



**ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ**  
**ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ**

Μιμητικός Αλγόριθμος για το πρόβλημα δρομολόγησης και  
χρονικού προγραμματισμού φορτηγών πλοίων βιομηχανικού  
φορτίου και ελεύθερων φορτηγών πλοίων ξηρού φορτίου.

Διπλωματική Εργασία  
Γιώργος Καλιβούρης

Επιβλέπων Καθηγητής : Δρ. Ιωάννης Μαρινάκης

Χανιά, 2021

## **Ευχαριστίες**

Αρχικά θα ήθελα να ευχαριστήσω τον καθηγητή μου, Δρ. *Ιωάννη Μαρινάκη* για την εμπιστοσύνη που μου έδειξε και τη σημαντική καθοδήγηση του σε κομβικά σημεία της εργασίας. Επιπλέον, να ευχαριστήσω ιδιαίτερω τον *υποψήφιο διδάκτωρ Ελευθέριο Τσακιράκη* για την πολύτιμη βοήθεια που μου έδωσε και τη καλή συνεργασία που είχαμε, σε κάθε στάδιο αυτής της εργασίας . Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου και τους φίλους μου για τη συμπαράσταση που μου παρείχαν όλο αυτό ο καιρό.

## Πίνακας περιεχομένων

|                                                                                                              |           |
|--------------------------------------------------------------------------------------------------------------|-----------|
| <b>1. Περίληψη .....</b>                                                                                     | <b>1</b>  |
| <b>2. Εισαγωγή.....</b>                                                                                      | <b>5</b>  |
| 2.1 Θαλάσσιες μεταφορές .....                                                                                | 5         |
| 2.1.1 Χαρακτηριστικά Πλοίων και Λιμανιών.....                                                                | 6         |
| 2.1.2 Τύποι εν πλω μεταφορών .....                                                                           | 8         |
| 2.1.3 Χαρακτηριστικά Εμπορευμάτων.....                                                                       | 9         |
| 2.2 Εφοδιαστική Αλυσίδα και Διαχείριση Εφοδιαστικής Αλυσίδας.....                                            | 10        |
| 2.3 Ναυτιλιακή εφοδιαστική Αλυσίδα.....                                                                      | 12        |
| 2.4 Περιγραφή του Προβλήματος.....                                                                           | 12        |
| 2.5 Πρόβλημα Δρομολόγησης Οχημάτων (VRP).....                                                                | 13        |
| 2.5.1 Ένα προς Ένα Πρόβλημα Διανομής και Παραλαβής Προϊόντων .....                                           | 14        |
| (1-1 PDP) .....                                                                                              | 14        |
| 2.5.2 Η Έγπαρξη πολλαπλών σημείων εκκίνησης.....                                                             | 16        |
| (Multidepot Vehicle Routing) .....                                                                           | 16        |
| <b>3. Μοντελοποίηση Προβλήματος.....</b>                                                                     | <b>17</b> |
| <b>4 Αλγόριθμοι Βελτιστοποίησης .....</b>                                                                    | <b>21</b> |
| 4.1 Ευρετικοί.....                                                                                           | 21        |
| 4.1.1 Αλγόριθμοι Απληστίας .....                                                                             | 21        |
| 4.1.2 Προσεγγιστικοί Αλγόριθμοι.....                                                                         | 22        |
| 4.1.3 Αλγόριθμοι Τοπικής Αναζήτησης .....                                                                    | 22        |
| 4.2 Μεθευρετικοί .....                                                                                       | 24        |
| 4.3 Γενετικοί Αλγόριθμοι .....                                                                               | 25        |
| 4.4 Μιμητικοί Αλγόριθμοι .....                                                                               | 29        |
| <b>5. Τρόπος επίλυσης του προβλήματος.....</b>                                                               | <b>30</b> |
| 5.1 Εισαγωγή.....                                                                                            | 30        |
| 5.2 Αρχικοποίηση μεταβλητών – Προετοιμασία Δεδομένων.....                                                    | 31        |
| Διάγραμμα Ροής Αλγορίθμου .....                                                                              | 32        |
| 5.3 Δημιουργία Αρχικής Λύσης με χρήση διαδικασίας άπληστης τυχαιοποιημένης<br>προσαρμοστικής αναζήτησης..... | 34        |
| 5.3.1 GRASP.....                                                                                             | 34        |
| 5.4 Βελτίωση Των Λύσεων Με Χρήση Μιμητικού Αλγορίθμου.....                                                   | 37        |
| 5.4.1 Εφαρμογή Γενετικού.....                                                                                | 37        |
| 5.5 Βελτίωση του Πληθυσμού Λύσεων με Εφαρμογή Τοπικής Αναζήτησης.....                                        | 39        |
| 5.5.1 1-0 Επανατοποθέτηση.....                                                                               | 39        |

|                                            |           |
|--------------------------------------------|-----------|
| 5.5.2 1 -1 Ανταλλαγή.....                  | 40        |
| 5.6 Εισαγωγή φορτίων στη λύση.....         | 41        |
| <b>6. Παρουσίαση Αποτελεσμάτων.....</b>    | <b>42</b> |
| 6.1 Παραδείγματα εφαρμογής αλγορίθμου..... | 42        |
| 6.2 Αποτελέσματα Παραδειγμάτων.....        | 44        |
| 6.3 Συμπέρασμα .....                       | 43        |
| 6.4 Τροποποιήσεις στο Τρόπο Επίλυσης.....  | 44        |
| <b>7. Γενικός Επίλογος.....</b>            | <b>54</b> |
| <b>Βιβλιογραφία .....</b>                  | <b>57</b> |

## 1. Περίληψη

Η συγκεκριμένη διπλωματική εργασία αφορά την εφαρμογή ενός υβριδικού γενετικού αλγόριθμου στο πρόβλημα δρομολόγησης και χρονικού προγραμματισμού φορτηγών πλοίων βιομηχανικού φορτίου και ελεύθερων φορτηγών πλοίων ξηρού φορτίου. Η ναυτιλιακή εφοδιαστική αλυσίδα και οι θαλάσσιες μεταφορές στηρίζουν σημαντικά εδώ και πολλά χρόνια το παγκόσμιο εμπόριο κάτι που οδήγησε στην ανάγκη βελτιστοποίησης τους ώστε να γίνονται με όσο το δυνατό πιο αποδοτικό τρόπο. Η εργασία αναφέρεται στη δρομολόγηση και το χρονικό προγραμματισμό δύο τύπων θαλάσσιων μεταφορών. Στη μεταφορά βιομηχανικού φορτίου, που συναντάται όταν ο ιδιοκτήτης του φορτίου ελέγχει το στόλο των πλοίων και προσπαθεί να ελαχιστοποιήσει το κόστος μεταφοράς του και σε εκείνη των ελεύθερων φορτηγών πλοίων ξηρού φορτίου, στην οποία τα πλοία κινούνται ανάλογα με τα διαθέσιμα φορτία, εκτελώντας ένα συνδυασμό προαιρετικών και επιτακτικών συμβολών με σκοπό τη μεγιστοποίηση του κέρδους. Το πρόβλημα μοντελοποιείται ως ένα πρόβλημα διανομής και παραλαβή προϊόντων μέσα σε δεδομένα χρονικά περιθώρια με χρήση των κατάλληλων περιορισμών χωρητικότητας, συμβατότητας φορτίων – πλοίων, ροής μονοπατιού, χρονικών παραθύρων και διάρκειας της κάθε διαδρομής. Για την επίλυση του συγκεκριμένου προβλήματος δρομολόγησης και χρονικού προγραμματισμού χρησιμοποιείται αλγόριθμος που συνδυάζει ένα γενετικό αλγόριθμο με κάποιο αλγόριθμο τοπικής αναζήτησης. Η χρήση του μιμητικού αλγορίθμου γίνεται καθώς με αυτό το τρόπο συνδυάζεται η δύναμη ενός γενετικού αλγόριθμου με τη ταχύτητα μίας μεθόδου τοπικής αναζήτησης. Όσον αφορά το γενετικό αλγόριθμο πέρα από διασταύρωση των γονιδίων γίνεται κάποια μετάλλαξη στα γονίδια ώστε να βελτιωθεί όσον το δυνατόν περισσότερο η λύση που δίνεται στον αλγόριθμο τοπικής αναζήτησης, αποτέλεσμα του οποίου είναι μία βελτιωμένη λύση του προβλήματος.

