

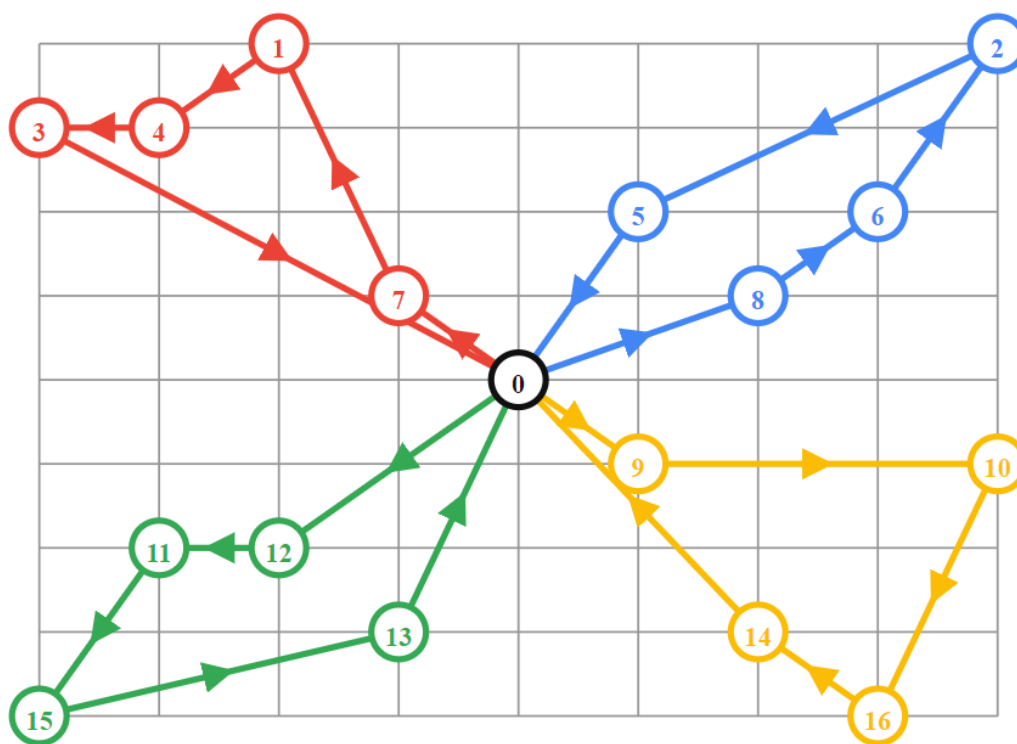


ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΚΑΙ ΤΟΠΟΓΡΑΦΩΝ
ΜΗΧΑΝΙΚΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΒΕΛΤΙΣΤΟΣ ΣΧΕΔΙΑΣΜΟΣ ΔΙΚΤΥΟΥ ΔΡΟΜΟΛΟΓΗΣΗΣ ΣΤΟΛΟΥ
ΟΧΗΜΑΤΩΝ

ΝΙΚΟΛΑΟΣ ΖΑΦΤΗΣ



ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΔΡ. ΝΙΚΟΛΑΟΣ ΔΟΥΛΑΜΗΣ

ΑΘΗΝΑ, 2021

ΕΥΧΑΡΙΣΤΙΕΣ

Ευχαριστώ θερμά τον επιβλέποντα καθηγητή, Δρ. Δουλάμη Νικόλαο για την πολύτιμη και ουσιαστική υποστήριξη που μου πρόσφερε σε όλα τα στάδια εκπόνησης της πτυχιακής μου εργασίας.

Τέλος, ευχαριστώ την οικογένεια μου για την ηθική και πνευματική υποστήριξη που μου προσφέρει καθημερινά.

*“No one who achieves success does so without acknowledging the help of others.
The wise and confident acknowledge this help with gratitude.”*

~Alfred North Whitehead

ΠΕΡΙΛΗΨΗ

Το παγκόσμιο εμπόριο, η τεράστια εκπομπή καυσαερίων από τα οχήματα και οι ανάγκες της κοινωνίας μετά την πανδημία του COVID-19, οδηγούν στην ανάπτυξη οργανωμένων και πιο αποτελεσματικών συστημάτων μεταφοράς των εμπορευμάτων έτσι ώστε να βελτιωθεί η αποτελεσματικότητα των παραδόσεων και να μειωθεί το κόστος των μεταφορών. Το αντικείμενο της παρούσας διπλωματικής άπτεται της ανάπτυξης προγραμμάτων βελτιστοποίησης, για προβλήματα δρομολόγησης στόλου οχημάτων (VRP). Συγκεκριμένα μελετήθηκε το πρόβλημα διανομής φαρμάκων στην περιφερειακή ενότητα της Αιτωλοακαρνανίας, με δεδομένα από την αποθήκη φαρμάκων του Αγρινίου, Pharmanet. Το οδικό δίκτυο αναπτύχθηκε σε GIS με γεωγραφικά δεδομένα από το OpenStreetMaps και η επίλυση έγινε μέσω του ελεύθερου λογισμικού OR-Tools.

ABSTRACT

Global trade, the massive emission of exhaust gases from vehicles and the needs of the society after the pandemic of COVID-19, are leading to the development of an organized and more efficient freight transport systems in order to improve the efficiency of deliveries and reduce transport costs. The subject of this thesis, relates to the development of optimization programs for vehicles routine problems (VRP). In particular, the problem of distribution of medicines in the regional unit of Aitoloakarnania, with data from the Agrinio's drug depot, Pharmanet, was studied. The road network was developed in GIS with geographic data from OpenStreetMaps and the resolution was done through the free OR-Tools software.

ΠΕΡΙΕΧΟΜΕΝΑ

Κεφάλαιο 1	10
1. Ορισμός θέματος μελέτης.....	10
2. Βασικές έννοιες.....	11
2.1 Μεταφορές, Εφοδιαστική Αλυσίδα & Logistics.....	11
2.2 Αποθηκευτικοί χώροι & Εγκαταστάσεις.....	12
2.3 Δρομολόγηση.....	12
2.4 Συστήματα Διαχείρισης Στόλου Οχημάτων	13
2.5 Συλλογή και Παρουσίαση Δεδομένων με ArcGIS.....	14
3. Η Σχετική Σημασία και Χρησιμότητα του Προβλήματος για το Κοινωνικό Σύνολο	14
4. Δομή εργασίας.....	15
5. Περιορισμοί	16
5.1 Τηλεματική διαχείριση στόλου.....	16
5.2 Google Distance Matrix API	16
Κεφάλαιο 2	18
2. Θεωρητική Ανάλυση.....	18
2.1 Ιστορική Αναδρομή Παραλλαγών VRP.	18
2.1.1 Πρόβλημα Δρομολόγησης Οχημάτων με περιορισμό Χωρητικότητας (1959)	19
2.1.2 Χρονικά εξαρτώμενο Πρόβλημα Δρομολόγησης (1966).....	19
2.1.3 Πρόβλημα Παραλαβής και Παράδοσης	20
2.1.4 Πρόβλημα Δρομολόγησης με Πολλαπλές Αφετηρίες (1969).....	20
2.1.5 Στοχαστικό Πρόβλημα Δρομολόγησης (1969).....	21
2.1.6 Πρόβλημα Δρομολόγησης Τοποθεσίας (1973).....	21
2.1.7 Περιοδικό Πρόβλημα Δρομολόγησης (1974).....	21
2.1.8 Δυναμικό Πρόβλημα Δρομολόγησης (1976).....	22
2.1.9 Πρόβλημα Δρομολόγησης Αποθεμάτων (1984).....	22
2.1.10 Μέγεθος Στόλου και Πρόβλημα Συνδυασμού Δρομολόγησης Οχημάτων(1984).....	22
2.1.11 Γενικευμένο Πρόβλημα Δρομολόγησης (1984)	23
2.1.12 Πρόβλημα Δρομολόγησης Πολλαπλών Αντικειμένων(1984)	23
2.1.13 Ασαφές Πρόβλημα Δρομολόγησης (1995).....	24
2.1.14 Ανοιχτό Πρόβλημα Δρομολόγησης (2000)	24
2.1.15 Πρόβλημα Δρομολόγησης με Περιορισμό Φόρτωσης (2003).....	24
2.1.16 Πρόβλημα Δρομολόγησης Multi-echelon (2009)	25
2.2 Πρόβλημα Δρομολόγησης με Χρονικά Παράθυρα (1977).....	25
2.3 Αλγόριθμοι & Βασικές Μέθοδοι Επίλυσης	27

2.4 Υφιστάμενη Κατάσταση.....	29
2.4.1 Εθνική Στρατηγική για την Εφοδιαστική στην Ελλάδα	29
2.4.3 Οργανισμοί Έρευνας στη Δρομολόγηση και την Εφοδιαστική	30
2.4.4 Υπολογιστικά Εργαλεία & Νέες Τεχνολογίες στη Δρομολόγηση Οχημάτων	31
2.5 Νομικό Πλαίσιο	31
Κεφάλαιο 3	33
3.1 Στόχοι έρευνας.....	33
3.2 Είδος έρευνας	34
3.3 Google’s OR-Tools	35
Κεφάλαιο 4	36
4.1 Το Πρόβλημα του Πλανόδιου Πωλητή	36
4.2 Παρουσίαση του αλγόριθμου	37
4.3 Δημιουργία δεδομένων.....	39
4.4 Δημιουργία Μοντέλου Δρομολόγησης	41
4.5 Δημιουργία Συνάρτησης Επιστροφής	42
4.6 Ορισμός Κόστους Διαδρομής.....	42
Κεφάλαιο 5	46
5.1 Πρόβλημα Δρομολόγησης Στόλου Οχημάτων	46
5.2 Δημιουργία δεδομένων.....	49
5.3 Δημιουργία Μοντέλου Δρομολόγησης	50
5.4 Δημιουργία Συνάρτησης Επιστροφής	50
5.5 Προσθήκη Διάστασης Απόστασης.....	51
5.6 Ανάλυση Αντικειμένου Διάστασης (Dimensions)	52
5.7 Συνάρτηση Εκτύπωσης	54
5.8 Κύρια Συνάρτηση.....	54
5.9 Παρουσίαση Λύσης σε Χάρτη.....	55
Κεφάλαιο 6	57
6.1 Πρόβλημα Δρομολόγησης Στόλου Οχημάτων Με Περιορισμό Χωρητικότητας	57
6.2 Παρουσίαση του αλγόριθμου	58
6.3 Δημιουργία δεδομένων.....	60
6.4 Δημιουργία Μοντέλου Δρομολόγησης	61
6.5 Δημιουργία Συνάρτησης Επιστροφής Ζήτησης.....	61
6.6 Ποινές & Απόσυρση Δρομολόγιου	62
6.7 Συνάρτηση Εκτύπωσης	63
6.8 Κύρια Συνάρτηση.....	64
6.9 Παρουσίαση Λύσης σε Χάρτη.....	65

Κεφάλαιο 7	67
7.1 Πρόβλημα Δρομολόγησης Στόλου Οχημάτων Με Χρονικά Παράθυρα	67
7.2 Παρουσίαση του αλγόριθμου	69
7.3 Δημιουργία δεδομένων.....	70
7.4 Δημιουργία Συνάρτησης Επιστροφής Χρόνου	71
7.5 Συνάρτηση Εκτύπωσης	73
7.6 Παράθυρο Λύσης.....	74
7.7 Κύρια Συνάρτηση.....	74
7.8 Παρουσίαση Λύσης σε Χάρτη.....	75
7.9 Δημιουργία δεδομένων.....	76
7.10 Προσθήκη Χρόνων Φόρτωσης – Εκφόρτωσης.....	77
7.11 Προσθήκη Περιορισμού Πόρων στην Αποθήκη.....	77
Κεφάλαιο 8	79
8. Συμπεράσματα & Προτάσεις.....	79
8.1 Συμπεράσματα	79
8.2 Προτάσεις.....	82
8.2.1 Εισηγήσεις για Περαιτέρω Έρευνα.....	82
8.2.2 Πρόβλημα Πράσινης Δρομολόγησης.....	83
8.2.2.1 Green-VRP	83
8.2.2.2 Pollution Routing Problem (PRP).....	84
8.2.2.3 VRP in Reverse Logistics (VRPRL).....	85
8.2.3 Grid Computing & QoS	86
Κεφάλαιο 9	89
9. Βιβλιογραφία	89
9.1 Ελληνική.....	89
9.2 Ξενόγλωσση.....	90
9.3 Δικτυογραφία.....	95

1.Ορισμός θέματος μελέτης

Οι μεταφορές και η βιώσιμη κινητικότητα είναι απαραίτητες συνιστώσες της οικονομίας και διαδραματίζουν σημαντικό ρόλο στις σχέσεις μεταξύ χωρών και τοποθεσιών, δημιουργώντας αλληλεπιδράσεις μεταξύ των ανθρώπων. Ως μεταφορά ορίζεται η διακομιδή αγαθών από μία τοποθεσία σε μία άλλη με την χρήση ενός μέσου-οχήματος. Οι περισσότερες κοινωνικές και οικονομικές δραστηριότητες, περιλαμβάνουν κάποιο είδος μεταφοράς. Η χρησιμότητα των μεταφορών στην σύγχρονη κοινωνία και η ευρεία χρήση της, έχει δημιουργήσει παρενέργειες. Η αυξημένη κινητικότητα προκαλεί περιβαλλοντικά και κυκλοφοριακά προβλήματα, οι παροχές σε αρκετές περιπτώσεις υπερβαίνουν το κόστος με αποτέλεσμα το μεταφορικό σύστημα να μην λειτουργεί σε βέλτιστο επίπεδο. Οι σύγχρονες απαιτήσεις για ελαχιστοποίηση του κόστους μεταφοράς, του χρόνου διαδρομής, την μεταφορά προϊόντων πόρτα-πόρτα και ο περιβαλλοντικός προβληματισμός, οδηγούν στην ανάγκη ανάπτυξης ενός οργανωμένου και πιο αποτελεσματικού συστήματος μεταφοράς. Το σύστημα μεταφοράς καταλαμβάνει το μεγαλύτερο μέρος του ποσού των δαπανών της εφοδιαστικής αλυσίδας μιας επιχείρησης και να είναι το κλειδί για την επίτευξη στόχων, μια υπηρεσία που προσθέτει αξία στην επιχειρηματική κοινότητα και την πολιτεία, και βασικός πυλώνας του εγχώριου και παγκόσμιου εμπορίου (Αποστολόπουλος Β., 2018).

Για το λόγο αυτό στην παρούσα εργασία γίνεται ανάπτυξη και επίλυση των βασικών προβλημάτων δρομολόγησης οχημάτων σε πραγματικά δίκτυα διανομής επιχειρήσεων, με στόχο την ελαχιστοποίηση του κόστους διαδρομής. Συγκεκριμένα, μελετούνται τα προβλήματα του πλανόδιου πωλητή και δρομολόγησης στόλου οχημάτων με περιορισμούς χωρητικότητας ή/και χρονικών παραθύρων. Το οδικό δίκτυο αναπτύχθηκε σε GIS με γεωγραφικά δεδομένα από OpenStreetMap και η επίλυση έγινε μέσω του ελεύθερου λογισμικού OR-Tools σε γλώσσα προγραμματισμού python. Η διερεύνηση έδειξε ότι η δρομολόγηση των οχημάτων εξαρτάται από τους περιορισμούς που θέτουμε την γεωγραφική θέση των εξυπηρετούμενων περιοχών, το νομικό πλαίσιο καθώς και τα χαρακτηριστικά του οδικού δικτύου.

Το πρόβλημα δρομολόγησης οχημάτων (VRP - Vehicle Routing Problem), λόγω της αποδοτικότητας του στην προσέγγιση πραγματικών καταστάσεων, θεωρείται ένα σημαντικό κεφάλαιο έρευνας. Η πρόοδος της τεχνολογίας, των μεθόδων και τεχνικών διαχείρισης του προβλήματος δρομολόγησης οχημάτων βελτιώνει τους χειρισμούς των φορτίων, την παράδοση των εμπορευμάτων, την ταχύτητα και την ποιότητα των υπηρεσιών, μειώνοντας το κόστος μετακίνησης και συμβάλλοντας στην εξοικονόμηση ενέργειας.

2.Βασικές έννοιες

2.1 Μεταφορές, Εφοδιαστική Αλυσίδα & Logistics

Εφοδιαστική αλυσίδα είναι η ευρύτερη έννοια των logistics. Είναι μια πολυδιάστατη δραστηριότητα που αναφέρεται στον σχεδιασμό και στην διαχείριση όλων των ενεργειών που σχετίζονται με την προμήθεια, μεταποίηση και τη διανομή ενός προϊόντος. Βασικός της στόχος η διαχείριση των σχέσεων όλων των κρίκων της αλυσίδας, ώστε να επιτυγχάνεται ικανοποιητικό κέρδος. Η τυπική δομή ενός συστήματος εφοδιαστική αλυσίδας παρουσιάζεται στην παρακάτω εικόνα.



Logistics είναι εκείνο το τμήμα της Διαχείρισης Εφοδιαστικής Αλυσίδας που περιλαμβάνει τη σχεδιασμένη διακίνηση και διαχείριση των προϊόντων από την παραγωγή έως την κατανάλωση, με το μικρότερο δυνατό κόστος (ελαχιστοποίηση κόστους). Οι υπηρεσίες των logistics αναφέρονται στην μεταφορά, παραλαβή και διανομή εμπορευμάτων, αποθήκευση, φορτο-εκφόρτωση, διαχείριση αποθεμάτων.

Κατά μήκος της αλυσίδας εφοδιασμού απαιτείται καλή διαχείριση logistics και στρατηγική των υπηρεσιών για να υπάρξει σωστή μεταφορά και διανομή των προϊόντων. Η δρομολόγηση οχημάτων είναι ένα τμήμα των logistic, που σχεδιάζει υλοποιεί και ελέγχει την κανονική και την αντίστροφη ροή των προϊόντων, των υπηρεσιών και των πληροφοριών από το σημείο προέλευσης μέχρι το σημείο προορισμού.

2.2 Αποθηκευτικοί χώροι & Εγκαταστάσεις

Η ταχεία ανάπτυξη της τεχνολογίας σε όλο τον κόσμο έχει μετατρέψει τις παλιές αποθήκες, σε κέντρα διανομής με γρήγορες λειτουργίες. Η βασική μονάδα οργάνωσης και υλοποίησης των νέων στρατηγικών διανομής, είναι τα κέντρα αποθήκευσης. Ένα κέντρο αποθήκευσης παίζει πολύ σημαντικό ρόλο, μέσω της κατάλληλης υποδομής, τεχνογνωσίας και οργάνωσης του. Δεν αποτελεί πλέον ένα τμήμα της επιχείρησης όπου απλά φυλάσσονται τα προϊόντα μέχρι τη στιγμή της παράδοσής τους. Είναι κέντρα διανομής και εξυπηρέτησης των πελατών. Ο αριθμός και η χωροθέτηση τους αποτελεί στρατηγικό θέμα που εξαρτάται από το συνολικό κόστος και το επιθυμητό επίπεδο εξυπηρέτησης των πελατών.

Πολλές επιχειρήσεις, που δεν έχουν ανάγκη να διατηρήσουν μια δικιά τους αποθήκη ή να ασχοληθούν με το πρόβλημα της διανομής αναθέτουν την υπηρεσία αυτή σε τρίτους. Η εξασφάλιση ολοκληρωμένων υπηρεσιών logistic μέσω εξωτερικού συνεργάτη της επιχείρησης, ονομάζεται Third Party Logistics (3PL).

2.3 Δρομολόγηση

Δρομολόγηση ονομάζεται η σχεδίαση της βέλτιστης διαδρομής σε ένα σύνολο τοποθεσιών (κόμβων). Η βέλτιστη διαδρομή συνήθως αναφέρεται στην ελαχιστοποίηση του συνολικού κόστους του δρομολογίου. Το κόστος εξαρτάται από το πρόβλημα που τίθεται κάθε φορά και επηρεάζεται από πολλούς παράγοντες. Συνεπώς η δρομολόγηση συνιστά ένα πρόβλημα βελτιστοποίησης. Πράγματι, οι αλγόριθμοι και οι τεχνικές που χρησιμοποιούνται για την εύρεση της βέλτιστης διαδρομής, απαντώνται σε προβλήματα συνδυαστικής βελτιστοποίησης. Το απλούστερο πρόβλημα δρομολόγησης είναι το γνωστό πρόβλημα συντομότερου μονοπατιού (Shortest Path Problem) ή του πλανόδιου πωλητή (Travel Salesman Problem, TSP), σημείο αναφοράς για σχεδόν όλα τα προβλήματα δρομολόγησης.

Τα προβλήματα δρομολόγησης στόλου οχημάτων VRP (Vehicle Routing Problem) αναφέρονται σε μία κατηγορία προβλημάτων προσδιορισμού διαδρομών κάθε οχήματος του στόλου, που ξεκινούν και καταλήγουν από μία κοινή αφετηρία (Depot). Σκοπός είναι η εξυπηρέτηση των σημείων ζήτησης, ελαχιστοποιώντας το συνολικό κόστος της διαδικασίας διανομής/συλλογής.

Υπάρχουν πολλές παραλλαγές του προβλήματος της δρομολόγησης οχημάτων που σχετίζεται με τους περιορισμούς και την προσπάθεια προσέγγισης των πραγματικών συνθηκών. Παλιότερα η δρομολόγηση ήταν χρήσιμη μόνο σε επίπεδο στρατηγικού σχεδιασμού γιατί παρείχαν γενικές πληροφορίες και όχι στην πρακτική δραστηριοποίηση στον φορέα της μεταφοράς. Στην σημερινή εποχή έχουν αναπτυχθεί συστήματα διαχείρισης στόλου οχημάτων, που συνδέονται με συστήματα GIS (Geographical Information Systems) για τη δημιουργία αναλυτικών διαδρομών, υψηλής ακρίβειας και σε ελάχιστο χρόνο.

2.4 Συστήματα Διαχείρισης Στόλου Οχημάτων

Τα συστήματα διαχείρισης στόλου, λαμβάνουν αποφάσεις σχετικά με τις διαδρομές που πρέπει να ακολουθήσουν τα οχήματα ενός στόλου. Πρόκειται για ψηφιακά συστήματα, που γνωστοποιούν με τη μέγιστη δυνατή ακρίβεια τις θέσεις των οχημάτων πάνω στο δίκτυο. Αποτελούν ισχυρό εργαλείο καταγραφής και ελέγχου των οχημάτων του στόλου, κατά την διάρκεια πραγματοποίησης του δρομολογίου. Ένα ακόμη μεγάλο όφελος είναι ότι συλλέγονται πολλά δεδομένα όπως στοιχεία αποστάσεων, χρόνων, αριθμού πελατών, μετρήσεις παραγωγικότητας και αποδοτικότητας, συγκρίσεις πραγματικών σε σχέση με τα προϋπολογισμένα και γενικότερα λεπτομερείς αναφορές των δρομολογίων που πριν ήταν αδύνατο. Η συλλογή αυτών των πληροφοριών, βοηθά στην αξιολόγηση και καταδεικνύει τρόπους βελτίωσης και αποτελεσματικότερης αξιοποίησης της υπάρχουσας υποδομής, γεγονός πολύ σημαντικό αν αναλογιστεί κανείς το επίπεδο ανταγωνισμού στις σύγχρονες επιχειρήσεις.

2.5 Σύλλογή και Παρουσίαση Δεδομένων με ArcGIS

Για την υλοποίηση των προβλημάτων και την παρουσίαση των αποτελεσμάτων, ήταν απαραίτητος ο σχεδιασμός ενός οδικού δικτύου και τα αντίστοιχα γεωγραφικά δεδομένα. Η εισαγωγή και επεξεργασία των δεδομένων πραγματοποιήθηκε με την βοήθεια του γεωγραφικού πληροφοριακού συστήματος ArcGIS. Στο επεξεργαστικό περιβάλλον των GIS τα δεδομένα αντιπροσωπεύονται από τα αντίστοιχα γεωμετρικά στοιχεία και η απεικόνισή τους γίνεται με σημεία, γραμμές και πολύγωνα. Το ArcMap είναι το βασικό περιβάλλον εργασιών του ArcGIS στο οποίο γίνεται η προβολή και διαχείριση των δεδομένων με σκοπό παραγωγή χαρτών.

3. Η Σχετική Σημασία και Χρησιμότητα του Προβλήματος για το Κοινωνικό Σύνολο

Οι τρέχουσες στρατηγικές εφοδιαστικής παραγωγής και διανομής δεν είναι βιώσιμες μακροπρόθεσμα. Το παραδοσιακό πρόβλημα δρομολόγησης οχημάτων (VRP), επικεντρώνεται στις οικονομικές επιπτώσεις των διαδρομών των οχημάτων, στον οργανισμό που εκτελεί την υπηρεσία διανομής. Πλέον υπάρχει ανάγκη να λαμβάνονται υπόψη οι περιβαλλοντικές, οικολογικές και κοινωνικές επιπτώσεις κατά τον σχεδιασμό των πολιτικών εφοδιαστικής, εκτός από το συμβατικό οικονομικό κόστος.

Υπάρχει μεγάλη ποικιλία προβλημάτων όσον αφορά τις πράσινες μεταφορές, όπως η προώθηση εναλλακτικών καυσίμων, τα ηλεκτρονικά οχήματα επόμενης γενιάς, τα οικολογικά ευφυή συστήματα μεταφορών και άλλες φιλικές προς το περιβάλλον υποδομές. Μια καλύτερη χρήση των οχημάτων και μια οικονομικά αποδοτική λύση δρομολόγησης οχημάτων θα μπορούσε να επιτύχει πιο άμεσα βιώσιμα συστήματα μεταφορών. Ο παραδοσιακός στόχος του VRP για τη μείωση της συνολικής απόστασης θα συμβάλει από μόνος του στη μείωση της κατανάλωσης καυσίμων και των εκπομπών ρύπων του περιβάλλοντος.

Η εξέταση ευρύτερων στόχων και περισσότερο λειτουργικών περιορισμών που αφορούν ζητήματα βιώσιμης εφοδιαστικής θέτουν νέα μοντέλα δρομολόγησης οχημάτων και νέα σενάρια εφαρμογής, τα οποία φυσικά οδηγούν σε πιο σύνθετα προβλήματα συνδυαστικής βελτιστοποίησης.

Η συνεισφορά αυτής της διπλωματικής είναι σε πρώτη φάση να παρουσιάσει την εξέλιξη της βιβλιογραφίας των προβλημάτων δρομολόγησης VRP, που αποτελούν

προβλήματα συνδυαστικής βελτιστοποίησης. Στην συνέχεια παρέχει μια επισκόπηση της σύνθεσης και επίλυσης προβλημάτων δρομολόγησης και πως σε αυτά μπορούν να προστεθούν περισσότεροι περιορισμοί που θα προσεγγίζουν τα πραγματικά προβλήματα της κοινωνίας και της οικολογίας.

