



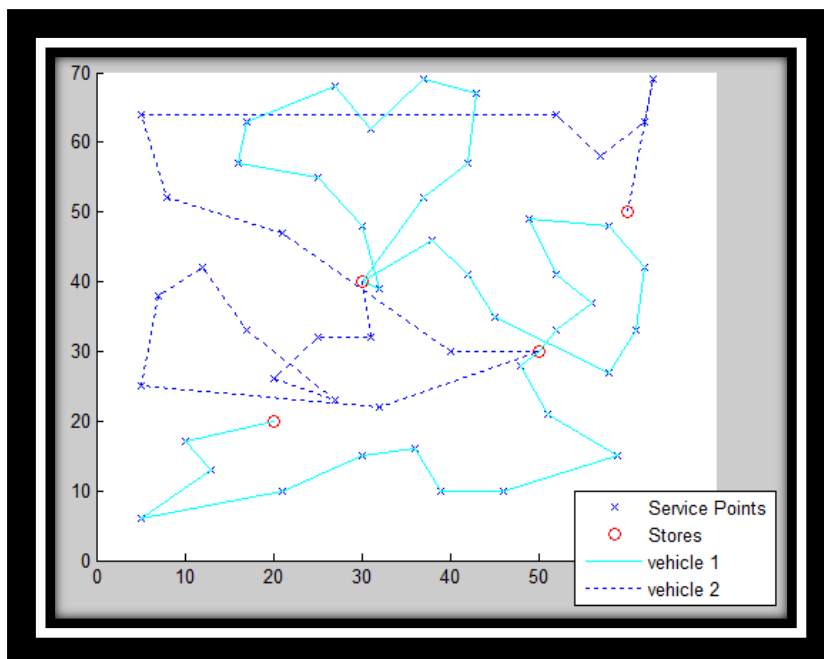
Πολυτεχνείο Κρήτης

Τμήμα Μηχανικών Παραγωγής και Διοίκησης

Τομέας Επιχειρησιακής Έρευνας

Διπλωματική Εργασία

Το πρόβλημα της δρομολόγησης οχημάτων με πολλαπλές αποθήκες (MDVRP), λυμένο με τη μέθοδο της βελτιστοποίησης με νοημοσύνη σμήνους σωματιδίων (PSO)



Αλεβίζος Μιχαήλ

Επιβλέπων καθηγητής: Μαρινάκης Ιωάννης

Χανιά, Ιανουάριος 2012

Περιεχόμενα

Ευχαριστίες.....	4
1. Εισαγωγή	5
2. Γενικά στοιχεία για το VRP, το PSO και τον 2-opt	6
2.1 Το πρόβλημα της δρομολόγησης των οχημάτων (VRP).....	6
2.1.1 Περιγραφή του προβλήματος της δρομολόγησης των οχημάτων (VRP)	6
2.1.2 Το πρόβλημα της δρομολόγησης των οχημάτων με ύπαρξη πολλαπλών αποθηκών (MDVRP)	7
2.2 Βελτιστοποίηση με νοημοσύνη σμήνους σωματιδίων (PSO)	9
2.3 Αλγόριθμος τοπικής αναζήτησης 2-opt	13
3. Περιγραφή του προβλήματος και μοντελοποίηση	16
3.1 Μοντελοποίηση του προβλήματος	16
3.2 Μορφή των δεδομένων	19
3.3 Βοηθητικοί πίνακες	23
4. Επιλογή μεταβλητών κατά την υλοποίηση του αλγορίθμου	28
4.1 Επιλογή των μεταβλητών για το PSO	28
4.1.1 Επιλογή του μεγέθους του σμήνους και του μέγιστου αριθμού επαναλήψεων ..	28
4.1.2 Επιλογή των μεταβλητών επιτάχυνσης $c1, c2$	29
4.1.3 Επιλογή των τυχαίων μεταβλητών $r1, r2$	29
4.1.4 Επιλογή του δείκτη αδράνειας w	29
4.2 Επιλογή της καταλληλότερης μεθόδου 2-opt με χρήση τυχαίων αρχικών λύσεων στο PSO	29
4.2.1 2-opt με χρήση των δύο μεγαλύτερων τόξων	30
4.2.2 2-opt με χρήση του μεγαλύτερου τόξου και ενός τυχαίου	30
4.2.3 2-opt με χρήση δύο τυχαίων τόξων	30
4.3 Επιλογή της καταλληλότερης μεθόδου 2-opt με χρήση βελτιωμένων αρχικών λύσεων στο PSO	31
5. Υλοποίηση και εκτέλεση του αλγορίθμου και παρουσίαση των αποτελεσμάτων.....	32
5.1 Ένα ολοκληρωμένο παράδειγμα.....	32
5.2 Εκτέλεση αλγορίθμου με τυχαίες αρχικές λύσεις.....	35
5.3 Εκτέλεση του αλγορίθμου με καλύτερες αρχικές λύσεις	37
5.4 Συγκεντρωτικός πίνακας και σχόλια	40
6. Συμπεράσματα	41

Βιβλιογραφία	44
--------------------	----

Ευχαριστίες

Σε αυτό το σημείο θα ήθελα να εκφράσω τις ευχαριστίες μου στον επιβλέπων καθηγητή μου κ. Ιωάννη Μαρινάκη για την αμέριστη υποστήριξή του σε όλα τα μέρη και βήματα της παρούσας εργασίας.

1. Εισαγωγή

Στόχος αυτής της διπλωματικής εργασίας είναι να εξετάσουμε το πρόβλημα της δρομολόγησης οχημάτων με πολλαπλές αποθήκες. Ειδικότερα, θα γίνει προσπάθεια επίλυσης του προβλήματος με τη βοήθεια μεθευρετικών αλγορίθμων, όπως της μεθόδου βελτιστοποίησης με νοημοσύνη σμήνους σωματιδίων. Στο δεύτερο κεφάλαιο θα γίνει προσπάθεια δημιουργίας του κατάλληλου θεωρητικού υπόβαθρου στον αναγνώστη ώστε να είναι ευκολότερη η κατανόηση και η μοντελοποίηση του προβλήματος και η εκτέλεση του αλγορίθμου. Θα περιοριστούμε βέβαια όσο μπορούμε στα στοιχεία που πραγματικά χρησιμοποιήθηκαν στην ανάπτυξη του αλγορίθμου και είναι απαραίτητα στη διαφώτιση του μελετητή. Αμέσως μετά, θα γίνει πλήρης παρουσίαση του μοντέλου που χρησιμοποιήθηκε. Κατόπιν, θα παρουσιαστούν τα αποτελέσματα του αλγορίθμου και θα ακολουθήσει σχολιασμός για την καλύτερη κατανόηση αυτών.

2. Γενικά στοιχεία για το VRP, το PSO και τον 2-opt

2.1 Το πρόβλημα της δρομολόγησης των οχημάτων (VRP)

2.1.1 Περιγραφή του προβλήματος της δρομολόγησης των οχημάτων (VRP)

[2], [3], [9]. Το πρόβλημα δρομολόγησης οχημάτων είναι πολύ κλασσικό, αφού το συναντάμε σε πάμπολλες εκφάνσεις του στην καθημερινότητά μας. Αφορά την διανομή αγαθών από μία αποθήκη σε κάποιους πελάτες με χρήση ενός αριθμού οχημάτων. Το πρόβλημα είναι ουσιαστικά η καλή διαχείριση των οχημάτων αυτών, έτσι ώστε να βελτιστοποιήσουμε κάθε φορά την διαδικασία, ανάλογα με τους περιορισμούς και τις συμβάσεις που κάθε φορά παρουσιάζονται. Αν το δούμε με μια άλλη ματιά, είναι μια επέκταση του προβλήματος του πλανόδιου πωλητή. Ιστορικά, οι πρώτοι που ασχολήθηκαν με τέτοιου είδους προβλήματα, ήταν οι Dantzig και Ramser (1959).

Τα βασικά χαρακτηριστικά ενός τέτοιου προβλήματος είναι το οδικό δίκτυο, οι πελάτες, οι αποθήκες, τα οχήματα και οι οδηγοί. Αυτά τα χαρακτηριστικά, ανάλογα με την περίπτωση, είναι αυτά που θέτουν και τους περιορισμούς του προβλήματος.

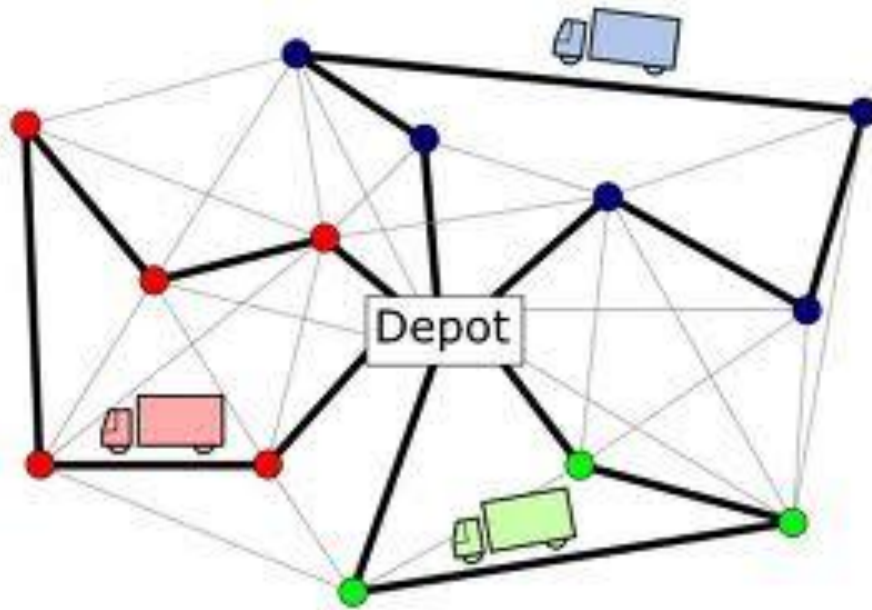
- Οδικό δίκτυο: Η κατάστασή του, θα μας πληροφορήσει για το ποιες μεταβάσεις είναι εφικτές να γίνουν. Συνήθως είναι εφικτό το σύνολο των πιθανών μεταβάσεων.
- Πελάτες: Παραγγέλνουν μια συγκεκριμένη ποσότητα προς παράδοση και μπορεί ακόμη να μας υποδείξουν ένα συγκεκριμένο χρονικό περιθώριο παράδοσης.
- Αποθήκες: Έχουν συγκεκριμένη χωρητικότητα, επομένως παίζουν καθοριστικό ρόλο στην εξέλιξη του προβλήματος.
- Οχήματα: Έχουμε στη διάθεσή μας κάποιο προκαθορισμένο αριθμό διαθέσιμων οχημάτων. Αυτά με τη σειρά τους έχουν ένα όριο χωρητικότητας προϊόντων που μπορούν να μεταφέρουν.
- Οδηγοί: Οι οδηγοί θα μας προσθέσουν με τη σειρά τους κάποιους περιορισμούς που αναφέρονται κυρίως στο χρονικό διάστημα που πρέπει να εργάζονται και στο πλήθος και τη διάρκεια των διαλλειμάτων που κάνουν.

Οι στόχοι που τίθενται στην επίλυση ενός προβλήματος δρομολόγησης οχημάτων έχουν να κάνουν προφανώς με τη βελτίωση των συνθηκών εργασίας και την ελαχιστοποίηση του κόστους παράδοσης. Πιο αναλυτικά, θα προσπαθήσουμε να ελαχιστοποιήσουμε:

- Το κόστος της διαδρομής (σε καύσιμα)

- Το χρόνο παράδοσης των προϊόντων (και της εργασίας των οδηγών)
- Τον αριθμό των οχημάτων

Στην μοντελοποίηση του προβλήματος, θα πρέπει να λάβουμε υπόψιν όλες εκείνες τις παραμέτρους που επηρεάζουν τις συνθήκες, τους συγκεκριμένους στόχους που έχουμε θέσει και τους περιορισμούς που διέπουν το πρόβλημά μας.



Σχηματικά μια λύση ενός VRP

2.1.2 Το πρόβλημα της δρομολόγησης των οχημάτων με ύπαρξη πολλαπλών αποθηκών (MDVRP)

[4], [5]. Είναι μια παραλλαγή του απλού προβλήματος της δρομολόγησης οχημάτων, όπου απλά αντί μίας αποθήκης, διαθέτουμε περισσότερες. Υπάρχουν δύο θεωρίες για τη μοντελοποίηση του MDVRP. Σύμφωνα με την πρώτη, η εκάστοτε αποθήκη, συνδέεται αποκλειστικά με συγκεκριμένους πελάτες και οχήματα. Ουσιαστικά, έχουμε να αντιμετωπίσουμε διαδοχικά προβλήματα τύπου VRP. Στη δεύτερη περίπτωση, ένα όχημα δύναται να εκκινήσει και να τερματίσει σε οποιαδήποτε αποθήκη, έχοντας και την ευχέρεια να έχει χρησιμοποιήσει και άλλες αποθήκες στην πορεία του. Η μοντελοποίηση ενός προβλήματος MDVRP έχει ως εξής:

Ελαχιστοποίηση

$$\min \sum_{i=1}^{depots} \sum_{j=1}^{nodes} c'_{ij} \sum_{k=1}^K x_{ijk} + \sum_{i=1}^{depots} \sum_{j=1}^{nodes} c_{ij} \sum_{k=1}^K x_{ijk} \quad 2.1.2.1$$

Υπό

$$\sum_{k=1}^K y_{ik} = 1, \forall i = 1, \dots, nodes \quad 2.1.2.2$$

$$\sum_{i=1}^{depots} \sum_{k=1}^K y_{ik} = K \quad 2.1.2.3$$

$$\sum_{j=1}^{nodes} x_{jik} = \sum_{j=1}^{nodes} x_{ijk} = y_{ik}, \begin{cases} \forall i = 1, \dots, nodes \\ k = 1, \dots, K \end{cases} \quad 2.1.2.4$$

$$\sum_{k=1}^K x_{ijk} = \sum_{k=1}^K x_{bik}, \begin{cases} \forall i = 1, \dots, depots \\ \forall j, b \in V \end{cases} \quad 2.1.2.5$$

$$\sum_{i \in S} d_i y_{ik} \leq C, \begin{cases} S \subseteq V \\ k = 1, \dots, K \end{cases} \quad 2.1.2.6$$

$$\sum_{i \in S} \sum_{j \in S} c_{ij} x_{ijk} = D, \begin{cases} S \subseteq V \\ k = 1, \dots, K \end{cases} \quad 2.1.2.7$$

$$\sum_{i \in S_1} \sum_{j \notin S_1} x_{ijk} \geq y_{hk}, \begin{cases} \forall S_1 \subseteq V, h \in S_1 \\ k = 1, \dots, K \end{cases} \quad 2.1.2.8$$

$$y_{ik} \in \{0,1\}, \begin{cases} \forall i = 1, \dots, nodes \\ k = 1, \dots, K \end{cases} \quad 2.1.2.9$$

$$x_{ijk} \in \{0,1\}, \begin{cases} i = 1, \dots, nodes \\ j = 1, \dots, nodes \\ k = 1, \dots, K \end{cases} \quad 2.1.2.10$$

όπου

depots: Ο αριθμός των αποθηκών από τις οποίες διανέμονται τα προϊόντα

nodes: Ο αριθμός προς εξυπηρέτηση πελατών

K: Ο αριθμός των φορτηγών που απαιτούνται για την πλήρη εξυπηρέτηση των πελατών

C : Η χωρητικότητα του κάθε οχήματος.

