



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

**Ένας μεθευρετικός αλγόριθμος περιορισμένης αναζήτησης
για το πρόβλημα δρομολόγησης οχημάτων με χωριστή
εξυπηρέτηση**



Συγγραφέας: Αλεξάκης Ιωάννης

Επιβλέπων: Μαρινάκης Ιωάννης

ΧΑΝΙΑ 2018

Περιεχόμενα

ΠΕΡΙΛΗΨΗ.....	3
---------------	---

1. ΕΙΣΑΓΩΓΗ.....	4
1.1 Εισαγωγή στην εφοδιαστική αλυσίδα	4
1.2 Βασικές Λειτουργίες της Εφοδιαστικής Αλυσίδα.....	6
2. ΠΡΟΒΛΗΜΑΤΑ ΔΡΟΜΟΛΟΓΗΣΗΣ ΟΧΗΜΑΤΩΝ	8
2.1 Εισαγωγή στο πρόβλημα δρομολόγησης οχημάτων (VRP)	8
2.2 Το πρόβλημα δρομολόγησης οχημάτων (VRP)	8
2.3 Το περιορισμένης χωρητικότητας πρόβλημα δρομολόγησης οχημάτων.....	10
2.4 Το πρόβλημα δρομολόγησης οχημάτων με χωριστή εξυπηρέτηση	12
2.4 Άλλα προβλήματα δρομολόγησης οχημάτων.....	14
3. ΕΥΡΕΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ.....	17
3.1 Εισαγωγή στους αλγόριθμους εύρεσης βέλτιστης λύσης	17
3.2 Αλγόριθμοι απληστίας (GreedyAlgorithms).....	18
3.3 Προσεγγιστικοί αλγόριθμοι (Approximationalgorithms)	18
3.4 Αλγόριθμοι τοπικής αναζήτησης	19
4. ΜΕΘΕΥΡΕΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ.....	19
4.1 Εισαγωγή στους μεθευρετικούς αλγορίθμους	19
4.2 Ο αλγόριθμος περιορισμένης αναζήτησης (TabuSearch)	20
4.3 Ο αλγόριθμος Προσομοιωμένης Ανόπτησης (SimulatedAnnealing)	22
5. ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ	22
5.1 Συνοπτική περιγραφή του προβλήματος	22
5.2 Μοντελοποίηση του προβλήματος	23
5.3 Εύρεση αρχικής εφικτής λύσης μέσω του Αλγόριθμου Απληστίας	26
5.4 Εύρεση της βέλτιστης λύσης με τη χρήση του αλγόριθμου TabuSearch.....	27
6. ΑΝΑΛΥΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ.....	29
6.1 Περιγραφή και ανάλυση των αποτελεσμάτων	29
6.2 Αποτελέσματα των αλγορίθμων	29
.....	31
7. ΣΥΜΠΕΡΑΣΜΑΤΑ	39
Ανάλυση αποτελεσμάτων και συμπεράσματα.....	39
Βιβλιογραφία.....	41

Ευχαριστίες

Η παρούσα διπλωματική εργασία εκπονήθηκε υπό την επίβλεψη του επίκουρου καθηγητή κ. Ιωάννη Μαρινάκη.

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον κ. Ιωάννη Μαρινάκη για το ενδιαφέρον θέμα διπλωματικής εργασίας που μου ανέθεσε, καθώς και για την ουσιαστική καθοδήγηση και συμβολή του στην υλοποίηση της εργασίας και τον Ελευθέριο Τσακιράκη για την πολύτιμη βοήθεια του.

Εν συνεχεία θα ήθελα να ευχαριστήσω τους φίλους μου για την ηθική και πρακτική υποστήριξη.

Κλείνοντας θα ήθελα να ευχαριστήσω την οικογένεια μου , για τις θυσίες που κάνουν όλα αυτά τα χρόνια.

ΠΕΡΙΛΗΨΗ

Οι διαρκώς αυξανόμενες απαιτήσεις των πελατών καθιστούν απαραίτητη την βελτίωση της Εφοδιαστικής Αλυσίδας η οποία αποτελεί καθοριστικό παράγοντα για την ανταγωνιστικότητα ακόμα και για τη βιωσιμότητα της κάθε επιχείρησης. Ένας από τους στόχους της Εφοδιαστικής είναι η μείωση του κόστους των μεταφορών των προϊόντων και των αποθεμάτων.

Η συγκεκριμένη εργασία έχει ως αντικείμενο το πρόβλημα δρομολόγησης οχημάτων με ξεχωριστή εξυπηρέτηση (SPLIT-VRP), με στόχο την εύρεση του βέλτιστου δρομολογίου των οχημάτων για την εξυπηρέτηση ενός συνόλου πελατών οι οποίοι δεν χρειάζεται να εξυπηρετηθούν αποκλειστικά από ένα όχημα. Γίνεται δηλαδή αξιοποίηση ολόκληρου του φορτίου το οποίο μπορεί να φορτωθεί σε ένα όχημα από έναν στόλο πανομοιότυπων οχημάτων.

Για την περιγραφή του προβλήματος ορίζονται κατάλληλα οι απαραίτητοι περιορισμοί για τον αριθμό και τις τοποθεσίες των πελατών και το συνολικό φορτίο που έχει τη δυνατότητα να φορτωθεί σε κάθε φορτηγό. Αρχικά υλοποιείται ένας ευρετικός αλγόριθμος του πλησιέστερου γείτονα και στη συνέχεια γίνεται διερεύνηση της κατάλληλης γειτονιάς αναζήτησης με την εφαρμογή μεθόδων τοπικής αναζήτησης (1-1 exchange και 2-opt) με την χρήση του μεθευρετικού αλγορίθμου περιορισμένης αναζήτησης (TABUsearch). Η αποτελεσματικότητα του προτεινόμενου αλγορίθμου δοκιμάζεται μέσω υπολογιστικών μελετών σε προβλήματα διαφορετικών μεγεθών. Τα αποτελέσματα που δίδει ο προτεινόμενος αλγόριθμος συγκρίνονται με τις καλύτερες λύσεις που υπάρχουν στη βιβλιογραφία και δίδονται προτάσεις για περαιτέρω βελτίωσή του.

Για την ανάπτυξη του αλγορίθμου χρησιμοποιήθηκε το προγραμματιστικό περιβάλλον της MATLAB.

1. ΕΙΣΑΓΩΓΗ

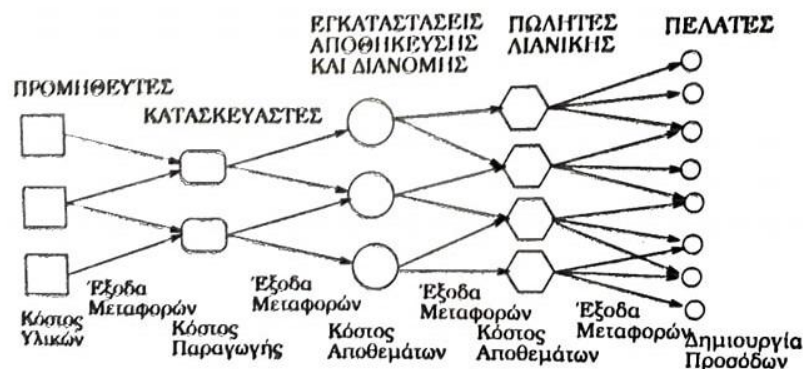
1.1 Εισαγωγή στην εφοδιαστική αλυσίδα

Είναι αδιαμφισβήτητο το γεγονός ότι τις τελευταίες δεκαετίες υπάρχει ραγδαία ανάπτυξη της τεχνολογίας και κατ' επέκταση των αναγκών της κοινωνίας. Έτσι όσον αφορά τις επιχειρήσεις αυξάνονται οι ανάγκες και οι προσδοκίες των πελατών. Επομένως κρίνεται αναγκαία η χρήση νέων μέσων και διαδικασιών για την βελτιστοποίηση της λειτουργίας τους και την μεγιστοποίηση του κέρδους τους. Ο έντονος ανταγωνισμός στην αγορά ανάγκασε τις επιχειρήσεις να δώσουν ιδιαίτερη έμφαση στην ανάπτυξη και στη βελτιστοποίηση της εφοδιαστικής αλυσίδας.

Με τον όρο *Εφοδιαστική Αλυσίδα* εννοούμε όχι μόνο τη ροή υλικών από τον προμηθευτή πρώτων υλών ή τον κατασκευαστή μέχρι τον τελικό καταναλωτή, αλλά παράλληλα και τη ροή πληροφοριών μεταξύ των μελών της αλυσίδας. Η διαχείρισή της γίνεται σε δύο επίπεδα:

- Επίπεδο προγραμματισμού: στο επίπεδο αυτό, αναλύονται τα δεδομένα προμηθειών, αναλώσεων παραγωγής, αποθεματοποίησης και πωλήσεων, γίνονται προβλέψεις και πλάνα πάνω στα οποία βασίζεται ο προγραμματισμός.
- Επίπεδο εκτέλεσης: στο στάδιο αυτό εκτελείται το πλάνο που έχει καθορισθεί στο επίπεδο προγραμματισμού και παρακολουθείται η εξέλιξη του βάσει των δεδομένων και πληροφοριών που συλλέγονται από όλο το εύρος της Εφοδιαστικής Αλυσίδας.

Μία τυπική Εφοδιαστική Αλυσίδα (supplychain) αποτελείται από κατασκευαστές και προμηθευτές, από χώρους αποθήκευσης, κέντρα διανομών, μεταφορείς, πωλητές λιανικής, πελάτες, αλλά και από πρώτες ύλες, αποθέματα κατά την παραγωγική διαδικασία και έτοιμα προϊόντα που ρέουν μεταξύ των παραπάνω σημείων.



Σχήμα1.1(πηγή; Ιωάννης Μαρινάκης, Αθανάσιος Μυγδαλάς 'Σχεδιασμός και Βελτιστοποίηση της Εφοδιαστικής Αλυσίδας' εκδόσεις σοφία)

1.2 Βασικές Λειτουργίες της Εφοδιαστικής Αλυσίδα

Τροφοδοσία -Αγορές – Προμήθειες

Ο όρος αγορές ή προμήθειες αναφέρεται στην διαδικασία της απόκτησης προϊόντων ή υπηρεσιών από πηγές εκτός της επιχείρησης με σκοπό την υποστήριξη των διαδικασιών παραγωγής ,πωλήσεων, προβολής των προϊόντων. Οι αγορές αυτές μπορεί να αναφέρονται σε προϊόντα, που θα ενσωματωθούν στο τελικό προϊόν και μπορεί να είναι πρώτες ύλες ή μπορεί να είναι υλικά συσκευασίας, ακόμη και ενέργεια (καύσιμα, ηλεκτρισμός) ή τέλος, να είναι και υπηρεσίες.

Διαχείριση αποθεμάτων

Ο όρος διαχείριση αποθεμάτων αναφέρεται στο σύνολο των διαδικασιών που εξασφαλίζουν την σωστή ποσότητα προϊόντων στις αποθήκες τα οποία είναι απαραίτητα για την διεκπεραίωση των παραγωγικών διαδικασιών. Τα αποθέματα και όποια διαδικασία τα περιλαμβάνει είναι πολύ σημαντικά για την βελτιστοποίηση της παραγωγικής διαδικασίας.