4. Δομή εργασίας

Η εργασία αποτελείται από έξι κεφάλαια συμπεριλαμβανόμενης και της εισαγωγής. Το κεφάλαιο 1 αποτελεί το εισαγωγικό κεφάλαιο, όπου εξηγήθηκε ο σκοπός, οι βασικές έννοιες, η χρησιμότητα του προβλήματος για το κοινωνικό σύνολο καθώς και οι περιορισμοί του.

Στο κεφάλαιο 2 γίνεται ανασκόπηση της υφιστάμενης βιβλιογραφίας σχετικά με το πρόβλημα της δρομολόγησης οχημάτων, η παρουσίαση των αλγορίθμων που χρησιμοποιούνται για την επίλυση των προβλημάτων δρομολόγησης καθώς και το νομικό πλαίσιο που περιβάλλει το πρόβλημα της διανομής.

Στο κεφάλαιο 3 πραγματοποιείται μια εισαγωγή στο τρόπο μεθοδολογίας και έρευνας που χρησιμοποιήθηκε στην παρούσα διπλωματική. Στη συνέχεια αναλύεται το Πρόβλημα της Δρομολόγησης Οχημάτων,

1. στην περίπτωση του προβλήματος του πλανόδιου πωλητή,
2. στην περίπτωση της απλής δρομολόγησης στόλου οχημάτων,
3. στην περίπτωση δρομολόγησης στόλου οχημάτων με περιορισμό χωρητικότητας,
4. στην περίπτωση δρομολόγησης στόλου οχημάτων με χρονικά παράθυρα,

ως τα προβλήματα που πραγματεύεται η παρούσα εργασία. Ακολούθως, γίνεται μια σύντομη εισαγωγή στον διακομιστή επίλυσης OR-Tools των προβλημάτων και την επεξήγηση των ευρετικών αλγορίθμων που χρησιμοποιεί, προγραμματισμό περιορισμού (constraint programming) και συνδυαστικής βελτιστοποίησης (combinatorial optimization).

Στο κεφάλαιο 4 παρουσιάζονται οι τέσσερις περιπτώσεις του προβλήματος της δρομολόγησης, γίνεται επίλυση αυτών με την χρήση του ελεύθερου λογισμικού OR-Tools και γίνεται λεπτομερής περιγραφή της υλοποίησης του. Σε κάθε ενότητα

παρουσιάζονται τα αρχικά δεδομένα καθώς και τα αποτελέσματα μετά την επίλυση των προβλημάτων.

Στο κεφάλαιο 5 παρουσιάζονται τα συμπεράσματα που προέκυψαν από την λύση των προβλημάτων και παρουσιάζονται προτάσεις για περαιτέρω έρευνα του προβλήματος της δρομολόγησης. Στο κεφάλαιο 6 δίνεται η λίστα με τις βιβλιογραφικές αναφορές

5. Περιορισμοί

Κατά την υλοποίηση της διπλωματικής και συγκεκριμένα κατά την εφαρμογή και επίλυση των προβλημάτων δρομολόγησης παρουσιάστηκαν κάποιοι περιορισμοί όπως αναφέρονται στην συνέχεια.

5.1 Τηλεματική διαχείριση στόλου.

Μία από τις επιδιώξεις της διπλωματικής θα ήταν η επίλυση του προβλήματος της διαχείρισης στόλου οχημάτων με χρήση τηλεματικής. Το σύστημα δρομολόγησης συνδυάζεται με ένα σύστημα ασύρματης εποπτείας στόλου. Αν τοποθετηθεί ένας δέκτης GPS (Global Positioning System) σε κάθε όχημα, μπορούμε να έχουμε απεικόνιση της ακριβούς θέσης του. Η σημαντικότητα ανάπτυξης συστημάτων τηλεματικής είναι δεδομένη καθώς παρέχει σε πραγματικό χρόνο δεδομένα, παρέχει πλήρες ιστορικό διαδρομών, μέτρηση κατανάλωσης καυσίμου και αναδρομολόγηση του στόλου σε έκτακτες περιπτώσεις. Για την υλοποίηση ενός τέτοιου συστήματος είναι απαραίτητος ο εξοπλισμός του πομπού και του δέκτη GPS, για το λόγο αυτό ήταν αδύνατη η επίλυση του προβλήματος της τηλεματικής διαχείρισης στόλου.

5.2 Google Distance Matrix API

Στην παρούσα διπλωματική για την επίλυση των προβλημάτων δρομολόγησης στόλου οχημάτων VRP, χρησιμοποιείται ο διακομιστής επίλυσης προβλημάτων συνδυαστικής βελτιστοποίησης OR-Tools. Εκτός του διακομιστή επίλυσης που αποτελεί ελεύθερο λογισμικό με ανοικτό κώδικα, η Google, παρέχει και μία συμπληρωματική εφαρμογή την, Google Distance Matrix API. Η εφαρμογή αυτή παρέχει πληροφορίες για οπουδήποτε διαδρομή της ζητηθεί στον παγκόσμιο χάρτη. Οι πληροφορίες που παρέχει αφορούν διαφορετικούς τρόπους μετακίνησης (πχ. ποδήλατο, περπάτημα, αυτοκίνητο), απόσταση και χρόνους ταξιδιών σε πραγματικό χρόνο καθώς και πληροφορίες για την κυκλοφοριακή συμφόρηση. Η χρήση της συγκεκριμένης εφαρμογής συμφέρει στην διαχείριση μεγάλου πλήθους οχημάτων και

καθώς είναι επί πληρωμή δεν συμπεριλήφθηκε κατά την επίλυση των προβλημάτων δρομολόγησης.

2. Θεωρητική Ανάλυση **Εισαγωγή**

Η ανάγκη για αξιόπιστη διανομή προϊόντων και υπηρεσιών σε μικρό χρονικό διάστημα και με όσο το δυνατόν χαμηλότερο κόστος, συντελεί το πρόβλημα της δρομολόγησης που καλείται να λύσει μια σύγχρονη επιχείρηση. Οι συνεχείς εφαρμογές σε πραγματικές συνθήκες αναδεικνύουν ότι η χρήση υπολογιστικών μεθόδων τόσο σε επίπεδο σχεδιασμού όσο και σε επιχειρησιακό επίπεδο, αποφέρει αποτελεσματικότερη αντιμετώπιση του προβλήματος. Η εξέλιξη των τεχνικών βελτιστοποίησης και η ανάπτυξη μαθηματικών μοντέλων, συντελούν στην ευρεία επιτυχία και χρήση τους. Η εξέλιξη των τεχνικών βελτιστοποίησης και η ανάπτυξη μαθηματικών μοντέλων κατάφερε είναι δυνατή η ενσωμάτωση όλων των παραμέτρων και περιορισμών ενός προβλήματος δρομολόγησης οχημάτων, που προσεγγίζουν το πρόβλημα σε πραγματικές συνθήκες.

2.1 Ιστορική Αναδρομή Παραλλαγών VRP.

Προκειμένου να αποδείξουμε την εξέλιξη του προβλήματος δρομολόγησης οχημάτων, θα παρουσιαστούν οι παραλλαγές που έχουν μελετηθεί στην βιβλιογραφία. Αν και διακριτές, οι παραλλαγές συνδέονται στενά. Στην συνέχεια εισάγεται συνοπτικά ο ορισμός, η εφαρμογή και τα σχετικά αξιοσημείωτα άρθρα κάθε παραλλαγής. Μετά την πρώτη παρουσίαση του προβλήματος δρομολόγησης οχημάτων, Vehicle Routine Problem (VRP), των Dantzig και Ramster (1959), το πρόβλημα μπήκε σε εκτεταμένη έρευνα για σχεδόν 50 χρόνια. Η εγγενής συνάφεια του προβλήματος με εφαρμογές στην πραγματική ζωή και την αυξανόμενη πολυπλοκότητα που δημιουργούν οι επιχειρησιακοί περιορισμοί, αυξάνουν συνεχώς τις απαιτήσεις του VRP και καταβάλλονται συνεχώς προσπάθειες για την ανάπτυξη πιο πρακτικών μαθηματικών μοντέλων και αλγορίθμων με υψηλότερες επιδόσεις. Διάφορες παραλλαγές έχουν προσδιοριστεί ανάλογα με τις απαιτήσεις που δημιουργούν οι πραγματικές συνθήκες και τις τάσεις της κάθε εποχής. Οι πιο πρόσφατες παραλλαγές, αναφέρονται σε δυναμικές προσεγγίσεις του προβλήματος που μπορεί να προσδιορίζεται η επίλυση με upto-date δεδομένα. Μια άλλη στρατηγική κατεύθυνση του προβλήματος μελετήθηκε για πρώτη φορά το 2006, αναφέρεται στην

πράσινη δρομολόγηση οχημάτων, χαρακτηριστικά σημεία της έρευνας είναι το πρόβλημα δρομολόγησης της ρύπανσης και η δρομολόγηση στην αντίστροφη διοικητική μέριμνα (VRP in Reverse Logistics).

2.1.1 Πρόβλημα Δρομολόγησης Οχημάτων με περιορισμό Χωρητικότητας (1959)

Το πρόβλημα της δρομολόγησης οχημάτων συστάθηκε για πρώτη φορά από τον Dantzig και Ramster το 1959. Περιέγραψαν το πραγματικό πρόβλημα που παρατήρησαν κατά την διανομή πετρελαίου, από τον τερματικό σταθμό στα πρατήρια πώλησης. Παρατήρησαν ότι όταν ο αριθμός των πρατηρίων μεγάλωνε, η εύρεση μίας βέλτιστης λύσης ήταν σχεδόν αδύνατη. Προκειμένου να αντικαταστήσουν αυτή την αδυναμία που υπήρχε, ανέπτυξαν έναν αλγόριθμο, που σαν βάση είχε την γραμμική διατύπωση ακέραιων αριθμών για την επίτευξη μιας σχεδόν βέλτιστης λύσης. Στο πρόβλημα εξετάζεται ρητά και ο περιορισμός χωρητικότητας των οχημάτων (Capacitated VRP, CVRP). Αργότερα το πρόβλημα δρομολόγησης με χωρητικότητα (CVRP) λαμβανομένων των ιδιοτήτων του κόστους στον πίνακα της απόστασης, χωρίζεται σε συμμετρικό CVRP (SCVRP) και ασύμμετρο CVRP (ACVRP) (Toth & Vigo, 2002).

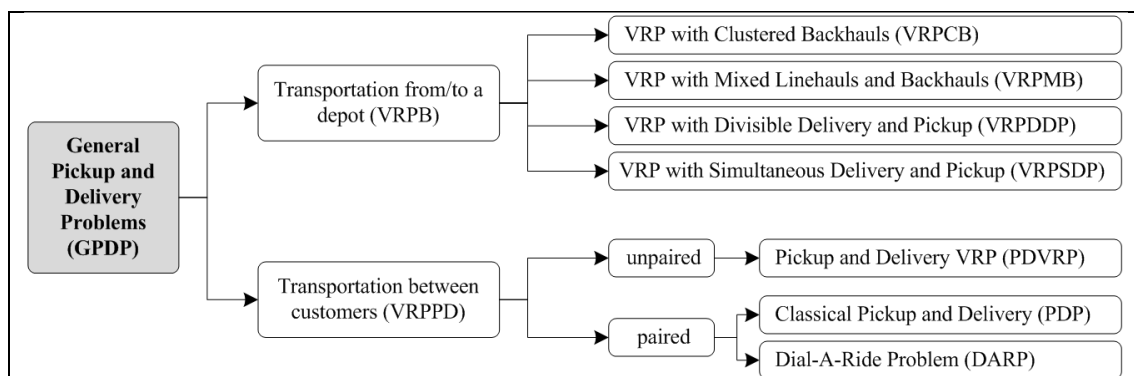
2.1.2 Χρονικά εξαρτώμενο Πρόβλημα Δρομολόγησης (1966)

Το χαρακτηριστικό του VRP, που εξαρτάται από τον χρόνο (TDVRP) είναι ότι, ο χρόνος ταξιδιού μεταξύ τοποθεσιών εξαρτάται από την απόσταση μεταξύ των σημείων ή την ώρα της ημέρας (π.χ. ώρες αιχμής, καιρικές συνθήκες). Αυτή η θεώρηση της κυμαινόμενης διάρκειας των ταξιδιών επιτρέπει στο VRP, να συνυπολογίζει πραγματικές συνθήκες όπως η αστική συμφόρηση. Ενά χρήσιμο μοντέλο για την αποκάλυψη των επαναλαμβανομένων προβλημάτων κυκλοφοριακής συμφόρησης (Lecluyse et al., 2013) και για την διερεύνηση του τρόπου αποφυγής τους (Kok et al., 2012). Όμως στην βασική θεώρηση χρησιμοποιείται η Ευκλείδεια απόσταση ως σταθερά, πράγμα που έρχεται σε αντίθεση με τις πραγματικές συνθήκες που τα οχήματα κινούνται σε οδικό δίκτυο. Συνεπώς, η εκτίμηση του κόστους απόστασης δεν είναι πειστική, διότι η διακύμανση του κόστους σε σχέση με το χρόνο έχει παραμεληθεί σε μεγάλο βαθμό (Polimeni & Vitetta, 2013). Μεγάλη προσοχή αργότερα προσέδωσε και το πρόβλημα TDVRP με Χρονικά Παράθυρα (TDVRP Time Windows), που διατυπώθηκε από τον Solomon (1987). Άλλες έρευνες της περίπτωσης του TDVRPTW περιλαμβάνουν τους Chen et al. (2006), Soler et al. (2009), Kritzing et al. (2012). Η χρονική εξάρτηση στο πρόβλημα της

δρομολόγησης έδωσε την δυνατότητα να γίνει εισαγωγή περισσότερων περιορισμών , οπότε και μεγαλύτερη προσέγγιση των πραγματικών συνθηκών. Η πιο πρόσφατη χρήση χρονικών περιορισμών καθιστούν δυνατή την χρήση του VRP, την μελέτη οικολογικών ζητημάτων στις μεταφορές και την μέτρηση εκπομπών που συνδέονται άμεσα με την κυμαινόμενη ταχύτητα σε πραγματικό χρόνο σε αστικές περιοχές.

2.1.3 Πρόβλημα Παραλαβής και Παράδοσης

Το πρόβλημα παραλαβής και παράδοσης (PDP), εξετάστηκε από τους Wilson & Weeissberg (1967). Στο ερευνητικό πεδίο του VRP, υπάρχουν πολλές κατηγορίες και υποκατηγορίες του PDP. Άλλες αναφέρονται σε VRP με ταυτόχρονη παραλαβή και παράδοση, σε πρόβλημα με προκαθορισμένες ώρες παραλαβών και παραδόσεων, κλπ. Ορισμένες από αυτές έχουν παρόμοια δομή με τη φύση του προβλήματος, αλλά έχουν μικρές διαφορές, συγκεντρωτικά αυτές οργανώνονται στο παρακάτω σχήμα.



2.1.4 Πρόβλημα Δρομολόγησης με Πολλαπλές Αφετηρίες (1969)

Μελετήθηκε για πρώτη φορά από τον Tillman (1969), με τη λογική ότι οι επιχειρήσεις συχνά χρησιμοποιούν περισσότερα από ένα, κέντρα διανομής (MDVRP). Κάθε πελάτης ανατίθεται σε μόνο ένα από αυτά τα κέντρα. Διάφορες επεκτάσεις αυτού του προβλήματος συζητούνται στην βιβλιογραφία, MDVRP με χρονικά παράθυρα (Giosa et al., 2002; Polacek et al., 2004; Dondo and Cerdá, 2007), MDVRP με παράδοση και παραλαβή (Nagy and Salhi, 2005). Ενδιαφέρον παρουσιάζει η περίπτωση MDVRP με Inter-depots (Angelelli and Speranza, 2002; Crevier et al., 2007) όπου οι ενδιάμεσες αποθήκες λειτουργούν είτε ως αποθήκες για ανεφοδιασμός σε σύστημα διανομής ή ως εγκαταστάσεις ανακύκλωσης για εκφόρτωση οχημάτων σύστημα συλλογής.

2.1.5 Στοχαστικό Πρόβλημα Δρομολόγησης (1969)

Σαν στοχαστικό VRP (SVRP), αναφέρεται οποιοδήποτε πρόβλημα δρομολόγησης όπου τα δεδομένα του, ζήτηση, χρόνος ταξιδιού, και ακόμα το πλήθος πελατών είναι τυχαία (Gendreau et al., 1996). Ως βασικό εργαλείο για την αποκατάσταση της αβεβαιότητας σε αυτά τα προβλήματα είναι η θεωρία πιθανοτήτων. Η στοχαστική προσέγγιση των προβλημάτων SVRP, κατηγοριοποιείται στην βιβλιογραφία σε όλες τις παραλλαγές VRP.

2.1.6 Πρόβλημα Δρομολόγησης Τοποθεσίας (1973)

Το πρόβλημα δρομολόγησης με τοποθεσία (LRP), είναι η γενίκευση του προβλήματος, δρομολόγησης με χωρητικότητα (CVRP) ή του προβλήματος δρομολόγησης με πολλαπλές αφετηρίες (MDVRP) χωρίς να απευθύνονται στο θέμα τοποθεσίας (Belenguer et al., 2011). Παρατηρήθηκε ότι ο ξεχωριστός σχεδιασμός των τοποθεσιών των αποθηκών και η δρομολόγηση των οχημάτων, δημιουργούσε μία μη βέλτιστη λύση, πράγμα που ενθάρρυνε την εμφάνιση του LRP (Watson-Gandy & Dohrn, 1973). Το 1998 έγινε κατηγοριοποίηση των LRP από διαφορετικές οπτικές, ανάλογα με τις ντετερμινιστικές ή στοχαστικές απαιτήσεις, με περιορισμό χωρητικότητας ή όχι, κτλπ. (Min et al., 1998).

2.1.7 Περιοδικό Πρόβλημα Δρομολόγησης (1974)

Ο Beltrami και ο Bodin (1974) ανέπτυξαν αλγόριθμους οι οποίοι έλυναν προβλήματα δρομολόγησης με χρονικούς περιορισμούς που οι τοποθεσίες απαιτούσαν διαφορετικό πλήθος επισκέψεων και συνδυασμό διαφορετικών ημερών για επίσκεψη μέσα σε μία βδομάδα. Δεδομένου του αυστηρού χρονοδιαγράμματος δρομολόγησης των πελατών, εξέλιξε το VRP, στο να καθορίζει τα δρομολόγια σε συγκεκριμένες μέρες μέσα στην εβδομάδα. Ο σκοπός ήταν η ελαχιστοποίηση του συνολικού κόστους των δρομολογίων σε χρονικό ορίζοντα εβδομάδας. Το πρόβλημα αυτό ονομάστηκε περιοδικό πρόβλημα δρομολόγησης (PVRP). Κύριες εφαρμογές αυτού του προβλήματος είναι η συλλογή σκουπιδιών και η διανομή φυσικού αερίου (Alegre et al., 2007). Στην βιβλιογραφία συναντιούνται και άλλες κατηγορίες περιοδικής δρομολόγησης, που εντάσσουν κι άλλους περιορισμούς όπως τα χρονικά παράθυρα (Bélangier et al., 2006; Pirkwieser and Gunther, 2009).

2.1.8 Δυναμικό Πρόβλημα Δρομολόγησης (1976)

Στο κλασικό VRP, γινόταν χρήση προκαθορισμένων δεδομένων σε σταθερό περιβάλλον και «offline» δεδομένα, και παρέμενε το ίδιο κατά την διάρκεια υλοποίησης του δρομολογίου. Στην πραγματικότητα όμως οι συνθήκες είναι μεταβαλλόμενες, όπως η κίνηση, οι βλάβες των αυτοκινήτων, κλπ. Η αβεβαιότητα αυτών των συνθηκών, επιλύθηκε σε ένα δυναμικό λειτουργικό περιβάλλον (Dynamic VRP). Το DVRP, χρονολογείται στον Spreidel (1976) και Psaraftis (1980), όπου τα δεδομένα των τοποθεσιών και παραγγελιών θα ανανεώνονταν κατά την διάρκεια του δρομολογίου. Η δυναμική προσέγγιση του προβλήματος ενθαρρύνεται από πολλές καθημερινές εφαρμογές όπως συστήματα διανομής (courier), υπηρεσίες επισκευής ή διάσωσης, υπηρεσίες ταξί, καθώς και υπηρεσίες έκτακτης ανάγκης (Ghiani et al., 2003). Η δυναμική επίλυση μπορεί να εφαρμοστεί σε όλες τις παραλλαγές του VRP. Αξίζει να αναφερθεί η περίπτωση της διακοπτόμενης δρομολόγησης οχημάτων (Disrupted VRP, DVRP), με αναδρομολόγηση και επαναπρογραμματισμό σε πραγματικό χρόνο (Li et al., 2009α· Mu et al., 2011). Η διακοπή των αρχικών σχεδίων δρομολόγησης οχημάτων μερικές φορές προκαλείται από απρόβλεπτα συμβάντα, όπως κυκλοφοριακή συμφόρηση, διακοπές ή ανακληθείσες αναχώρηση από τις αποθήκες ή τα σημεία εξυπηρέτησης πελατών (Mu και Eglese, 2011).

2.1.9 Πρόβλημα Δρομολόγησης Αποθεμάτων (1984)

Το πρόβλημα δρομολόγησης απογραφής αποθεμάτων (IRP), εξετάστηκε αρχικά από την Bell (1983), που αφορούσε την ολοκληρωμένη διαχείριση αποθεμάτων και διανομή. Το βασικό χαρακτηριστικό του IRP, είναι ότι εγγυάται ότι θα υπάρχουν αποθέματα για κάθε πελάτη. Αρχικά το πρόβλημα δεν μπορούσε να επιλυθεί σε πραγματικά δεδομένα, παρά μόνο σε προβλήματα με ένα δρομολόγιο ή με έναν πελάτη. Ο Archetti (2007) πρότεινε τον ακριβές αλγόριθμο που κατάφερε να λύσει το πρόβλημα με πάνω από 50 πελάτες, το πρόβλημα μετονομάστηκε σε VMI (Vender Managed Inventory). Ωστόσο παρέμενε η επίλυση στο ένα όχημα. Η επίλυση σε πραγματικές συνθήκες με την χρήση περισσότερων οχημάτων ήρθε το 2008 από το Zhao.

2.1.10 Μέγεθος Στόλου και Πρόβλημα Συνδυασμού Δρομολόγησης Οχημάτων(1984)

Σε πραγματικές συνθήκες, το σύνηθες πρόβλημα στα «logistic» και στην δρομολόγηση, είναι ποιο το πλήθος και ποια η χωρητικότητα που απαιτείται για την διανομή ενός προϊόντος (Golden et al., 1984). Ο στόχος του προβλήματος

δρομολόγησης με μέγεθος στόλου (Fleet Size and Mix VRP, FSMVRP), είναι να επιλύσει το πρόβλημα αυτό. Να ελαχιστοποιήσει δηλαδή, το κόστος και να βρεί τον βέλτιστο συνδυασμό κόστους οχημάτων και κόστους των υπόλοιπων μεταβλητών σε συνάρτηση με την απόσταση. Μεγαλύτερη πολυπλοκότητα εμφανίζουν τα προβλήματα που περιέχουν οχήματα με διαφορετικές χωρητικότητες. Επεκτάσεις του προβλήματος εντάσσουν περιορισμούς με χρονικά παράθυρα, πολλαπλές αφετηρίες ακόμη και των συνδιασμό των δύο αυτών σε πρόβλημα FDMVRP, για την συνολική βελτιστοποίηση της σύνθεσης δρομολογίων του στόλου οχημάτων (Dondo & Cerdá , 2007).

2.1.11 Γενικευμένο Πρόβλημα Δρομολόγησης (1984)

Στο γενικευμένο πρόβλημα VRP, οι πελάτες κατηγοριοποιούνται σε ομάδες και τα οχήματα υποχρεούνται να επισκέπτονται μόνο έναν πελάτη σε κάθε ομάδα (Ghiani & Improta, 2000). Το πρωτότυπο επίλυσης του VRP με αυτό τον τρόπο παρουσιάστηκε για πρώτη φορά από τον Tsiligirides (1984) και επεκτάθηκε σαν πρόβλημα ομαδοποίησης από τον Chao (1996). Η κατηγοριοποίηση γίνεται με βάση το κόστος ανάλογα με τον πελάτη καθώς και τους χρονικούς περιορισμούς που συνδέονται με αυτούς. Ποιο υποσύνολο των πελατών πρόκειται να επισκεφθούν, πώς θα αναθέσουν σε αυτούς τους επιλεγμένους πελάτες τα οχήματα και πώς θα αποστείλουν τα οχήματα, ώστε να επιτευχθεί το μέγιστο συνολικό κέρδος, θα καταστούν στόχοι και, ως εκ τούτου, θα καταστήσουν ένα πρόβλημα βελτιστοποίησης πολλαπλών επιπέδων όσον αφορά τη δρομολόγηση. Στην σύγχρονη βιβλιογραφία ο τρόπος επίλυσης αναφέρεται ως Team Orienting Problem.

2.1.12 Πρόβλημα Δρομολόγησης Πολλαπλών Αντικειμένων(1984)

Το 1985 ο Christophides, παρουσιάζει μια άλλη επέκταση του προβλήματος της δρομολόγησης (Multiple Compartments VRP, MCVRP). Κάθε πελάτης αιτείται μία παραγγελία η οποία συχνά περιλαμβάνει διαφορετικά αντικείμενα, τα οποία θα πρέπει να παραδοθούν από το ίδιο όχημα, ωστόσο πολλαπλές επισκέψεις επιτρέπονται σε αυτό το πρόβλημα. Το πρόβλημα αυτό εμφανίζεται κυρίως σε εταιρίες διανομής φαγητού ή διανομής καυσίμων. Εξαιτίας της πολυπλοκότητας του προβλήματος νέες μέθοδοι δοκιμάζονται για την επίλυση του. Ο El Fallahi (2008), ανέπτυξε έναν υβριδικό γενετικό αλγόριθμο με τοπική διαδικασία αναζήτησης (local search), ονομαστικά, μιμητικό αλγόριθμο και με χρήση «tabu search» για την επίλυση του MCVRP. Μια σύγχρονη επισκόπηση του MCVRP, που περιλαμβάνει μια σειρά

από 200 συγκριτικές μεθόδους επίλυσης καθώς και τα θετικά της MCVRP, συντάσσεται από τον Derigs (2011).

2.1.13 Ασαφές Πρόβλημα Δρομολόγησης (1995)

Σε πραγματικές εφαρμογές το πρόβλημα δρομολόγησης, διατυπώνεται από ασαφή δεδομένα όπως π.χ. «η ώρα παράδοσης 12:00 με 15:00 προτιμάται», «περίπου χρειάζονται 500-600 τεμάχια από το προϊόν». Σε αυτές τις συνθήκες απαιτείται ένα σύστημα που να αποκαταστεί την ασάφεια του προβλήματος, η λογική αυτή ονομάζεται «fuzzy logic» και μπορεί να χρησιμοποιηθεί στο VRP. Η πρώτη διερεύνηση έγινε στο VRP με ασαφή χρονικά παράθυρα και έπαιξε σημαντικό ρόλο στον τομέα των «logistics» (Cheng et al., 1995). Για την επίλυση του έγινε χρήση γενετικών αλγορίθμων, με σκοπό την εύρεση της βέλτιστης λύσης που θα ικανοποιούσε και τα υπόλοιπα δεδομένα του VRP. Στην συνέχεια πολλές προσπάθειες έγιναν και συνεχίζουν να γίνονται κάνοντας χρήση ερευνητικών, μετα-ευρετικών μεθόδων καθώς και περιορισμούς του εγγύτερου γείτονα, όπως αυτή του Xu (2011), με έναν αλγόριθμο που χώριζε το πρόβλημα σε δύο επιμέρους προβλήματα.