D : Η μέγιστη επιτρεπόμενη τιμή κόστους (το μέγεθος αυτό μπορεί να είναι χρόνος ή απόσταση) που μπορεί να επιτραπεί σε ένα όχημα να διανύσει κατά την διάρκεια μιας διαδρομής.

c'_{ij} : Το κόστος για την μεταφορά των προϊόντων από την αποθήκη i στον πελάτη j. Στην περίπτωση δηλαδή που το όχημα εξυπηρετεί τον πρώτο του πελάτη μετά την έξοδο του από την αποθήκη.

c'_{ij} : Το κόστος για την μεταφορά των προϊόντων από τον πελάτη i στον πελάτη j όταν το όχημα εξυπηρετεί μια σειρά από πελάτες.

d_i : Η ποσότητα των προϊόντων που απαιτεί ο κάθε πελάτης.

x_{ijk} : Η μεταβλητή x_{ijk} παίρνει την τιμή 1 όταν το τόξο $\{i,j\}$ διασχίζεται από το όχημα k ενώ σε αντίθετη περίπτωση είναι ίση με 0.

y_{ik} : Η μεταβλητή y_{ik} παίρνει την τιμή 1 αν ο πελάτης i εξυπηρετείται από τον όχημα k .

S : Ένα υποσύνολο πελατών των οποίων τις ανάγκες το όχημα καλύπτει σε μια διαδρομή αφού ο συνολικός όγκος των προϊόντων των συγκεκριμένων πελατών είναι μικρότερος από των χωρητικότητα του οχήματος.

S_1 : Ένα υποσύνολο πελατών.

Η αντικειμενική συνάρτηση που προφανώς είναι προς ελαχιστοποίηση, αποτελείται από δύο αθροίσματα. Το πρώτο μάς δίνει το κόστος της μεταφοράς των προϊόντων από την αποθήκη προς κάποιο πελάτη, ενώ το δεύτερο εκφράζει το κόστος για τις ενδιάμεσες μετακινήσεις στη διάρκεια μιας συγκεκριμένης διαδρομής.

Οι επόμενες σχέσεις επιβάλλουν τους περιορισμούς στο πρόβλημα.

1. Η 2.1.2.2 μας εγγυάται πως ο κάθε πελάτης θα δέχεται επίσκεψη μόνο μία φορά.
2. Η 2.1.2.3 εξασφαλίζει ότι όλα τα διαθέσιμα οχήματα θα απασχοληθούν για την επίλυση.
3. Η 2.1.2.4 θα υποχρεώνει το όχημα που εξυπηρετεί τον εκάστοτε πελάτη, να αποχωρεί αμέσως μετά.
4. Ακόμη, ένα φορτηγό θα πρέπει πάντα να ξεκινάει και να τερματίζει σε κάποια αποθήκη, αυτό δεσμεύεται από την 2.1.2.5.
5. Οι περιορισμοί που έχουν να κάνουν με την χωρητικότητα και τις δυνατότητες γενικά των οχημάτων δίνονται από τις 2.1.2.6 και 2.1.2.7.
6. Τέλος η σχέση 2.1.2.8 εξασφαλίζει την συνέχεια της διαδρομής του οχήματος k .

2.2 Βελτιστοποίηση με νοημοσύνη σμήνους σωματιδίων (PSO)

[6], [7], [8]. Ο αλγόριθμος βελτιστοποίησης με νοημοσύνη σμήνους σωματιδίων ανήκει στη γενικότερη κατηγορία των μεθευρετικών αλγορίθμων. Οι μεθευρετικοί αλγόριθμοι, για να επιλύσουν ένα πρόβλημα, συχνά συνδυάζουν

διαδικασίες τοπικής αναζήτησης, αλλά και διαδικασίες πολλών ειδών στρατηγικής με σκοπό να βρουν κάποιο βέλτιστο. Σημαντικό τους στοιχείο είναι ότι συνήθως εκτελούνται ταχύτατα. Αυτό είναι πολύ σημαντικό σε περιπτώσεις όπου ο χρόνος έχει ζωτική σημασία. Βέβαια πολλές φορές οι αλγόριθμοι αυτοί αρκούνται σε υποβέλτιστα, που παρ' όλα αυτά είναι αρκετά καλά ώστε να είναι αποδεκτά. Τα βασικά χαρακτηριστικά της οικογένειας των μεθευρετικών αλγορίθμων είναι:

1. Μοντελοποιούν φυσικά φαινόμενα ή φυσικές διαδικασίες
2. Μπορούν να μεταφερθούν εύκολα σε παράλληλη μορφή
3. Είναι προσαρμοστικοί αλγόριθμοι

Ο αλγόριθμος βελτιστοποίησης με νοημοσύνη σμήνους σωματιδίων προτάθηκε αρχικά από τους Kennedy και Eberhart και έχει σαν βασική ιδέα τον τρόπο κίνησης μιας ομάδας, όπως ενός σμήνους πουλιών ή ενός κοπαδιού ψαριών. Ο μηχανισμός που χρησιμοποιείται για την αλλαγή της θέσης των ατόμων, βασίζεται στην εμπειρία του ίδιου, αλλά και της ομάδας σαν σύνολο, είναι δηλαδή ένας συνεργατικός αλγόριθμος. Αυτό είναι και το βασικό πλεονέκτημα αυτού του αλγορίθμου συν το ότι η καταχθείσα γνώση δεν χάνεται, αφού υπάρχει μνήμη. Επομένως οι καλές λύσεις και η εμπειρία μεταβιβάζονται από γενιά σε γενιά λύσεων. Μπορούμε να προσθέσουμε ακόμα, την εύκολη (γενικά) υλοποίηση του αλγορίθμου στα θετικά στοιχεία του.

Η αναλογία του πραγματικού σμήνους με τον αλγόριθμο, είναι ότι το κάθε πουλί για το σμήνος, είναι για τον αλγόριθμο ένα σωματίδιο όπου η θέση του αναπαριστά μια εφικτή λύση για το προς επίλυση πρόβλημα. Το σμήνος και στις δύο περιπτώσεις σημαίνει το σύνολο (των πουλιών ή σωματιδίων) που με την κίνησή τους στο χώρο προσπαθούν να βελτιστοποιήσουν την αντικειμενική συνάρτηση που έχει τεθεί.

Η κίνηση του κάθε σωματιδίου είναι μια πολύπλοκη διαδικασία που εξαρτάται από την υπάρχουσα θέση του, την εμπειρία (ή γνώση) του ίδιου και την εμπειρία του σμήνους. Για να γίνει πιο κατανοητή η διαδικασία αυτή θα περιγράψουμε τις εξισώσεις της κίνησης του σωματιδίου.

$$\bullet \quad v_{ij}(t+1) = v_{ij}(t) + c_1 * rand_1(pbest_{ij} - x_{ij}(t)) + c_2 * rand_2(gbest_j - x_{ij}(t)) \quad 2.2.1$$

και

$$\bullet \quad x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad 2.2.2$$

, όπου t είναι ο μετρητής των επαναλήψεων, c_1 και c_2 είναι μεταβλητές επιτάχυνσης, $rand_1$, $rand_2$ είναι δύο τυχαίες μεταβλητές στο διάστημα $[0, 1]$. Οι μεταβλητές $pbest_{ij}$ και $gbest_j$ εκφράζουν τις θέσεις όπου τα σωματίδια και το

σμήνος έχουν πάρει την καλύτερη τους βαθμολογία. Οι μεταβλητές $pbest_{ij}$ και $gbest_j$ υπολογίζονται από τις ακόλουθες σχέσεις:

$$\bullet \quad pbest_{ij} = \begin{cases} x_{ij}(t+1), & \text{εάν } f(x_{ij}(t+1)) < f(x_{ij}(t)) \\ pbest_{ij}, & \text{αλλιώς} \end{cases} \quad \text{για ελαχιστοποίηση}$$

2.2.3

$$\bullet \quad pbest_{ij} = \begin{cases} x_{ij}(t+1), & \text{εάν } f(x_{ij}(t+1)) > f(x_{ij}(t)) \\ pbest_{ij}, & \text{αλλιώς} \end{cases} \quad \text{για μεγιστοποίηση}$$

2.2.4

$$\bullet \quad gbest_j \in \{pbest_{1j}, pbest_{2j}, \dots, pbest_{Nj} | f(gbest_j)\} = \min\{f(pbest_{1j}), f(pbest_{2j}), \dots, f(pbest_{Nj})\} \quad 2.2.5$$

Οι μεταβλητές επιτάχυνσης ελέγχουν το πόσο μακριά ή κοντά μπορούν να μετακινηθούν τα σωματίδια σε μία κίνηση τους. Οι τιμές $pbest_{ij}$ αλλάζουν αν η παρούσα θέση του εκάστοτε σωματιδίου βαθμολογείται υψηλότερα από την προηγούμενη τιμή του $pbest_{ij}$. Το $gbest_j$ είναι πάντα ένα από τα $pbest_{ij}$, αυτό με την υψηλότερη βαθμολογία. Το $v_{ij}(t)$ χρησιμοποιείται σαν μνήμη των προηγούμενων κατευθύνσεων, έτσι ώστε να απαγορεύει στα σωματίδια να αλλάξουν πολύ απότομα κατεύθυνση. Ο δεύτερος όρος της εξίσωσης $c_1 * rand_1(pbest_{ij} - x_{ij}(t))$, ονομάζεται γνωστικός όρος, αφού προσπαθεί να "σπρώξει" το σωματίδιο στην καλύτερη κατάσταση που είχε φτάσει ποτέ το ίδιο. Αντίθετα, ο τρίτος όρος $c_2 * rand_2(gbest_j - x_{ij}(t))$, αναφέρεται ως κοινωνικός, εφόσον "σπρώχνει" το σωματίδιο προς το γενικό βέλτιστο, που έχει αποκτηθεί από το σμήνος σαν σύνολο.

Παραθέτουμε τον ψευδοκώδικα του αλγορίθμου:

- Επιλογή του μέγιστου αριθμού επαναλήψεων
- Επιλογή του μεγέθους του σμήνους
- Επιλογή των μεταβλητών c_1, c_2
- Αρχικοποίηση των λύσεων (αρχική θέση κάθε σωματιδίου)
- Αρχικοποίηση των ταχυτήτων των σωματιδίων
- Υπολογισμός κόστους αρχικών λύσεων
- Εύρεση βέλτιστης λύσης κάθε σωματιδίου ($pbest_{ij}$)
- Εύρεση βέλτιστης λύσης του σμήνους ($gbest_j$)
- Επανάληψη έως (να φτάσει ο μέγιστος αριθμός επαναλήψεων)
 - Υπολογισμός της ταχύτητας κάθε σωματιδίου
 - Υπολογισμός της νέας θέσης κάθε σωματιδίου
 - Υπολογισμός κόστους της λύσης κάθε σωματιδίου

- Εύρεση βέλτιστης λύσης κάθε σωματιδίου ($pbest_{ij}$)
- Εύρεση βέλτιστης λύσης του σμήνους ($gbest_j$)
- Τέλος επανάληψης
- Επιστροφή βέλτιστης λύσης

Μια παραλλαγή του αλγορίθμου προτάθηκε (για να βελτιώσει τον αρχικό) από τους Shi και Eberhart η οποία, εν συντομία, προτείνει τη χρήση μιας μεταβλητής w που δίνει την ιδιότητα της αδράνειας στα σωματίδια. Στόχος της εισαγωγής του βάρους αδράνειας w είναι η αύξηση της ικανότητας διασποράς της αναζήτησης και της ικανότητας εντατικοποίησης της αναζήτησης. Η εξίσωση του αλγορίθμου παίρνει πλέον την ακόλουθη μορφή:

$$v_{ij}(t+1) = w * v_{ij}(t) + c_1 * rand_1(pbest_{ij} - x_{ij}(t)) + c_2 * rand_2(gbest_j - x_{ij}(t)) \quad 2.2.3$$

Το w είναι σε θέση να επηρεάσει σημαντικά τη σύγκλιση του αλγορίθμου σε τοπικό ή ολικό βέλτιστο καθώς και την ταχύτητα σύγκλισης. Μια μικρή τιμή του w θα μειωθεί η ικανότητα διασποράς των λύσεων, αυξάνοντας την πιθανότητα εγκλωβισμού σε τοπικό βέλτιστο. Αντίθετα, μια πολύ μεγάλη τιμή θα αναγκάσει τις λύσεις να αλλάζουν πολύ γρήγορα με αποτέλεσμα να μην μπορούν να πλησιάσουν εύκολα το ολικό βέλτιστο. Αυτό που στην πράξη συνηθίζεται επειδή δουλεύει καλά, είναι να μειώνουμε την τιμή του w σε κάθε επανάληψη του αλγορίθμου σύμφωνα με τον τύπο

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} * t \quad 2.2.4$$

Τα w_{max} και w_{min} υποδεικνύουν τα άκρα του διαστήματος των επιτρεπτών τιμών του w και το $iter_{max}$ είναι ο μέγιστος αριθμός επαναλήψεων.



Στην εικόνα φαίνεται πως το μεγαλύτερο μέρος του σμήνους συγκεντρώνεται σε ένα βέλτιστο, ενώ κάποια σωματίδια προσπαθούν να βρουν ακόμη καλύτερη λύση. Βέβαια στη φύση, το πρόβλημα που συνήθως τίθεται είναι η εύρεση τροφής.