Διαχείριση υλικών

Ο όρος διαχείριση υλικών αναφέρεται στο σύνολο των εσωτερικών μετακινήσεων από τους αποθηκευτικούς χώρους στη παραγωγή και αντίστροφα.

Αποθήκευση

Η αποθήκευση αναφέρεται στον σχεδιασμό, την οργάνωση και τη λειτουργία της αποθήκης. Δηλαδή, σχετίζεται με την εκτέλεση της εργασίας παραλαβής, φύλαξης, εξαγωγής και παράδοσης των προϊόντων στους πελάτες.

Μεταφορές – διανομές

Το τρίτο στοιχείο της *Εφοδιαστικής Αλυσίδας* είναι η διακίνηση (μεταφορές και διανομή). Ο όρος διακίνηση συνδέεται με την εργασία εξεύρεσης του άριστου τρόπου της φυσικής μετακίνησης των προϊόντων που έχει παραγγείλει η επιχείρηση, από τις εγκαταστάσεις του προμηθευτή έως τις εγκαταστάσεις της και από τις εγκαταστάσεις στους πελάτες της. Το κόστος διακίνησης αποτελεί ένα από τα πιο αξιόλογα στοιχεία του συνολικού κόστους που πρέπει να μελετηθεί με προσοχή κατά τον σχεδιασμό της Εφοδιαστικής Αλυσίδας. Οι μεταφορές χωρίζονται σε δύο κατηγορίες, η πρώτη κατηγορία μεταφορών είναι αυτή που περιλαμβάνει την διακίνηση των πρώτων υλών στους χώρους επεξεργασίας και στα εργοστάσια και σε αυτές που περιλαμβάνουν την μεταφορά του τελικού προϊόντος από τα εργοστάσια ή την αποθήκη προς τον καταναλωτή. Τα προβλήματα που συναντώνται στις δύο παραπάνω κατηγορίες αφορούν τον αριθμό των οχημάτων, τη δρομολόγηση, το σχεδιασμό του δικτύου διανομής, τον χρονοπρογραμματισμό των δρομολογίων, την επιλογή του προσωπικού που θα εκτελέσει τα δρομολόγια καθώς και υποκατηγορίες αυτών.

Οι επιχειρήσεις για να αντιμετωπίσουν τα παραπάνω προβλήματα έχουν αναπτύξει μεθοδολογίες και αλγόριθμους βελτιστοποίησης της παραγωγικής διαδικασίας. Παρακάτω θα αναφερθούν οι κύριες κατηγορίες προβλημάτων και οι τρόποι επίλυσής τους.

2. ΠΡΟΒΛΗΜΑΤΑ ΔΡΟΜΟΛΟΓΗΣΗΣ ΟΧΗΜΑΤΩΝ

2.1 Εισαγωγή στο πρόβλημα δρομολόγησης οχημάτων (VRP)

Τις τελευταίες δεκαετίες παρατηρείται ραγδαία αύξηση στη χρήση πακέτων βελτιστοποίησης, τα οποία βασίζονται σε τεχνικές Μαθηματικού προγραμματισμού, για την αποτελεσματική διαχείριση της παροχής προϊόντων και υπηρεσιών σε δίκτυα διανομής. Επίσης έχει διαπιστωθεί η συνεχώς αυξανόμενη ενσωμάτωση των πληροφοριακών συστημάτων στις παραγωγικές και εμπορικές διαδικασίες και η συμβολή τους στην μείωση του κόστους μεταφορών και κατ' επέκταση στην αύξηση των κερδών των επιχειρήσεων.

Σημαντικός παράγοντας είναι η ανάπτυξη μοντέλων και αλγοριθμικών εργαλείων που εφαρμόζονται τα τελευταία χρόνια. Τα προτεινόμενα μοντέλα λαμβάνουν υπόψη όλα τα χαρακτηριστικά και τους περιορισμούς των προβλημάτων διανομής και μεταφορών και βρίσκουν βέλτιστες λύσεις.

2.2 Το πρόβλημα δρομολόγησης οχημάτων (VRP)

Τα προβλήματα που έχουν ως αντικείμενο την δρομολόγηση οχημάτων εντός των επιχειρησιακών δραστηριοτήτων και περιλαμβάνουν την παράδοση και την παραλαβή προϊόντων από τους πελάτες ονομάζονται *Προβλήματα Δρομολόγησης Οχημάτων*. Το πλήθος των πελατών που θα εξυπηρετηθούν είναι συγκεκριμένο όπως και οι γεωγραφικές θέσεις αυτών. Η αποθήκη ή οι αποθήκες από τις οποίες θα ξεκινούν τα δρομολόγια όπως και ο αριθμός των διαθέσιμων οχημάτων είναι και αυτός συγκεκριμένος. Σκοπός τα επίλυσης του συγκεκριμένου προβλήματος είναι η ανεύρεση των δρομολογίων με τα οποία θα επιτυγχάνεται η εξυπηρέτηση όλων των πελατών και παράλληλα η μείωση του κόστους μεταφορών. Η λύση του προβλήματος θα πρέπει να ικανοποιεί όλους τους περιορισμούς οι οποίοι προέρχονται από τα χαρακτηριστικά των οχημάτων, των πελατών, των οδηγών και των διαδρομών.

Μερικά από τα χαρακτηριστικά των οχημάτων είναι από ποια αποθήκη προέρχονται

και αν υπάρχει πιθανότητα να τερματίσουν σε μία άλλη ή στην ίδια αποθήκη, η χωρητικότητα τους, αν υπάρχουν οχήματα για εξειδικευμένα προϊόντα, οι δρόμοι που μπορούν να προσπελάσουν, αν υπάρχουν διαθέσιμα μηχανήματα για την φόρτωση-εκφόρτωση.

Μερικά από τα χαρακτηριστικά των πελατών είναι το σημείο που βρίσκεται ο κάθε πελάτης, η ποσότητα των αγαθών που πρέπει να παραλάβει ή να παραδώσει ο κάθε πελάτης, τα χρονικά διαστήματα που μπορεί να εξυπηρετηθεί ο πελάτης, ο χρόνος που απαιτείται για την παράδοση ή παραλαβή του προϊόντος από τον πελάτη .

Μερικά χαρακτηριστικά των οδηγών αφορούν τις συνθήκες και τις ώρες εργασίας τους πιο αναλυτικά πρέπει να τηρείται το οκτάωρο ύπνου, να μην οδηγούν παραπάνω από 10 ώρες συνεχούς οδήγησης και να μην οδηγούν παραπάνω από 6 μέρες .

Τα χαρακτηριστικά που αφορούν τις διαδρομές σχετίζονται με το προϊόν που διανέμεται ή εξαρτώνται από τα χαρακτηριστικά που προαναφέρθηκαν τέτοια είναι η πεπερασμένη χωρητικότητα των οχημάτων, υπάρχουν πελάτες οι οποίοι εξυπηρετούνται μέσω παράδοσης ή παραλαβής ή και των δύο μαζί, τα χρονικά παράθυρα εξυπηρέτησης των πελατών, τα ωράρια των οδηγών κτλ

Όλα τα παραπάνω χαρακτηριστικά καθορίζουν τους περιορισμούς που πρέπει να ληφθούν υπόψη έτσι ώστε η λύση που θα βρεθεί να είναι βέλτιστη, εφικτή και υλοποιήσιμη. Οι στόχοι ορίζονται από την εκάστοτε επιχείρηση ανάλογα με το τι θέλει να επιτύχει, τις απαιτήσεις των πελατών και τις προτεραιότητες που η ίδια βάζει. Έτσι οι στόχοι της επίλυσης προβλημάτων δρομολόγησης οχημάτων είναι οι ακόλουθοι:

- Η ελαχιστοποίηση του συνολικού κόστους μεταφορών των προϊόντων. Το κόστος εξαρτάται από την συνολική απόσταση ή από τον συνολικό χρόνο που θα κάνει για να εξυπηρετηθούν εξ' ολοκλήρου οι πελάτες και το πάγιο κόστος ανά όχημα και ανά οδηγό.
- Η ελαχιστοποίηση του αριθμού των οχημάτων.
- Η ισορροπία μεταξύ των διαδρομών σε σχέση με τις απαιτούμενες ώρες που χρειάζονται.
- Η ελαχιστοποίηση των ποινών που πηγάζει από την μερική εξυπηρέτηση των πελατών.

Για την επίλυση των VRP προβλημάτων έχουν αναπτυχθεί διάφοροι αλγόριθμοι που έχουν σκοπό να βελτιστοποιήσουν το τελικό αποτέλεσμα και θα παρουσιαστούν σε επόμενο κεφάλαιο.

2.3 Το περιορισμένης χωρητικότητας πρόβλημα δρομολόγησης οχημάτων

Στο πρόβλημα δρομολόγησης οχημάτων με περιορισμένη χωρητικότητα έχουμε μία αποθήκη, ένα στόλο οχημάτων με το κάθε όχημα να έχει συγκεκριμένη χωρητικότητα. Ένας αριθμός πελατών πρέπει να εξυπηρετηθεί με τον κάθε πελάτη να έχει μία συγκεκριμένη ζήτηση. Εδώ υπάρχει ο περιορισμός χωρητικότητας, με την συνολική ζήτηση των πελατών που θα εξυπηρετηθούν από ένα όχημα, να απαγορεύεται να υπερβεί τη συνολική χωρητικότητα του οχήματος. Η εξυπηρέτηση για τον κάθε πελάτη διαρκεί μία συγκεκριμένη χρονική διάρκεια, και το φορτηγό μετά από κάθε εξυπηρέτηση θα πρέπει να έχει ένα χρονικό περιθώριο ικανό για να μπορέσει να επιστρέψει στην αποθήκη. Μετά την εξυπηρέτηση όλων των πελατών που ανήκουν στον κάθε κύκλο, το όχημα επιστρέφει στην αποθήκη. Εδώ στόχος είναι η ελαχιστοποίηση της ευκλείδειας απόστασης που θα διανυθεί συνολικά. Τα χαρακτηριστικά είναι ότι ο κάθε κύκλος περνάει από την αποθήκη, κάθε πελάτης θα δεχθεί επίσκεψη από έναν μόνο κύκλο και το άθροισμα της ζήτησης των κόμβων που επισκέπτονται από έναν κύκλο δεν πρέπει να ξεπερνάει τη συνολική χωρητικότητα του οχήματος.

