	2025-05-26	CLOSED
101	38,0314804	23,7467804	A2	B2	40	2025-05-26	CLOSED
102	38,0234299	23,7257183	A2	B2	40	2025-05-26	CLOSED
103	37,9359108	23,7509022	A3	B2	20	2025-05-26	CLOSED
104	38,0018121	23,7499768	A3	B2	20	2025-05-26	OPEN
105	37,9671167	23,7437072	A3	B1	20	2025-05-26	CLOSED
106	37,9344239	23,7167525	A2	B2	40	2025-05-26	CLOSED
107	37,9702844	23,6807088	A2	B2	40	2025-05-26	CLOSED
108	38,0137826	23,718095	A2	B2	40	2025-05-26	CLOSED
109	38,0190162	23,6964341	A1	B2	20	2025-05-26	CLOSED
110	37,9969482	23,6819025	A2	B2	40	2025-05-26	CLOSED
111	37,9700654	23,7403982	A1	B1	20	2025-05-26	CLOSED
112	38,0161039	23,7251726	A2	B2	40	2025-05-26	CLOSED
113	38,0281925	23,6948979	A3	B2	20	2025-05-26	CLOSED
114	37,9701742	23,722012	A2	B1	40	2025-05-26	CLOSED
115	37,9384481	23,7117177	A3	B2	20	2025-05-26	CLOSED
116	38,0066144	23,7627436	A2	B2	40	2025-05-26	CLOSED
117	38,0310486	23,6999457	A3	B2	20	2025-05-26	CLOSED
118	37,9874526	23,7410488	A3	B1	20	2025-05-26	CLOSED
119	38,0300582	23,705834	A1	B2	40	2025-05-26	CLOSED
120	38,0065065	23,6878981	A2	B2	40	2025-05-26	CLOSED
121	38,0216428	23,6899647	A2	B2	40	2025-05-26	CLOSED
122	37,9836876	23,6746058	A2	B2	40	2025-05-26	CLOSED
123	38,0021911	23,787535	A3	B2	20	2025-05-26	OPEN
124	37,9948363	23,7127156	A3	B1	20	2025-05-26	OPEN
125	38,0132396	23,7197681	A2	B2	40	2025-05-26	CLOSED
126	37,9810042	23,7886499	A2	B2	40	2025-05-26	CLOSED

2.5 ΠΙΝΑΚΑΣ ΣΤΗΝ ΑΡΧΗ ΤΗΣ 5^{ΗΣ} ΕΠΑΝΑΛΗΨΗΣ

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9634817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	CLOSED
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,0338068	23,7091981	A2	B2	75	2025-05-24	CLOSED
33	37,9400333	23,7544334	A2	B2	75	2025-05-24	CLOSED
34	37,9842731	23,6731588	A1	B2	20	2025-05-24	CLOSED
35	37,9995306	23,682598	A2	B2	75	2025-05-24	CLOSED
36	37,9969159	23,7803971	A2	B2	75	2025-05-24	CLOSED
37	37,9882392	23,6755466	A3	B2	20	2025-05-24	CLOSED
38	37,9743537	23,6916898	A2	B2	75	2025-05-24	CLOSED
39	37,9664718	23,7327596	A2	B1	40	2025-05-24	CLOSED
40	37,9505651	23,7275189	A2	B2	40	2025-05-24	CLOSED
41	38,0053307	23,7757093	A3	B2	20	2025-05-24	CLOSED
42	37,9619506	23,7015338	A3	B2	20	2025-05-24	CLOSED
43	37,9886403	23,7677957	A2	B2	40	2025-05-24	CLOSED
44	37,943421	23,7299333	A2	B2	40	2025-05-24	CLOSED
45	37,9518225	23,7613104	A3	B2	20	2025-05-24	CLOSED
46	38,002855	23,674653	A3	B2	20	2025-05-24	CLOSED

47	37,9437648	23,7027373	A2	B2	40	2025-05-24	CLOSED
48	37,96635	23,7671193	A3	B2	20	2025-05-24	CLOSED
49	37,9597588	23,6772071	A3	B2	20	2025-05-24	CLOSED
50	37,9845642	23,7124659	A3	B1	20	2025-05-24	CLOSED