2.3 Αλγόριθμος τοπικής αναζήτησης 2-opt

[1] Ο αλγόριθμος 2-opt ανήκει στην οικογένεια των αλγορίθμων τοπικής αναζήτησης, αφού κρατάει το μεγαλύτερο μέρος της λύσης ίδιο και αλλάζει μόνο μερικούς κόμβους. Είναι σύνηθες (όπως γίνεται και στην παρούσα εργασία) να χρησιμοποιείται σαν βοηθητικός αλγόριθμος μαζί με τον PSO. Ουσιαστικά, αυτό που πραγματοποιεί είναι να επιλέγει δύο τόξα και αφού τα διαγράψει, να τα αντικαταστήσει με δύο νέα, με την ελπίδα ότι έτσι θα μειωθεί το κόστος. Σε αυτό το σημείο πρέπει να ξεκαθαρίσουμε ότι υπάρχουν πολλοί τρόποι για να επιλεγούν αυτά τα τόξα. Ένας συνηθισμένος τρόπος, είναι να επιλέξουμε τα δύο χειρότερα τόξα, δηλαδή αυτά με τα μεγαλύτερα κόστη. Ακόμη, μπορούμε να διαγράψουμε το χειρότερο τόξο και άλλο ένα με τυχαία επιλογή. Μια άλλη λογική είναι να επιλεγούν και τα δύο τόξα, με τυχαίο τρόπο. Και οι τρεις τρόποι που φυσικά δεν είναι οι μοναδικοί, έχουν πλεονεκτήματα και μειονεκτήματα. Ανάλογα με την κατάσταση

κάθε φορά πρέπει να επεξεργαζόμαστε τα δεδομένα, και να επιλέγουμε όποιον εμείς θεωρούμε πως θα δώσει καλύτερα αποτελέσματα.

Ένα σημαντικό θέμα είναι ο τρόπος ανεύρεσης του κόστους. Δύο είναι οι πιο συνηθισμένοι. Στον πρώτο υπολογίζουμε το κόστος όλης της διαδρομής πριν και μετά την αλλαγή των κόμβων. Το μεγάλο μειονέκτημα της μεθόδου αυτής είναι ότι είναι αργή, αφού πρέπει να προσπελάσει όλους τους κόμβους και να υπολογίσει το κόστος των μεταβάσεων 2 φορές. Το πρόβλημα αυτό μπορεί να είναι απαγορευτικό σε προβλήματα με πολύ μεγάλο αριθμό κόμβων. Στο δεύτερο, υπολογίζουμε μόνο τη διαφορά του κόστους των τόξων που επιδέχονται αλλαγή. Έτσι, χρειάζεται υπολογισμός κόστους μόνο σε 4 τόξα, ανεξάρτητα από το πλήθος των κόμβων προς εξυπηρέτηση.

Ο τρόπος αποθήκευσης της λύσης παίζει ρόλο στο πόσο εύκολα μπορούν να γίνουν οι αλλαγές των κόμβων. Μια πολύ καλή πρόταση, είναι η λύση να αποθηκεύεται σε ένα διάνυσμα με τη μορφή:

$$\text{Sol} = (n_1, n_2, \dots, n_k)$$

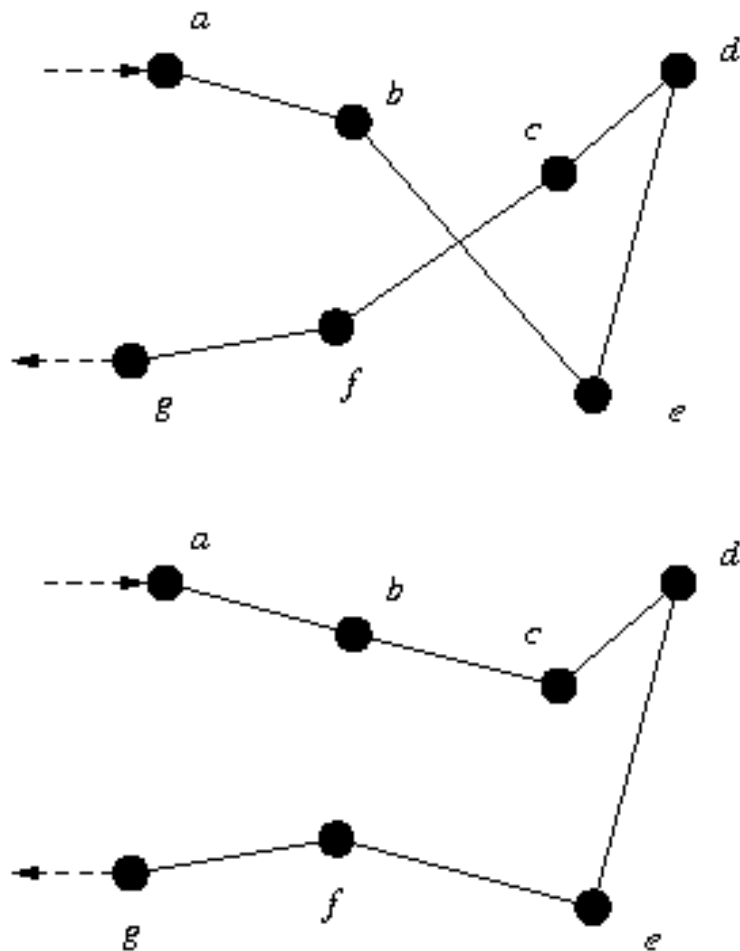
όπου n_i είναι η ονομασία του κόμβου i

Για παράδειγμα, έχουμε τη λύση $\text{Sol} = (1, 4, 2, 3, 7, 6, 5)$ και θέλουμε να αλλάξουμε τα τόξα $4 \rightarrow 2$ και $6 \rightarrow 5$. Το διάνυσμα μετά την αλλαγή θα πάρει τη μορφή $\text{Sol}' = (1, 4, 6, 7, 3, 2, 5)$. Παρατηρούμε πως από το διάνυσμα, οι κόμβοι που βρίσκονται εξωτερικά των τόξων, δεν επηρεάζονται καθόλου. Αντίθετα, οι εσωτερικοί, αντιστρέφονται. Δηλαδή, η αλληλουχία $2 \rightarrow 3 \rightarrow 7 \rightarrow 6$, μετατρέπεται στην $6 \rightarrow 7 \rightarrow 3 \rightarrow 2$. Έτσι, με μια απλή αντιστροφή της αλληλουχίας του εσωτερικού διανύσματος της αλλαγής, έχουμε έτοιμη τη νέα λύση για να βρούμε το κόστος της. Αν βρούμε το κόστος της νέας λύσης με τον πρώτο τρόπο, πρέπει να υπολογίσουμε τα κόστη μεταβάσεων για όλα τα τόξα, ενώ με τον δεύτερο τρόπο, μόνο τα κόστη των τόξων $4 \rightarrow 2$, $6 \rightarrow 5$, $4 \rightarrow 6$, $2 \rightarrow 5$.

Παραθέτουμε τον ψευδοκώδικα του αλγορίθμου:

- Υπολογισμός κόστους της αρχικής λύσης sol_1
- Επανάληψη έως (να φτάσει ένας μέγιστος αριθμός επαναλήψεων)
 - Εύρεση δύο τόξων με κάποιον από τους τρόπους που εξηγούνται παραπάνω
 - Αντιστροφή της αλληλουχίας του εσωτερικού διανύσματος της αλλαγής
 - Εύρεση του κόστους της νέας λύσης sol_2
 - Εάν $(\text{sol}_1 > \text{sol}_2)$
 - Κρατάμε τη νέα λύση
 - Σπάμε την επανάληψη

- Τέλος επανάληψης



Στο παραπάνω σχήμα φαίνεται πως ο αλγόριθμος 2-opt, επιφέρει μια αλλαγή στη λύση. Αν το κόστος της δεύτερης λύσης είναι μικρότερο από της πρώτης, κρατάμε όντως την πρώτη. Στην αντίθετη περίπτωση, ο 2-opt θα πρέπει να εξετάσει άλλη περίπτωση αλλαγής και ούτω καθεξής.

3. Περιγραφή του προβλήματος και μοντελοποίηση

3.1 Μοντελοποίηση του προβλήματος

Στην παρούσα εργασία έχουμε να αντιμετωπίσουμε το πρόβλημα της δρομολόγησης των οχημάτων με ύπαρξη πολλαπλών αποθηκών (MDVRP). Θα κάνουμε μια προσέγγιση με τη βοήθεια των αλγορίθμων PSO και 2-opt όπως έχουμε ήδη αναφέρει. Ένα πολύ σημαντικό ζήτημα που πρέπει να μας απασχολήσει είναι η μοντελοποίηση του προβλήματος έτσι ώστε να είναι δυνατή η επίλυσή του με τους παραπάνω αλγορίθμους. Ειδικότερα, το PSO δουλεύει για συνεχή προβλήματα, ενώ στη δικιά μας περίπτωση έχουμε να κάνουμε με ένα πρόβλημα διακριτό, εφόσον τα σημεία εξυπηρέτησης αλλά και οι αποθήκες είναι διακριτά σημεία στο δισδιάστατο χώρο που μας ενδιαφέρει. Επομένως είμαστε αναγκασμένοι να κωδικοποιήσουμε κατάλληλα τα δεδομένα μας σε μορφή που να μπορεί να επεξεργαστεί το PSO. Στη βιβλιογραφία αναφέρονται διάφοροι τρόποι μοντελοποίησης, που όμως δε φάνηκαν πολύ χρήσιμοι στην συγκεκριμένη περίπτωση, έτσι αναπτύχθηκε ένα νέο μοντέλο προσέγγισης που φαίνεται να διευκολύνει την κατανόηση και την ανάπτυξη τέτοιων προβλημάτων. Η βασική του ιδέα είναι ότι στο διάλυμα της λύσης του εκάστοτε σωματιδίου κρατάμε την ονομασία των σημείων εξυπηρέτησης με τη σειρά που αυτά επισκέπτονται, ενώ χρησιμοποιείται και ένας βοηθητικός πίνακας που μας παρέχει πληροφορίες για τις ακριβείς συντεταγμένες των σημείων εξυπηρέτησης και των αποθηκών. Για την καλύτερη κατανόηση του μοντέλου αυτού παραθέτουμε ένα μικρό παράδειγμα.

Έστω ένα απλό πρόβλημα VRP με ένα φορτηγό, μια αποθήκη και πέντε σημεία εξυπηρέτησης. Οι συντεταγμένες τους παρουσιάζονται παρακάτω:

Nodes/Depots	Coordinates
1	2 , 3
2	6 , 8
3	-7, 2
4	-2, -11
5	5 , -1
D	0 , 0

Θεωρούμε σαν τυχαία αρχική λύση την D->2->1->D->5->3->4->D. Για να αρχίσουν οι εξισώσεις 2.2.1-4 να επεξεργάζονται αυτή τη λύση και να μας δίνουν επόμενες λύσεις, αντί να χρησιμοποιήσουν τα 1,2,3,4,5,D, θα πάρουν τις συντεταγμένες από τον αντίστοιχο πίνακα, δηλαδή τις τιμές Coordinates(1-D). Έστω λοιπόν ότι μετά από τις πράξεις, οι εξισώσεις μας δίνουν την επόμενη λύση, προφανώς σε μορφή συντεταγμένων, π.χ. -3,2 , 1,9 -> 6,4 , -8,1 -> -9,2 , 4,5 -> 2,2 , 2,1 -> 6,1 , 0,2 -> 3,8 ,

0,9 -> -1,1 , -1,7 -> 7,5 , -3,2 ->. Τώρα με ψάξιμο θα βρούμε για το κάθε σημείο που μας έδωσε η εξίσωση, τον πιο κοντινό διαθέσιμο κόμβο και θα κάνουμε αντικατάσταση. Έτσι, η λύση θα πάρει και πάλι κατανοητή μορφή: D -> 2 -> 3 -> D -> 4 -> 5 -> 1 -> D.

Δύο ακόμα σημαντικές παραδοχές γίνονται για την απλοποίηση του μοντέλου μας. Καταρχήν, στο αρχικό διάνυσμα των λύσεων δεν εισάγουμε τις αποθήκες σαν ενδιάμεσους κόμβους, άρα γίνεται έμμεσα η παραδοχή της άπειρης χωρητικότητας των οχημάτων μεταφοράς. Επιπλέον, κάνουμε χρήση ενός μόνο οχήματος για την εύρεση των λύσεων. Φυσικά, επειδή το πρόβλημα δεν είναι συμβατό με αυτές τις απλοποιήσεις, σε κάθε βήμα του αλγορίθμου πρέπει να ακολουθείται μια διαδικασία που να επαναφέρει τις λύσεις σε μορφή που να εκφράζουν στην πραγματικότητα μια εφικτή λύση. Σε αυτό το σημείο κρίνουμε σκόπιμο να παραθέσουμε ένα μικρό δείγμα της διαδικασίας αυτής για την καλύτερη παρακολούθηση της από τον αναγνώστη.

Έστω το διάνυσμα λύσης ενός ατόμου: 2 -> 1 -> 6 -> 4 -> 3 -> 5. Πρώτα θα ερευνήσουμε πόσες φορές πρέπει να γίνει ανεφοδιασμός στην περίπτωση που χρησιμοποιούμε ένα όχημα. Ξεκινάμε λοιπόν το φορτηγό από την κοντινότερη αποθήκη (σε σχέση με τον πρώτο κόμβο) και εξυπηρετούμε όσα περισσότερα σημεία μπορούμε. Κατόπιν, οδηγούμε το φορτηγό στην κοντινότερη αποθήκη. Αυτό συνεχίζεται έως ότου εξυπηρετηθούν όλοι οι κόμβοι. Οπότε και το διάνυσμα έχει ως εξής:

- D -> 2 -> 1 -> 6 -> 4 -> D -> 3 -> 5 -> D

Τώρα θα πρέπει να γίνει ο καταμερισμός της διαδρομής σε όλα τα διαθέσιμα οχήματα. Αν για παράδειγμα εδώ έχουμε δύο διαθέσιμα οχήματα θα μοιραστούν τη διαδρομή ως εξής:

1. D -> 2 -> 1 -> 6 -> 4 -> D
2. D -> 3 -> 5 -> D

Στη φάση αυτή θα μπορούσε να αναρωτηθεί κανείς αν υπάρχει κέρδος με τη διαδικασία του καταμερισμού. Αυτό που αλλάζει προς το καλύτερο σε αυτήν την περίπτωση είναι ότι επειδή κάθε φορτηγό ξεκινάει και τελειώνει σε οποιαδήποτε αποθήκη, στην περίπτωση που έχουμε περισσότερα από ένα διαθέσιμα, μπορούμε να μεταβούμε από και προς τους “ακριανούς” κόμβους πιο γρήγορα. Στο προηγούμενο παράδειγμα μετά τον κόμβο 4 επισκεπτόμαστε την συγκεκριμένη αποθήκη επειδή είναι η πιο κοντινή στον συγκεκριμένο κόμβο. Όμως μετά πρέπει υποχρεωτικά (στην περίπτωση του ενός οχήματος) από εκεί να προχωρήσουμε στον 3. Αντίθετα, αν έχουμε δύο διαθέσιμα οχήματα, το δεύτερο θα ξεκίναγε τη διαδρομή του από την αποθήκη που είναι κοντύτερα στον κόμβο 3.