2.1.14 Ανοιχτό Πρόβλημα Δρομολόγησης (2000)

Ένα ανοιχτό πρόβλημα δρομολόγησης προτάθηκε από τον Sariklis και Powell (2000), όπου σε κάθε δρομολόγιο δεν υποχρεούταν το όχημα να επιστρέφει στη αφετηρία της διαδρομής (OVRP). Αυτή η εφαρμογή γίνεται εμφανής σε εταιρίες διανομή εφημερίδων ή γραμμάτων. Γενικότερα παρουσιάζεται σε οποιαδήποτε εταιρία που χρησιμοποιεί εξωτερικό συνεργάτη για την διανομή των προϊόντων της. Ο Brandao (2004), πρότεινε μία μέθοδο «tabu search» για την επίλυση του OVRP. Αρχικά γινόταν εισαγωγή μίας λύσης του προβλήματος που προέκυπτε από μία ευρετική μέθοδο εγγύτερου γείτονα και στην συνέχεια η λύση βελτιστοποιούνταν με «tabu search». Αρκετά πρόσφατα μελετήθηκε το πρόβλημα OVRP, σε ανομοιογενή περιορισμένο στόλο, που στην ουσία περιγράφει προσεκτικά την πραγματική κατάσταση μεταφορών από εξωτερικούς συνεργάτες της εταιρίας. Το πρόβλημα διατυπώθηκε και λύθηκε με συνδυασμό μετα-ευρετικών αλγορίθμων σε συνδυασμό με μια τροποποιημένη μέθοδο «tabu search» (Li et al., 2012).

2.1.15 Πρόβλημα Δρομολόγησης με Περιορισμό Φόρτωσης (2003)

Το πρόβλημα δρομολόγησης με περιορισμούς χωρητικότητας καθορίζει από κοινού τις βέλτιστες διαδρομές και τα πρότυπα συσκευασίας (Zachariadis et al., 2011). Ο

περιορισμός της χωρητικότητας, εμφανίστηκε για πρώτη φορά σε πρόβλημα πλανόδιου πωλητή (Travel Salesman Problem, TSP), το οποίο αναφερόταν σε παράδοση και παραλαβή προϊόντων (Ladany & Mehrez , 1984). Οι πιο σύγχρονες αναφορές που γίνονται στο πρόβλημα αυτό είναι για επίλυση του προβλήματος με δισδιάστατη απεικόνιση της χωρητικότητας των οχημάτων (Iori et al., 2003, 2007, Gendreau et al., 2008, Fuellerer et al., Zachariadis et al., Khebbache et al., 2009, Duhamel et al., Leung et al., 2011). Το πρόβλημα απαιτεί την ελαχιστοποίηση του συνολικού κόστους των διαδρομών, με ένα ορθογώνιο σχήμα συσκευασίας των ειδών στην δισδιάστατη επιφάνεια φόρτωσης κάθε οχήματος, χωρίς να υπερβαίνει το βάρος του οχήματος. Άλλες επεκτάσεις της VRPLC περιλαμβάνουν Δισδιάστατο Πρόβλημα Δρομολόγησης Παραλαβής και Παράδοσης. Αναφορές γίνονται και για τρισδιάστατη πύκνωση του VRP (Gendreau et al., 2006, Tarantilis et al., 2009).

2.1.16 Πρόβλημα Δρομολόγησης Multi-echelon (2009)

Multi – Echelon (Πολλαπλής Κλιμάκωσης) είναι μία στρατηγική βελτιστοποίησης των ροών σε μια μεγαλύτερη κλίμακα διανομής που αναφέρετε σε πολλές και διαφορετικού μεγέθους αποθήκες. Η παράδοση των εμπορευμάτων στο Multi-echelon VRP (MEVRP), γίνεται υποχρεωτικά μέσω ενδιάμεσης αποθήκης (Perboli et al., 2011). Αποσκοπεί στην ελαχιστοποίηση του συνολικού κόστους μεταφοράς των οχημάτων σε όλα τα επίπεδα. Τα συστήματα διανομής πολλαπλών κλιμάκιων προέρχονται κυρίως από το ηλεκτρονικό εμπόριο και υπηρεσίες οικιακής διανομής. Το συνηθέστερο παράδειγμα είναι την κύρια αποθήκη να συνδέεται με τις ενδιάμεσες αποθήκες (που ονομάζονται δορυφόροι) και το δεύτερο επίπεδο να συνδέει τους δορυφόρους με τους πελάτες. Ο Crainic και άλλοι (2009), διερεύνησαν πρόβλημα VRP δύο βαθμίδων κλιμάκωσης και με περιορισμό χωρητικότητας στο πλαίσιο πολεοδομικού σχεδιασμού. Εξετάστηκαν πολλαπλά ταξίδια, πολλαπλές αποθήκες, πολλαπλά προϊόντα, ετερογενή οχήματα, παράθυρα με μικρό χρόνο (σε πελάτες) και παράθυρα με μικρό χρόνο (σε δορυφόρους).

2.2 Πρόβλημα Δρομολόγησης με Χρονικά Παράθυρα (1977)

Οι ευρετικές προσεγγίσεις των VRP, δεν θεώρησαν τα χρονικά παράθυρα εξυπηρέτησης ή της ημερομηνίας λήξης ως περιορισμούς του μοντέλου μέχρι που ο Russell (1977), παρουσίασε έναν ευρετικό αλγόριθμο για την επίλυση του προβλήματος του πλανόδιου πωλητή. Πριν την μελέτη του Russell, το VRP με

χρονικά παράθυρα είχε ασχοληθεί σε μελέτες περιπτώσεων (Solomon ,1987). Γενικά υπάρχουν δύο περιπτώσεις που μελετώνται εκτενώς στην βιβλιογραφία:

- Αυστηρά χρονικά παράθυρα, Η καθυστερημένη άφιξη δεν είναι επιτρέπεται. Εάν το όχημα φτάσει νωρίτερα από το χρονικό διάστημα, πρέπει να περιμένει.
- Επιεική χρονικά παράθυρα, όπου παραβιάζεται ο περιορισμός του χρονικού παραθύρου αποδεκτή με την καταβολή ορισμένης ποινής (Kallehauge, 2008).

Τα αυστηρά παράθυρα τείνουν να περιγράφουν τις πραγματικές συνθήκες, ωστόσο δεν είναι πάντα εφικτή η επίλυση των προβλημάτων όταν απαιτείται να ικανοποιηθούν όλα τα χρονικά παράθυρα. Δημιουργώντας μια επιείκεια στην αυστηρότητα αυτού του περιορισμού, γίνεται δυνατή η επίλυση με σεβασμό στην συνολική απόσταση και στον αριθμό των οχημάτων. Επιπρόσθετα μια μικρή παράκληση από το χρονικό διάστημα που επιτρέπει ο πελάτης είναι κατανοητή και στην πραγματική ζωή. Η υιοθέτηση των χαλαρών χρονικών παραθύρων, αποκαταστεί το πρακτικό πρόβλημα παραβίασης των χρονικών παραθύρων σε πολλές καθημερινές εφαρμογές. Η χαλάρωση των παραθύρων σε δύσκολες ώρες μπορεί να οδηγήσει σε χαμηλότερο κόστος χωρίς να πληγεί σημαντικά η ικανοποίηση των πελατών (Figliozzi, 2010). Γενικά το πρόβλημα αναφέρεται ημι-χαλαρά χρονικά διαστήματα (Semi Soft Time Windows), στο οποίο η πρόωρη άφιξη επιτρέπεται χωρίς κόστος ενώ η καθυστερημένη άφιξη επιβαρύνεται με πρόστιμο.

Το πρόβλημα δρομολόγησης με χρονικά παράθυρα (VRP with Time Windows, VRPTW) είναι η πιο γνωστή παραλλαγή του VRP στην βιβλιογραφία. Η εισαγωγή των χρονικών διαστημάτων εξυπηρέτησης οδήγησαν σε μία ευρεία έρευνα και εφαρμογή σε πραγματικές συνθήκες. Οι πρόσφατες έρευνες στο VRPTW, δεν αποσκοπούν στην ελαχιστοποίηση του κόστους δρομολόγησης των οχημάτων. Στοχεύουν στην προσαρμογή τους με τις νέες στρατηγικές παροχής υπηρεσιών του αναπτυσσόμενου κλάδου των «logistics». Μελέτες δείχνουν ότι οι περιορισμοί των χρονικών διαστημάτων και η ζήτηση από τους πελάτες, επηρεάζουν την μέση απόσταση των προβλημάτων δρομολόγησης. Αυτό αποτελεί σημαντικό δείκτη για τον σχεδιασμό των δικτύων, την θέση των εγκαταστάσεων και το μέγεθος του στόλου οχημάτων. Αντί να χρησιμοποιηθούν κλασσικές ευρετικές, πλέον έχουν αναπτυχθεί μέθοδοι προσέγγισης με μοντέλα πιθανοτήτων για την εύρεση της μέσης απόστασης του συνολικού δρομολογίου (Figliozzi, 2009). Η σύγχρονες συνθήκες ζωής, απαιτούν

παράδοση πόρτα-πόρτα, η οποία είναι μια αυξανόμενη βιομηχανία εφοδιαστικής πόλεων, συχνά πάσχει από τη μεγάλη πίεση τόσο των χρονικών διαστημάτων εξυπηρέτησης που καθορίζονται από τον πελάτη όσο και από την απροσδόκητη διαταραχή των συνθηκών κυκλοφορίας σε αστικές περιοχές. Για την αντιμετώπιση των προκληθειών δυναμικών προβλημάτων αλλαγής πορείας, ο Qureshi και άλλοι, (2012) πρότεινε ένα δυναμικό πρόβλημα δρομολόγησης οχημάτων με το μοντέλο Soft Time Windows για να βοηθήσει τους μεταφορείς να αποφύγουν το πρόσθετο κόστος, καθώς και την καθυστέρηση της παράδοσης των εμπορευμάτων.

Μια αξιοσημείωτη παρατήρηση από τους Bektaş και Laporte (2011) είναι ότι ένας κατάλληλος χρονικός περιορισμός καθιστά τις επιπτώσεις της μείωσης της ενέργειας πιο σημαντικές. Με βάση αυτή τη σχέση, η μελλοντική έρευνα μπορεί να περιλαμβάνει την εξερεύνηση της σχέσης μεταξύ του οικονομικού κόστους (συμπεριλαμβανομένων των ποινών) και του περιβαλλοντικού κόστους για τη δρομολόγηση προβλημάτων με περιορισμούς παραθύρων λόγω ήπιου χρόνου.

2.3 Αλγόριθμοι & Βασικές Μέθοδοι Επίλυσης

Το πρόβλημα της δρομολόγησης οχημάτων, VRP, είναι ένα πρόβλημα συνδιαστικής βελτιστοποίησης NP-hard (nondeterministic polynomial time - hard). NP-hard στην θεωρία της υπολογιστικής πολυπλοκότητας θεωρείται η δυσκολία μη – προσδιορισμού πολυωνύμου, με καθοριστική ανεπίσημη ιδιότητα της κατηγορίας προβλημάτων "τουλάχιστον τόσο δύσκολα όσο τα δυσκολότερα προβλήματα στο NP". Η βέλτιστη ή σχεδόν βέλτιστη λύση γενικά επιτυγχάνεται με τη χρήση ακριβών αλγορίθμων ή κατά προσέγγιση αλγορίθμων. Οι ακριβείς αλγόριθμοι για το VRP ταξινομούνται σε τρεις μεγάλες κατηγορίες (Laporte, 1992):

1. Μέθοδοι απευθείας αναζήτησης δέντρων,
2. Δυναμικός προγραμματισμός,
3. Ακέραιος γραμμικός προγραμματισμός

Ωστόσο οι αλγόριθμοι κατά προσέγγιση μπορούν να βρουν σχεδόν βέλτιστες λύσεις για προβλήματα μεγάλης κλίμακας μέσα σε πολύ ικανοποιητικό χρόνο υπολογισμού, και επομένως χρησιμοποιούνται συχνά στην πράξη. Η βιβλιογραφία, περιλαμβάνει μια ποικιλία από ευρετικούς και μετα- ευρετικούς αλγορίθμους που επιλύουν

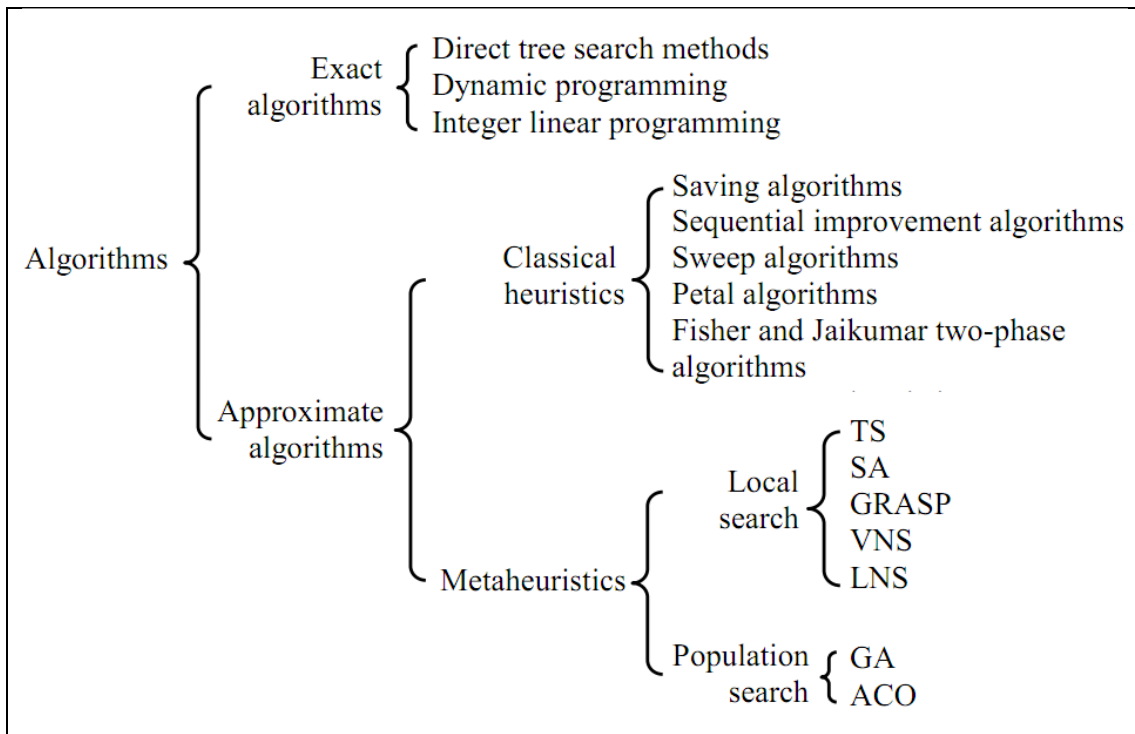
ικανοποιητικά τις διάφορες παραλλαγές του VRP. Στην έρευνα του ο Laporte, (2000) καθώς και ο Cordeau και άλλοι (2002), αναφέρουν τις βασικές κατηγορίες κλασικών ευρετικών αλγορίθμων:

1. Αλγόριθμοι αποθήκευσης
2. Αλγόριθμοι διαδοχικής βελτίωσης,
3. Αλγόριθμοι σάρωσης,
4. Αλγόριθμοι του πέταλου,
5. Αλγόριθμοι δύο φάσεων Fisher and Jaikumar,
6. Βελτιωτικών ευρετικών

Σε σύγκριση με τις κλασικές ευρετικές μεθόδους, οι μετα-ευρετικοί πραγματοποιούν μια πιο λεπτομερή έρευνα του πεδίου λύσεων, επιτρέποντας μεγαλύτερο εύρος στον ανασυνδυασμό λύσεων για την δημιουργία καλύτερων λύσεων. Ως αποτέλεσμα οι μετα-ευρετικοί αλγόριθμοι είναι ικανοί να παράγουν λύσεις υψηλής ποιότητας, παρά το μεγαλύτερο χρόνο υπολογισμού, σε αντίθεση με τους πρώιμους ευρετικούς αλγορίθμους (Cordeau et al., 2002). Οι μετα -ευρετικοί κατηγοριοποιούνται σε δύο τύπους:

1. Τοπική αναζήτηση (Local Search), οι τοπικές μέθοδοι που βασίζονται στην αναζήτηση συνεχίζουν να εξερευνούν το χώρο της λύσης μεταβαίνοντας διαδοχικά από την τρέχουσα λύση σε μια άλλη πολλά υποσχόμενη λύση στη γειτονιά της.
2. Αναζήτηση πληθυσμού. Οι μέθοδοι που βασίζονται στην αναζήτηση πληθυσμού διατηρούν ένα σύνολο καλών γονικών λύσεων, επιλέγοντας συνεχώς γονικές λύσεις για την παραγωγή υποσχόμενων απογόνων, ώστε να ενημερώνεται το σύνολο. Τυπικά παραδείγματα είναι: ο γενετικός αλγόριθμος (Genetic algorithm, GA), η βελτιστοποίηση αποικιών μυρμηγκιών (Ant Colony Optimization, ACO).

Το παρακάτω σχήμα συνοψίζει τη σχέση μεταξύ των προαναφερθέντων αλγορίθμων.



2.4 Υφιστάμενη Κατάσταση

2.4.1 Εθνική Στρατηγική για την Εφοδιαστική στην Ελλάδα

Στο χώρο των μεταφορικών υπηρεσιών έχει αποφασιστεί τα τελευταία χρόνια να εφαρμοστεί στρατηγική αναδιάρθρωσης η οποία συνοψίζεται σε πέντε βασικά σημεία:

1. Μείωση του κόστους μεταφοράς.
2. Βελτιστοποίηση της αξιοποίησης όλων των υπάρχοντων διαθέσιμων μέσων.
3. Παροχή της βέλτιστης δυνατής ποιότητας υπηρεσίας στον πελάτη.
4. Άμεση συνεχής πληροφόρηση σε όλα τα στάδια.
5. Συνεχής αναζήτηση νέων τρόπων και προοπτικών για απόκτηση πλεονεκτημάτων έναντι του ανταγωνισμού.

Η ανάγκη μείωσης του κόστους μεταφορών και προστασίας του περιβάλλοντος επιτάσσουν την δημιουργία ενός νέου πλέγματος παραγωγής και διάθεσης των μεταφορικών υπηρεσιών και μιας νέας μορφής εταιριών παροχής υπηρεσιών μεταφοράς, που θα συντονίζει όλα τα μεταφορικά μέσα.

Η ανάπτυξη του κλάδου των Logistics αναμένεται να διαδραματίσει σημαντικό ρόλο στην ανάκτηση της ελληνικής οικονομίας. Στα πλαίσια της παγκοσμιοποίησης και της ανάπτυξης του παγκόσμιου εμπορίου, θα απαιτηθεί να αξιοποιηθεί πλήρως η δυναμική της χώρας λόγω ναυτιλίας και γεωγραφικής θέσης. Στην υφιστάμενη κατάσταση της Ελλάδας ο κλάδος της εφοδιαστικής αλυσίδας υπολογίζεται στο 10% του ΑΕΠ, ενώ υπάρχουν πολλές δυνατότητες περαιτέρω ανάπτυξης. Βασικός σκοπός της εθνικής στρατηγικής είναι η ανάδειξη της Ελλάδας σε διαμετακομιστικό κόμβο της Νοτιο-Ανατολικής Ευρώπης.

Ωστόσο ορισμένες συνθήκες περιορίζουν την χρήση των logistics ως μέσο ανάπτυξης της Ελληνικής Οικονομίας. Το θεσμικό και νομικό πλαίσιο στον τομέα των μεταφορών δημιουργεί μια ανούσια και δυσχερή πολυπλοκότητα. Απλοποίηση και εκσυγχρονισμός των διαδικασιών απαιτείται και στον τομέα για την αδειοδότηση εταιριών logistics/μεταφορών και για την κατασκευή νέων εγκαταστάσεων διανομής. Τα ελληνικά logistics χρειάζεται προτυποποιηθούν με βάση τα διεθνή πρότυπα, να δοθούν χρηματοδοτήσεις και οικονομικά κίνητρα στις επιχειρήσεις καθώς και στην προώθηση πανεπιστημιακών σπουδών του κλάδου.

2.4.3 Οργανισμοί Έρευνας στη Δρομολόγηση και την Εφοδιαστική

Η πρακτική σημασία των προβλημάτων της δρομολόγησης και η δυσκολία τους ως συνδυαστικά προβλήματα βελτιστοποίησης έχουν κερδίσει το ενδιαφέρον την διεθνούς επιστημονικής κοινότητας. Στη Βόρειο Αμερική, όσοι ασχολούνται με τα θέματα της εφοδιαστικής αλυσίδας και των μεταφορών είναι μέλη της κοινότητας Transportation Science & Logistics (TSL) στο INFORMS (Ινστιτούτο Επιχειρήσεων και Διοικητικής Επιστήμης), ενώ στην Ευρώπη έχει ιδρυθεί μέσω του EURO (Association of European Operational Research Societies) μία ερευνητική ομάδα που ονομάζεται VeRoLog. Σκοπός του VeRoLog είναι να δημιουργήσει ένα περιβάλλον που θα ενθαρρύνεται η ανταλλαγή πληροφοριών μεταξύ των επαγγελματιών, θα ευνοείται η ανάπτυξη και εφαρμογή λειτουργικών ερευνητικών μοντέλων, μεθόδων και εργαλείων στον τομέα της δρομολόγησης οχημάτων και της εφοδιαστικής. Επίσης το VeRoLog προωθεί ενεργά το ενδιαφέρον για τη δρομολόγηση των οχημάτων και τα logistics στην επιστημονική κοινότητα με τη διοργάνωση ειδικών εργαστηρίων και συνεδρίων.

2.4.4 Υπολογιστικά Εργαλεία & Νέες Τεχνολογίες στη Δρομολόγηση Οχημάτων

Η μεγάλη πολυπλοκότητα και η ύπαρξη πολλών λειτουργικών παραμέτρων των προβλημάτων της δρομολόγησης οχημάτων και του σχεδιασμού δικτύων μεταφορών καθιστά απαραίτητη την χρήση υπολογιστών συστημάτων για την προσέγγιση της βέλτιστης λύσης. Οι πιο αποτελεσματικές μέθοδοι αντιμετώπισης είναι η χρήση λογισμικών προσομοίωσης των προβλημάτων, Γεωγραφικών Πληροφοριακών Συστημάτων (GIS) καθώς και ο συνδυασμός τους. Τα λογισμικά προσομοίωσης χρησιμοποιούν αλγορίθμους για την δομή και την επίλυση του προβλήματος της δρομολόγησης ενώ τα GIS, υποστηρίζουν την ψηφιακή παρουσίαση και οπτικοποίηση των αποτελεσμάτων.

Με τη σύγχρονη τεχνολογία τα πληροφοριακά συστήματα επιτρέπουν καλύτερη οργάνωση και διαχείριση των διαδρομών και συνεχώς προκύπτουν πιο καινοτόμες προσεγγίσεις. Τα τελευταία χρόνια έχουν αναπτυχθεί και εργαλεία λογισμικού που ενσωματώνουν υπηρεσίες τηλεματικής με ηλεκτρονική διαβίβαση δεδομένων μεταξύ οχημάτων και εταιριών έχοντας ως στόχο να καταστεί δυνατή η ταχύτερη αντίδραση των σχεδιαστών στη δυναμική του συστήματος μεταφοράς και στις ενδεχόμενες διαταραχές που προκαλούνται από την ποιότητα των οχημάτων ή από τις βαριές συνθήκες κυκλοφορίας στους δρόμους . Κάθε αυτοκίνητο της εταιρείας συνδέεται στο δίκτυο και επικοινωνεί και παρακολουθείται για όλες τις εργασίες που διεκπεραιώνει σε πραγματικό χρόνο. Η χρήση τέτοιων συστημάτων δίνει την δυνατότητα παρακολούθησης και αναδρομολόγησης των οχημάτων σε πραγματικό χρόνο.

2.5 Νομικό Πλαίσιο

N.4302/2014

ΦΕΚ

225/A/8-10-2014

Ρύθμιση θεμάτων Εφοδιαστικής και άλλες διατάξεις.

ΜΕΡΟΣ ΠΡΩΤΟ

ΚΕΦΑΛΑΙΟ Α

ΓΕΝΙΚΕΣ ΔΙΑΤΑΞΕΙΣ

Άρθρο 4

Αστική Μεταφορά, Προσωρινή Αποθήκευση και Διανομή Εμπορευμάτων

Με προεδρικό διάταγμα που εκδίδεται μετά από πρόταση των Υπουργών Υποδομών, Μεταφορών και Δικτύων, Εσωτερικών και Περιβάλλοντος, Ενέργειας και Κλιματικής Αλλαγής, και αφού προηγηθεί διαβούλευση επί του σχεδίου με την Κεντρική Ένωση Δήμων Ελλάδας, καθορίζονται οι κανόνες για τη μεταφορά και τη διανομή αγαθών και εμπορευμάτων εντός και περιμετρικά αστικής περιοχής με γνώμονα τη διασφάλιση της ελάχιστης δυνατής διατάραξης της αστικής κυκλοφορίας. Με το προεδρικό διάταγμα ρυθμίζονται ιδίως: α) Η δημιουργία ειδικών εγκαταστάσεων προσωρινής αποθήκευσης, ενοποίησης και αποομαδοποίησης φορτίων και φορτοεκφόρτωσης εμπορευμάτων και στάθμευσης φορτηγών αυτοκινήτων σε περιοχές των δήμων, προκειμένου να επιτευχθεί συγκεντρωμένη μεταφορά και εκφόρτωση προς τα καταστήματα και η ρυθμιζόμενη πρόσβαση φορτηγών αυτοκινήτων εντός και διαμέσου της πόλης.

1. Η χρήση καθαρών τεχνολογιών εντός του αστικού ιστού και οι ελάχιστες απαιτήσεις σε σχέση με τους όρους φορτοεκφόρτωσης εμπορευμάτων.
2. Οι όροι για τη χωροθέτηση των εγκαταστάσεων του παρόντος άρθρου σε εντός σχεδίου περιοχές.