Μία κλασική μορφοποίηση του προβλήματος δρομολόγησης οχημάτων περιορισμένης χωρητικότητας έχει παρουσιαστεί από τους Fischer & Jaikumar [**]. Έστω ότι έχουμε ένα πλήρη γράφημα $G=(V,A)$, όπου $V=\{1,...,n\}$ είναι το σύνολο των κόμβων με τον κόμβο 1 να αντιστοιχεί στην αποθήκη. Οι κόμβοι δηλαδή οι πελάτες και η αποθήκη αντιστοιχούν με σημεία στο επίπεδο (τα δεδομένα είναι συντεταγμένες) και το κόστος υπολογίζεται από την Ευκλείδεια απόσταση ανάμεσα σε δυο κόμβους και συμβολίζεται ως c_{ij} . Έτσι το x_{ij} δηλώνει την εξυπηρέτηση του j πελάτη αμέσως μετά την εξυπηρέτηση του i και είναι το πλήθος των

οχημάτων. Έτσι έχουμε

$$X_{ijk} = \begin{cases} 1, & \text{εάν το όχημα } k \text{ επισκέπτεται τον πελάτη } j \text{ αμέσως μετά τον πελάτη } i \\ 0, & \text{αλλιώς} \end{cases}$$

$$Y_{jk} = \begin{cases} 1, & \text{εάν ο πελάτης } i \text{ επισκέπτεται από το όχημα } k \\ 0, & \text{αλλιώς} \end{cases}$$

Τότε το βασικό πρόβλημα δρομολόγησης οχημάτων μπορεί να καθορισθεί ως εξής :

$$\min \sum_{i,j} c_{ij} \sum_k x_{ijk}$$

υπό

$$\sum_k y_{ik} = \begin{cases} 1, & i = 2, \dots, n \\ m, & i = 1 \end{cases} \quad (1)$$

$$\sum_j q_i y_{ij} \leq Q_k \quad k = 1, \dots, m \quad (2)$$

$$\sum_j x_{ijk} = \sum_j x_{jik} = y_{ik} \quad i = 1, \dots, n, \quad k = 1, \dots, m \quad (3)$$

$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1, \text{ για όλα τα } S \subseteq \{2, \dots, n\}, k = 1, \dots, m \quad (4)$$

$$y_{ik} \in \{0,1\}, \quad k = 1, \dots, m, \quad i = 1, \dots, n \quad (5)$$

$$x_{ijk} \in \{0,1\}, \quad k = 1, \dots, m, \quad i, j = 1, \dots, n \quad (6)$$

Ο περιορισμός (1) δείχνει ότι κάθε πελάτης εκχωρείται σε ένα μόνο όχημα, εκτός από την αποθήκη που την επισκέπτονται όλα τα οχήματα.

Ο περιορισμός (2) δείχνει ότι δεν πρέπει να παραβιαστεί η χωρητικότητα κάθε οχήματος σε ένα δρομολόγιο.

Ο περιορισμός (3) δείχνει ότι ένα όχημα που φεύγει επισκέπτεται ένα πελάτη φεύγει από τον πελάτη.

Ο περιορισμός (4) δείχνει ότι δεν επιτρέπει την υπερφόρτωση.

2.4 Το πρόβλημα δρομολόγησης οχημάτων με χωριστή εξυπηρέτηση

Το πρόβλημα δρομολόγησης οχημάτων με χωριστή εξυπηρέτηση (*The Split Delivery Vehicle Routing Problem*) διαφέρει από το κλασικό πρόβλημα δρομολόγησης οχημάτων στο γεγονός ότι ο κάθε πελάτης μπορεί να εξυπηρετηθεί από παραπάνω από ένα όχημα εάν αυτό επιτρέπει την μείωση του συνολικού κόστους. Αυτό συμβαίνει κυρίως σε προβλήματα όπου έχουμε ομογενή οχήματα, δηλαδή οχήματα με την ίδια χωρητικότητα και η ζήτηση του πελάτη είναι μεγαλύτερη από την χωρητικότητα των οχημάτων. Δύο είναι οι τρόποι με τους οποίους μπορεί να εξυπηρετηθεί ένας πελάτης είτε να σταλεί ένα όχημα και μετά να επιστρέψει το ίδιο όχημα στην αποθήκη να ξαναφορτώσει και να μετά να επιστρέψει το ίδιο όχημα στον πελάτη είτε να σταλεί και ένα δεύτερο όχημα που να περάσει από τον πελάτη και να του παραδώσει την επιπλέον ποσότητα. Μία κλασική μορφοποίηση του προβλήματος δρομολόγησης οχημάτων με χωριστή εξυπηρέτηση έγινε από τους CArchetti και M. G. Speranza [**] και είναι η παρακάτω.

$$\min \sum_{i=0}^n \sum_{j=0}^n \sum_{u=1}^m c_{ij} * x_{ij}^u \quad (1)$$

Υπό

$$\sum_{i=0}^n \sum_{u=1}^m x_{ij}^u \geq 1, \quad j = 0, \dots, n \quad (2)$$

$$\sum_{i=0}^n x_{ip}^u - \sum_{j=0}^n x_{pj}^u = 0, \quad p = 0, \dots, n; u = 1, \dots, m \quad (3)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij}^u \leq |S| - 1, \quad v = 1, \dots, m; S \subseteq V - \{0\} \quad (4)$$

$$y_{iu} \leq d_i \sum_{j=0}^n x_{ij}^u, \quad i = 1, \dots, n; u = 1, \dots, m \quad (5)$$

$$\sum_{u=1}^m y_{iu} = d_i, \quad i = 1, \dots, n \quad (6)$$

$$\sum_{i=1}^n y_{iu} \leq Q, u = 1, \dots, m \quad (7)$$

$$x_{ij}^u \in \{0,1\}, \quad i = 0, \dots, n; j = 0, \dots, n; u = 1, \dots, m \quad (8)$$

$$y_{iu} \geq 0, \quad i = 1, \dots, n; u = 1, \dots, m \quad (9)$$

Οι περιορισμοί (2) - (4) είναι οι κλασικοί περιορισμοί δρομολόγησης. Ο περιορισμός (2) επιβάλλει ότι κάθε πελάτης δέχεται επίσκεψη τουλάχιστον μία φορά, ενώ ο (3) είναι ο περιορισμός διατήρησης της ροής και (4) είναι ο περιορισμός που δεν επιτρέπει την υπερφόρτωση.

Ο περιορισμός (5) επιβάλλει ότι ο πελάτης i εξυπηρετείται από το όχημα u μόνο αν επισκέπτεται τον συγκεκριμένο πελάτη.

Ο περιορισμός (6) εξασφαλίζει ότι ικανοποιείται ολόκληρη η ζήτηση κάθε πελάτη, ενώ ο περιορισμός (7) εγγυάται ότι η ποσότητα που παραδίδεται από κάθε όχημα δεν υπερβαίνει την χωρητικότητα του οχήματος.

Αξίζει να σημειωθεί ότι οι περιορισμός (2) δεν είναι απαραίτητος για να διαμορφωθεί σωστά το πρόβλημα. Χρησιμοποιούνται για την ενίσχυση της διαμόρφωσης και τη βελτίωση της ποιότητας της λύσης της γραμμικής χαλάρωσης.

u : μετρητής οχημάτων

Q : το μέγιστο φορτίο που μπορεί να δεχτεί το κάθε όχημα.

d_i : η ζήτηση του εκάστοτε πελάτη

x_{ij}^u : δυαδική μεταβλητή η οποία γίνεται ίση με 1 όταν το u όχημα πηγαίνει απευθείας από τον i σταθμό στον j και 0 διαφορετικά.

y_{iu} : η ποσότητα που παραδίδει το όχημα u στον σταθμό i .

2.4 Άλλα προβλήματα δρομολόγησης οχημάτων

- **Πρόβλημα Δρομολόγησης Οχημάτων με Χρονικά Παράθυρα (VRPTW)**

Στο συγκεκριμένο πρόβλημα οι πελάτες-κόμβοι έχουν περισσότερα χαρακτηριστικά και η λύση θα πρέπει να ικανοποιήσει περισσότερους περιορισμούς. Σε αυτή την κατηγορία προβλημάτων έχουμε ένα σύνολο πελατών γεωγραφικά διασκορπισμένων σε συγκεκριμένη περιοχή. Ο κάθε ένας από αυτούς έχει συγκεκριμένη ποσότητα φορτίου που πρέπει να παραλάβει και ιδιαιτερότητα του προβλήματος είναι ότι ο πελάτης πρέπει να εξυπηρετηθεί μέσα σε μια χρονική

περίοδο, το χρονικό παράθυρο. Πριν και μετά το χρονικό παράθυρο, ο πελάτης δεν μπορεί να εξυπηρετηθεί. Τα οχήματα περιορισμένης χωρητικότητας βρίσκονται την στιγμή $t_c = 0$ στην αποθήκη. Έτσι το πρόβλημα δρομολόγησης οχημάτων με χρονικά παράθυρα (Vehicle Routing Problem with Time Windows/VRPTW) προκύπτει σαν επέκταση του CVRP με ίδια αντικειμενική συνάρτηση ελαχιστοποίησης του κόστους, αλλά με επιπλέον περιορισμό την εξυπηρέτηση των πελατών μέσα σε συγκεκριμένα χρονικά παράθυρα (timewindows).

- **Πρόβλημα Δρομολόγησης οχημάτων με την ύπαρξη πολλών αποθηκών (Multidepot Vehicle Routing Problem)**

Στο συγκεκριμένο πρόβλημα έχουμε παραπάνω από μία αποθήκες για την εξυπηρέτηση των πελατών. Το πρόβλημα μπορεί να επιλυθεί σε δύο φάσεις στην πρώτη φάση να γίνεται ο καθορισμός των αποθηκών που θα εξυπηρετήσουν τους πελάτες και στην δεύτερη να καθοριστούν τα βέλτιστα δρομολόγια. Ουσιαστικά το παραπάνω πρόβλημα είναι πρόβλημα ομαδοποίησης καθώς στην δεύτερη φάση είναι ένας απλός αλγόριθμος VRP.

- **Πρόβλημα δρομολόγησης οχημάτων χωρίς την επιστροφή στην αποθήκη (Open Vehicle Routing Problem)**

Στα προβλήματα που προαναφέρθηκαν μετά την εκτέλεση του κάθε δρομολογίου το όχημα επέστρεφε στην αποθήκη εντός ενός χρονικού ορίου. Στη συγκεκριμένη κατηγορία προβλημάτων, το φορτηγό δεν επιστρέφει στην αποθήκη, αλλά συνεχίζει να εκτελεί δρομολόγια μέχρι να αδειάσει το φορτίο του. Τέτοιου είδους προβλήματα συναντώνται στην περίπτωση όπου η επιχείρηση δεν έχει στην ιδιοκτησία της τον στόλο των φορτηγών που καλύπτουν τις ανάγκες των πελατών. Έτσι αντί να χαθεί χρόνος στην επιστροφή του φορτηγού στην αποθήκη αξιοποιείται στην εξυπηρέτηση περισσότερων πελατών.

- **Πρόβλημα Δρομολόγησης Οχημάτων με Δύο Είδη Πελατών (Vehicle routing problem with backhauls and line-hauls customers)**

Ένα ακόμη πρόβλημα που αποτελεί μία επέκταση του VRP με συγκεκριμένη χωρητικότητα αποτελεί το πρόβλημα δρομολόγησης οχημάτων με δύο είδη πελατών κατά την διάρκεια μίας διαδρομής. Το πρώτο είδος είναι οι πελάτες που απαιτούν την διανομή κάποιας ποσότητας προϊόντων (line-haul customers) και το δεύτερο απαιτεί μια ποσότητα του προϊόντος να περισυλλεχθεί από τον ίδιο. Οι περιορισμοί που πρέπει να προστεθούν είναι ότι σε κάθε διαδρομή οι πελάτες του δεύτερου τύπου εξυπηρετούνται αφού έχουν εξυπηρετηθεί οι πελάτες του πρώτου τύπου και ότι οι διαδρομές που έχουν πελάτες μόνο του δεύτερου τύπου θα απορρίπτονται.