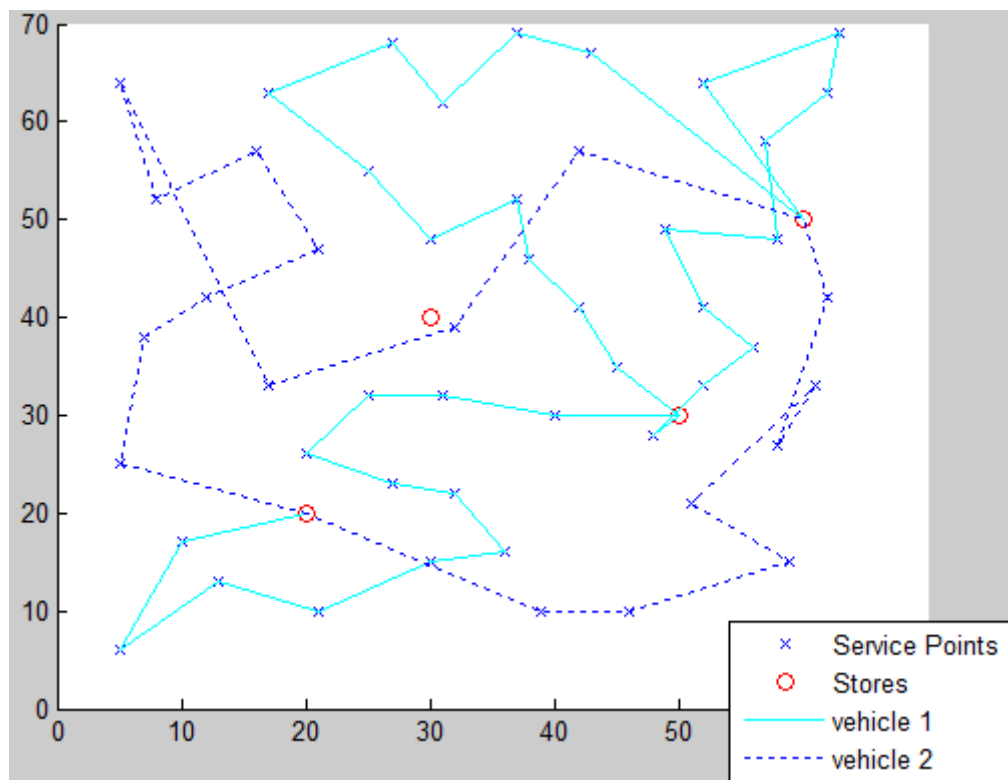
Δεν πρέπει να ξεχνάμε τον αρχικό στόχο του προβλήματος που δεν είναι άλλος από την ελαχιστοποίηση του κόστους του συνόλου των μεταβάσεων. Στο συγκεκριμένο πρόβλημα το κόστος δεν έχει σχέση με το χρόνο ή τα καύσιμα, αλλά απορρέει από την ευκλείδεια απόσταση των κόμβων. Για να είναι εφικτοί οι έλεγχοι στο PSO, σε κάθε επανάληψη που παίρνουμε μια νέα λύση βρίσκουμε το κόστος της και κατόπιν το συγκρίνουμε με τα κόστη των βέλτιστων λύσεων (PersonalBest, GlobalBest). Εννοείται πως το κόστος υπολογίζεται, αφού η εκάστοτε λύση είναι στην τελική της μορφή, αφού δηλαδή έχουμε υπολογίσει όλη την αλληλουχία που πρέπει να ακολουθήσουν τα οχήματα για να εξυπηρετήσουν όλους τους κόμβους. Στο προηγούμενο παράδειγμα θα έπρεπε να αθροίσουμε τις αποστάσεις D->2, 2->1, 1->6, 6->4, 4->D, D->3, 3->5, 5->D. Η ευκλείδεια απόσταση υπολογίζεται από τον ακόλουθο τύπο:

$$ED = \sqrt{\Delta x^2 + \Delta y^2}$$

όπου $\Delta x = x_2 - x_1$ και $\Delta y = y_2 - y_1$

Στα x και y χρησιμοποιούμε τις συντεταγμένες των κόμβων από τον αντίστοιχο πίνακα Coordinates.

Είναι σημαντικό να γίνει σαφές το γεγονός ότι υπάρχει περίπτωση να μη χρησιμοποιηθούν όλες οι αποθήκες αν αυτό δεν είναι συμφέρον για τη λύση. Αυτό δεν παραβαίνει τους περιορισμούς που πουθενά δεν αναφέρουν κάτι τέτοιο. Παρακάτω διακρίνεται αυτή ακριβώς η περίπτωση σχηματικά.



3.2 Μορφή των δεδομένων

Σημαντικό επίσης, είναι να το βήμα όπου λαμβάνουμε τα δεδομένα του κάθε προβλήματος και τα μετατρέπουμε σε μορφή κατάλληλη για να επεξεργαστούν από τον αλγόριθμό μας. Η αρχική τους μορφή είναι η ακόλουθη:

```
2 2 50 4
0 160
0 160
0 160
0 160
1 37 52 0 7 1 4 1 2 4 8
2 49 49 0 30 1 4 1 2 4 8
3 52 64 0 16 1 4 1 2 4 8
4 20 26 0 9 1 4 1 2 4 8
5 40 30 0 21 1 4 1 2 4 8
6 21 47 0 15 1 4 1 2 4 8
7 17 63 0 19 1 4 1 2 4 8
8 31 62 0 23 1 4 1 2 4 8
9 52 33 0 11 1 4 1 2 4 8
10 51 21 0 5 1 4 1 2 4 8
11 42 41 0 19 1 4 1 2 4 8
12 31 32 0 29 1 4 1 2 4 8
13 5 25 0 23 1 4 1 2 4 8
14 12 42 0 21 1 4 1 2 4 8
15 36 16 0 10 1 4 1 2 4 8
16 52 41 0 15 1 4 1 2 4 8
17 27 23 0 3 1 4 1 2 4 8
18 17 33 0 41 1 4 1 2 4 8
19 13 13 0 9 1 4 1 2 4 8
20 57 58 0 28 1 4 1 2 4 8
21 62 42 0 8 1 4 1 2 4 8
22 42 57 0 8 1 4 1 2 4 8
23 16 57 0 16 1 4 1 2 4 8
24 8 52 0 10 1 4 1 2 4 8
25 7 38 0 28 1 4 1 2 4 8
26 27 68 0 7 1 4 1 2 4 8
27 30 48 0 15 1 4 1 2 4 8
28 43 67 0 14 1 4 1 2 4 8
29 58 48 0 6 1 4 1 2 4 8
30 58 27 0 19 1 4 1 2 4 8
31 37 69 0 11 1 4 1 2 4 8
32 38 46 0 12 1 4 1 2 4 8
33 46 10 0 23 1 4 1 2 4 8
34 61 33 0 26 1 4 1 2 4 8
35 62 63 0 17 1 4 1 2 4 8
36 63 69 0 6 1 4 1 2 4 8
37 32 22 0 9 1 4 1 2 4 8
38 45 35 0 15 1 4 1 2 4 8
```

39	59	15	0	14	1	4	1	2	4	8
40	5	6	0	7	1	4	1	2	4	8
41	10	17	0	27	1	4	1	2	4	8
42	21	10	0	13	1	4	1	2	4	8
43	5	64	0	11	1	4	1	2	4	8
44	30	15	0	16	1	4	1	2	4	8
45	39	10	0	10	1	4	1	2	4	8
46	32	39	0	5	1	4	1	2	4	8
47	25	32	0	25	1	4	1	2	4	8
48	25	55	0	17	1	4	1	2	4	8
49	48	28	0	18	1	4	1	2	4	8
50	56	37	0	10	1	4	1	2	4	8
51	20	20	0	0	0	0				
52	30	40	0	0	0	0				
53	50	30	0	0	0	0				
54	60	50	0	0	0	0				

Η πρώτη γραμμή μας δίνει τις βασικότερες πληροφορίες του εκάστοτε προβλήματος. Από το πρώτο ψηφίο εξαγούμε την πληροφορία ότι το πρόβλημα είναι όντως MDVRP. Το δεύτερο ψηφίο μας αποκαλύπτει πόσο διαθέσιμα οχήματα υπάρχουν. Στο επόμενο, φαίνεται πόσα είναι τα σημεία που πρέπει να εξυπηρετηθούν. Τέλος, ο αριθμός των αποθηκών που μπορούν να χρησιμοποιηθούν για ανεφοδιασμό από τα οχήματα, φαίνεται στο τελευταίο ψηφίο της πρώτης γραμμής.

Στις επόμενες γραμμές, που αποτελούνται από δύο ψηφία, εμφανίζεται η ελάχιστη και η μέγιστη δυνατότητα φόρτωσης των αποθηκών. Δηλαδή η ελάχιστη και μέγιστη ποσότητα προμηθειών που μπορεί να παραλάβει ένα όχημα κάθε φορά που μεταβαίνει σε μια αποθήκη.

Οι επόμενες σειρές εκτός τις τέσσερις τελευταίες μας δίνουν τον κύριο όγκο των απαραίτητων πληροφοριών. Οι γραμμές αυτές έχουν την ακόλουθη μορφή:

i x y d q f a list e l

όπου

- **i**: Αριθμός του πελάτη (σημείου εξυπηρέτησης)
- **x**: Τετμημένη του σημείου
- **y**: Τεταγμένη του σημείου
- **d**: Διάρκεια εξυπηρέτησης *
- **q**: Ζήτηση (Απαραίτητη ποσότητα εκφόρτωσης στον κόμβο)
- **f**: Συχνότητα επισκέψεων (Πόσες φορές χρειάζεται να επισκεψη ο κόμβος)
- **a**: Αριθμός των πιθανών συνδυασμένων επισκέψεων *
- **list**: Λίστα των πιθανών συνδυασμένων επισκέψεων *

- **e**: Αρχή του χρονικού περιθωρίου *
- **I**: Τέλος του χρονικού περιθωρίου *

Οι παραπάνω αστερίσκοι (*) σημαίνουν πως αυτά τα στοιχεία δεν είναι απαραίτητα στην περίπτωση μας.

Οι τελευταίες σειρές μας δίνουν πληροφορίες για τις αποθήκες. Ειδικότερα, κάθε σειρά αναφέρεται σε μία αποθήκη, όπου στο πρώτο ψηφίο την ονοματίζει και στα επόμενα δύο μας δίνει τις συντεταγμένες τις στο δισδιάστατο χώρο.

Εμείς, διαβάζοντας όλα αυτά τα στοιχεία, μέσα από κατάλληλες υπορουτίνες του αλγορίθμου, τα φιλτράρουμε και κρατάμε τελικά μόνο τα χρήσιμα. Για τις αποθήκες και τα σημεία εξυπηρέτησης δημιουργούμε ένα πίνακα όπου αποθηκεύουμε τις συντεταγμένες τους, τη ζήτηση και τη συχνότητα των επισκέψεων. Ο πίνακας που δημιουργείται για το συγκεκριμένο πρόβλημα παρουσιάζεται ακολούθως:

name	x	y	d	f
1	37	52	7	1
2	49	49	30	1
3	52	64	16	1
4	20	26	9	1
5	40	30	21	1
6	21	47	15	1
7	17	63	19	1
8	31	62	23	1
9	52	33	11	1
10	51	21	5	1
11	42	41	19	1
12	31	32	29	1
13	5	25	23	1
14	12	42	21	1
15	36	16	10	1
16	52	41	15	1
17	27	23	3	1
18	17	33	41	1
19	13	13	9	1
20	57	58	28	1
21	62	42	8	1
22	42	57	8	1
23	16	57	16	1
24	8	52	10	1
25	7	38	28	1
26	27	68	7	1
27	30	48	15	1

28	43	67	14	1
29	58	48	6	1
30	58	27	19	1
31	37	69	11	1
32	38	46	12	1
33	46	10	23	1
34	61	33	26	1
35	62	63	17	1
36	63	69	6	1
37	32	22	9	1
38	45	35	15	1
39	59	15	14	1
40	5	6	7	1
41	10	17	27	1
42	21	10	13	1
43	5	64	11	1
44	30	15	16	1
45	39	10	10	1
46	32	39	5	1
47	25	32	25	1
48	25	55	17	1
49	48	28	18	1
50	56	37	10	1
51	20	20	0	0
52	30	40	0	0
53	50	30	0	0
54	60	50	0	0

Ένα τέχνασμα που χρησιμοποιήθηκε για την αποφυγή δύο ξεχωριστών πινάκων για τα σημεία εξυπηρέτησης και των αποθηκών, είναι η ονομάτιση των αποθηκών με αριθμούς που ξεκινάνε από το πλήθος των σημείων εξυπηρέτησης αντί το 1. Δηλαδή η πρώτη αποθήκη θα είναι για μας η 51 και όχι η 1. Αντίστοιχα και για τις υπόλοιπες. Γι αυτό και οι τέσσερις τελευταίες σειρές έχουν μηδενικά στη θέση της ζήτησης και της συχνότητας.

Όσο αφορά τα στοιχεία της πρώτης γραμμής, αποθηκεύονται σε ξεχωριστά διανύσματα και χρησιμοποιούνται όποτε χρειαστεί.

3.3 Βοηθητικοί πίνακες

Έχουμε ήδη αναφέρει πως σε κάθε επανάληψη του αλγορίθμου, όλα τα σωματίδια, παίρνουν καινούρια θέση σύμφωνα με τις εξισώσεις 2.2.1 και 2.2.2. Για να δουλέψουν όμως αυτές οι εξισώσεις χρειάζονται απαραίτητα οι πίνακες KAT και Vpast.

