ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Επίλυση του Προβλημάτων P-Μέσων και Χωροθέτησης Εγκαταστάσεων Περιορισμένης Χωρητικότητας με τον

Αλγόριθμο Βελτιστοποίησης Αποικίας Μυρμηγκιών

Συγγραφέας:

ΓΕΩΡΓΙΟΣ ΞΥΔΙΑΣ

Επιβλέπων Καθηγητής:

ΔΡ. ΣΤΑΥΡΑΚΑΚΗΣ ΓΙΩΡΓΟΣ

Μέλη επιτροπής:

ΔΡ. ΜΑΡΙΝΑΚΗΣ ΙΩΑΝΝΗΣ

ΔΡ. ΣΤΑΥΡΟΥΛΑΚΗΣ ΠΕΤΡΟΣ

Νοέμβριος 2012

ΕΥΧΑΡΙΣΤΙΕΣ

Δράττομαι της ευκαιρίας για να ευχαριστήσω προσωπικά τον επιβλέποντα καθηγητή μου κύριο Μαρινάκη Ιωάννη, που μου έδωσε την ευκαιρία να εργασθώ σε ένα τόσο ενδιαφέρον αλλά και δυναμικά εξελισσόμενο ερευνητικό πεδίο.

Ακολούθως θα ήθελα να αφιερώσω αυτή την διπλωματική εργασία στην οικογένεια μου και ιδιαίτερα στην μητέρα μου Μαρία ως ελάχιστη ένδειξη ευγνωμοσύνης για την αμέριστη συμπαράσταση που μου παρείχε καθόλη τη διάρκεια των σπουδών μου στο Πολυτεχνείο Κρήτης.

Τέλος, ευχαριστώ όλους μου τους φίλους για την ψυχολογική υποστήριξη που μου παρείχαν συνεχώς όλα αυτά τα χρόνια. Ιδιαίτερη μνεία αξίζει στον μοναδικό Φίλο συνάδελφο, Μαρκόπουλο Παναγιώτη, χωρίς τον οποίο η περάτωση των υποχρεώσεών μου στην σχολή θα ήταν αδύνατη.

ΠΕΡΙΛΗΨΗ

Ο αλγόριθμος Βελτιστοποίησης Αποικίας Μυρμηγκιών, οφείλει την ύπαρξή του στη φύση και ειδικότερα στον τρόπο εντοπισμού τροφής από τα μυρμήγκια. Τα μυρμήγκια ακολουθούν τυχαίες διαδρομές και όταν εντοπίσουν τροφή επιστρέφουν στην αποικία τους, μεταφέροντάς την, και αφήνοντας κατά τη διαδρομή αυτή ίχνη φερομόνης. Εάν άλλα μυρμήγκια ανακαλύψουν κάποιο τέτοιο μονοπάτι, τότε τείνουν να το ακολουθούν, επιβάλλοντάς το. Με το πέρασμα του χρόνου επικρατούν τα συντομότερα μονοπάτια, αφού εκεί η φερομόνη που εξατμίζεται, αντικαθίσταται συντομότερα από νέες ποσότητες που αφήνουν τα διερχόμενα μυρμήγκια. Ο αλγόριθμος Βελτιστοποίησης Αποικίας Μυρμηγκιών, προσομοιώνει αυτή τη συμπεριφορά με «τεχνητά μυρμήγκια» που κινούνται μέσα σε ένα γράφο που αναπαριστά το προς επίλυση πρόβλημα.

Στην παρούσα εργασία παρουσιάζουμε μία μέθοδο εφαρμογής του αλγόριθμου Βελτιστοποίησης Αποικίας Μυρμηγκιών για την επίλυση του προβλήματος p-median. Ο αλγόριθμος αυτός εφαρμόζει μια σειρά τροποποιήσεων στο κλασσικό πρόβλημα δρομολόγησης οχημάτων, ώστε να κάνει εφικτή την αναζήτηση της βέλτιστης λύσης του προβλήματος p-median. Επίσης, παρουσιάζουμε ένα δεύτερο αλγόριθμο Βελτιστοποίησης Αποικίας Μυρμηγκιών, ο οποίος υλοποιεί την παραλλαγή του βασικού προβλήματος χωροθέτησης εγκαταστάσεων πεπερασμένης χωρητικότητας κατά την οποία κάθε πελάτης μπορεί να εξυπηρετηθεί από μία μόνο εγκατάσταση.

Τα αποτελέσματα των προτεινόμενων αλγορίθμων που παρουσιάζονται, είναι αξιοσημείωτα και συγκρίσιμα με τα βέλτιστα μιας σειράς ενδεικτικών προβλημάτων (test problems) που χρησιμοποιήθηκαν ως προβλήματα σύγκρισης (benchmarks) για να καταδείξουν την αποτελεσματικότητα των προτεινόμενων αλγορίθμων.

ΠΕΡΙΕΧΟΜΕΝΑ

1. ΕΙΣΑΓΩΓΗ 9

2. ΤΟ ΠΡΟΒΛΗΜΑ ΕΥΡΕΣΗΣ P-ΜΕΣΩΝ (P-MEDIAN PROBLEM) 12

2.1. ΜΑΘΗΜΑΤΙΚΗ ΔΙΑΤΥΠΩΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ 13

2.2. ΔΙΑΤΥΠΩΣΗ ΑΚΕΡΑΙΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ (INTEGER PROGRAMMING FORMULATION) 18

2.3. ΤΡΟΠΟΙ ΕΠΙΛΥΣΗΣ ΤΩΝ ΠΡΟΒΛΗΜΑΤΩΝ P-MEDIAN 19

3. ΤΟ ΠΡΟΒΛΗΜΑ ΧΩΡΟΘΕΤΗΣΗΣ ΕΓΚΑΤΑΣΤΑΣΕΩΝ ΠΕΠΕΡΑΣΜΕΝΗΣ ΧΩΡΗΤΙΚΟΤΗΤΑΣ (CAPACITATED FACILITY LOCATION PROBLEM) 21

3.1. ΤΟ ΠΡΌΒΛΗΜΑ UFLP 21

3.2. ΜΑΘΗΜΑΤΙΚΗ ΔΙΑΤΥΠΩΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ 22

4. ΑΛΓΟΡΙΘΜΟΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΑΠΟΙΚΙΑΣ ΜΥΡΜΗΓΚΙΩΝ (ANT COLONY OPTIMIZATION - ACO) 25

4.1. ΕΙΣΑΓΩΓΗ ΚΑΙ ΒΑΣΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΑΛΓΟΡΙΘΜΟΥ ACO 25

4.2. ΕΠΙΜΕΡΟΥΣ ΒΗΜΑΤΑ ΑΛΓΟΡΙΘΜΟΥ ACO 27

4.2.1. ΔΗΜΙΟΥΡΓΙΑ ΛΥΣΕΩΝ (ΕΠΙΛΟΓΗ ΔΙΑΔΡΟΜΩΝ) 28

4.2.2. ΒΟΗΘΗΤΙΚΕΣ ΕΝΕΡΓΕΙΕΣ (DEAMON ACTIONS) 29

4.2.3. ΑΝΑΝΕΩΣΗ ΦΕΡΟΜΟΝΗΣ 29

4.3. ΠΑΡΑΛΛΑΓΕΣ ΑΛΓΟΡΙΘΜΟΥ ACO 30

4.4. ΣΥΓΚΛΙΣΗ ΑΛΓΟΡΙΘΜΟΥ ACO 31

4.5. ΕΦΑΡΜΟΓΕΣ ΑΛΓΟΡΙΘΜΟΥ ACO 31

5. ΕΦΑΡΜΟΓΗ ΑΛΓΟΡΙΘΜΟΥ ACO ΣΤΟ ΠΡΟΒΛΗΜΑ *P*-MEDIAN 35

5.1. ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ *P*-MEDIAN 37

5.2. ΕΠΙΜΕΡΟΥΣ ΒΗΜΑΤΑ ΑΛΓΟΡΙΘΜΟΥ 39

5.3. ΥΛΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΟΥ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΚΤΕΛΕΣΗΣ 43

6. ΕΦΑΡΜΟΓΗ ΑΛΓΟΡΙΘΜΟΥ ACO ΣΤΟ ΠΡΟΒΛΗΜΑ CFL 57

6.1. ΟΡΙΣΜΟΣ ΤΟΥ ΜΟΝΤΕΛΟΥ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ 57

6.2. ΕΠΙΜΕΡΟΥΣ ΒΗΜΑΤΑ ΑΛΓΟΡΙΘΜΟΥ 61

6.3. ΥΛΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΟΥ – ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΚΤΕΛΕΣΗΣ 65

7. ΣΥΜΠΕΡΑΣΜΑΤΑ **97**

8. ΒΙΒΛΙΟΓΡΑΦΙΑ **99**

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 1: Κίνηση τεχνητών μυρμηγκιών σε ένα γράφο. 26

Σχήμα 2a: Λύση μετασχηματισμένου προβλήματος. 38

Σχήμα 2b: Λύση αρχικού προβλήματος. 39

Σχήμα 3: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 1. 45

Σχήμα 4: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 2. 46

Σχήμα 5: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 3. 46

Σχήμα 6: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 4. 47

Σχήμα 7: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 5. 47

Σχήμα 8: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 6. 48

Σχήμα 9: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 7. 48

Σχήμα 10: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 8. 49

Σχήμα 11: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 9. 49

Σχήμα 12: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 10. 50

Σχήμα 13: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 11. 50

Σχήμα 14: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 12. 51

Σχήμα 15: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 13. 51

Σχήμα 16: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 14. 52

Σχήμα 17: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 15. 52

Σχήμα 18: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 16. 53

Σχήμα 19: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 17. 53

Σχήμα 20: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 18. 54

Σχήμα 21: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 19. 54

Σχήμα 22: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 20. 55

Σχήμα 23: Πρόβλημα χωροθέτησης εγκαταστάσεων πεπερασμένης χωρητικότητας από μία πηγή 58

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Κύρια Χαρακτηριστικά προβλημάτων P-median. 44

Πίνακας 2: Λύσεις προβλήματος P-median και σύγκριση με τις βέλτιστες λύσεις. 56

Πίνακας 3: Παράμετροι αλγορίθμου ACO για το πρόβλημα CFL. 65

Πίνακας 4: Λύσεις προβλήματος CFL και σύγκριση με τις βέλτιστες λύσεις (a=1.5). 66

1. Εισαγωγη

Ονομάζουμε Ανάλυση Χωροθέτησης (Location Analysis) τη διαδικασία κατά την οποία σε κάποιο περιβάλλον, εγκαθιστούμε κέντρα παροχής υπηρεσιών (εξυπηρέτησης) έτσι ώστε να καλύψουμε τις ανάγκες των χρηστών των κέντρων αυτών (ζήτηση) κατά τον «καλύτερο δυνατό τρόπο». Όπου ο «καλύτερος δυνατός τρόπος» επιτυγχάνεται βελτιστοποιώντας μία αντικειμενική συνάρτηση η οποία περιγράφει ακριβέστερα τους στόχους του προβλήματος. Με τη βελτιστοποίηση της αντικειμενικής συνάρτησης επιτυγχάνεται είτε η μεγιστοποίηση του οφέλους είτε η ελαχιστοποίηση της απώλειας από τη χρησιμοποίηση των εν λόγο κέντρων παροχής υπηρεσιών. Το περιβάλλον αποτελεί το χωρικό σύστημα ζήτησης (χώρος) στο οποίο τα κέντρα παροχής υπηρεσιών και οι πελάτες είναι τοποθετημένοι. Το περιβάλλον αυτό μπορεί να είναι Συνεχές ή Διακριτό ή ένα δίκτυο (πλήρες ή διακριτό). Το περιβάλλον κάθε προβλήματος στην Ανάλυση Χωροθέτησης αποτελεί κριτήριο για την κατηγοριοποίηση του κάθε προβλήματος. Όλες οι πιθανές θέσεις / τοποθεσίες στις οποίες μπορούν να τοποθετηθούν τα διάφορα κέντρα παροχής υπηρεσιών αποτελούν υποσύνολα του περιβάλλοντος. Τα κέντρα παροχής υπηρεσιών μπορούν να απεικονιστούν με σημεία, γραμμές, μονοπάτια ή κύκλους. Τα κέντρα παροχής υπηρεσιών περιορίζονται ανάλογα με τις ικανότητες της κάθε υπηρεσίας, το κόστος της εγκατάστασης της και την ακτίνα κάλυψης (coverage radius) της. Στο κάθε περιβάλλον μπορούν να τοποθετηθούν το πολύ P-κέντρα παροχής υπηρεσιών, όπου P είναι ένας πεπερασμένος αριθμός. Οι θέσεις των πελατών κάθε επιχείρησης αποτελούν υποσύνολο του κάθε περιβάλλοντος. Οι πελάτες έχουν συγκεκριμένες απαιτήσεις (σημεία ζήτησης) οι οποίες πρέπει να καλυφθούν.Η λύση των διαφόρων προβλημάτων χωροθέτησης παρέχεται με τη βελτιστοποίηση της αντικειμενικής συνάρτησης.

Στην παρούσα εργασία παρουσιάζονται δύο μεθευρετικοί αλγόριθμοι οι οποίοι επιλύουν δύο σημαντικά προβλήματα με πολλές εφαρμογές. Τα προβλήματα που επιλύονται είναι η εύρεση των p-μέσων (p-median) και η χωροθέτηση εγκαταστάσεων πεπερασμένης χωρητικότητας (capacitated facility location problem). Βάση των προτεινόμενων

υλοποιήσεων αποτελεί ο αλγόριθμος βελτιστοποίησης αποικίας μυρμηγκιών (ant colony optimization), ο οποίος στηρίζεται στη συμπεριφορά των μυρμηγκιών.

Ειδικότερα, σε ένα σύστημα διανομής, το πρόβλημα εύρεσης p-μέσων, στοχεύει στη χωροθέτηση p εγκαταστάσεων (medians). Το εν λόγω πρόβλημα μπορεί μαθηματικά να διατυπωθεί ως πρόβλημα ελαχιστοποίησης των αθροισμάτων των συντομότερων αποστάσεων σε ένα γράφο, ενώ μπορεί ακόμη να δομηθεί και ως πρόβλημα ακεραίου προγραμματισμού (integer programming problem).
Το πρόβλημα χωροθέτησης εγκαταστάσεων πεπερασμένης χωρητικότητας, συνίσταται στον προσδιορισμό των εγκαταστάσεων που πρέπει να χρησιμοποιηθούν, από ένα σύνολο εγκαταστάσεων, και την ανάθεση πελατών στις εγκαταστάσεις αυτές, ώστε να καλύπτεται πλήρως η ζήτηση των πελατών. Στόχος είναι να ελαχιστοποιηθεί το συνολικό κόστος διανομής, κάτω από τους περιορισμούς της κάλυψης της ζήτησης για το σύνολο των πελατών, χωρίς να παραβιάζεται η μέγιστη χωρητικότητα των εγκαταστάσεων.
Ο αλγόριθμος Βελτιστοποίησης Αποικίας Μυρμηγκιών, οφείλει την ύπαρξή του στη φύση και ειδικότερα στον τρόπο εντοπισμού τροφής από τα μυρμήγκια. Τα μυρμήγκια ακολουθούν τυχαίες διαδρομές και όταν εντοπίσουν τροφή επιστρέφουν στην αποικία τους, μεταφέροντάς την, και αφήνοντας κατά τη διαδρομή αυτή ίχνη φερομόνης (χημική ουσία). Εάν άλλα μυρμήγκια ανακαλύψουν κάποιο τέτοιο μονοπάτι, τότε τείνουν να το ακολουθούν, επιβάλλοντάς το. Με το πέρασμα του χρόνου επικρατούν τα συντομότερα μονοπάτια, αφού εκεί η φερομόνη που εξατμίζεται, αντικαθίσταται συντομότερα από νέες ποσότητες που αφήνουν τα διερχόμενα μυρμήγκια. Ο αλγόριθμος ACO, προσομοιώνει αυτή τη συμπεριφορά με «τεχνητά μυρμήγκια» που κινούνται μέσα σε ένα γράφο που αναπαριστά το προς επίλυση πρόβλημα. Συνήθως η μέθοδος περιλαμβάνει πολλές παραμέτρους, των οποίων ο υπολογισμός και η ενημέρωση, επηρεάζει σημαντικά την ποιότητα της προκύπτουσας λύσης.

Στην παρούσα εργασία παρουσιάζουμε μία μέθοδο εφαρμογής του αλγόριθμου ACO για την επίλυση του προβλήματος p-median όπως αυτό έχει διατυπωθεί στο (Branco, I. και Coelho, J., 1990) και με βάση τον αλγόριθμο που παρουσιάζεται στο (Zohrehbandian, M., και Hamidnia Namini, S., 2010). Ο αλγόριθμος αυτός εφαρμόζει μια σειρά

τροποποιήσεων στο κλασσικό πρόβλημα δρομολόγησης οχημάτων, ώστε να κάνει εφικτή την αναζήτηση της βέλτιστης λύσης του προβλήματος p-median.

Επίσης, παρουσιάζουμε ένα αλγόριθμο ACO, ο οποίος υλοποιεί την παραλλαγή του βασικού CFLP κατά την οποία κάθε πελάτης μπορεί να εξυπηρετηθεί από μία μόνο εγκατάσταση (Nozick και Turnquist 2001). Πρόκειται για το πρόβλημα χωροθέτησης εγκαταστάσεων πεπερασμένης χωρητικότητας από μία πηγή (single source capacitated facility location problem), όπως αυτό περιγράφεται στο (Kumweang και Kawtummachai, 2005).

Τα αποτελέσματα των προτεινόμενων αλγορίθμων που παρουσιάζονται, είναι αξιοσημείωτα και συγκρίσιμα με τα βέλτιστα μιας σειράς ενδεικτικών προβλημάτων (test problems) που χρησιμοποιήθηκαν ως προβλήματα σύγκρισης (benchmarks) για να καταδείξουν την αποτελεσματικότητα των προτεινόμενων αλγορίθμων.

2. Το Πρόβλημα Εύρεσης P-Μέσων (P-Median Problem)

Το πρόβλημα εύρεσης p-μέσων (p-median) σε αδρές γραμμές αποτελεί ένα πρόβλημα χωροθέτησης p εγκαταστάσεων (medians) σε ένα δίκτυο. Σε μια περισσότερο μαθηματική διατύπωση του προβλήματος, θα μπορούσαμε να πούμε πως το πρόβλημα εύρεσης p-μέσων μπορεί να διατυπωθεί ως εξής: δεδομένου ενός γράφου ή ενός δικτύου G = (V, E), με V κορυφές και E ακμές, ζητείται η εύρεση ενός $V_p \subseteq V$ τέτοιου ώστε $|V_p| = p$, όπου p μπορεί να είναι είτε μεταβλητή είτε σταθερά, και ταυτόχρονα ελαχιστοποιείται το άθροισμα των συντομότερων αποστάσεων από τις κορυφές του $\{V \setminus V_p\}$ στην κοντινότερη κορυφή του V_p. Τις τελευταίες δεκαετίες, μεγάλος αριθμός συγγραμμάτων εξετάζει μεθόδους και τρόπους υλοποίησης του εν λόγω αλγορίθμου σε διάφορες επιστημονικές περιοχές. Ενδεικτικά αναφέρουμε τα (Beasley, J. 1985, Branco I. κ.α. 1990, Fathali, J. κ.α. 2006, Pardalos, P. κ.α. 2000, Reese, J. 2005, Talbi G. 2009, Yaghini, M. κ.α. 2010).

Ειδικότερα, στο πρόβλημα P-median δίνεται σαν είσοδος ένα σταθμικό σταθερό κλειστό γράφημα / δίκτυο όπου κάθε κορυφή συμβολίζει ένα πελάτη (σημείο ζήτησης) ο οποίος έχει κάποια απαίτηση (σημείο προσφοράς). Η απαίτηση του πελάτη (ζήτηση) είναι συνήθως μια απόσταση η οποία βρίσκεται αθροίζοντας τα βάρη που βρίσκονται στις ακμές που διανύονται στο γράφημα σε κάθε μονοπάτι.

Μπορούμε να τοποθετήσουμε ένα ή P-κέντρα παροχής υπηρεσιών σε οποιεσδήποτε P-κορυφές του γραφήματος μας οι οποίες θα μπορούν να εξυπηρετήσουν όλους τους πελάτες. Το ζητούμενο είναι σε ποια/ες κορυφές θα πρέπει να τοποθετήσουμε το ένα ή τα P-κέντρα παροχής υπηρεσιών έτσι ώστε να ελαχιστοποιήσουμε το συνολικό κόστος των διαδρομών των πελατών προς αυτό/ά τα κέντρα παροχής υπηρεσιών.

Το πρόβλημα μπορεί να διατυπωθεί ως εξής: Να χωροθετηθεί ένας δοσμένος αριθμός κέντρων παροχής υπηρεσιών (έστω P) κατ' «άριστο τρόπο», ώστε το άθροισμα των συντομότερων αποστάσεων των κορυφών του γραφήματος από τα πλησιέστερα προς

αυτά κέντρα παροχής υπηρεσιών να είναι ελάχιστο. Το πρόβλημα αυτό είναι γνωστό ως πρόβλημα P-Διάμεσος ενός γραφήματος.

2.1. Μαθηματική Διατύπωση του Προβλήματος

Στην Ενότητα αυτή παρουσιάζουμε αναλυτικά η βασική διατύπωση του προβλήματος p-μέσων. Έστω G = (V, E) ένας πλήρης, ζυγισμένος μη κατευθυνόμενος γράφος, όπου V είναι ένα σύνολο κορυφών και E ένα σύνολο ακμών. Με κάθε ακμή, συσχετίζεται μια απόσταση $d(v_i, v_j)$, η οποία είναι η μικρότερη απόσταση ανάμεσα στις κορυφές v_i και v_j. Ο n × n συμμετρικός πίνακας με στοιχεία $d_{ij} = d(v_i, v_j)$ είναι ο πίνακας κοντινότερων αποστάσεων. Σε κάθε κορυφή v_i αντιστοιχίζεται ένα βάρος w_i και ο πίνακας ζυγισμένης απόστασης είναι ο $W_{ij} = w_i d_{ij}$. Αυτός ο πίνακας δεν είναι στη γενική περίπτωση συμμετρικός, εκτός εάν $w_i = w_j$ για κάθε i και j. Το p-median πρόβλημα ορίζεται συχνά πάνω σε ένα δίκτυο και στην περίπτωση αυτή ένας γράφος μπορεί να δημιουργηθεί συνδέοντας τους κόμβους με ακμές των οποίων η απόσταση είναι η κοντινότερη απόσταση ανάμεσα στους κόμβους του δικτύου.