- **Πρόβλημα Δρομολόγησης Οχημάτων με διανομή και Παραλαβή κατά τη διάρκεια της διαδρομής (Vehicle Routing Problem with pick-up and delivery)**

Στο συγκεκριμένο πρόβλημα ο κάθε πελάτης απαιτεί να του διανεμηθούν προϊόντα από το όχημα και να παραλάβει προϊόντα από αυτόν. Η διαφορά με το προαναφερθέν πρόβλημα είναι ότι ο πελάτης είχε είτε να παραλάβει προϊόντα είτε να διανέμει. Σε αυτή την περίπτωση σε κάθε πελάτη αντιστοιχούν δυο ποσότητες q_i η οποία δείχνει την ποσότητα που πρέπει να διανεμηθεί και p_i η οποία δείχνει την απαιτούμενη ποσότητα παραλαβής. Επίσης αξίζει να αναφερθεί ότι για κάθε πελάτη καθορίζονται δύο κόμβοι, O_i και D_i , που είναι οι κόμβοι από τους οποίους ξεκινούν τα προϊόντα που πρέπει να διανεμηθούν στον πελάτη και καταλήγουν τα προϊόντα που συλλέγονται από τον πελάτη.

- **Στοχαστικό πρόβλημα δρομολόγησης οχημάτων (Stochastic or Probabilistic Vehicle Routing Problem)**

Σε αυτής της κατηγορίας τα προβλήματα ένα όχημα πεπερασμένης χωρητικότητας φεύγει από την αποθήκη και πρέπει να παραδώσει ένα μέρος του φορτίου σε πελάτες των οποίων το πλήθος είναι συγκεκριμένο και η ζήτηση του καθενός ξεχωριστά γίνεται γνωστή τη χρονική στιγμή που θα φτάσει το όχημα. Μία σχεδιασμένη διαδρομή μέσα σε αυτό το πλαίσιο είναι μία διαδρομή από την αποθήκη και περνάει από όλους τους πελάτες

ακριβώς μία φορά. Αυτό καλείται μία εκ των προτέρων διαδρομή και καθορίζει την σειρά εξυπηρέτησης των πελατών αλλά η ακριβής διαδρομή που θα κάνει το όχημα περιλαμβάνει και διαδρομές επιστροφής στην αποθήκη. Τα στοχαστικά σημεία στο πρόβλημα είναι αυτά από τα οποία πραγματοποιούνται διαδρομές επιστροφής.

- **Προβλήματα δρομολόγησης με δυναμική ζήτηση(DynamicVehicleRoutingProblem)**

Στα συγκεκριμένα προβλήματα δεν είναι γνωστή από την αρχή η θέση των πελατών σε αντίθεση με όλες τις προηγούμενες κατηγορίες. Στα προβλήματα με δυναμική ζήτηση μπορεί να προκύψει κάποιος καινούριος πελάτης κατά την διάρκεια εκτέλεσης ενός δρομολογίου. Αυτή η κατηγορία συναντάται κυρίως σε εργασίες που σχετίζονται με την παροχή υπηρεσιών σε πελάτες με τον οδηγό να λαμβάνει εντολές σε πραγματικό χρόνο όπως αυτή του ταξιτζή ή του οδηγού ασθενοφόρου.

3. ΕΥΡΕΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ

3.1 Εισαγωγή στους αλγόριθμους εύρεσης βέλτιστης λύσης

Σε αυτό το κεφάλαιο θα μιλήσουμε για τους ευρετικούς αλγορίθμους. Όσο αυξάνεται το μέγεθος ενός προβλήματος συνδυαστικής βελτιστοποίησης η επίλυση γίνεται ολοένα και πιο δύσκολη, ειδικά αν προσπαθούμε να βρούμε την ολικά βέλτιστη λύση σε λογικό χρόνο. Προκειμένου να επιλύσουμε το συγκεκριμένο πρόβλημα καταφεύγουμε σε διαφορετικές τεχνικές οι οποίες μας οδηγούν σε μία σχεδόν βέλτιστη λύση. Οι τεχνικές που χρησιμοποιούμε για την εύρεση βέλτιστης λύσης είναι οι ευρετικοί αλγόριθμοι. Οι κατηγορίες που χωρίζονται οι ευρετικοί αλγόριθμοι είναι οι παρακάτω:

1. Αλγόριθμοι Απληστίας (Greedy algorithms)
2. Προσεγγιστικοί Αλγόριθμοι (Approximation algorithms)

Αλγόριθμοι τοπικής αναζήτησης(LocalSearchalgorithms)

3.2 Αλγόριθμοι απληστίας (Greedy Algorithms)

Οι αλγόριθμοι απληστίας είναι εύκολο να κατανοηθεί βέβαια δεν υπάρχει σαφής ορισμός του. Στόχος τους είναι να οδηγήσουν σε μία εφικτή λύση του προβλήματος ξεκινώντας από μία μερική λύση και ελέγχεται ποια μπορεί να είναι η επόμενη κίνηση που θα πραγματοποιηθεί χωρίς να παραβιαστεί κανείς περιορισμός του προβλήματος, βέβαια πολλές φορές χρειάζονται πολύ μεγάλο χρόνο διότι ακολουθούν την ιδέα του μικρού αδελφού δηλαδή ότι φαίνεται προς στιγμή καλύτερο ως επιλογή υποθέτουμε ότι είναι και συνολικά η καλύτερη επιλογή.

Σε όλα τα προβλήματα συνδυαστικής βελτιστοποίησης έχουμε η εισόδους. Στα προβλήματα δρομολόγησης οι εισόδοι αντιστοιχούν σε κόμβους και σε αυτού του είδους τα προβλήματα θέλουμε να καθορίσουμε ένα υποσύνολο που να ικανοποιεί τους περιορισμούς του προβλήματος έτσι ώστε να δημιουργηθεί μία εφικτή λύση. Η μέθοδος της απληστίας κατασκευάζει την λύση σε τμήματα όπου σε κάθε τμήμα γίνεται η καλύτερη επιλογή από το σημείο που βρισκόμαστε κάθε χρονική στιγμή δίχως να γίνεται έλεγχος άλλων πιθανών επιλογών οι οποίοι πιθανότατα να έδιναν μία τελική καλύτερη λύση.

Η παραπάνω διαδικασία θα ακολουθηθεί σε αυτή την διπλωματική προκειμένου να βρεθεί μία αρχική λύση , η οποία στη συνέχεια θα βελτιστοποιηθεί από τον μεθευρετικό αλγόριθμο περιορισμένης αναζήτησης (TabuSearch) .

3.3 Προσεγγιστικοί αλγόριθμοι (Approximationalgorithms)

Οι προσεγγιστικοί αλγόριθμοι λειτουργούν με τον ίδιο τρόπο με τους αλγόριθμους απληστίας , αλλά διαφέρουν στο ότι για να λύσουν το πρόβλημα χρησιμοποιούν επιπλέον πληροφορία. Ο αλγόριθμος του Christofides είναι ένα κλασσικό παράδειγμα. . Αυτός ο αλγόριθμος ξεκινάει από ένα ελάχιστο τανύον δέντρο και αφού το μετατρέψει σε κύκλο του Euler στην συνέχεια το μετατρέπει σε ένα κύκλο του Hamilton.

3.4 Αλγόριθμοι τοπικής αναζήτησης

Οι αλγόριθμοι τοπικής αναζήτησης ξεκινούν από μια αρχική λύση και συνεχίζουν την αναζήτηση εκτελώντας διαφορετικές παραλλαγές της αρχικής λύσης. Σε περίπτωση θετικής κίνησης ως προς τη μείωση του κόστους τότε η υπάρχουσα λύση θα αντικατασταθεί από κάποια γειτονική λύση η οποία έχει χαμηλότερο κόστος. Η καινούρια λύση θα χρησιμοποιηθεί ως αφετηρία για τις επόμενες δοκιμές.

Οι βασικές μορφές τοπικής αναζήτησης είναι οι ακόλουθες:

- 2-opt , 3-opt , λ -opt: Αυτή η μέθοδος διαγράφει 2, 3 ή λ ακμές αντίστοιχα και κάνει την επανασύνδεση 2, 3 ή λ μονοπατιών αντίστοιχα με διαφορετικό τρόπο με στόχο να δημιουργηθεί μία νέα διαδρομή με χαμηλότερο κόστος. Όσο μεγαλύτερο το λ τόσο πιο αποτελεσματική αναζήτηση έχουμε.
- (λ -n)-exchange: Η πιο γνωστή περίπτωση αυτής της κατηγορίας αναζήτησης είναι η (1-1) exchange κατά την οποία γίνεται ανταλλαγή μεταξύ δύο κόμβων της ίδιας ή διαφορετικής διαδρομής.
- (1-0)-relocate: Με μία διαγραφή ενός κόμβου από μία διαδρομή και με την προσθήκη του σε μία άλλη επιτυγχάνεται η συγκεκριμένη μέθοδος.

4. ΜΕΘΕΥΡΕΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ

4.1 Εισαγωγή στους μεθευρετικούς αλγορίθμους

Με την χρήση των αλγορίθμων τοπικής αναζήτησης είναι δυνατόν να επιτευχθεί η εύρεση μίας εφικτής και καλής λύσης , το πρόβλημα είναι ότι είναι μεγάλη η πιθανότητα να μην είναι η βέλτιστη λύση και αυτό γιατί η απλή εφαρμογή των συγκεκριμένων αλγορίθμων ελέγχει λύσεις που βρίσκονται στην γειτονιά της αρχικής λύσης . Το συγκεκριμένο πρόβλημα έρχονται να το λύσουν οι μεθευρετικοί αλγόριθμοι οι οποίοι είναι μέθοδοι επίλυσης που συνδυάζουν διαδικασίες τοπικής αναζήτησης και στρατηγικών μέσω των

οποίων αποφεύγεται η παγίδευση σε τοπικά ελάχιστα. Μερικοί βασικοί μεθευρετικοί είναι η Προσομοιωμένη Ανόπτηση (*SimulatedAnnealing*) , η Περιορισμένη αναζήτηση (*TabuSearch*), οι Γενετικοί και Εξελικτικοί αλγόριθμοι (*GeneticandEvolutionaryAlgorithms*) και τα Νευρωνικά δίκτυα(*NeuralNets*).

4.2 Ο αλγόριθμος περιορισμένης αναζήτησης (TabuSearch)

Ο αλγόριθμος περιορισμένης αναζήτησης είναι κατεξοχήν μεθευρετικός αλγόριθμος καθώς συνδυάζει τις δύο διαδικασίες που προαναφέρθηκαν δηλαδή την διαδικασία της τοπικής αναζήτησης και τη στρατηγική υψηλότερου επιπέδου για την αποφυγή παγίδευσης σε τοπικό ελάχιστο. Η μέθοδος παρουσιάστηκε αρχικά από τον Glover το 1986 [**].