## 2. Εισαγωγή

### 2.1 Θαλάσσιες μεταφορές

Η μεταφορά αγαθών σε παγκόσμιο επίπεδο εξαρτάται σε μεγάλο βαθμό από τις θαλάσσιες μεταφορές. Δισεκατομμύρια τόνοι εμπορευμάτων μεταφέρονται ετησίως με χρήση φορηγών πλοίων, η χωρητικότητα των οποίων ξεπερνά παγκοσμίως τους 1.6 δισεκατομμύρια τόνους (UNCTAD, 2013). Για αυτό το λόγο, η ναυτιλιακή εφοδιαστική αλυσίδα αποτελεί πυλώνα του παγκόσμιου εμπορίου και κατ' επέκταση της παγκόσμιας οικονομίας, στηρίζοντας τις ανθρώπινες δραστηριότητες εδώ και εκατοντάδες χρόνια. Όπως είναι προφανές, εξαιτίας του μεγάλου όγκου εμπορευμάτων που μεταφέρονται με χρήση πλοίων απαιτείται η συστηματοποίηση των διαδικασιών που ακολουθούνται από τις εταιρίες. Παρ' όλα αυτά η ανάγκη για βελτιστοποίηση των διαδικασιών ήρθε με την έναρξη της οικονομικής κρίσης του 2008 που έπληξε παγκοσμίως τις θαλάσσιες μεταφορές. Συγκεκριμένα, ο τομέας των μεταφορών ξηρού φορτίου σημείωσε δραματική πτώση με τον BDI (Baltic Dry Index) να μειώνεται κατά 80% (UNCTAD, 2017). Έτσι πολλές εταιρίες οδηγήθηκαν σε μείωση του στόλου. Δεδομένων αυτών των συνθηκών, μία ναυτιλιακή εταιρία ήταν εφικτό να επιβιώσει μόνο μέσω αύξησης της αποδοτικότητας της, κάτι που οδήγησε στην ανάπτυξη εξειδικευμένων συστημάτων υποστήριξης αποφάσεων με σκοπό τη βελτιστοποίηση των διαδικασιών. Για αυτό το λόγο, το αντικείμενο της συγκεκριμένης εργασίας αποτελεί κόμματα της λήψης αποφάσεων πολλών εταιριών του συγκεκριμένου τομέα (HEMMATI, 2014).

Οι θαλάσσιες μεταφορές αποτελούν ένα από τα μεγαλύτερα και πιο σύνθετα προβλήματα οργάνωσης και σχεδιασμού που συναντώνται στο χώρο της επιχειρησιακής έρευνας. Η πληθώρα των παραμέτρων που πρέπει να οριστούν και των υποπροβλημάτων που χρήζουν επίλυσης, κάνουν το πρόβλημα των θαλάσσιων μεταφορών αντικείμενο για αρκετούς ερευνητές και εμάς ανήμπορους να καλύψουμε όλη την έκταση του στα πλαίσια της παρούσας διπλωματικής εργασίας. Συγκεκριμένα, τα προβλήματα που πρέπει να αντιμετωπιστούν στις εν πλω μεταφορές μπορούν να χωριστούν με το κλασσικό τρόπο ανάλογα με το φάσμα σχεδιασμού σε :

- Προβλήματα Στρατηγικής
  - Επιλογή αγοράς και εμπορικών συναλλαγών
  - Σχεδιασμός Πλοίων
  - Σχεδιασμός δικτύου μεταφορών και συστήματος μεταφορών και καθορισμός σημείων μεταφόρτωσης
  - Καθορισμός μεγέθους στόλου, είδος και μέγεθος πλοίων

- Καθορισμός μεγέθους και τοποθεσίας λιμανιών και τερματικών σταθμών και σχεδιασμός λιμανιών.
- Προβλήματα Τακτικής
  - Προσαρμογές στο μέγεθος και το είδος του στόλου πλοίων
  - Ανάθεση πλοίων σε συγκεκριμένες εμπορικές οδούς
  - Δρομολόγηση και χρονικός προγραμματισμός φορτηγών πλοίων
  - Δρομολόγηση πλοίων καταγραφής αποθεμάτων
  - Χρονικός προγραμματισμός ελλιμενισμού πλοίων
  - Χρονικός προγραμματισμός λειτουργίας γερανών
  - Διαχείριση χώρου αποθήκευσης κοντέινερ
  - Σχεδιασμός αποθήκευσης κοντέινερ
  - Διαχείριση πλοίων
  - Διαχείριση αδειών κοντέινερ
- Λειτουργικά Προβλήματα
  - Καθορισμός ταχύτητας πλεύσης
  - Φόρτωση και εκφόρτωση των πλοίων
  - Φιλική προς το περιβάλλον δρομολόγηση

Επιπλέον, εμπόδια μπορεί να δημιουργηθούν στη μεταφορά επικίνδυνων υλικών. Ωστόσο, τέτοιου είδους θέματα αφορούν λειτουργίες που γίνονται στα εκάστοτε λιμάνια κάτι που δεν απασχολεί τη παρούσα εργασία .

### 2.1.1 Χαρακτηριστικά Πλοίων και Λιμανιών

Ένας στόλος πλοίων απαρτίζεται από πλοία πολλών μεγεθών και μορφών έτσι ώστε να είναι ικανός να μεταφέρει εμπορεύματα όλων των ειδών και να προσεγγίσει λιμάνια κάθε μεγέθους. Το μέγεθος ενός φορτηγού πλοίου μετριέται από τη μάζα εμπορεύματος που είναι ικανό να μεταφέρει και από τον όγκο των φορτίων που μπορούν να χωρέσουν σε αυτό χωρίς να υπερβεί την μέγιστη μάζα φορτίου. Η μονάδα μέτρησης του συνολικού φορτίου ενός φορτηγού πλοίου, δηλαδή του αθροίσματος της μάζας του εμπορεύματος, των καυσίμων, προμηθειών και οτιδήποτε άλλο βρίσκεται εντός του πλοίου, είναι ο μετρικός τόνος. Ο συνολικός όγκος εσωτερικού χώρου ενός πλοίου ονομάζεται Ολική Χωρητικότητα (Gross Tons) και μετριέται σε εκατοντάδες κυβικά πόδια. Επιπλέον, είναι αναγκαία η ύπαρξη πλοίων διαφορετικών τύπων, έτσι ώστε να είναι εφικτή η εξυπηρέτηση εμπορευμάτων όλων των ειδών. Ο διαχωρισμός γίνεται ανάλογα με το είδος εμπορευμάτων που μπορούν να μεταφέρουν. Συγκεκριμένα χωρίζονται σε :

- *Δεξαμενόπλοια*. Αυτά τα πλοία είναι ικανά με την χρήση ενσωματωμένων δεξαμενών να μεταφέρουν χύμα υγρά φορτία. Ανάλογα από το μέγεθος τους μπορούν να μεταφέρουν από προϊόντα πετρελαίου, χημικά, μέχρι και αργό πετρέλαιο.

- ο Πλοίο μεταφοράς φυσικού αερίου (LNG Carrier). Τέτοιου είδους πλοία ειδικεύονται στη μεταφοράς υγροποιημένου φυσικού αερίου. Συνήθως διαθέτουν έξι δεξαμενές, οι οποίες βρίσκονται σε σειρά στο κέντρο του πλοίου και μέσα σε αυτές μεταφέρεται το υγροποιημένο φυσικό αέριο.
- ο Πλοίο μεταφοράς χύδην φορτίου. Σκοπός τους είναι να μεταφέρουν χύμα ξηρά φορτία όπως άνθρακα, σιτηρά, βωξίτη, σίδηρο, φώσφορο και άλλα μεταλλεύματα. Τέτοιου είδους πλοία είναι εξοπλισμένα με ειδικό εξοπλισμό ώστε να μπορούν να φορτώνουν και να ξεφορτώνουν χωρίς να χρειάζεται το λιμάνι να τους εξυπηρετήσει.
- ο Πλοία μεταφοράς εμπορευματοκιβωτίων. Πλοία τέτοιου είδους χρησιμοποιούνται για τη μεταφορά εμπορευματοκιβωτίων συγκεκριμένου μεγέθους. Μέσα σε εμπορευματοκιβώτια τοποθετούνται συσκευασμένα προϊόντα παντός είδους.
- ο Φορτηγά πλοία γενικού φορτίου. Σε αντίθεση με τα πλοία μεταφοράς εμπορευματοκιβωτίων, αυτό το είδος πλοίων μεταφέρει τα συσκευασμένα προϊόντα χύμα στα πολλαπλά καταστρώματα που διαθέτει και σε ειδικούς χώρους. Όπως είναι προφανές, η φόρτωση και εκφόρτωση του πλοίου αποτελεί μία πολύ χρονοβόρα και δαπανηρή διαδικασία. Ως εκ τούτου, τις τελευταίες τέσσερις δεκαετίες όλα τα φορτηγά πλοία τέτοιου είδους έχουν μετατραπεί σε πλοία μεταφοράς εμπορευματοκιβωτίων κάνοντας τη χρήση τους πιο συμφέρουσα..
- ο Φορτηγά πλοία-ψυγεία (reefers), τα οποία χρησιμοποιούνται για τη μεταφορά εμπορευμάτων που η θερμοκρασία τους πρέπει να διατηρείται σε χαμηλές τιμές κατά τη διάρκεια της μετακίνησης τους.
- ο Οχηματαγωγά. Τα οχηματαγωγά πλοία είναι εξοπλισμένα με ράμπες τις οποίες τοποθετούν στις άκρες των λιμανιών ώστε να επιβιβαστούν οδικώς φορτηγά, αυτοκίνητα και άλλα οχήματα παντός τύπου.