Ο Hakimi (Hakimi, L., 1964) όρισε ένα σημείο m ως απόλυτο μέσο (absolute median) του G αν για κάθε σημείο v_j του G ισχύει η σχέση:

$$\sum_{i=1}^{n} w_i d(v_i, m) \le \sum_{i=1}^{n} w_i d(v_i, v_j) \quad . \qquad (1)$$

Αργότερα (Hakimi, L., 1965) γενίκευσε αυτή την ιδέα στον δημοφιλή σήμερα ορισμό του p-median: Έστω V_p ένα σύνολο p σημείων πάνω στο G και $d(v_i, V_p)$ και $d(v_i, V_p^*)$ οι συντομότερες αποστάσεις από την κορυφή v_i στο κοντινότερο στοιχείο του V_p και V_p^*,

αντίστοιχα. Τότε σύμφωνα με τον Hakimi ένα σύνολο σημείων V_p^* είναι ένας p-median του G αν, για κάθε V_p στο G, ισχύει

$$\sum_{i=1}^{n} w_i d\left(v_i, V_p^*\right) \leq \sum_{i=1}^{n} w_i d\left(v_i, V_p\right) . \qquad (2)$$

Κάτω από αυτή τη γενίκευση, ο απόλυτος μέσος είναι ο 1-median του γράφου (p = 1). Ο όρος p-median αναφέρεται σε ένα σύνολο κορυφών V_p. Οι κορυφές του V_p καλούνται p-median κορυφές.

Ένας p-median φυσιολογικά τμηματοποιεί ένα γράφο γιατί για κάθε p-median κορυφή m_j, υπάρχει ένα σύνολο κορυφών που είναι κοντινότερα στην κορυφή αυτή από κάθε άλλη p-median κορυφή. Ο κοντινότερος γείτονας που τμηματοποιεί τα κελιά P_j, για $1 \leq j \leq p$, είναι:

$$P_j = \left\{v_i : d\left(v_i, m_j\right) \leq d\left(v_i, m_k\right)\right\}, \qquad i = 1,2,...,1 \leq k \leq p . \qquad (3)$$

Αν $d\left(v_i, m_j\right) = d\left(v_i, m_k\right)$, η κορυφή v_i συνήθως αποδίδεται στην p-median κορυφή με το μικρότερο δείκτη. Η συνολική ζυγισμένη απόσταση D είναι:

$$D = \sum_{i=1}^{n} \sum_{v_j \in P_j} w_i d\left(v_i, v_j\right) \qquad (4)$$

Αυτή η τυποποίηση του προβλήματος P-median επιτρέπει στα κέντρα παροχής υπηρεσιών να είναι τοποθετημένα σε ένα πεπερασμένο σύνολο των ενδεχόμενων θέσεων (potential sites). Αυτές οι θέσεις αναπαριστάνουν τους κόμβους του δικτύου. Για οποιονδήποτε αριθμό των κέντρων παροχής υπηρεσιών P υπάρχει το λιγότερο μια κατάλληλη λύση του προβλήματος P-Διάμεσος η οποία τοποθετείται μόνο στους

κόμβους του δικτύου. Έτσι, η απλοποιημένη τυποποίηση περιλαμβάνει μόνο τους κόμβους σαν ενδεχόμενη τοποθεσία κάποιου κέντρου παροχής υπηρεσιών και δεν αλλοιώνει την τιμή της αντικειμενικής συνάρτησης.

Όταν το πρόβλημα P-Διάμεσος εφαρμόζεται σε ένα γενικό δίκτυο, δύσκολα μπορεί να επιλυθεί με την ευνοϊκότερη συνθήκη διότι το πρόβλημα P-Διάμεσος ανήκει στην κλάση των NP-πλήρη προβλημάτων. Όμως, περιορίζοντας τις ενδεχόμενες θέσεις ενός κέντρου παροχής υπηρεσιών στους κόμβους του δικτύου τότε μειώνεται ο αριθμός των πιθανών θέσεων. Ο αριθμός αυτός διαμορφώνεται ως εξής:

$$\binom{N}{P} = \frac{N!}{P!(N-P)!} \qquad (5)$$

όπου N : ο αριθμός των κόμβων στο δίκτυο

P : ο αριθμός των κέντρων παροχής υπηρεσιών

Για μια σταθερή τιμή του P, το πρόβλημα P-Διάμεσος λύνεται σε πολυωνυμικό χρόνο. Παρόλα αυτά είναι πιθανό μια αριθμητική προσέγγιση να είναι υπολογιστικά απαγορευτική για μετρήσιμες τιμές του N (από εκατοντάδες σε χιλιάδες κόμβους) και επίσης για μετρήσιμες τιμές του P (δεκάδες τοποθετήσεις). Μερικά θέματα πολυπλοκότητας επιτρέπουν την ανάπτυξη των αλγορίθμων για την λύση αυτού του προβλήματος.

Η παραπάνω τυποποίηση προτείνει τη χρησιμοποίηση των τεχνικών του Ακέραιου Προγραμματισμού για τη λύση του προβλήματος P-Διάμεσος. Ενώ αυτές οι τεχνικές είναι συχνά ικανές να επεκτείνουν τις κατάλληλες ακέραιες λύσεις για συγκεκριμένου μεγέθους προβλήματα σε μετρήσιμο χρόνο, διάφορα αποτελεσματικά ευρετικά έχουν αναπτυχθεί για την λύση των ενδιάμεσων προβλημάτων

Το πρόβλημα αυτό έχει διάφορες εναλλαγές που προκύπτουν από πρόσθετους περιορισμούς οι οποίοι μπορεί να ισχύσουν. Αναφέρουμε μερικούς από αυτούς :

Περιορισμός χωρητικότητας:

Η δημιουργία ενός κέντρου παροχής υπηρεσιών (π.χ. σχολείο) που εξυπηρετεί ένα μικρό αριθμό χρηστών, έχει συνήθως ως αποτέλεσμα τη μη αποτελεσματική χρήση του κέντρου και την ελάχιστη απόδοση της υπηρεσίας των υπαλλήλων που ασχολούνται σε αυτό, αφού η ζήτηση είναι ελάχιστη. Αυτό βέβαια σημαίνει ότι σε πολλές περιπτώσεις περιορισμοί, σχετικοί με την ελάχιστη ζήτηση που ένα κέντρο πρέπει να εξυπηρετεί είναι αναγκαίοι.

Από την άλλη, η έλλειψη μεγάλων εκτάσεων για την εγκατάσταση μεγάλων κέντρων παροχής υπηρεσιών ή ακόμη και η επιθυμία για την δημιουργία μικρού ή μεσαίου μεγέθους κέντρων, δημιουργεί την ανάγκη για περιορισμούς που να αναφέρονται στην μέγιστη χωρητικότητα. Τέτοιοι περιορισμοί μπορούν να ενσωματωθούν στο μοντέλο P-Διάμεσος με την εξής μορφή:

$$u_j y_j \leq \sum_{i=1}^{n} x_{ij} w_i \leq v_j y_j \,,\; j = 1 \ldots n \qquad (6)$$

Περιορισμοί προϋπολογισμού:

Ο τρόπος προσδιορισμού του προβλήματος P-Διάμεσος (σχέσεις 1-4,6) έμμεσα προϋποθέτει ότι το κόστος ενός κέντρου παροχής υπηρεσιών σχετίζεται πολύ λίγο ή καθόλου με τη θέση που είναι χωροθετημένο, με αποτέλεσμα ο περιορισμός (1) να είναι στην ουσία και περιορισμός προϋπολογισμού. Όταν όμως το κόστος ενός κέντρου διαφέρει από θέση σε θέση, τότε ο προϋπολογισμός δημιουργίας του συστήματος των P-κέντρων θα διαφέρει ανάλογα με το χωρικό πρότυπο αυτού του συστήματος. Σε μια τέτοια περίπτωση είναι πιο ρεαλιστικό να θεωρούμε ότι ο προϋπολογισμός είναι γνωστός και να ερευνούμε για τον αριθμό των κέντρων που μπορούν να δημιουργηθούν με αυτόν τον προϋπολογισμό. Ένας τέτοιος τρόπος διατύπωσης του προβλήματος P-Διάμεσος επιτυγχάνεται αν αντικαταστήσουμε τον περιορισμό (6) με έναν από τους παρακάτω περιορισμούς:

$$\sum_{i=1}^{m}\sum_{j=1}^{n} w_i x_{ij} y_j + \sum_{j=1}^{n} f_j y_j \leq B$$

$$\sum_{j=1}^{n} f_j y_j \leq B \qquad (7)$$

Περιορισμοί μέγιστου κόστους:

Η εφαρμογή του μοντέλου p-διάμεσος, όπως διατυπώθηκε στη γενική μορφή του, μπορεί να δημιουργεί μια χωρική διάταξη κέντρων. Στη διάταξη αυτή για μερικούς χρήστες, η προσιτότητα μπορεί να μην είναι ικανοποιητικού επιπέδου, καθώς στο μοντέλο p-διάμεσος δίνεται έμφαση στην ελαχιστοποίηση του συνολικού χρόνου ταξιδιού προς το κέντρο παροχής υπηρεσιών ή στο συνολικό χρόνο ανταπόκρισης στη ζήτηση από κάποιο κινητό κέντρο παροχής υπηρεσιών.

Επομένως, καμία μέριμνα δεν υπάρχει για τους ακραίους χρόνους ταξιδιού ή ανταπόκρισης. Επίσης, είναι δυνατόν ορισμένες λύσεις να αφήνουν μερικά σημεία ζήτησης να έχουν υπερβολικούς χρόνους από τα πλησιέστερα κέντρα. Η λύση του προβλήματος μπορεί να επιτευχθεί με την εισαγωγή του εξής περιορισμού :

$$\sum_{j \in N_j}^{n} x_{ij} = 1, i = 1, \ldots, m \qquad (8)$$

έτσι ώστε για το σύνολο των πιθανών λύσεων το κόστος προσιτότητας να είναι ίσο ή μικρότερο ενός δοσμένου κρίσιμου χρόνου ή απόστασης T που δίνεται από τη σχέση:

$$N_j = [\, j \,/\, d_{ij} \leq T \,] \qquad (9)$$

2.2. Διατύπωση Ακέραιου Προγραμματισμού (Integer Programming Formulation)

Το πρόβλημα p-median τυπικά μπορεί να δομηθεί ως ένα Πρόγραμμα Ακεραίου (Integer Program - IP) (Talbi G., 2009.). Έστω ξ_{ij} μια μεταβλητή απόδοσης τέτοια ώστε:

$$\xi_{ij} = \begin{cases} 1 & \textit{αν η κορυφή } x_j \textit{ έχει αποδοθεί στην κορυφή } x_i \\ 0 & \text{διαφορετικά} \end{cases}$$

Τότε, το πρόβλημα IP διατυπώνεται ως εξής:

$$\min Z = \sum_{ij} W_{ij} \xi_{ij} \qquad (10)$$

υπό την προϋπόθεση ότι ισχύουν οι παρακάτω συνθήκες:

$$\sum_i \xi_{ij} = 1, \qquad (11)$$

$$\sum_i \xi_{ii} = p, \qquad (12)$$

$$\xi_{ij} \le \xi_{ii}, \forall \text{ j = 1, . . . , n,} \qquad (13)$$

$$\xi_{ij} \in \{0,1\} \qquad (14)$$

Η πρώτη από τις παραπάνω συνθήκες διασφαλίζει ότι κάθε κορυφή αποδίδεται σε ένα και μόνο ένα στοιχείο του υποσυνόλου p-στοιχείων. Η δεύτερη συνθήκη(11) εγγυάται ότι υπάρχουν p κορυφές που αποδίδονται στον εαυτό τους, πράγμα που επιβάλει στον πληθικό αριθμό (cardinality) του υποσυνόλου p-median να είναι ίσος με p. Η τρίτη συνθήκη(12) δηλώνει ότι οι κορυφές δε μπορούν να αποδοθούν σε μη p-median

κορυφές. Ο p-median δίνεται από τη σχέση $\{v_i | \xi_{ii} = 1\}$. Η πιο συνηθισμένη παραλλαγή ελαστικοποίησης των παραπάνω συνθηκών είναι να αντικαταστήσουμε την τέταρτη συνθήκη(14) με $\xi_{ij} \geq 0$ (15)

2.3. Τρόποι επίλυσης των προβλημάτων p-median

Για την επίλυση των προβλημάτων των P-Διαμέσων έχουν προταθεί πολλοί τρόποι, άλλοι ακριβείς και άλλοι προσεγγιστικοί. Οι Handler S.L. and Mirchandani P.B. ταξινομούν τους τρόπους επίλυσης σε πέντε κατηγορίες:

Απαρίθμηση: Σ' αυτήν την διαδικασία απαριθμούνται όλες οι δυνατές λύσεις για να βρεθεί η βέλτιστη. Αν και με αυτόν τον τρόπο είναι δυνατόν να λυθούν προβλήματα ενός μόνο κέντρου παροχής υπηρεσιών ή προβλήματα σε μικρά δίκτυα, παρέχει όμως τη δυνατότητα για καλύτερη κατανόηση των προβλημάτων των Διαμέσων. Ένας τέτοιος τρόπος επίλυσης έχει προταθεί από τον Hakimi s. L.

Με τη Θεωρία των Γράφων: Αυτές οι μέθοδοι εκμεταλλεύονται τη δομή του δικτύου για να προσδιορίσουν τις θέσεις των P-Διαμέσων και είναι αρκετά αποτελεσματικές όταν το δίκτυο έχει τη μορφή δέντρου. Τέτοιες μεθόδους έχουν προτείνει ο Goldman A.J. και οι Kariv O. And Hakimi S.L.

Ευριστική Μέθοδος: Αυτή η μέθοδος βασίζεται στις εμπνεύσεις του κάθε ερευνητή και δεν μπορεί αν εγγυηθεί την εύρεση της βέλτιστης λύσης. Είναι όμως πολύ χρήσιμη όταν ζητείται μια απλώς «καλή» λύση και δεν απαιτείται η εύρεση της βέλτιστης λύσης. Διάφορες Ευριστικές μεθόδους επίλυσης έχουν προταθεί από τον Khumawala B.M. και τον Shapiro J.F.

Μαθηματικού Προγραμματισμού: Οι προσεγγίσεις αυτές βασίζονται στην κατάστρωση των προβλημάτων ως προβλήματα Ακέραιου Προγραμματισμού, και υπάρχουν πολλά υπολογιστικά προγράμματα που επιλύουν τέτοιου είδους προβλήματα. Ενδεικτικά αναφέρονται οι Khumawala B.M. και Garfinkel R.S. and Nemhauser G.L.

Δυϊκού Μαθηματικού Προγραμματισμού: Τα προβλήματα των διαμέσων καταστρώνονται ως προβλήματα Ακέραιου Προγραμματισμού, αλλά χρησιμοποιούνται

δυϊκές τεχνικές επίλυσης όπως για παράδειγμα η χαλάρωση κατά Lagrange. Αυτή η μέθοδος έχει αποδεχθεί πολύ επιτυχής και γι' αυτό πολλοί ερευνητές έχουν προτείνει διάφορες προσεγγίσεις όπως οι Geoffrion A.M. , Held M. , Wolfe P. and Crower H.P.

3. Το Πρόβλημα Χωροθέτησης Εγκαταστάσεων Πεπερασμένης Χωρητικότητας (Capacitated Facility Location Problem)

3.1. Το πρόβλημα UFLP

Το πρόβλημα χωροθέτησης εγκαταστάσεων χωρίς περιορισμούς (Uncapacitated Facility Location Problem – UFLP) αποτελεί επέκταση του p-median με την διαφορά ότι η αντικειμενική συνάρτηση μπορεί να επεκταθεί με έναν όρο για σταθερό κόστος τοποθεσία εγκατάστασης και ως εκ τούτου, ο αριθμός των εγκαταστάσεων που πρέπει να εγκατασταθεί μετατρέπεται σε ενδογενή απόφαση του μοντέλου, με σκοπό την ελαχιστοποίηση της συνολικής απόστασης ή του συνολικού κόστους που συνδέεται με την ικανοποίησης της ζήτησης των πελατών, όπως αναφέρει ο Lawrence V.Snyder.

Μορφοποίηση UFLP

Παράμετροι

f_j = Κόστος εγκατάστασης μονάδας

Αντικειμενική συνάρτηση

$$\text{Minimize} \sum_{j \in J} f_j X_j + \sum_{i \in I} \sum_{j \in J} h_i d_{ij} Y_{ij} \qquad (16)$$

Περιορισμοί

$$\sum_{j \in J} Y_{ij} = 1 \qquad \forall i \in I \qquad (17)$$

$$Y_{ij} \le X_j \qquad \forall i \in I, \forall j \in J \qquad (18)$$

$$X_j \in \{0,1\} \qquad \forall j \in J \qquad (19)$$

$$Y_{ij} \ge 0 \qquad \forall i \in I, \forall j \in J \qquad (20)$$

Σχολιασμός εξισώσεων

Αντικαθιστούμε τη (4) με την (16) εισάγοντας τον όρο

$$\sum_{j \in J} f_j X_j \qquad (21)$$

Που προστίθεται για να ελαχιστοποιηθεί το συνολικό κόστος εγκατάστασης των μονάδων. Αφαιρούμε τον περιορισμό (7) και οι υπόλοιποι περιορισμοί παραμένουν.

(17) Διασφαλίζει πως κάθε πελάτης εξυπηρετείται από μια εγκατάσταση

(18) Διασφαλίζει ότι η εγκατάσταση από την οποία εξυπηρετείται ο πελάτης είναι ανοιχτή

(19)-(20) Αντιπροσωπεύουν τις μη αρνητικές υποθέσεις καθώς και τις υποθέσεις πληρότητας.

3.2. Μαθηματική Διατύπωση του Προβλήματος

Σε ένα σύστημα διανομής, το Πρόβλημα Χωροθέτησης Εγκαταστάσεων Πεπερασμένης Χωρητικότητας (Capacitated Facility Location Problem – CFL Problem), συνίσταται στο να αποφασίσουμε ποιες εγκαταστάσεις πρέπει να χρησιμοποιήσουμε, από ένα σύνολο εγκαταστάσεων, ώστε να εξυπηρετηθεί ένα σύνολο πελατών και ποιοι πελάτες θα εξυπηρετηθούν από ποιες εγκαταστάσεις (Beasley, J. 1988, Akino, U. κ.α. 1977, Clarke, G κ.α. 1964). Ο στόχος είναι να ελαχιστοποιήσουμε το συνολικό κόστος διανομής, κάτω από τους περιορισμούς της κάλυψης της ζήτησης για το σύνολο των πελατών, χωρίς να παραβιάζεται η μέγιστη χωρητικότητα των εγκαταστάσεων. Το συνολικό κόστος αποτελείται από το κόστος παράδοσης των εμπορευμάτων στους πελάτες, τα κόστη παραγωγής στις εγκαταστάσεις (λειτουργικό κόστος) και το προκαθορισμένο κόστος δημιουργίας και συντήρησης των εγκαταστάσεων (σταθερό κόστος). Ένας μεγάλος αριθμός ερευνητών έχει προτείνει μεθόδους μοντελοποίησης και αλγόριθμους επίλυσης του εν λόγω προβλήματος. Ενδεικτικά αναφέρουμε τα (Mulvey, J. κ.α. 1984, Koskosidis, I. κ.α 1992, Nauss, M. 1978, Osman, I. κ.α. 1994, Sherali, H. κ.α. 1992, Singhtaun, C.

κ.α. 2007, Yaghini, M. κ.α. 2010), ενώ μια πληρέστερη καταγραφή των υφιστάμενων μεθόδων και αλγορίθμων υπάρχει στο (Talbi G., 2009).

Μαθηματικά, το εν λόγω πρόβλημα μπορεί να μοντελοποιηθεί ως εξής:

$$\min \sum_{k \in K} \sum_{j \in J} \left(c_{kj} + v_j d_k \right) z_{kj} + \sum_{j \in J} f_j y_j \qquad (21)$$

υπό τις ακόλουθες συνθήκες:

$$\sum_j z_{kj} = 1 \quad \text{για κάθε k} \qquad (22)$$

$$\sum_i d_k z_{kj} \le s_j y_j \quad \text{για κάθε j} \qquad (23)$$

$$0 \le z_{kj} \le y_j \quad \text{για κάθε k, j} \qquad (24)$$

$$y_j \in \{0,1\} \quad \text{για κάθε j} \qquad (25)$$

όπου,

K το σύνολο των πελατών

J το σύνολο των δυνητικών εγκαταστάσεων

c_{kj} το κόστος παράδοσης d_k μονάδων από την εγκατάσταση j στον πελάτη k

v_j το λειτουργικό κόστος της εγκατάστασης j ανά μονάδα

f_j το σταθερό κόστος ανά εγκατάσταση j

d_k η ζήτηση του πελάτη k

s_j η χωρητικότητα της εγκατάστασης j

y_j δείκτης ίσος με 1 αν η εγκατάσταση j είναι σε λειτουργία

z_{kj} δείκτης μονάδων εμπορευμάτων που η εγκατάσταση j παραδίδει στον πελάτη k

Το ανωτέρω πρόβλημα μπορεί να διατυπωθεί και με κάποιες παραλλαγές, όπως για παράδειγμα εκείνη των (Nozick, L. και Turnquist, M., 2001) που επιτρέπει κάθε πελάτης να μπορεί να τροφοδοτηθεί μόνο από μία εγκατάσταση. Αυτή η παραλλαγή αποτελεί και τη βάση του αλγόριθμου που θα υλοποιήσουμε στην παρούσα εργασία.

4. Αλγόριθμος Βελτιστοποίησης Αποικίας Μυρμηγκιών (Ant Colony Optimization - ACO)

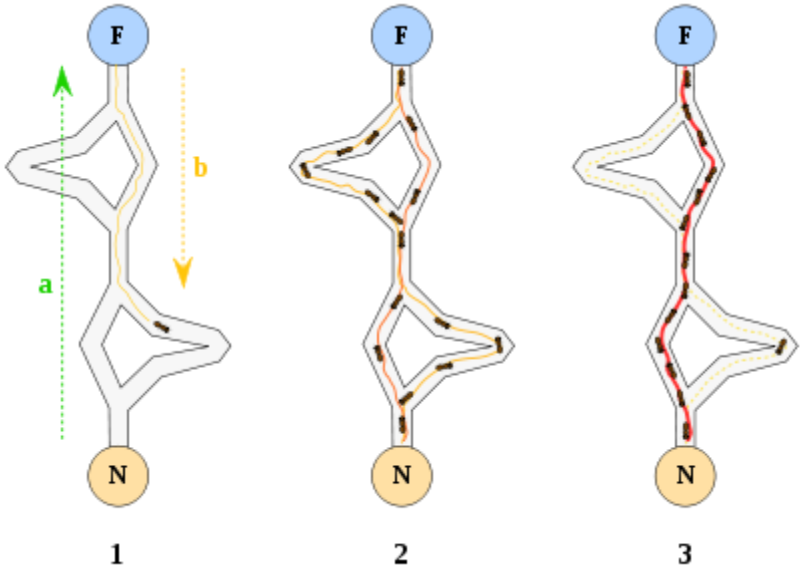
4.1. Εισαγωγή και Βασικά Χαρακτηριστικά Αλγόριθμου ACO

Ο αλγόριθμος Βελτιστοποίησης Αποικίας Μυρμηγκιών (Ant Colony Optimization - ACO) αρχικά προτάθηκε από τους (Colorni κ.α. 1992α, Colorni κ.α. 1992β,). Η έμπνευση της μεθόδου προέρχεται από τη φύση. Η βασική ιδέα είναι να χρησιμοποιήσουμε τη στατιστική πληροφορία που προέρχεται από προηγούμενες επαναλήψεις μιας διαδικασίας με σκοπό την αναζήτηση καλύτερων λύσεων. Συνήθως η μέθοδος περιλαμβάνει πολλές παραμέτρους, των οποίων ο υπολογισμός και η ενημέρωση, επηρεάζει σημαντικά την ποιότητα της προκύπτουσας λύσης. Πληθώρα ερευνητικών συγγραμμάτων εξετάζει την εφαρμογή του αλγορίθμου σε διάφορους τομείς. Ενδεικτικά αναφέρουμε τα (Bell J. κ.α. 2004, Dorigo, M. κ.α. 2004, Kumweang, K. κ.α. 2005, Pardalos, P. κ.α. 2000, Talbi G. 2009, Vlahos, A. 2006, Zohrehbandian, M. κ.α. 2010).