51	37,9952524	23,7720967	A1	B2	40	2025-05-24	CLOSED
52	37,9785149	23,7462318	A2	B1	40	2025-05-24	CLOSED
53	37,9970325	23,7351688	A3	B1	20	2025-05-24	CLOSED
54	37,9546785	23,7247726	A3	B2	20	2025-05-24	CLOSED
55	37,99191	23,7663477	A2	B2	40	2025-05-24	CLOSED
56	37,9976946	23,7350731	A2	B1	40	2025-05-24	CLOSED
57	37,9917038	23,7312294	A2	B1	40	2025-05-24	CLOSED
58	37,9498422	23,7506832	A2	B2	40	2025-05-24	CLOSED
59	37,9774362	23,6636983	A1	B2	20	2025-05-24	CLOSED
60	37,9939115	23,6690323	A1	B2	40	2025-05-24	CLOSED
61	37,977176	23,7099784	A2	B1	40	2025-05-24	CLOSED
62	37,9837138	23,7825302	A2	B2	40	2025-05-24	CLOSED
63	38,0065378	23,7375275	A2	B2	40	2025-05-24	CLOSED
64	38,0025799	23,722796	A3	B1	20	2025-05-25	CLOSED
65	38,0204165	23,7416259	A3	B2	20	2025-05-25	CLOSED
66	38,0125403	23,7362023	A1	B2	75	2025-05-25	CLOSED
67	37,9836954	23,7399425	A3	B1	20	2025-05-25	CLOSED
68	38,014754	23,6828747	A1	B2	75	2025-05-25	CLOSED
69	37,9491741	23,7712022	A3	B2	20	2025-05-25	CLOSED
70	37,9541015	23,7330198	A2	B2	75	2025-05-25	CLOSED
71	38,0239282	23,73086	A2	B2	75	2025-05-25	CLOSED
72	37,9574949	23,7514015	A3	B2	20	2025-05-25	CLOSED
73	37,9679271	23,7611341	A2	B2	75	2025-05-25	CLOSED
74	37,9981354	23,775022	A1	B2	75	2025-05-25	CLOSED
75	37,9390389	23,7556215	A2	B2	75	2025-05-25	CLOSED
76	37,9875477	23,7690288	A2	B2	40	2025-05-25	CLOSED
77	38,020017	23,7121164	A2	B2	40	2025-05-25	CLOSED
78	38,0057368	23,6675468	A3	B2	20	2025-05-25	CLOSED
79	37,956039	23,7510721	A2	B2	40	2025-05-25	CLOSED
80	37,9655982	23,7433831	A2	B1	40	2025-05-25	CLOSED
81	38,035337	23,7285907	A2	B2	40	2025-05-25	CLOSED
82	37,9464925	23,7017619	A2	B2	40	2025-05-25	CLOSED
83	38,0117907	23,715673	A3	B2	20	2025-05-25	CLOSED
84	37,9424264	23,7250789	A3	B2	20	2025-05-25	CLOSED
85	38,0129063	23,6938307	A2	B2	40	2025-05-25	CLOSED
86	37,9562468	23,7277496	A3	B2	20	2025-05-25	CLOSED
87	37,9417008	23,7275471	A2	B2	40	2025-05-25	CLOSED
88	37,994624	23,6640273	A2	B2	40	2025-05-25	CLOSED
89	38,0053228	23,6972949	A2	B2	40	2025-05-25	CLOSED
90	37,9891989	23,7306058	A3	B1	20	2025-05-25	CLOSED
91	38,0305623	23,7093367	A2	B2	40	2025-05-25	CLOSED
92	37,9558837	23,7253656	A1	B2	20	2025-05-25	CLOSED
93	38,0234399	23,6961097	A3	B2	20	2025-05-26	CLOSED

94	38,0361726	23,7257941	A1	B2	20	2025-05-26	CLOSED
95	37,939023	23,721652	A2	B2	75	2025-05-26	CLOSED
96	37,9537484	23,7397933	A2	B2	75	2025-05-26	CLOSED
97	38,0193547	23,7770111	A2	B2	75	2025-05-26	CLOSED
98	37,932473	23,7345134	A2	B2	75	2025-05-26	CLOSED
99	37,9613433	23,7046758	A2	B2	75	2025-05-26	CLOSED
100	38,0236393	23,7079981	A2	B2	75	2025-05-26	CLOSED
101	38,0314804	23,7467804	A2	B2	40	2025-05-26	CLOSED