- Ο KAT αναλαμβάνει να κρατάει τη λύση(στην αρχική της μορφή) όλων των σωματιδίων. Είναι λοιπόν ένας δισδιάστατος πίνακας KAT(nodes x particles), που έχει γραμμές αντίστοιχες των κόμβων προς εξυπηρέτηση και στήλες όσα τα άτομα που αποτελούν το σμήνος. Έτσι, κάθε στήλη εκφράζει την παρούσα λύση του εκάστοτε σωματιδίου. Ένα παράδειγμα του KAT παρουσιάζεται αμέσως:

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

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

- Ο Vpast είναι ο πίνακας που κρατάει το στοιχείο $v_{ij}(t)$ των εξισώσεων 2.2.1 και 2.2.2. Ουσιαστικά, δίνει σε κάθε επανάληψη την προηγούμενη θέση του κάθε κόμβου του κάθε σωματιδίου. Είναι τρισδιάστατος πίνακας αφού πρέπει να παρέχει τις x και y συντεταγμένες για όλους τους κόμβους των σωματιδίων. Η μορφή του είναι $Vpast(nodes \times particles \times 2)$. Παρακάτω φαίνεται ο πίνακας Vpast σπασμένος κατά την τρίτη του διάσταση:

Οι τεταγμένες των κόμβων:

15,7	15,0	50,5	55,2	37,7	35,9	37,6	32,5	28,8	25,4	39,0	22,5	6,5	32,0	42,2	17,8	46,0	18,9	41,4	38,0
20,7	25,9	59,9	51,0	33,3	42,1	45,8	47,0	39,4	25,2	42,5	18,6	18,0	29,1	36,0	22,4	47,1	17,6	44,5	40,0
21,0	17,9	42,0	40,1	30,1	52,9	55,5	43,6	45,9	34,0	42,3	14,2	17,6	20,1	32,8	34,4	48,6	29,0	42,5	47,5
34,3	34,5	55,1	50,6	34,9	51,7	47,4	41,9	47,1	33,5	44,2	14,4	30,5	19,2	35,8	37,4	58,6	37,8	46,1	45,1
37,5	31,0	59,2	46,1	33,4	48,9	48,3	45,0	43,4	37,7	42,7	27,2	39,4	31,5	33,2	41,4	54,3	37,6	48,7	47,9
35,6	35,8	61,2	55,1	26,0	52,4	48,0	50,3	43,0	42,5	46,8	36,0	41,6	37,5	41,3	49,3	58,6	41,9	56,3	52,4
32,6	40,1	49,0	49,8	27,8	57,7	47,3	50,0	41,1	35,0	48,9	26,9	36,9	38,0	37,9	38,6	47,6	38,1	52,0	51,8
36,8	39,9	58,6	51,8	17,5	57,3	58,9	56,6	56,1	46,5	46,5	43,4	43,9	40,7	45,0	36,7	55,3	44,2	61,5	52,4
40,9	35,1	58,3	53,9	42,8	63,8	60,9	56,9	47,0	52,5	43,9	47,9	53,4	48,1	48,9	50,9	56,1	52,7	59,5	52,2
35,9	28,7	56,6	61,2	37,5	54,4	52,8	51,2	42,9	53,9	40,3	50,9	58,7	46,6	48,8	44,3	52,2	55,9	49,9	56,4
45,3	44,9	68,4	68,6	44,7	55,8	56,3	50,9	44,4	60,2	45,9	51,8	61,6	49,1	46,5	50,1	60,0	60,1	51,7	49,1
37,6	38,4	66,5	75,8	43,9	51,4	57,3	48,9	50,1	71,1	41,4	46,5	55,7	55,5	54,7	58,1	64,1	58,8	52,0	42,6

42,3	41,5	60,1	73,0	51,4	53,8	59,4	47,4	45,5	64,7	50,5	62,9	62,3	61,1	59,9	57,7	57,3	55,0	46,3	41,9
38,6	43,2	54,4	58,2	44,5	45,9	47,9	48,7	37,8	56,4	35,9	54,6	63,7	55,7	53,4	49,7	48,7	51,2	40,0	31,3
41,9	50,5	56,9	54,3	50,4	43,0	46,8	55,8	48,4	73,2	53,4	67,7	75,0	62,7	61,6	63,1	50,9	62,7	41,9	48,9
43,2	36,4	44,0	44,7	49,5	40,5	46,5	43,0	27,4	51,2	39,0	45,1	62,9	49,2	48,4	60,6	44,9	57,2	42,0	33,3
41,5	28,7	34,9	40,2	40,5	32,4	42,5	35,6	30,3	41,6	37,9	35,5	56,7	56,2	51,4	51,0	37,4	43,7	38,8	30,5
39,9	36,5	27,8	39,7	46,7	26,0	37,2	32,3	36,7	38,8	35,7	24,8	50,1	57,1	44,4	42,8	28,4	40,2	35,8	19,7
38,7	30,1	21,9	40,9	44,6	18,6	28,1	27,0	35,2	31,6	30,8	25,7	44,5	56,3	40,1	34,3	20,0	35,0	26,9	24,4
50,9	41,4	29,8	31,9	43,0	19,6	32,4	30,6	32,2	33,1	35,5	25,9	40,8	42,9	40,9	38,6	13,7	31,0	24,6	29,9
40,2	36,4	20,5	29,3	40,7	16,6	28,1	31,2	32,1	24,6	30,0	25,0	32,6	41,7	44,8	57,3	24,7	25,8	17,1	40,0
43,0	26,5	9,8	22,8	30,1	1,6	21,7	24,0	25,1	13,6	25,5	12,2	21,8	38,0	42,7	38,6	17,8	10,7	14,0	42,8
31,8	34,7	10,3	16,6	35,3	-0,5	18,0	19,5	27,7	8,4	31,4	8,8	22,5	27,6	18,4	25,2	11,8	4,4	16,9	27,6
44,5	44,3	17,9	19,7	44,1	1,7	10,0	21,3	39,5	23,7	35,5	16,9	28,4	39,6	30,5	37,9	29,3	5,5	28,2	28,0
37,2	28,9	10,0	0,1	38,8	2,0	11,8	15,5	37,7	19,9	30,0	5,0	24,7	20,6	24,1	35,5	41,4	9,7	19,9	31,3
32,0	38,6	24,8	10,8	37,8	12,9	18,5	18,2	48,9	20,5	23,6	9,0	30,4	28,4	27,5	49,0	41,1	8,2	8,6	30,9
37,4	38,7	25,5	19,1	40,9	3,6	27,2	11,8	58,0	29,5	21,3	17,4	30,2	31,0	30,7	27,9	31,8	27,6	20,8	43,3
40,6	37,0	19,5	18,1	35,9	27,5	24,6	5,3	59,8	25,9	20,3	24,3	23,6	28,1	22,6	22,3	24,6	28,8	24,6	36,1
46,1	55,0	34,9	31,1	22,8	43,7	35,9	12,4	55,0	35,5	28,3	35,7	30,1	33,7	27,3	17,8	31,5	44,7	29,4	39,5
42,4	55,3	42,7	32,2	32,6	51,9	40,3	32,8	60,0	44,5	33,1	45,2	42,9	40,8	35,3	23,9	37,3	51,6	44,3	38,5
45,3	52,6	56,4	38,3	17,7	36,6	39,0	34,1	60,3	51,2	37,7	59,2	55,1	40,8	45,1	25,0	38,8	64,0	56,5	35,4
48,8	66,6	54,1	45,6	41,2	56,6	35,0	49,5	70,2	60,6	29,1	64,3	51,6	24,6	42,6	48,2	39,9	67,0	47,9	38,1
29,2	60,6	46,2	32,9	52,1	64,3	47,2	63,3	70,5	71,3	42,6	75,1	35,0	23,9	46,2	62,9	64,6	78,2	63,2	46,6
23,4	49,2	51,1	35,5	49,2	64,0	26,9	47,0	65,6	74,3	50,8	73,1	46,9	32,2	37,1	58,6	37,8	68,7	55,2	26,1
30,9	52,7	44,3	26,3	38,5	61,7	21,3	36,1	65,0	72,0	13,7	57,9	34,9	51,5	25,2	51,5	41,0	50,4	61,2	30,6
26,4	35,6	50,6	47,3	35,4	55,6	36,8	47,3	63,4	78,2	38,8	72,3	49,0	55,6	59,7	73,9	69,8	63,6	70,5	21,9
28,0	41,2	63,0	50,0	40,2	44,6	26,2	46,9	44,5	57,8	18,3	53,5	24,3	26,4	29,5	56,0	50,1	33,6	49,1	22,0
16,4	42,8	55,3	37,6	29,1	35,7	23,9	32,1	36,8	51,9	29,4	49,7	22,7	38,9	37,9	66,0	45,5	12,7	31,0	29,1
8,6	26,8	43,3	16,4	27,7	18,0	12,5	19,6	10,4	14,2	11,9	47,2	9,6	17,3	17,2	29,0	27,6	4,6	23,1	15,3
10,6	38,3	28,8	13,8	14,0	2,5	0,7	26,6	8,0	16,0	34,6	51,6	11,4	7,8	12,1	29,7	23,5	3,6	10,8	11,5
11,9	24,6	5,2	19,7	20,3	4,2	18,9	32,2	5,7	4,8	35,8	47,6	1,3	2,6	38,4	17,8	11,4	13,5	1,2	23,7
8,7	53,0	14,9	25,0	24,3	9,8	14,4	26,8	-3,9	6,6	31,8	40,0	22,9	32,9	52,3	15,9	16,1	12,8	-0,8	24,5
25,5	28,4	-6,3	2,8	23,8	-1,8	16,1	31,5	-6,7	1,2	35,0	40,5	8,1	20,2	19,3	7,2	0,1	18,1	2,6	18,1
26,6	44,8	-1,9	4,6	48,9	31,6	10,8	47,7	1,3	0,7	22,8	31,0	20,7	9,0	30,8	0,8	4,2	50,3	5,4	7,4
50,6	39,6	-4,9	3,1	50,9	47,3	31,8	55,2	7,2	7,1	48,1	25,6	33,6	18,5	16,4	-11,4	-3,4	30,2	3,1	-3,3
61,5	30,4	6,7	2,8	15,5	59,0	38,0	1,4	17,4	11,2	34,6	32,0	17,9	32,9	4,9	3,1	7,2	33,8	-7,9	18,9
34,5	4,9	10,6	21,3	17,6	54,1	28,4	-1,2	6,4	21,7	44,8	28,8	34,5	45,7	8,6	8,3	21,3	47,3	50,2	38,9
40,8	0,7	-2,7	25,4	-0,8	24,6	31,8	6,3	-4,6	-9,8	41,0	0,6	41,8	37,8	-5,8	0,4	10,2	18,5	40,1	45,3
42,2	7,7	22,8	40,3	3,5	12,8	52,2	40,0	24,6	32,8	56,2	19,6	45,5	15,3	45,1	21,6	41,9	14,0	51,2	40,0
5,0	7,4	4,7	51,6	1,3	3,1	39,2	31,8	-8,6	-1,8	34,8	-8,6	24,1	14,1	-17,5	-3,8	-1,9	1,6	33,2	22,4

Οι τετμημένες των κόμβων:

22,7	44,8	35,3	42,6	20,4	38,7	59,0	37,2	48,9	34,9	32,2	32,7	63,6	56,1	60,9	51,5	46,8	34,6	32,0	32,5
28,8	52,4	46,6	44,7	27,4	46,5	62,4	40,6	50,5	48,8	41,2	33,9	62,5	56,8	57,3	50,4	50,9	45,2	30,7	41,0
10,1	33,9	49,0	45,8	25,5	41,9	50,6	43,3	47,5	55,3	48,2	33,2	59,6	63,0	61,6	57,4	53,1	42,9	36,4	46,5
31,8	48,2	59,8	34,2	32,1	50,6	53,4	39,7	45,8	50,4	40,4	37,5	52,6	55,6	52,8	49,8	50,9	48,3	29,3	52,8
22,1	30,0	51,1	34,8	36,2	55,2	44,1	31,8	42,8	45,9	37,2	39,2	46,7	52,9	49,9	64,7	43,2	41,8	28,3	52,5
25,6	25,8	42,8	28,9	24,3	43,5	31,1	29,5	31,9	37,0	28,9	41,1	47,5	41,2	38,1	39,6	30,4	31,9	23,5	36,6
26,0	22,1	34,0	32,8	28,6	38,8	32,6	29,8	23,0	31,0	21,5	35,1	39,5	37,6	42,2	38,3	26,1	26,9	28,1	29,9
27,2	16,6	28,3	40,1	32,5	31,2	32,0	37,2	12,9	27,0	20,6	34,5	42,8	43,7	34,7	38,8	28,3	27,0	37,9	25,5
33,0	11,6	24,3	43,9	30,0	29,1	28,1	42,2	13,2	34,6	18,9	39,3	42,6	37,5	36,2	34,8	24,5	27,1	45,5	24,3
35,6	22,2	35,0	50,3	35,2	26,1	25,5	47,9	16,0	44,0	22,6	44,3	40,4	36,9	38,5	42,7	24,0	34,9	47,7	25,6
34,9	30,5	42,8	59,5	39,2	19,5	36,6	54,5	18,9	38,4	27,4	49,2	37,3	34,0	32,9	36,8	30,6	36,3	42,5	19,5
37,8	28,8	41,3	46,4	36,3	9,8	34,2	47,0	20,7	35,9	25,1	49,8	28,0	29,4	29,8	33,9	33,5	36,2	33,9	9,6
38,7	13,4	50,0	54,8	33,7	12,2	31,7	40,7	21,0	26,5	25,4	40,9	13,7	25,9	26,1	24,0	37,8	31,0	37,6	16,2
48,6	22,9	39,3	54,5	34,6	12,9	41,2	34,7	19,2	19,1	22,9	44,1	15,4	21,0	21,4	13,2	40,9	25,2	33,6	8,9
40,2	18,5	46,5	33,5	25,8	12,5	43,7	22,5	13,2	8,6	17,3	30,4	1,8	11,9	10,7	7,8	21,7	31,8	39,5	12,0
31,7	26,6	31,2	13,4	-1,4	-1,6	42,4	20,3	21,1	4,1	17,0	29,8	13,6	30,9	31,0	23,9	27,2	28,6	22,6	1,5
20,6	39,2	27,6	19,2	3,8	15,6	41,2	5,8	14,2	1,1	12,2	33,9	19,8	31,6	28,4	24,4	22,3	29,7	35,7	-0,3
29,6	35,5	53,6	22,6	18,6	24,8	50,5	12,2	15,7	7,8	21,9	39,6	30,1	27,1	19,4	29,1	19,6	47,5	50,1	20,6
21,3	29,9	40,4	24,3	23,3	18,6	32,7	1,3	23,9	7,5	24,4	17,2	28,9	15,5	13,5	31,6	15,4	36,6	44,6	18,5
32,4	40,5	46,6	31,3	25,1	20,4	52,9	17,9	40,5	18,3	31,8	37,6	47,3	14,4	18,9	33,1	17,4	37,7	47,0	20,5
29,4	30,7	29,9	20,2	27,6	22,3	32,8	11,4	33,7	13,9	30,7	24,6	48,5	11,4	16,3	38,8	21,4	55,6	53,9	25,4
29,9	27,6	31,6	25,5	31,6	25,6	29,6	11,7	36,2	21,1	35,9	25,3	35,4	-1,2	12,8	48,4	15,6	50,0	41,5	26,8
30,8	30,0	34,0	26,6	40,6	25,2	46,1	27,0	40,2	35,6	58,7	33,1	53,3	12,2	26,7	52,9	20,9	48,1	44,1	32,3
23,8	34,2	28,8	33,9	37,3	15,3	39,5	15,8	42,5	32,7	66,8	33,4	35,4	10,3	23,8	54,4	9,3	48,9	44,8	34,6
33,4	50,8	43,5	49,3	45,2	19,5	37,4	36,7	45,0	54,5	65,1	27,2	45,4	38,6	31,2	70,9	13,2	43,6	47,8	50,8
40,1	59,3	33,6	50,2	47,7	42,8	56,9	55,2	57,2	70,1	68,5	18,9	33,3	38,3	42,8	75,1	30,8	41,2	47,2	58,5
20,1	60,9	50,6	65,3	48,9	27,7	42,8	44,4	48,9	70,0	66,6	21,0	51,0	45,9	51,3	70,3	37,8	59,4	64,6	77,6
53,3	64,0	58,8	85,2	62,7	56,6	46,1	53,0	50,5	72,3	63,9	24,5	45,2	65,3	48,8	58,5	43,7	64,0	77,1	90,6
53,8	50,4	32,5	68,7	25,1	52,1	57,1	63,0	61,7	75,8	63,1	50,2	41,8	48,5	57,3	54,8	55,8	75,5	60,4	83,6
61,3	46,4	25,8	71,1	51,8	62,5	47,3	69,2	70,9	73,5	60,5	45,6	62,4	67,9	57,9	60,4	56,8	49,9	66,0	89,3
66,2	38,3	47,3	82,1	26,4	44,7	40,2	70,1	71,2	68,1	58,4	48,9	62,5	75,9	52,9	44,2	55,4	58,8	71,6	88,0
59,3	34,3	49,7	70,2	42,5	59,9	27,9	58,6	69,6	58,3	54,5	44,8	52,2	71,3	50,4	51,5	58,3	40,0	41,9	70,8
67,7	23,2	50,5	67,4	59,9	65,0	41,1	64,7	76,6	69,3	66,6	42,4	23,2	50,7	58,9	55,5	72,8	39,8	49,9	74,1
63,8	27,6	61,3	71,1	64,0	63,8	29,7	56,2	79,4	75,2	62,2	48,7	47,3	59,7	59,5	50,0	65,8	41,5	51,2	65,7
53,8	21,3	47,7	63,8	47,1	50,9	24,7	50,3	54,2	40,6	22,0	35,4	28,6	51,5	47,4	30,1	60,3	32,2	34,5	53,4
46,3	45,2	40,2	62,2	49,7	54,7	33,5	49,4	47,4	34,7	23,3	41,8	34,2	40,7	43,4	29,7	53,7	30,6	45,3	33,1
45,0	54,5	55,4	53,7	51,2	58,7	45,8	50,8	59,7	44,8	11,9	43,6	36,4	28,3	31,9	20,1	55,0	28,6	47,1	27,0
43,1	61,3	58,9	40,3	70,7	61,7	56,8	67,1	58,4	54,7	16,7	56,3	50,0	46,4	47,8	45,9	72,4	25,9	43,3	38,5
36,3	72,7	68,8	37,6	76,4	69,5	63,7	63,0	58,1	63,5	45,9	73,2	67,4	55,7	58,5	67,4	84,2	56,9	68,9	41,8
53,6	69,8	59,8	35,7	70,2	64,3	62,8	63,6	52,5	57,3	36,2	47,6	45,3	38,2	48,4	61,2	76,3	48,7	60,2	33,4
54,5	53,5	57,4	19,6	56,9	56,3	42,5	66,9	44,3	53,1	39,1	38,2	54,6	35,3	22,2	42,1	60,1	33,1	56,9	38,9
52,4	57,8	47,9	20,4	48,0	45,6	32,0	54,7	34,7	40,4	40,4	22,7	37,1	60,3	59,6	41,9	64,5	36,9	35,5	52,4
57,1	42,9	44,4	21,8	51,1	39,6	29,4	52,3	29,2	40,1	41,1	29,0	30,4	71,3	50,9	18,2	49,8	32,2	17,0	52,0