Η μέθοδος ξεκίνα από μία αρχική λύση και χρησιμοποιώντας έναν ευρετικό αλγόριθμο για να μετακινηθεί από μία περιοχή σε μία άλλη . Βέβαια υπάρχει ο κίνδυνος να εγκλωβιστεί σε κάποιο τοπικό ελάχιστο και αυτό να έχει ως αποτέλεσμα την ύπαρξη ανεξερεύνητων περιοχών καθώς υπάρχει μεγάλη πιθανότητα να πραγματοποιούνται οι ίδιες εναλλαγές. Το παραπάνω πρόβλημα αντιμετωπίζεται με την χρήση της μνήμης απαγορευμένων κινήσεων. Για να αποφευχθεί η επανάληψη των κινήσεων χρησιμοποιείται μία λίστα η οποία ονομάζεται λίστα περιορισμένων κινήσεων (tabulist). Η ανανέωση της λίστας απαγορευμένων κινήσεων γίνεται δυναμικά και λειτουργεί σύμφωνα με το σύστημα Firstin – Firstout. Οι κινήσεις που θα εισαχθούν στη λίστα είναι απαγορευμένες για συγκεκριμένο αριθμό επαναλήψεων ο οποίος αριθμός καθορίζεται από τον χρήστη ή προσαρμόζεται δυναμικά ανάλογα με τις απαιτήσεις του εκάστοτε προβλήματος. Η διαδικασία αυτή ονομάζεται μνήμη μικρής περιόδου (shorttermmemory). Άλλη μια στρατηγική είναι αυτή της μνήμης μεγάλης διάρκειας (longtermmemory) κατά την οποία αν μία κίνηση εμφανίζει μεγάλη συχνότητα μετά το πέρας ενός αριθμού επαναλήψεων γίνεται προσπάθεια εμπόδισης της επανάληψης της συγκεκριμένης κίνησης. Αυτή η στρατηγική ονομάζεται στρατηγική διάχυσης καθώς βοηθά στην εξερεύνηση χώρων που δεν έχουν ελεγχθεί. Υπάρχει άλλη μία στρατηγική η οποία ονομάζεται στρατηγική εντατικοποίησης (intensificationstrategies) και όταν εφαρμόζεται λέμε ότι ενεργοποιήθηκε το κριτήριο φιλοδοξίας. Στόχο έχει να παραμείνουν οι λύσεις που εμφανίζονται συχνά, στην τελική λύση. Μερικές φορές είναι ευνοϊκό να αγνοήσουμε τους περιορισμούς . Η αγνόηση αυτή

ονομάζεται κριτήριο απενεργοποίησης των περιορισμών και γίνεται όταν μία αλλαγή που βρίσκεται στην λίστα απαγορευμένων κινήσεων γίνει αποδεκτή καθώς το συνολικό κόστος της λύσης που προκύπτει είναι το βέλτιστο. Η μέθοδος της περιορισμένης αναζήτησης δεν συγκλίνει με φυσικό τρόπο και επιλέγουμε εμείς το κριτήριο σταματήματος.

Η tabusearchείναι μία από τις πιο ισχυρές προσδιοριστικές μεθόδους βασισμένη στην τοπική αναζήτηση. Σε πολύ δύσκολα προβλήματα έχουν επιτευχθεί σημαντικές λύσεις βέβαια δεν παύει να είναι δύσκολη στην εφαρμογή της. Το βασικό της προτέρημα είναι ότι χρησιμοποιεί την μνήμη μικρής , μεσαίας και μεγάλης διάρκειας για την καθοδήγηση της τοπικής αναζήτησης. Η διάχυση που χρησιμοποιεί η tabusearchδιαφέρει από την τυχαιοποίηση η οποία θα αναφερθεί στον αλγόριθμο της προσομοιωμένης απόπτωσης καθώς δεν βασίζεται σε μόνο στην διάχυση μιας συλλογής λύσεων αλλά και στην διάχυση ακολουθιών λύσεων.

Ο ψευδοκώδικας του αλγορίθμου είναι ο ακόλουθος.

Κατασκευή μίας αρχικής λύσης S_0

$S^*=S_0$! Αρχικοποίηση της βέλτιστης λύσης

$k=0$

$t(k)=0$

repeat

 επιλογή της γειτονιάς $s \in N(S_0)$ για την οποία το $\delta_c(S)$ ελαχιστοποιείται

 if $m(S) \neq t(k)$ ή κάποιο κριτήριο απενεργοποίησης των περιορισμών ικανοποιείται

$S_0=S$

 If $c(S) < c(S^*)$

$S^*=S$

 Endif

 Update $k, t(k)$

 Endif

Until κάποιο κριτήριο σταματήματος ικανοποιηθεί

Return S^*

4.3 Ο αλγόριθμος Προσομοιωμένης Ανόπτωσης (SimulatedAnnealing)

Ένας μεθευρετικός αλγόριθμος όπου δεν κάνει κινήσεις τέτοιες ώστε να βρίσκει μεγάλα κόστη είναι σίγουρα μη πλήρης επειδή μπορεί να παγιδευτεί σε ένα τοπικό ελάχιστο. Αντίθετα ένας αλγόριθμος που κάνει κινήσεις οι οποίες βασίζονται στην τύχη και η κάθε κίνηση διαλέγεται ομοιόμορφα στην τύχη είναι πλήρης αλλά έχει χαμηλή απόδοση. Η προσομοιωμένη ανόπτωση προέρχεται από τις φυσικές διαδικασίες δημιουργίας κρυσταλλικών δομών στη φύση. Ο αλγόριθμος ξεκινά από μία αρχική λύση, και σε και κάθε επανάληψη γίνεται μία τυχαία κίνηση. Εάν η κίνηση βελτιώνει το αποτέλεσμα τότε είναι αποδεκτή. Εάν όμως δεν βελτιώνει το αποτέλεσμα τότε γίνεται δεκτή με μία πιθανότητα που προέρχεται από τους νόμους της θερμοδυναμικής και πιο συγκεκριμένα από τον τύπο $p(\delta) = e^{-\Delta/t}$, όπου t μία παράμετρος που ονομάζεται θερμοκρασία και δ η ποσότητα ενέργειας. Ο παραπάνω τύπος δείχνει ότι η προσομοιωμένη ανόπτωση αποδέχεται μία μικρή αύξηση στην αντικειμενική συνάρτηση ελέγχοντας την πιθανότητα αποδοχής.

5. ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

5.1 Συνοπτική περιγραφή του προβλήματος

Στην παρούσα διπλωματική εργασία ασχολούμαστε με την επίλυση του προβλήματος δρομολόγησης οχημάτων περιορισμένης χωρητικότητας με χωριστή εξυπηρέτηση (SplitVRP). Σκοπός του προβλήματος, όπως έχει αναφερθεί σε προηγούμενα κεφάλαια, είναι ο καθορισμός των βέλτιστων διαδρομών στόλου οχημάτων που ξεκινάνε από την αποθήκη και εξυπηρετούν όλους τους πελάτες, ώστε να ελαχιστοποιηθεί το συνολικό κόστος. Το όχημα έχει πεπερασμένη χωρητικότητα, πεπερασμένη είναι και η ζήτηση κάθε πελάτη η οποία πρέπει να καλυφθεί. Θεωρούμε ότι το κόστος της διαδρομής προκύπτει αθροιστικά από τις επιμέρους αποστάσεις μεταξύ των πελατών που ανήκουν στην διαδρομή.

Οι περιορισμοί που διέπουν το πρόβλημα είναι οι ακόλουθοι:

1. Το σύνολο της ζήτησης που πρέπει να καλυφθεί σε κάθε διαδρομή, δεν θα πρέπει να ξεπερνάει την χωρητικότητα του οχήματος.
2. Το όχημα πρέπει να ξεκινά το δρομολόγιο του από την αποθήκη και να τερματίζει σε αυτή.

Το παραπάνω πρόβλημα θα λυθεί σε δύο φάσεις καθώς στην πρώτη φάση θα χρησιμοποιηθεί ο αλγόριθμος της απληστίας , για την εύρεση μιας εφικτής λύσης και στην δεύτερη φάση θα γίνει η εύρεση της βέλτιστης λύσης μέσω του μεθευρετικού αλγόριθμου TabuSearch. Οι αλγόριθμοι επιλύθηκαν στο προγραμματιστικό περιβάλλον της Matlab.

5.2 Μοντελοποίηση του προβλήματος

Βάση της παραπάνω περιγραφής τα δεδομένα που θα χρειαστούν για την επίλυση του προβλήματος είναι το πλήθος των πελατών , η ζήτηση του καθένα , η γεωγραφική τους θέση καθώς και η χωρητικότητα του οχήματος. Τα δεδομένα τα εισάγουμε στο περιβάλλον της Matlab μέσω φύλλων excel και διαβάστηκαν με την εντολή `importdata`.

<div> <div>Αρχείο</div> <div>Κεντρική</div> <div>Εισαγωγή</div> <div>Διάταξη σελίδας</div> <div>Τύποι</div> <div>Δεδομένα</div> <div>Αναθεώρηση</div> </div>												
H1					1							
	A	B	C	D	E	F	G	H	I	J	K	L
1	51		1	30	40				1	0		
2	1		2	37	52				2	7		
3	160		3	49	49				3	30		
4	200		4	52	64				4	16		
5	10		5	20	26				5	9		
6			6	40	30				6	21		
7			7	21	47				7	15		
8			8	17	63				8	19		
9			9	31	62				9	23		
10			10	52	33				10	11		
11			11	51	21				11	5		
12			12	42	41				12	19		
13			13	31	32				13	29		
14			14	5	25				14	23		
15			15	12	42				15	21		
16			16	36	16				16	10		
17			17	52	41				17	15		
18			18	27	23				18	3		
19			19	17	33				19	41		
20			20	13	13				20	9		
21			21	57	58				21	28		
22			22	62	42				22	8		
23			23	42	57				23	8		
24			24	16	57				24	16		
25			25	8	52				25	10		
26			26	7	38				26	28		
27			27	27	68				27	7		
28			28	30	48				28	15		
29			29	43	67				29	14		
30			30	58	48				30	6		
31			31	58	27				31	19		
32			32	37	69				32	11		
33			33	38	46				33	12		
34			34	46	10				34	23		
35			35	61	33				35	26		
36			36	62	63				36	17		
37			37	63	69				37	6		
38			38	32	22				38	9		
39			39	45	35				39	15		
40			40	59	15				40	14		
41			41	5	6				41	7		
42			42	10	17				42	27		
43			43	21	10				43	13		

Στη στήλη Α βρίσκονται τα στοιχεία του προβλήματος . Στο πρώτο κελί βρίσκεται το πλήθος των πελατών (numc), στο τρίτο κελί είναι η μέγιστη χωρητικότητα (Q)του οχήματος, στο τέταρτο κελί φαίνεται ο μέγιστος χρόνος (routetime)του κάθε δρομολογίου και στο τέταρτο κελί απεικονίζεται ο χρόνος εξυπηρέτησης (servicetime).

Στις στήλες Dκαι E αποτυπώνονται οι συντεταγμένες που καθορίζουν τη γεωγραφική θέση των πελατών.

Στη στήλη Ιβρίσκεται η ζήτηση του κάθε πελάτη (demand).