102	38,0234299	23,7257183	A2	B2	40	2025-05-26	CLOSED
103	37,9359108	23,7509022	A3	B2	20	2025-05-26	CLOSED
104	38,0018121	23,7499768	A3	B2	20	2025-05-26	OPEN
105	37,9671167	23,7437072	A3	B1	20	2025-05-26	CLOSED
106	37,9344239	23,7167525	A2	B2	40	2025-05-26	CLOSED
107	37,9702844	23,6807088	A2	B2	40	2025-05-26	CLOSED
108	38,0137826	23,718095	A2	B2	40	2025-05-26	CLOSED
109	38,0190162	23,6964341	A1	B2	20	2025-05-26	CLOSED
110	37,9969482	23,6819025	A2	B2	40	2025-05-26	CLOSED
111	37,9700654	23,7403982	A1	B1	20	2025-05-26	CLOSED
112	38,0161039	23,7251726	A2	B2	40	2025-05-26	CLOSED
113	38,0281925	23,6948979	A3	B2	20	2025-05-26	CLOSED
114	37,9701742	23,722012	A2	B1	40	2025-05-26	CLOSED
115	37,9384481	23,7117177	A3	B2	20	2025-05-26	CLOSED
116	38,0066144	23,7627436	A2	B2	40	2025-05-26	CLOSED
117	38,0310486	23,6999457	A3	B2	20	2025-05-26	CLOSED
118	37,9874526	23,7410488	A3	B1	20	2025-05-26	CLOSED
119	38,0300582	23,705834	A1	B2	40	2025-05-26	CLOSED
120	38,0065065	23,6878981	A2	B2	40	2025-05-26	CLOSED
121	38,0216428	23,6899647	A2	B2	40	2025-05-26	CLOSED
122	37,9836876	23,6746058	A2	B2	40	2025-05-26	CLOSED
123	38,0021911	23,787535	A3	B2	20	2025-05-26	OPEN
124	37,9948363	23,7127156	A3	B1	20	2025-05-26	OPEN
125	38,0132396	23,7197681	A2	B2	40	2025-05-26	CLOSED
126	37,9810042	23,7886499	A2	B2	40	2025-05-26	CLOSED
127	38,027137	23,7154531	A2	B2	75	2025-05-27	OPEN
128	38,0181367	23,6810499	A2	B2	75	2025-05-27	OPEN
129	37,9875201	23,7556049	A2	B1	75	2025-05-27	OPEN
130	38,006527	23,7041834	A2	B2	75	2025-05-27	OPEN
131	37,9795074	23,7618303	A2	B2	75	2025-05-27	OPEN
132	38,0239588	23,7439313	A1	B2	75	2025-05-27	OPEN
133	37,9358228	23,7563657	A2	B2	75	2025-05-27	OPEN
134	37,9681531	23,7314359	A3	B1	20	2025-05-27	OPEN
135	37,9685572	23,7538421	A2	B2	40	2025-05-27	OPEN
136	37,9771792	23,7386739	A2	B1	40	2025-05-27	OPEN
137	37,9647444	23,7510135	A3	B2	20	2025-05-27	OPEN
138	38,0189824	23,6908164	A2	B2	40	2025-05-27	OPEN
139	37,9863998	23,7135799	A2	B1	40	2025-05-27	OPEN
140	37,9909246	23,7927212	A3	B2	20	2025-05-27	OPEN

141	37,9638153	23,7319762	A2	B1	40	2025-05-27	OPEN
142	38,0108939	23,7134697	A1	B2	75	2025-05-27	OPEN
143	37,9350918	23,7304947	A2	B2	40	2025-05-27	OPEN
144	38,0126832	23,6976868	A3	B2	20	2025-05-27	OPEN
145	37,9523862	23,7542582	A2	B2	40	2025-05-27	OPEN
146	38,0039768	23,7388017	A2	B1	40	2025-05-27	OPEN
147	38,02319	23,7691841	A2	B2	40	2025-05-27	OPEN
148	37,9905703	23,7268402	A3	B1	20	2025-05-27	OPEN
149	38,0233707	23,7403862	A3	B2	20	2025-05-27	OPEN
150	37,9572596	23,6815526	A3	B2	20	2025-05-27	OPEN
151	38,0018071	23,6790675	A2	B2	40	2025-05-27	OPEN