Πιο συγκεκριμένα, ο εν λόγω αλγόριθμος βασίζεται στη συμπεριφορά των μυρμηγκιών όταν αυτά αναζητούν τροφή. Γενικά, τα μυρμήγκια ακολουθούν τυχαίες διαδρομές και όταν εντοπίσουν τροφή επιστρέφουν στην αποικία τους, μεταφέροντάς την, και αφήνοντας κατά τη διαδρομή αυτή ίχνη φερομόνης (μια χημική ουσία). Αν άλλα μυρμήγκια ανακαλύψουν κάποιο τέτοιο μονοπάτι, τότε τείνουν να το ακολουθούν, επιβάλλοντάς το αν τελικά οδηγεί σε τροφή. Με το πέρασμα του χρόνου η φερομόνη εξατμίζεται. Αυτό έχει ως συνέπεια, να επικρατούν τελικά τα συντομότερα μονοπάτια, αφού εκεί η φερομόνη που εξατμίζεται, αντικαθίσταται συντομότερα από νέες ποσότητες που αφήνουν τα διερχόμενα μυρμήγκια. Επίσης, η εξάτμιση της φερομόνης παρέχει το επιπλέον πλεονέκτημα της αποφυγής σύγκλισης σε τοπικά βέλτιστες λύσεις. Κάτι τέτοιο θα συνέβαινε σε περίπτωση μη εξάτμισης, αφού τα μονοπάτια που θα επιλέγονταν από τα πρώτα μυρμήγκια θα ήταν ελκυστικά και για τα επόμενα, περιορίζοντας με τον τρόπο αυτό το χώρο τον υποψήφιων λύσεων.

Συνοπτικά, αυτό που συμβαίνει είναι ότι, όταν ένα μυρμήγκι βρει ένα σύντομο μονοπάτι που οδηγεί σε τροφή, τα μυρμήγκια που ακολουθούν είναι πολύ πιθανό να πάρουν το ίδιο μονοπάτι ακολουθώντας τα ίχνη φερομόνης και τελικά όλα τα μυρμήγκια θα ακολουθήσουν το ίδιο μονοπάτι. Με βάση τον τρόπο λειτουργίας των μυρμηγκιών στη φύση, έχουν δημιουργηθεί οι αλγόριθμοι ACO, οι οποίοι προσομοιώνουν αυτή τη συμπεριφορά με «τεχνητά μυρμήγκια» που κινούνται μέσα σε ένα γράφο που αναπαριστά το προς επίλυση πρόβλημα (βλ. Σχήμα 1).

Σχήμα 1: Κίνηση Τεχνητών Μυρμηγκιών σε ένα Γράφο.



Το πρώτο μυρμήγκι ανακαλύπτει την πηγή τροφής (F), ακολουθώντας ένα τυχαίο μονοπάτι (a), και εν συνεχεία επιστρέφει στη φωλιά (N), αφήνοντας πίσω του ίχνη φερομόνης (b).

Τα μυρμήγκια με τυχαίο τρόπο ακολουθούν 4 πιθανά μονοπάτια, αλλά η κίνηση στο κοντινότερο μονοπάτι είναι πυκνότερη λόγω του συντομότερου της διαδρομής.

Τα μυρμήγκια τελικά επιλέγουν το συντομότερο μονοπάτι αφού, σταδιακά, μεγάλα τμήματα των άλλων μονοπατιών χάνουν τα ίχνη φερομόνης λόγω εξάτμισής της.

4.2. Επιμέρους Βήματα Αλγορίθμου ACO

Παρακάτω, παραθέτουμε το βασικό κορμό του ACO αλγορίθμου.

```
procedure ACO_MetaHeuristic
while (not_termination)
generateSolutions()
daemonActions()
pheromoneUpdate()
end while
end procedure
```

Ο αλγόριθμος εκτελεί μια σειρά επαναλήψεων έως ότου προσεγγίσει ικανοποιητικά τη λύση, οπότε και τερματίζει. Σε κάθε επανάληψη εκτελούνται τρία επιμέρους βήματα:

Το βήμα δημιουργίας διαδοχικών διαδρομών-λύσεων (επιλογή διαδρομών), βάσει των εκάστοτε δεδομένων,
Το βήμα κατά το οποίο εκτελούνται ενέργειες που διασφαλίζουν την ομαλή ολοκλήρωση του αλγόριθμου, δηλαδή να μην περιέλθει ο αλγόριθμος σε κατάσταση «εγκλωβισμού»,
Το βήμα ενημέρωσης των διαδρομών-λύσεων που επιλέγονται, με την απαιτούμενη ποσότητα φερομόνης, ανάλογα με το πόσο ικανοποιητική είναι η λύση.

Παρακάτω αναλύουμε κάθε ένα από τα βήματα αυτά.

4.2.1. Δημιουργία Λύσεων (Επιλογή Διαδρομών)

Αυτό που στην πράξη κάνει κάθε μυρμήγκι σε ένα αλγόριθμο ACO, είναι να δομεί επαναληπτικά μια λύση του προβλήματος. Οι ενδιάμεσες λύσεις συχνά αναφέρονται ως «καταστάσεις λύσης». Σε κάθε επανάληψη του αλγόριθμου, κάθε μυρμήγκι κινείται από την κατάσταση x στην κατάσταση y που αντιστοιχεί σε μία πληρέστερη συγκριτικά με την προηγούμενη ενδιάμεση λύση. Έτσι, κάθε μυρμήγκι k υπολογίζει ένα σύνολο Ak(x) εφικτών λύσεων σε κάθε επανάληψη και μεταβαίνει σε μία από αυτές με πιθανότητα p. Για το μυρμήγκι k, η πιθανότητα p_{xy}^{k} να μεταβεί από την κατάσταση x στην κατάσταση y εξαρτάται από στο συνδυασμό δύο τιμών: της ελκυστικότητας ηxy της μετάβασης, όπως αυτή υπολογίζεται από κάποιο heuristic που δείχνει το βαθμό στον οποίο η μετάβαση αυτή είναι «εκ των προτέρων» (a priori) επιθυμητή, και του επιπέδου του ίχνους τxy της μετάβασης, το οποίο δείχνει πόσο αποτελεσματική έχει υπάρξει στο παρελθόν η εν λόγω μετάβαση. Τα επίπεδα του ίχνους αναπαριστούν μία «εκ των υστέρων» (a posteriori) ένδειξη του πόσο επιθυμητή ήταν αυτή η μετάβαση. Τα ίχνη ενημερώνονται συνήθως όταν όλα τα μυρμήγκια έχουν ολοκληρώσει τη λύση τους, αυξάνοντας ή ελαττώνοντας τα επίπεδα του ίχνους που αντιστοιχούν σε μεταβάσεις οι οποίες ήταν τμήμα των «καλών» (good) ή των «κακών» (bad) λύσεων αντίστοιχα (Βλ. Ενότητα 4.2.3).

Γενικά, το kστό μυρμήγκι μεταβαίνει από την κατάσταση x στην κατάσταση y με πιθανότητα: $p_{xy}^{k} = \frac{\left(\tau_{xy}^{\alpha}\right)\left(\eta_{xy}^{\beta}\right)}{\sum\left(\tau_{xy}^{\alpha}\right)\left(\eta_{xy}^{\beta}\right)}$, όπου τxy είναι η ποσότητα της φερομόνης που εναποτίθεται κατά τη μετάβαση από την κατάσταση x στην y, α (≥ 0) είναι μια παράμετρος που ελέγχει το βαθμό επιρροής του τxy, ηxy είναι ο βαθμός στον οποίο είναι επιθυμητή η μετάβαση από την κατάσταση x στην κατάσταση y (a priori γνωστός, με τυπική τιμή ίση με 1/dxy, όπου d είναι η απόσταση μεταξύ x και y) και β ≤ 1 είναι παράμετρος που ελέγχει το βαθμό επιρροής του ηxy.

4.2.2. Βοηθητικές Ενέργειες (Deamon Actions)

Οι βοηθητικές ενέργειες μπορούν να χρησιμοποιηθούν για την υλοποίηση κεντρικών ενεργειών που δεν μπορούν να εκτελούνται από μεμονωμένα μυρμήγκια. Χαρακτηριστικά παραδείγματα βοηθητικών ενεργειών είναι:

η εφαρμογή μιας τοπικής διαδικασίας αναζήτησης στις λύσεις που σταδιακά υπολογίζονται από τα μυρμήγκια,
η συλλογή των «σφαιρικής πληροφορίας» που μπορεί να χρησιμοποιηθεί για να αποφασίσουμε αν είναι χρήσιμο ή όχι να εναποτεθεί επιπλέον φερομόνη για να πολώσουμε τη διαδικασία αναζήτησης. Πρακτικά, αυτό που μπορεί κανείς να κάνει κεντρικά είναι να παρακολουθεί την πορεία που ακολουθεί κάθε μυρμήγκι της αποικίας και εν συνεχεία να επιλέγει να εναποθέσει επιπλέον φερομόνη σε εκείνα τα επιμέρους τμήματα των λύσεων που έχουν χρησιμοποιηθεί από το μυρμήγκι που κατασκεύασε η καλύτερη λύση.

4.2.3. Ανανέωση Φερομόνης

Όταν όλα τα μυρμήγκια έχουν ολοκληρώσει την εύρεση κάποιας λύσης, τα ίχνη ενημερώνονται με νέα τιμή φερομόνης, η οποία είναι ίση με $\tau_{xy}^{k} = (1-\rho)\tau_{xy}^{k} + \Delta\tau_{xy}^{k}$, όπου τ_{xy}^{k} είναι η ποσότητα φερομόνης που εναποτίθεται για μία μετάβαση κατάστασης xy, ρ είναι ο συντελεστής εξάτμισης φερομόνης και $\Delta\tau_{xy}^{k}$ είναι η ποσότητα φερομόνης που εναποτίθεται και η οποία για ένα τυπικό ACO πρόβλημα υπολογίζεται από τη σχέση:

$$\Delta\tau_{xy}^{k} = \begin{cases} Q/L_{\kappa} & \textit{αν το μυρμήγκι k χρησιμοποιεί τη διαδρομή xy στο μονοπάτι του} \\ 0 & \textit{διαφορετικά} \end{cases}$$

όπου Lk είναι το κόστος της kστης διαδρομής του μυρμηγκιού (συνήθως το μήκος της διαδρομής) και Q μια σταθερά.

4.3. Παραλλαγές Αλγόριθμου ACO

Με την πάροδο του χρόνου, λόγω της εντεινόμενης ερευνητικής δραστηριότητας γύρω από τον αλγόριθμο ACO, προέκυψαν πολλές παραλλαγές του. Παρακάτω αναφέρονται ορισμένες από τις πιο διαδεδομένες:

Σύστημα Ανώτερων Μυρμηγκιών (Elitist Ant System - EAS): Βασικό χαρακτηριστικό αυτής της μεθόδου είναι ότι η συνολικά καλύτερη λύση, εναποθέτει φερομόνη σε κάθε επανάληψη μαζί με όλα τα άλλα μυρμήγκια, ενισχύοντας έτσι την τάση σύγκλισης του αλγορίθμου σε συνολικά βέλτιστες λύσεις.

Σύστημα Μυρμηγκιών Μεγίστου – Ελαχίστου (Max-Min Ant System - MMAS): Η μέθοδος αυτή διαφοροποιείται λόγω του γεγονότος ότι σε κάθε επανάληψη, προσθέτει μέγιστά ή ελάχιστα ποσά φερομόνης (tmax, tmin) ανάλογα με την εκάστοτε λύση. Μόνο η συνολικά καλύτερη λύση ή οι καλύτερες λύσεις κάθε επανάληψης εναποθέτουν φερομόνη. Όλες οι ακμές (edges) των διαδρομών αρχικοποιούνται σε tmax και επαναρχικοποιούνται σε tmax όταν τείνουν να γίνουν στάσιμες.

Σύστημα Κατάταξης Μυρμηγκιών (Rank-based Ant System - ASrank): Βασικό χαρακτηριστικό αυτής της παραλλαγής είναι ότι όλες οι λύσεις βαθμολογούνται με βάση την καταλληλότητά τους. Το ποσό της φερομόνης που εναποτίθεται υπολογίζεται στη συνέχεια για κάθε λύση, έτσι ώστε οι καταλληλότερες λύσεις να εναποθέτουν περισσότερη φερομόνη.

Συνεχής Ορθογώνια Αποικία Μυρμηγκιών (Continuous Orthogonal Ant Colony - COAC): Η παραλλαγή αυτή εστιάζει και πάλι στο μηχανισμό εναπόθεσης φερομόνης. Χρησιμοποιώντας τη μέθοδο αυτή, τα μυρμήγκια που βρίσκονται εντός του πεδίου εφικτότητας (feasible domain) μπορούν να εξερευνήσουν τις περιοχές που επιλέγουν, τάχιστα και με αυξημένη ακρίβεια.

4.4. Σύγκλιση Αλγορίθμου ACO

Κάποιες παραλλαγές του αλγόριθμου ACO είναι εφικτό να αποδειχθεί ότι συγκλίνουν, για παράδειγμα ότι έχουν τη δυνατότητα εύρεσης ενός γενικού βέλτιστου εντός πεπερασμένου χρόνου. Η πρώτη απόδειξη σύγκλισης ACO αλγόριθμου δόθηκε το 2000, για ένα βασισμένο σε γράφημα. Ακολούθησαν αποδείξεις για το γενικό Αλγόριθμο Συστήματος Μυρμηγκιών και Σύστημα Μυρμηγκιών Μεγίστου – Ελαχίστου (MMAS) (Dorigo και Stutzle, 2004). Όπως και στις περισσότερες περιπτώσεις μεθευρετικών αλγορίθμων, είναι εξαιρετικά δύσκολο να υπολογίσουμε τη θεωρητική ταχύτητα σύγκλισης, ενώ για πολλές παραλλαγές του αλγορίθμου δεν έχουν ακόμη βρεθεί θεωρητικές αποδείξεις σύγκλισης.

4.5. Εφαρμογές Αλγόριθμου ACO

Οι αλγόριθμοι ACO εφαρμόζονται σε πολλά προβλήματα συνδυαστικής βελτιστοποίησης (combinatorial optimization problems), όπως είναι τα προβλήματα ανάθεσης και τα προβλήματα δρομολόγησης. Ως χαρακτηριστική εφαρμογή μπορούμε να αναφέρουμε αυτή του Προβλήματος Περιπλανώμενου Πωλητή (Travelling Salesman Problem). Το μεγάλο πλεονέκτημα των μεθόδων ACO, έναντι άλλων, όπως οι simulated annealing και οι γεννητικοί αλγόριθμοι (Dorigo και Stutzle, 2004), είναι ότι έχουν τη δυνατότητα άμεσης επίλυσης δυναμικά μεταβαλλόμενων προβλημάτων, αφού μπορούν και προσαρμόζονται στις μεταβολές, δυναμικά, σε πραγματικό χρόνο. Τέτοιου τύπου δυναμικά προβλήματα, είναι για παράδειγμα τα προβλήματα δρομολόγησης σε δίκτυα και τα προβλήματα αστικών μεταφορών.

Παρακάτω αναφέρουμε συνοπτικά τις βασικές κατηγορίες προβλημάτων που μπορούν να επιλύσουν ACO αλγόριθμοι.

Προβλήματα Δρομολόγησης

Πρόβλημα Χρονοπρογραμματισμού Εργασιών

Πρόβλημα Χρονοπρογραμματισμού Ανοιχτών Καταστημάτων

Πρόβλημα Αντιμετάθεσης Χρονοπρογραμματισμού Ροής Εργασιών

Πρόβλημα Συνολικής Απραξίας Απλής Μηχανής

Πρόβλημα Συνολικής Ζυγισμένης Απραξίας Μηχανής

Πρόβλημα Χρονοπρογραμματισμού Έργου με χρήση Περιορισμένων Πόρων

Πρόβλημα Χρονοπρογραμματισμού Ομάδας Εργασιών

Πρόβλημα Συνολικής Απραξίας Απλής Μηχανής με Ακολουθιακά Εξαρτώμενους Χρόνους Εκκίνησης

Πολυεπίπεδο Πρόγραμμα Δρομολόγησης Ροής Καταστημάτων με Ακολουθιακά Εξαρτώμενους Χρόνους Εκκίνησης/ Μεταβολής

Προβλήματα Διαδρομών Οχημάτων

Πρόβλημα Δρομολόγησης Οχημάτων Πεπερασμένης Χωρητικότητας

Πρόβλημα Δρομολόγησης Οχημάτων Πολλαπλών Εγκαταστάσεων

Πρόβλημα Δρομολόγησης Οχημάτων Περιόδου

Πρόβλημα Δρομολόγησης Οχημάτων Τμηματικών Παραδόσεων

Πρόβλημα Στοχαστικής Δρομολόγησης Οχημάτων

Πρόβλημα Δρομολόγησης Οχημάτων με Παραλαβή και Παράδοση

Πρόβλημα Δρομολόγησης Οχημάτων με Χρονικά Παράθυρα

Χρονικά Εξαρτημένο Πρόβλημα Δρομολόγησης Οχημάτων με Παράθυρα

Προβλήματα Ανάθεσης

Πρόβλημα Τετραγωνική Ανάθεσης

Πρόβλημα Γενικευμένης Ανάθεσης

Πρόβλημα Ανάθεσης Συχνοτήτων

Πρόβλημα Ανάθεσης Πλεονάσματος

Προβλήματα Συνόλων

Πρόβλημα Κάλυψης Συνόλου

Πρόβλημα Τμηματοποίησης Συνόλου

Πρόβλημα Τμηματοποίησης Γράφου – Δέντρου με Περιορισμούς Βάρους

Πρόβλημα Ζυγισμένων Κατευθύνσεων l-Πολλαπλότητας

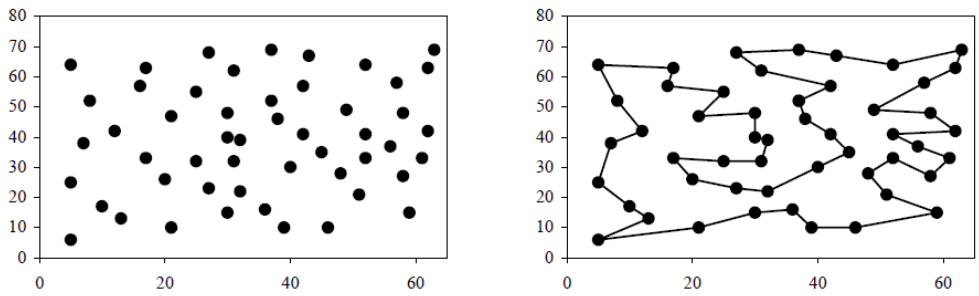
Πρόβλημα Πολλαπλών Σάκων

Πρόβλημα Μέγιστου Ανεξάρτητου Συνόλου

Παράδειγμα: Το πρόβλημα του πλανόδιου πωλητή με ACO

Ο πρώτος αλγόριθμος που αναπτύχθηκε είναι ο AntSystem (AS) του Dorigo [24], [25]. Ο αλγόριθμος αυτός χρησιμοποιήθηκε για πρώτη φορά για την επίλυση του προβλήματος του περιπλανώμενου πωλητή (Traveling Salesman Problem - TSP) καθώς το πρόβλημα αυτό είναι συναφές με τη πραγματική συμπεριφορά των μυρμηγκιών κατά την ανίχνευση και συγκομιδή τροφής, δηλαδή της εύρεσης της γρηγορότερης διαδρομής. Για το λόγο αυτό ο αλγόριθμος AS θα παρουσιασθεί στη μορφή για την επίλυση του TSP.
Πιο συγκεκριμένα, στο TSP το ζητούμενο είναι η εύρεση της συντομότερης διαδρομής που συνδέει n πόλεις. Οι περιορισμοί που τίθενται είναι:
κάθε πόλη πρέπει να περιλαμβάνεται μια φορά στη διαδρομή
η διαδρομή πρέπει να είναι κλειστή, δηλαδή ο «πωλητής» πρέπει να επιστρέψει στην πόλη από όπου ξεκίνησε.

Οι αποστάσεις μεταξύ των πόλεων δεν είναι απαραίτητο να είναι συμμετρικές καθώς η απόσταση από την πόλη i στη j μπορεί να είναι διαφορετική από την απόσταση από την j πόλη στην i (τότε ορίζεται το ATSP: Asymmetric Traveling Salesman Problem).



Η κάθε λύση για το TSP σχηματίζεται με τη διαδοχική μετάβαση των «ψηφιακών» μυρμηγκιών από τη μια πόλη στην άλλη με κάποια πιθανότητα. Το «ταξίδι» ολοκληρώνεται όταν το «ψηφιακό» μυρμήγκι επιστέψει στην αρχική πόλη. Τότε βαθμολογείται η λύση που έχει επιτύχει με κριτήριο το συνολικό μήκος της διαδρομής και προστίθεται η ανάλογη φερομόνη στο μονοπάτι που ακολούθησε. Η παραπάνω διαδικασία επαναλαμβάνεται για κάθε άτομο της αποικίας μέχρι να συμπληρωθεί ο ζητούμενος αριθμός επαναλήψεων.

5. Εφαρμογή Αλγόριθμου ACO στο Πρόβλημα *P*-Median

Στην ενότητα αυτή παρουσιάζουμε μία μέθοδο εφαρμογής του αλγόριθμου ACO για την επίλυση του προβλήματος p-median όπως αυτό έχει διατυπωθεί στο (Branco, I. και Coelho, J., 1990). Ειδικότερα, η προσέγγιση που ακολουθούμε, στηρίζεται στο (Zohrehbandian, M., και Hamidnia Namini, S., 2010) και εφαρμόζει μια σειρά τροποποιήσεων στο κλασσικό Πρόβλημα Δρομολόγησης Οχημάτων, οι οποίες επιτρέπουν την αναζήτηση της βέλτιστης λύσης του προβλήματος p-median.