Οπότε στα συγκεκριμένα δεδομένα έχουμε:

Στοιχεία Οχήματος:

- Μέγιστη Χωρητικότητα $Q=160$
- Μέγιστη Χρονική διάρκεια Δρομολογίου $routetime=200$

Στοιχεία Πελατών:

<u>i</u>	<u>/</u>	<u>x</u>	<u>/</u>	<u>y</u>	<u>/</u>	<u>d</u>
1		30		40		0 (Αποθήκη)
2		37		52		7 (1 ^{ος} πελάτης)
3		49		49		30 (2 ^{ος} πελάτης)
.						
.						
.						
51		56		37		10 (50 ^{ος} πελάτης)

Όπου

i= αριθμός πελάτη

x= συντεταγμένες στον άξονα των x

y= συντεταγμένες στον άξονα των y

d= ζήτηση κάθε πελάτη

Έτσι αποθηκεύουμε τα δεδομένα σε αντίστοιχες μεταβλητές:

Numc→ Αριθμός Πελατών

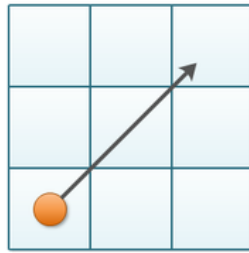
Q → Χωρητικότητα του οχήματος

Cords → Πίνακας Συντεταγμένων

d → Διάνυσμα με τη ζήτηση κάθε πελάτη

Το επόμενο βήμα είναι η δημιουργία ενός πίνακα που περιέχει τις αποστάσεις μεταξύ όλων των κόμβων και συμβολίζεται $cost(i,j)$. Η απόσταση μεταξύ δύο κόμβων ισούται με την Ευκλείδεια απόσταση. Ο πίνακας που δημιουργείται είναι συμμετρικός το οποίο σημαίνει ότι η απόσταση μεταξύ του i και του j είναι ίση με αυτήν από τον j στον i. Μία ακόμη επεξεργασία που γίνεται στον πίνακα των αποστάσεων είναι ότι απειρίζεται η διαγώνιος του , καθώς πρέπει να διατηρείται η ροή.

Euclidean Distance



$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Τώρα έχουμε ότι χρειαζόμαστε έτσι ώστε να ξεκινήσει η εύρεση της αρχικής εφικτής λύσης.

5.3 Εύρεση αρχικής εφικτής λύσης μέσω του Αλγόριθμου Απληστίας

Έχοντας αποθηκευμένα τα παραπάνω δεδομένα , το επόμενο βήμα είναι να ξεκινήσουμε την επίλυση του προβλήματος. Αρχικά δηλώνονται οι μεταβλητές που δηλώνουν την αρχική κατάσταση που επικρατεί πριν ξεκινήσει η διαδικασία επίλυσης του προβλήματος.

Ορίζουμε τις μεταβλητές.

Diadromi → Διάνυσμα Δρομολογίων

D → Διάνυσμα με συνολικό φορτίο κάθε διαδρομής

Plithos_dromologion → Πλήθος Δρομολογίων

staseis → Διάνυσμα Στάσεων κάθε Δρομολογίου

Στη αρχή ο αλγόριθμος εισάγει τον κοντινότερο πελάτη στην αποθήκη και απειρίζει την ζήτηση του αντίστοιχου πελάτη. Στη συνέχεια διαμορφώνονται τα δρομολόγια έτσι ώστε ,σε περίπτωση που ο τελευταίος πελάτης ενός δρομολογίου δεν ικανοποιείται πλήρως, να εξυπηρετείται με όση ποσότητα έχει απομείνει στο όχημα. Σε αυτή την περίπτωση η ζήτηση του πελάτη που δέχεται χωριστή εξυπηρέτηση γίνεται ίση με την διαφορά της συνολικής του

ζήτησης μείον την εναπομείνασα ποσότητα του οχήματος. Στη συνέχεια ο συγκεκριμένος πελάτης θα ξαναδεχθεί επίσκεψη από το όχημα μέχρις ότου να ικανοποιηθεί πλήρως. Ο αλγόριθμος συνεχίζει μέχρι να μην υπάρχουν πελάτες που δεν έχουν εξυπηρετηθεί.

5.4 Εύρεση της βέλτιστης λύσης με τη χρήση του αλγόριθμου TabuSearch

Η λύση που βρέθηκε προηγουμένως είναι εφικτή, αλλά δεν είναι βέλτιστη. Για να βελτιστοποιηθεί το συνολικό κόστος και για να μειωθεί ο αριθμός των διαδρομών γίνεται χρήση του αλγορίθμου περιορισμένης αναζήτησης (TabuSearch). Η μέθοδος αυτή παρότι είναι αρκετά περίπλοκη και δύσκολο να εφαρμοστεί εξ' ολοκλήρου, αποδίδει, δίνοντας αρκετά καλά αποτελέσματα. Η μέθοδος στηρίζεται σε εφαρμογές της τοπικής αναζήτησης, όπως είναι οι ανταλλαγές κόμβων μεταξύ διαφορετικών διαδρομών, οι εναλλαγές κόμβων ίδιας διαδρομής.

.Το κύριο χαρακτηριστικό της περιορισμένης αναζήτησης είναι ότι, χρησιμοποιεί μνήμη για την αποφυγή επανάληψης ίδιων κινήσεων. Για να εφαρμοστεί η μέθοδος απαιτείται η ύπαρξη μίας αρχικής εφικτής λύσης, την οποία καλούμαστε να βελτιώσουμε με διάφορες στρατηγικές.

Σε πρώτο βήμα ορίζουμε τις μεταβλητές από τα υπάρχοντα δεδομένα, που θα χρειαστούν στη συνέχεια. Επιπλέον θα χρειαστεί μια ουρά (FIFO) `tabu_list`, που θα αρχικοποιηθεί με μηδενικά και θα είναι διαστάσεων `(kmegethos,2)`. Αυτός είναι ο πίνακας απαγορευμένων κινήσεων και όταν πραγματοποιείται κάποια αλλαγή, τα στοιχεία που συμμετέχουν στην αλλαγή θα εισέρχονται σε αυτόν. Έτσι θα αποκλειστεί η περίπτωση να ξανά πραγματοποιηθεί αυτή η εναλλαγή για έναν αριθμό `kmegethos` επαναλήψεων. Έτσι θα αποκλειστεί το ενδεχόμενο να οδηγηθούμε σε κύκλους γύρω από μία λύση.

Στον αλγόριθμο χρησιμοποιήθηκαν τρεις μέθοδοι εμπνευσμένες από την τοπική αναζήτηση για την βελτίωση του αποτελέσματος. Η πρώτη είναι η ανταλλαγή κόμβων μεταξύ διαφορετικών διαδρομών, η δεύτερη είναι η εφαρμογή 2-opt μετά από κάθε ανταλλαγή κόμβων. Τέλος η τρίτη αποτελεί την στρατηγική της διάχυσης κατά την οποία ελέγχει ποιο

τόξο έχει την μεγαλύτερη συχνότητα και στη συνέχεια το σπάει με την χρήση 2-opt. Παρακάτω θα αναλυθεί ο τρόπος λειτουργίας της κάθε μίας και τι αποτέλεσμα μπορεί να μας αποφέρει.

Στο πρώτο τμήμα του αλγορίθμου θα εξεταστεί αν μπορεί να βελτιωθεί η λύση μέσω της ανταλλαγής κόμβων μεταξύ των διαδρομών. Η βασική ιδέα της μεθόδου αυτής είναι να αλλάξει την σειρά εξυπηρέτησης ανάμεσα σε δύο πελάτες και να ελεγχθεί αν μειώνεται η συνολική απόσταση δηλαδή το κόστος. Εφόσον ικανοποιούνται οι περιορισμοί του προβλήματος. Η πρώτη διαδικασία που εκτελείται είναι ο υπολογισμός του κόστους κάθε κόμβου. Ο κόμβος που έχει το μεγαλύτερο άθροισμα σημαίνει ότι είναι και ο πιο απομακρυσμένος άρα και ο πρώτος υποψήφιος για αλλαγή.

Ο δεύτερος υποψήφιος επιλέγεται μέσω της εντολής rand. Γίνεται ο υπολογισμός του κόστους των νέων διαδρομών και ο έλεγχος αν ικανοποιείται ο περιορισμός της χωρητικότητας. Αν είναι εφικτή η νέα λύση και αν το κόστος μετά την αλλαγή είναι μικρότερο ή ελαφρώς μεγαλύτερο από μέχρι τότε βέλτιστο ελέγχει αν η αλλαγή βρίσκεται στην λίστα k. Αν η αλλαγή βρίσκεται στην λίστα k και το κόστος που έχει προκύψει είναι βέλτιστο τότε ενεργοποιείται το κριτήριο της φιλοδοξίας αλλιώς η αλλαγή απορρίπτεται.

Στο επόμενο στάδιο του αλγορίθμου επιτελείται ο 2-opt στις δύο διαδρομές με το μεγαλύτερο κόστος. Αρχικά επιλέγεται το τόξο με το μεγαλύτερο κόστος στην διαδρομή, έπειτα χωρίζοντας την διαδρομή σε δύο τμήματα γίνεται αναζήτηση εκατέρωθεν του κάθε κόμβου και γίνεται ανεύρεση του τόξου με το μικρότερο κόστος, τότε οι κόμβοι που εφαρμόζεται ο 2-opt είναι ο κόμβος του τόξου με το μεγαλύτερο κόστος και ο δεύτερος κόμβος είναι ο κόμβος που βρέθηκε το μικρότερο, ως προς απόσταση, τόξο.

Πιο αναλυτικά έστω ότι έχουμε την παρακάτω διαδρομή:

1 47 13 48 5 18 38 16 45 43 1

Γίνεται αναζήτηση του τόξου με το μεγαλύτερο κόστος, δηλαδή επιλέγεται το τόξο με την μεγαλύτερη η μεταξύ τους απόσταση. Αν υποθέσουμε ότι το 18 38 είναι το τόξο με το μεγαλύτερο τόξο τότε γίνεται προσπάθεια να χωριστεί και να βρεθεί μια καλύτερη διαδρομή.

Έτσι συγκρίνουμε τις αποστάσεις του 18 με το 16 , του 18 με το 45 και του 18 με το 43. Η ίδια διαδικασία επαναλαμβάνεται και για τον κόμβο 38. Τότε επιλέγεται το τόξο με το μικρότερο κόστος έτσι εφαρμόζεται ο 2-ορτ μεταξύ του κόμβου που ανήκει στο μεγαλύτερο τόξο , το οποίο βρέθηκε στην πρώτη αναζήτηση και δεν υπάρχει στην δεύτερη αναζήτηση. Για παράδειγμα αν επιλεγθεί το 18 45 με το μικρότερο κόστος τότε τα άκρα της διαδικασίας 2ορτείνονται το 38 και 45 , άρα η νέα διαδρομή θα γίνει

1 47 13 48 5 18 45 16 38 43 1

Το τρίτο στάδιο είναι αυτό της διάχυσης και σε αυτό μέσω του πίνακα μεγάλης διάρκειας βρίσκεται το τόξο με την μεγαλύτερη συχνότητα και εφαρμόζεται 2ορτ σύμφωνα με την παραπάνω διαδικασία έτσι ώστε να σπάσει και να μελετηθούν ανεξερεύνητες περιοχές.

6. ΑΝΑΛΥΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

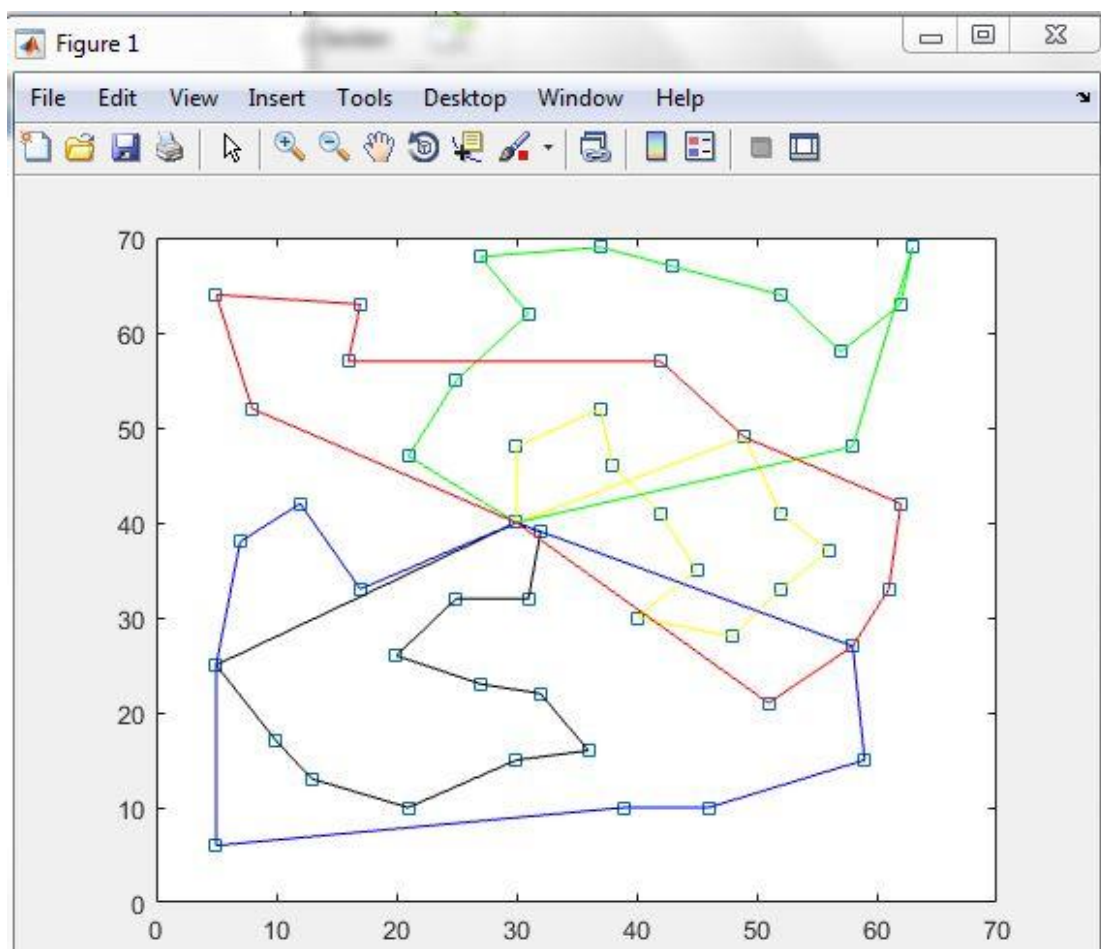
6.1 Περιγραφή και ανάλυση των αποτελεσμάτων

Ο προτεινόμενος αλγόριθμος υλοποιήθηκε στο προγραμματιστικό περιβάλλον της Matlab και δοκιμάζεται σε δεδομένα τα οποία περιέχουν από 50 έως 200 πελάτες. Το κάθε πρόβλημα επιλύθηκε 10 φορές και κρατήθηκαν τα καλύτερα αποτελέσματα. Κάθε βελτιωτικό τμήμα είχε ως κριτήριο τερματισμού της 50.000 επαναλήψεις. Ακόμα το όριο της tabulisteπρινε δύο τιμές ανάλογα με το μέγεθος του προβλήματος , 10 θέσεις για μικρού μεγέθους και 13 για μεγάλου μεγέθους .

6.2 Αποτελέσματα των αλγορίθμων

Αποτέλεσμα δρομολογίων για το 1^ο πρόβλημα:

1	1	1	1	1
47	28	7	19	23
13	2	49	15	3
48	33	9	26	22
5	12	27	14	35
18	39	32	41	31
38	6	29	46	11
16	50	4	34	24
45	10	21	40	8
43	51	36	31	44
20	17	37	1	25
42	3	30		1
14	1	1		
1				



Με συνολικό βέλτιστο κόστος 680,5

Αποτελέσματα δρομολογίων στο 2^ο πρόβλημα:

1	1	1	1	1	1	1	1	1	1
7	68	27	53	31	3	17	6	11	50
69	35	5	28	75	2	4	38	39	25
76	47	46	14	29	44	45	21	60	19
52	9	30	55	63	42	33	71	12	51
18	36	49	20	74	43	10	61	67	56
41	8	48	15	34	65	40	72	66	24
13	54	22	12	64	23	73	37	32	57
27	15	75	1	17	62	59	70	26	1
1	1	1		1	1	11	62	51	
						1	16	1	
							58		
							66		
							1		

Με συνολικό βέλτιστο κόστος 1068,8.

Αποτελέσματα δρομολογίων στο 3^ο πρόβλημα

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

54	28	90	14	53	70	32	17
59	29	7	88	19	2	89	47
41	27	95	3	61	51	8	71
22	13	96	58	84	77	63	66
74	81	98	16	85	30	11	1
73	69	93	44	6	25	91	
68	78	60	43	18	55	33	
23	4	94	101	46	56	64	
76	80	97	38	9	26	12	
57	34	100	99	83	5	20	
40	82	86	62	49	42	50	
24	10	92	87	48	75	65	
75	1	101	45	37	79	31	
1		1	15	50	35	67	
			39	1	36	71	
			17		72	1	
			1		10		
					52		
					21		
					31		
					1		

Με συνολικό βέλτιστο κόστος 1139,8.

Αποτελέσματα δρομολογίων για το 4^ο πρόβλημα:

1	1	1	1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---

54	28	113	59	14	77	19	51	107	94	81
106	147	90	41	138	117	61	103	8	86	69
27	53	148	22	88	78	119	34	83	92	151
150	128	7	74	145	4	6	82	49	62	135
110	32	95	73	58	80	85	52	125	87	25
13	89	96	75	116	130	18	123	48	141	30
139	149	118	134	3	79	114	31	37	39	122
29	63	98	23	146	35	17	21	144	120	55
112	11	93	76	42	121	142	129	50	84	131
133	109	60	57	16	10	45	132	20	115	56
70	91	100	140	44	104	15	67	124	9	26
2	127	97	40	143	72	120	66	108	126	5
102	64	105	24	43	137	1	136	1	46	111
71	33	94	1	101	36		122		1	24
31	132	1		38	136		1			68
1	1			99	1					1
				86						
				1						

Με συνολικό βέλτιστο κόστος 1140,5 .

Αποτελέσματα δρομολογίων για το 5^ο πρόβλημα:

1	1	1	1	1	1	1	1	1	1	1	1	1
157	54	28	59	153	155	53	167	97	51	61	199	84
113	106	147	41	138	185	128	19	100	103	119	180	200
90	27	168	181	14	77	191	154	60	159	6	55	115
148	150	70	22	88	197	32	107	105	69	85	131	9
7	196	133	74	145	117	89	195	94	81	174	166	175
184	110	163	73	58	78	149	8	86	178	62	56	126
95	13	102	75	179	4	63	183	194	151	17	26	46
96	139	71	172	3	80	160	124	92	164	142	171	18
118	29	31	23	116	130	11	20	101	135	192	188	114
98	112	123	134	146	186	109	108	38	25	45	140	87
93	177	52	76	42	34	190	176	99	30	120	40	141
60	2	10	57	198	158	1	12	152	122	193	187	39
1	133	121	187	111	103		127	173	170	15	24	1
	1	82	1	199	1		64	43	79	1	68	
		34		1			1	143	35		1	
		1						1	1			

Με συνολικό βέλτιστο κόστος 1637,7

Αποτελέσματα δρομολογίων για το 6^ο πρόβλημα:.

1	1	1	1	1
47	28	7	19	23
13	2	49	15	3
48	33	9	26	22
5	12	27	14	35
18	39	32	41	31
38	6	29	46	11
16	50	4	34	24
45	10	21	40	8
43	51	36	31	44
20	17	37	1	25
42	3	30		1
14	1	1		
1				

Με συνολικό βέλτιστο κόστος 682,36 .

Τα αποτελέσματα των δρομολογίων στο 7^ο πρόβλημα είναι:

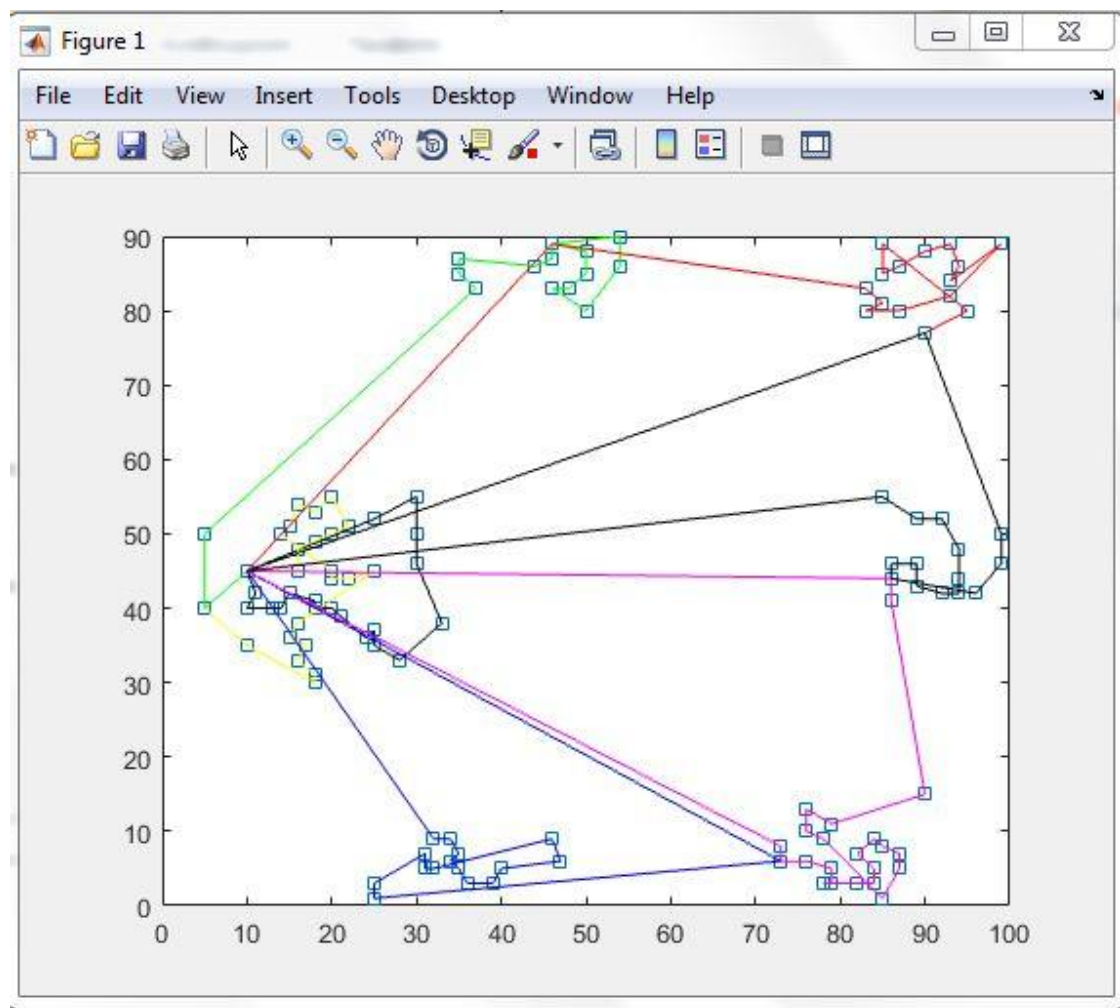
1	1	1	1	1	1	1	1	1	1
7	68	27	53	31	3	17	6	11	50
69	35	5	28	75	2	4	38	39	25
76	47	46	14	29	44	45	21	66	19
52	9	30	55	63	42	33	71	67	51
18	36	49	20	74	43	10	61	60	56
41	8	48	15	34	65	40	72	12	24
13	54	22	67	64	23	73	37	32	57
27	15	75	1	17	62	59	70	26	1
1	1	1		1	1	11	62	51	
						1	16	1	
							58		
							12		
							1		

Με συνολικό βέλτιστο κόστος 1076,4 .