152	37,939754	23,7152032	A3	B2	20	2025-05-27	OPEN
153	37,9592776	23,7070066	A1	B2	40	2025-05-27	OPEN
154	38,0002818	23,732035	A3	B1	20	2025-05-27	OPEN
155	37,9635006	23,7002961	A1	B2	75	2025-05-27	OPEN
156	37,9666318	23,738528	A3	B1	20	2025-05-27	OPEN
157	37,941025	23,7240715	A2	B2	40	2025-05-27	OPEN
158	37,955236	23,7422422	A2	B2	40	2025-05-27	OPEN
159	37,9500809	23,6888438	A2	B2	40	2025-05-27	OPEN
160	38,0184152	23,7053803	A2	B2	40	2025-05-27	OPEN

2.6 ΠΙΝΑΚΑΣ ΣΤΟ ΤΕΛΟΣ ΤΗΣ 5^{ΗΣ} ΕΠΑΝΑΛΗΨΗΣ

A/A	Πλάτος	Μήκος	Ομάδα Εργασίας	Περιοχή	Χρόνος	Ημερομηνία	Κατάσταση
1	38,0247064	23,7497493	A2	B2	75	2025-05-23	CLOSED
2	37,97967	23,7222932	A2	B1	75	2025-05-23	CLOSED
3	37,9798292	23,7079059	A2	B1	75	2025-05-23	CLOSED
4	38,0114689	23,7437032	A2	B2	75	2025-05-23	CLOSED
5	37,9634817	23,7615485	A2	B2	75	2025-05-23	CLOSED
6	37,9721274	23,7432154	A3	B1	20	2025-05-23	CLOSED
7	37,9691651	23,7258157	A2	B1	40	2025-05-23	CLOSED
8	37,9912048	23,6992155	A1	B2	40	2025-05-23	CLOSED
9	38,0004608	23,6817504	A2	B2	40	2025-05-23	CLOSED
10	37,9624618	23,6973224	A3	B2	20	2025-05-23	CLOSED
11	38,031627	23,7220714	A2	B2	40	2025-05-23	CLOSED
12	38,0154784	23,7283312	A2	B2	40	2025-05-23	CLOSED
13	38,0154743	23,6984763	A3	B2	20	2025-05-23	CLOSED
14	37,941795	23,7188034	A2	B2	40	2025-05-23	CLOSED
15	37,9575984	23,696655	A2	B2	40	2025-05-23	CLOSED
16	38,0197628	23,7538455	A1	B2	75	2025-05-23	CLOSED
17	38,0330942	23,7134193	A1	B2	20	2025-05-23	CLOSED
18	37,9876589	23,6657213	A2	B2	40	2025-05-23	CLOSED
19	37,9653709	23,7881108	A3	B2	20	2025-05-23	CLOSED
20	37,9585606	23,754938	A3	B2	20	2025-05-23	CLOSED
21	37,95473	23,7262241	A3	B2	20	2025-05-23	CLOSED
22	37,9584463	23,6806828	A2	B2	40	2025-05-23	CLOSED
23	37,9798097	23,6618507	A2	B2	40	2025-05-23	CLOSED
24	37,9897201	23,6610492	A2	B2	40	2025-05-23	CLOSED
25	37,9587447	23,7075962	A2	B2	40	2025-05-23	CLOSED
26	38,0107657	23,7178194	A3	B2	20	2025-05-23	CLOSED
27	38,0316	23,7158291	A3	B2	20	2025-05-23	CLOSED
28	38,014468	23,6895591	A3	B2	20	2025-05-23	CLOSED
29	37,9511714	23,7022007	A3	B2	20	2025-05-23	CLOSED
30	38,0240869	23,725119	A2	B2	40	2025-05-23	CLOSED
31	37,9953676	23,7672342	A1	B2	75	2025-05-23	CLOSED
32	38,0338068	23,7091981	A2	B2	75	2025-05-24	CLOSED
33	37,9400333	23,7544334	A2	B2	75	2025-05-24	CLOSED
34	37,9842731	23,6731588	A1	B2	20	2025-05-24	CLOSED
35	37,9995306	23,682598	A2	B2	75	2025-05-24	CLOSED
36	37,9969159	23,7803971	A2	B2	75	2025-05-24	CLOSED
37	37,9882392	23,6755466	A3	B2	20	2025-05-24	CLOSED
38	37,9743537	23,6916898	A2	B2	75	2025-05-24	CLOSED
39	37,9664718	23,7327596	A2	B1	40	2025-05-24	CLOSED
40	37,9505651	23,7275189	A2	B2	40	2025-05-24	CLOSED