Άρθρο 5

Πράσινη Εφοδιαστική

Με απόφαση του Υπουργού Περιβάλλοντος, Ενέργειας και Κλιματικής Αλλαγής καθορίζονται οι όροι λειτουργίας του συστήματος καταγραφής των περιβαλλοντικών επιδόσεων των επιχειρήσεων που δραστηριοποιούνται στην Εφοδιαστική, στους οποίους συμπεριλαμβάνεται και το αποτύπωμα άνθρακα, σε κεντρική βάση δεδομένων που τηρείται στο Υπουργείο Υποδομών, Μεταφορών και Δικτύων, καθώς και οι τρόποι γνωστοποίησης στην αγορά και στο ευρύ κοινό των επιχειρήσεων που δραστηριοποιούνται στην Εφοδιαστική, οι οποίες διατηρούν ή επαυξάνουν τις περιβαλλοντικές τους επιδόσεις ή εφαρμόζουν σύστημα περιβαλλοντικής διαχείρισης πιστοποιημένο από αρμόδιο φορέα.

Οι **κυριότερες αλλαγές** για την εφοδιαστική αλυσίδα, όπως αυτές περιγράφονται στο πολυνομοσχέδιο, είναι οι ακόλουθες:

Στον τομέα της «Εγκατάσταση μεταποιητικών και συναφών δραστηριοτήτων», μετά το άρθρο 48 του ν. 4442/2016 (Α' 230). απλουστεύεται ουσιωδώς η διαδικασία έκδοσης της έγκρισης εγκατάστασης, απλούστευση που θα αποτυπωθεί κυρίως στην υπουργική απόφαση που θα εκδοθεί κατ' εξουσιοδότηση των νέων διατάξεων και αφορά στη διαδικασία και τα δικαιολογητικά για την έγκριση εγκατάστασης.

Η εν λόγω ρύθμιση αποσκοπεί στην ενίσχυση της επιχειρηματικότητας που λόγω της μεγάλης οικονομικής κρίσης έχει πληγεί ιδιαίτερος και έχει μεγάλη σημασία τόσο κατά την περίοδο προσπαθειών ανάκαμψης της δραστηριότητας με στόχο αυτή να επανέλθει στην κανονική λειτουργία όσο και στη διευκόλυνση μεταβίβασης της δραστηριότητας στην περίπτωση που οι προσπάθειες ανάκαμψης υπό τον ίδιο φορέα δεν αποδώσουν.

Ωστόσο δεδομένο πως μέχρι σήμερα οι αδειοδοτούσες αρχές οφείλουν να διενεργούν ελέγχους σε κάθε καταγγελία, γεγονός που ελλείπει ανθρώπινων πόρων δυσχεραίνει την εύρυθμη λειτουργία της υπηρεσίας, καθώς και η έντονη παρουσία της γραφειοκρατίας, καταστείλουν τις περισσότερες πρωτοβουλίες για εξέλιξη του κλάδου.

Στο υποκεφάλαιο υποκεφάλαιο β' ρυθμίσεις για την εφοδιαστική του άρθρου 159 του ν. 4442/2016 στην παράγραφο 1 ορίζεται ότι για την εγκατάσταση Κέντρων Αποθήκευσης και Διανομής τα οποία **κατατάσσονται περιβαλλοντικά** λόγω των σημαντικών περιβαλλοντικών τους επιπτώσεων στην κατηγορία Α' και τις υποκατηγορίες Α1 ή Α2, απαιτείται έγκριση εγκατάστασης. Έγκριση εγκατάστασης απαιτείται ακόμα και για τα Κέντρα Αποθήκευσης και Διανομής που κατατάσσονται περιβαλλοντικά στην κατηγορία Β' εφόσον αυτά εγκαθίστανται σε περιοχές που βρίσκονται εκτός εγκεκριμένου χωρικού σχεδιασμού.

Κεφάλαιο 3

3.1 Στόχοι έρευνας

Αντικείμενο της παρούσας διπλωματικής εργασίας είναι η διερεύνηση θεμάτων διανομής και η ανάπτυξη της μεθοδολογίας σε γλώσσα προγραμματισμού για την επίλυση των βασικών προβλημάτων δρομολόγησης οχημάτων VRP. Η μοντελοποίηση βασίζεται στο ελεύθερο λογισμικό OR-Tools και γίνεται εφαρμογή

στα προβλήματα, πλανόδιου πωλητή, δρομολόγηση στόλου οχημάτων και δρομολόγηση στόλου οχημάτων με περιορισμό χωρητικότητας και χρονικών παραθύρων.

Συγκεκριμένα η έρευνα εστιάζει στον υπολογισμό των βέλτιστων διαδρομών, λαμβάνοντας τα τεχνικά χαρακτηριστικά της μεταφοράς σε κάθε πρόβλημα. Στο πρόβλημα του πλανόδιου πωλητή (TSP), υπάρχει ένα μόνο όχημα και απαιτείται να βρεθεί η διαδρομή με την μικρότερη απόσταση που εξυπηρετεί όλους τους πελάτες. Στην συνέχεια το πρόβλημα της δρομολόγησης στόλου οχημάτων (VRP), απαιτεί την ελαχιστοποίηση της συνολικής απόστασης, με την διαφορά ότι πρόκειται για περισσότερα του ενός οχήματα. Η επόμενη ενότητα εξετάζει την παραλλαγή του VRP, με τον περιορισμό της χωρητικότητας των οχημάτων. Το πρόβλημα δρομολόγησης με περιορισμό χωρητικότητας (CVRP), περιορίζει κάθε όχημα να έχει συγκεκριμένο αριθμό αντικειμένων για παράδοση, συνεπώς ένα άδειο όχημα δεν μπορεί να επισπευτεί κάποιον προορισμό. Τέλος αναλύεται το πρόβλημα της δρομολόγησης οχημάτων με χρονικά παράθυρα (VRPTW), στην ουσία γίνεται προσθήκη των χρονικών διαστημάτων που είναι διαθέσιμοι οι πελάτες/τοποθεσίες για επίσκεψη.

Με τον τρόπο αυτό προσπαθεί να γίνει μία παρουσίαση του προγράμματος για την επίλυση προβλημάτων βελτιστοποίησης στην δρομολόγηση οχημάτων. Από το πιο απλοποιημένο πρόβλημα της δρομολόγησης ενός οχήματος, χτίζεται το πρόγραμμα σταδιακά σε ενότητες, για να μπορεί εν τέλει να ανταποκριθεί στις απαιτήσεις που συναντώνται σε πραγματικές συνθήκες. Τα χρονικά παράθυρα και η χωρητικότητα των οχημάτων, είναι δύο βασικοί περιορισμοί που απασχολούν το μεγαλύτερο ποσοστό προβλημάτων δρομολόγησης, ωστόσο υπάρχουν και άλλοι περιορισμοί που προσεγγίζουν καλύτερα κάποιες εφαρμογές διανομής. Δεν γίνεται να δημιουργηθεί ένα μοντέλο το οποίο θα επιλύει αποτελεσματικά όλες τις παραλλαγές VRP, αλλά μπορεί να προσαρμοστεί στις διαφορετικές ανάγκες του προβλήματος εισάγοντας ή αφαιρώντας περιορισμούς.

3.2 Είδος έρευνας

Εφαρμόστηκε προγραμματιστική μέθοδος για την επίλυση των προβλημάτων δρομολόγησης σε γλώσσα προγραμματισμού python και τα αποτελέσματα παρουσιάζονται με την βοήθεια του γεωγραφικού συστήματος πληροφοριών (GIS).

Για την αποτελεσματική επίλυση των προβλημάτων έγινε χρήση του διακομιστή επίλυσης OR-Tools. Η δομή και παρουσίαση των ενοτήτων γίνεται με σκοπό να υπάρξει μία κλιμάκωση της πολυπλοκότητας των προβλημάτων, που θα αναδείξει τις διαφορές, αλλά και την ευκολία προσέγγισης των πραγματικών συνθηκών μέσα από προγραμματιστικές μεθόδους. Η έρευνα βασίστηκε σε πρωτογενή δεδομένα που συλλέχθηκαν από το φαρμακευτικό κέντρο διανομής του Αγρινίου.

3.3 Google's OR-Tools

Τα Εργαλεία βελτιστοποίησης Google (OR-Tools) [16] είναι μια λύση ανοικτού κώδικα για συνδυαστικά προβλήματα βελτιστοποίησης. Τα OR-Tools περιέχουν μία από τις καλύτερες διαθέσιμες λύσεις VRP, η οποία έχει υλοποιήσει πολλές ευρισκόμενες πρακτικές (π.χ., Clarke-Wright), Sweep heuristic [36], Christofides' heuristic [8] και μερικές άλλες για την εξεύρεση αρχικής λύσης και μετα-ευρετικών καθοδηγούμενης τοπικής αναζήτησης [34], αναζήτηση πινάκων (tabu search [14]) και προσομοίωση προσάρτησης (simulated annealing [21]) για την αποφυγή τοπικών ελαχίστων στην αναζήτηση της καλύτερης λύσης. Η προεπιλεγμένη έκδοση του προγράμματος επίλυσης VRP OR-Tools χρησιμοποιείται ως βάση. Ο πρώτος περιορισμός είναι ότι το OR-Tools δέχεται μόνο ακέραιες θέσεις για τους πελάτες και τις αποθήκες. Η δεύτερη διαφορά είναι ότι το OR-Tools ορίζεται για ένα VRP με πολλαπλά οχήματα, καθένα από τα οποία μπορεί να έχει το πολύ μία περιήγηση.

4.1 Το Πρόβλημα του Πλανόδιου Πωλητή

Το γνωστότερο και πιο απλό πρόβλημα της επιχειρησιακής έρευνας στα δίκτυα διανομής είναι αυτό του πλανόδιου πωλητή (ΠΠΠ) - Traveling Salesman Problem (TSP). Το πρόβλημα αυτό συνίσταται στην εύρεση μίας μόνο διαδρομής που να συνδέει k πόλεις (κόμβους) με τέτοιο τρόπο ώστε να ελαχιστοποιείται η συνολική διανυθείσα απόσταση. Κάθε σημείο πρέπει να δέχεται επίσκεψη μόνο μία φορά.

Παρά την ιδιαίτερα απλή διατύπωσή του, το πρόβλημα είναι NP-hard, με αποτέλεσμα να είναι πρακτικά αδύνατη η αναλυτική επίλυση ακόμα και σε περιπτώσεις μικρών προβλημάτων.

Αν αντιστοιχιστεί ένα μονοπάτι για κάθε ζευγάρι σημείων, ο συνολικός αριθμός δυνατών λύσεων για k σημεία είναι $k!/2$. Έτσι, για $k=10$ ο συνολικός αριθμός διαδρομών είναι 1.814.400, ενώ για $k=15$ οι διαφορετικές διαδρομές φτάνουν τις 653.837.184.000. Αποτέλεσμα είναι να μην μπορεί να ευρεθεί η βέλτιστη λύση για το πρόβλημα ακόμα και για μικρές σχετικά τιμές του k και ως εκ τούτου έχουν προταθεί διάφοροι ευρετικοί αλγόριθμοι που σε σχετικά μικρό υπολογιστικό χρόνο παρέχουν αρκετά καλές λύσεις.

Παράδειγμα επίλυσης του προβλήματος του Πλανόδιου Πωλητή σε πραγματικά δεδομένα στην εταιρία διανομής φαρμάκων, Pharmanet .

Σε αυτή την ενότητα θα παρουσιαστεί ένα παράδειγμα που δείχνει πώς επιλύετε το πρόβλημα των πλανόδιων πωλητών (TSP) για τις τοποθεσίες που εμφανίζονται στον παρακάτω χάρτη.



Οι τοποθεσίες αντιστοιχούν στους πελάτες (κόμβους), και ο στόλος οχημάτων απαρτίζεται από ένα όχημα που θα πρέπει να επισκεφτεί όλους τους πελάτες χρησιμοποιώντας την ελάχιστη συνολική απόσταση, ξεκινώντας από την αποθήκη (Depot).

4.2 Παρουσίαση του αλγόριθμου

Παρακάτω παρουσιάζεται η δομή του προγράμματος που χρησιμοποιήθηκε σε μορφή ψευδοκώδικα. Αναλυτικότερα ο κώδικας παρουσιάζεται στις επόμενες ενότητες.

```
1  Παράσταση Αλγορίθμου
2  Ψευδοκώδικας
3
4  Αρχή
5  ##εισαγωγή λειτουργικής μονάδας OR-Tools
6
7  #ορισμός συνάρτησης αρχικοποίησης δεδομένων του προβλήματος
8
9  #ορισμός συνάρτησης εκτύπωσης την λύσης
10
11  #ορισμός main():
12
13      (1) αρχικοποίηση δεδομένων
14
15      (2) δημιουργία μοντέλου διαχείρισης ευρετηρίου δρομολόγησης
16          //εσωτερικοί δείκτες μεταβλητών που χρησιμοποιεί
17          //ο διακομιστής επίλυσης του προβλήματος
18
19      (3) δημιουργία του μοντέλου δρομολόγησης
20
21      #ορισμός συνάρτησης επιστροφής
22          //αντιστοίχιση των αποστάσεων των αρχικοποιημένων δεδομένων
23          //με τον διακομιστή επίλυσης
24
25      (4) ορισμός κόστους διαδρομής
26
27      (5) καθορισμός ευρετικού αλγορίθμου
28          //για την εύρεση μιας πρώτης λύσης
29
30      (6) επίλυση του προβλήματος
31
32      (7) εκτύπωση της λύσης
33
34  Τέλος
```

Οι παρακάτω ενότητες παρουσιάζουν προγράμματα σε Python που λύνουν το TSP χρησιμοποιώντας OR-Tools

4.3 Δημιουργία δεδομένων

Ο παρακάτω κώδικας δημιουργεί τα δεδομένα (data) για το πρόβλημα.

```
7 def create_data_model():
8     """Stores the data for the problem."""
9     data = {}
10    data['distance_matrix'] = [
11        [0, 1778, 2086, 885, 2834, 2741],
12        [1778, 0, 2203, 1793, 3199, 3058],
13        [2086, 2203, 0, 1214, 996, 855],
14        [885, 1793, 1214, 0, 1962, 1862],
15        [2834, 3199, 996, 1962, 0, 147],
16        [2741, 3058, 3058, 1862, 147, 0],
17    ]
18    data['num_vehicles'] = 1
19    data['depot'] = 0
20    return data
```

Ο πίνακας αποστάσεων ('distance_matrix') είναι ένας πίνακας του οποίου η καταχώρηση i, j είναι η απόσταση από το σημείο i έως το σημείο j σε μίλια, όπου οι δείκτες πίνακα αντιστοιχούν στις θέσεις με την ακόλουθη σειρά:

0. Αποθήκη Pharmed, 1.Αγ.Νικόλαος, 2.Βόνιτσα, 3.Σαρδηνία, 4.Αμφιλοχία, 5.Κατούνα, 6.Μύτικας, 7.Λεπενού, 8.Νεάπολη, 9.Καραϊσκάκι, 10.Αστακός, 11.Αγγελόκαστρο, 12.Σταμνά, 13.Λεσίνι, 14.Κατοχή, 15.Αιτωλικό, 16.Μεσσολόγγι, 17.Ευηνοχώρι, 18.Ναύπακτος, 19.Κάτω μακρινού, 20.Γαβαλού, 21.Ματαράγκα, 22.Ζευγαράκι, 23.Παναιτώλιο, 24.Καινούργιο, 25.Παραβόλα, 26.Θέρμο

Σημείωση: Η σειρά των θέσεων στον πίνακα αποστάσεων είναι αυθαίρετη και δεν σχετίζεται με τη σειρά των θέσεων σε οποιαδήποτε λύση του TSP.

Τα δεδομένα ακόμα περιλαμβάνουν:

- Τον αριθμό οχημάτων του προβλήματος, στην συγκεκριμένη περίπτωση 1 επειδή είναι πρόβλημα πλανόδιου πωλητή (για το πρόβλημα δρομολόγησης οχημάτων (VRP), ο αριθμός των οχημάτων μπορεί να είναι μεγαλύτερος από 1, όπως φαίνεται στην επόμενη ενότητα)
- Η αποθήκη ορίζεται με την θέση 0, που αντιστοιχεί στην θέση του εργοστασίου παραγωγής.

- ❖ Καθώς η επίλυση της δρομολόγησης από τον “solver” εκτελεί όλους τους υπολογισμούς με ακέραιους αριθμούς, η συνάρτηση επιστροφής των αποστάσεων από τον πίνακα “distance_matrix” πρέπει να επιστρέφει μια ακέραια απόσταση για δύο θέσεις. Εάν κάποια από τις καταχωρήσεις δεν είναι ακέραιοι, πρέπει να στρογγυλοποιηθούν είτε οι καταχωρήσεις του πίνακα είτε τις τιμές που επιστρέφουν από την συνάρτηση, δηλαδή να επέλθει μία κλιμάκωση του πίνακα αποστάσεων.

Κλιμάκωση του πίνακα απόστασης

Για να αποφευχθεί οποιοδήποτε πρόβλημα με τη στρογγυλοποίηση, μπορεί να κλιμακωθεί ο πίνακα απόστασης: πολλαπλασιάζοντας όλες τις καταχωρήσεις του πίνακα με έναν μεγάλο αριθμό ας πούμε 100. Αυτό πολλαπλασιάζει το μήκος οποιασδήποτε διαδρομής με ένα συντελεστή 100, αλλά δεν αλλάζει τη λύση. Το πλεονέκτημα είναι ότι τώρα όταν στρογγυλοποιηθούν οι καταχωρήσεις του πίνακα, το ποσό στρογγυλοποίησης είναι πολύ μικρό σε σύγκριση με τις αποστάσεις, επομένως δεν θα επηρεάσει σημαντικά τη λύση.

Εάν γίνει κλιμάκωση στον πίνακα αποστάσεων, πρέπει επίσης να γίνει αλλαγή στην εκτύπωση της λύσης, δηλαδή να διαιρεθούν τα μήκη της κλιμακούμενης διαδρομής με το συντελεστή κλιμάκωσης, ώστε να εμφανίζει τις πραγματικές αποστάσεις των διαδρομών.

Άλλοι τρόποι δημιουργία του πίνακα δεδομένων

Σε αυτό το παράδειγμα, ο πίνακας αποστάσεων ορίζεται ρητά στο πρόγραμμα. Είναι επίσης δυνατό να χρησιμοποιηθεί συνάρτηση για να υπολογίζει τις αποστάσεις μεταξύ των σημείων και να συμπληρώνει τον πίνακα. Μια άλλη εναλλακτική λύση είναι η χρήση του Google Maps Distance Matrix API για τη δυναμική δημιουργία ενός πίνακα αποστάσεων για ένα πρόβλημα δρομολόγησης. Η αξία αυτής της δυναμικής συμπλήρωσης του πίνακα αποστάσεων γίνεται αντιληπτή στις επόμενες ενότητες κατά την επίλυση προβλημάτων δρομολόγησης πολλαπλών οχημάτων.

Το Distance Matrix API είναι μια υπηρεσία που παρέχει απόσταση και χρόνο ταξιδιού για έναν πίνακα προελεύσεων και προορισμών, με βάση τη συνιστώμενη διαδρομή μεταξύ των σημείων έναρξης και λήξης. Οι πληροφορίες που συνοδεύουν την συνιστώμενη διαδρομή μεταξύ των σημείων, παρέχονται ανάλογα με το αίτημα

του χρήστη. Ο χρήστης μπορεί να ζητήσει δεδομένα αποστάσεων για διαφορετικούς τρόπους ταξιδιού, σε διαφορετικές μονάδες (χιλιόμετρα, μίλια) και την εκτίμηση χρόνου ταξιδιού ανάλογα με τον κυκλοφοριακό φόρτο.

4.4 Δημιουργία Μοντέλου Δρομολόγησης

Ο παρακάτω κώδικας στο κύριο τμήμα του αλγόριθμου δημιουργεί έναν δείκτη (manager) και το μοντέλο δρομολόγησης (routing). Ο επιλύτης (solver) της δρομολόγησης, χρησιμοποιεί εσωτερικούς δείκτες για τις μεταβλητές του προβλήματος. Αυτοί οι δείκτες μεταβλητών είναι δύσκολο να διαχειριστούν άμεσα, επειδή ένας κόμβος μπορεί να αντιστοιχεί σε ένα πλήθος μεταβλητών, ανάλογα με τον αριθμό που εμφανίζονται στο μοντέλο και εάν χρησιμοποιούνται ως σημεία έναρξης ή/και λήξης. Η κλάση (manager) αποσκοπεί στην απλοποίηση της χρήσης μεταβλητού ευρετηρίου (NodeIndex), επιτρέποντας στους χρήστες να χρησιμοποιούν τις μεθόδους της κλάσης όποτε το μοντέλο απαιτεί μεταβλητό δείκτη. Το μοντέλο δρομολόγησης (routing) δημιουργείται με την μέθοδο RoutingModel, δέχεται σαν παράμετρο τον δείκτη που κατασκευάστηκε από την μέθοδο RoutingIndexManager.

```
41 # Instantiate the data problem.
42 data = create_data_model()
43 # Create the routing index manager.
44 manager = pywrapcp.RoutingIndexManager(len(data['distance_matrix']),
45                                       data['num_vehicles'], data['depot'])
46 # Create Routing Model.
47 routing = pywrapcp.RoutingModel(manager)
```

Η μέθοδος RoutingIndexManager δημιουργεί ένα μεταβλητό ευρετήριο για αντιστοίχιση ευρετηρίου μεταβλητών για ένα πρόβλημα που περιέχει αριθμό κόμβων, αριθμό οχημάτων και τις δοσμένες αρχές και άκρα για κάθε διαδρομή/όχημα. Τα δεδομένα εισόδου είναι:

- Ο αριθμός των γραμμών του πίνακα απόστασης, που είναι ο αριθμός των θέσεων (συμπεριλαμβανομένης της αποθήκης) `len(('distance_matrix'))`
- Ο αριθμός των οχημάτων στο πρόβλημα `('num_vehicles')`
- Ο κόμβος που αντιστοιχεί στην αποθήκη `('depot')`

- ❖ Στο συγκεκριμένο πρόβλημα παίρνουμε σαν δεδομένο ότι όλα τα οχήματα ξεκινούν και τελειώνουν σε μία μόνο τοποθεσία, την αποθήκη. Ωστόσο μπορεί να υπάρχουν περισσότερες από μία αποθήκες ή γενικότερα το πρόβλημα να απαιτεί διαφορετικές θέσεις αρχής και τέλους για κάθε όχημα. Σε αυτή την περίπτωση, θα προστεθούν δύο διανύσματα στην αρχική εισαγωγή των δεδομένων (`create_data_model()`), τα οποία περιέχουν τους δείκτες των θέσεων έναρξης ('start') και λήξης ('end'), ως δεδομένα εισόδου στη μέθοδο `RoutingModel`. Όπως φαίνεται στην συνέχεια.

```
data['starts'] = [1, 2, 15, 16]
data['ends'] = [0, 0, 0, 0]
```

4.5 Δημιουργία Συνάρτησης Επιστροφής

Για να λειτουργήσει σωστά ο επιλυτής (solver), χρειάζεται μια συνάρτηση επιστροφής που να παίρνει οποιοδήποτε ζευγάρι τοποθεσιών και να επιστρέφει την απόσταση μεταξύ τους. Ο ευκολότερος τρόπος για να γίνει αυτό είναι με την χρήση του πίνακα αποστάσεων ('`distance_matrix`') και του εσωτερικού δείκτη της κλάσης `manager`. Η ακόλουθη συνάρτηση δέχεται σαν παραμέτρους δύο δείκτες, `from_index`, `to_index`, και επιστρέφει την αντίστοιχη καταχώρηση από τον πίνακα αποστάσεων.

```
50 def distance_callback(from_index, to_index):
51     """Returns the distance between the two nodes."""
52     # Convert from routing variable Index to distance matrix NodeIndex.
53     from_node = manager.IndexToNode(from_index)
54     to_node = manager.IndexToNode(to_index)
55     return data['distance_matrix'][from_node][to_node]
56
57 transit_callback_index = routing.RegisterTransitCallback(distance_callback)
```

4.6 Ορισμός Κόστους Διαδρομής

Για τον υπολογισμό του κόστους διαδρομής χρησιμοποιείται η μέθοδος δρομολόγησης, `CostEvaluator`, η οποία δίνει στον επιλυτή τον τρόπο με τον οποίο θα υπολογίσει το κόστος διαδρομής μεταξύ οποιονδήποτε αποστάσεων. Στο πρόβλημα του πλανόδιου πωλητή, ο εκτιμητής κόστους είναι ο δείκτης `transit_callback_index`, που αποτελεί την εσωτερική αναφορά του επιλυτή στην συνάρτηση επιστροφής απόστασης. Αυτό σημαίνει ότι το κόστος ταξιδιού μεταξύ δύο τοποθεσιών είναι

ακριβώς η απόσταση μεταξύ τους. Ωστόσο, γενικά, το κόστος μπορεί να περιλαμβάνει και άλλους παράγοντες.

```
59 # Define cost of each arc.  
60 routing.SetArcCostEvaluatorOfAllVehicles(transit_callback_index)
```

4.6 Καθορισμός Παραμέτρων Αναζήτησης

Ο παρακάτω κώδικας θέτει τους προκαθορισμένους παραμέτρους αναζήτησης και μια ευρετική μέθοδο για να βρει γρήγορα μια πρώτη λύση στο πρόβλημα.

```
62 # Setting first solution heuristic.  
63 search_parameters = pywrapcp.DefaultRoutingSearchParameters()  
64 search_parameters.first_solution_strategy = (  
65     routing_enums_pb2.FirstSolutionStrategy.PATH_CHEAPEST_ARC)
```

Ο κώδικας ορίζει την πρώτη στρατηγική λύσης στο PATH_CHEAPEST_ARC, το οποίο δημιουργεί μια αρχική διαδρομή για το διακομιστή επίλυσης προσθέτοντας επανειλημμένα άκρα με το μικρότερο βάρος που δεν οδηγούν σε κόμβο που έχει ήδη επισκεφθεί κανείς (εκτός από την αποθήκη). Υπάρχουν κι άλλες επιλογές για εύρεση της πρώτης λύσης, χαρακτηριστικά οι μέθοδοι οργανώνονται σε τρεις ομάδες, ευρετικές με insertion, μετα-ευρετικές και constraint propagation.