Η λίστα των πλοίων εκτείνεται και σε πλοία ειδικού τύπου που εξειδικεύονται σε συγκεκριμένες εργασίες και δεν αποτελούν αντικείμενο του παρόντος προβλήματος. Πλοία όπως επιβατικά, ερευνητικά πλοία, τράτες, ρυμουλκά κλπ. Χαρακτηριστικά πλοίων από τη παραπάνω λίστα μπορούν συνδυαστούν σε κάποιες περιπτώσεις για τη καλύτερη εξυπηρέτηση όπως επιβατικό και οχηματαγωγό πλοίο.

Τα λιμάνια που εξυπηρετούν τα πλοία συνήθως χωρίζονται σε λιμάνια φόρτωσης και λιμάνια εκφόρτωσης. Όπως είναι προφανές στα λιμάνια φόρτωσης το εκάστοτε πλοίο πηγαίνει ώστε να παραλάβει εμπορεύματα και αντίστοιχα επισκέπτεται κάποιο λιμάνι εκφόρτωσης για να παραδώσει κάποιο εμπόρευμα. Ένα λιμάνι μπορεί να αποτελέσει λιμάνι φόρτωσης για κάποιο πλοίο και ταυτόχρονα λιμάνι εκφόρτωσης για κάποιο άλλο. Πέρα από τη φόρτωση και εκφόρτωση εμπορευμάτων τα πλοία σε οποιοδήποτε λιμάνι μπορούν να εφοδιαστούν με καύσιμα, καθαρό νερό και προμήθειες. Τέλος, ένα λιμάνι μπορεί να χρησιμοποιηθεί και για την μεταφόρτωση κάποιου εμπορεύματος.



### 2.1.2 Τύποι εν πλω μεταφορών

Οι θαλάσσιες μεταφορές συχνά διαχωρίζονται στις τρεις παρακάτω εκδοχές ανάλογα με τις διαδικασίες που ακολουθούνται.

- ο *Μεταφορές με πλοία γραμμής (Liner Shipping)*. Στη συγκεκριμένη περίπτωση θαλάσσιων μεταφορών τα δρομολόγια των πλοίων είναι συγκεκριμένα και γνωστά, με τη λειτουργία τους να θυμίζει εκείνη των μέσων μαζικής μεταφοράς.
- ο *Μεταφορές βιομηχανικού φορτίου (Industrial Shipping)*. Χαρακτηριστικό αυτών των μεταφορών είναι πως ο ιδιοκτήτης των φορτίων ελέγχει και τον στόλο των πλοίων, με σκοπό την ελαχιστοποίηση του κόστους μεταφοράς των εμπορευμάτων του. Η μεταφορά γίνεται από συγκεκριμένα λιμάνια φόρτωσης, τα οποία αποτελούν είτε τα λιμάνια παραγωγής της επιχείρησής που διαχειρίζεται το στόλο, είτε κάποιον μεταβατικό χώρο αποθήκευσης. Το συγκεκριμένο είδος μεταφορών συναντάται κυρίως σε μεταφορές μεγάλων ποσοτήτων υγρών όπως, πετρέλαιο, χημικά και στο εμπόριο ξηρών φορτίων.
- ο *Μεταφορές ελεύθερων πλοίων ξηρού φορτίου (tramp shipping)*. Σε αυτό το είδος θαλάσσιων μεταφορών τα πλοία δρομολογούνται ανάλογα με τα διαθέσιμα φορτία που υπάρχουν, επιλέγοντας ανάμεσα σε προαιρετικά φορτία και σε φορτία που η μεταφορά τους είναι υποχρεωτική. Η μεταφορά των τελευταίων είναι δεσμευτική για την ναυτιλιακή εταιρία βάσει μακροπρόθεσμων συμβολαίων με την ενδιαφερόμενη εταιρία (NORSTAD,2011). Συνεπώς, δίνεται προτεραιότητα στη μεταφορά αυτών των φορτίων και στη συνέχεια εξετάζεται η μεταφορά των προαιρετικών φορτίων, μόνο στη περίπτωση που κάτι τέτοιο είναι κερδοφόρο για την εταιρία, χωρίς να επηρεάζεται η μεταφορά των υπολοίπων εμπορευμάτων.

Σε οποιοδήποτε τύπο εν πλω μεταφοράς είναι δυνατή η εκμίσθωση επιπλέον πλοίων, στη περίπτωση που η ζήτηση δε μπορεί να ικανοποιηθεί από το διαθέσιμο στόλο που διαθέτει η εταιρία. Στις μεταφορές με πλοία γραμμής και στις μεταφορές ελεύθερων πλοίων ξηρού φορτίου, σε περιπτώσεις υπερβολικής ζήτησης, είναι πιθανή η άρνηση εξυπηρέτησης κάποιου πελάτη αν η ικανοποίηση του με οποιοδήποτε τρόπο είναι ζημιογόνα για την εταιρία. Από την άλλη, στις μεταφορές βιομηχανικού φορτίου η εταιρία οφείλει να μεταφέρει όλα τα προϊόντα με οποιοδήποτε τρόπο ανεξαρτήτως του όγκου τους. Αντιθέτως σε περιπτώσεις χαμηλής ζήτησης, τα πλοία του στόλου της εταιρίας μπορεί να νοικιαστούν σε άλλους ενδιαφερόμενους, να γίνει παύση της λειτουργίας τους ή ακόμα και να καταστραφούν. Ωστόσο, στη περίπτωση των μεταφορών βιομηχανικού φορτίου η μείωση του στόλου είναι ένα πολύ απίθανο σενάριο, καθώς η εκάστοτε εταιρία συνήθως διαθέτει λιγότερα πλοία από αυτά που χρειάζεται, ώστε να μισθώνει επιπλέον ελεύθερα φορτηγά πλοία (tramp shipping) για να καλύψει τις ανάγκες της.

Η συγκεκριμένη διπλωματική αφορά τις δύο τελευταίες κατηγορίες θαλάσσιων μεταφορών και επικεντρώνεται στη δρομολόγηση χύμα φορτίων όπως αργό πετρέλαιο, χημικά, παράγωγα πετρελαίου (wet bulk) αλλά και μεταλλεύματα σιδήρου, βωξίτη, αλουμινία και φωσφορικό άλας (dry bulk). Μέχρι το 2016, τέτοιου είδους προϊόντα αποτελούσαν περισσότερο από το 60% του καθαρού φορτίου που μεταφερόταν μέσω τη θάλασσας (HOMSI,2020).