Τα αποτελέσματα των δρομολογίων στο 8^ο πρόβλημα είναι:

1	1	1	1	1	1	1
89	96	120	7	76	18	41
83	103	121	8	54	17	44

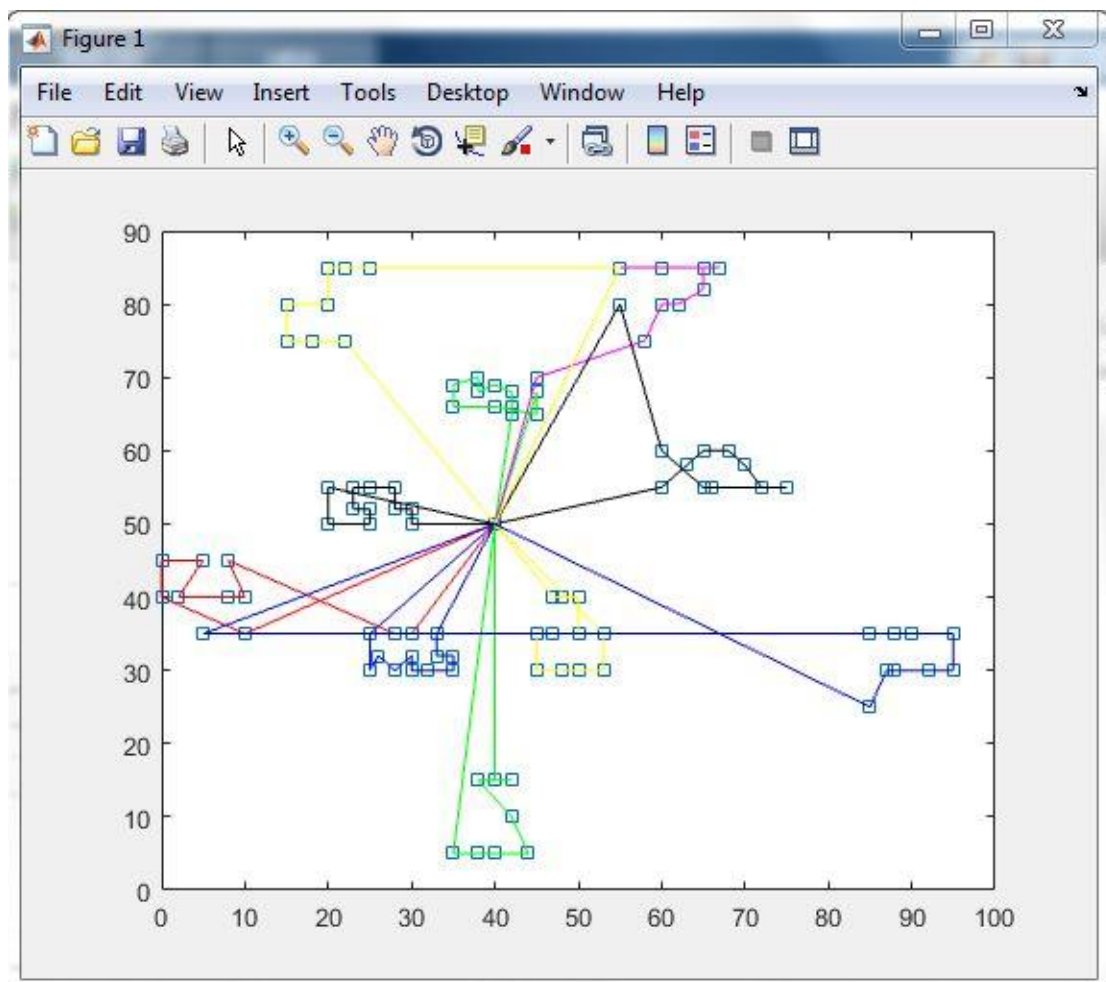
112	102	68	10	55	20	46
87	100	70	11	53	26	49
88	101	71	12	58	25	48
93	104	72	16	62	23	47
90	105	75	15	67	28	45
92	108	73	14	63	31	42
91	107	79	13	65	32	43
19	106	78	9	64	29	40
115	97	77	5	61	33	39
119	94	74	4	59	36	50
109	95	69	6	56	37	51
110	98	80	3	57	35	52
116	86	81	2	66	34	60
111	113	76	17	60	24	1
99	85	1	1	1	21	0
117	118	0	0	0	22	0
1	114	0	0	0	27	0
0	84	0	0	0	30	0
0	82	0	0	0	38	0
0	120	0	0	0	39	0
0	1	0	0	0	1	0



Με συνολικό βέλτιστο κόστος 1292,3 .

Τα αποτελέσματα δρομολογίων στο 9^ο πρόβλημα :

1	1	1	1	1	1	1	1	1	1
21	68	6	44	48	3	91	14	58	36
22	66	4	43	50	99	90	18	56	32
23	64	5	42	35	97	89	19	60	82
24	63	7	41	33	96	86	20	55	79
27	67	9	45	34	95	85	16	54	77
29	70	10	46	38	94	84	17	57	72
28	69	12	47	37	93	83	15	59	71
26	65	11	49	40	98	87	13	61	74
25	62	8	51	39	101	88	101	1	78
30	73	76	52	32	1	92	1		80
31	75	2	53	1		100			81
1	1	1	1			1			1



Με συνολικό βέλτιστο κόστος 998,88 .

7. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στο προηγούμενο κεφάλαιο παρουσιάστηκαν τα αποτελέσματα των περισσότερων από τους αλγορίθμους που έτρεξαν. Σε αυτό το κεφάλαιο θα γίνει μία στατιστική αξιολόγηση των αποτελεσμάτων που προέκυψαν. Αναλυτικά παρουσιάζονται στον επόμενο πίνακα.

	1	2	3	4
P01	690	665,6	672	9,8648
P02	1081,1	1068,8	1074,933	16,20222
P03	1205	1139,8	1174,289	441,7832
P04	1323,3	1140,1	1150,3	39,09
P05	1665	1637,,7	1654,2	64,71
P06	690	673,78	681,59	7,89
P07	1080,6	1076,4	1079,633	1,475556
P08	1205	1179,4	1194,289	64,06321
P09	1353,2	1290,9	1343,956	363,7336
P10	1665	1637,7	1649,478	26,25951
P11	1322,8	1292,3	1303,99	66,9121
P12	1213,2	1203,1	1210,9	19,14
P13	1558,1	1521,8	1530,644	63,13136
P14	1022,3	996,29	1003,317	14,65004

Στήλη 1:Κόστος split-vrp

Στήλη 2:Βέλτιστο κόστος

Στήλη 3: Μέσος Όρος βέλτιστων κοστών

Στήλη 4: Διακυμάνσεις τιμών κόστους

Ανάλυση αποτελεσμάτωνκαι συμπεράσματα

Από τον παραπάνω πίνακα οδηγούμαστε σε κάποιες στατιστικές παρατηρήσεις , βάση των οποίων μπορεί να εκτιμηθεί η ποιότητα των αποτελεσμάτων και εν συνεχεία των αλγορίθμων. Οι κυριότερες στατιστικές παρατηρήσεις είναι :

- Σε όλες τις περιπτώσεις έχουμε μείωση του κόστους αφού χρησιμοποιηθεί ο μεθευρετικός αλγόριθμος. Βέβαια η μείωση του κόστους δεν είναι όση μπορεί να προέλθει από την χρήση του TabuSearch. Διαπιστώνεται έτσι ότι έμειναν πολλές ανεξερεύνητες περιοχές. Αυτό οφείλεται στο γεγονός ότι δεν υπήρχε σωστή κατεύθυνση στην εξερεύνηση των λύσεων. Πιο αναλυτικά η τυχαία επιλογή του δεύτερου κόμβου στον 1-1 exchange έριξε κατά πολύ την απόδοση του αλγορίθμου.
- Στα προβλήματα όπου οι πελάτες βρίσκονται μακριά από την αποθήκη ο αλγόριθμος έχει καλύτερη επίδοση σε σχέση με τα άλλα προβλήματα.

Βιβλιογραφία

1. Αθανάσιος Μυγδαλάς, Ιωάννης Μαρινάκης, Αθανασία Μαυρομάτη. Σημειώσεις μαθηματικής Διαχείρισης Εφοδιαστικής αλυσίδας.
2. Ιωάννης Μαρινάκης - Μυγδαλάς Αθανάσιος: 'Σχεδιασμός Και Βελτιστοποίηση Της Εφοδιαστικής Αλυσίδας'
3. Βλαχαβάς Ιωάννης, Κεφαλάς Πέτρος, Βασιλειάδης Νικόλαος, Κόκκορας Φώτης, Σακελλαρίου Ηλίας 'Τεχνητή Νοημοσύνη', εκδόσεις Πανεπιστημίου Μακεδονίας
4. Stuart Russell and Peter Norvig , 'Artificial Intelligence: A modern Approach'
5. C.Archetti and M.G. Speranza, 2008, 'The Split Delivery Vehicle Routing Problem: A survey'
6. Marcos Melo Silva, 2015, 'An iterated local search heuristic for the split delivery vehicle'
7. C. Archetti and A. Hertz, 2003, 'A tabu search algorithm for the split delivery vehicle routing problem'
8. Marshall Fischer and RamchandrauJaikumar, 1986, 'A generalized Assignment Heuristic for vehicle routing Problems'
9. Glover F. , Laguna M. , 1995, 'Tabu Search Modern heuristic Techniques for Combinatorial Problems'
10. Fred Glover (1990). "Tabu Search: A Tutorial". Interfaces.
11. M. Laguna & R. Marti (2003). Scatter Search: Methodology and Implementations in C. Kluwer Academic Publishers, Boston.