Το πρόβλημα VRP

Σε ένα πρόβλημα τύπου V.R.P. (Vehicle Routing Problem) αναζητούνται οι διαδρομές που πρέπει να ακολουθήσει κάθε ένα από τα διαθέσιμα οχήματα για να εξυπηρετηθούν όλοι οι πελάτες (ζήτηση). Κάθε όχημα ξεκινάει από την αποθήκη και καταλήγει πάλι σε αυτήν. Συνεπώς οι περιορισμοί του προβλήματος αφορούν την ικανοποίηση της ζήτησης όλων των πελατών. Τέλος, επιδιώκεται η ελαχιστοποίηση του μεταφορικού κόστους μεταφοράς των προϊόντων [27].

Κάθε όχημα επισκέπτεται μονάχα μία φορά κάθε πελάτη. Όπως επίσης και κάθε διαδρομή (που ξεκινάει από την αποθήκη και καταλήγει σε αυτήν) εκτελείται από ένα μόνο όχημα. Οι περιορισμοί περιγράφουν τις αποστάσεις μεταξύ των πελατών, τη συνολική διανυόμενη απόσταση για κάθε ένα όχημα, την χωρητικότητα των οχημάτων καθώς επίσης οτιδήποτε περιγράφει τα χαρακτηριστικά του οδικού δικτύου διανομής. Δεν υπάρχει περιορισμός για τον αριθμό των πελατών που πρέπει να επισκεφτεί κάθε όχημα, εκτός του ότι πρέπει να επισκεφτεί τουλάχιστον έναν πελάτη.

Η συνολική ζήτηση των πελατών αλλά και η ζήτηση του κάθε πελάτη καθορίζουν τον αριθμό των οχημάτων που θα χρειαστούν. Κάθε όχημα έχει συγκεκριμένη χωρητικότητα η οποία εκφράζεται σαν ένα άνω όριο σχετικά με το βάρος, ή τον όγκο, ή αριθμό παλετών ή προϊόντων που μπορεί να μεταφέρει το κάθε όχημα. Το κόστος που συνδέεται με την χρήση του εκάστοτε οχήματος, στην μοντελοποίηση που ακολουθήθηκε στην

παρούσα μεταπτυχιακή διατριβή, είναι ευθέως ανάλογο με την απόσταση που διανύει το κάθε όχημα.

Τα δεδομένα για τον κάθε πελάτη αφορούν τις συντεταγμένες θέσης του και τη ζήτησή του η οποία πρέπει να ικανοποιηθεί εξ' ολοκλήρου. Σε κάθε διαδρομή πρέπει να ικανοποιούνται λειτουργικοί περιορισμοί που έχουν να κάνουν με την ποσότητα των μεταφερόμενων προϊόντων, και τα χαρακτηριστικά των οχημάτων. Για παράδειγμα, σε κάθε διαδρομή η μεταφερόμενη ποσότητα δεν θα πρέπει να ξεπερνά την χωρητικότητα του οχήματος. Οι οδηγοί των οχημάτων είναι απαραίτητο να ικανοποιούν διάφορους περιορισμούς που έχουν να κάνουν με συμβόλαια εργατικών ενώσεων καθώς και κανονισμούς της εταιρείας για την οποία εργάζονται. Το σύνολο των περιορισμών που αναφέρεται στα οχήματα μπορεί να περικλείει και περιορισμούς που προκύπτουν από τους οδηγούς. Ο χρόνος εργασίας κατά τη διάρκεια της ημέρας, ο αριθμός και η διάρκεια των διαλειμμάτων και οι υπερωρίες αποτελούν ορισμένους μόνο από τους "κανόνες" τους οποίους πρέπει να ληφθούν υπόψη κατά τη διαδικασία μοντελοποίησης για τον καθορισμό των περιορισμών.

δρομολογίου, αθροίζονται τα κόστη μεταφοράς από τον ένα πελάτη στον άλλον καθώς και τα κόστη μεταφοράς από και προς την αποθήκη. Το κόστος μετάβασης από έναν κόμβο α σε έναν κόμβο β, είναι το cαβ το οποίο ανήκει στη συντομότερη διαδρομή μεταξύ των κόμβων α και β. Ταυτόχρονα και ο χρόνος μετάβασης ταβ , ισούται με το άθροισμα των χρόνων μετάβασης όλων των επιμέρους μεταβάσεων (από πελάτη σε πελάτη και από και προς την αποθήκη) που ανήκουν στην συντομότερη διαδρομή μεταξύ των πελατών α και β.

Για το πρόβλημα δρομολόγησης οχημάτων ακολουθείται η μοντελοποίηση που περιγράφεται παρακάτω. Έστω ότι οι πελάτες και η αποθήκη αντιστοιχούν σε κάποιους κόμβους με συγκεκριμένες συντεταγμένες στο χώρο και μαζί με τα τόξα που ενώνουν τους κόμβους των πελατών και της αποθήκης αποτελούν ένα γράφημα A(K,T). Οι κόμβοι K=(2,...,M) αντιπροσωπεύουν τους πελάτες και την αποθήκη αντιστοιχεί στον κόμβο 1. Έστω ότι ο αριθμός των οχημάτων είναι ν όπου κάθε όχημα έχει χωρητικότητα X. Ο πίνακας με τα κόστη (κ) έχει διαστάσεις MxM και καβ είναι το κόστος μεταφοράς από τον κόμβο α στον κόμβο β. Η ζήτηση του κόμβου α είναι ζα και φυσικά η ζήτηση του κόμβου 1 (δηλαδή της αποθήκης) είναι ζ1=0. Στόχος είναι η εύρεση του βέλτιστου δ,

των διαδρομών λοιπόν εκείνων που ελαχιστοποιούν το συνολικό κόστος. Άρα, η αντικειμενική συνάρτηση θα έχει ως εξής:

$$\min \sum_{\alpha=1}^{M} \sum_{\beta=1}^{M} \sum_{\gamma=1}^{\nu} \kappa_{\alpha\beta}\, \delta_{\alpha\beta\gamma} \qquad (26)$$

Οι περιορισμοί θα έχουν ως εξής:

Όλες οι μεταβλητές απόφασης είναι δυαδικές (0 ή 1):

$$\delta_{\alpha\beta\gamma} \in \{0,1\} \qquad \forall\, \gamma \in 1, \dots, \nu \quad \kappa\alpha\iota \quad \forall\, \alpha \in 1, \dots, M \quad \kappa\alpha\iota \quad \forall\, \beta \in 1, \dots, M$$

Από την αποθήκη αναχωρούν ν οχήματα και όλα τα οχήματα έχουν χωρητικότητα ίση με Χ:

$$\sum_{\alpha=1}^{M} \sum_{\beta=1}^{M} \delta_{\alpha\beta\gamma}\, \zeta_{\alpha} \leq X \qquad \forall\, \gamma \in 1, \dots, \nu \qquad (27)$$

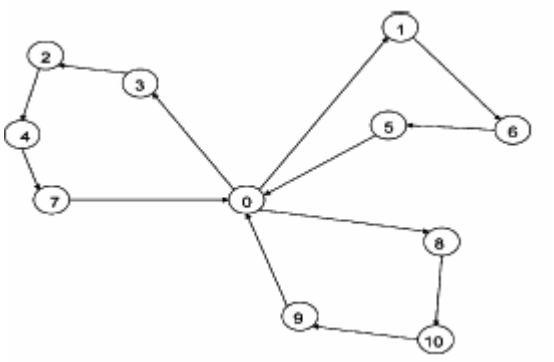
5.1. Μετασχηματισμός του Προβλήματος *P*-Median

Όπως έχουμε ήδη αναφέρει στην Ενότητα 2, το πρόβλημα p-median μπορεί να αναπαρασταθεί με τη βοήθεια γράφων ως εξής: Έστω G = (V, E) ένας πλήρης κατευθυνόμενος γράφος με κόστος cij για κάθε μετάβαση (i, j). Το p-median πρόβλημα συνίσταται στην επιλογή p κορυφών (εγκαταστάσεις) από το V και ανάθεσή των κορυφών του V (πελάτες) σε p κατευθυνόμενους υπογράφους χωρίς κοινές κορυφές με τέτοιο τρόπο ώστε να επιτυγχάνεται ελαχιστοποίηση του κόστους διανομής. Κάθε κορυφή ενός τέτοιου υπογράφου, μπορεί να επιλεγεί ως θέση εγκατάστασης. Συνοπτικά, το πρόβλημα p-median είναι ισοδύναμο με το προσδιορισμό p ασύνδετων κατευθυνόμενων γράφων μεταξύ τους που καλύπτουν το σύνολο κορυφών V, υπό τη συνθήκη ισχύος μιας αντικειμενικής συνάρτησης.

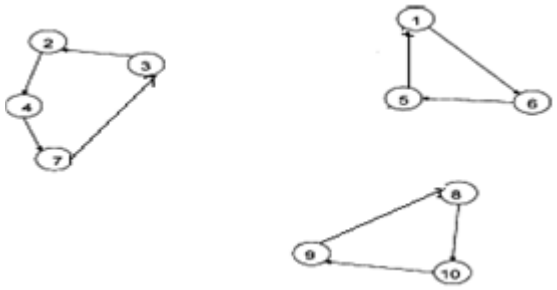
Στην ενότητα αυτή θα δείξουμε ότι το p-median μπορεί να μετασχηματιστεί σε Πρόβλημα Δρομολόγησης Οχημάτων (Vehicle Routing Problem – VPR). Αυτό που επιτυγχάνουμε μέσω του μετασχηματισμού αυτού, είναι ότι μας παρέχεται πλέον η δυνατότητα να αξιοποιήσουμε μεθόδους επίλυσης του VPR (όπως είναι ο αλγόριθμος ACO που αποτελεί αντικείμενο μελέτης της παρούσας εργασίας) για την επίλυση του προβλήματος p-median.

Για το μετασχηματισμό του p-median σε VRP, ας υποθέσουμε ότι η κορυφή 0 είναι μια «υποθετική» εγκατάσταση. Έστω ci0 = c0i = 0 για κάθε πελάτη i = 1, ..., n. Επιπλέον, με κάθε κορυφή i συσχετίζουμε ένα μη αρνητικό συντελεστή ζήτησης qi = 1 και με κάθε όχημα, συσχετίζουμε ένα μη αρνητικό συντελεστή χωρητικότητας Q = n. Τα Σχήματα 2a και 2b δείχνουν ότι κάθε λύση του μετασχηματισμένου VRP αντιστοιχεί σε μια λύση του αρχικού p-median. Θα πρέπει εδώ να σημειώσουμε ότι η αντικειμενική τιμή (objective value) του p-median είναι ίση με το άθροισμα της αντικειμενικής τιμής του VRP και του κόστους μετάβασης (i, j), δεδομένου ότι και οι δύο κορυφές i και j είναι γειτονικές της «υποθετικής» εγκατάστασης στον ίδιο υπογράφο.

Σχήμα 2a: Λύση μετασχηματισμένου προβλήματος.



Σχήμα 2b: Λύση αρχικού προβλήματος.



5.2. Επιμέρους Βήματα Αλγορίθμου

Η γενική μας προσέγγιση για τη χρησιμοποίηση του αλγόριθμου ACO για επίλυση του προβλήματος p-median, βασίζεται στο (Bell, J. και McMullen, P., 2004) οι οποίοι χρησιμοποίησαν τον αλγόριθμο ACO για την επίλυση του VRP. Αρχικά, κάθε μυρμήγκι ξεκινά από την «υποθετική» εγκατάσταση και το σύνολο κορυφών (πελάτες) που περιέχονται στη λύση είναι κενό. Το μυρμήγκι επιλέγει τον επόμενο πελάτη που θα επισκεφτεί από μία λίστα εφικτών (feasible) θέσεων μέσω μιας στοχαστικής διαδικασίας (probabilistic formula), ενώ ταυτόχρονα οι κορυφές τις οποίες έχει ήδη επισκεφτεί τοποθετούνται σε μια βοηθητική μνήμη (working memory). Όταν ένα μυρμήγκι k βρίσκεται στην κορυφή i, επιλέγει να μεταβεί σε κάποια κορυφή j την οποία δεν έχει ακόμη επισκεφτεί, χρησιμοποιώντας την ακόλουθη σχέση:

$$j = \begin{cases} ArgMax\left\{(\tau_{iu})(\eta_{iu})^{\beta}\right\} & q \le q_0 \\ s & q > q_0 \end{cases}, \qquad (28)$$

όπου τ_{iu} είναι η ποσότητα της φερομόνης που εναποτίθεται στη μετάβαση από την τρέχουσα κορυφή i στην ακόμη μη επισκεφθείσα κορυφή u, ενώ ως η_{iu} ορίζεται το αντίστροφο της απόστασης ανάμεσα στις δύο κορυφές. Η παράμετρος β > 0 συσχετίζει την απόσταση με την ποσότητα φερομόνης που χρησιμοποιείται στον αλγόριθμο επιλογής. Οι κορυφές τις οποίες έχει ήδη επισκεφτεί το μυρμήγκι k αποθηκεύονται, όπως ήδη αναφέραμε, σε μια βοηθητική μνήμη Mk και δεν υπολογίζονται εκ νέου ως

δυνητικές επιλογές μετάβασης. Τέλος, το q είναι μια τυχαία μεταβλητή ομοιόμορφα κατανεμημένη στο διάστημα [0,1], ενώ qo είναι παράμετρος του προβλήματος που πρέπει να οριστεί εκ των προτέρων.

Σε κάθε βήμα, το μυρμήγκι επιλέγει τη μετάβαση με την υψηλότερη τιμή από την εξίσωση (28) παραπάνω όταν το q είναι μικρότερο ή ίσο του qo, ενώ αν το q είναι μεγαλύτερο του qo η νέα κορυφή επιλέγεται μέσω μιας τυχαίας μεταβλητής s, βάσει της κατανομής πιθανότητας Pij, η οποία ευνοεί τις συντομότερες μεταβάσεις με τα υψηλότερα επίπεδα φερομόνης:

$$P_{i,j} = \begin{cases} \dfrac{(\tau_{iu})(\eta_{iu})^{\beta}}{\sum_{u \notin M_k} (\tau_{iu})(\eta_{iu})^{\beta}} & j \notin M_k \\ 0 & j \in M_k \end{cases} \qquad (29)$$

Χρησιμοποιώντας τις σχέσεις (28) και (29), το kστο μυρμήγκι μπορεί, ξεκινώντας από μία εγκατάσταση, είτε να ακολουθήσει την καλύτερη διαδρομή από αυτές που έχουν ήδη χρησιμοποιηθεί, είτε να επιλέξει μια διαδρομή τυχαία, βάσει μιας πιθανοτικής κατανομής που στηρίζεται στην απόσταση και τη συσσωρευμένη φερομόνη. Επιπλέον, κάθε φορά που ένα μυρμήγκι k επισκέπτεται την εγκατάσταση από την οποία ξεκίνησε, ο αριθμός των υπογράφων που δημιουργούνται από το μυρμήγκι k αυξάνεται κατά έναν. Η διαδικασία αυτή συνεχίζεται έως ότου τουλάχιστον X(p - t) κορυφές να απομένουν προς επίσκεψη, όπου t είναι ο αριθμός των υπογράφων που κατασκευάζονται από το μυρμήγκι k. Αν οι κορυφές που δεν έχει επισκεφθεί είναι λιγότερες από X(p-t), τότε το μυρμήγκι k τελειώνει το τρέχον κύκλωμα και δημιουργεί ένα νέο κύκλωμα, ξεκινώντας από τον επόμενο κόμβο. Αυτό είναι απαραίτητο για τη δόμηση μιας εφικτής (feasible) λύσης του p-median. Στην περίπτωση αυτή, το μυρμήγκι k ολοκληρώνει την κατασκευή του υπογράφου στη συγκεκριμένη εγκατάσταση και ξεκινά την κατασκευή ενός νέου υπογράφου ξεκινώντας από την εγκατάσταση αυτή. Η παραπάνω διαδικασία επαναλαμβάνεται έως ότου έχουμε επισκεφτεί κάθε πελάτη.

Ο αλγόριθμος ACO που προτείνεται, δημιουργεί μια πλήρη VRP λύση για κάθε μυρμήγκι, πριν το επόμενο αρχίσει να κινείται, και όπως έχουμε ήδη αναφέρει στην προηγούμενη ενότητα, το συνολικό κόστος διανομής (αντικειμενική τιμή του p-median) υπολογίζεται αθροίζοντας το κόστος των μεταβάσεων (i, j), όπου οι πελάτες i και j είναι γειτονικοί της «υποθετικής» εγκατάστασης σε έναν υπογράφο.

Τα ίχνη της φερομόνης που εναποθέτουν τα μυρμήγκια πρέπει να μεταβάλλονται ώστε να αντανακλούν την ποιότητα της επιλογής κάθε μυρμηγκιού ή με άλλα λόγια την ποιότητα των εκάστοτε λύσεων. Αυτή η μεταβολή στα ίχνη φερομόνης, είναι ζωτικής σημασίας στη διαδικασία «προσαρμοστικής μάθησης» του αλγορίθμου ACO και συμβάλλει στη διασφάλιση της βελτίωσης των μεταγενέστερων λύσεων. Στις τεχνικές ενημέρωσης του ίχνους φερομόνης, συμπεριλαμβάνονται η τοπική ενημέρωση του ίχνους, η οποία λαμβάνει χώρα μετά την εξεύρεση κάποιας λύσης από κάθε μυρμήγκι και η συνολική ενημέρωση του ίχνους της καλύτερης λύσης που έχει επιτευχθεί μετά από ένα προκαθορισμένο αριθμό λύσεων.

Οι τοπικές ενημερώσεις λαμβάνουν χώρα αρχικά, με μείωση της ποσότητας φερομόνης σε όλες τις διαδρομές που έχουν επιλεγεί, ώστε να προσημειωθεί η φυσική «εξάτμιση» της φερομόνης και να διασφαλιστεί ότι καμία διαδρομή δε θα καταστεί κυρίαρχη. Αυτό γίνεται με την εξίσωση τοπικής ενημέρωσης ίχνους φερομόνης $\tau_{ij} = (1-\alpha)\tau_{ij} + \alpha\tau_0$, όπου α είναι μια προεπιλεγμένη παράμετρος που ελέγχει την ταχύτητα εξάτμισης και τ_0 είναι η αρχική ποσότητα φερομόνης που υπάρχει σε όλες τις διαδρομές.

Μετά από ένα προκαθορισμένο αριθμό εφικτών (feasible) λύσεων, μια συνολική ενημέρωση ίχνους πραγματοποιείται, προσθέτοντας φερομόνη σε όλες τις ακμές που εμπεριέχονται στη καλύτερη λύση που έχει βρεθεί μέχρι τότε από όλα τα μυρμήγκια. Η συνολική ενημέρωση ίχνους φερομόνης γίνεται βάσει της σχέσης $\tau_{ij} = (1-\alpha)\tau_{ij} + \alpha D^{-1}$, όπου D^{-1} είναι ο αντίστροφος της τιμής της καλύτερης μέχρι τότε λύσης που έχει βρεθεί. Αυτή η ενημέρωση της φερομόνης, ενθαρρύνει τη μελλοντική χρήση συντομότερων διαδρομών και αυξάνει την πιθανότητα ότι οι μελλοντικές διαδρομές θα εμπεριέχουν

ακμές που περιλαμβάνονται στις καλύτερες λύσεις. Η διαδικασία αυτή επαναλαμβάνεται ανά τακτά χρονικά διαστήματα και η καλύτερη λύση που προκύπτει από όλες τις επαναλήψεις, παρουσιάζεται τελικά ως τελική λύση του προτεινόμενου αλγορίθμου και αναπαριστά μια προσέγγιση της βέλτιστης λύσης του προβλήματος.

Για την περαιτέρω βελτίωση της παραπάνω λύσης του προβλήματος, μπορούμε ακόμη να χρησιμοποιήσουμε τεχνικές βελτίωσης διαδρομών, όπως για παράδειγμα τη διαδικασία τοπικής αντικατάστασης κόμβων η οποία παίζει το ρόλο ενός τοπικού heuristic βελτίωσης εντός κάθε επιμέρους διαδρομής που δημιουργείται από το κάθε μυρμήγκι. Μία από τις τεχνικές που χρησιμοποιούνται για το σκοπό αυτό, είναι το δημοφιλές «2-opt heuristic», όπου ελέγχεται αν κάποια από τις πιθανές λύσεις που προκύπτουν από τις αμοιβαίες αντικαταστάσεις ζευγαριών κορυφών, τις οποίες έχει επισκεφτεί ένα μυρμήγκι, οδηγεί τελικά σε βελτίωση της αντικειμενικής τιμής.

Τα επιμέρους βήματα του παραπάνω αλγόριθμου παρουσιάζονται συνοπτικά παρακάτω:

<u>Βήμα 1-Αρχικοποίηση:</u> Αρχικοποίησε τις μεταβλητές q0, α, β, τ0, ΑΕ, kmax. Για κάθε μυρμήγκι k αρχικοποίησε επίσης τις μεταβλητές Sk = 0 (σύνολο κορυφών που περιέχονται στην υπό ανάπτυξη λύση), Mk = 0 (βοηθητική μνήμη).

Επανέλαβε για ένα προκαθορισμένο αριθμό επαναλήψεων ΑΕ:

<u>Βήμα 2-Δημιουργία Λύσεων:</u>: Για κάθε μυρμήγκι k επανέλαβε την παρακάτω διαδικασία έως ότου έχεις επισκεφτεί κάθε κορυφή:

<u>Βήμα 2α-Μετάβαση στην Επόμενη Κορυφή:</u> Επέλεξε να μεταβείς από την κορυφή i, στην κορυφή j την οποία δεν έχεις ακόμη επισκεφτεί, χρησιμοποιώντας τη formula:

$$j = \begin{cases} ArgMax\left\{(\tau_{iu})(\eta_{iu})^{\beta}\right\} & q \le q_0 \\ s & q > q_0 \end{cases}.$$

Αποθήκευσε τη μετάβαση (i, j) στα Sk και Mk.

<u>Βήμα 3-Τοπική Εξάτμιση Φερομόνης</u> Μείωσε την ποσότητα φερομόνης σε όλες τις διαδρομές που έχουν επιλεγεί, ώστε να προσομοιωθεί η φυσική «εξάτμιση»,

χρησιμοποιώντας την εξίσωση τοπικής ενημέρωσης ίχνους φερομόνης $\tau_{ij} = (1-\alpha)\tau_{ij} + \alpha\tau_0$.