41	38,0053307	23,7757093	A3	B2	20	2025-05-24	CLOSED
42	37,9619506	23,7015338	A3	B2	20	2025-05-24	CLOSED
43	37,9886403	23,7677957	A2	B2	40	2025-05-24	CLOSED
44	37,943421	23,7299333	A2	B2	40	2025-05-24	CLOSED
45	37,9518225	23,7613104	A3	B2	20	2025-05-24	CLOSED
46	38,002855	23,674653	A3	B2	20	2025-05-24	CLOSED

47	37,9437648	23,7027373	A2	B2	40	2025-05-24	CLOSED
48	37,96635	23,7671193	A3	B2	20	2025-05-24	CLOSED
49	37,9597588	23,6772071	A3	B2	20	2025-05-24	CLOSED
50	37,9845642	23,7124659	A3	B1	20	2025-05-24	CLOSED
51	37,9952524	23,7720967	A1	B2	40	2025-05-24	CLOSED
52	37,9785149	23,7462318	A2	B1	40	2025-05-24	CLOSED
53	37,9970325	23,7351688	A3	B1	20	2025-05-24	CLOSED
54	37,9546785	23,7247726	A3	B2	20	2025-05-24	CLOSED
55	37,99191	23,7663477	A2	B2	40	2025-05-24	CLOSED
56	37,9976946	23,7350731	A2	B1	40	2025-05-24	CLOSED
57	37,9917038	23,7312294	A2	B1	40	2025-05-24	CLOSED
58	37,9498422	23,7506832	A2	B2	40	2025-05-24	CLOSED
59	37,9774362	23,6636983	A1	B2	20	2025-05-24	CLOSED
60	37,9939115	23,6690323	A1	B2	40	2025-05-24	CLOSED
61	37,977176	23,7099784	A2	B1	40	2025-05-24	CLOSED
62	37,9837138	23,7825302	A2	B2	40	2025-05-24	CLOSED
63	38,0065378	23,7375275	A2	B2	40	2025-05-24	CLOSED
64	38,0025799	23,722796	A3	B1	20	2025-05-25	CLOSED
65	38,0204165	23,7416259	A3	B2	20	2025-05-25	CLOSED
66	38,0125403	23,7362023	A1	B2	75	2025-05-25	CLOSED
67	37,9836954	23,7399425	A3	B1	20	2025-05-25	CLOSED
68	38,014754	23,6828747	A1	B2	75	2025-05-25	CLOSED
69	37,9491741	23,7712022	A3	B2	20	2025-05-25	CLOSED
70	37,9541015	23,7330198	A2	B2	75	2025-05-25	CLOSED
71	38,0239282	23,73086	A2	B2	75	2025-05-25	CLOSED
72	37,9574949	23,7514015	A3	B2	20	2025-05-25	CLOSED
73	37,9679271	23,7611341	A2	B2	75	2025-05-25	CLOSED
74	37,9981354	23,775022	A1	B2	75	2025-05-25	CLOSED
75	37,9390389	23,7556215	A2	B2	75	2025-05-25	CLOSED
76	37,9875477	23,7690288	A2	B2	40	2025-05-25	CLOSED
77	38,020017	23,7121164	A2	B2	40	2025-05-25	CLOSED
78	38,0057368	23,6675468	A3	B2	20	2025-05-25	CLOSED
79	37,956039	23,7510721	A2	B2	40	2025-05-25	CLOSED
80	37,9655982	23,7433831	A2	B1	40	2025-05-25	CLOSED
81	38,035337	23,7285907	A2	B2	40	2025-05-25	CLOSED
82	37,9464925	23,7017619	A2	B2	40	2025-05-25	CLOSED
83	38,0117907	23,715673	A3	B2	20	2025-05-25	CLOSED
84	37,9424264	23,7250789	A3	B2	20	2025-05-25	CLOSED
85	38,0129063	23,6938307	A2	B2	40	2025-05-25	CLOSED
86	37,9562468	23,7277496	A3	B2	20	2025-05-25	CLOSED
87	37,9417008	23,7275471	A2	B2	40	2025-05-25	CLOSED
88	37,994624	23,6640273	A2	B2	40	2025-05-25	CLOSED
89	38,0053228	23,6972949	A2	B2	40	2025-05-25	CLOSED
90	37,9891989	23,7306058	A3	B1	20	2025-05-25	CLOSED