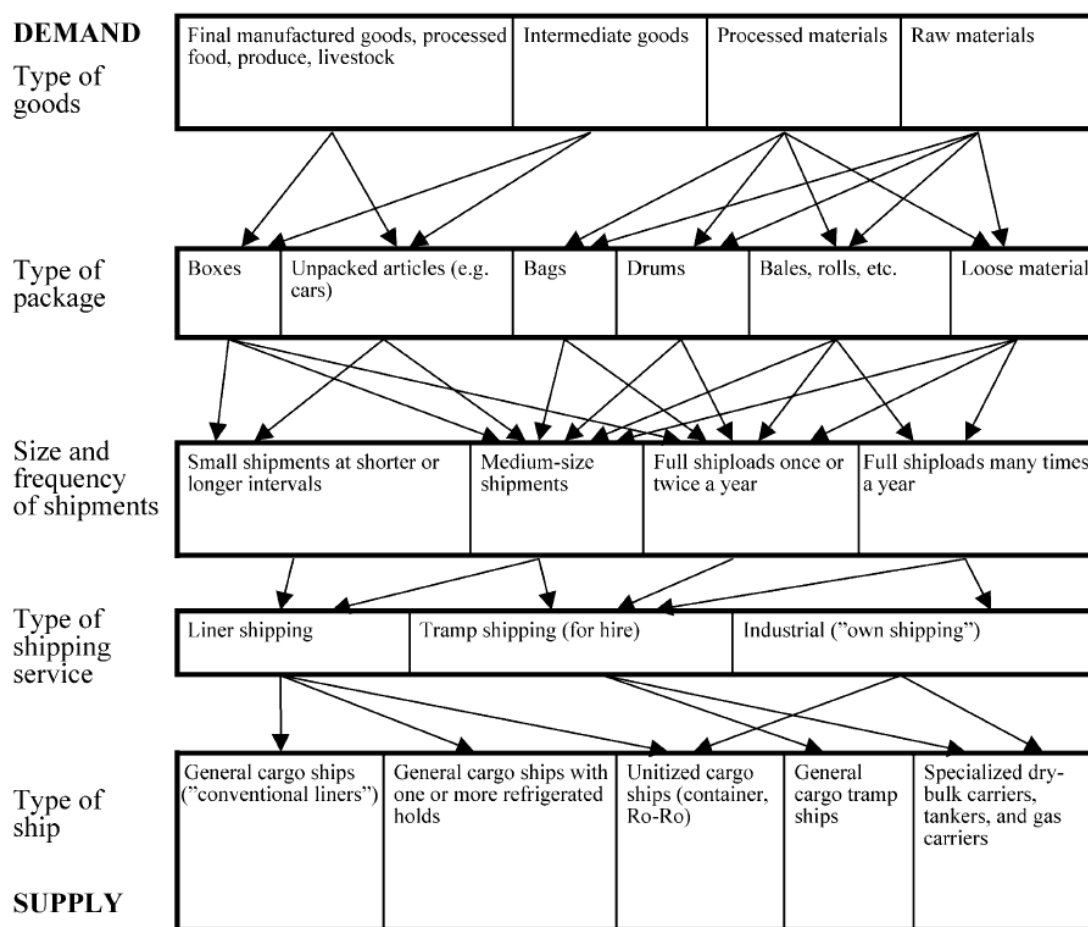
### 2.1.3 Χαρακτηριστικά Εμπορευμάτων

Με πλοία μεταφέρονται σχεδόν όλα τα προϊόντα που μπορούμε να συναντήσουμε σε οποιαδήποτε αγορά, από φρούτα, λαχανικά, επεξεργασμένες τροφές και ζωικό κεφάλαιο μέχρι βιομηχανικό εξοπλισμό, βιομηχανικά και οικοδομικά υλικά, μεταλλεύματα και προϊόντα πετρελαίου. Τα φορτία που είναι είτε χύμα, ασυσκευάστα είτε συσκευασμένα με κάποιο τρόπο. Ο τρόπος που συσκευάζονται τα προϊόντα ποικίλει ανάλογα με το είδος του προϊόντος και συνήθως είναι σε κουτιά, σακούλες, ρολά ή και δεμάτια. Τις περισσότερες φορές επίσης, μεταφέρονται συγκεντρωμένα σε παλέτες, εμπορευματοκιβώτια ή καρότσες ώστε η φόρτωση και εκφόρτωση τους στα λιμάνια να είναι ευκολότερη. Το μεγαλύτερο μέρος των εμπορευμάτων μεταφέρονται είτε χύμα με χρήση κατάλληλων πλοίων, είτε συγκεντρωμένα σε μεγαλύτερες συσκευασίες ώστε η διαχείριση τους να είναι μηχανοποιημένη. Εμπορεύματα, που δε φορτωνόταν με αυτούς τους τρόπους και απαιτούσαν χειρωνακτική εργασία για την διαχείριση τους, μεταφέρονταν συνήθως με χρήση πλοίων γραμμής. Παρ' όλα αυτά, τις τελευταίες δεκαετίες κάτι τέτοιο έχει σταματήσει να συμβαίνει και για τη μεταφορά τεμαχίων γίνεται χρήση εμπορευματοκιβωτίων ή καρότσας σε οποιοδήποτε είδος εν πλω μεταφοράς.

Σχεδόν όλα τα συσκευασμένα προϊόντα τοποθετούνται στα πλοία σε εμπορευματοκιβώτια καθώς η χρήση τους έχει πολλά πλεονεκτήματα, αφού δίνεται η δυνατότητα μηχανοποίησης της διαδικασίας και μείωσης του χρόνου παραμονής των πλοίων στα λιμάνια αλλά αποτρέπει και τα περιστατικά μικροκλοπών.

Το *γράφημα 1* που ακολουθεί παρουσιάζει συνοπτικά τη σχέση μεταξύ προσφοράς και ζήτησης και δείχνει τη σύνδεση των χαρακτηριστικών των εμπορευμάτων με τα είδη των συσκευασιών, τα είδη μεταφορών και τους τύπους πλοίων. Με αυτό τον τρόπο δίνεται μία εικόνα του τρόπου συνεργασίας και λειτουργίας των διαφορετικών στοιχείων των θαλάσσιων μεταφορών που περιεγράφηκαν παραπάνω.

Γράφημα 1



Πηγή: CHRISTIANSEN, Marielle, et al. *Maritime transportation. Handbooks in operations research and management science*, 2007, 14: 189-284.

## 2.2 Εφοδιαστική Αλυσίδα και Διαχείριση Εφοδιαστικής Αλυσίδας

Όσον αφορά τη γενικότερη έννοια της Διαχείρισης Εφοδιαστικής Αλυσίδας (ΔΕΑ), στην οποία υπάγεται και η ναυτιλιακή εφοδιαστική αλυσίδα, η ίδια αποτελεί βοήθημα για πολλά από τα μεγαλύτερα και σημαντικότερα προβλήματα που η ανθρωπότητα καλείται να αντιμετωπίσει. Σε γενικές γραμμές, έτσι ορίζεται ένα σύνολο προτύπων κι διαδικασιών που αφορούν δραστηριότητες της εφοδιαστικής αλυσίδας και περιλαμβάνουν το προγραμματισμό και τη διαχείριση πόρων που σχετίζονται με σχεδόν όλες τις δραστηριότητες μίας επιχείρησης. Ξεκάθαρο παράδειγμα της αναγκαιότητας της ΔΕΑ για τη λύση προβλημάτων στις μέρες μας, αποτέλεσε το πρόβλημα της έγκαιρης διανομής του εμβολίου κατά του COVID-19 σε παγκόσμιο επίπεδο από πολλά μέρη του πλανήτη. Οι πολλές παράμετροι και ιδιαιτερότητες μεταφοράς του εμβολίου άλλα και η ανάγκη για άμεση διανομή του, κατέστησε το συγκεκριμένο πρόβλημα ένα από τα μεγαλύτερα προβλήματα ΔΕΑ μετά το Δεύτερο

Παγκόσμιο Πόλεμο. Εκτός αυτού σε εταιρικό επίπεδο, πολλές επιχειρήσεις εξαιτίας του εξαιρετικά ανταγωνιστικού περιβάλλοντος αλλά και της παγκοσμιοποίησης, έχουν επεκτείνει τις δραστηριότητες τους στο παγκόσμιο εμπόριο. Εκμεταλλευόμενες έτσι, τα επιτεύγματα της ΔΕΑ έχουν πλέον πρόσβαση σε πρώτες ύλες από οποιοδήποτε μέρος τη Γης εξασφαλίζοντας με αυτό τον τρόπο το μικρότερο κόστος απόκτησης τους. Δίνεται επιπλέον η δυνατότητα στις επιχειρήσεις να είναι ανταγωνιστικές σε παγκόσμιο επίπεδο και να διαθέτουν τα προϊόντα τους παράλληλα σε πολλές αγορές με αποδοτικό τρόπο. Συνεπώς, δίνεται η δυνατότητα στην εκάστοτε εταιρία να διεισδύσει σε νέες αγορές, στις οποίες ωστόσο θα πρέπει να λειτουργήσει αποδοτικά έτσι ώστε να ανταπεξέλθει στις νέες απαιτήσεις και να εξασφαλίσει ικανοποιητικό μερίδιο αγοράς. Με αυτό τον τρόπο θα αυξήσει τα κέρδη της ελέγχοντας παράλληλα την αύξηση του κόστους. Πέρα από την εμπόριο προϊόντων και πρώτων υλών ανάμεσα στις επιχειρήσεις, μέσω της ΔΕΑ γίνεται ανταλλαγή ιδεών, τεχνοτροπιών, προτύπων και πληροφοριών βοηθώντας έτσι στην εξέλιξη των εταιριών και στη καλύτερη εξυπηρέτηση και κάλυψη αναγκών των καταναλωτών. Έτσι, η επιχείρηση που θα μπορέσει να ικανοποιήσει τις ανάγκες της αγοράς θα καταφέρει να εδραιώσει τη θέση της και να αυξήσει τα κέρδη της. Η ΔΕΑ αποτελεί αναπόσπαστο κομμάτι κάθε εταιρίας που θέτει τον καταναλωτή στο επίκεντρο και μεριμνά για τη γρήγορη και αποτελεσματική εξυπηρέτηση του, με το μικρότερο δυνατό κόστος. Αυτή η σύνδεση μεταξύ ΔΕΑ και καταναλωτικές ανάγκες είναι ο λόγος που η ΔΕΑ αποτελεί συνεχώς εξελισσόμενο σύστημα, το οποίο επηρεάζει σε σημαντικό βαθμό την απόδοση και οργάνωση της εκάστοτε εταιρίας.