<u>Βήμα 4-Γενική Ανανέωση Φερομόνης:</u> Μετά από ένα προκαθορισμένο αριθμό εφικτών (feasible) λύσεων, πραγματοποίησε μια συνολική ενημέρωση ίχνους, προσθέτοντας φερομόνη σε όλες τις ακμές που εμπεριέχονται στη καλύτερη λύση που έχει βρεθεί μέχρι τότε από όλα τα μυρμήγκια, χρησιμοποιώντας τη σχέση $\tau_{ij} = (1-\alpha)\tau_{ij} + \alpha D^{-1}$.
Τέλος Βρόγχου Επαναλήψεων ΑΕ

<u>Βήμα 5-Τελική Λύση:</u> Τελική λύση είναι η καλύτερη λύση που έχει βρεθεί μετά την ολοκλήρωση του προεπιλεγμένου αριθμού επαναλήψεων

5.3. Υλοποίηση Αλγόριθμου – Αποτελέσματα Εκτέλεσης

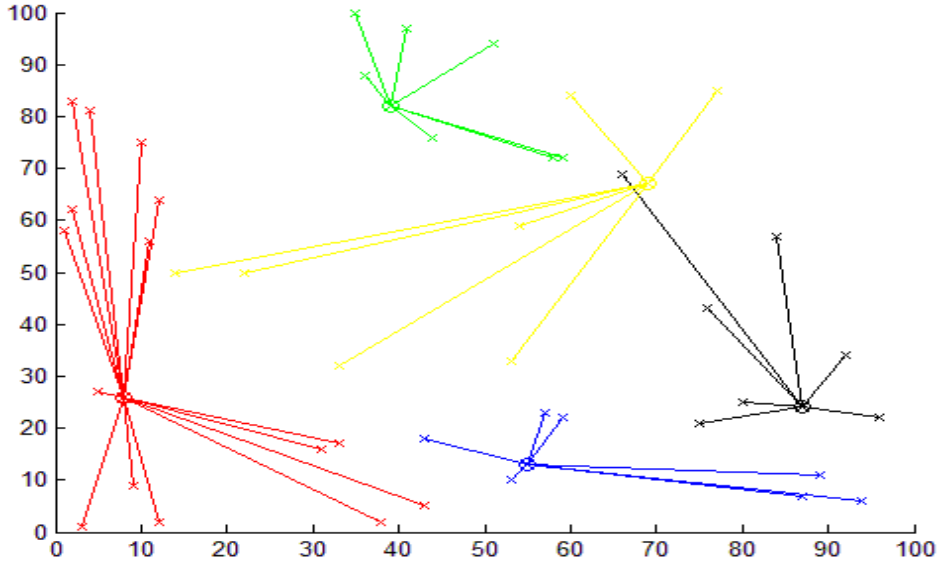
Ο αλγόριθμος εκτελέστηκε με χρήση του Matlab για 20 test προβλήματα, δέκα από τα οποία 5-median και δέκα 10-median, όπως αυτά έχουν οριστεί στο (Osman, I. και Christofides, N., 1994). Στα σχήματα που ακολουθούν παρουσιάζουμε γραφικά, το αποτέλεσμα εκτέλεσης του αλγορίθμου. Κάθε υπογράφος
που σχηματίζεται παρουσιάζεται με διαφορετικό χρώμα. Οι παράμετροι που χρησιμοποιήθηκαν είναι: q0=0.5 (προεπιλεγμένη τιμή στο (0,1)), alpha=0.0010 (συντελεστής εξάτμισης φερομόνης στο (0,1)), beta=1 (συντελεστής συσχέτισης απόστασης – φερομόνης), t0=1 (αρχική ποσότητα φερομόνης που υπάρχει σε όλες τις διαδρομές), ΑΕ=100 (αριθμός επαναλήψεων), kmax=50 (αριθμός μυρμηγκιών). Στον παρακάτω πινάκα παρουσιάζονται τα επιμέρους χαρακτηριστικά για όλα τα προβλήματα pmedian που επιλύθηκαν.

Πίνακας 1: Κύρια Χαρακτηριστικά προβλημάτων P-median.

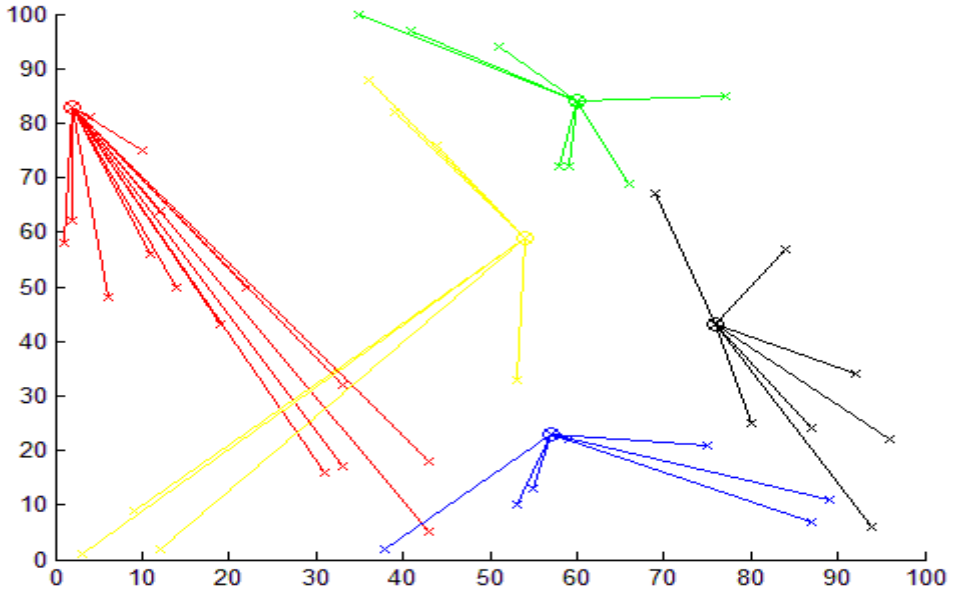
Test Problem	Customers	Medians	Median's Capacity	Best Solution
1	50	5	120	713
2	50	5	120	740
3	50	5	120	751
4	50	5	120	651
5	50	5	120	664
6	50	5	120	778
7	50	5	120	787
8	50	5	120	820
9	50	5	120	715
10	50	5	120	829
11	100	10	120	1006
12	100	10	120	966
13	100	10	120	1026
14	100	10	120	982
15	100	10	120	1091
16	100	10	120	954
17	100	10	120	1034
18	100	10	120	1043
19	100	10	120	1031
20	100	10	120	1005

Επίσης, στον Πίνακα 1 που ακολουθεί, εμφανίζονται οι λύσεις που υπολογίζει ο παρόν αλγόριθμος σε σύγκριση με τη βέλτιστη λύση από το (Osman, I. και Christofides, N., 1994). Είναι εμφανές ότι ο προτεινόμενος αλγόριθμος δίνει λύσεις συγκρίσιμες και ελαφρώς μεγαλύτερες από τις βέλτιστες για τα test problems που χρησιμοποιήθηκαν.

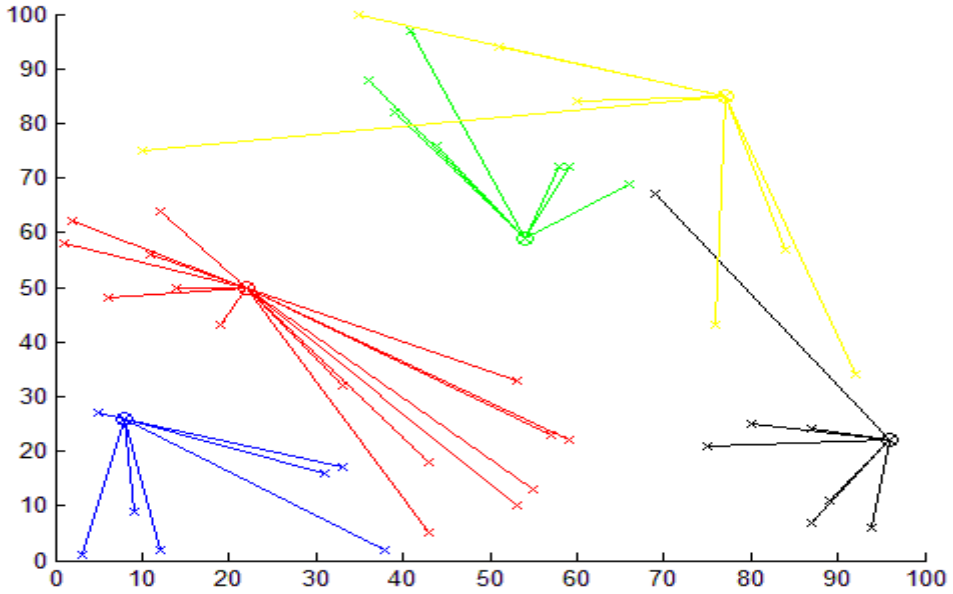
Σχήμα 3: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 1.



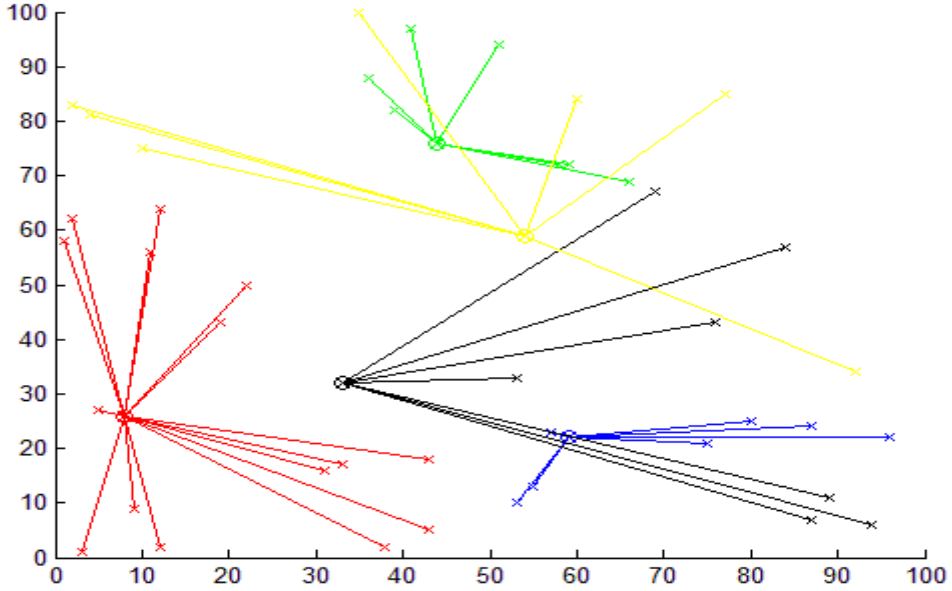
Σχήμα 4: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 2.



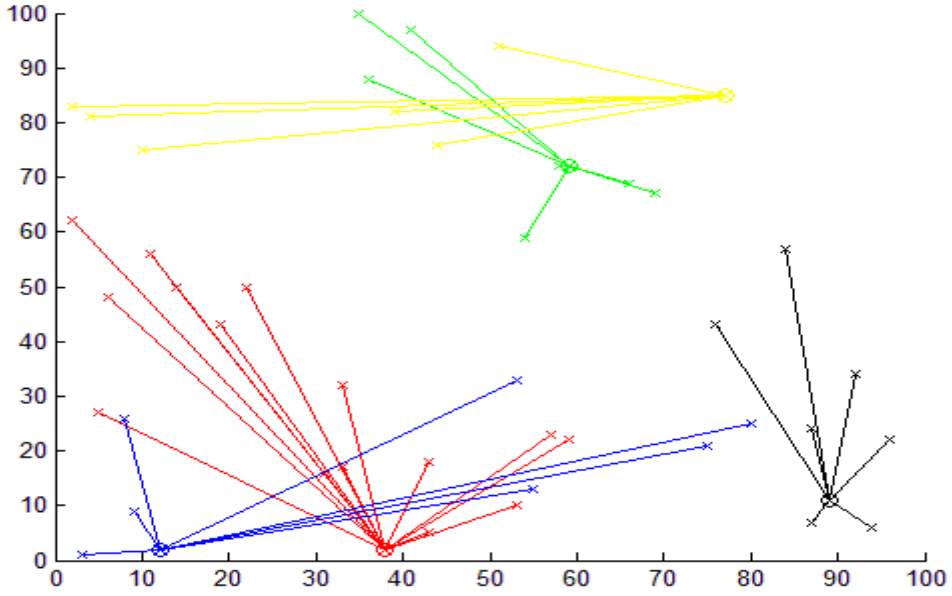
Σχήμα 5: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 3.



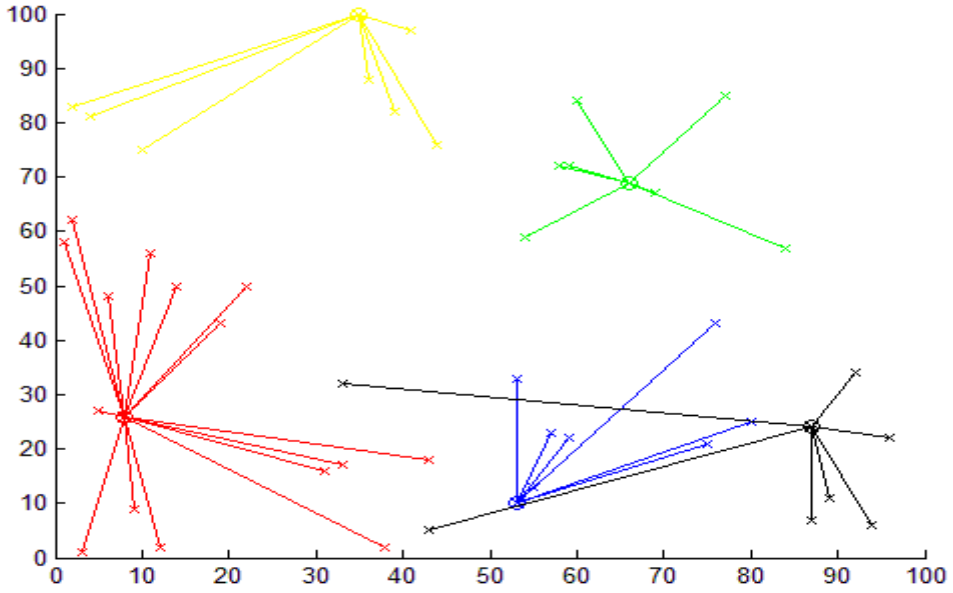
Σχήμα 6: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 4.



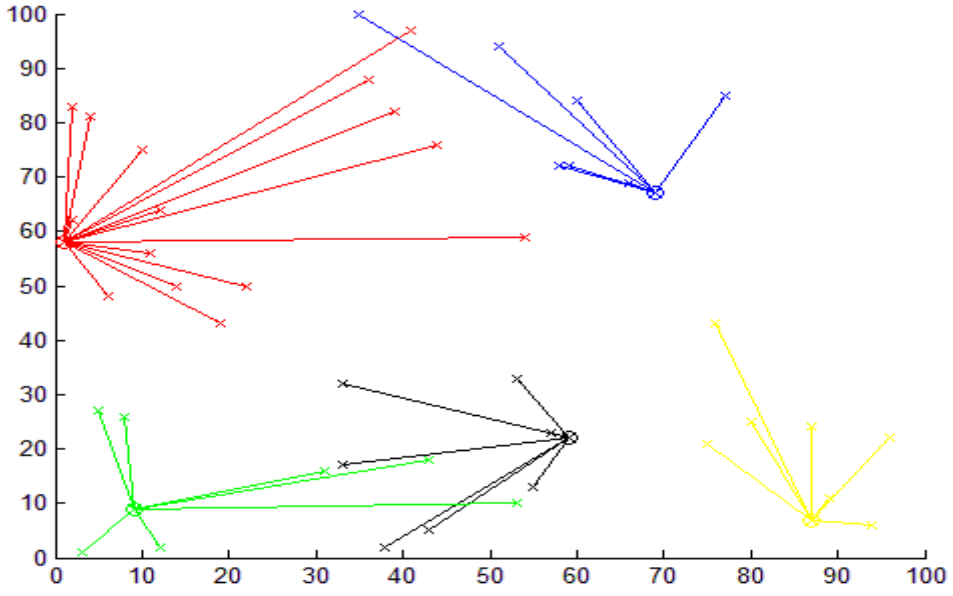
Σχήμα 7: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 5.



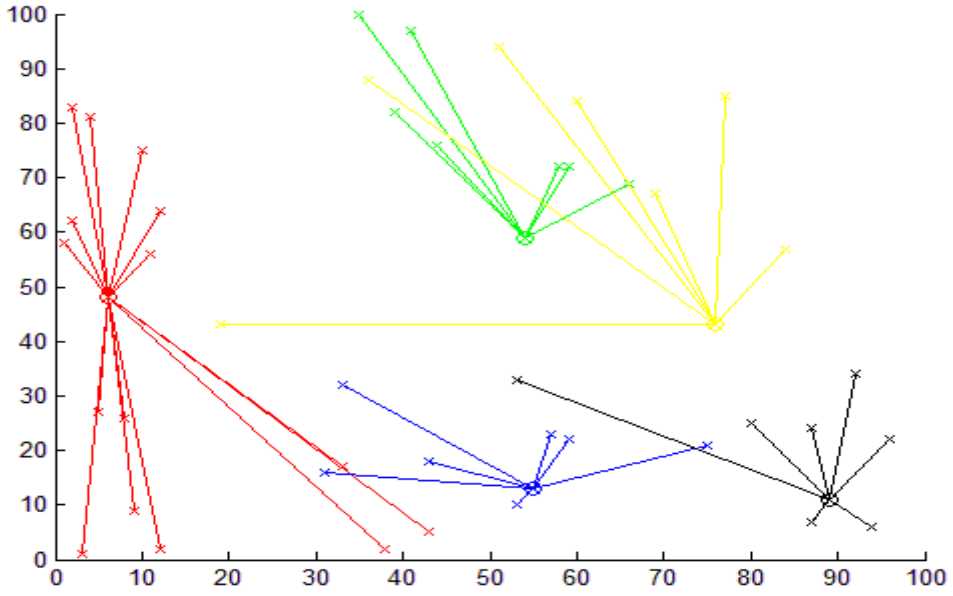
Σχήμα 8: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 6.



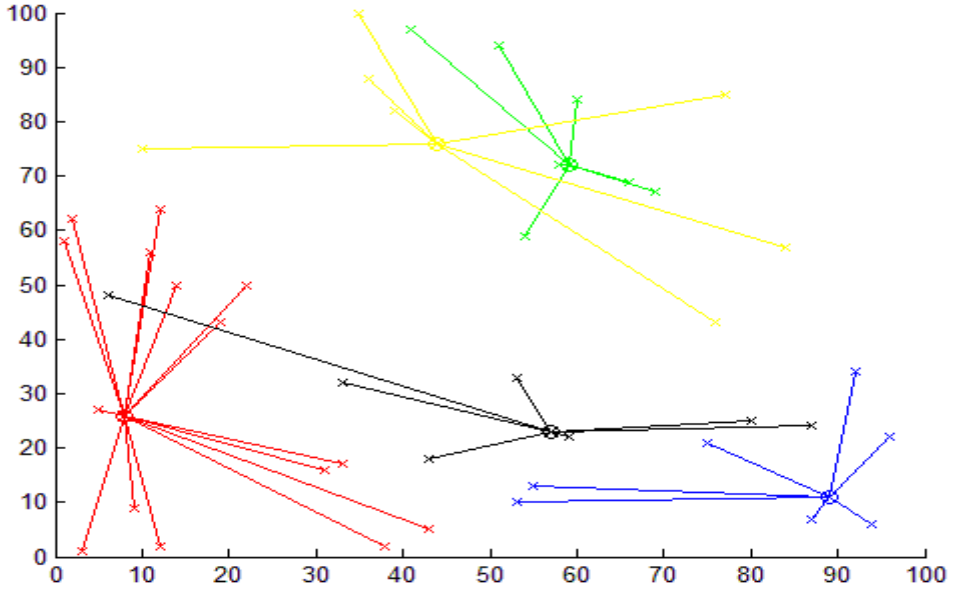
Σχήμα 9: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 7.



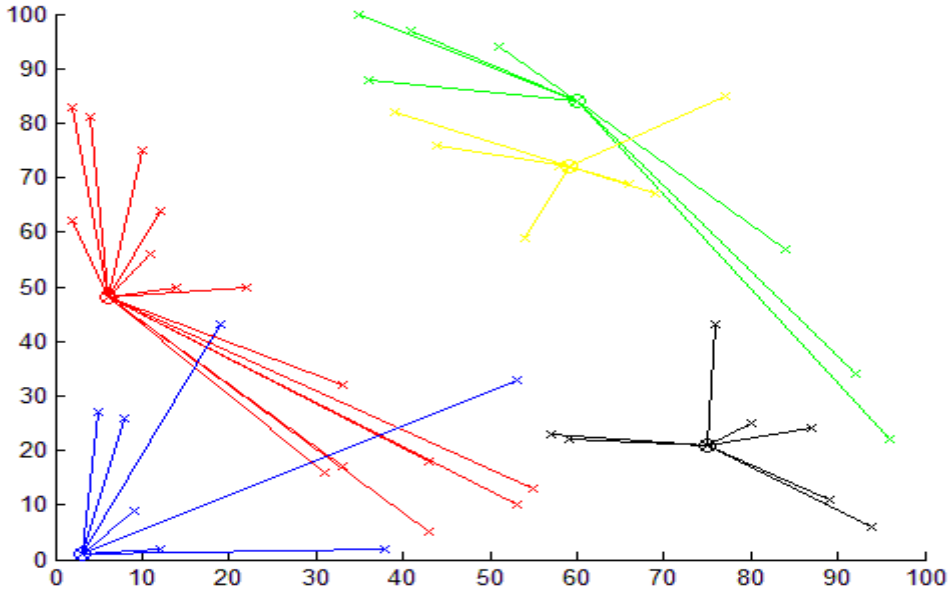
Σχήμα 10: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 8.



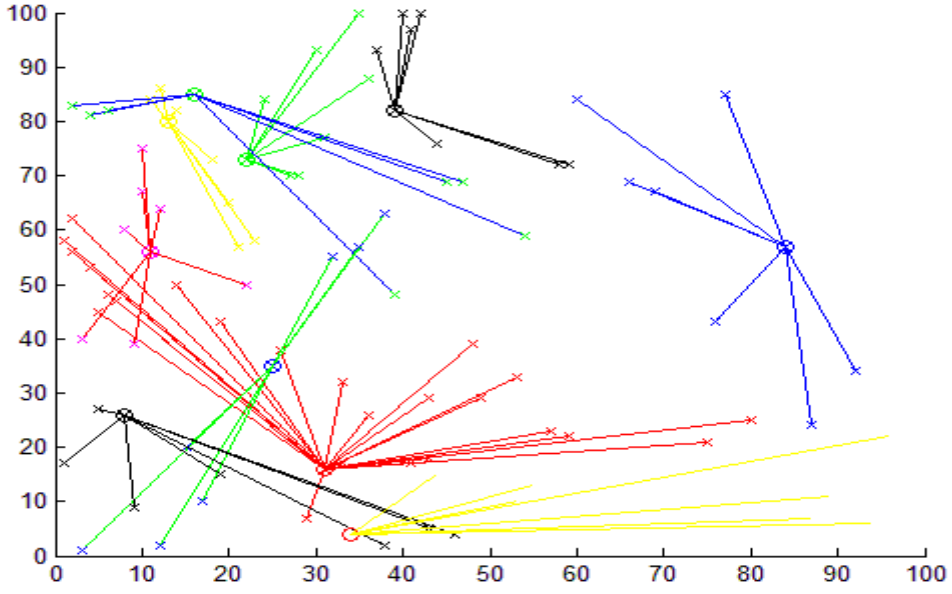
Σχήμα 11: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 9.