42,9	49,4	29,5	14,3	59,0	47,9	22,5	53,4	14,6	23,8	35,3	42,9	7,0	46,3	46,7	10,3	52,3	55,1	18,0	36,0
52,4	46,3	10,8	13,1	48,1	46,9	5,0	62,4	12,1	16,6	51,3	31,4	-0,5	32,9	24,2	-7,9	27,7	27,8	-6,3	50,0
62,5	44,6	1,8	1,2	56,3	57,3	7,8	60,4	8,9	12,4	35,7	35,9	17,4	33,0	50,2	3,3	42,5	24,3	-14,8	18,8
28,9	13,9	-11,8	-4,4	36,1	44,8	-0,5	0,5	4,4	9,2	42,6	26,1	23,6	40,0	39,0	9,4	9,9	40,5	40,4	29,5
34,3	19,0	-13,0	-0,7	16,7	-7,6	22,5	-1,8	16,7	-13,1	50,0	15,8	31,9	34,5	16,2	23,2	2,7	40,3	46,2	43,4
35,6	45,3	1,8	15,7	65,1	47,7	48,7	4,4	34,3	25,6	43,2	42,5	32,2	31,4	36,0	34,9	22,5	43,6	37,4	15,8
14,0	58,3	67,2	23,2	18,2	70,1	70,1	29,5	71,2	67,6	53,0	44,1	64,3	60,9	86,9	52,2	40,1	27,1	77,1	70,9

4. Επιλογή μεταβλητών κατά την υλοποίηση του αλγορίθμου

4.1 Επιλογή των μεταβλητών για το PSO

Στην υλοποίηση του αλγορίθμου, είναι απαραίτητο να ορίσουμε αρκετές μεταβλητές με τρόπο ώστε να τρέχει αποτελεσματικά το PSO. Αυτό γίνεται με τη μέθοδο του πειράματος – σφάλματος. Ουσιαστικά, για να εξετάσουμε μια μεταβλητή, κρατάμε όλες τις άλλες παραμέτρους σταθερές και εκτελούμε ένα επαρκές αριθμό παραδειγμάτων αλλάζοντας μόνο την υπό εξέταση μεταβλητή. Έτσι, ελέγχοντας τα αποτελέσματα που μας δίνει αυτό το πείραμα, είμαστε σε θέση να επιλέξουμε την καταλληλότερη τιμή.

4.1.1 Επιλογή του μεγέθους του σμήνους και του μέγιστου αριθμού επαναλήψεων

Με τον τρόπο που περιγράψαμε πιο πάνω, έγινε το ακόλουθο πείραμα που μας βοήθησε να επιλέξουμε το μέγεθος του σμήνους και τον μέγιστο αριθμό των επαναλήψεων που θα πραγματοποιούνται κάθε φορά που εκτελούμε τον αλγόριθμο. Το πείραμα έγινε για το δεύτερο πρόβλημα και τα αποτελέσματα φαίνονται στον πίνακα:

p02	Επαναλήψεις→										
Σμήνος ↓	1	100	200	300	400	500	600	700	800	900	1000
1	1650	1729	1620	1688	1708	1793	1801	1654	1723	1698	1782
10	1413	1121	1011	1062	926	930	907	879	840	829	815
20	1455	1006	896	880	888	849	832	810	809	805	802
30	1360	894	951	855	860	845	821	813	807	812	803
40	1305	1100	1008	941	899	853	840	809	796	760	756
50	1352	929	785	780	767	779	802	834	842	848	842
60	1280	1008	864	812	798	775	781	763	802	815	812
70	1261	994	916	852	803	757	745	749	742	705	690
80	1253	990	880	837	790	771	769	753	709	693	687
90	1383	906	829	812	759	760	752	738	698	681	685
100	1292	875	815	792	762	876	821	739	685	670	664

Βλέπουμε πως όσο ανεβαίνουν οι επαναλήψεις, το κόστος μειώνεται. Όμως η πιο έντονη βελτίωση γίνεται μέχρι τις 200. Από εκεί μέχρι τις 500 η λύση βελτιώνεται πιο ασθενικά και έπειτα δε φαίνεται να αλλάζει κάτι ιδιαίτερο. Όσο αφορά το μέγεθος του σμήνους, μέχρι τα 20 άτομα υπάρχει μια απότομη μείωση του

κόστους. Από εκεί μέχρι τα 50 παρατηρούμε μια μικρή βελτίωση των λύσεων, ενώ έπειτα ασήμαντες μεταβλητές πραγματοποιούνται. Επομένως, μια καλή πρόταση είναι να χρησιμοποιούμε κάθε φορά που εκτελούμε τον αλγόριθμο, σμήνος με 50 άτομα και μέγιστο αριθμό επαναλήψεων 500.

4.1.2 Επιλογή των μεταβλητών επιτάχυνσης c_1, c_2

Οι μεταβλητές c_1, c_2 παίζουν το ρόλο ενός ελεγκτή. Από αυτές εξαρτάται το πόσο μακριά μπορεί να μεταβεί ένα άτομο κατά τη διάρκεια μιας επανάληψης. Συνήθως, οι τιμές που παίρνουν είναι $c_1 = c_2 = 2$. Παρόλα αυτά μετά από ένα μεγάλο αριθμό δοκιμών, ανακαλύφθηκαν τιμές που στη συγκεκριμένη περίπτωση βοηθούν περισσότερο στην καλυτέρευση των λύσεων. Αυτές είναι οι: $c_1 = c_2 = 0,7$.

4.1.3 Επιλογή των τυχαίων μεταβλητών r_1, r_2

Οι τιμές r_1, r_2 είναι τυχαίες στο διάστημα $[0, 1]$. Ουσιαστικά είναι το στοχαστικό στοιχείο του αλγορίθμου. Όμως στον αλγόριθμό μας χρησιμοποιήθηκε μια άλλη παραλλαγή που η βασική της ιδέα είναι αυτή των συμπληρωματικών. Δηλαδή το άθροισμά τους είναι σταθερό και ίσο με 1. Το r_1 επιλέγεται τυχαία στο διάστημα $[0, 1]$ και το r_2 βρίσκεται από τη σχέση $r_2 = 1 - r_1$. Έτσι, τυχαίο είναι απλά το ποσοστό που συνεισφέρουν τα r_1, r_2 .

4.1.4 Επιλογή του δείκτη αδράνειας w

Ο δείκτης αδράνειας αλλάζει σε κάθε επανάληψη σύμφωνα με τον τύπο:

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} * t$$

Τα w_{max} και w_{min} τα ορίζουμε ίσα με 0,5 και 0 αντίστοιχα, βέβαια πάντα με την ίδια διαδικασία του πειράματος. Το $iter_{max}$ είναι ουσιαστικά ο μέγιστος αριθμός επαναλήψεων που έχει ήδη ορισθεί ίσος με 500. Το t είναι ένας μετρητής που ακολουθεί τις επαναλήψεις. Έτσι, το w αρχίζει ίσο με 0,5 και καταλήγει ίσο με 0 μετά από 500 επαναλήψεις.

4.2 Επιλογή της καταλληλότερης μεθόδου 2-opt με χρήση τυχαίων αρχικών λύσεων στο PSO

Όπως έχουμε ήδη αναφέρει, υπάρχουν αρκετές παραλλαγές του 2-opt, που η καθεμία έχει διαφορετική φιλοσοφία στον τρόπο που ευρίσκει τα τόξα προς αλλαγή. Στην περίπτωσή μας έγινε πειραματισμός με τις τρεις βασικότερες παραλλαγές για να δούμε ποια μας εξυπηρετεί περισσότερο. Για να το

καταφέρουμε, εκτελέσαμε ένα συγκεκριμένο πρόβλημα αρκετές φορές χρησιμοποιώντας κάθε φορά μια παραλλαγή και κρατώντας όλες τις άλλες παραμέτρους σταθερές. Στο τέλος έγινε σύγκριση των αποτελεσμάτων των τριών μεθόδων και επιλογή της καλύτερης.

4.2.1 2-opt με χρήση των δύο μεγαλύτερων τόξων

Ο 2-opt με χρήση των δύο μεγαλύτερων τόξων προσπαθεί να μειώσει γρήγορα και αποτελεσματικά το κόστος στην περίπτωση βέβαια που η αλλαγή των διαδρομών αυτών όντως θα επιφέρει βελτίωση. Ο μέσος όρος του κόστους μετά από πέντε επαναλήψεις είναι 919,8.

4.2.2 2-opt με χρήση του μεγαλύτερου τόξου και ενός τυχαίου

Ο 2-opt που χρησιμοποιεί το μεγαλύτερο τόξο με ένα τυχαία, προσπαθεί να εξαλείψει την μεγαλύτερη διαδρομή. Αντί όμως να χρησιμοποιήσει το δεύτερο μεγαλύτερο, παίρνει ένα τυχαίο, που φυσικά υπάρχει στη διαδρομή. Με αυτήν την παραλλαγή, ο αλγόριθμός μας είναι λίγο πιο αργός και το κόστος ανεβαίνει λίγο στα 1029,4.

4.2.3 2-opt με χρήση δύο τυχαίων τόξων

Η παραλλαγή αυτή είναι λογικό να χρησιμοποιείται, όταν στην περίπτωση που προσπαθούμε να αλλάξουμε το μεγαλύτερο τόξο, δεν παίρνουμε τα επιθυμητά αποτελέσματα. Η δοκιμή στο πρόβλημά μας, έδειξε χειρότερα αποτελέσματα σε σχέση με την πρώτη μέθοδο, αλλά καλύτερα από τη δεύτερη. Ο μέσος όρος του κόστους ήταν 953,2.

Προφανώς μετά την υλοποίηση του πειράματος, έγινε επιλογή της πρώτης μεθόδου. Στον παρακάτω πίνακα παραθέτουμε τα αποτελέσματα σε ένα συνοπτικό πίνακα:

Μέθοδος->	2-opt_1	2-opt1_2	2-opt_3
	846	919	822
	1017	966	885
	944	1031	1128
	881	993	1002
	911	1238	929
Μέσος όρος	919,8	1029,4	953,2

4.3 Επιλογή της καταλληλότερης μεθόδου 2-opt με χρήση βελτιωμένων αρχικών λύσεων στο PSO

Η διαδικασία που θα ακολουθήσουμε είναι αντίστοιχη με του 4.2, απλά σε αυτήν την περίπτωση στα πειράματα θα εκτελούμε τον αλγόριθμο χρησιμοποιώντας βελτιωμένες αρχικές τιμές.