Στην ουσία η ΔΕΑ απαρτίζεται από βασικές μεθόδους και τεχνικές διαχείρισης των διαδικασιών που είναι αναγκαίες για την παράδοση ενός προϊόντος ή μίας υπηρεσίας στον καταναλωτή, συμπεριλαμβανομένων των ενεργειών μεταφοράς και μετατροπής της πρώτης ύλης ώστε, να ικανοποιηθούν οι ανάγκες του καταναλωτή. Το σύνολο όλων αυτών των διαδικασιών, πόρων, πρώτων υλών, πληροφοριών και ανθρώπων που συνδέονται με τη μεταφορά της πρώτης ύλης, την μετατροπή της στο τελικό προϊόν και η επακόλουθη αποτελεσματική παράδοση του στο πελάτη είναι γνωστό ως *Εφοδιαστική Αλυσίδα (ΕΑ)*. Η ΕΑ και η ΔΕΑ έχουν ως σκοπό την αύξηση της ολικής αξίας του τελικού προϊόντος ικανοποιώντας στο έπακρο τις ανάγκες του πελάτη, λαμβάνοντας ωστόσο υπόψιν το συνολικό κόστος της όλης διαδικασίας. Πέρα από τη στενή προσέγγιση του κέρδους (έσοδα-έξοδα), στην προκειμένη περίπτωση στα έσοδα δε υπολογίζεται μόνο η χρηματική αξία του προϊόντος αλλά και τα οφέλη από την αποτελεσματική εξυπηρέτηση του πελάτη, τα οποία έχουν αντίκτυπο στη φήμη και στην αξιοπιστία της εταιρίας όσο και στην εμπιστοσύνη του καταναλωτή προς αυτήν. Εκτός αυτού, στο κόστος εκτέλεσης όλων των διαδικασιών που προαναφέρθηκαν πέρα από τη χρηματική επιβάρυνση απόκτησης των υλών, μετατροπής τους και διάθεσής του τελικού προϊόντος, λαμβάνεται υπόψιν και το κόστος ποιότητας για την επιχείρηση. Το κόστος ποιότητας αποτελεί το επιπλέον κόστος που θα υπάρξει σε περίπτωση έλλειψης επαρκούς ποιότητας σε κάποια από τις διαδικασίες της ΕΑ και αυτό θα έχει επίπτωση στο τελικό προϊόν και στην εμπειρία του καταναλωτή με την επιχείρηση. Σημαντικό στη συγκεκριμένη περίπτωση είναι το κόστος ποιότητας να συγκρίνεται με το κόστος απόκτησης και διατήρησης του συγκεκριμένου επιπέδου ποιότητας ώστε να αποφευχθεί η επιβάρυνση μη αναγκαίου κόστους.

Συνεπώς, είναι προφανές πως η ΕΑ και κατ' επέκταση η ΔΕΑ επηρεάζουν σε σημαντικό βαθμό την πορεία μίας εταιρίας στην αγορά, καθώς ελέγχει σημαντικές λειτουργίες της. Καθορίζει τη στρατηγική που θα ακολουθήσει, ώστε να πετύχει τους στόχους της, προγραμματίζει. Σχεδιάζει την πορεία που θα ακολουθηθεί από πρώτες

ύλες και τελικό προϊόν, με σκοπό η παραλαβή και η παράδοση τους να γίνει έγκαιρα και αποδοτικά μειώνοντας το απαιτούμενο κόστος και παράλληλα μεγιστοποιώντας την ικανοποίηση των πελατών. Επιπλέον, διαχειρίζεται το επίπεδο των αποθεμάτων ελέγχοντας με αυτό το τρόπο το αναγκαίο κεφάλαιο κίνησης για την εκτέλεση των διαδικασιών αλλά και τις ταμειακές ροές. Έλεγχος υπάρχει μέσω ΔΕΑ και στα απαιτούμενα έξοδα τόσο για την παραγωγή (όπου αυτή υπάρχει) αλλά και για τη διάθεση του προϊόντος. Εκτός αυτού, δίνεται η δυνατότητα αξιολόγησης των επιδόσεων των προμηθευτών ώστε να διασφαλίζεται η καλή συνεργασία και να αποφεύγεται η υποβάθμιση της ποιότητας. Αξιολόγηση μπορεί να υπάρξει και στην ίδια την εταιρία, στο επίπεδο της επαρκούς εξυπηρέτησης του καταναλωτή, μέσω της εκτίμησης των αποδόσεων των δραστηριοτήτων της ΕΑ βελτίωση της απόδοσης της επιχείρησης.

### 2.3 Ναυτιλιακή εφοδιαστική Αλυσίδα

Η κύρια διαφορά μεταξύ της κλασικής εφοδιαστικής αλυσίδας και της ναυτιλιακής εφοδιαστικής αλυσίδας είναι ότι στη τελευταία ένα μέρος της αποτελείται από τις εν πλω μεταφορές. Πολλές εταιρίες, που έχουν συνεργάτες και πελάτες στο εξωτερικό ή έχουν αναπτυχθεί σε καινούργιες αγορές, η εφοδιαστική αλυσίδα τους περιέχει με κάποια μορφή, θαλάσσιες μεταφορές. Αυτές εμφανίζονται είτε μέσω κάποιου εξωτερικού συνεργάτη (tramp shipping) ή μέσω ιδιωτικού στόλου της εταιρίας (industrial shipping) ή τέλος, με χρήση πλοίου γραμμής (liner). Πέραν από τη κλασική ΕΑ, η ναυτιλιακή εφοδιαστική αλυσίδα δεν αποτελεί αντικείμενο ερευνάς για μελετητές που ασχολούνται με τη βελτιστοποίηση.

Όσον αφορά τις διαδικασίες που ακολουθούνται, η δρομολόγηση των πλοίων γίνεται υπο αυστηρούς χρονικούς και ποσοτικούς περιορισμούς. Η εκάστοτε εταιρία καθορίζει τις ποσότητες που θα μεταφερθούν και τα αυστηρά χρονικά πλαίσια στα οποία πρέπει να γίνει η εξυπηρέτηση. Στόχος της εταιρίας μέσω του χρονικού προγραμματισμού είναι να εξυπηρετήσει όλες τις απαιτήσεις (ζήτηση), ενώ ταυτόχρονα να μεγιστοποιεί το κέρδος της. Συχνά, η ναυτιλιακή εφοδιαστική αλυσίδα ασχολείται πέρα από τη δρομολόγηση των πλοίων και με λειτουργίες που αφορούν τη διαχείριση αποθεμάτων στα λιμάνια παραγωγής, σε λιμάνια αποθήκευσης αλλά και σε λιμάνια κατανάλωσης του προϊόντος μεταφοράς.

### 2.4 Περιγραφή του Προβλήματος

Στα συγκεκριμένα είδη θαλάσσιων μεταφορών (Tramp Shipping – Industrial Shipping) η εκάστοτε ναυτιλιακή εταιρία καλείται να μεταφέρει μία λίστα από προαιρετικά και υποχρεωτικά προς μεταφορά φορτία. Η φόρτωση και η εκφόρτωση των φορτίων στα αντίστοιχα λιμάνια παραλαβής και παράδοσης πρέπει να γίνεται μέσα σε γνωστά χρονικά περιθώρια (time windows). Συνεπώς, το πλοίο στο οποίο ανατέθηκε η παραλαβή ενός συγκεκριμένου φορτίου, θα πρέπει να έχει φτάσει στο αντίστοιχο λιμάνι (origin port) και να έχει ολοκληρώσει τη φόρτωση του εμπορεύματος

(service time) πριν το άνω χρονικό περιθώριο, ώστε η μεταφορά του συγκεκριμένου πλοίου να είναι εφικτή από αυτό το πλοίο.

Παρομοίως στα λιμάνια παράδοσης, για να είναι εφικτή η λύση, το πλοίο θα πρέπει να έχει φτάσει και να έχει ολοκληρώσει την εκφόρτωση κάθε εμπορεύματος που του έχει ανατεθεί μέσα στα χρονικά πλαίσια που έχουν ορισθεί για το κάθε φορτίο. Εκτός από τα χρονικά παράθυρα, κάτι άλλο που πρέπει να ληφθεί υπόψιν είναι η συμβατότητα μεταξύ πλοίου και φορτίου, καθώς κάθε πλοίο δεν είναι ικανό να μεταφέρει οποιοδήποτε εμπόρευμα. Έτσι, για να είναι εφικτή η λύση, θα πρέπει κάθε φορτίο, που εισάγεται στο δρομολόγιο ενός πλοίου, να είναι συμβατό με το συγκεκριμένο τύπο πλοίου. Εκτός αυτού, κάθε φορτίο πρέπει να εξυπηρετείτε στην ίδια διαδρομή, από ένα μόνο πλοίο. Τέλος, δεδομένου ότι έχουν ικανοποιηθεί οι παραπάνω προϋποθέσεις και το πλοίο φτάσει στο ζητούμενο λιμάνι φόρτωσης ή εκφόρτωσης, η εξυπηρέτηση του πλοίου από το λιμάνι διαρκεί κάποιες ώρες εωσότου το πλοίο μπορέσει να συνεχίσει τη πορεία του.