4.7 Συνάρτηση Εκτύπωσης

Η συνάρτηση εμφανίζει την λύση που επιστράφηκε από τον διακομιστή επίλυσης. Εμφανίζει την βέλτιστη διαδρομή και την απόδοση της, που δίνεται από την μέθοδο ObjectiveValue().

```
23 def print_solution(manager, routing, solution):  
24     """Prints solution on console."""  
25     print('Objective: {} miles'.format(solution.ObjectiveValue()))  
26     index = routing.Start(0)  
27     plan_output = 'Route for vehicle 0:\n'  
28     route_distance = 0  
29     while not routing.IsEnd(index):  
30         plan_output += ' {} ->'.format(manager.IndexToNode(index))  
31         previous_index = index  
32         index = solution.Value(routing.NextVar(index))  
33         route_distance += routing.GetArcCostForVehicle(previous_index, index, 0)  
34     plan_output += ' {}\n'.format(manager.IndexToNode(index))  
35     print(plan_output)  
36     plan_output += 'Route distance: {}miles\n'.format(route_distance)
```

Συνάρτηση Αποθήκευσης σε Λίστα

Εναλλακτικά, αντί να γίνει εκτύπωση, μπορεί να αποθηκευτεί η διαδρομή (ή τις διαδρομές, για ένα VRP) σε μια λίστα ή συστοιχία. Αυτό έχει το πλεονέκτημα ότι καθιστά διαθέσιμες τις διαδρομές σε περίπτωση που χρειάζονται περαιτέρω επεξεργασία. Για παράδειγμα, μπορεί να εκτελεστεί το πρόγραμμα αρκετές φορές με διαφορετικές παραμέτρους και να αποθηκευτούν, οι διαδρομές-λύσεις που επιστρέφονται, σε ένα αρχείο για σύγκριση.

```
def get_routes(solution, routing, manager):
    """Get vehicle routes from a solution and store them in an array."""
    # Get vehicle routes and store them in a two dimensional array whose
    # i,j entry is the jth location visited by vehicle i along its route.
    routes = []
    for route_nbr in range(routing.vehicles()):
        index = routing.Start(route_nbr)
        route = [manager.IndexToNode(index)]
        while not routing.IsEnd(index):
            index = solution.Value(routing.NextVar(index))
            route.append(manager.IndexToNode(index))
        routes.append(route)
    return routes
```

4.8 Εκτέλεση Προγραμμάτων

Τελικά αφού έχει ολοκληρωθεί η δομή του προγράμματος, μπορεί να γίνει η κλήση στον διακομιστή επίλυσης και να εκτυπωθεί η λύση του προβλήματος.

```
67         # Solve the problem.
68         solution = routing.SolveWithParameters(search_parameters)
69
70         # Print solution on console.
71         if solution:
72             print_solution(manager, routing, solution)
```

4.9 Παρουσίαση Λύσης σε Χάρτη

Ο αλγόριθμος εμφανίζει την παρακάτω λύση,

```
Objective: 7871 miles
Route for vehicle 0:
0 -> 3 -> 5 -> 4 -> 2 -> 1 -> 0
```

Υπενθυμίζεται ότι, 0. Αποθήκη Pharmanet, 1.Αγ.Νικόλαος, 2.Βόνιτσα, 3.Σαρδηνία, 4.Αμφιλοχία, 5.Κατούνα, 6.Μύτικας, 7.Λεπενού, 8.Νεάπολη, 9.Καραϊσκάκι, 10.Αστακός, 11.Αγγελόκαστρο, 12.Σταμνά, 13.Λεσίσι, 14.Κατοχή, 15.Αιτωλικό,

16.Μεσολόγγι, 17.Ευηνοχώρι, 18.Ναύπακτος, 19.Κάτω μακρινού, 20.Γαβαλού,
21.Ματαράγκα, 22.Ζεγγαράκι, 23.Παναιτώλιο, 24.Καινούργιο, 25.Παραβόλα,
26.Θέρμο



Η παρουσίαση της βέλτιστης διαδρομής σε χάρτη, έγινε με την βοήθεια του συστήματος γεωγραφικών πληροφοριών ArcGis.

5.1 Πρόβλημα Δρομολόγησης Στόλου Οχημάτων

Ένα από τα σημαντικότερα προβλήματα δρομολόγησης που αναφέρονται στην επιχειρησιακή έρευνα είναι η δρομολόγηση στόλου οχημάτων, VRP (Vehicle Routine Problem), που αναφέρονται σε μία κατηγορία προβλημάτων προσδιορισμού διαδρομών κάθε οχήματος. Το πρόβλημα αναφέρεται σε ένα δοσμένο στόλο οχημάτων που έχουν αφετηρία μία ή περισσότερες αποθήκες (depot), με σκοπό την διανομή των προϊόντων σε διάφορες τοποθεσίες (nodes). Ο στόχος του VRP είναι η εξυπηρέτηση των σημείων ζήτηση με το ελάχιστο δυνατό «κόστος» συνολικής διαδρομής του στόλου οχημάτων.

Στο συγκεκριμένο πρόβλημα, τα οχήματα διακινούνται μεταξύ των πελατών σε προκαθορισμένες γνώστες τοποθεσίες των πελατών της επιχείρησης. Η διακίνηση γίνεται μέσω ενός πραγματικού δικτύου και όχι με ευκλείδειες αποστάσεις μεταξύ των σημείων. Στο πραγματικό αυτό δίκτυο που υλοποιείται μέσω του “google.maps”, η κυκλοφοριακή κίνηση μπορεί να επηρεάσει σημαντικά τις αποφάσεις που σχετίζονται με το πρόβλημα. Η επίλυση του δεν γίνεται με δυναμική προσέγγιση αλλά με την λήψη των δεδομένων, χρόνου διαδρομής και χιλιομετρική απόσταση μεταξύ των τοποθεσιών (nodes), μία τυχαία στιγμή μέσα στο ωράριο της διανομής. Ωστόσο είναι δυνατή και η δυναμική επίλυση του προβλήματος, με up to date δεδομένα από την “google distance matrix API”. Έτσι το πρόβλημα δρομολόγησης VRP, είναι ένα σύνθετο πρόβλημα προγραμματισμού, που αντικειμενικοί στόχοι του προβλήματος είναι η ελαχιστοποίηση του λειτουργικού κόστους και του προγραμματισμού της εργασίας των πληρωμάτων τους. Η επίτευξη αυτών των στόχων έχει σαν αποτέλεσμα, τον προσδιορισμό του ακριβούς αριθμού εμπλεκόμενων οχημάτων, τον προσδιορισμό βέλτιστων διαδρομών και τον προσδιορισμό του συνολικού κόστους του στόλου. Σε πραγματικές συνθήκες, συνήθως είναι γνωστό το πλήθος οχημάτων της επιχείρησης και γίνεται προσπάθεια για την εύρεση της βέλτιστης χρήση αυτών. Η ίδια παραδοχή γίνεται και στη παρακάτω παράδειγμα επίλυσης του VRP, σε επιχείρηση.

*Παράδειγμα επίλυσης του προβλήματος δρομολόγησης στόλου οχημάτων σε πραγματικά δεδομένα στην εταιρία διανομής φαρμάκων, **Pharmanet**.*

Σε αυτή την ενότητα θα παρουσιαστεί ένα παράδειγμα που δείχνει πώς επιλύετε το πρόβλημα δρομολόγησης στόλου οχημάτων (VRP) για τις τοποθεσίες που εμφανίζονται στον παρακάτω χάρτη.



Οι τοποθεσίες αντιστοιχούν στους πελάτες (κόμβους), και ο στόλος οχημάτων απαρτίζεται από δύο όχηματα που θα πρέπει να επισκεφτούν όλους τους πελάτες χρησιμοποιώντας την ελάχιστη συνολική απόσταση, ξεκινώντας από την αποθήκη (Depot).

5.1 Παρουσίαση του αλγόριθμου

Παρακάτω παρουσιάζεται η δομή του προγράμματος που χρησιμοποιήθηκε σε μορφή ψευδοκώδικα. Αναλυτικότερα ο κώδικας παρουσιάζεται στις επόμενες ενότητες.

```

1  Παράσταση Αλγορίθμου
2  Ψευδοκώδικας
3
4  Αρχή
5  ##εισαγωγή λειτουργικής μονάδας OR-Tools
6
7  #ορισμός συνάρτησης αρχικοποίησης δεδομένων του προβλήματος
8
9  #ορισμός συνάρτησης εκτύπωσης την λύσης
10
11  ορισμός main():
12
13      (1) αρχικοποίηση δεδομένων
14
15      (2) δημιουργία μοντέλου διαχείρισης ευρετηρίου δρομολόγησης
16          //εσωτερικοί δείκτες μεταβλητών που χρησιμοποιεί
17          //ο διακομιστής επίλυσης του προβλήματος
18
19      (3) δημιουργία του μοντέλου δρομολόγησης
20
21      #ορισμός συνάρτησης επιστροφής
22          //αντιστοίχιση των αποστάσεων των αρχικοποιημένων δεδομένων
23          //με τον διακομιστή επίλυσης
24
25      (4) ορισμός κόστους διαδρομής
26
27      (5) προσθήκη περιορισμού απόστασης
28
29      (6) καθορισμός ευρετικού αλγορίθμου
30          //για την εύρεση μιας πρώτης λύσης
31
32      (7) επίλυση του προβλήματος
33
34      (8) εκτύπωση της λύσης
35
36  Τέλος

```

Οι παρακάτω ενότητες παρουσιάζουν προγράμματα σε Python που λύνουν το VRP χρησιμοποιώντας OR-Tools

5.2 Δημιουργία δεδομένων

Ο παρακάτω κώδικας δημιουργεί τα δεδομένα (data) για το πρόβλημα με τον ίδιο τρόπο όπως και στο πρόβλημα πλανόδιου πωλητή (TSP). Η διαφορά είναι ότι τώρα για να δημιουργηθεί το πραγματικό δίκτυο, εισάγονται οι πραγματικές αποστάσεις μεταξύ δύο τοποθεσιών χρησιμοποιώντας το οδικό δίκτυο του google/maps.

```
def create_data_model():
    """Stores the data for the problem."""
    data = {}
    data['distance_matrix'] = [
        [0, 88, 78, 48, 42, 42, 63, 19, 7, 47, 52, 19, 28, 37, 38, 27, 37, 46, 78, 33, 22, 17, 11, 7, 9, 12, 30],
        [88, 0, 10, 52, 46, 43, 37, 78, 81, 62, 68, 91, 91, 90, 95, 103, 118, 120, 152, 129, 118, 112, 107, 93, 96, 98, 117],
        [78, 10, 0, 42, 36, 33, 35, 68, 71, 60, 65, 80, 81, 93, 97, 92, 108, 110, 142, 118, 108, 102, 97, 83, 86, 88, 107],
        [48, 52, 42, 0, 6, 25, 58, 43, 47, 57, 62, 56, 56, 68, 72, 68, 82, 85, 117, 93, 82, 77, 72, 59, 61, 64, 82],
        [42, 46, 36, 6, 0, 19, 52, 31, 35, 44, 50, 44, 45, 57, 61, 56, 71, 74, 106, 82, 71, 66, 60, 47, 50, 52, 70],
        [42, 43, 33, 25, 19, 0, 33, 32, 35, 34, 39, 45, 45, 57, 62, 57, 72, 74, 107, 83, 72, 67, 61, 48, 50, 53, 71],
        [63, 37, 35, 58, 52, 33, 0, 53, 57, 27, 32, 67, 70, 55, 61, 70, 93, 95, 127, 104, 93, 87, 82, 69, 72, 74, 92],
        [19, 78, 68, 43, 31, 32, 53, 0, 12, 36, 42, 24, 37, 49, 53, 49, 64, 66, 98, 50, 40, 34, 28, 24, 26, 29, 47],
        [11, 107, 97, 72, 60, 61, 82, 28, 17, 52, 48, 13, 19, 27, 27, 18, 27, 36, 66, 21, 10, 5, 0, 8, 13, 16, 34],
        [7, 93, 83, 59, 47, 48, 69, 24, 12, 52, 55, 19, 25, 34, 34, 24, 34, 43, 73, 30, 19, 14, 8, 0, 5, 8, 25],
        [9, 96, 86, 61, 50, 50, 72, 26, 14, 54, 60, 24, 30, 39, 39, 29, 38, 47, 78, 35, 24, 19, 13, 5, 0, 3, 20],
        [12, 98, 88, 64, 52, 53, 74, 29, 17, 57, 63, 27, 33, 42, 42, 32, 41, 50, 80, 37, 26, 22, 16, 8, 3, 0, 18],
        [30, 117, 107, 82, 70, 71, 92, 47, 35, 75, 80, 45, 50, 60, 60, 50, 60, 68, 42, 15, 25, 31, 34, 25, 20, 18, 0],
    ]
    data['num_vehicles'] = 2
    data['depot'] = 0
    return data
```

Η συνάρτηση περιλαμβάνει την δημιουργία:

- distance_matrix : Ακολουθία αποστάσεων των μεταξύ προορισμών σε μέτρα
- num_vehicles : Αριθμός των οχημάτων που κάνουν διανομή ,περισσότερα από ένα όχηματα
- depot: ο δείκτης της αποθήκης, δηλαδή η τοποθεσία που θα αποτελεί την αφετηρία και τον τερματισμό της διαδρομής του στόλου οχημάτων

Ο πίνακας αποστάσεων ('distance_matrix') είναι ένας πίνακας του οποίου η καταχώρηση i, j είναι η απόσταση από το σημείο i έως το σημείο j σε μέτρα, όπου οι δείκτες πίνακα αντιστοιχούν στις θέσεις με την ακόλουθη σειρά:

0. Αποθήκη Pharmanet, 1.Αγ.Νικόλαος, 2.Βόνιτσα, 3.Σαρδηνία, 4.Αμφιλοχία, 5.Κατούνα, 6.Μύτικας, 7.Λεπενού, 8.Νεάπολη, 9.Καραϊσκάκι, 10.Αστακός, 11.Αγγελόκαστρο, 12.Σταμνά, 13.Λεσίνι, 14.Κατοχή, 15.Αιτωλικό, 16.Μεσσολόγγι, 17.Ευηνοχώρι, 18.Ναύπακτος, 19.Κάτω μακρινού, 20.Γαβαλού, 21.Ματαράγκα, 22.Ζευγαράκι, 23.Παναιτώλιο, 24.Καινούργιο, 25.Παραβόλα, 26.Θέρμο

Σημείωση: Η σειρά των θέσεων στον πίνακα αποστάσεων είναι αυθαίρετη και δεν σχετίζεται με τη σειρά των θέσεων σε οποιαδήποτε λύση του VRP.

Συντεταγμένες των Τοποθεσιών

Για να καθοριστεί το πρόβλημα χρησιμοποιήθηκαν οι αποστάσεις όπως αυτές μετρήθηκαν από το οδικό δίκτυο στο google.maps που αντιστοιχούν στους πελάτες της επιχείρησης και έχουν εκχωρηθεί στις θέσεις που εμφανίζονται στο διάγραμμα της πόλης. Ο πίνακας των αποστάσεων υλοποιήθηκε στο υπολογιστικό φύλλο του excel και παρουσιάζεται στο τέλος του κεφαλαίου στην ολοκληρωμένη μορφή του.

Είναι άξιο να επισημανθεί ότι οι συντεταγμένες των τοποθεσιών δεν περιλαμβάνονται στα δεδομένα του προβλήματος. Το μόνο που χρειάζεται για την επίλυση του προβλήματος είναι ο πίνακας αποστάσεων. Οι συντεταγμένες είναι χρήσιμες μόνο στην οπτική αναπαράσταση, του προβλήματος και της επίλυσης στο διάγραμμα της περιοχής.

5.3 Δημιουργία Μοντέλου Δρομολόγησης

Ο παρακάτω κώδικας στο κύριο τμήμα του αλγόριθμου δημιουργεί έναν δείκτη (manager) και το μοντέλο δρομολόγησης (routing), ακριβώς με τον ίδιο τρόπο που γίνεται και στο πρόβλημα του πλανόδιου πωλητή (TSP).

```
41 # Instantiate the data problem.
42 data = create_data_model()
43 # Create the routing index manager.
44 manager = pywrapcp.RoutingIndexManager(len(data['distance_matrix']),
45                                     data['num_vehicles'], data['depot'])
46 # Create Routing Model.
47 routing = pywrapcp.RoutingModel(manager)
```

5.4 Δημιουργία Συνάρτησης Επιστροφής

Όπως στο παράδειγμα TSP, η συνάρτηση δημιουργεί την επιστροφή απόστασης μεταξύ δύο τοποθεσιών και την τοποθετεί στον διακομιστή επίλυσης. Ακόμη ορίζει το κόστος κάθε επιμέρους διαδρομής, το οποίο καθορίζει το συνολικό κόστος δρομολόγησης, να είναι ίσο με την απόσταση της διαδρομής. Άλλοι παράγοντες που επηρεάζουν το κόστος διαδρομής είναι η χωρητικότητα και τα χρονικά παράθυρα, που θα παρουσιασθούν πιο αναλυτικά στις επόμενες ενότητες.

```

# Create and register a transit callback.
def distance_callback(from_index, to_index):
    """Returns the distance between the two nodes."""
    # Convert from routing variable Index to distance matrix NodeIndex.
    from_node = manager.IndexToNode(from_index)
    to_node = manager.IndexToNode(to_index)
    return data['distance_matrix'][from_node][to_node]

transit_callback_index = routing.RegisterTransitCallback(distance_callback)

# Define cost of each arc.
routing.SetArcCostEvaluatorOfAllVehicles(transit_callback_index)

```

5.5 Προσθήκη Διάστασης Απόστασης

Για την επίλυση του VRP, χρειάζεται να δημιουργηθεί η διάσταση της απόστασης του κάθε οχήματος. Η προσθήκη αυτής της διάστασης είναι η βασική διαφορά που μετασχηματίζει το πρόβλημα του πλανόδιου πωλητή, που αναφέρεται σε ένα όχημα, με το πρόβλημα της δρομολόγησης του στόλου οχημάτων, που αναφέρεται σε περισσότερα οχήματα. Με την εισαγωγή αυτής της διάστασης γίνεται η παρακολούθηση των ποσοτήτων που συσσωρεύονται στη διαδρομή ενός οχήματος. Έτσι καθορίζονται τα χαρακτηριστικά κάθε οχήματος, όπως η μέγιστη απόσταση που μπορεί να διανύσει, η χωρητικότητα, τυχόν παράμετροι που καθορίζουν την ώρα αναμονής ή/και χρονικά παράθυρα. Στο συγκεκριμένο ορίζεται μόνο η μέγιστη απόσταση και οι υπόλοιποι παράμετροι απενεργοποιούνται.

Ο ακόλουθος κώδικας δημιουργεί την διάσταση της απόστασης χρησιμοποιώντας την μέθοδο του διακομιστή επίλυσης `AddDimension`. Η μεταβλητή «`transit_callback_index`» είναι ο δείκτης αντιστοίχισης της συνάρτησης επιστροφής «`distance_callback`», και εισάγεται ως παράμετρος στην μέθοδο `AddDimension`.

```

# Create and register a transit callback.
def distance_callback(from_index, to_index):
    """Returns the distance between the two nodes."""
    # Convert from routing variable Index to distance matrix NodeIndex.
    from_node = manager.IndexToNode(from_index)
    to_node = manager.IndexToNode(to_index)
    return data['distance_matrix'][from_node][to_node]

transit_callback_index = routing.RegisterTransitCallback(distance_callback)

# Define cost of each arc.
routing.SetArcCostEvaluatorOfAllVehicles(transit_callback_index)

```

Η μέθοδος «SetGlobalSpanCostCoefficient» ορίζει έναν μεγάλο συντελεστή της τάξης του «100», σαν μέγιστο αριθμό των αποστάσεων στη διαδρομή ενός οχήματος. Αυτό δεν καθορίζει μόνο το συνολικό εύρος των διαδρομών αλλά ορίζεται και ως καθοριστικός παράγοντας της συνάρτησης ώστε να ελαχιστοποιείται η απόσταση της μεγαλύτερης διαδρομής.

5.6 Ανάλυση Αντικειμένου Διάστασης (Dimensions)

Ο διακομιστής επίλυσης της δρομολόγησης χρησιμοποιεί ένα αντικείμενο που καλείται «Dimension», για να παρακολουθεί τις ποσότητες που συσσωρεύονται κατά την διάρκεια που υλοποιείται ένα δρομολόγιο, όπως ο χρόνος διαδρομής, ή αν το όχημα κάνει διανομή και παραλαβή, το συνολικό βάρος που μεταφέρει. Αν το πρόβλημα δρομολόγησης χρησιμοποιεί κάποια τέτοια είτε σαν περιορισμός είτε στην συνάρτηση επίλυσης, χρειάζεται να οριστούν σαν επιπρόσθετη διάσταση. Σε αυτό το σημείο θα αναλυθεί η διαδικασία ορισμού και χρήσης της διάστασης (dimension).

Ορισμός μεθόδου

Το ακόλουθο παράδειγμα ορίζει την προσθήκης διάστασης, χρησιμοποιώντας την μέθοδο AddDimension. Επίσης ορίζεται και η μέγιστη χιλιομετρική απόσταση που μπορούν να διανύσουν τα οχήματα διανομής σε 600km.

```
# Add Distance constraint.
dimension_name = 'Distance'
routing.AddDimension(
    transit_callback_index,
    0, # no slack
    600, # vehicle maximum travel distance
    True, # start cumul to zero
    dimension_name)
distance_dimension = routing.GetDimensionOrDie(dimension_name)
distance_dimension.SetGlobalSpanCostCoefficient(100)
```

Η μέθοδος AddDimension, έχει τις εξής παραμέτρους:

- `callback_index`: Ο δείκτης που αντιστοιχεί στην συνάρτηση επιστροφής «distance_callback». Ο δείκτης αυτός που είναι η εσωτερική αναφορά του διακομιστή επίλυσης (solver), δημιουργείτε από μεθόδους όπως «RegisterTransitCallback» ή «RegisterUnitaryTransitCallback».
- `slack_max`: Μεταβλητή που καθορίζει τον χρόνο αναμονής στις τοποθεσίες που γίνεται η διανομή. Εάν το πρόγραμμα δεν περιλαμβάνει χρόνο αναμονής, τότε, `slack_max = 0`. Στην ενότητα με τους χρονικούς περιορισμούς θα παρουσιαστεί με περισσότερο λεπτομέρεια η συμπλήρωση της μεταβλητής αδράνειας.
- `capacity`: Μεταβλητή χωρητικότητας ανά διαδρομή. Η χρήση του θα γίνει πιο εμφανής στην επόμενη ενότητα που θα αναλυθεί το πρόβλημα δρομολόγησης στόλου οχημάτων με περιορισμό χωρητικότητας (CVRP). Εάν δεν υπάρχει περιορισμός χωρητικότητας όπως στο VRP, μπορεί να οριστεί η μεταβλητή constraints αρκετά μεγάλη ώστε να μην δημιουργεί πρόβλημα περιορισμού.
- `fix_start_cumulative_to_zero`: Είναι μια μεταβλητή τύπου Boolean. Εάν ισχύει «True» τότε η ποσότητα, η οποία αναφέρεται σε χρόνο τοποθετείται στο 0. Στις περισσότερες περιπτώσεις ισχύει αυτό, ωστόσο σε προβλήματα δρομολόγησης με χρονικά παράθυρα μερικά οχήματα ίσως χρειαστεί να ξεκινήσουν μετά την ώρα «0». Αυτό συμβαίνει διότι δημιουργούνται περιορισμοί από χρονικά παράθυρα που θέτουν οι πελάτες. Γι' αυτά τα προβλήματα η μεταβλητή πρέπει να αλλάξει σε «False».
- `dimension_name`: Συμβολοσειρά για το όνομα της διάστασης, η οποία θα χρησιμοποιηθεί για την πρόσβαση των μεταβλητών σε άλλο σημείο του προγράμματος.

Ο διακομιστής επίλυσης διαθέτει και άλλες μεθόδους προσθήκης, για να γίνεται καταλληλότερη προσαρμογή ανάλογα με τις απαιτήσεις του κάθε προβλήματος μερικά παραδείγματα μεθόδων είναι:

- AddDimensionDependentDimensionWithVehicleCapacity
- AddPickupAndDelivery

5.7 Συνάρτηση Εκτύπωσης

Η συνάρτηση εμφανίζει την λύση που επιστράφηκε από τον διακομιστή επίλυσης. Εμφανίζει τις διαδρομές και την συνολική διανυόμενη απόσταση των διαδρομών, που δίνεται από έναν επαναληπτικό βρόχο όπως φαίνεται παρακάτω.

```
def print_solution(data, manager, routing, solution):
    """Prints solution on console."""
    max_route_distance = 0
    for vehicle_id in range(data['num_vehicles']):
        index = routing.Start(vehicle_id)
        plan_output = 'Route for vehicle {}:\n'.format(vehicle_id)
        route_distance = 0
        while not routing.IsEnd(index):
            plan_output += ' {} -> '.format(manager.IndexToNode(index))
            previous_index = index
            index = solution.Value(routing.NextVar(index))
            route_distance += routing.GetArcCostForVehicle(
                previous_index, index, vehicle_id)
        plan_output += ' {}\n'.format(manager.IndexToNode(index))
        plan_output += 'Distance of the route: {}Km\n'.format(route_distance)
        print(plan_output)
        max_route_distance = max(route_distance, max_route_distance)
    print('Maximum of the route distances: {}Km'.format(max_route_distance))
```

Εναλλακτικά, μπορεί αρχικά να αποθηκευτεί η λύση σε μία ακολουθία και στην συνέχεια να εκτυπωθεί.

5.8 Κύρια Συνάρτηση

Το μεγαλύτερο μέρος του κώδικα στην κύρια λειτουργία για το πρόγραμμα VRP είναι το ίδιο με το προηγούμενο παράδειγμα TSP. Το καινούργιο κομμάτι είναι η προσθήκη της διάστασης απόστασης, που περιγράφηκε παραπάνω. Στην συνέχεια παρουσιάζεται η δημιουργία του μοντέλου δρομολόγησης και του διαχειριστή των δεικτών του, στην συνάρτηση την “main”.

```
def main():
    """Solve the CVRP problem."""
    # Instantiate the data problem.
    data = create_data_model()

    # Create the routing index manager.
    manager = pywrapcp.RoutingIndexManager(len(data['distance_matrix']),
                                           data['num_vehicles'], data['depot'])

    # Create Routing Model.
    routing = pywrapcp.RoutingModel(manager)
```

5.9 Παρουσίαση Λύσης σε Χάρτη

Ο αλγόριθμος εμφανίζει την παρακάτω λύση,

```
Route for vehicle 0:
0 -> 8 -> 5 -> 2 -> 1 -> 6 -> 9 -> 10 -> 23 -> 25 -> 26 -> 24 -> 0
Distance of the route: 264Km

Route for vehicle 1:
0 -> 3 -> 4 -> 7 -> 11 -> 12 -> 13 -> 14 -> 15 -> 16 -> 17 -> 18 -> 19 -> 20 -> 21 -> 22 -> 0
Distance of the route: 264Km

Maximum of the route distances: 264Km

Process finished with exit code 0
```

Υπενθυμίζεται ότι :

0. Αποθήκη Pharmanet, 1.Αγ.Νικόλαος, 2.Βόνιτσα, 3.Σαρδηνία, 4.Αμφιλοχία, 5.Κατούνα, 6.Μύτικας, 7.Λεπενού, 8.Νεάπολη, 9.Καραϊσκάκι, 10.Αστακός, 11.Αγγελόκαστρο, 12.Σταμνά, 13.Λεσίνοι, 14.Κατοχή, 15.Αιτωλικό, 16.Μεσσολόγγι, 17.Ευηνοχώρι, 18.Ναύπακτος, 19.Κάτω μακρινού, 20.Γαβαλού, 21.Ματαράγκα, 22.Ζευγαράκι, 23.Παναιτόλιο, 24.Καινούργιο, 25.Παραβόλα, 26.Θέρμο

Οι τοποθεσίες των πελατών στις διαδρομές συμβολίζονται σύμφωνα με τους δείκτες τους, όπως αυτοί εμφανίζονται στην λίστα τοποθεσιών. Όλες οι διαδρομές ξεκινούν και τερματίζουν στην τοποθεσία της αποθήκης (Pharmanet).