1. 2-opt με χρήση των δύο μεγαλύτερων τόξων
2. 2-opt με χρήση του μεγαλύτερου τόξου και ενός τυχαίου
3. 2-opt με χρήση δύο τυχαίων τόξων

Μέθοδος->	2-opt_1	2-opt1_2	2-opt_3
	572	610	581
	576	596	559
	568	571	549
	566	609	578
	571	598	569
Μέσος όρος	570,6	596,8	567,2

Από τα αποτελέσματα, εξακριβώνουμε εύκολα ότι κατά την εκτέλεση του αλγορίθμου με βελτιωμένες αρχικές τιμές, θα χρησιμοποιούμε όποτε είναι απαραίτητο τον 2-opt με τη μέθοδο των δύο τυχαίων τόξων.

5. Υλοποίηση και εκτέλεση του αλγορίθμου και παρουσίαση των αποτελεσμάτων

Μετά από τη διαδικασία της επιλογής όλων των μεταβλητών που θα μας χρησίμευαν, ήρθε η ώρα της εκτέλεσης του αλγορίθμου. Όπως έχει γίνει φανερό, το πρόγραμμα αποτελείται από αρκετά υποπρογράμματα. Έτσι, μπορούμε να δώσουμε λύση στο πρόβλημά μας με διάφορες παραλλαγές για να αποφασίσουμε ποια από αυτές είναι καλύτερη.

Αρχικά θα δώσουμε την εκτέλεση και τη λύση ενός προβλήματος ολοκληρωμένα, για να γίνει πιο κατανοητή η διαδικασία και κατόπιν θα παραθέσουμε συνοπτικά τις λύσεις που μας έδωσε ο αλγόριθμος με διάφορες παραλλαγές του. Σημειώνεται πως σε όλα τα αποτελέσματα αναγράφεται ο μέσος όρος των αποτελεσμάτων μετά από εκτέλεση περίπου 10 πειραμάτων κάθε φορά.

5.1 Ένα ολοκληρωμένο παράδειγμα

Θα ασχοληθούμε με το πρόβλημα p02 με τη χρήση του οποίου κάναμε και την επιλογή των παραμέτρων σε προηγούμενο κεφάλαιο. Αφού εισάγουμε τα δεδομένα και τις μεταβλητές με τις τιμές που έχουμε ήδη αποφασίσει, εκτελούμε τον αλγόριθμο με βελτιωμένες αρχικές τιμές και χωρίς τη χρήση του 2-opt.

Αρχικά παίρνουμε μια διαδρομή που φαίνονται απλά οι κόμβοι που θα εξυπηρετηθούν με τη σειρά, στην περίπτωση που ήταν μόνο ένα διαθέσιμο όχημα. Ο παρακάτω πίνακας διαβάζεται κατά στήλες και μας δίνει αυτήν ακριβώς την ακολουθία.

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

Ακολουθεί η διαδικασία ενσωμάτωσης των αποθηκών στη διαδρομή. Ο πίνακας πιο κάτω διαβάζεται και πάλι κατά στήλες και περιέχει πλέον τη διαδρομή

με τις αποθήκες. Με αρνητικές τιμές φαίνονται οι αποθήκες, όπου για παράδειγμα η -1 είναι η πρώτη αποθήκη, η -2 είναι η δεύτερη και ούτω καθεξής.

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

Πλέον είμαστε έτοιμοι να κατακερματίσουμε την αλληλουχία των κόμβων σε διαδρομές. Έτσι δημιουργούμε τον παρακάτω πίνακα, που περιλαμβάνει τις απαραίτητες διαδρομές που πρέπει να γίνουν. Αυτές ξεκινούν και τελειώνουν πάντα σε αρνητικό κόμβο που υποδηλώνει αποθήκη.

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

Αυτές οι 5 διαδρομές πρέπει τώρα να κατανεμηθούν σε όλα τα διαθέσιμα οχήματα, που στο παρών πρόβλημα είναι δύο. Έτσι, το πρώτο φορτηγό θα αναλάβει τις τρεις πρώτες και το δεύτερο τις άλλες δύο.

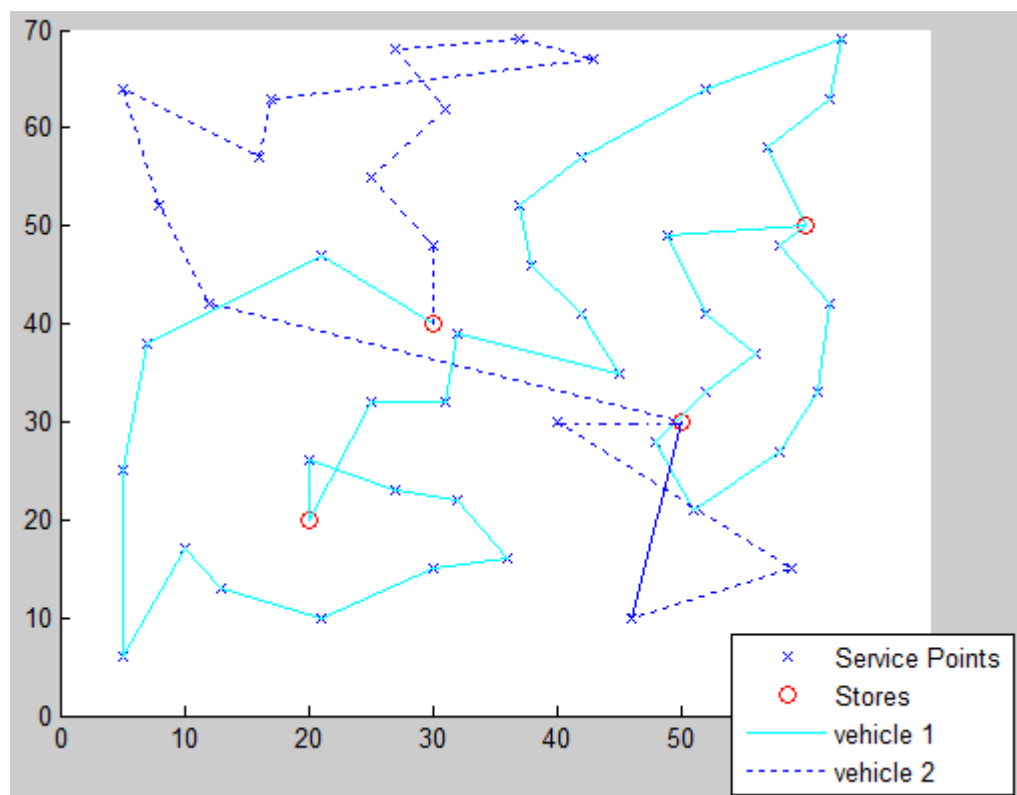
- Πρώτο όχημα:

-4	-4	-1
2	20	4
16	35	17
50	36	37
9	3	15
49	22	44
10	1	42
30	32	19
34	11	41
21	38	40
29	46	13
-4	12	25
-	47	6
-	-1	-2

- Δεύτερο όχημα:

-2	-3
27	18
48	39
8	45
26	5
31	33
28	-3
7	-
23	-
43	-
24	-
14	-
-3	-

Τέλος, δίνουμε το εξαγόμενο γράφημα, που μας δείχνει τις διαδρομές που ακολουθούν τα οχήματα ώστε να εξυπηρετήσουν τα σημεία διανομής.



5.2 Εκτέλεση αλγορίθμου με τυχαίες αρχικές λύσεις

Σε αυτό το σημείο, θα εκτελέσουμε τον αλγόριθμο με τυχαίες αρχικές λύσεις για τις περιπτώσεις που κάνουμε ή όχι χρήση του βοηθητικού αλγορίθμου 2-opt. Έτσι θα δούμε αν τελικά και κατά πόσο βοηθάει στην εύρεση καλύτερων λύσεων η ενσωμάτωση του αλγορίθμου 2-opt. Τα αποτελέσματα του πειράματος παρουσιάζονται στους συγκεντρωτικούς πίνακες που ακολουθούν.

- Χωρίς 2-opt

Πρόβλημα	Βέλτιστα	Αποτελέσματα	Απόκλιση
p01	576,87	1103,25	91,25
p02	473,53	945,87	99,75

p03	641,19	1249,15	94,82
p04	1001,59	1934,80	93,17
p05	750,03	1441,18	92,15
p06	876,50	1700,56	94,02
p07	885,80	1713,18	93,40
p08	4437,68	8764,31	97,50
p09	3900,22	7597,75	94,80
p10	3663,02	7158,29	95,42
p11	3554,18	7023,10	97,60
p12	1318,95	2516,39	90,79
p13	1318,95	2539,77	92,56
p14	1360,12	2666,33	96,04
p15	2505,42	4920,26	96,38
p16	2572,23	4961,98	92,91
p17	2709,09	5181,60	91,27
p18	3702,85	7139,69	92,82
p19	3827,06	7304,84	90,87
p20	4058,07	7714,53	90,10
p21	5474,84	10778,26	96,87
p22	5702,16	11048,19	93,75
p23	6095,46	11591,12	90,16
pr01	861,32	1699,44	97,31
pr02	1307,61	2580,92	97,38
pr03	1806,60	3453,57	91,16
pr04	2072,52	4126,23	99,09
pr05	2385,77	4647,09	94,78
pr06	2723,27	5209,25	91,29
pr07	1089,56	2137,48	96,18
pr08	1666,60	3308,10	98,49
pr09	2153,10	4186,89	94,46
pr10	2921,85	5733,11	96,22

- Με 2-opt

Πρόβλημα	Βέλτιστα	Αποτελέσματα	Απόκλιση
p01	576,87	970,64	68,26
p02	473,53	817,69	72,68
p03	641,19	1124,39	75,36
p04	1001,59	1638,20	63,56
p05	750,03	1279,25	70,56
p06	876,50	1507,40	71,98
p07	885,80	1577,26	78,06
p08	4437,68	7308,42	64,69
p09	3900,22	6535,60	67,57

p10	3663,02	6553,16	78,90
p11	3554,18	6074,64	70,92
p12	1318,95	2266,31	71,83
p13	1318,95	2218,17	68,18
p14	1360,12	2409,88	77,18
p15	2505,42	4407,62	75,92
p16	2572,23	4303,18	67,29
p17	2709,09	4830,05	78,29
p18	3702,85	6427,40	73,58
p19	3827,06	6744,77	76,24
p20	4058,07	7198,13	77,38
p21	5474,84	9011,01	64,59
p22	5702,16	9790,57	71,70
p23	6095,46	10438,91	71,26
pr01	861,32	1418,81	64,73
pr02	1307,61	2212,71	69,22
pr03	1806,60	3081,05	70,54
pr04	2072,52	3583,65	72,91
pr05	2385,77	4037,69	69,24
pr06	2723,27	4581,07	68,22
pr07	1089,56	1898,31	74,23
pr08	1666,60	2759,06	65,55
pr09	2153,10	3669,39	70,42
pr10	2921,85	5224,64	78,81

5.3 Εκτέλεση του αλγορίθμου με καλύτερες αρχικές λύσεις

Εδώ γίνεται μια προσπάθεια υποβοήθησης του αλγορίθμου, δίνοντάς του τη δυνατότητα να ξεκινάει την αναζήτηση από πιο καλές(λογικές) αρχικές λύσεις. Η βασική του συνίσταται στο ότι ένα όχημα θα ήταν καλύτερο κάθε φορά που έχει εξυπηρετήσει ένα κόμβο, να πηγαίνει στον αμέσως κοντύτερο, παρά σε ένα τυχαίο, όπως γινόταν προηγουμένως. Εν συντομία θα παρουσιάσουμε τη ρουτίνα που φτιάχνει τις αρχικές λύσεις των σωματιδίων. Αρχικά επιλέγουμε τυχαία το πρώτο σημείο εξυπηρέτησης. Έπειτα βρίσκουμε τον κοντινότερο διαθέσιμο κόμβο και στέλνουμε εκεί το όχημα. Όταν λέμε διαθέσιμο, εννοούμε ότι ο κόμβος αυτός δεν έχει ήδη εξυπηρετηθεί. Έτσι συνεχίζουμε μέχρι να εξυπηρετηθούν όλοι οι κόμβοι. Ένα πρόβλημα δημιουργούνται στην περίπτωση που οι κόμβοι ήταν λίγοι σε σχέση με τον αριθμό των σωματιδίων του σμήνους. Τότε αναπόφευκτα, κάποια σωματίδια θα τύχαινε να έχουν την ίδια ακριβώς αρχική λύση, αφού αν ο πρώτος κόμβος που επιλεγόταν για αυτά ήταν ίδιος, τότε και η σειρά των υπολοίπων θα ήταν υποχρεωτικά ίδια. Ακριβώς γι αυτόν το λόγο ο αλγόριθμος παραλλάχθηκε λίγο και πλέον μετά την τυχαία επιλογή του πρώτου κόμβου ακολουθεί η παρακάτω

διαδικασία. Βρίσκουμε τους δύο πιο κοντινούς κόμβους. Με πιθανότητα 90% στέλνουμε το όχημα στον αμέσως πιο κοντινό του κόμβο και με πιθανότητα 10% στον δεύτερο. Αυτό συνεχίζεται και πάλι μέχρι να εξυπηρετηθούν όλοι οι κόμβοι. Με αυτό το τέχνασμα και πάλι βρίσκουμε πολύ καλές αρχικές λύσεις και ταυτόχρονα μπορούμε να χρησιμοποιήσουμε ένα πολύ μεγάλο σμήνος ακόμα και σε προβλήματα με ελάχιστους κόμβους. Η διαδικασία της ενσωμάτωσης των αποθηκών στις λύσεις και ο καταμερισμός της διαδρομής σε όλα τα διαθέσιμα οχήματα, έχει ήδη περιγραφεί στο τρίτο κεφάλαιο.