Για την λύση του συγκεκριμένου προβλήματος και τη δρομολόγηση των πλοίων με βάσει τις παραπάνω προϋποθέσεις αναπτύχθηκε ένας υβριδικός γενετικός αλγόριθμος (Μιμητικός), ο οποίος εφαρμόστηκε σε συγκεκριμένα παραδείγματα βασισμένα σε πραγματικές καταστάσεις της αγοράς. Αυτό έγινε χάρη στην πρόσφατη δημιουργία των συγκεκριμένων παραδειγμάτων από τους Hemmati et. al. (2014) οι οποίοι δημιούργησαν ένα σύνολο ποικίλων παραδειγμάτων μεγέθους από 8 έως 130 εμπορεύματα, τα οποία εμφανίζονται ως ζευγάρια λιμανιών (κόμβων) παραλαβής και παράδοσης αντίστοιχα (origin-and-destination pairs). Αξίζει να σημειωθεί πως λαμβάνοντας υπόψιν το μέγεθος των προβλημάτων που αντιμετωπίζονται σε πραγματικές συνθήκες, υπάρχει σημαντική απόκλιση μεταξύ των πραγματικών προβλημάτων και των συγκεκριμένων παραδειγμάτων, η οποία πρέπει να μειωθεί ώστε να ικανοποιηθούν οι ανάγκες της βιομηχανίας. Σκοπός του συγκεκριμένου αλγορίθμου είναι η δημιουργία δρομολογίων και διαδρομών για κάθε διαθέσιμο πλοίο σε μορφή αλληλουχίας κόμβων, όπου κάθε κόμβος αντιπροσωπεύει ένα λιμάνι. Με αυτό τον τρόπο καθορίζεται ποια εμπορεύματα θα μεταφερθούν από το στόλο πλοίων της εταιρίας, και για ποια θα χρειαστεί η μίσθωση άλλων πλοίων. Με αυτό το τρόπο η εταιρία θα μεταφέρει όλα τα εμπορεύματα που της απαιτούνται. Απώτερος σκοπός είναι η μείωση του κόστους και η χρήση όσο το δυνατόν λιγότερων μισθωμένων πλοίων. Η συγκεκριμένη εργασία ασχολείται με τύπους εμπορευμάτων, των οποίων το μέγεθος είναι συγκεκριμένο (Fixed) και ικανοποιεί πλήρως τη χωρητικότητα των πλοίων. Ως εκ τούτου, τα εμπορεύματα εμφανίζονται σε ζευγάρια κόμβων (λιμανιών), ενώ μετά τη δημιουργία της διαδρομής, κάθε λιμάνι παραλαβής (Origin port) ακολουθείται πάντα από το αντίστοιχο λιμάνι παράδοσης του συγκεκριμένου εμπορεύματος.

## 2.5 Πρόβλημα Δρομολόγησης Οχημάτων (VRP)

Το κλασικό Πρόβλημα Δρομολόγησης Οχημάτων αφορά τη διανομή προϊόντων σε ένα σύνολο πελατών μέσα σε συγκεκριμένο χρονικό διάστημα. Η διακίνηση των προϊόντων γίνεται με αφετηρία μία ή περισσότερες αποθήκες από τις οποίες εκκινούν τα οχήματα. Ο αριθμός των οχημάτων είναι περιορισμένος και η κίνηση τους γίνεται μέσω συγκεκριμένων διαδρομών. Το οδικό δίκτυο που χρησιμοποιείται, στη συγκεκριμένη περίπτωση απεικονίζεται ως ένα διάγραμμα, όπου κάθε τόξο του αντιπροσωπεύει δρόμους και κάθε κόμβος την αποθήκη ή την τοποθεσία κάποιου

πελάτη. Κάθε τόξο μπορεί να είναι προσανατολισμένο ανάλογα με την κατεύθυνση του δρόμου και το μήκος του σχετίζεται με το κόστος που θα επιβαρυνθεί το όχημα, αν επιλέξει να περάσει από το συγκεκριμένο κομμάτι του δρόμου. Κάθε χρόνος ωστόσο μπορεί να μην είναι ίδιος για κάθε όχημα αφού εξαρτάται από το είδος του οχήματος. Γενικά, για τον ορισμό του κλασικού προβλήματος δρομολόγησης πρέπει να ληφθούν υπόψιν οι παρακάτω ενδεικτικές παράμετροι, που σχετίζονται με το μέγεθος της εταιρίας και τη διαδικασία διανομής προϊόντων:

- ο Το μέγεθος του στόλου των οχημάτων που χρησιμοποιούνται για τη διανομή.
- ο Ο αριθμός των οδηγών.
- ο Ο αριθμός των δρομολογίων (διαδρομών) που γίνονται καθημερινά και ο αριθμός των στάσεων ανά δρομολόγιο.
- ο Το συνολικό ετήσιο κόστος δραστηριοτήτων διαδρομής
- ο Η τρέχουσα υπολογιστική δύναμη της εταιρίας και η δυνατότητα υποστήριξης του δικτύου διανομής.

Η λύση του προβλήματος αποτελείται από ένα σύνολο διαδρομών, μία για κάθε διαθέσιμο όχημα. Όλες οι διαδρομές οφείλουν να ξεκινούν και να καταλήγουν σε μία ή περισσότερες αποθήκες με το ελάχιστο δυνατό κόστος και ικανοποιώντας τους περιορισμούς του προβλήματος. Κάθε πελάτης εξυπηρετείται μόνο από μία διαδρομή. Πέρα από την ελαχιστοποίηση του κόστους, δευτερεύοντα στόχο αποτελεί και χρήση όσον το δυνατόν λιγότερων οχημάτων για την ικανοποίηση της ζήτησης. Σε κάθε περίπτωση η λύση, για να είναι εφικτή, πρέπει να ικανοποιεί τους βασικούς περιορισμούς του προβλήματος. Μερικοί απ' αυτούς είναι:

- ο Σε κάθε δρομολόγιο (διαδρομή) ενός οχήματος η ποσότητα που μεταφέρεται δε μπορεί να ξεπερνά τη συνολική χωρητικότητα του.
- ο Είναι δυνατόν να υπάρξουν πελάτες που επιθυμούν μόνο παραλαβή είτε μόνο παράδοση προϊόντων. Ενώ μπορεί να επιθυμούν και τα δύο παράλληλα.
- ο Μπορεί κάποιος πελάτης να θέλουν να εξυπηρετηθούν πρώτοι οπότε η εξυπηρέτηση των υπολοίπων να ακολουθήσει μετά από αυτούς.
- ο Είναι δυνατόν η εξυπηρέτηση κάποιων πελατών να γίνεται μόνο σε κάποια συγκεκριμένη χρονική περίοδο.
- ο Η εργασία των οδηγών περιορίζεται σε συγκεκριμένο χρονικό διάστημα.

### 2.5.1 Ένα προς Ένα Πρόβλημα Διανομής και Παραλαβής Προϊόντων (1-1 PDP)

Σε μία πιο συνθέτη εκδοχή του VRP και πλησιάζοντας στο πρόβλημα που απασχολεί την ναυτιλιακή εφοδιαστική αλυσίδα προχωράμε στην 1-1 PDP παραλλαγή. Στη συγκεκριμένη παραλλαγή η ζήτηση του πελάτη δεν αποτελείται από μόνο παραλαβή ή μονό παράδοση προϊόντων ή ταυτόχρονη παραλαβή και παράδοση, αλλά εκφράζει τη μεταφορά μία συγκεκριμένης ποσότητας από ένα κόμβο παραλαβής (origin port) σε ένα κόμβο προορισμού (destination port). Στη προκειμένη περίπτωση, που σε κάθε λιμάνι φόρτωσης, (origin port) η ποσότητα εμπορεύματος που



παραλαμβάνεται, ικανοποιεί πλήρως τη χωρητικότητα του πλοίου, μετά από τη παραλαβή του προϊόντος για έναν συγκεκριμένο πελάτη και ακολουθεί η άμεση παράδοση του εμπορεύματος στο αντίστοιχο λιμάνι παράδοσης (destination port). Όσον αφορά τον ορισμό του προβλήματος, έχουμε ένα σύνολο κόμβων παραλαβής  $P = \{1, \dots, n\}$  και το αντίστοιχο σύνολο κόμβων διανομής  $D = \{n + 1, \dots, 2n\}$  αλλά και ένα σύνολο κόμβων που συμβολίζουν αποθήκες. Η εφαρμογή του 1-1 PDP στη ναυτιλιακή εφοδιαστική αλυσίδα δε διαθέτει αποθήκες αλλά σημεία εκκίνησης των πλοίων. Η ζήτηση  $i$  κάθε πελάτη περιλαμβάνει τη μεταφορά ενός φορτίου δεδομένου μεγέθους (fixed) από ένα κόμβο παραλαβής  $i \in P$  στον αντίστοιχο κόμβο παράδοσης  $n + i \in D$ .