Το διάγραμμα στην συνέχεια παρουσιάζει την λύση του προβλήματος, δηλαδή την εμφάνιση των διαδρομών και των τοποθεσιών όπως έχουν προκύψει από την λύση στις αντίστοιχες συντεταγμένες x-y.



Η παρουσίαση των διαδρομών σε χάρτη, έγινε με την βοήθεια του συστήματος γεωγραφικών πληροφοριών ArcGis.

6.1 Πρόβλημα Δρομολόγησης Στόλου Οχημάτων Με Περιορισμό Χωρητικότητας

Το πρόβλημα δρομολόγησης στόλου οχημάτων με περιορισμό χωρητικότητας, CVRP – Capacitated Vehicle Routine Problem, ανήκει στην κατηγορία προβλημάτων VRP στο οποίο τα οχήματα έχουν περιορισμένη μεταφορική ικανότητα. Τα αντικείμενα που μεταφέρουν έχουν μία ποσότητα, όπως βάρος ή όγκο, και τα οχήματα μέγιστη χωρητικότητα σε βάρος ή όγκο αντίστοιχα. Το πρόβλημα είναι να γίνεται η διανομή των αντικειμένων χωρίς να γίνεται υπέρβαση της χωρητικότητας των οχημάτων. Ο περιορισμός της χωρητικότητας, αποτελεί προβληματισμό και σε άλλες κατηγορίες του VRP, όπως το πρόβλημα δρομολόγησης στόλου οχημάτων με χρονικά παράθυρα (VRPTW) και δρομολόγησης στόλου οχημάτων διανομής και παραλαβής (Pickup & Delivery VRP). Ειδικά στην περίπτωση της διανομής και παραλαβής, ο περιορισμός χωρητικότητας είναι απαραίτητος.

Στο συγκεκριμένο παράδειγμα επίλυσης γίνεται η παραδοχή ότι όλα τα αντικείμενα διανέμονται. Έτσι ο περιορισμός μη υπέρβασης της χωρητικότητας εφαρμόζεται όταν τα οχήματα ξεκινούν από την αποθήκη (depot 0). Ωστόσο, οι περιορισμοί χωρητικότητας εφαρμόζονται με τον ίδιο τρόπο και στις δύο περιπτώσεις μέσα στο πρόγραμμα.

Παράδειγμα επίλυσης του προβλήματος δρομολόγησης στόλου οχημάτων σε πραγματικά δεδομένα στην εταιρία διανομής φαρμάκων, Pharmanet.

Το παράδειγμα επεκτείνει το πρόβλημα της προηγούμενης ενότητας VRP, προσθέτοντας τις ακόλουθες απαιτήσεις. Σε κάθε τοποθεσία/πελάτη της επιχείρησης, υπάρχει μία ζητούμενη ποσότητα που αντιστοιχεί στη ποσότητα που θα φορτώσει το όχημα από την αποθήκη κατά την εκκίνηση του. Η ποσότητα που δεν μπορεί να υπερβεί το όχημα είναι 30 τεμάχια και τα οχήματα για τη διανομή είναι δύο. Στο διάγραμμα του χάρτη, παρουσιάζονται οι τοποθεσίες στις οποίες θα γίνει η διανομή και η αποθήκη/αφετηρία (Pharmanet). Οι απαιτήσεις εμφανίζονται μέσα στην παρένθεση σε κάθε τοποθεσία.



Το πρόβλημα είναι να βρεθεί μια κατανομή των διαδρομών σε οχήματα με την μικρότερη συνολική απόσταση, και το συνολικό ποσό που μεταφέρει ένα όχημα να μην υπερβαίνει ποτέ την χωρητικότητά του. Η συνολική ζήτηση των πελατών είναι 50 τεμάχια, ενώ η συνολική χωρητικότητα των οχημάτων είναι 60 τεμάχια. Σε περίπτωση που τα ζητούμενα τεμάχια είναι περισσότερα από την χωρητικότητα των οχημάτων απαιτείται διαφορετική προσέγγιση του προβλήματος.

6.2 Παρουσίαση του αλγόριθμου

Παρακάτω παρουσιάζεται η δομή του προγράμματος που χρησιμοποιήθηκε σε μορφή ψευδοκώδικα. Αναλυτικότερα ο κώδικας παρουσιάζεται στις επόμενες ενότητες.

```

1  Παράσταση Αλγορίθμου
2  Ψευδοκώδικας
3
4  Αρχή
5  ##εισαγωγή λειτουργικής μονάδας OR-Tools
6
7  #ορισμός συνάρτησης αρχικοποίησης δεδομένων του προβλήματος
8
9  #ορισμός συνάρτησης εκτύπωσης την λύσης
10
11  ορισμός main():
12
13      (1) αρχικοποίηση δεδομένων
14
15      (2) δημιουργία μοντέλου διαχείρισης ευρετηρίου δρομολόγησης
16          //εσωτερικοί δείκτες μεταβλητών που χρησιμοποιεί
17          //ο διακομιστής επίλυσης του προβλήματος
18
19      (3) δημιουργία του μοντέλου δρομολόγησης
20
21  #ορισμός συνάρτησης επιστροφής
22      //αντιστοίχιση των αποστάσεων των αρχικοποιημένων δεδομένων
23      //με τον διακομιστή επίλυσης
24
25      (4) ορισμός κόστους διαδρομής
26
27      (5) προσθήκη περιορισμού χωρητικότητας
28
29      (6) καθορισμός ευρετικού αλγορίθμου
30          //για την εύρεση μιας πρώτης λύσης
31
32      (7) επίλυση του προβλήματος
33
34      (8) εκτύπωση της λύσης
35
36  Τέλος

```

Οι παρακάτω ενότητες παρουσιάζουν προγράμματα σε Python που λύνουν το CVRP χρησιμοποιώντας OR-Tools

6.3 Δημιουργία δεδομένων

Ο παρακάτω κώδικας δημιουργεί τα δεδομένα (data) για το πρόβλημα με τον ίδιο τρόπο όπως και στο πρόβλημα δρομολόγησης στόλου οχημάτων (VRP). Στα δεδομένα προστίθενται επίσης δυο νέες ποσότητες, ζήτηση και χωρητικότητα οχημάτων.

```
def create_data_model():
    """Stores the data for the problem."""
    data = {}
    data['distance_matrix'] = [
        [0, 88, 78, 48, 42, 42, 63, 19, 7, 47, 52, 19, 28, 37, 38, 27, 37, 46,
         88, 0, 10, 52, 46, 43, 37, 78, 81, 62, 68, 91, 91, 90, 95, 103, 118, 120,
         78, 10, 0, 42, 36, 33, 35, 68, 71, 60, 65, 80, 81, 93, 97, 92, 108, 110,
         48, 52, 42, 0, 6, 25, 58, 43, 47, 57, 62, 56, 56, 68, 72, 68, 82, 85,
         7, 93, 83, 59, 47, 48, 69, 24, 12, 52, 55, 19, 25, 34, 34, 24, 34, 43,
         9, 96, 86, 61, 50, 50, 72, 26, 14, 54, 60, 24, 30, 39, 39, 29, 38, 47,
         12, 98, 88, 64, 52, 53, 74, 29, 17, 57, 63, 27, 33, 42, 42, 32, 41, 50,
         30, 117, 107, 82, 70, 71, 92, 47, 35, 75, 80, 45, 50, 60, 60, 50, 60, 68,
    ]
    data['demands'] = [0, 1, 2, 1, 3, 1, 2, 2, 1, 1, 4, 2, 1, 1, 2, 3, 5, 1, 4, 1, 1, 1, 1, 2, 3, 1, 3]
    data['vehicle_capacities'] = [30, 30]
    data['num_vehicles'] = 2
    data['depot'] = 0
    return data
```

Τα καινούργια αντικείμενα είναι:

- Demands : Κάθε τοποθεσία κάνει μια παραγγελία η οποία αντιστοιχεί σε μία συγκεκριμένη ποσότητα που θα πρέπει να παραλάβει το όχημα από την αποθήκη.
- Vehicle Capacities : Κάθε όχημα έχει μία χωρητικότητα, η μέγιστη χωρητικότητα για κάθε όχημα μπορεί να διαφέρει. Καθώς το όχημα διανύει την διαδρομή δεν μπορεί ούτε τότε να υπερβεί αυτή την τιμή.

Σημείωση: Η σειρά των θέσεων στον πίνακα αποστάσεων είναι αυθαίρετη και δεν σχετίζεται με τη σειρά των θέσεων σε οποιαδήποτε λύση του CVRP. Οι δείκτες πίνακα αντιστοιχούν στις θέσεις με την ακόλουθη σειρά και την ζήτηση της κάθε τοποθεσίας:

0. Αποθήκη *Pharmanet*, 1.Αγ.Νικόλαος(1), 2.Βόνιτσα(2), 3.Σαρδηνία(1), 4.Αμφιλοχία(3), 5.Κατούνα(1), 6.Μύτικας(2), 7.Λεπενού(1), 8.Νεάπολη(1), 9.Καραϊσκάκι(1), 10.Αστακός(4), 11.Αγγελόκαστρο(2), 12.Σταμνά(1), 13.Λεσίνι(1), 14.Κατοχή(2), 15.Αιτωλικό(3), 16.Μεσσολόγγι(5), 17.Ευηνοχώρι(1), 18.Ναόπακτος(4), 19.Κάτω μακρινού(1), 20.Γαβαλού(1), 21.Ματαράγκα(1),

22. Ζευγαράκι(1), 23. Παναιτώλιο(2), 24. Καινούργιο(3), 25. Παραβόλα(1),
26. Θέρμο(3)

6.4 Δημιουργία Μοντέλου Δρομολόγησης

Η κύρια δομή του αλγόριθμου δεν διαφέρει από τον αλγόριθμο που χρησιμοποιήθηκε στην προηγούμενη ενότητα για το πρόβλημα του VRP, τόσο στην δομή της «main», όσο και στην συνάρτηση επιστροφής «distance_callback». Η διαφορά είναι η προσθήκη συνάρτησης επιστροφής ζήτησης και ο περιορισμός χωρητικότητας. Επίσης γίνεται μια προσθήκη και στην συνάρτηση εκτύπωσης ώστε να εμφανίζει και το αθροιστικό φορτίο κάθε οχήματος.

6.5 Δημιουργία Συνάρτησης Επιστροφής Ζήτησης

Επιπρόσθετα της συνάρτησης επιστροφής, που επιστρέφει την απόσταση δύο τοποθεσιών, ο διακομιστής επίλυσης για το πρόβλημα του CVRP απαιτεί την ύπαρξη και μιας συνάρτησης επιστροφής της ζήτησης. Η συνάρτηση επιστρέφει την ζητούμενη ποσότητα από κάθε τοποθεσία. Ακόμη ο διακομιστής επίλυσης, χρειάζεται μια διάσταση (dimension) για τον περιορισμό της χωρητικότητας. Η διάσταση είναι της ίδιας λογικής που χρησιμοποιήθηκε στην προηγούμενη ενότητα για την εισαγωγή της διάστασης των οχημάτων.

```
# Add Capacity constraint.
def demand_callback(from_index):
    """Returns the demand of the node."""
    # Convert from routing variable Index to demands NodeIndex.
    from_node = manager.IndexToNode(from_index)
    return data['demands'][from_node]

demand_callback_index = routing.RegisterUnaryTransitCallback(
    demand_callback)
routing.AddDimensionWithVehicleCapacity(
    demand_callback_index,
    0, # null capacity slack
    data['vehicle_capacities'], # vehicle maximum capacities
    True, # start cumul to zero
    'Capacity')
```

Το πρόγραμμα CVRP, δημιουργεί με μικρή διαφοροποίηση από την διάσταση απόστασης του VRP, την διάσταση χωρητικότητας. Χρησιμοποιεί την μέθοδο,

AddDimansionWithVehicleCapacity. Η μέθοδος αυτή χρησιμοποιεί μία ακολουθία από ποσότητες που αντιστοιχούν στην μέγιστη δυνατή χωρητικότητα του κάθε οχήματος. Σε αντίθεση, η μέθοδος AddDimension, παίρνει μία ίδια τιμή χωρητικότητας και σχηματίζεται η υπόθεση ότι έχουν όλα την χωρητικότητα.

6.6 Ποινές & Απόσυρση Δρομολόγιου

Σε αυτό το κομμάτι θα γίνει αναφορά για το πρόβλημα δρομολόγησης που δεν έχει εφικτή λύση, λόγω περιορισμών. Παράδειγμα μη εφικτής λύσης είναι, αν δοθεί ένα VRP με περιορισμούς χωρητικότητας, στο οποίο η συνολική ζήτηση σε όλες τις τοποθεσίες υπερβαίνει τη συνολική χωρητικότητα των οχημάτων, καμία λύση δεν είναι δυνατή. Σε τέτοιες περιπτώσεις, τα οχήματα πρέπει να αποσύρουν την διανομή σε ορισμένες τοποθεσίες. Το πρόβλημα είναι στο πως θα αποφασιστεί ποιες επισκέψεις θα αποσυρθούν.

Για την επίλυση του προβλήματος, χρειάζεται να εισαχθεί μία τιμή «ποινής» σε όλες τις τοποθεσίες. Όποτε αποσύρεται μία τοποθεσία, η ποινή εισάγεται στο συνολικό κόστος της διαδρομής. Έως τώρα το συνολικό κόστος διαδρομής ισούταν με την απόσταση της διαδρομής. Ο διακομιστής επίλυσης, με αυτό τον τρόπο βρίσκει την βέλτιστη δρομολόγηση των οχημάτων, ελαχιστοποιώντας το συνολικό κόστος διαδρομής που συμπεριλαμβάνει και την ποινή απόσυρσης τοποθεσίας. Στην πραγματικότητα ο όρος της ποινής όντως υπάρχει και έχει καθοριστεί στην συμφωνία που κάνει ο πελάτης με κάθε επιχείρηση.

Υλοποίηση Περιορισμού Απόσυρσης Τοποθεσιών

Για την υλοποίηση του περιορισμού απόσυρσης, χρειάζεται να γίνει εισαγωγή της ποινής. Μπορεί να καθοριστεί ποινή διαφορετική για κάθε τοποθεσία, διαφορετικά αρχικοποιούμε ίδια ποινή για όλες τις τοποθεσίες. Ορίζοντας την ίδια ποινή σε όλες τις τοποθεσίες, τότε το βάρος για την επιλογή της τοποθεσίας που θα αποσυρθεί πέφτει στην απόσταση της διαδρομής. Οι ποινές χρησιμοποιούνται με την βοήθεια της μεθόδου AddDisjunction.

```

def demand_callback(from_index):
    """Returns the demand of the node."""
    # Convert from routing variable Index to demands NodeIndex.
    from_node = manager.IndexToNode(from_index)
    return data['demands'][from_node]

demand_callback_index = routing.RegisterUnaryTransitCallback(
    demand_callback)
routing.AddDimensionWithVehicleCapacity(
    demand_callback_index,
    0, # null capacity slack
    data['vehicle_capacities'], # vehicle maximum capacities
    True, # start cumul to zero
    'Capacity')
# Allow to drop nodes.
penalty = 1000
for node in range(1, len(data['distance_matrix'])):
    routing.AddDisjunction([manager.NodeToIndex(node)], penalty)

```

Στον παραπάνω κώδικα, η μέθοδος `AddDisjunction`, κάνει μια διάκριση που χρησιμοποιεί ο διακομιστής επίλυσης για να αποφασίσει εάν θα συμπεριλάβει μία συγκεκριμένη τοποθεσία στην λύση. Σε αυτό το πρόγραμμα, η μέθοδος προσθέτει την ίδια ποινή σε κάθε τοποθεσία, αλλά γενικά μπορεί να προστεθεί διαφορετική ποινή σε κάθε τοποθεσία.

Η μέθοδος προσθήκης του περιορισμού της απόσυρσης, τοποθετεί την ποινή σε κάθε τοποθεσία της διαδρομής. Η αρχή και το τέλος κάθε διαδρομής δεν είναι μέρος της μεθόδου οπότε δεν δίνεται και τιμή ποινής, συνήθως αυτή η περιοχή είναι αποθήκη (depot 0).

6.7 Συνάρτηση Εκτύπωσης

Η συνάρτηση εμφανίζει την λύση που επιστράφηκε από τον διακομιστή επίλυσης. Εμφανίζει τις διαδρομές την συνολική διανυόμενη απόσταση των διαδρομών και το αθροιστικό φορτίο. Αθροιστικό φορτίο είναι, η συνολική ποσότητα που μεταφέρει το κάθε όχημα κατά τις στάσεις του, στην διαδρομή του. Επίσης εμφανίζει σε τυχόν απόσυρση τοποθεσιών, τις τοποθεσίες που αποσύρθηκαν (Dropped nodes).

```

def print_solution(data, manager, routing, solution):
    """Prints solution on console."""
    total_distance = 0
    total_load = 0
    for vehicle_id in range(data['num_vehicles']):
        index = routing.Start(vehicle_id)
        plan_output = 'Route for vehicle {}:\n'.format(vehicle_id)
        route_distance = 0
        route_load = 0
        while not routing.IsEnd(index):
            node_index = manager.IndexToNode(index)
            route_load += data['demands'][node_index]
            plan_output += ' {0} Load({1}) -> '.format(node_index, route_load)
            previous_index = index
            index = solution.Value(routing.NextVar(index))
            route_distance += routing.GetArcCostForVehicle(
                previous_index, index, vehicle_id)
        plan_output += ' {0} Load({1})\n'.format(manager.IndexToNode(index),
            route_load)
        plan_output += 'Distance of the route: {}m\n'.format(route_distance)
        plan_output += 'Load of the route: {}\n'.format(route_load)
        print(plan_output)
        total_distance += route_distance
        total_load += route_load
    print('Total distance of all routes: {}m'.format(total_distance))
    print('Total load of all routes: {}'.format(total_load))

```

Εναλλακτικά, μπορεί αρχικά να αποθηκευτεί η λύση σε μία ακολουθία και στην συνέχεια να εκτυπωθεί.

6.8 Κύρια Συνάρτηση

Το μεγαλύτερο μέρος του κώδικα στην κύρια λειτουργία για το πρόγραμμα CVRP είναι το ίδιο με το προηγούμενο παράδειγμα VRP. Το κομμάτι που προστίθεται στον αλγόριθμο είναι, η συνάρτηση επιστροφής ζήτησης των τοποθεσιών και η διάσταση της χωρητικότητας.

```

def main():
    """Solve the CVRP problem."""
    # Instantiate the data problem.
    data = create_data_model()

    # Create the routing index manager.
    manager = pywrapcp.RoutingIndexManager(len(data['distance_matrix']),
        data['num_vehicles'], data['depot'])

    # Create Routing Model.
    routing = pywrapcp.RoutingModel(manager)

```


6.9 Παρουσίαση Λύσης σε Χάρτη

Ο αλγόριθμος εμφανίζει την παρακάτω λύση,

```
Route for vehicle 0:
0 Load(0) -> 22 Load(1) -> 20 Load(2) -> 21 Load(3) -> 15 Load(6) -> 16 Load(11) -> 17 Load(12) -> 18 Load(16) -> 19 Load(17) -> 26 Load(20) -> 25 Load(21) -> 24
Load(24) -> 23 Load(26) -> 0 Load(26)
Distance of the route: 183Km
Load of the route: 26

Route for vehicle 1:
0 Load(0) -> 11 Load(2) -> 12 Load(3) -> 14 Load(5) -> 13 Load(6) -> 10 Load(10) -> 9 Load(11) -> 6 Load(13) -> 1 Load(14) -> 2 Load(16) -> 5 Load(17) -> 3 Load
-> 4 Load(21) -> 7 Load(23) -> 8 Load(24) -> 8 Load(24)
Distance of the route: 265Km
Load of the route: 24

Total distance of all routes: 448Km
Total load of all routes: 50

Process finished with exit code 0
```

Υπενθυμίζεται ότι :

0. Αποθήκη *Pharmanet*, 1.Αγ.Νικόλαος(1), 2.Βόνιτσα(2), 3.Σαρδηνία(1), 4.Αμφιλοχία(3), 5.Κατούνα(1), 6.Μότικας(2), 7.Λεπενού(1), 8.Νεάπολη(1), 9.Καραϊσκάκι(1), 10.Αστακός(4), 11.Αγγελόκαστρο(2), 12.Σταμνά(1), 13.Λεσίνι(1), 14.Κατοχή(2), 15.Αιτωλικό(3), 16.Μεσσολόγγι(5), 17.Ευηνοχώρι(1), 18.Ναύπακτος(4), 19.Κάτω μακρινού(1), 20.Γαβαλού(1), 21.Ματαράγκα(1), 22.Ζευγαράκι(1), 23.Παναιτώλιο(2), 24.Καινούργιο(3), 25.Παραβόλα(1), 26.Θέρμο(3)

Σε κάθε τοποθεσία που εμφανίζεται στο διάγραμμα του χάρτη, αντιστοιχεί ο αριθμός της τοποθεσίας και το συνολικό φορτίο που μεταφέρει το όχημα μετά την αναχώρηση του από κάθε τοποθεσία

Το διάγραμμα στην συνέχεια παρουσιάζει την λύση του προβλήματος, δηλαδή την εμφάνιση των διαδρομών και των τοποθεσιών όπως έχουν προκύψει από την λύση στις αντίστοιχες συντεταγμένες x-y.



Η παρουσίαση των διαδρομών σε χάρτη, έγινε με την βοήθεια του συστήματος γεωγραφικών πληροφοριών ArcGis.

7.1 Πρόβλημα Δρομολόγησης Στόλου Οχημάτων Με Χρονικά Παράθυρα

Στις προηγούμενες ενότητες αναλύθηκε το πιο από πρόβλημα δρομολόγησης του πλανόδιου πωλητή TSP, το οποίο μετατράπηκε σε πρόβλημα δρομολόγησης στόλου οχημάτων VRP, αυξάνοντας το πλήθος των οχημάτων. Στη συνέχεια προστέθηκε και ο περιορισμός της χωρητικότητας, οπότε μετατράπηκε σε πρόβλημα με περιορισμό της χωρητικότητας CVRP. Με αυτό τον τρόπο γίνεται προσπάθεια προσέγγισης των πραγματικών συνθηκών του προβλήματος της δρομολόγησης οχημάτων σε μία επιχείρηση. Σε αυτήν την ενότητα θα προστεθεί άλλος ένας περιορισμός που συναντιέται στην πραγματικότητα.

Ο περιορισμός αυτός δημιουργείται από των καθορισμό χρονικών παραθύρων των πελατών, στα οποία είναι διαθέσιμοι και μπορεί να τους επισκεφτεί το όχημα. Τα προβλήματα αυτά είναι γνωστά ως προβλήματα δρομολόγησης οχημάτων με χρονικά περιθώρια (VRPTW).

*Παράδειγμα επίλυσης του προβλήματος δρομολόγησης στόλου οχημάτων σε πραγματικά δεδομένα στην εταιρία διανομής φαρμάκων, **Pharmanet**.*

Στην ενότητα αυτή θα γίνει επίλυση του προβλήματος δρομολόγησης στόλου οχημάτων με χρονικά παράθυρα. Εφόσον το πρόβλημα περιέχει χρονικά παράθυρα τα οποία είναι προκαθορισμένα από τους πελάτες, τα δεδομένα θα περιέχουν πίνακα χρόνων, που θα περιλαμβάνει τους χρόνους μεταξύ των τοποθεσιών. Όχι τους πίνακες αποστάσεων που χρησιμοποιήθηκαν στις προηγούμενες ενότητες.



Στόχος είναι να βρεθεί μια κατανομή των διαδρομών σε οχήματα με το μικρότερο συνολικό κόστος διαδρομής, το συνολικό ποσό που μεταφέρει ένα όχημα να μην υπερβαίνει ποτέ την χωρητικότητα του και οι τοποθεσίες να επισκέπτονται μέσα στα χρονικά παράθυρα που έχουν προτείνει.

7.2 Παρουσίαση του αλγόριθμου

Παρακάτω παρουσιάζεται η δομή του προγράμματος που χρησιμοποιήθηκε σε μορφή ψευδοκώδικα. Αναλυτικότερα ο κώδικας παρουσιάζεται στις επόμενες ενότητες.

```
1  Παράσταση Αλγορίθμου
2  Ψευδοκώδικας
3
4  Αρχή
5  ##εισαγωγή λειτουργικής μονάδας OR-Tools
6
7  #ορισμός συνάρτησης αρχικοποίησης δεδομένων του προβλήματος
8
9  #ορισμός συνάρτησης εκτύπωσης την λύσης
10
11  □ ορισμός main() :
12
13      (1) αρχικοποίηση δεδομένων
14
15      □ (2) δημιουργία μοντέλου διαχείρισης ευρετηρίου δρομολόγησης
16          ..... //εσωτερικοί δείκτες μεταβλητών που χρησιμοποιεί
17          ..... //ο διακομιστής επίλυσης του προβλήματος
18
19      (3) δημιουργία του μοντέλου δρομολόγησης
20
21      #ορισμός συνάρτησης επιστροφής
22      ..... //αντιστοίχιση των αποστάσεων των αρχικοποιημένων δεδομένων
23      ..... //με τον διακομιστή επίλυσης
24
25      (4) ορισμός κόστους διαδρομής
26
27      □ (5) προσθήκη περιορισμού χρονικών παραθύρων
28          ..... //προσθήκη χρονικού περιορισμού εκτός της αποθήκης
29          ..... //προσθήκη χρονικού περιορισμού για κάθε τοποθεσία
30
31      □ (6) καθορισμός χρονικών στιγμών έναρξης και τερματισμού
32          ..... //για την παραγωγή εφικτών δρομολόγιων
33
34      □ (7) καθορισμός ευρετικού αλγορίθμου
35          ..... //για την εύρεση μιας πρώτης λύσης
36
37      (8) επίλυση του προβλήματος
38
39      (9) εκτύπωση της λύσης
40
41  Τέλος
```

Οι παρακάτω ενότητες παρουσιάζουν προγράμματα σε Python που λύνουν το VRPTW χρησιμοποιώντας OR-Tools

7.3 Δημιουργία δεδομένων

Ο παρακάτω κώδικας δημιουργεί τα δεδομένα (data) για το πρόβλημα χρησιμοποιώντας πίνακα χρόνων, αντί για πίνακα αποστάσεων που υλοποιήθηκαν στα προηγούμενα VRP. Ο πίνακας χρόνων είναι απαραίτητος διότι απαιτείται μονάδα χρόνου για να υλοποιηθεί ο περιορισμός των χρονικών παραθύρων. Στα δεδομένα προστίθενται επίσης η ακολουθία των χρονικών παραθύρων.

```
data['time_windows'] = [
    (0, 500), # depot
    (0, 180), # 1
    (0, 180), # 2
    (0, 180), # 3
    (0, 180), # 4
    (0, 180), # 5
    (0, 180), # 6
    (0, 180), # 7
    (0, 180), # 8
    (0, 180), # 9
    (0, 180), # 10
    (0, 180), # 11
    (0, 180), # 12
    (0, 180), # 13
    (0, 180), # 14
    (0, 180), # 15
    (0, 180), # 16
    (0, 180), # 17
    (0, 180), # 18
    (0, 180), # 19
    (0, 180), # 20
    (0, 180), # 21
    (0, 180), # 22
    (0, 180), # 23
    (0, 180), # 24
    (0, 180), # 25
    (0, 180), # 26
```

Τα καινούργια αντικείμενα είναι:

- Time_matrix: Ακολουθία τιμών χρονικής διάρκειας μεταξύ τοποθεσιών. Εάν γίνει η υπόθεση ότι τα οχήματα ταξιδεύουν με την ίδια ταχύτητα, η λύση του προβλήματος είναι η ίδια ανεξαρτήτως αν θα χρησιμοποιηθεί πίνακας αποστάσεων ή πίνακας χρονικών τιμών.
- Time_windows: Ακολουθία των χρονικών παραθύρων, δηλαδή ο χρόνος που ο προορισμός είναι διαθέσιμος για επίσκεψη. Το χρονικό παράθυρό ενός προορισμού ορίζεται με δύο τιμές, ώρα έναρξης και ώρα λήξης χρονικού παράθυρου.