Εδώ θα ακολουθήσουμε αντίστοιχη διαδικασία με την περίπτωση του αλγορίθμου με τυχαίες αρχικές λύσεις. Τα αποτελέσματα παρουσιάζονται ακολούθως:

- Χωρίς 2-opt

Πρόβλημα	Βέλτιστα	Αποτελέσματα	Απόκλιση
p01	576,87	660,98	14,58
p02	473,53	544,00	14,88
p03	641,19	741,40	15,63
p04	1001,59	1140,60	13,88
p05	750,03	920,40	22,72
p06	876,50	1006,20	14,80
p07	885,80	1015,00	14,59
p08	4437,68	5063,20	14,10
p09	3900,22	4610,80	18,22
p10	3663,02	4222,60	15,28
p11	3554,18	4281,60	20,47
p12	1318,95	1499,00	13,65
p13	1318,95	1478,40	12,09
p14	1360,12	1452,80	6,81
p15	2505,42	2772,50	10,66
p16	2572,23	2864,44	11,36
p17	2709,09	3060,73	12,98
p18	3702,85	4304,19	16,24
p19	3827,06	4518,99	18,08
p20	4058,07	4627,42	14,03
p21	5474,84	6240,22	13,98
p22	5702,16	6570,60	15,23
p23	6095,46	6831,18	12,07
pr01	861,32	1016,62	18,03
pr02	1307,61	1485,71	13,62
pr03	1806,60	2066,03	14,36
pr04	2072,52	2400,39	15,82
pr05	2385,77	2708,80	13,54

pr06	2723,27	3164,17	16,19
pr07	1089,56	1284,59	17,9
pr08	1666,60	1830,76	9,85
pr09	2153,10	2434,51	13,07
pr10	2921,85	3299,35	12,92

- Με 2-opt

Πρόβλημα	Βέλτιστα	Αποτελέσματα	Απόκλιση
p01	576,87	688,62	19,37
p02	473,53	560,67	18,40
p03	641,19	754,40	17,66
p04	1001,59	1191,06	18,92
p05	750,03	883,26	17,76
p06	876,50	1049,55	19,74
p07	885,80	1043,82	17,84
p08	4437,68	5157,74	16,23
p09	3900,22	4657,16	19,41
p10	3663,02	4390,01	19,85
p11	3554,18	4222,68	18,81
p12	1318,95	1548,61	17,41
p13	1318,95	1555,11	17,91
p14	1360,12	1610,31	18,39
p15	2505,42	2967,39	18,44
p16	2572,23	2994,69	16,42
p17	2709,09	3217,47	18,77
p18	3702,85	4433,29	19,73
p19	3827,06	4497,87	17,53
p20	4058,07	4780,33	17,80
p21	5474,84	6367,66	16,31
p22	5702,16	6798,86	19,23
p23	6095,46	7179,01	17,78
pr01	861,32	1008,36	17,07
pr02	1307,61	1516,89	16,00
pr03	1806,60	2095,84	16,01
pr04	2072,52	2448,32	18,13
pr05	2385,77	2791,71	17,02
pr06	2723,27	3168,12	16,33
pr07	1089,56	1280,91	17,56
pr08	1666,60	1949,20	16,96

pr09	2153,10	2520,96	17,09
pr10	2921,85	3496,83	19,68

5.4 Συγκεντρωτικός πίνακας και σχόλια

Συγκεντρωτικός πίνακας

Μέθοδος	Απόκλιση
με καλές αρχικές και 2-opt	17,93%
με καλές αρχικές χωρίς 2- opt	14,59%
με τυχαίες αρχικές και 2- opt	71,51%
με τυχαίες αρχικές χωρίς 2- opt	94,39%

Παρατηρούμε εύκολα ότι υπάρχουν τεράστιες αποκλίσεις στα αποτελέσματα, ανάλογα με τις μεθόδους που χρησιμοποιούμε. Το περίεργο είναι ότι ο αλγόριθμος 2-opt φαίνεται να ευνοεί τις λύσεις στην περίπτωση που λύνουμε το πρόβλημα με τυχαίες αρχικές λύσεις, ενώ δυσχεραίνει την κατάσταση όταν το λύνουμε με βελτιωμένες αρχικές λύσεις. Μια πιθανή εξήγηση είναι ότι στη δεύτερη περίπτωση, ο 2-opt δεν αφήνει το PSO να αλλάξει ελεύθερα λύσεις στις πρώτες επαναλήψεις και έτσι να εγκλωβίζεται γρήγορα σε κάποιο υποβέλτιστο.

6. Συμπεράσματα

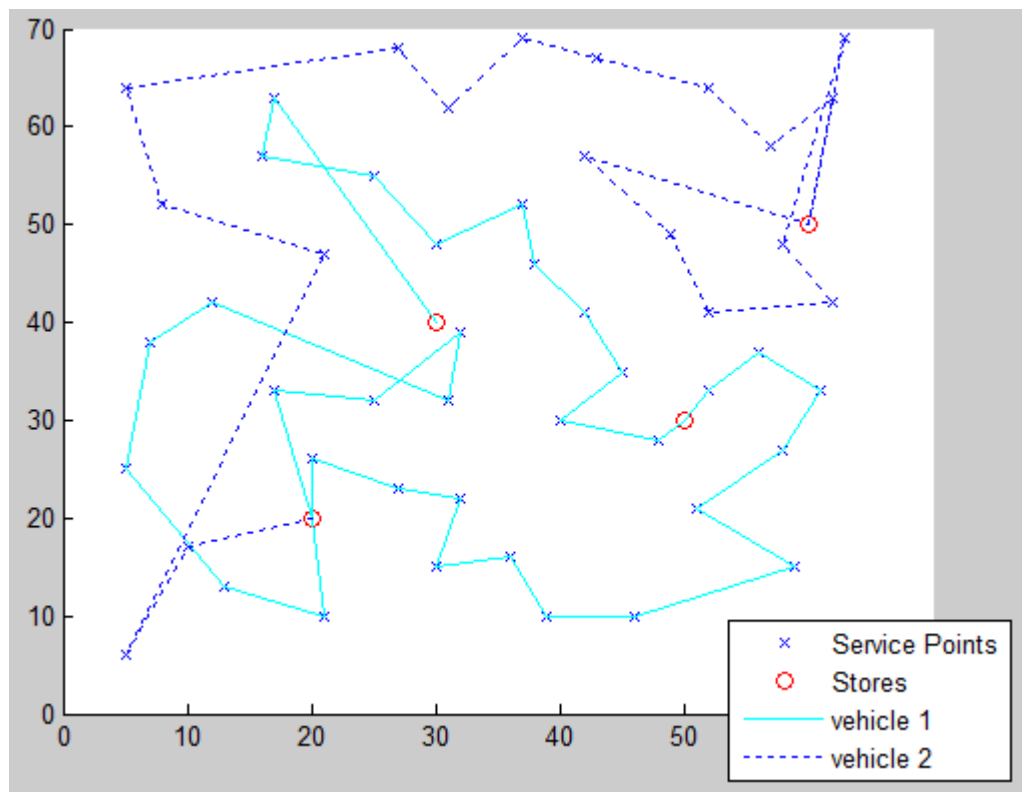
Στο τελείωμα αυτής της εργασίας και κάνοντας μια αναδρομή σε όλη την πορεία της, μπορούμε να εξάγουμε κάποια χρήσιμα συμπεράσματα.

Κατ' αρχήν, ο αλγόριθμος είναι εξαιρετικά γρήγορος και γι' αυτό μπορεί να ενσωματωθεί σε δυναμικά προβλήματα όπου απαιτείται άμεση λύση ακόμα και με εύρεση υποβέλτιστων. Τέτοιες περιπτώσεις δεν είναι σπάνιες. Μάλιστα είναι η συνήθης τακτική σε δυναμικά προβλήματα με μεγάλο αριθμό κόμβων, όπου έτσι κι αλλιώς δεν μπορούν να λυθούν με μεθόδους που προσπαθούν να βρουν το ολικό βέλτιστο.

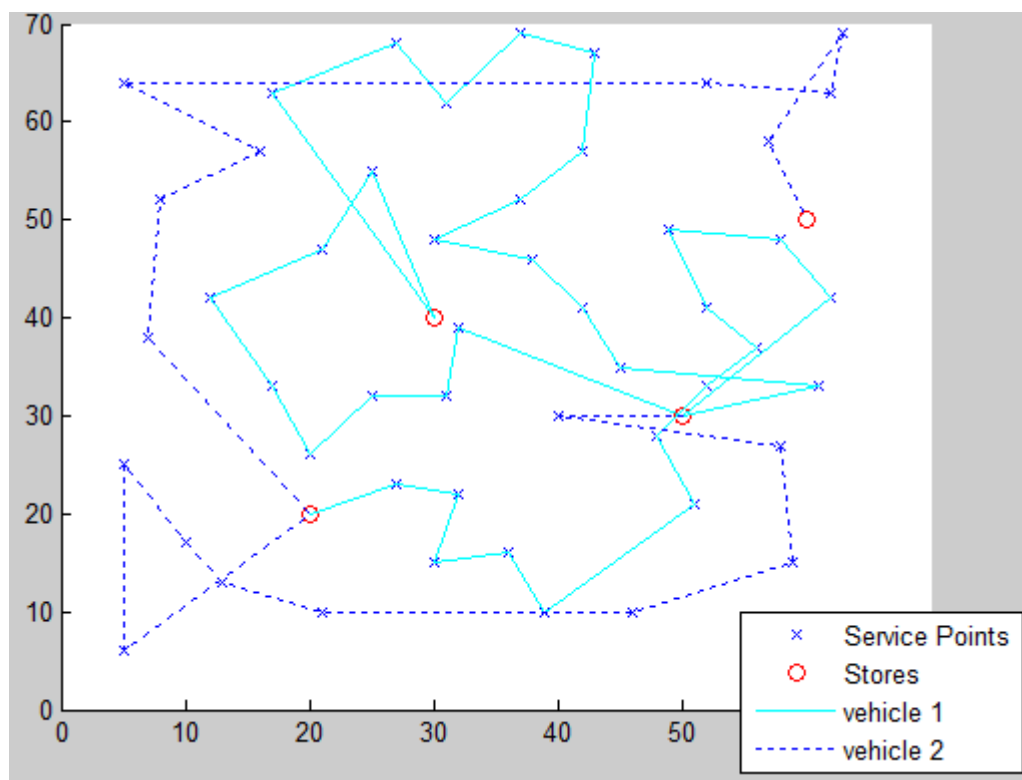
Επίσης, η αρχική μοντελοποίηση και οι ευρετικές μέθοδοι που χρησιμοποιούνται για την αντιμετώπιση συγκεκριμένων προβλημάτων που ανακύπτουν μπορεί να αλλάξουν πολύ τα αποτελέσματα. Κατά την υλοποίηση του αλγορίθμου, δοκιμάσαμε πολλές παραλλαγές στις μεθόδους και στις παραμέτρους, ώστε να ανακαλύψουμε ποια φόρμουλα είναι αυτή που μας εξυπηρετεί περισσότερο ανά περίπτωση. Σε αυτές τις δοκιμές, εμφανίστηκαν αξιοσημείωτες διαφορές από περίπτωση σε περίπτωση. Οι αποκλίσεις από τα γνωστά μέχρι σήμερα βέλτιστα κυμαινόταν από 6,81% στις ευνοϊκότερες, μέχρι και 99,75% στις δυσμενέστερες συνθήκες εκτέλεσης του αλγορίθμου.

Στα επόμενα σχήματα φαίνεται οπτικά, πως μπορούν οι μέθοδοι που επιλέγουμε στην επίλυση του αλγορίθμου να επηρεάσουν το τελικό αποτέλεσμα.

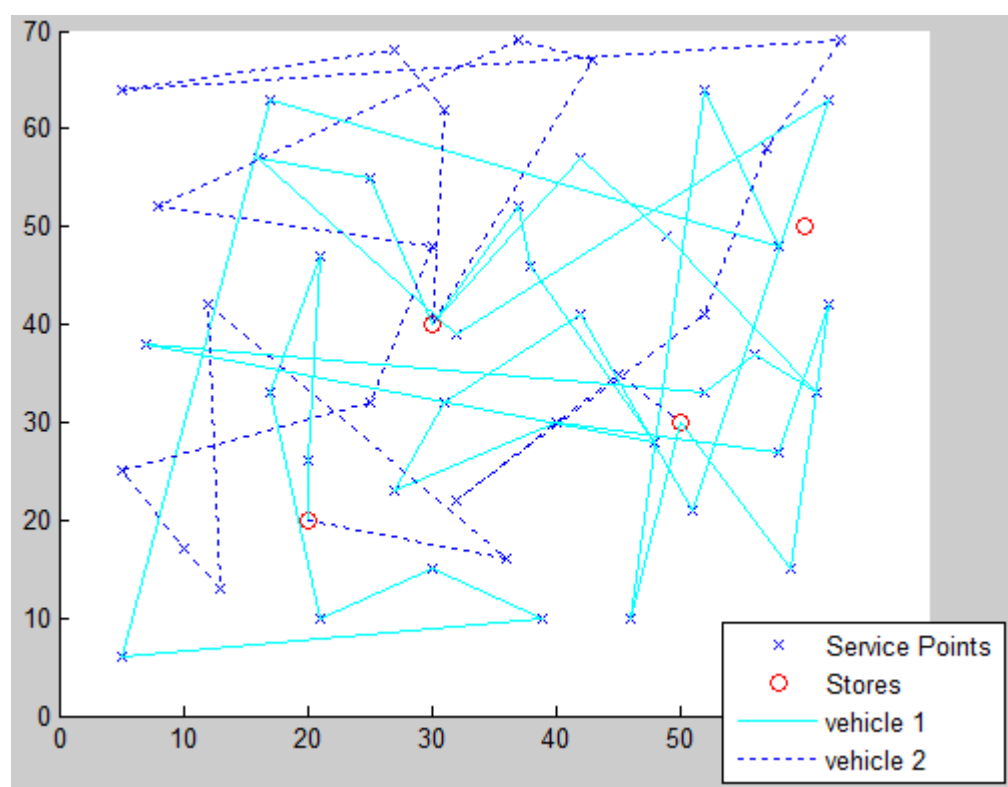
- Με κόστος 570.



- Με κόστος 629.



- Με κόστος 1194.



Βιβλιογραφία

- [1] Ιωάννης Μαρινάκης, Αθανάσιος Μυγδαλάς. Σχεδιασμός και Βελτιστοποίηση της Εφοδιαστικής Αλυσίδας. Εκδόσεις σοφία, 2008
- [2] Tonci Caric and Hrvoje Gold. The vehicle routing problem, 2008
- [3] Paolo Toth, Daniele Vigo. The vehicle routing problem
- [4] Yupo Chan, William B. Carter, Michael D. Burnes. A multiple-depot, multiple-vehicle, location-routing problem with stochastically processed demands, 1999
- [5] Eric D. Taillard, Gilbert Laporte¹ and Michel Gendreau. Vehicle routing with multiple use of vehicles, 1995
- [6] James Kennedy and Russell C. Eberhart. Swarm Intelligence
- [7] Konstantinos E. Parsopoulos, Michael N. Vrahatis. Particle Swarm Optimization and Intelligence: Advances and Applications, 2010
- [8] Maurice Clerc. Particle Swarm Optimization, 2006
- [9] Ιορδανίδου Γεωργία – Ρουμπίνη. Επίλυση προβλήματος δρομολόγησης οχημάτων με στοχαστική ζήτηση με χρήση του αλγορίθμου βελτιστοποίησης σμήνους σωματιδίων (PSO). Μεταπτυχιακή εργασία στο πολυτεχνείο Κρήτης, τμήμα ΜΠΔ, 2011