Η εξυπηρέτηση των πλοίων στους κόμβους παραλαβής ή παράδοσης γίνεται για κάθε κόμβο  $i \in V$  όπου  $V$  το σύνολο των κόμβων, σε συγκεκριμένα χρονικά διαστήματα  $[a_i, b_i]$ , όπου  $a_i$  ο νωρίτερος χρόνος που μπορεί να φτάσει ένα πλοίο στο συγκεκριμένο κόμβο και  $b_i$  το αργότερο. Διαφορετικά, το συγκεκριμένο χρονικό διάστημα αποτελεί το χρόνο «ανοίγματος» και «κλεισίματος» του εκάστοτε λιμανιού. Τέλος, κάθε μετακίνηση (τόξο)  $i \rightarrow j$  χαρακτηρίζεται από το κόστος μετακίνησης  $c_{ij}$  και το χρόνο ολοκλήρωσης του ταξιδιού  $t_{ij}$ .

Στη συνέχεια, όσον αναφορά τους περιορισμούς που χαρακτηρίζουν το συγκεκριμένο πρόβλημα, είναι 11 και εξασφαλίζουν :

- ο Ότι η ζήτηση ενός πελάτη ικανοποιείται μία φορά.
- ο Ότι ο κόμβος παραλαβής και ο κόμβος παράδοσης μίας ζήτησης εξυπηρετείται από το ίδιο όχημα (πλοίο).
- ο Ότι όλες οι διαδρομές ξεκινούν από το κόμβο παραλαβής και καταλήγουν στο κόμβο παράδοσης.
- ο Τη συνοχή στις τιμές των μεταβλητών χρόνου και φόρτωσης, οι οποίες πρέπει να κινούνται αυξητικά, ανάλογα με τον αριθμό των κόμβων της διαδρομής.
- ο Τη περιορισμένη διάρκεια κάθε διαδρομής ανάλογα με το μέγιστο χρονικό παράθυρο, ενώ αντίστοιχοι περιορισμοί καθορίζουν τα χρονικά παράθυρα.
- ο Τη περιορισμένη χωρητικότητά των οχημάτων.

Αξίζει να σημειωθεί, πως τα πλοία εκκινούν άδεια και κινούνται προς τον πρώτο κόμβο παραλαβής. Δεν γίνεται χρήση αποθηκών, στην συγκεκριμένη εφαρμογή της παραλλαγής, έτσι κάθε πλοίο για να έχει πρόσβαση σε κάποιο εμπόρευμα πρέπει να κινηθεί στο αντίστοιχο λιμάνι παραλαβής.

Ένας ακόμη περιορισμός του προβλήματος, που δε συναντάτε στη κλασική εκδοχή της συγκεκριμένης παραλλαγής του VRP, αλλά απασχολεί ιδιαιτέρως τη δρομολόγηση πλοίων στη ναυτιλιακή εφοδιαστική αλυσίδα, είναι ο περιορισμός συμβατότητας μεταξύ εμπορευμάτων και πλοίων. Συγκεκριμένα, μία ναυτιλιακή εταιρία έχει τη δυνατότητα να μεταφέρει μεγάλη ποικιλία εμπορευμάτων, κάποια από τα οποία απαιτούν πλοίο συγκεκριμένου τύπου, για να μεταφερθούν. Κατ' επέκταση, τα πλοία που δύνανται να μεταφέρουν το συγκεκριμένο είδος φορτίου, δε μπορούν να εκτελέσουν τη μεταφορά εμπορευμάτων άλλων ειδών. Όπως είναι προφανές, η ανάθεση οποιουδήποτε εμπορεύματος σε οποιοδήποτε πλοίο είναι ανέφικτη, στο συγκεκριμένο πρόβλημα δρομολόγησης, για αυτό το λόγο προστίθεται ο περιορισμός συμβατότητας πλοίων - φορτίων.



### 2.5.2 Η Ύπαρξη πολλαπλών σημείων εκκίνησης (Multidepot Vehicle Routing)

Προσθέτοντας στην εκδοχή του προβλήματος δρομολόγησης οχημάτων, που περιεγράφηκε παραπάνω, τη παραλλαγή των πολλαπλών αποθηκών, σκιαγραφούμε πλήρως το πρόβλημα δρομολόγησης πλοίων που χρησιμοποιείται στη ναυτιλιακή εφοδιαστική αλυσίδα. Στη προκειμένη περίπτωση, που δε γίνεται χρήση αποθηκών, ο όρος, ύπαρξη πολλαπλών αποθηκών, μετατρέπεται σε ύπαρξη πολλαπλών σημείων εκκίνησης των πλοίων. Τα σημεία εκκίνησης κάθε πλοίου αντιστοιχούν σε συγκεκριμένα λιμάνια. Εκεί θα βρίσκεται το πλοίο τη χρονική στιγμή που θα είναι διαθέσιμο. Από κάποια λιμάνια μπορεί να εκκινούν πάνω από ένα πλοία σε ίδιες ή σε διαφορετικές χρονικές στιγμές. Κάθε πλοίο μπορεί να χρησιμοποιηθεί στη λύση μόνο μετά από τη χρονική στιγμή που θα είναι διαθέσιμο. Αυτή η χρονική στιγμή μπορεί να είναι στη αρχή ( $T=0$ ) ή σε οποιαδήποτε χρονική στιγμή, εφόσον είναι μικρότερη από τη μέγιστη διάρκεια της διαδρομής ( $T < T_{\max}$ ).

Συνεπώς, τα πλοία, που ανήκουν σε ένα συγκεκριμένο λιμάνι εκκίνησης, ξεκινούν τη διαδρομή τους τη χρονική στιγμή που είναι διαθέσιμα, κατευθυνόμενα προς κάποιο λιμάνι παραλαβής για να φορτώσουν ή στην περίπτωση που το λιμάνι εκκίνησης συμπίπτει με κάποιο λιμάνι παραλαβής, μπορεί να φορτώσουν απευθείας διανύοντας μηδενική απόσταση. Κάθε πλοίο ξεκινάει τη διαδρομή του άδειο. Τα πλοία μετά την εκκίνηση τους ανάλογα με τη λύση είναι δυνατόν να επιστρέψουν στο λιμάνι εκκίνησης για φόρτωση ή εκφόρτωση κάποιου φορτίου. Σε αντίθεση με τη κλασσική εκδοχή του Multidepot, η επιστροφή των πλοίων σε κάποιο λιμάνι εκκίνησης δεν είναι απαραίτητη.

Όπως και στις προηγούμενες παραλλαγές του κλασσικού VRP σκοπός είναι να εξυπηρετηθεί κάθε πελάτης ακριβώς μία φορά από το εκάστοτε πλοίο, η ζήτηση κάθε πελάτη δε πρέπει να ξεπερνάει τη χωρητικότητα των πλοίων (fixed capacity) και ο απώτερος σκοπός, πέρα από την εξυπηρέτηση των πελατών, είναι η ελαχιστοποίηση του κόστους.

Όσον αφορά του περιορισμούς που διατρέχουν τη συγκεκριμένη παραλλαγή του προβλήματος δρομολόγησης οχημάτων, εξασφαλίζουν ότι:

- Κάθε πελάτης εξυπηρετείται από ένα και μόνο ένα όχημα.
- Τηρείται η σωστή συνέχεια και αλληλουχία των κόμβων.
- Στη λύση δεν υπερβαίνεται τη χωρητικότητα κάποιου οχήματος.
- Στη λύση δεν υπερβαίνεται, από κάποιο όχημα, η μέγιστη χρονική διάρκεια κάθε διαδρομής.
- Τα οχήματα χρησιμοποιούνται τη χρονική στιγμή που είναι διαθέσιμα και όχι νωρίτερα.

Με αυτό τον τρόπο παρουσιάζεται και το τελευταίο κομμάτι του θεωρητικού υποβάθρου, στο οποίο στηρίζεται το συγκεκριμένο πρόβλημα, δρομολόγησης φορτηγών πλοίων βιομηχανικού φορτίου και ελεύθερων φορτηγών πλοίων ξηρού

φορτίου. Συμπερασματικά, αποτελεί μία πιο περιπλοκή εκδοχή του κλασικού προβλήματος δρομολόγησης οχημάτων (VRP), έτσι ώστε να ικανοποιηθούν οι επιπλέον απαιτήσεις του προβλήματος. Ξεκινώντας από το κλασικό VRP και λαμβάνοντας υπόψιν ότι: i) γίνεται χρήση ετερογενούς στόλου πλοίων, ii) η ζήτηση κάθε πελάτη ικανοποιείται από παραλαβή και αντίστοιχη παράδοση εμπορεύματος, iii) κάθε πλοίο μπορεί να εξυπηρετηθεί από κάποιο λιμάνι μόνο μέσα σε συγκεκριμένα χρονικά παράθυρα και iv) κάθε πλοίο ξεκινάει από το λιμάνι εκκίνησής του και μπορεί να χρησιμοποιηθεί στη λύση μόνο μετά τη χρονική στιγμή που θα είναι διαθέσιμο και v) η εξυπηρέτηση των πλοίων σε κάθε λιμάνι δε γίνεται ακαριαία αλλά διαρκεί συγκεκριμένο χρόνο, καταλήγουμε στο πρόβλημα δρομολόγησης και χρονικού προγραμματισμού φορτηγών πλοίων, που απασχολεί τη παρούσα διπλωματική.