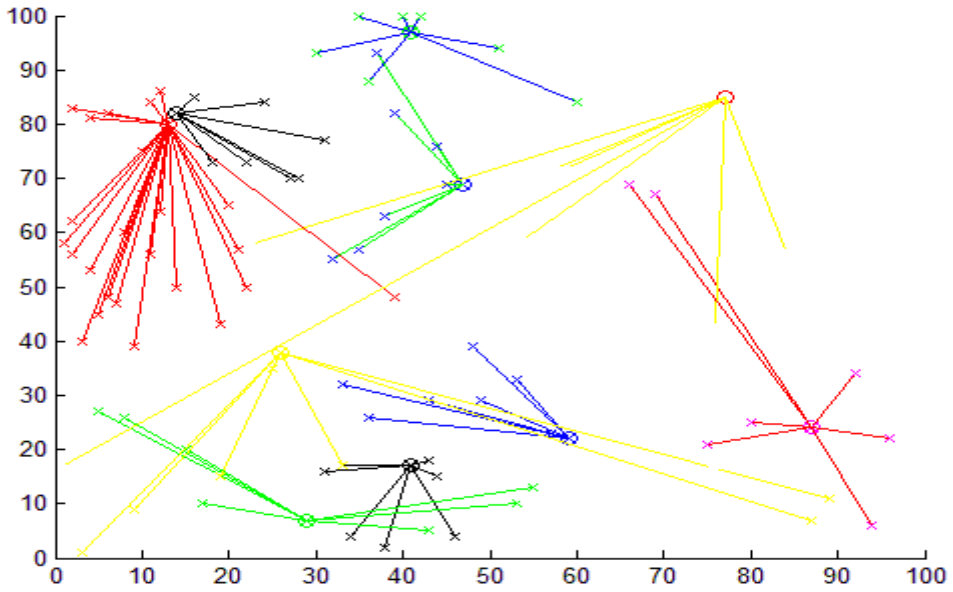
Σχήμα 12: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 10.



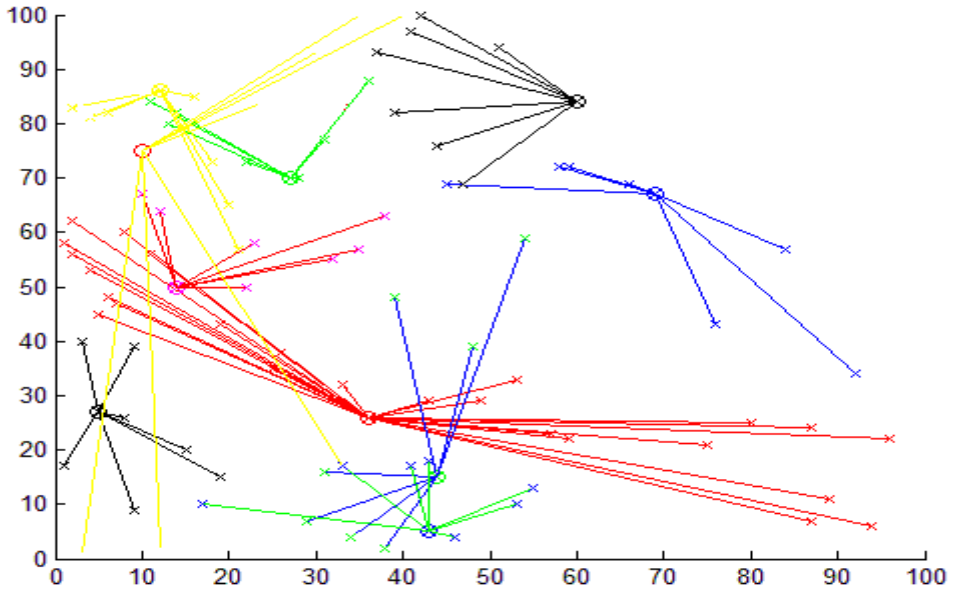
Σχήμα 13: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 11.



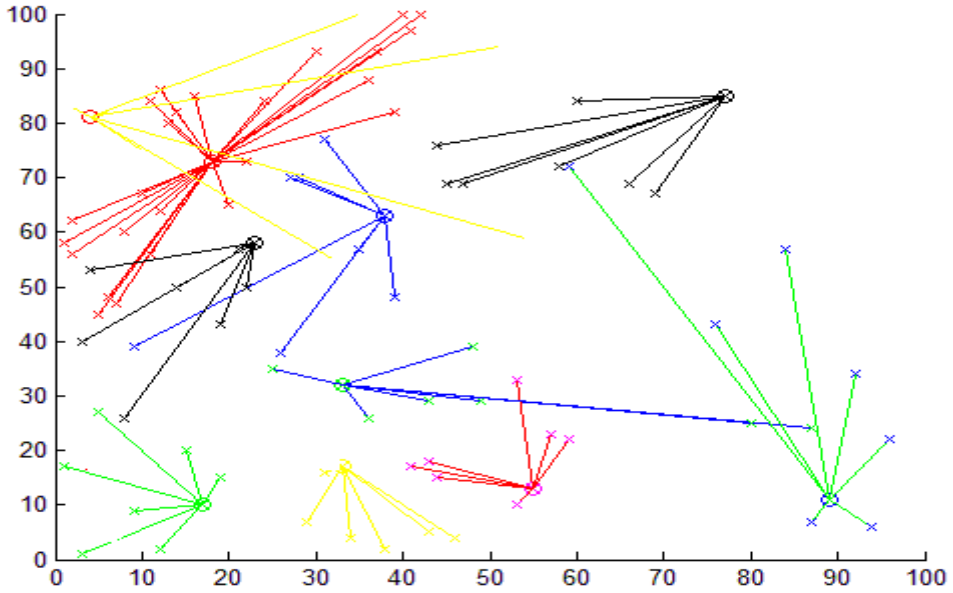
Σχήμα 14: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 12.



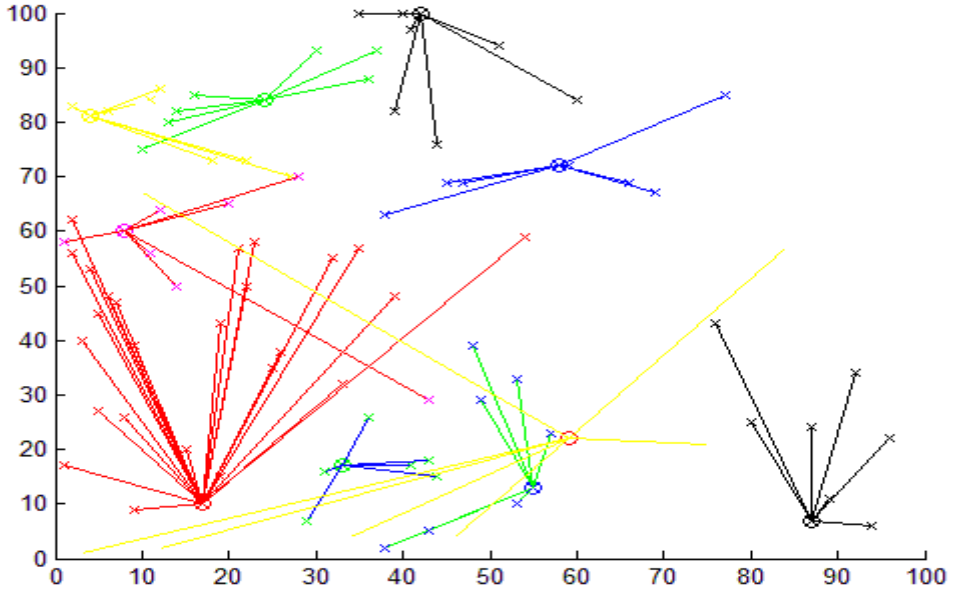
Σχήμα 15: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 13.



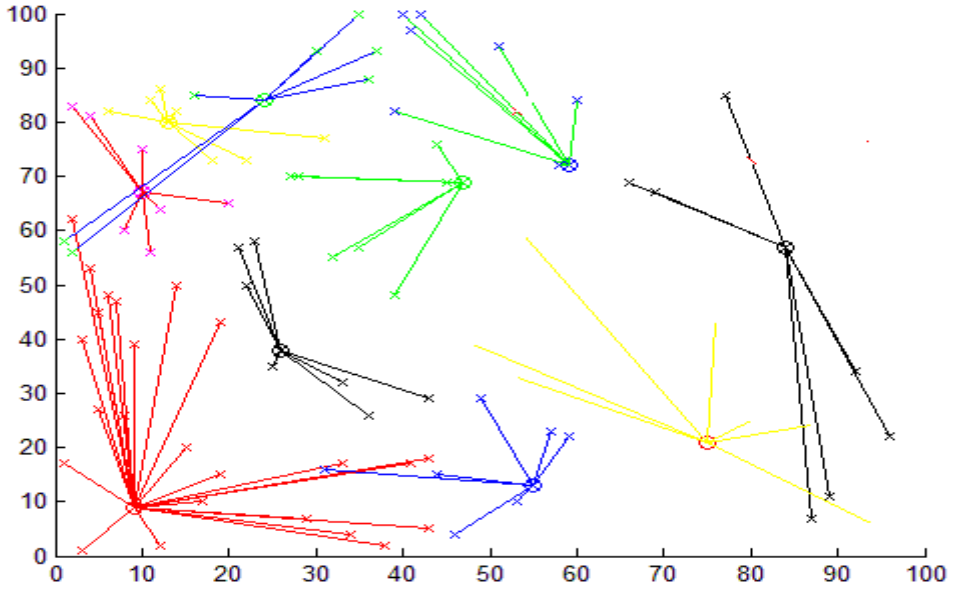
Σχήμα 16: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 14.



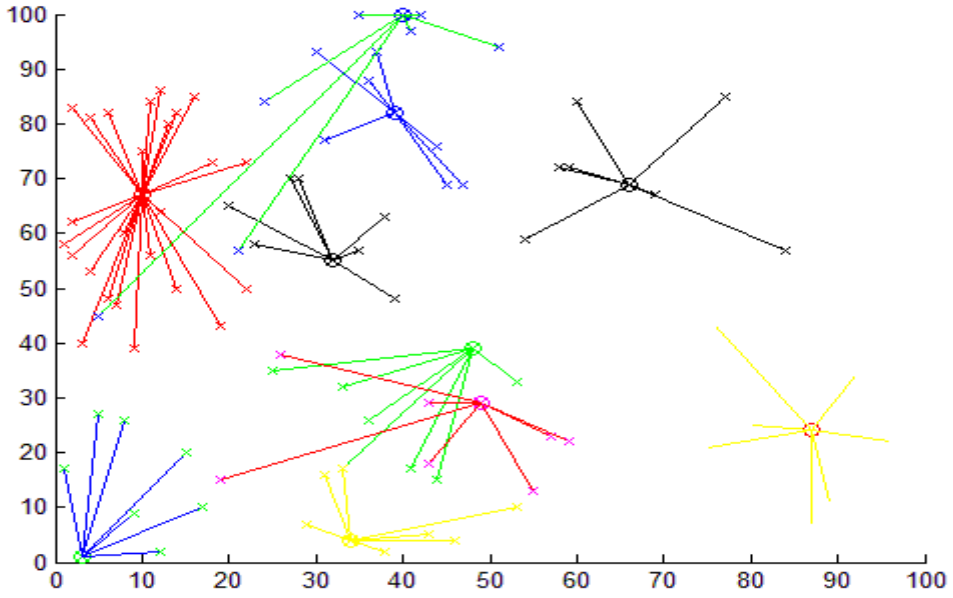
Σχήμα 17: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 15.



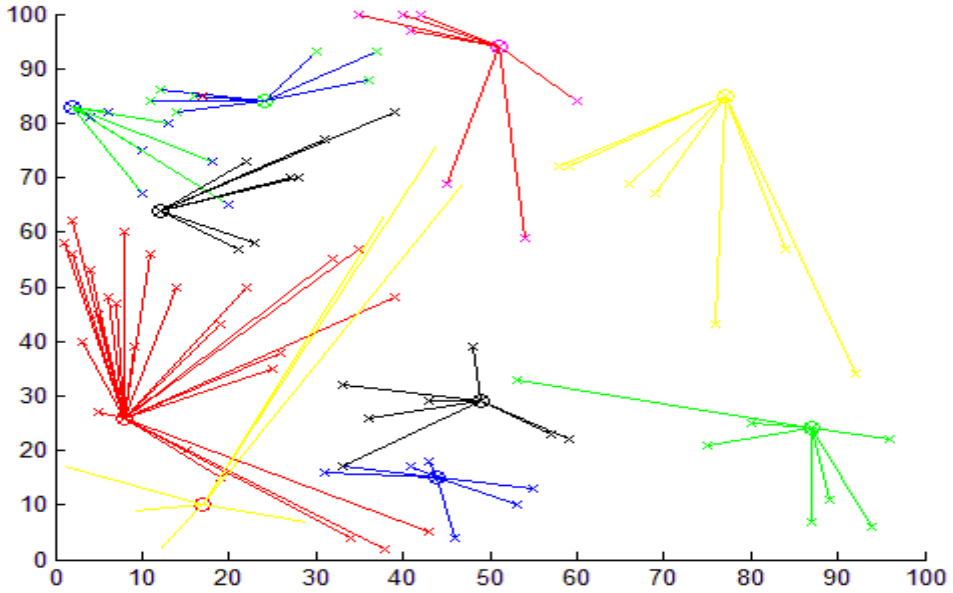
Σχήμα 18: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 16.



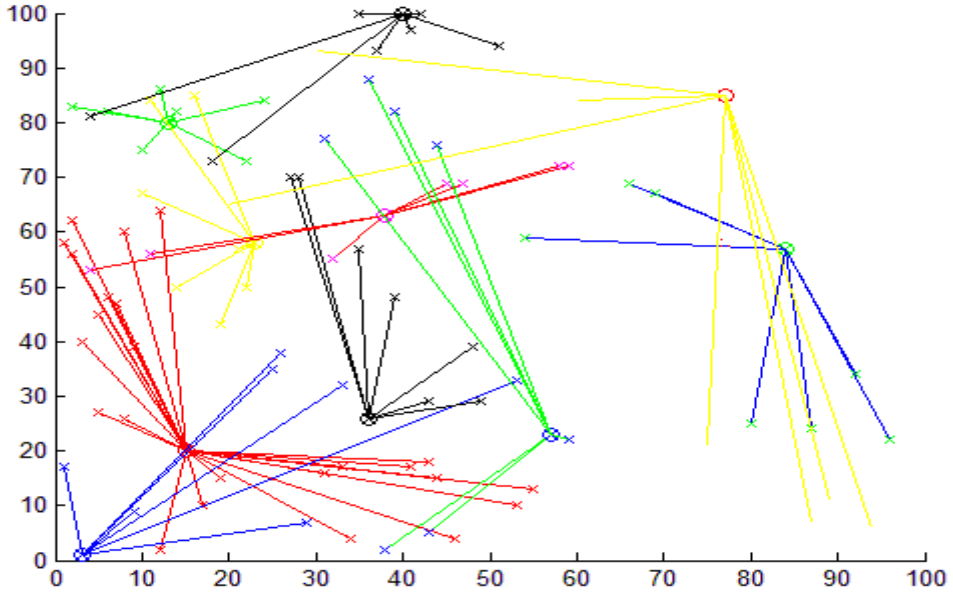
Σχήμα 19: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 17.



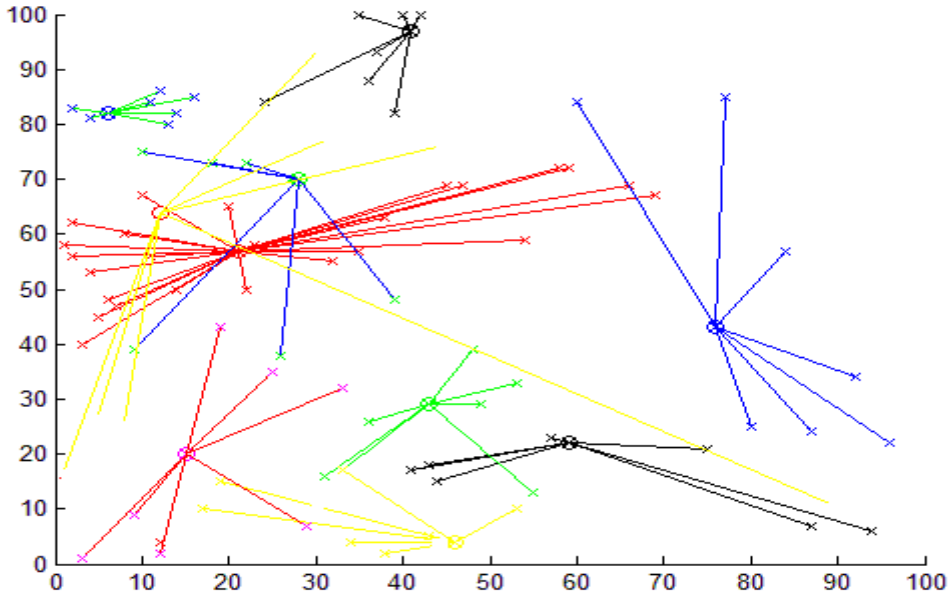
Σχήμα 20: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 18



Σχήμα 21: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 19.



Σχήμα 22: Λύση μετασχηματισμένου προβλήματος Pmedian – Test Problem 20.



Πίνακας 2: Λύσεις προβλήματος P-median και σύγκριση με τις βέλτιστες λύσεις.

Test Problem	Λύση	Βέλτιστη Λύση	Απόκλιση από βέλτιστο	Επαναλήψεις	Χρόνος
TP1	735	713	3.09%	7	0.7933
TP2	757	740	2.29%	9	0.7886
TP3	791	751	5.33%	41	1.4014
TP4	708	651	8.75%	212	3.2693
TP5	714	664	7.53%	129	2.5786
TP6	797	778	2.44%	3	1.0701
TP7	794	790	0.51%	17	0.9322
TP8	823	820	0.36%	11	0.8478
TP9	748	715	4.61%	7	0.7594
T10	838	829	1.09%	28	1.1864
TP11	1153	1006	14.6%	169	3.6075
TP12	1089	966	12.7%	105	3.2683
TP13	1143	1026	11.4%	112	4.1555
TP14	1097	982	11.7%	209	3.1174
TP15	1181	1091	8.24%	168	3.3364
TP16	1054	954	10.48%	195	3.2150
TP17	1187	1034	14.7%	254	3.4756
TP18	1100	1043	5.4%	186	3.5512
TP19	1096	1031	6.3%	204	4.2311
TP20	1161	1005	14.5%	263	4.7377

6. Εφαρμογή Αλγόριθμου ACO στο Πρόβλημα CFL

Στην ενότητα αυτή παρουσιάζουμε ένα αλγόριθμο ACO, ο οποίος υλοποιεί την παραλλαγή του βασικού CFLP κατά την οποία κάθε πελάτης μπορεί να εξυπηρετηθεί από μία μόνο εγκατάσταση (Nozick και Turnquist 2001). Πρόκειται για το Πρόβλημα Χωροθέτησης Εγκαταστάσεων Πεπερασμένης Χωρητικότητας από μία Πηγή (Single Source Capacitated Facility Location Problem – CFL Problem - SSCFLP) το οποίο είναι μια ειδική περίπτωση του CFLP.

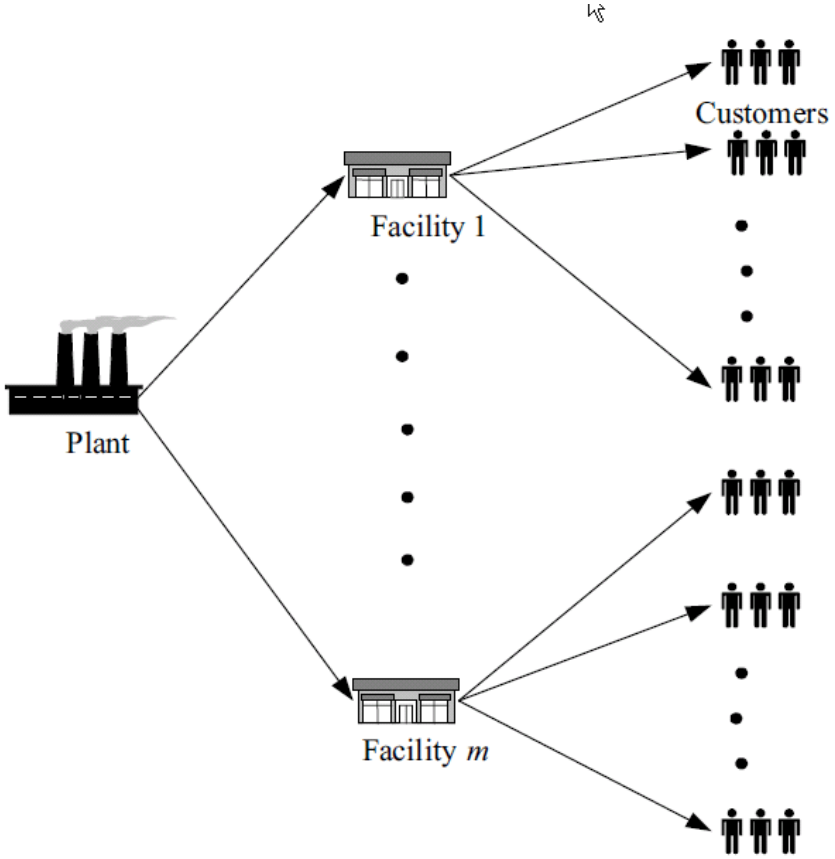
Το SSCFLP είναι πρόβλημα συνδυαστικής βελτιστοποίησης (combinatorial optimization problem) το οποίο ανήκει στην κατηγορία των NP-hard προβλημάτων. Για την επίλυσή του, έχουν προταθεί κατά καιρούς διάφορες μέθοδοι όπως, Lagrangian heuristics, Lagrangian relaxation, Subgradient optimization, Repeated matching κλπ. Στην παρούσα εργασία αναπτύσσεται ένας αλγόριθμος βασισμένος στον ACO. Ο αλγόριθμος που παρουσιάζεται στην ενότητα αυτή περιγράφεται από στο (Kumweang και Kawtummachai, 2005).

6.1. Ορισμός του Μοντέλου του Προβλήματος

Ο στόχος του παρόντος αλγορίθμου είναι, δεδομένων ενός συνόλου πιθανών εγκαταστάσεων, να αποφασιστεί ποιες εγκαταστάσεις θα πρέπει να λειτουργήσουν και ποιοι πελάτες θα εξυπηρετούνται από κάθε εγκατάσταση. Κάθε πελάτης θα πρέπει να εξυπηρετείται από μία μόνο εγκατάσταση. Και φυσικά η ζήτηση όλων των πελατών θα πρέπει να καλύπτεται. Το Σχήμα 23 περιγράφει το μοντέλο.