Σημείωση: Η σειρά των θέσεων στον πίνακα αποστάσεων είναι αυθαίρετη και δεν σχετίζεται με τη σειρά των θέσεων σε οποιαδήποτε λύση του VRPTW.

0. Αποθήκη *Pharmanet*, 1.Αγ.Νικόλαος, 2.Βόνιτσα, 3.Σαρδηνία, 4.Αμφιλοχία, 5.Κατούνα, 6.Μύτικας, 7.Λεπενού, 8.Νεάπολη, 9.Καραϊσκάκι, 10.Αστακός, 11.Αγγελόκαστρο, 12.Σταμνά, 13.Λεσίνι, 14.Κατοχή, 15.Αιτωλικό, 16.Μεσσολόγγι, 17.Ευηνοχώρι, 18.Ναύπακτος, 19.Κάτω μακρινού, 20.Γαβαλού, 21.Ματαράγκα, 22.Ζευγαράκι, 23.Παναιτόλιο, 24.Καινούργιο, 25.Παραβόλα, 26.Θέρμο

7.4 Δημιουργία Συνάρτησης Επιστροφής Χρόνου

Η ακόλουθη συνάρτηση δημιουργεί την χρονική επιστροφή και την τοποθετεί στον διακομιστή επίλυσης του προβλήματος. Επίσης ορίζει το κόστος μιας διαδρομής, που καθορίζει το κόστος της δρομολόγησης, να είναι η χρονική απόσταση μεταξύ των τοποθεσιών.

```
# Create and register a transit callback.
def time_callback(from_index, to_index):
    """Returns the travel time between the two nodes."""
    # Convert from routing variable Index to time matrix NodeIndex.
    from_node = manager.IndexToNode(from_index)
    to_node = manager.IndexToNode(to_index)
    return data['time_matrix'][from_node][to_node]

transit_callback_index = routing.RegisterTransitCallback(time_callback)

# Define cost of each arc.
routing.SetArcCostEvaluatorOfAllVehicles(transit_callback_index)
```

Στο πρόγραμμα του VRPTW, δεν υπάρχει πλέον η συνάρτηση επιστροφής των αποστάσεων.

7.3 Προσθήκη Περιορισμού Χρονικών Παραθύρων

Ο ακόλουθος κώδικας προσθέτει σε όλους τους προορισμούς περιορισμό χρονικών παραθύρων.

```
# Add Time Windows constraint.
time = 'Time'
routing.AddDimension(
    transit_callback_index,
    100, # allow waiting time
    200, # maximum time per vehicle
    False, # Don't force start cumul to zero.
    time)
time_dimension = routing.GetDimensionOrDie(time)
# Add time window constraints for each location except depot.
for location_idx, time_window in enumerate(data['time_windows']):
    if location_idx == 0:
        continue
    index = manager.NodeToIndex(location_idx)
    time_dimension.CumulVar(index).SetRange(time_window[0], time_window[1])
# Add time window constraints for each vehicle start node.
for vehicle_id in range(data['num_vehicles']):
    index = routing.Start(vehicle_id)
    time_dimension.CumulVar(index).SetRange(data['time_windows'][0][0],
                                             data['time_windows'][0][1])
```

Ο κώδικας δημιουργεί μία διάσταση για τους χρόνους διαδρομών των οχημάτων, όμοια με την διάσταση απόστασης στην ενότητα του VRP. Με αυτό τον τρόπο παρακολουθούνται οι ποσότητες που συσσωρεύονται στη διαδρομή ενός οχήματος.

- `time_dimension.CumulVar(index)`: είναι η χρονική απόσταση που έχει διανύσει το όχημα μέχρι την στιγμή που φτάνει στην τοποθεσία με δείκτη (`index`).

Η διάσταση για τον χρόνων διαδρομών, δημιουργείται από την μέθοδο `AddDimension` με τα εξής χαρακτηριστικά:

- Εσωτερικό δείκτη αναφοράς για την επιστροφή των χρόνων ταξιδιού: `transit_callback_index`
- Ένα ανώτατο όριο για την τιμή της αδράνειας (*slack*), δηλαδή τον χρόνο αναμονής του οχήματος στον προορισμό για ξεφόρτωση να είναι ίσο με 30. Ενώ στο πρόβλημα CVRP, είχε οριστεί 0, εδώ είναι απαραίτητο να επιτραπεί χρόνος αναμονής εξαιτίας των χρονικών παράθυρων.
- Ένα ανώτατο όριο για τον συνολικό χρόνο διαδρομής κάθε οχήματος ίσο με: 30, αντίστοιχο της μέγιστης απόστασης διαδρομής που ορίστηκε στις προηγούμενες ενότητες.
- Μεταβλητή τύπου `Boolean`, η οποία να καθορίζει το αν το όχημα θα ξεκινήσει την στιγμή 0 κατά την εκκίνηση των δρομολόγιων. Λόγο των χρονικών

παραθύρων, κάποιο όχημα να χρειαστεί να ξεκινήσει λίγο αργότερα από τα υπόλοιπα οχήματα.

- Το όνομα της διάστασης

Οι επόμενες γραμμές κώδικα,

```
time_dimension.CumulVar(index).SetRange(data['time_windows'][0][0],
                                         data['time_windows'][0][1])
```

απαιτούν από το όχημα να επισκεφτεί τον προορισμό μέσα στο χρονικό παράθυρο που έχει ορίσει.

7.5 Συνάρτηση Εκτύπωσης

Η συνάρτηση εμφανίζει την λύση που επιστράφηκε από τον διακομιστή επίλυσης. Εμφανίζει τις διαδρομές την συνολική διανυόμενη χρονική απόσταση των διαδρομών. Στην επίλυση δίνεται η σειρά επίσκεψης των τοποθεσιών και τα χρονικά παράθυρα λύσης.

```
def print_solution(data, manager, routing, solution):
    """Prints solution on console."""
    time_dimension = routing.GetDimensionOrDie('Time')
    total_time = 0
    for vehicle_id in range(data['num_vehicles']):
        index = routing.Start(vehicle_id)
        plan_output = 'Route for vehicle {}:\n'.format(vehicle_id)
        while not routing.IsEnd(index):
            time_var = time_dimension.CumulVar(index)
            plan_output += '{0} Time({1},{2}) -> '.format(
                manager.IndexToNode(index), solution.Min(time_var),
                solution.Max(time_var))
            index = solution.Value(routing.NextVar(index))
        time_var = time_dimension.CumulVar(index)
        plan_output += '{0} Time({1},{2})\n'.format(manager.IndexToNode(index),
                                                    solution.Min(time_var),
                                                    solution.Max(time_var))
        plan_output += 'Time of the route: {}min\n'.format(
            solution.Min(time_var))
        print(plan_output)
        total_time += solution.Min(time_var)
    print('Total time of all routes: {}min'.format(total_time))
```

Εναλλακτικά, μπορεί αρχικά να αποθηκευτεί η λύση σε μία ακολουθία και στην συνέχεια να εκτυπωθεί.

7.6 Παράθυρο Λύσης

Το παράθυρο λύσης σε μια θέση είναι το χρονικό διάστημα κατά το οποίο πρέπει να φτάσει ένα όχημα, προκειμένου να παραμείνει εντός του χρονοδιαγράμματος. Το παράθυρο λύσης περιέχεται και συνήθως είναι μικρότερο από το χρονικό διάστημα περιορισμού στη θέση.

Στην παραπάνω λειτουργία εκτυπωτή λύσης, το παράθυρο λύσης επιστρέφεται από,

```
(assignment.Min(time_var), assignment.Max(time_var))
```

Όπου,

```
time_var = time_dimension.CumulVar(index)
```

Είναι ο συνολικός χρόνος της διαδρομής.

Εάν η ελάχιστη και η μέγιστη τιμή της μεταβλητής `time_var` είναι ίδιες, σημαίνει ότι ο προορισμός πρέπει να επισκεφτεί σε μία συγκεκριμένη χρονική στιγμή. Αυτό ακόμη σημαίνει ότι πρέπει να αποχωρήσει από την τοποθεσία την στιγμή που φτάνει. Αντίστοιχα αν η ελάχιστη είναι μικρότερη από την μέγιστη, το όχημα μπορεί να περιμένει εκεί πριν αναχωρήσει για τον επόμενο προορισμό.

7.7 Κύρια Συνάρτηση

Το μεγαλύτερο μέρος του κώδικα στην κύρια λειτουργία για το πρόγραμμα VRPTW είναι το ίδιο με το παράδειγμα TSP. Το κομμάτι που προστίθεται στον αλγόριθμο είναι, ο περιορισμός των χρονικών παραθύρων, όπως φαίνεται στην συνέχεια.

```
time_callback = create_time_callback(data)
add_time_window_constraints(routing, data, time_callback)
```

7.8 Παρουσίαση Λύσης σε Χάρτη Εκτέλεση Προγράμματος

Όταν εκτελείτε το πρόγραμμα, εμφανίζει τα ακόλουθα αποτελέσματα:

```
Route for vehicle 0:  
0 Time(0,0) -> 7 Time(20,20) -> 9 Time(55,55) -> 10 Time(62,62) -> 6 Time(93,93) -> 5 Time(137,137) -> 8 Time(168,168) -> 0 Time(177,177)  
Time of the route: 177min  
  
Route for vehicle 1:  
0 Time(0,0) -> 3 Time(40,40) -> 4 Time(47,47) -> 2 Time(88,88) -> 1 Time(98,98) -> 0 Time(167,167)  
Time of the route: 167min  
  
Route for vehicle 2:  
0 Time(0,0) -> 11 Time(22,22) -> 12 Time(32,32) -> 13 Time(50,50) -> 14 Time(57,57) -> 15 Time(71,71) -> 16 Time(87,87) -> 17 Time(96,96) -> 18 Time(129,129) -> 22 Time(178,178)  
-> 0 Time(190,190)  
Time of the route: 190min  
  
Route for vehicle 3:  
0 Time(0,0) -> 21 Time(19,19) -> 20 Time(26,26) -> 19 Time(37,37) -> 26 Time(59,59) -> 25 Time(81,81) -> 24 Time(84,84) -> 23 Time(90,90) -> 0 Time(100,100)  
Time of the route: 100min  
  
Total time of all routes: 634min  
  
Process finished with exit code 0
```

Για κάθε τοποθεσία της διαδρομής αντιστοιχεί , $Time(a,b)$, που είναι το παράθυρο λύσης. Το όχημα πρέπει να επισκεφτεί την κάθε τοποθεσία μέσα σε αυτό το χρονικό περιθώριο.

0. Αποθήκη Pharmanet, 1.Αγ.Νικόλαος, 2.Βόνιτσα, 3.Σαρδηνία, 4.Αμφιλοχία, 5.Κατούνα, 6.Μύτικας, 7.Λεπενού, 8.Νεάπολη, 9.Καραϊσκάκι, 10.Αστακός, 11.Αγγελόκαστρο, 12.Σταμνά, 13.Λεσίνι, 14.Κατοχή, 15.Αιτωλικό, 16.Μεσσολόγγι, 17.Ευηνοχώρι, 18.Ναύπακτος, 19.Κάτω μακρινού, 20.Γαβαλού, 21.Ματαράγκα, 22.Ζευγαράκι, 23.Παναιτώλιο, 24.Καινούργιο, 25.Παραβόλα, 26.Θέρμο

Σε κάθε τοποθεσία που εμφανίζεται στο διάγραμμα του χάρτη, αντιστοιχεί ο αριθμός της τοποθεσίας και τα χρονικά παράθυρα λύσεων.

Το διάγραμμα στην συνέχεια παρουσιάζει την λύση του προβλήματος, δηλαδή την εμφάνιση των διαδρομών και των τοποθεσιών όπως έχουν προκύψει από την λύση στις αντίστοιχες συντεταγμένες x-y.



Η παρουσίαση των διαδρομών σε χάρτη, έγινε με την βοήθεια του συστήματος γεωγραφικών πληροφοριών ArcGis.

Περιορισμοί Πόρων

Στην συνέχεια παρουσιάζεται άλλος ένας χρονικός περιορισμός που συναντάτε συχνά στην δρομολόγηση οχημάτων. Ο περιορισμός αυτός έχει να κάνει με τον χρόνο φόρτωσης των οχημάτων πριν την εκκίνηση της διαδρομής του και την εκφόρτωση κατά το τέλος της. Εάν δεν υπάρχουν θέσεις φόρτωσης των οχημάτων όσες τα οχήματα τότε αναγκαστικά κάποια οχήματα θα περιμένουν. Σκοπός είναι να βρεθούν οι βέλτιστες διαδρομές για το VRPTW, που ικανοποιούν και τους περιορισμούς φόρτωσης εκφόρτωσης στην αποθήκη.

7.9 Δημιουργία δεδομένων

Κατά την δημιουργία των δεδομένων, θα προστεθούν τρία ακόμη δεδομένα, ο χρόνος που χρειάζεται για να φορτωθεί το όχημα και ο χρόνος εκφόρτωσης του, καθώς και τον μέγιστο αριθμό οχημάτων που μπορούν να φορτώνονται.

```
data['num_vehicles'] = 4
data['vehicle_load_time'] = 5
data['vehicle_unload_time'] = 5
data['depot_capacity'] = 2
data['depot'] = 0
return data
```

Τα δεδομένα περιλαμβάνουν:

- `Time_matrix`: Ακολουθία των χρονικών αποστάσεων μεταξύ τοποθεσιών
- `Time_windows`: Ακολουθία χρονικών παραθύρων που είναι διαθέσιμοι οι πελάτες
- `Vehicle_load_time`: Χρόνος που απαιτείται για την φόρτωση ενός οχήματος
- `Vehicle_unload_time`: Χρόνος που απαιτείται για την εκφόρτωση ενός οχήματος
- `Depot_capacity`: Ο μέγιστος αριθμός οχημάτων που μπορούν να φορτώσουν ή να εκφορτώσουν ταυτόχρονα

7.10 Προσθήκη Χρόνων Φόρτωσης – Εκφόρτωσης

Ο ακόλουθος κώδικας προσθέτει τα χρονικά παράθυρα φόρτωσης, εκφόρτωσης που χρειάζεται κάθε όχημα στην αποθήκη. Αυτά τα παράθυρα δημιουργούνται από την μέθοδο `FixedDurationIntervalVar`, που είναι μεταβλητά χρονικά παράθυρα και δεν έχουν αυστηρές ώρες έναρξης και λήξης όπως τα χρονικά παράθυρα στις τοποθεσίες. Το εύρος των χρονικών αυτών παραθύρων καθορίζεται από τις τιμές των `vehicle_load_time` και `vehicle_unload_time`, που τυχάνει να είναι το ίδιο στο συγκεκριμένο παράδειγμα.

```
solver = routing.solver()
intervals = []
for i in range(data['num_vehicles']):
    # Add time windows at start of routes
    intervals.append(
        solver.FixedDurationIntervalVar(
            time_dimension.CumulVar(routing.Start(i)),
            data['vehicle_load_time'], 'depot_interval'))
    # Add time windows at end of routes.
    intervals.append(
        solver.FixedDurationIntervalVar(
            time_dimension.CumulVar(routing.End(i)),
            data['vehicle_unload_time'], 'depot_interval'))
```

7.11 Προσθήκη Περιορισμού Πόρων στην Αποθήκη

Ο κώδικας που ακολουθεί δημιουργεί την αντίθεση ότι τουλάχιστον δύο οχήματα μπορούν να φορτωθούν ή να εκφορτωθούν ταυτόχρονα.

```
depot_usage = [1 for i in range(len(intervals))]  
solver.Add(  
    solver.Cumulative(intervals, depot_usage, data['depot_capacity']  
                      'depot'))
```

- `depot_capacity`, είναι ο μέγιστος αριθμός οχημάτων που μπορούν να φορτώνονται ή να εκφορτώνονται ταυτόχρονα.
- `depot_usage`, είναι ένα διάνυσμα που περιέχει σχετικές τιμές του χρόνου που χρειάζεται ένα όχημα να φορτώσει, εκφορτώσει. Όταν ο χρόνος φόρτωσης και εκφόρτωσης είναι ο ίδιος τότε, το διάνυσμα είναι ένα διάνυσμα που αποτελείται μόνο από τιμές του (1).

8. Συμπεράσματα & Προτάσεις

8.1 Συμπεράσματα

Στην επίλυση όλων των προβλημάτων χρησιμοποιήθηκαν πραγματικές αποστάσεις/χρόνοι ταξιδιού από το οδικό δίκτυο του google.maps. Η χρήση του οδικού δικτύου διαφέρει από την ευκλείδεια απόσταση μεταξύ των κόμβων και προσφέρει μια πιο αληθινή προσέγγιση του προβλήματος. Από τις επιλύσεις συνεπάγεται ότι φαινομενικά χωρικά αποστασιοποιημένες περιοχές μπορεί να είναι πιο κοντά, από ένα άλλο ζευγάρι τοποθεσιών, εξαιτίας της ύπαρξης μιας εθνικής οδού που ενώνει τις δύο τοποθεσίες απευθείας. Σε αντίθεση ένα επαρχιακό δίκτυο μεγαλώνει τις αποστάσεις και τους χρόνους ταξιδιού μεταξύ των τοποθεσιών.

Πιο συγκεκριμένα στο πρώτο πρόβλημα του πλανόδιου πωλητή, το όχημα ολοκληρώνει την διαδρομή σε 426 χιλιόμετρα, αποτέλεσμα δεκτό που και στην πραγματικότητα μπορεί να εξυπηρετηθεί όλη η περιοχή σε μία ημέρα. Το πρόβλημα αυτό θα προσέγγιζε την πραγματικότητα αν δεν υπήρχαν και άλλοι περιορισμοί κατά την διανομή. Όπως ο χρόνος παράδοσης. Ο χρόνος παράδοσης απαιτεί την εξυπηρέτηση των πελατών σε συγκεκριμένα χρονικά διαστήματα. Αποτέλεσμα αυτού να απαιτούνται περισσότερα οχήματα στο πρόβλημα της δρομολόγησης.

συνολικής διανυθείσας απόστασης, μικρότερη σε σχέση με το VRP. Το αποτέλεσμα αυτό φανερώνει και την βάση των βελτιστοποιητικών αλγορίθμων, που βρίσκουν μία πολύ καλή λύση και όχι πάντα την βέλτιστη.



Τέλος με το πρόβλημα της δρομολόγησης με τον περιορισμό χρονικών διαστημάτων VRPTW, προσεγγίζεται το πρόβλημα σε ένα πολύ καλό επίπεδο σε σχέση με αυτό που αντιμετωπίζει η εταιρία στην καθημερινότητα της. Η εταιρία Pharmanet διανέμει φάρμακα σε φαρμακεία και η παράδοση πρέπει να γίνεται σε συγκεκριμένο χρονικό διάστημα 14:00 με 17:00, που τα φαρμακεία είναι κλειστά. Αυτό συνεπάγεται ότι ο συνολικός χρόνος όλων των δρομολογίων δεν θα ξεπερνάνε τις τρεις ώρες. Σε μεγάλο φόρτο εργασίας, δηλαδή σε ταυτόχρονη παραγγελία όλων των πελατών ούτε τα δύο, ούτε τα τρία οχήματα δεν επαρκούν για την επίλυση αυτού του προβλήματος. Η λύση δίνεται με την χρήση τριών οχημάτων που καταφέρνουν να εξυπηρετήσουν όλους του πελάτες σε τρεις ώρες. Σε πραγματικές συνθήκες μεγάλου φόρτου η εταιρία, χρησιμοποιεί εξωτερικό συνεργάτη που διαχειρίζεται την διανομή του υπολειπόμενου φόρτου. Συνολική διανυθείσα απόσταση δεν υπάρχει σε αυτό το πρόβλημα καθώς υπολογίστηκε με χρόνο ταξιδιού η επίλυση. Ο μέσος χρόνος ταξιδιού του κάθε οχήματος είναι 160 λεπτά. Τα αποτελέσματα παρουσιάζονται στον παρακάτω χάρτη.



8.2 Προτάσεις

8.2.1 Εισηγήσεις για Περαιτέρω Έρευνα

Συνέχεια της παρούσας εργασίας μπορεί να αποτελέσει η έρευνα σχετικά με θέματα, επέκτασης του μοντέλου στο συγκεκριμένο δίκτυο διανομής ώστε να προσεγγίζει καλύτερα τις απαιτήσεις της διανομής αντικειμένων. Οι απαιτήσεις των μεταφορών διαφοροποιούνται ανάλογα τα προϊόντα ή τις υπηρεσίες της εκάστοτε επιχείρησης. Για την πλήρωση των απαιτήσεων απαιτείται μεγάλη συλλογή πληροφοριών και δεδομένων από τις επιχειρήσεις και τις μεταφορικές εταιρίες. Με την συγκέντρωση δεδομένων μπορούν να πραγματοποιηθούν πιο αποτελεσματικές εκτιμήσεις της ζήτησης και των χαρακτηριστικών που ανταποκρίνονται στην πραγματικότητα. Η ζήτηση επηρεάζει και τις διαδικασίες χειρισμού των φορτίων. Πρόσθετα στοιχεία που αφορούν χρόνους αναμονής, εισόδου και εξόδου οχημάτων από και προς τους χώρους φόρτωσης, μπορούν να προσδιορίσουν τους μέσους χρόνους που απαιτούνται για τις αντίστοιχες διαδικασίες, ώστε να αποτιμηθούν στο μοντέλο ακόμα καλύτερα.

Ο κυκλοφοριακός φόρτος, έχει γίνει ένας από τους βασικούς παράγοντες επιρροής του προβλήματος της δρομολόγησης, ειδικά στις αστικές περιοχές. Η ανάπτυξη μοντέλου που να εκτιμά τις κυκλοφοριακές συνθήκες και να εξετάζει δυναμικά το πρόβλημα της δρομολόγησης του στόλου οχημάτων, είναι μια συνέχεια της εργασίας

που θα μπορούσε να αποτελέσει περαιτέρω έρευνα. Τα συστήματα τηλεματικής δρομολόγησης στόλου για να παράγουν ολοκληρωμένες λύσεις στον τομέα των logistics, συνδυάζονται με ένα σύστημα ασύρματης εποπτείας στόλου. Η ασύρματη επικοινωνία των οχημάτων με το σύστημα δρομολόγησης επιτυγχάνεται με δέκτες GPS. Με τον τρόπο αυτό υπάρχει απεικόνιση της θέσης και κατάστασης των οχημάτων σε πραγματικό χρόνο, με πλήρες ιστορικό των διαδρομών και άμεση αναφορά της κυκλοφοριακής κίνησης.

8.2.2 Πρόβλημα Πράσινης Δρομολόγησης

Η πράσινη δρομολόγηση (GVRP) παρέχει ένα τοπίο ανάπτυξης της τελευταίας τεχνολογίας των VRP, η οποία ερευνήθηκε για πρώτη φορά το 2006. Από τότε επανεξετάζεται συνεχώς και με βάση το σύστημα ταξινόμησης ορίζονται τρεις κύριες κατηγορίες GVRP. Το πρόβλημα δρομολόγησης της ρύπανσης (Pollution Routing Problem, PRP), δρομολόγηση στην αντίστροφη εφοδιαστική (Reverse Logistics) και Green-VRP.

8.2.2.1 Green-VRP

Η έρευνα της Green-VRP, ασχολείται με την βελτιστοποίηση της κατανάλωσης της ενέργειας των μεταφορών. Οι δραστηριότητες εφοδιαστικής, όπως η ανάπτυξη προϊόντων, η διαδικασία παραγωγής, οι μεταφορές, η διαχείριση αποβλήτων, μπορούν να έχουν μεγάλο αντίκτυπο στο περιβάλλον και, ως εκ τούτου, απαιτούν περισσότερο φιλικές προς το περιβάλλον πρακτικές.

Οι μεταφορές, που είναι ένα από τα σημαντικότερα μέρη της εφοδιαστικής, είναι η αναντικατάστατη βασική υποδομή για την οικονομική ανάπτυξη. Ωστόσο, είναι επίσης ένας από τους μεγαλύτερους καταναλωτές πετρελαίου και ευθύνεται για μεγάλο μέρος των συνολικών ρύπων (Salimifard et al., 2012). Οι ερευνητές και οι επιχειρηματίες τείνουν να δίνουν ιδιαίτερη προσοχή στο ρόλο που θα διαδραματίσουν οι μεταφορές στην επίτευξη θετικών περιβαλλοντικών επιπτώσεων. Η νέα λύση πράσινων μεταφορών ενδέχεται να έρθει σε σύγκρουση με τους καθορισμένους οικονομικούς στόχους. Η διερεύνηση της σχέσης μεταξύ των περιβαλλοντικών επιπτώσεων και των μεταφορών μέσω του σχεδιασμού των διαδρομών θα είναι σε θέση να παράσχει πρακτικές και πολύτιμες προτάσεις σχετικά με αυτή την πράσινη εκστρατεία.

8.2.2.2 Pollution Routing Problem (PRP)

Ο τομέας των οδικών μεταφορών αντιπροσωπεύει ένα μεγάλο ποσοστό των αερίων του θερμοκηπίου (GHG) και ιδίως των εκπομπών CO₂. Η ρύπανση από τις εκπομπές έχει άμεσες ή έμμεσες επικίνδυνες επιπτώσεις στους ανθρώπους και σε ολόκληρο το οικοσύστημα. Οι αυξανόμενες ανησυχίες σχετικά με τις εν λόγω αρνητικές επιπτώσεις των μεταφορών στο περιβάλλον απαιτούν τον εκ νέου προγραμματισμό του δικτύου και της ροής των οδικών μεταφορών, λαμβάνοντας ρητά υπόψη τις εκπομπές αερίων του θερμοκηπίου (Bektas και Laporte, 2011). Η τοποθέτηση του VRP, στο κέντρο του σχεδιασμού των μεταφορών, δημιουργεί ευκαιρίες για τη μείωση των εκπομπών, συμπεριλαμβάνοντας ευρύτερους στόχους που αντικατοπτρίζουν το περιβαλλοντικό κόστος. Το Πρόβλημα Δρομολόγησης της Ρύπανσης (PRP) αποσκοπεί στην επιλογή ενός συστήματος διανομής οχημάτων με λιγότερη ρύπανση, ιδίως μείωση των εκπομπών άνθρακα.