91	38,0305623	23,7093367	A2	B2	40	2025-05-25	CLOSED
92	37,9558837	23,7253656	A1	B2	20	2025-05-25	CLOSED
93	38,0234399	23,6961097	A3	B2	20	2025-05-26	CLOSED

94	38,0361726	23,7257941	A1	B2	20	2025-05-26	CLOSED
95	37,939023	23,721652	A2	B2	75	2025-05-26	CLOSED
96	37,9537484	23,7397933	A2	B2	75	2025-05-26	CLOSED
97	38,0193547	23,7770111	A2	B2	75	2025-05-26	CLOSED
98	37,932473	23,7345134	A2	B2	75	2025-05-26	CLOSED
99	37,9613433	23,7046758	A2	B2	75	2025-05-26	CLOSED
100	38,0236393	23,7079981	A2	B2	75	2025-05-26	CLOSED
101	38,0314804	23,7467804	A2	B2	40	2025-05-26	CLOSED
102	38,0234299	23,7257183	A2	B2	40	2025-05-26	CLOSED
103	37,9359108	23,7509022	A3	B2	20	2025-05-26	CLOSED
104	38,0018121	23,7499768	A3	B2	20	2025-05-26	OPEN
105	37,9671167	23,7437072	A3	B1	20	2025-05-26	CLOSED
106	37,9344239	23,7167525	A2	B2	40	2025-05-26	CLOSED
107	37,9702844	23,6807088	A2	B2	40	2025-05-26	CLOSED
108	38,0137826	23,718095	A2	B2	40	2025-05-26	CLOSED
109	38,0190162	23,6964341	A1	B2	20	2025-05-26	CLOSED
110	37,9969482	23,6819025	A2	B2	40	2025-05-26	CLOSED
111	37,9700654	23,7403982	A1	B1	20	2025-05-26	CLOSED
112	38,0161039	23,7251726	A2	B2	40	2025-05-26	CLOSED
113	38,0281925	23,6948979	A3	B2	20	2025-05-26	CLOSED
114	37,9701742	23,722012	A2	B1	40	2025-05-26	CLOSED
115	37,9384481	23,7117177	A3	B2	20	2025-05-26	CLOSED
116	38,0066144	23,7627436	A2	B2	40	2025-05-26	CLOSED
117	38,0310486	23,6999457	A3	B2	20	2025-05-26	CLOSED
118	37,9874526	23,7410488	A3	B1	20	2025-05-26	CLOSED
119	38,0300582	23,705834	A1	B2	40	2025-05-26	CLOSED
120	38,0065065	23,6878981	A2	B2	40	2025-05-26	CLOSED
121	38,0216428	23,6899647	A2	B2	40	2025-05-26	CLOSED
122	37,9836876	23,6746058	A2	B2	40	2025-05-26	CLOSED
123	38,0021911	23,787535	A3	B2	20	2025-05-26	OPEN
124	37,9948363	23,7127156	A3	B1	20	2025-05-26	OPEN
125	38,0132396	23,7197681	A2	B2	40	2025-05-26	CLOSED
126	37,9810042	23,7886499	A2	B2	40	2025-05-26	CLOSED
127	38,027137	23,7154531	A2	B2	75	2025-05-27	CLOSED
128	38,0181367	23,6810499	A2	B2	75	2025-05-27	OPEN
129	37,9875201	23,7556049	A2	B1	75	2025-05-27	CLOSED
130	38,006527	23,7041834	A2	B2	75	2025-05-27	CLOSED
131	37,9795074	23,7618303	A2	B2	75	2025-05-27	CLOSED
132	38,0239588	23,7439313	A1	B2	75	2025-05-27	CLOSED
133	37,9358228	23,7563657	A2	B2	75	2025-05-27	CLOSED
134	37,9681531	23,7314359	A3	B1	20	2025-05-27	OPEN
135	37,9685572	23,7538421	A2	B2	40	2025-05-27	CLOSED
136	37,9771792	23,7386739	A2	B1	40	2025-05-27	CLOSED
137	37,9647444	23,7510135	A3	B2	20	2025-05-27	CLOSED
138	38,0189824	23,6908164	A2	B2	40	2025-05-27	CLOSED
139	37,9863998	23,7135799	A2	B1	40	2025-05-27	CLOSED
140	37,9909246	23,7927212	A3	B2	20	2025-05-27	OPEN