Για την υλοποίηση του εν λόγω αλγορίθμου, θα χρησιμοποιήσουμε τις παρακάτω μεταβλητές:

Σχήμα 23: Πρόβλημα χωροθέτησης εγκαταστάσεων πεπερασμένης χωρητικότητας από μία πηγή



Πηγή: (Kumweang, K. και Kawtummachai, R., 2005)

m ο αριθμός των πιθανών εγκαταστάσεων,

n ο αριθμός των πελατών,

dj η ζήτηση του πελάτη j (j = 1, 2, …, n),

fi το σταθερό κόστος της εγκατάστασης i (i = 1, 2, …, m),

ci η χωρητικότητα της εγκατάστασης i,

lij η απόσταση της εγκατάστασης i από τον πελάτη j,

u το κόστος ανά μονάδα απόστασης,

v η χωρητικότητα του φορτηγού,

p η τιμή μονάδας,

α ποινή που επιβάλλεται όταν ένας πελάτης δε μπορεί να εξυπηρετηθεί,

yi = 1 αν μια εγκατάσταση i είναι σε λειτουργία, 0 διαφορετικά,

xij = 1 αν μια εγκατάσταση i εξυπηρετεί τον πελάτη j, 0 διαφορετικά,

qij = 1 αν η εγκατάσταση i δε μπορεί να καλύψει τη ζήτηση του πελάτη j, 0 διαφορετικά,

xij′ = xij, αν xij ≠ 0, 1 διαφορετικά (i = 1, 2, …, m). Η μεταβλητή αυτή δηλώνει ότι μια εγκατάσταση i δε μπορεί να εξυπηρετήσει τον πελάτη j μετά από την αντιστοίχηση όλων των εγκαταστάσεων σε πελάτες.

Γενικά, η αντικειμενική συνάρτηση ενός SSCFLP προσπαθεί να ελαχιστοποιήσει τόσο τα σταθερά όσο και το μεταβλητά κόστη. Τα σταθερά κόστη είναι τα κόστη ίδρυσης των εγκαταστάσεων και τα μεταβλητά κόστη είναι τα κόστη λειτουργίας και μεταφοράς από μια εγκατάσταση που λειτουργεί προς ένα πελάτη που εξυπηρετεί. Ένα μαθηματικό μοντέλο που στοχεύει στη μεγιστοποίηση της κάλυψης προσπαθεί να διασφαλίσει την κάλυψη της ζήτησης που βρίσκεται εντός μιας περιοχής.

Ο στόχος μεγιστοποίησης της κάλυψης της αναλογίας της συνολικής ζήτησης με τη συνδρομή N εγκαταστάσεων, περιγράφηκε αρχικά από τους Church και ReVelle (1974). Ο Hillsman (1984) πρότεινε ένα ισοδύναμο μοντέλο, το οποίο ελαχιστοποιεί τη μη καλυπτόμενη ζήτηση. Οι Nozick και Turnquist (2001), δεν έλαβαν υπόψη τη χωρητικότητα των εγκαταστάσεων, η οποία στην παρούσα εργασία έχει ενσωματωθεί. Ένα μαθηματικό μοντέλο για το SSCFLP που λαμβάνει υπόψη και το κόστος της μη καλυπτόμενης ζήτησης μπορεί να διατυπωθεί ως ακολούθως:

Ελαχιστοποίησε το $\sum_{i=1}^{m} f_i y_i + \sum_{i=1}^{m}\sum_{j=1}^{n} u\left(l_{ij}\left[\frac{d_j}{v}\right] x_{ij} \right) + \sum_{i=1}^{m}\sum_{j=1}^{n} \alpha p\left(q_{ij} d_j x'_{ij}\right)$ (30)

με τις παρακάτω συνθήκες

$\sum_{i=1}^{m} x_{ij} = 1;\ j = 1,2,...,n$ (31)

$\sum_{i=1}^{m} d_j x_{ij} \le c_i y_i ; i = 1,2,...,m$ (32)

$x_{ij} \le y_i$, i = 1, 2, …, m, j = 1, 2, …, n (33)

$y_i \in (0,1)$, i = 1, 2, …, m (34)

$x_{ij} \in (0,1)$, i = 1, 2, …, m, j = 1, 2, …, n (35)

$q_{ij} \in (0,1)$, i = 1, 2, …, m, j = 1, 2, …, n (36)

$x'_{ij} \in (0,1)$, i = 1, 2, …, m, j = 1, 2, …, n (37)

Η αντικειμενική συνάρτηση (30) λαμβάνει υπόψη το κόστος ίδρυσης των εγκαταστάσεων, το κόστος εξυπηρέτησης πελατών από εγκαταστάσεις που λειτουργούν και το κόστος της μη καλυπτόμενης ζήτησης. Ο περιορισμός (31) είναι περιορισμός ζήτησης και διασφαλίζει ότι κάθε πελάτης εξυπηρετείται από ακριβώς μία εγκατάσταση. Ο περιορισμός (32) είναι περιορισμός χωρητικότητας και διασφαλίζει ότι η ζήτηση του πελάτη που εξυπηρετείται από μια συγκεκριμένη εγκατάσταση δεν υπερβαίνει την χωρητικότητά της. Ο περιορισμός (34) διασφαλίζει ότι πελάτες εξυπηρετούνται μόνο από εγκαταστάσεις που λειτουργούν. Τέλος, οι περιορισμοί (34)-(37) είναι «περιορισμοί ακεραιότητας» (integrity constraints).

6.2. Επιμέρους Βήματα Αλγορίθμου

Τα βασικά βήματα του αλγορίθμου παρουσιάζονται παρακάτω:

<u>Βήμα 1:</u> Αρχικοποίησε το t = 0, τον Αριθμό Τρέχουσας Επανάληψης (AE = 0), τον Αριθμό Επαναλήψεων (AEmax) και το σύνολο δυνητικών αναθέσεων X = {xij}, για i = 1, 2, …, m, και j = 1, 2, …, n.

<u>Βήμα 2:</u> Υπολόγισε τα ίχνη φερομόνης, τ_{ij} για κάθε i, j: $\tau_{ij} = \dfrac{1}{f_i + \left(ul_{ij} \left[\dfrac{d_j}{v} \right] \right) + \left(\alpha p d_j \right)}$.

<u>Βήμα 3:</u> Ανάθεσε κατά προτεραιότητα σε πελάτες την εγκατάσταση με τη μεγαλύτερη χωρητικότητα. Για την εκτέλεση της ανάθεσης, δημιούργησε για κάθε i το Σύνολο Πιθανών Αναθέσεων Fi, το Σύνολο των Ήδη Ανατιθέμενων (already assigned) Si, το Σύνολο των μη Εφικτών (infeasible) Αναθέσεων Ui και το Σύνολο των Ανατιθέμενων (being assigned) Li.

<u>Βήμα 4:</u> Επέλεξε τυχαία μια τιμή για τη μεταβλητή $\alpha \in [0,1]$.

<u>Βήμα 4α:</u> Αν η τυχαία πιθανότητα α ανήκει στο εύρος των βέλτιστων επιλογών (α ≤ P0), τότε η ο πελάτης j ανατίθεται στην εγκατάσταση i με μέγιστη τιμή τij και το j προστίθεται στο Li.

<u>Βήμα 4β:</u>Αν η τυχαία πιθανότητα α ανήκει στο εύρος των μη βέλτιστων επιλογών (α > P0), τότε ο πελάτης j που επιλέγεται να ανατεθεί στην εγκατάσταση i, προκύπτει με βάση την ακόλουθη πιθανότητα.

$$P_{ij} = \begin{cases} \dfrac{\tau_{ij}}{\sum_{l \in K_{ij}} \tau_{il}} & \alpha\nu \ j \in K_{ij} \\ 0 & \delta\iota\alpha\varphi o\rho\varepsilon\tau\iota\kappa\acute{\alpha} \end{cases}$$

Για κάθε υπό εξέταση εγκατάσταση i, επέλεξε μια τυχαία πιθανότητα β για να επιλέξεις τον πελάτη j. Πρόσθεσε το j στο Li.

<u>Βήμα 5:</u> Έλεγξε τον περιορισμό της χωρητικότητας λαμβάνοντας υπόψη τη συνολική ζήτηση που ήδη εξυπηρετείται (being assigned customers) από την εγκατάσταση i.

<u>Βήμα 5α:</u> Αν η συνολική ζήτηση των εξυπηρετούμενων πελατών ικανοποιεί τους περιορισμούς χωρητικότητας, ανάθεσε την εξυπηρέτηση του πελάτη j στην εγκατάσταση i (πρόσθεσε το j στο Si) και ανανέωσε τα ίχνη φερομόνης $\tau_{ij}^{new} = \rho\tau_{ij}^{old} + \Delta\tau_{ij}$,όπου ρ είναι δείκτης «χρονικής παραμονής» του ίχνους (0 ≤ ρ < 1), και 1 – ρ είναι ο ρυθμός εξάτμισης. Η ποσότητα φερομόνης που εναποτίθεται από την τρέχουσα, τοπικά βέλτιστη λύση είναι $\Delta\tau_{ij} = \tau_{ij}(\max) \times Q$, όπου Q = 0.05 στην περίπτωση εφικτής (feasible) λύσης.

<u>Βήμα 5β:</u> Αν η συνολική ζήτηση των εξυπηρετούμενων πελατών υπερβαίνει τους περιορισμούς χωρητικότητας, συμπεριέλαβε τον πελάτη j στους μη εξυπηρετούμενους από την εγκατάσταση i (πρόσθεσε το j στο Ui) και ανανέωσε τα ίχνη φερομόνης $\tau_{ij}^{new} = \rho\tau_{ij}^{old} + \Delta\tau_{ij}$. Η ποσότητα φερομόνης που εναποτίθεται από την τρέχουσα, τοπικά βέλτιστη λύση είναι $\Delta\tau_{ij} = \tau_{ij}(\max) \times Q$, όπου Q = 0.01 στην περίπτωση μη εφικτής λύσης.

<u>Βήμα 6:</u> Έλεγξε αν όλοι οι πελάτες που έχουν εξεταστεί ανατέθηκαν προς εξυπηρέτηση.

<u>Βήμα 6α:</u> Εάν ο πελάτης j δεν έχει ανατεθεί, διέγραψε τον από το Fi, αύξησε το t κατά 1 και επέστρεψε στο Βήμα 4. Διέγραψε τον εξυπηρετούμενο πελάτη j, και επανέλαβε τα βήματα έως ότου εξεταστούν όλοι οι πελάτες.

<u>Βήμα 6β:</u> Εάν ο πελάτης j έχει ανατεθεί, έλεγξε αν έχεις εξετάσει όλες τις εγκαταστάσεις.

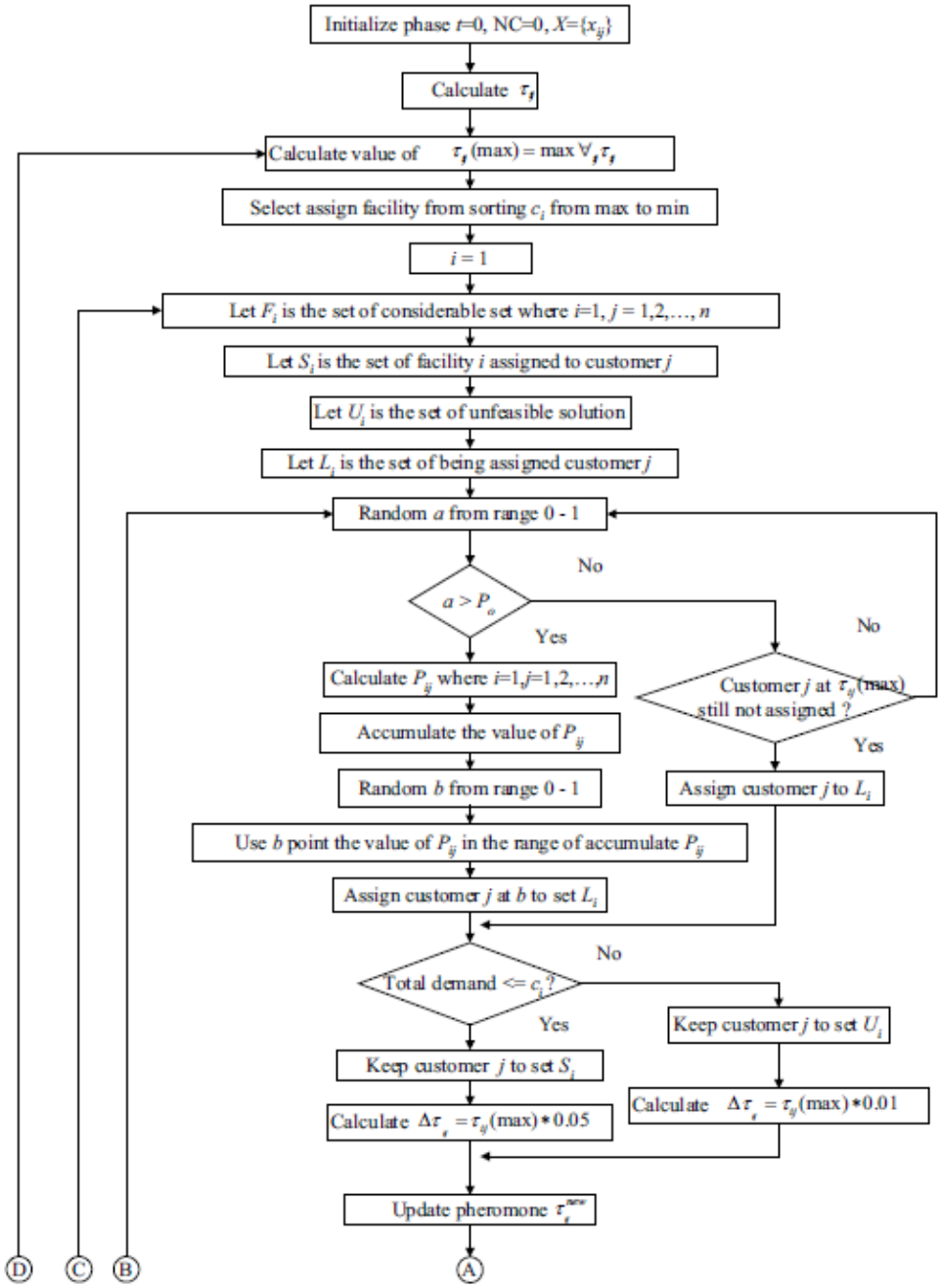
<u>Βήμα 6β(i):</u> Αν δεν έχουν εξεταστεί όλες οι εγκαταστάσεις, επανέλαβε τη διαδικασία για την επόμενη επιστρέφοντας στο Βήμα 3.

<u>Βήμα 6β(ii):</u> Αν έχουν εξεταστεί όλες οι εγκαταστάσεις, υπολόγισε το συνολικό κόστος Z χρησιμοποιώντας τη σχέση: $Z = \sum_{i=1}^{m} f_i y_i + \sum_{i=1}^{m}\sum_{j=1}^{n} u\left(l_{ij}\left[\frac{d_j}{v}\right] x_{ij} \right) + \sum_{i=1}^{m}\sum_{j=1}^{n} \alpha p\left(q_{ij} d_j x'_{ij}\right)$.

Κράτησε την μέχρι τότε ελάχιστη τιμή του Z και ανανέωσε τη φερομόνη $\tau_{ij} = \tau_{ij}^{new}$.

Αύξησε το AE κατά ένα και επανέλαβε τα Βήματα 2 έως 6 έως ότου το AE φτάσει την τιμή AEmax.

Ακολουθεί αναλυτικό διάγραμμα του αλγορίθμου



6.3. Υλοποίηση Αλγόριθμου – Αποτελέσματα Εκτέλεσης

Η διεξαγωγή των πειραμάτων έγινε με τη χρήση του Matlab. Το MATLAB είναι ένα πρόγραμμα υπολογιστών για ανθρώπους που χρησιμοποιούν αριθμητικούς υπολογισμούς, ειδικά στη γραμμική άλγεβρα (πίνακες). Ξεκίνησε ως ένα πρόγραμμα "Εργαστηρίου Πινάκων" ("MATrix LABoratory") που είχε σκοπό να παρέχει αλληλεπιδρώσα προσπέλαση στις βιβλιοθήκες Linpack και Eispack. Από τότε έχει αναπτυχθεί αρκετά, για να γίνει ένα ισχυρότατο εργαλείο στην οπτικοποίηση, στον προγραμματισμό, στην έρευνα, στην επιστήμη των μηχανικών, και στις επικοινωνίες.

Στο δυναμικό του Matlab συμπεριλαμβάνονται μοντέρνοι αλγόριθμοι, δυνατότητες χειρισμού τεράστιων ποσοτήτων δεδομένων, και ισχυρά προγραμματιστικά εργαλεία. Το Matlab δεν είναι σχεδιασμένο για συμβολικούς υπολογισμούς, αλλά αντισταθμίζει αυτή την αδυναμία του επιτρέποντας στο χρήστη να συνδέεται άμεσα με το Maple. Η επιφάνεια αλληλεπίδρασης βασίζεται κυρίως σε κείμενο, γεγονός που μπορεί να συγχύσει μερικούς χρήστες.

Οι παράμετροι που χρησιμοποιήθηκαν παρουσιάζονται στον παρακάτω Πίνακα 1:

Πίνακας 3: Παράμετροι αλγορίθμου ACO για το πρόβλημα CFL.

Σύμβολο	Ερμηνεία	Τιμές
m	Αριθμός δυνητικών εγκαταστάσεων	10,25,50
n	Αριθμός πελατών	25,50,100,500
u	Κόστος ανά μονάδα απόστασης	10 Ευρώ ανά km
p	Τιμή μονάδας	10 Ευρώ
v	Χωρητικότητα φορτηγού	1000 μονάδες ανά φορτηγό
A	Ποινή μη εξυπηρέτησης εγκατάστασης	1,5
ρ	Παραμονή ίχνους	0,9
AEmax	Αριθμός επαναλήψεων	100
P0	Πιθανότητα βέλτιστης επιλογής	0,1
ci	Χωρητικότητα εγκατάστασης i	Ομοιόμορφα κατανεμημένο στο

		[10,160]
fi	Σταθερό κόστος εγκατάστασης i	U[0,90] + U[100,110] * Sqrt(ci), όπου U[a,b] ομοιόμορφα κατανεμημένο στο [a,b]
dj	Ζήτηση πελάτη j	Ομοιόμορφα κατανεμημένο στο [5,35]
lij	Απόσταση εγκατάστασης i από τον πελάτη j	Προκύπτει με βάση τις ευκλείδειες συντεταγμένες

Ακολουθεί ο Πίνακας 4 με τις λύσεις που προκύπτουν μετά την εκτέλεση του παραπάνω αλγορίθμου, καθώς και σύγκρισή τους με τις βέλτιστες λύσεις. Είναι εμφανές ότι ο προτεινόμενος αλγόριθμος δίνει λύσεις συγκρίσιμες και ελαφρώς μεγαλύτερες από τις βέλτιστες για τα test problems που χρησιμοποιήθηκαν. Επίσης ακολοθούν αναλυτικά οι αντίστοιχες αναθέσεις πελατών σε εγκαταστάσεις για το κάθε πρόβλημα ξεχωριστά.

Πίνακας 4: Λύσεις προβλήματος CFL και σύγκριση με τις βέλτιστες λύσεις (a=1.5).

Test Problem	Λύση	Βέλτιστη Λύση	Απόκλιση από βέλτιστο	Χρόνος	Επανάληψη
Cap41	1057900	1040444.375	1.67%	0.1041	4
Cap42	1107800	1098000.450	0.89%	0.1015	2
Cap43	1159900	1153000.450	0.59%	0.1405	8
Cap44	1237200	1235500.450	0.14%	0.1191	5
Cap51	1124300	1025208.225	8.39%	0.0988	42
Cap61	1002310	932615.750	7.47%	0.1029	91
Cap62	1000045	977799.400	2.27%	0.0899	139
Cap63	1034500	1014062.050	2.02%	0.1034	77
Cap64	1091500	1045650.250	4.38%	0.1046	112
Cap71	1000131	932615.750	7.24%	0.0908	161

Cap72	1060655	977799.400	8.47%	0.0993	228
Cap73	1034500	1010641.450	2.36%	0.1121	209
Cap74	1071100	1034976.975	3.49%	0.0956	256
Cap81	888580	838499.288	5.97%	0.1241	66
Cap82	1000560	910889.563	9.18%	0.1137	47
Cap83	1021100	975889.563	4.63%	0.1320	83
Cap84	1069400	1069369.525	0.02%	0.1158	4
Cap91	861100	796648.438	8.09%	0.1234	89
Cap92	890610	855733.500	4.08%	0.1671	68
Cap93	937000	896617.538	4.51%	0.2456	145
Cap94	1020101	946051.325	7.83%	0.1694	132
Cap101	889000	796648.437	11.59%	0.1564	161
Cap102	892300	854704.200	4.39%	0.2654	96
Cap103	980011	893782.112	9.64%	0.2118	27
Cap104	1008320	928941.750	8.54%	0.2211	132
Cap111	842300	826124.713	1.96%	0.2804	50
Cap112	934500	901377.213	3.67%	0.2285	97
Cap113	1003100	970567.750	3.35%	0.1786	147
Cap114	1106700	1063356.488	4.08%	0.2091	39
Cap121	860130	793439.563	8.41%	0.2010	87
Cap122	880200	852524.625	3.25%	0.1922	154
Cap123	1006100	895302.325	8.37%	0.1654	235
Cap124	1020010	946051.325	5.81%	0.1018	221
Cap131	855600	793439.562	7.83%	0.2944	139
Cap132	901700	851495.325	5.89%	0.2408	55
Cap133	911700	893076.712	2.09%	0.1276	41

Cap134	991200	928941.750	6.70%	0.1071	79

Cap41								
Facility	1	2	3	6	7	8	9	12
Clients	13 14 15 28 29 30 31 32 33	4 5 6 7 9 19 20 21 24	12 25 26 27	16 17 18 22 23	1 2 3 8	39 45 46 47 48 49 50	35 36 37 38 45	10 40 41 42 43 44

Cap42									
Facility	2	7	8	9	12	13	14	15	16
Clients	10 43 44 45 46 47 48 49	1 2 4	14 15 16 17	13 18 23 24 30	3 6 7 8 9 12	5 19 20 21 22 39 40 41 42	35 36 37 38	32 33 50	25 26 27 28 29 30 31

Cap43								
Facility	2	3	5	6	11	12	13	15
Clients	40 41 44 46 47 48 49 50	3 4 5 6 7 9 14	24 27 30 31	12 13 15 16 17 18 19	8 20 32 33 35 36 37	10 38 39 42 43 45	1 2	21 22 23 25 26 28 29

Cap44								
Facility	4	5	7	8	10	14	15	16
Clients	12 13 20 21 22 23 24 25	1 2 3 4 5 6 8	31 32 33 35	29 30 45 47 48 49 50	7 9 14 15 16 17 18 19	26 36 37 38	27 28	10 39 40 41 42 43 44 46

Cap51						
Facility	2	5	7	10	15	
Clients	38 39 40 41 42 43 44 45 49 50	25 26 27 28 35 36 37 46 47 48	8 9 10 11 14 15 16 17 18 19 20 21 22 23 24	29 30 31 32 33	1 2 3 4 5 6 7 12 13	