Ο παραδοσιακός στόχος του VRP για τη μείωση της συνολικής απόστασης θα συμβάλει από μόνος του στη μείωση της κατανάλωσης καυσίμων και των εκπομπών ρύπων του περιβάλλοντος. Αλλά αυτή η σχέση πρέπει να μετρηθεί άμεσα χρησιμοποιώντας πιο ακριβείς διατυπώσεις. Οι Bektas και Laporte (2011) υπογράμμισαν αρκετά εμπνευσμένα συμπεράσματα βάσει της παρατήρησης των αποτελεσμάτων των υπολογιστικών τους πειραμάτων. Επίσης, παρείχαν τη δυνατότητα να εξεταστούν ετερογενή οχήματα και χρόνο-εξαρτώμενες συνθήκες στη μελλοντική έρευνα. Με την υποστήριξη πληροφοριών σε πραγματικό χρόνο σχετικά με τις συνθήκες κυκλοφορίας, τα οχήματα μπορούν να κατευθύνονται σε άλλους δρόμους που είναι λιγότερο συμφορημένοι. Αυτό συνεπάγεται μια πιο φιλική προς το περιβάλλον περίπτωση, διότι δημιουργούνται λιγότερες εκπομπές όταν τα οχήματα κινούνται με τις καλύτερες ταχύτητες. Μια αξιοσημείωτη παρατήρηση από τους Bektas και Laporte (2011) είναι ότι ο περιορισμός των χρονικών παραθύρων, καθιστά τις επιπτώσεις της μείωσης της ενέργειας πιο σημαντικές. Με βάση αυτή τη σχέση, η μελλοντική έρευνα μπορεί να περιλαμβάνει την εξερεύνηση της σχέσης μεταξύ του οικονομικού κόστους (συμπεριλαμβανομένων των ποινών) και του περιβαλλοντικού κόστους για τη δρομολόγηση προβλημάτων με περιορισμούς παραθύρων λόγω ήπιου χρόνου. Μια άλλη ενδιαφέρουσα παρατήρηση είναι ότι η μεγαλύτερη διαφοροποίηση

της ζήτησης των πελατών μπορεί να συμβάλει σε μεγαλύτερο περιθώριο για μείωση της κατανάλωσης ενέργειας.

8.2.2.3 VRP in Reverse Logistics (VRPRL)

Η αντίστροφη εφοδιαστική έχει τύχει ιδιαίτερης προσοχής τα τελευταία χρόνια. Ο Dekker και άλλοι (2004), όρισαν την αντίστροφη εφοδιαστική ως "Η διαδικασία σχεδιασμού, υλοποίησης και ελέγχου των ροών των πρώτων υλών προς τα πίσω, στο απόθεμα των διαδικασιών, στη συσκευασία και στα τελικά προϊόντα, από ένα σημείο κατασκευής, διανομής ή χρήσης, έως ένα σημείο ανάκτησης ή κατάλληλης διάθεσης".

Το VRP στην Αντίστροφη Εφοδιαστική (VRPRL) επικεντρώνεται στις πτυχές διανομής της αντίστροφης εφοδιαστικής. Οι περισσότερες μελέτες VRPRL αφορούν την ανακύκλωση αποβλήτων ή αγαθών στο τέλος του κύκλου ζωής τους σε μία ή περισσότερες αποθήκες για περαιτέρω επεξεργασία. το πρόβλημα χωρίζεται σε τέσσερις κατηγορίες, επιλεκτικές λήψεις με τιμολόγηση, συλλογή αποβλήτων, συλλογή αγαθών στο τέλος του κύκλου ζωής τους και ταυτόχρονη διανομή και συλλογή.

1. **Επιλεκτικές λήψεις με τιμολόγηση**, το πρόβλημα της επιλεκτικής δρομολόγησης οχημάτων με τιμές στην αντίστροφη εφοδιαστική χαρακτηρίζεται από την επιλογή μόνο επικερδών σημείων παραλαβής για επίσκεψη και από την όσο το δυνατόν πιο επικερδή εκμετάλλευση συλλογής.
2. **Συλλογή αποβλήτων**, η διαχείριση των αποβλήτων συμπεριλαμβανομένης της αποφυγής των αποβλήτων, της επαναχρησιμοποίησης και της ανακύκλωσης, αποτελεί βασική διαδικασία για την προστασία του περιβάλλοντος και τη διατήρηση των πόρων. Η μεταφορά αποβλήτων είναι σαφώς μέρος της ατζέντας της Πράσινης Εφοδιαστικής (Sbihi και Eglese, 2007a)
3. **Συλλογή αγαθών στο τέλος του κύκλου ζωής τους**, η συλλογή ορισμένων συστατικών των προϊόντων στο τέλος του κύκλου ζωής τους μπορεί να ωφελήσει τον αρχικό κατασκευαστή, διότι τα ανακυκλωμένα υλικά ή

εξαρτήματα παραμένουν λειτουργικά μετά από περαιτέρω απόρριψη ή ανακατασκευή.

4. **Ταυτόχρονη διανομή και συλλογή**, οι μελέτες αυτού του προβλήματος χρησιμοποιούν ένα μοντέλο VRP με ταυτόχρονη παράδοση και παραλαβή για να διαμορφώσουν τη διαδικασία διανομής της αντίστροφης εφοδιαστικής

Λαμβανομένων υπόψη των χαρακτηριστικών του συστήματος αντίστροφης εφοδιαστικής και της λειτουργίας του, η εγκατάσταση πολλαπλών καταστημάτων και οι ταυτόχρονες εργασίες παραλαβής και παράδοσης λαμβάνουν σχετικά μεγαλύτερη προσοχή στις μελέτες του VRPRL. Ωστόσο, τα χρονικά περιθώρια και η περιοδική παράδοση που επιβάλλεται από τον πελάτη σπάνια εξετάζονται στις υπάρχουσες μελέτες, παρόλο που τέτοιες καταστάσεις συναντώνται συχνά σε πραγματικά ζητήματα συλλογής αποβλήτων. Σημαντικό ρόλο στην πράσινη εφοδιαστική θα διαδραματίσουν τα δίκτυα αντίστροφης εφοδιαστικής πολλαπλών κλιμάκων (multi-echelon), η χρήση μοντέλων δρομολόγησης οχημάτων για τη βελτιστοποίηση αυτού του δικτύου θα έχει σημαντικές επιπτώσεις στην πράσινη φύση.

8.2.3 Grid Computing & QoS

Σήμερα, η ανάγκη για συνεργασία, κοινή χρήση δεδομένων και άλλους τρόπους αλληλεπίδρασης που περιλαμβάνουν πολλαπλούς ετερογενείς και γεωγραφικά κατακεμημένους πόρους, συμπεριλαμβανομένων υπερυπολογιστών, υπολογιστών, PDA, σταθμών εργασίας, συστημάτων αποθήκευσης, βάσεων δεδομένων και εφαρμογών ειδικού σκοπού με διάφορες απαιτήσεις (CPU, I/O ή μνήμης) γίνεται ολοένα και πιο αναγκαία. Σε μεγάλου βεληνεκούς εφαρμογές δρομολόγησης στόλου οχημάτων που περιλαμβάνει πολλαπλές αποθήκες και αλληλεπίδραση μεταξύ αυτών, η πολυπλοκότητα στην διαχείριση των δεδομένων αυξάνεται. Για το λόγο αυτό, θα πρέπει να εισαχθούν νέες έννοιες σε επίπεδο αρχιτεκτονικής δικτύου και μεσαίου λογισμικού, ώστε οι εφαρμογές να έχουν πρόσβαση και να μοιράζονται πόρους ή υπηρεσίες με αποτελεσματικό τρόπο. Σε τέτοιες εφαρμογές γίνεται χρήση δύο ευρέως γνωστών εφαρμογών, της ποιότητας υπηρεσιών (Quality of Service, QoS) και του προγραμματισμού πλέγματος (Grid Computing) ή το υπολογιστικό νέφος (Cloud Computing).

Ποιότητα υπηρεσιών, Quality of Service, QoS.

Ποιότητα υπηρεσιών (QoS) είναι η χρήση μηχανισμών ή τεχνολογιών για τη διαχείριση της ταχύτητας και τη διασφάλιση της απόδοσης κρίσιμων εφαρμογών. Επιτρέπει στους οργανισμούς να προσαρμόσουν τη συνολική κυκλοφορία δικτύου τους δίνοντας προτεραιότητα σε συγκεκριμένες εφαρμογές υψηλής απόδοσης. Είναι η ικανότητα παροχής διαφορετικών προτεραιοτήτων σε διαφορετικές εφαρμογές, χρήστες ή ροές δεδομένων ή η εγγύηση ενός συγκεκριμένου επιπέδου απόδοσης σε μια ροή δεδομένων.

Υπάρχουν διάφοροι τρόποι για να εκφραστούν οι απαιτήσεις ποιότητας υπηρεσίας (QoS) για τα καθήκοντα που υποβάλλονται σε κατανεμημένο υπολογιστικό περιβάλλον, όπως το «grid» και το «cloud»(Nikolaos D. Doulamis, et al. 2014).

Χρησιμοποιώντας το QoS, οι οργανισμοί μπορούν να βελτιστοποιήσουν την απόδοση διαφόρων εφαρμογών στο δίκτυό τους και να αποκτήσουν ορατότητα όσον αφορά το ρυθμό bit, την καθυστέρηση και το ρυθμό πακέτων του δικτύου τους. Αυτό διασφαλίζει ότι μπορούν να σχεδιάσουν την κυκλοφορία στο δίκτυό τους και να αλλάξουν τον τρόπο δρομολόγησης των πακέτων στο Διαδίκτυο ή σε άλλα δίκτυα για την αποφυγή καθυστέρησης της μετάδοσης. Ο χρονικός προγραμματισμός και η διαχείριση πόρων είναι σημαντικές για τη βελτιστοποίηση της κατανομής πόρων του πλέγματος πολλαπλών επεξεργαστών (Grid Computing), και τον καθορισμό της ικανότητάς του να παρέχει τις απαραίτητες απαιτήσεις ποιότητας υπηρεσιών (QoS), (Nikolaos D. Doulamis, et al. 2007).

Υπολογιστικό πλέγμα, Grid Computing.

Το υπολογιστικό πλέγμα (Grid Computing), είναι η πρακτική της μόχλευσης πολλαπλών υπολογιστών, συχνά γεωγραφικά κατανεμημένων αλλά συνδεδεμένων με δίκτυα, για συνεργασία για την επίτευξη κοινών εργασιών. Συνήθως εκτελείται σε "πλέγμα δεδομένων", ένα σύνολο υπολογιστών που αλληλεπιδρούν άμεσα μεταξύ τους για το συντονισμό εργασιών.. Το υπολογιστικό πλέγμα είναι ένα νέο υπολογιστικό πρότυπο που υποστηρίζει την κοινή χρήση, τη διασύνδεση και τη χρήση ποικίλων πόρων σε δυναμικά υπολογιστικά συστήματα για την παροχή

υπηρεσιών σε μια διαφανή, ολοκληρωμένη και κατανοητή υπολογιστική υποδομή.

«Προβλέπεται ότι τα δίκτυα επόμενης γενιάς (NGN) θα ενισχυθούν με χαρακτηριστικά που θα προωθήσουν τις προοπτικές των διαχειριστών δικτύων για ένα πλουσιότερο σύνολο δυνατοτήτων όσον αφορά τη διαχείριση, τον έλεγχο ποιότητας υπηρεσιών (QoS) και την εκμετάλλευση των πόρων τους», (Nikolaos Doulamis et al. 2006).

9.Βιβλιογραφία

9.1 Ελληνική

- Αποστολόπουλος Β. (2018), Διερεύνηση Θεμάτων Δρομολόγησης Στόλου Φορτηγών Οχημάτων, Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα
- Βάθης Μ. (2015), Σχεδιασμός και υλοποίηση υβριδικού μεταευρετικού αλγορίθμου βελτιστοποίησης προβλήματος δρομολόγησης οχημάτων με παράθυρα χρόνου, Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα
- Ηλιοπουλου Χ. (2013), Βέλτιστος Σχεδιασμός Δικτύου με Περιορισμούς στη Χωρητικότητα, Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα
- Καρλαύτης,Μ.Γ., Λαγαρός, Ν.Δ. (2010), Επιχειρησιακή Έρευνα και Βελτιστοποίηση για Μηχανικούς, Εκδόσεις Συμμετρία, Αθήνα
- Κορογιαννου Λ. (2011), Επίλυση του Προβλήματος Δρομολογησης Στολου Οχηματων με Εφαρμογες στις Μετρητικες Διαδικασιες Κοινοφελων Οργανισμων, , Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα
- Λιγνός Γ. (2010), Επίλυση Προβλημάτων Δρομολόγησης Φορτηγών Οχημάτων με Χρονικά Παράθυρα και Χωρητικότητα (VRPTW), Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα
- Μουρκούσης Γ. (2008), Μεθοδολογίες Ανάπτυξης Πληροφοριακών Συστημάτων Υποστήριξης Λήψης Αποφάσεων για την Δρομολόγηση Στόλου Οχημάτων και την Αξιολόγηση Εντολών Μεταφοράς Φορτίου στον Τομέα των Logistics, Διδακτορική Διατριβή, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα
- Τουρλούπης Β. (2007), Μαθηματική Μοντελοποίηση και Επίλυση του Προβλήματος Δρομολόγησης Οχημάτων με Κέρδη, Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, Αθήνα

9.2 Ξενόγλωσση

- Alegre, J., Laguna, M., & Pacheco, J. (2007). Optimizing the periodic pick-up of raw materials for a manufacturer of auto parts. *European Journal of Operational Research*, 179(3), 736-746.
- Archetti, C., Bertazzi, L., Laporte, G., & Speranza, M. G. (2007). A branch-and-cut algorithm for a vendor-managed inventory-routing problem. *Transportation Science*, 41(3), 382–391
- Bélangier, N., Desaulniers, G., Soumis, F., & Desrosiers, J. (2006). Periodic airline fleet assignment with time windows, spacing constraints, and time dependent revenues. *European Journal of Operational Research*, 175 (3), 1754-1766.
- Belenguer, J., Benavent, E., Prins, C. Prodhon, C., & Calvo R. W. (2011). A branch-andcut method for the capacitated location-routing problem. *Computers & Operations Research*, 38, 931-941.
- Bell, W. J., Dalberto, L. M., Fisher, M. L., Greenfield, A. J., Jaikumar, R., & Kedia, P. (1983). Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces*, 13(6), 4–23.
- Bektaş, T., & Laporte, G. (2011). The Pollution-Routing Problem. *Transportation Research Part B*, 45, 1232-1250.
- Brandão, J. (2004). A tabu search algorithm for the open vehicle routing problem. *European Journal of Operational Research*, 157(3), 552-564
- Cordeau, J.-F., Gendreau, G., Laporte, G., Potvin, J.-Y., & Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, 53, 512-522.
- Chao, I-M., Golden, B. L., & Wasil, E. A. (1996). The team orienteering problem. *European Journal of Operational Research*, 88, 464-474.
- Chen, H.-K., Hsueh, C.-F., & Chang, M.-S. (2006). The real-time time-dependent vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 42(5), 383-408.
- Chen, Z.-L., & Xu, H. (2006). Dynamic column generation for dynamic vehicle routing with time windows. *Transportation Science*, 40, 74-88.
- Cheng, R., Gen, M., & Tozawa, T. (1995). Vehicle routing problem with fuzzy due-time using genetic algorithms. *Japan Society for Fuzzy Theory and Systems*, 7, 1050-1061

- Christophides, N. (1985). Vehicle routing. In: Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., Shmoys, D.B. (Eds), *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley and Sons Ltd.
- Crainic, T. G., Ricciardi, N., & Storchi, G. (2009). Models for evaluating and planning city logistics systems. *Transportation Science*, 43, 432–454.
- D.D. Bochtis, C.G. Sørensen (2010), *The vehicle routing problem in field logistics: Part I, Part II*, Article, University of Aarhus, Faculty of Agricultural Sciences, Denmark, Tjele
- Derigs, U., Gottlieb, J., Kalkoff, J., Piesche, M., Rothlauf, F., & Vogel, U. (2011). Vehicle routing with compartments: applications, modeling and heuristics. *OR Spectrum*, 33, 885-914.
- Dondo, R. & Cerdá, J. (2007). A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. *European Journal of Operational Research*, 176, 1478-1507.
- Dondo, R., & Cerdá, J. (2013). A sweep-heuristic based formulation for the vehicle routing problem with cross-docking. *Computers & Chemical Engineering*, 48, 293-311.
- Doulamis N., Doulamis A., Litke A, Panagakis A, Varvarigou T., Varvarigos E.(2006). Adjusted fair scheduling and non-linear workload prediction for QoS guarantees in grid computing.
- Doulamis N., Doulamis A., Litke A, Panagakis A, Varvarigou T., Varvarigos E.(2006). Fair Scheduling Algorithms in Grids.
- Doulamis N., Kokkinos P., Varvarigos E. (2014). Resource Selection for Tasks with Time Requirements Using Spectral Clustering.
- Duhamel, C., Lacomme, P., Quilliot, A., & Toussaint, H. (2011). A multi-start evolutionary local search for the two-dimensional loading capacitated vehicle routing problem. *Computers & Operations Research*, 38, 617-640.
- El Fallahi, A., Prins, C., Wolfler Calvo, R., (2008) A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem. *Computers & Operations Research*, 35(5), 1725–1741.
- Figliozzi, M. A. (2009). Planning approximations to the average length of vehicle routing problems with time window constraints. *Transportation Research Part B: Methodological*, 43(4), 438-447.

- Figliozzi, M. A. (2010). An iterative route construction and improvement algorithm for the vehicle routing problem with soft time windows. *Transportation Research Part C: Emerging Technologies*, 18(5), 668-679.
- Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks*, 11, 109-124.
- Fuellerer, G., Doerner, K. F., Hartl, R. F., & Iori, M. (2009). Ant colony optimization for the two-dimensional loading vehicle routing problem. *Computers & Operations Research*, 36, 655-673.
- Gendreau, M., Guertin, F., Potvin, J.-Y., & Séguin, R. (2006a). Neighborhood search heuristics for a dynamic vehicle routing problem with pick-ups and deliveries. *Transport Research Part C: Emerging Technologies*, 14, 157-174.
- Gendreau, M., Iori, M., Laporte, G., & Martello, S. (2006b). A tabu search algorithm for a routing and container loading problem. *Transportation Science*, 40, 342-350.
- Gilbert Laporte (2007), What You Should Know about the Vehicle Routing Problem, Article, HEC Montréal 3000, Canada, Montréal
- Giosa, I., Tansini, I., & Viera, I. (2002). New assignment algorithms for the multi-depot vehicle routing problem. *Journal of the Operational Research Society*, 53, 977-984.
- Gendreau, M., Laporte, G., & Séguin, R. (1996). Stochastic vehicle routing. *European Journal of Operational Research*, 88, 3-12.
- Golden, B. L., Assad, A., & Dahl, R. (1984). Analysis of a large scale vehicle routing problem with an inventory component. *Large Scale Systems*, 7, 181-190.
- Ghiani, G., & Improta, G. (2000). An efficient transformation of the generalized vehicle routing problem. *European Journal of Operational Research*, 122, 11-17.
- Iori, M., Salazar-González, J. J., & Vigo, D. (2003). An exact approach for the symmetric capacitated vehicle routing problem with two dimensional loading constraints. Technical Report OR/03/04, DEIS, Università di Bologna, Bologna, Italy.
- Iori, M., Salazar-González, J. J., & Vigo, D. (2007). An exact approach for the vehicle routing problem with two-dimensional loading constraints. *Transportation Science*, 41, 253-264.

- Kallehauge, B. (2008). Formulations and exact algorithms for the vehicle routing problem with time windows. *Computers & Operations Research*, 35, 2307–2330.
- Khebbache, S., Prins, C., Yalaoui, A., & Reghioui, M. (2009). Memetic Algorithm for two-dimensional loading capacitated vehicle routing problem with time windows. In: *International Conference on Computers & Industrial Engineering*, 2009. CIE 2009.
- Kritzinger, S., Doerner, K. F., Hartl, R. F., Kiechle, G., Stadler, H., & Manohar, S. S. (2012). Using Traffic Information for Time-Dependent Vehicle Routing. *Procedia - Social and Behavioral Sciences*, 39, 217-229.
- Ladany, S., & Mehrez, A. (1984). Optimal routing of a single vehicle with loading constraints. *Transportation Planning and Technology*, 8, 301-306.
- Laporte, G., Louveaux, F. V., & Mercure, H. (1992). The vehicle routing problem with stochastic travel times. *Transportation Science*, 26, 161-170.
- Laporte, G., Gendreau, M., Potvin, J.-Y., & Semet, F. (2000). Classical and modern heuristics for the vehicle routing problem. *International Transactions in Operational Research*, 7, 285-300.
- Leung, S. C. H., Zhou, X, Zhang, D., & Zheng, J. (2011). Extended guided tabu search and a new packing algorithm for the two-dimensional loading vehicle routing problem. *Computers & Operations Research*, 38, 205-215.
- Lin, C., Choy, K.L., Ho, G.T.S., Chung, S.H., Lam, H.Y. (2013), Survey of Green Vehicle Routing Problem: Past and future trends, *Expert Systems with Applications*, Department of Industrial and Systems Engineering, Article, The Hong Kong Polytechnic University, Hong Kong
- Li, J.-Q., Mirchandani, P. B., & Borenstein, D. (2009a). Real-time vehicle routing problems with time windows. *European Journal of Operational Research*, 194, 711- 727.
- Li, X., Leung, S. C. H., Tian, P. (2012). A multistart adaptive memory-based tabu search algorithm for the heterogeneous fixed fleet open vehicle routing problem. *Expert Systems with Application*, 39, 365-374.
- Min, H., Jayaraman, V., & Srivastava, R. (1998). Combined location-routing problems: A synthesis and future research directions. *European Journal of Operational Research*, 108, 1-15.

- Mohammadreza N., Afshin O., Martin T., Lawrence V. (2018), Reinforcement Learning for Solving the Vehicle Routing Problem, Article, Lehigh University, Bethlehem
- Mu, Q., & Eglese, R. W. (2011). Disrupted capacitated vehicle routing problem with order release delay. *Annals of Operations Research*, DOI 10.1007/s10479-011-0947-7.
- Mu, Q., Fu, Z., Lysgaard, J., & Eglese, R. (2011). Disruption management of the vehicle routing problem with vehicle breakdown. *Journal of the Operational Research*, 62,742-749.
- Perboli, G., Tadei, R., & Vigo, D. (2011). The two-echelon capacitated vehicle routing problem: Models and math-based heuristics. *Transportation Science*, 45, 364-380.
- Pirkwieser, S. & Gunther, R. (2009). Boosting a variable neighborhood search for the periodic vehicle routing problem with time windows by ILP techniques. *Proceedings of the 8th Metaheuristic International Conference (MIC 2009)*, Hamburg, Germany.
- Polacek, M., Hartl, R. F., Doerner, K., & Reimann, M. (2004). A variable neighborhood search for the multi depot vehicle routing problem with time windows. *Journal of Heuristics*, 10, 613-627.
- Psaraftis, H. N. (1980). A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transportation Science*, 14, 130-154.
- Qureshi, A. G., Taniguchi, E., & Yamada, T. (2012). A Microsimulation Based Analysis of Exact Solution of Dynamic Vehicle Routing with Soft Time Windows. *Procedia - Social and Behavioral Sciences*, 39, 205-216.
- Russell, R. (1977). An effective heuristic for the M-Tour traveling salesman problem with some side conditions. *Operations Research*, 25, 517-524.
- Soler, D., Albiach, J., & Martínez, E. (2009). A way to optimally solve a time-dependent Vehicle Routing Problem with Time Windows. *Operations Research Letters*, 37(1), 37-42.
- Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35, 254-265.

- Salhi, S., & Nagy, G. (1999). A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling. *Journal of the Operational Research Society*, 50, 1034-1042.
- Sariklis, D., & Powell, S. (2000). A heuristic method for the open vehicle routing problem. *Journal of the Operational Research Society*, 51, 564–573.
- Speidel, V. (1976). EDP-assisted fleet scheduling in tramp and coastal shipping. *Proceedings of the 2nd International Ship Operation Automation Symposium*, Washington, D.C., August 30-September 2.
- Tarantilis, C. D., Zachariadis, E. E., & Kiranoudis, C. T. (2009). A hybrid metaheuristic algorithm for the integrated vehicle routing and three-dimensional container-loading problem. *IEEE Transaction on Intelligent Transportation Systems*, 10, 255-271.
- Thibaut V., Teodor G.C., Michel G., Christian P. (2014), A unified solution framework for multi-attribute vehicle routing problems, Article, *European Journal of Operational Research*
- Tsiligirides, T. (1984). Heuristic methods applied to orienteering. *Journal of the Operational Research Society*, 35, 797–809.
- Xu, J., Yan, F., & Li, S. (2011). Vehicle routing optimization with soft time windows in a fuzzy random environment. *Transportation Research Part E*, 47, 1075-1091.
- Wilson, H., & Weissberg, H. (1967). Advanced dial-a-ride algorithms research project: Final report. Technical Report. R76-20, Department of Civil Engineering, MIT, Cambridge, MA.
- Zachariadis, E. E., Tarantilis, C. D., & Kiranoudis, C. T. (2011). The pallet-packing vehicle routing problem. *Transportation Science*, in press.

9.3 Δικτυογραφία

Βάσω Βεγιάζη (2018), Αλλάζει η νομοθεσία για τα Κέντρα Αποθήκευσης και Διανομής, όλες οι διατάξεις για την εφοδιαστική αλυσίδα, Metaforespress, Διαθέσιμο στο: <https://www.metaforespress.gr/logistics/αλλάζει-η-νομοθεσία-για-τα-κέντρα-αποθ/>

Ημερομηνία προσπέλασης: 8/2/2021

ΦΕΚ (2020), Ρύθμιση θεμάτων Εφοδιαστικής και άλλες διατάξεις, e-nomothesia,

Διαθέσιμο στο:<https://www.e-nomothesia.gr/kat-epikheireseis/n-4302-2014>

Ημερομηνία προσπέλασης: 8/2/2021

Google (2020), Vehicle Routing, Google OR-Tools, Διαθέσιμο στο:

<https://developers.google.com/optimization/routing>,

Ημερομηνία προσπέλασης: 7/2/2021