Cap61					
Facility	5	10	13	16	
Clients	1 2 3 4 6 7 8 12 13	32 33 34 35 36 37	5 11 15 16 17 18 19 20 21 22 23 24 25 38 39 40	9 10 14 26 27 28 29 30 31 41 42 43 44 45 46 47 48 49 50	

Cap62					
Facility	1	3	6	13	
Clients	10 11 35 36 37 38 39 40 41 42	3 33 34 43 44 45 46 47 48 49 50	1 2 4 6 7 8 9 13 14	5 12 15 16 17 18 19 20 21 22 23 24 26 27 28 29 30 31 32	

Cap63					
Facility	1	2	15	16	
Clients	1 2 3 4 6 7 8 9 13 14 15 16 17	11 12 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 39	10 27 33 34 50	35 36 37 38 40 41 42 43 44 45 46 47 48 49	

Cap64						
Facility	6	8	9	10	16	
Clients	18 19 33 34	10 11 13 14 15 16 17	20 21 22 23 24 25 26 27 28 29 30 31 32 36	35 37 38 39 40 41 42 43 44 45 46 47 48 49 50	1 2 3 4 5 6 7 8 9 12	

Cap71							
Facility	4	6	7	13	15	16	
Clients	9 10 11 12 13 14 16	15 18 19 20 21 22 23 24 25 26 28	30 31 32 33 34 36	17 27 35 37 38	41 42 43 44	39 40 45 46 47 48 49 50	

Cap72							
Facility	1	2	8	11	12	14	
Clients	13 14 15 16 17 18 20	1 2 3 4 5 8 9 11 12 46	6 7 10 13 14 16	15 19 21 22 47 48 49 50	31 32 33 34 35 37 38 39 40 42	23 24 25 27 28 29 30 41 43 44 45	

Cap73								
Facility	4	5	7	10	11	14	15	
Clients	11 12 13 14 15 17 21 22	1 2 3 4 5 6 7 9 16 18	19 23 24 25 26 27 28	8 10 20 29 30 33	35 36 37 38 39 40 41 42 43	34 44 45 46	31 32 47 48 49 50	

Cap74								
Facility	1	2	5	6	9	12	13	
Clients	1 2 5 8 9 11	3 4 6 7 12 13 15 16	10 14 17 18	25 26 27 28 29 31 33	19 20 21 22 23 24 32	43 44 45 46 47	30 48 49 50	

Cap81								
Facility	3	4	5	6	18	21	22	25
Clients	10 42 43 44 45 46 50	22 23 24 40 41 48 49	37 38	6 7 12 13 14 15 16	1 2 3 8 9	32 33 47	5 17 18 19 20 21 28	4 25 26 27 29 30 31 35 36 39

Cap82								
Facility	2	8	9	10	13	15	16	21
Clients	1 2 3 4 8 9	15 23 28 44 45 46 47	29 30 31 32 33	16 17 18 19 20 21 22 24 25	5 6 7 12 13 14	26 27 39	35 36 37 38	10 40 41 42 43 48 49 50

Cap83									
Facility	1	7	8	11	15	20	22	23	24
Clients	2 3 4 6 7 9	12 15 16 17 18	38 39	25 26 27 35 36	8 10 40 41 42 43 44	1 4	24 32 33	37 45 46 47 48 49 50	5 13 14 19 20 21 22 23 28 29 30 31

Cap84								
Facility	1	3	4	5	7	10	12	15
Clients	3 4 5 7 14 15 16 20 21 23 24	31 41 42 43 44 45	35 36 39 40	1 2 6 8 9	37 38	10 32 33 46 47 48 49 50	12 25 26 27 28 29 30	13 17 18 19 22

Cap91					
Facility	13	16	17	25	
Clients	5 31 32 35 36 37 38 39 40 41 44 45 46 47 48 49	11 15 16 17 18 25 26 27 28 29 30	10 33 34 42 43	1 2 3 4 6 7 8 9 12 13 14 19 20 21 22 23 24	

Cap92					
Facility	5	10	12	22	
Clients	1 2 3 4 6 7 8 9 14 15 16 17 18	32 33 34 41 42 43	10 11 12 13 19 20 21 22 23 24 25 35 36 37	5 26 27 28 29 30 31 38 39 40 44 45 46 47 48 49 50	

Cap93					
Facility	1	5	13	17	
Clients	1 2 3 4	6 7 8 9 10 11 12 14 15 16 17 18 19 20 21 28 29 30	5 13 22 35 36 37 38 39 44 45 46 47 48 49 50	23 24 25 26 27 31 32 33 34 40 41 42 43	

Cap94					
Facility	1	11	20	22	
Clients	11 13 14 15 16 17 18 19 20 21 22 23 24 25 26 35	10 28 29 30 31 32 33 34 36 37	1 2 3 4 6 7 9 12	5 8 27 38 39 40 41 42 43 44 45 46 47 48 49 50	

Cap101							
Facility	2	8	13	14	15	16	
Clients	3 4 5 7 14 15 16 20 21 23 24	31 41 42 43 44 45	13 17 18 19 22 35 36 39 40	1 2 6 8 9	12 25 26 27 28 29 30 37 38	10 32 33 46 47 48 49 50	

Cap102							
Facility	5	6	9	13	16	23	
Clients	5 6 7 8 9 10 13	31 32 33 34 35 39	14 17 21 22 23 24 25 26	11 12 40 41	15 16 42 43 44 45 46	27 28 29 30 47 48 49 50	

Cap103							
Facility	4	5	7	10	19	20	
Clients	9 10 11 13 14 15 16	12 21 22 23 24 27 28 29 30	17 18 19 20 31 32 33	40 41 42	34 35 36 37 39	38 43 44 45 46 47 48 49 50	

Cap104							
Facility	1	3	6	17	22	23	
Clients	2 3 4 5 7 8 13	21 22 23 24 26 31	1 14 15 16 17 32 33 34 35	6 9 10 11 12 36 42 44	18 19 20 41	37 38 39 40 43 45 46 47 48 49 50	

Cap111									
Facility	4	8	10	16	21	24	34	48	
Clients	2 3 5 6 8 9	21 22 23 24 25 26 27 35 36	15 16 17 18 20 28	1 4 7 12 13 14	19 29	10 30 31 32 33 37	38 39 40 41 49 50	42 43 44 45 46 47 48	

Cap112										
Facility	2	10	14	20	30	31	34	45	46	
Clients	29 30 32 45 48 49 50	15 16 17 18 21 23 28	9 10	22 31 39 40 41 42 43	33 46 47	24 25 26 27	1 2 3 4 8	35 36 37 38 44	5 6 7 12 13 14 19 20	

Cap113										
Facility	1	18	22	25	27	39	45	48	50	
Clients	10 24 25 26 27	15 18 19 20 21 22 35 36	28 29 30 31 32 33	6 7 8 9 12 13	37 40 41 50	5 42 43 44 45 46 47	16 17	1 2 3 4	14 23 38 39 48 49	

Cap114									
Facility	5	24	28	29	33	34	43	50	
Clients	6 7 8 9 13 14 15 16 19 20 21	12 17 18 22 23 24 25 26	1 2 3 4	10 35 36 45 46 47 48 49	29 38 39 42 43 44	33 37	5 28 30 31 32	27 40 41 50	

Cap121					
Facility	19	25	46	49	
Clients	18 30 31 32 33 34 43	11 15 16 17 19 20 21 22 23 24 25 26 27 28 29	1 2 3 4 5 6 7 8 9 12 13	10 14 35 36 37 38 39 40 41 42 44 45 46 47 48 49 50	

Cap122					
Facility	36	37	39	40	
Clients	11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 30 31	10 27 28 35 36 37 38 39 43 44 45 50	5 29 32 33 34 40 41 42 46 47 48 49	1 2 3 4 6 7 8 9	

Cap123				
Facility	24	29	41	
Clients	11 17 18 23 26 27 28 29 32 33	1 2 3 4 6 7 8 9 12 13 14 15 16 19 20 21 22	10 24 25 30 31 34 43	

Cap124					
Facility	20	30	31	44	
Clients	10 27 28 35 36 37 38 39 40 41 42	1 2 3 4 5 6 7 8 9 13 17 19	33 34 43 44 45 46 47 48 49 50	11 12 14 15 16 18 20 21 22 23 24 25 26 29 30 31 32	

Cap131					
Facility	29	34	41	42	
Clients	6 7 8 9 10 11 12 14 15	1 2 3 4 16 17 18 19 20 21 28 29 30	5 23 24 25 26 27 31 32 33 34 40 44 45 46 47 48 49 50	13 22 35 36 37 38 39 41 42 43	

Cap132					
Facility	23	26	44	49	
Clients	23 24 25 26 27 31	1 2 3 4 5 13 20 21 28 29 30	22 35 36 37 38 39 44 45 46 47 48 49 50	6 7 8 9 10 11 12 14 15 16 17 18 19 32 33 34 40 41 42 43	

Cap133						
Facility	19	35	42	47	49	
Clients	8 9 10 11 12 15 16	27 28 29 30 31 33 36 37 43 44 49 50	1 2 13 14 17 19 20 21 22	3 4 5 6 7 8 18 23 24 25 26	38 39 40 41 42	

Cap134						
Facility	22	31	33	39	44	
Clients	39 40 41 42 47 48	6 7 8 9 11 12 13 19 21	10 14 15 16 17 18 43 44 45 46 37 50	1 2 3 4 5 19 20 22 23 24 25 26 27 28 29 30	31 32 33 34 35 36 38 49	

7. Συμπεράσματα

Χωροθετικές αποφάσεις (location decisions) εμφανίζονται σε μια ποικιλία προβλημάτων του ιδιωτικού και του δημόσιου τομέα. Για να παρθούν αυτές οι αποφάσεις θα πρέπει να προηγηθεί η Ανάλυση Χωροθέτησης (Location Analysis), δηλαδή η διαδικασία κατά την οποία σε κάποιο περιβάλλον, εγκαθιστούμε κέντρα παροχής υπηρεσιών (εξυπηρέτησης) έτσι ώστε να καλύψουμε τις ανάγκες των χρηστών των κέντρων αυτών (ζήτηση) κατά τον «καλύτερο δυνατό τρόπο». Όπου ο «καλύτερος δυνατός τρόπος» επιτυγχάνεται βελτιστοποιώντας μία αντικειμενική συνάρτηση η οποία περιγράφει ακριβέστερα τους στόχους του προβλήματος.

Στην παρούσα εργασία παρουσιάστηκαν δύο μεθευρετικοί αλγόριθμοι συνδυαστικής βελτιστοποίησης οι οποίοι επιλύουν δύο σημαντικά προβλήματα με ευρύτατη εφαρμογή στα logistics. Πρόκειται για την εύρεση των p-μέσων (p-median) και τη χωροθέτηση εγκαταστάσεων πεπερασμένης χωρητικότητας (capacitated facility location problem). Βάση των προτεινόμενων υλοποιήσεων αποτέλεσε ο αλγόριθμος βελτιστοποίησης αποικίας μυρμηγκιών (ant colony optimization).

Οι ACO είναι μια πιθανολογική τεχνική για την επίλυση υπολογιστικών προβλημάτων που μπορούν να περιοριστούν στην εύρεση των καλών «μονοπατιών – λύσεων» με τη χρήση γράφων. Είναι αλγόριθμοι βελτιστοποίησης που εμπνέονται από τη συμπεριφορά των μυρμηγκιών κατά την αναζήτηση και συγκομιδή τροφής. Το συγκεκριμένο ερευνητικό πεδίο αυτή τη στιγμή βρίσκεται υπό συνεχή ανάπτυξη και με πολλές εφαρμογές τόσο σε διακριτά όσο και σε συνεχή προβλήματα βελτιστοποίησης.

Ειδικότερα, για την επίλυση του προβλήματος p-median, υλοποιήθηκε ο αλγόριθμος που παρουσιάζεται στο (Zohrehbandian, M., και Hamidnia Namini, S., 2010) και ο οποίος εφαρμόζει μια σειρά τροποποιήσεων στο κλασσικό πρόβλημα δρομολόγησης οχημάτων, ώστε να κάνει εφικτή την αναζήτηση της βέλτιστης λύσης του προβλήματος p-median. Για την επίλυση του προβλήματος χωροθέτησης εγκαταστάσεων πεπερασμένης

χωρητικότητας, υλοποιήθηκε η παραλλαγή του (εξυπηρέτηση κάθε πελάτη από μία μόνο πηγή), όπως περιγράφεται στο (Kumweang και Kawtummachai, 2005).

Τα αποτελέσματα των προτεινόμενων αλγορίθμων, είναι αξιοσημείωτα και συγκρίσιμα με τα βέλτιστα μιας σειράς ενδεικτικών προβλημάτων (test problems) που χρησιμοποιήθηκαν ως ενδεικτικά (benchmarks) για να καταστήσουν εφικτή τη μέτρηση της αποτελεσματικότητας των προτεινόμενων αλγορίθμων. Αν και στη γενική περίπτωση οι λύσεις που παράγουν οι προτεινόμενοι αλγόριθμοι είναι υποδεέστεροι σε ποσοστά από 5% έως 15% περίπου (ανάλογα με το μέγεθος και τα επιμέρους χαρακτηριστικά του κάθε προβλήματος), είναι βέβαιο ότι αποτελούν αξιόλογες εναλλακτικές επιλογές για την επίλυση των εν λόγω προβλημάτων.

Οι ανάλυσεις των αποτελεσμάτων δείχνει ότι μπορούν να γίνουν μελλοντικές βελτιώσεις στους αλγορίθμους. Πιο συγκεκριμένα μπορούν να δοκιμαστούν τα εξής:
Πραγματοποίηση παράλληλης επεξεργασίας είτε αφιερώνοντας έναν επεξεργαστή σε κάθε μυρμήγκι είτε δημιουργώντας ανεξάρτητες αποικίες μυρμηγκιών (distributed colonies) οι οποίες ανταλλάσσουν δεδομένα ανά τακτά χρονικά διαστήματα.

Χρησιμοποίηση διαφορετικών μηχανισμών ομαλοποίησης της φερομόνης και επαναρχικοποίησης του αλγορίθμου. Για παράδειγμα, θα μπορούσαμε να κάνουμε ομαλοποίηση της φερομόνης ανά συγκεκριμένο αριθμό επαναλήψεων. Επίσης, φαίνεται καλύτερο μετά την επαναρχικοποίηση να γίνεται εναπόθεση φερομόνης κυρίως από το μυρμήγκι που έχει βρει τη βέλτιστη λύση μετά την επαναρχικοποίηση.

8. Βιβλιογραφία

Akino, U., Khumawala, B., 1977. An Efficient Branch and Bound Algorithm for the Capacitated Warehouse Location Problem. Management Science, Vol. 20.

Beasley, J., 1985. A note on solving large p-median problems. European Journal of Operational Research, 21, 270-273, North-Holland.

Beasley, J., 1988. An algorithm for solving large capacitated warehouse location problems. European Journal of Operational Research 33, 314-325, North-Holland

Bell J. and McMullen P., 2004. Ant colony optimization techniques for the vehicle routing problem. Advanced Engineering Informatics, 18, 41-48.

Bodin, L., Golden, B., Assad, A. and Ball, M., 1983. The State of the Art in the Routing and Scheduling of Vehicles and Crews. Computers and Operations Research 10 (2), 63-211.

Branco I. and Coelho J., 1990. The Hamiltonian p Median Problem. European Journal of Operational Research, 47, 86-95.

Clarke, G. and Wright, J., 1964. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. Operations Research 12, 568-581.

Colorni, A., Dorigo, M., Maniezzo, V., 1992. Distributed optimization by ant colonies. Proceedings of the First European Conference on Artificial Life, pp. 134–142. Cambridge, MA, MIT Press.

Colorni, A., Dorigo, M., Maniezzo, V., 1992. An investigation of some properties of an ant algorithm. Proceedings of PPSN-II, Second International Conference on Parallel Problem Solving from Nature, pp. 509–520, Amsterdam, Elsevier.

Cornuéjols, G., Sridharan, R., and Thizy, J, 1991. A Comparison of Heuristics and Relaxations for the Capacitated Plant Location Problem. European Journal of Operational Research, 50(3), pp. 280-297.

Dorigo, M., Stutzle, T., 2004. Ant Colony optimization. MIT Press.

Fathali, J., Kakhki, H., Burkard, R., 2006. An ant colony algorithm for the pos/neg weighted p-median problem. Central European Journal of Operations Research, Volume 14, Number 3, 229-246.

Hakimi, L., 1964. Optimum locations of switching centers and the absolute centers and medians of a graph. Operations Research, 12(3):450–459.

Hakimi, L., 1965. Optimum distribution of switching centers in a communication network and some related graph theoretic problems. Operations Research, 13(3):462–475.

Hakimi S.L., Mahesmary S.N. “Optimal Locations of Centers in Networks”.
Operations Research, Vol. 20, 967-973, 1972.

Levy J. “An Extended Theorem for Location on a Network”. Operational
Research, Vol.18, No 4, 433-442, 1967.

Koskosidis, I. and Powell, W., 1992, Clustering Algorithms for Consolidation of Customer Orders into Vehicle Shipments, Transportation Research, Vol. 26B.

Kumweang, K. and Kawtummachai, R., 2005. Solving a SSCFLP in a Supply Chain with ACO. Suranaree J. Sci. Technol. 12(1):28-38.

Mulvey, J. and Beck, M., 1984. Solving Capacitated Clustering Problems. European Journal of Operations Research, Vol. 18, pp. 339-348.

Nauss, M., 1978. An Improved Algorithm for the Capacitated Facility Location Problem. Journal of Operational Research Society. Vol.29, No.12.

Nozick, L. and Turnquist, M., 2001. Two-echelon Inventory Allocation and Distribution Center Location Analysis. Transportation Research Part E, Vol. 37, 2001, pp. 425-441.

Osman, I. and Christofides, N., 1994. Capacitated Clustering Problems by Hybrid Simulated Annealing and Tabu Search. Int. Trans. Opl. Res., Vol. 1. No. 3, pp. 317-336, Elsevier Science Ltd.

Pardalos, P., Romeijn, H. and Tuy, H., 2000, Recent developments and trends in global optimization. Vol. 124, pp. 209-228.

Reese, J., 2005. Methods for Solving the p-Median Problem: An Annotated Bibliography. Department of Mathematics, Trinity University.

Sherali, H. and Tuncbilek C., 1992. A Squared- Euclidean Distance Location-allocation Problem. Naval Research Logistics, Vol. 39, pp. 447 –469.

Singhtaun, C. and Charnsethikul, P., 2007. An Efficient Algorithms for Capacitated Multifacility Location Problems, Journal of Computer Science, Vol. 3, No. 8, pp. 583-591.

Talbi G., 2009. Metaheuristics: From Design To Implementation. Wiley.

Vlahos, A., 2006. Ant colony system solving capacitated location- allocation problems on a line. Journal of Information & Optimization Sciences, Vol. 27, No. 1, pp. 81-96, Taru Publications.

Yaghini, M., Lessan, J., Golami Mazinan, H., 2010. An Efficient Hybrid Metahuristic for Capacitated p-Median Problem. International Journal on Industrial Engineering and Production Research, Vol. 21, No. 1, pp.11-15.

Zohrehbandian, M., Hamidnia Namini, S., 2010. Ant colony optimization techniques for the Hamiltonian p-median problem. Mathematical Sciences Vol. 4: 383-390.

Wendell R.E., Hunter Jr.A.P. "Optimal Locations on a Network".
Transportation Science, Vol. 7, 18-33, 1973.

Scott A.J. "Location- Allocation Systems: A Review. Geographical Analysis", 2, 95-119, 1970

Owen S.H., Daskin M.S. "Strategic Facility Location: A review." European
Journal of Operations Research III, 423-447, 1998.

Σατρατζέμη Μαρία-Αικατερίνη. "Επίλυση Προβλημάτων Χωροθέτησης
Κέντρων Παροχής Υπηρεσιών". Διδακτορική Διατριβή, Θεσσαλονίκη, 1991.

Χατζηθωμά Ανδρούλα. «Η Σχέση της Ανάλυσης Χωροθέτησης με τους Αλγορίθμους Ομαδοποίησης». Διδακτορική Διατριβή, Πάτρα 2008

Handler G.Y., Mirchandani P.B. "Locations in Networks: Theory and
Algorithms". M.I.T. Press, 1979.

Neebe A. "A Procedure for Locating Emergency- Service Facilities for AllPossible Response Distances". Journal of the Operational Research Society, 39,743-748, 1988.

Kariv O., Hakimi S.L., "An Algorithm Approach to Network Location Problems II: The P-Medians". SIAM Journal Applied Mathematics, Vol.37, No 3, 539-560, 1979.

Shapiro J. F. "Mathematical Programming Structures and Algorithms". John Wiley & Sons, 1979.

Khumawala B. "Branch and Bound Algorithms for Locating Emergency
Service Facilities". Krannert Graduate School of Industrial Administration,Purdue University, No 355, 1972.

Toregas C., Swain R., Revelle C., Bergman L. “The Location of Emergency Service Facilities”. Operations Research, 19, 1363-1373, 1971.

Garfinkel R.S., Nemhauser G.L. “Integer Programming”. John Wiley & Sons, 1972.

Geofrion A.M. “Lagrangean Relaxation for Integer Programming”.
Mathematical Programming Study, Vol. 2, 82-114, 1974.

Held M., Wolfe P., Cravden H.P. “Validation of Subgradient Optimization”.
Mathematical Programming, Vol.6, 62-88, 1974.

Lawrence V. Snyder. «Supply Chain Robustness and Reliability: Models and Algorithms». Dissertation-Northwestern University, 2003

M. Dorigo, Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italie, 1992.

S. Goss, S. Aron, J.-L. Deneubourg et J.-M. Pasteels, The self-organized exploratory pattern of the Argentine ant, Naturwissenschaften, volume 76, pages 579-581, 1989

J.-L. Deneubourg, S. Aron, S. Goss et J.-M. Pasteels, The self-organizing exploratory pattern of the Argentine ant, Journal of Insect Behavior, volume 3, page 159, 1990

E. Bonabeau, M. Dorigo, G. Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, Oxford, 1999.

M. Dorigo, V. Maniezzo, A. Colorni, Ant System: An Autocatalytic Optimizing Process, Technical Report 91-016, Dip. di Elettronica, Politecnico di Milano, Italy, 1991

M. Zohrehbandian1, S. Hamidnia namini, “Ant colony optimization techniques for the Hamiltonian p-median problem”, Mathematical Sciences Vol. 4, No. 4 (2010) 383-390

Ρογδάκης Ιωάννης, «Ένα Σύστημα Υποστήριξης Αποφάσεων για την επίλυση προβλημάτων Εφοδιαστικής Αλυσίδας με την χρήση Εξελικτικών Αλγορίθμων», Μεταπτυχιακή Διατριβή, ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ, 2000.

Nozick, L.K., and Turnquist, M.A. (2001). Inventory, transportation, service quality and the location of distribution centers. European Journal of Operation Research, 129:362-371.

http://mscmga.ms.ic.ac.uk/