### 3. Μοντελοποίηση Προβλήματος

Η μοντελοποίηση, που ακολουθεί το συγκεκριμένο πρόβλημα, ορίζεται από ένα γράφημα  $G = (V, A)$  όπου  $V$  αντιστοιχεί στην ένωση του συνόλου των κόμβων παραλαβής  $P = \{1, \dots, n\}$  και του συνόλου των κόμβων παράδοσης  $D = \{n+1, \dots, 2n\}$ . Τα  $n$  διαθέσιμα φορτία καλείται να μεταφέρει ένα σύνολο  $m$  πλοίων  $K = \{1, \dots, m\}$ . Για να ολοκληρωθεί η εξυπηρέτηση ενός φορτίου  $i \in \{1, \dots, n\}$  θα πρέπει αυτό να μεταφερθεί από έναν κόμβο παραλαβής  $i \in P$  στον αντίστοιχο κόμβο παράδοσης  $n+i \in D$ . Στη παρούσα εκδοχή του προβλήματος, η ποσότητα του φορτίου είναι συγκεκριμένη (fixed) και καλύπτει όλη τη χωρητικότητα του πλοίου, οπότε ορίζεται η ποσότητα του φορτίου  $i \in \{1, \dots, n\}$  ως  $q_i = B$  και η χωρητικότητα του πλοίου  $k \in K$  ως  $Q_k = B$  όπου  $B$  σταθερά. Έτσι ισχύει ότι,  $q_i = B$  για  $i \in P$  και  $q_{n+i} = -B$  για  $n+i \in D$ . Εφόσον οι συγκεκριμένες μεταβλητές είναι σταθερές δε λαμβάνονται υπόψιν. Κάθε διαθέσιμο λιμάνι χαρακτηρίζεται από ένα αυστηρό χρονικό παράθυρο  $[a_i, b_i]$ , όπου  $a_i$  είναι η χρονική στιγμή που ανοίγει το λιμάνι  $i$  και μπορεί να εξυπηρετήσει κάποιο πλοίο και  $b_i$  η χρονική στιγμή που το λιμάνι  $i$  σταματάει την εξυπηρέτηση πλοίων. Όσον αφορά τα φορτηγά πλοία, κάθε πλοίο  $k \in K$  μπορεί να χρησιμοποιηθεί στη λύση, μετά από συγκεκριμένη χρονική στιγμή  $s_{0k}^D \geq 0$  και εκκινεί από συγκεκριμένο λιμάνι (λιμάνι εκκίνησης)  $0_k$ . Κάθε διαδρομή  $arc(i, j) \in A$  μπορεί να διασχισθεί από οποιοδήποτε πλοίο  $k \in K$  με κόστος  $c_{ij}^k$ , το οποίο αντιστοιχεί στο κόστος των καυσίμων που καταναλώθηκαν και στη πληρωμή των διωρύγων, κατά το πέρασμα του πλοίου. Η διάρκεια κάθε διαδρομής  $arc(i, j) \in A$  ορίζεται ως  $\delta_{ij}^k \geq 0$ . Για την εισαγωγή ενός επιπλέον περιορισμού, που αφορά τη συμβατότητα μεταξύ φορτίων και πλοίων, θα πρέπει να προστεθεί μία δυαδική μεταβλητή  $I_{ik}$  η οποία δείχνει αν το φορτίο  $i$  είναι συμβατό με το πλοίο  $k$ . Σε περίπτωση που με τον δεδομένο στόλο δε γίνεται να ικανοποιηθεί η ζήτηση κάποιων πελατών, στο κόστος της λύσης προστίθεται επιπλέον κόστος  $s_{i0}^C$  για κάθε φορτίο  $i$  που δεν εξυπηρετείται. Αυτό το κόστος αντιπροσωπεύει την απώλεια εσόδων από τη μη εξυπηρέτηση του συγκεκριμένου φορτίου ή τη μίσθωση επιπλέον πλοίου που θα αναλάβει τη μεταφορά του εμπορεύματος.

Η λύση του προβλήματος αποτελείται από διαδρομές αριθμητικά ίσες με τα διαθέσιμα πλοία και στοχεύει στην ελαχιστοποίηση του συνολικού κόστους μεταφοράς και στην εξυπηρέτηση όσο το δυνατόν περισσότερων φορτίων, έτσι ώστε να αποφευχθεί η μίσθωση επιπλέον πλοίων και να αποτραπεί η απώλεια εσόδων. Κάθε διαδρομή της λύσης ξεκινάει από το λιμάνι εκκίνησης του αντίστοιχου πλοίου και μπορεί να καταλήξει σε οποιοδήποτε λιμάνι, ανάλογα με τα φορτία που εξυπηρετεί το πλοίο. Σε αντίθεση με το κλασικό VRP, οι διαδρομές δεν είναι απαραίτητο να έχουν κάποιο συγκεκριμένο τερματικό κόμβο, καθώς τα πλοία εκτελούν δρομολόγια συνεχώς, όλο το εικοσιτετράωρο. Η λύση θεωρείται εφικτή μόνο όταν όλες οι διαδρομές της ικανοποιούν τους περιορισμούς της μοντελοποίησης. Συγκεκριμένα είναι εφικτή όταν :

- Όλα τα πλοία φτάνουν στα λιμάνια των φορτίων που εξυπηρετούν μέσα στα αντίστοιχα χρονικά παράθυρα.
- Όλα τα πλοία εξυπηρετούν συμβατά σε αυτά εμπορεύματα .
- Τηρείται η σωστή αλληλουχία κόμβων. Κάθε κόμβος παραλαβής  $i \in P$  πρέπει να ακολουθείται απευθείας από τον αντίστοιχο κόμβο παράδοσης  $n + i \in D$  χωρίς να μεσολαβεί άλλος κόμβος. Κάθε ζευγάρι  $(i \in P, n + i \in D)$  πρέπει να εξυπηρετείται από το ίδιο πλοίο  $k \in K$  . Ο κόμβος παραλαβής  $i \in P$  πρέπει να προηγείται από τον αντίστοιχο κόμβο παράδοσης. Η διαδρομή πρέπει να καταλήγει σε κόμβο παράδοσης  $n + i \in D$  .

Η μαθηματική μοντελοποίηση του προβλήματος παρουσιάζεται στις εξισώσεις (1)-(14). Πριν προχωρήσουμε στις εξισώσεις που περιγράφουν το πρόβλημα, πρέπει να οριστούν μεταβλητές που είναι χρήσιμες για τη μαθηματική μοντελοποίηση. Έστω  $\Omega_k$  το σύνολο των κόμβων που μπορούν να επισκεφθούν από το πλοίο  $k$  και  $l_{ik}$  ο συνολικός όγκος φορτίου, που μεταφέρει το πλοίο  $k$  μετά την αναχώρηση του από το κόμβο  $i$  . Εκτός αυτής, ορίζεται η δυαδική μεταβλητή  $x_{ijk}$ , που παίρνει τη τιμή 1 όταν το πλοίο  $k$  κινείται απευθείας από το κόμβο  $i$  προς το κόμβο  $j$  , η δυαδική μεταβλητή  $y_i$  που δείχνει αν το φορτίο  $i$  μεταφέρεται από το διαθέσιμο στόλο πλοίων ή χρησιμοποιείται μισθωμένο πλοίο και η μεταβλητή  $t_{ik}$ , που αντιστοιχεί στη χρονική στιγμή που ξεκινάει η εξυπηρέτηση του πλοίου  $k$  στο κόμβο  $i$ .

$$\min \sum_{k \in K} \sum_{(i,j) \in \Omega_k} C_{ijk} x_{ijk} + \sum_{i \in P} s_{i0}^C y_i \quad (1)$$

Υπό

$$\sum_{k \in K} \sum_{(i,j) \in \Omega_k} x_{ijk} + y_i = 1 \quad i \in P, \quad (2)$$

$$\sum_{j \in \Omega_k} x_{pjk} = 1, \quad k \in K, \quad (3)$$

$$\sum_{j \in \Omega_k} x_{ijk} - \sum_{j \in \Omega_k} x_{jik} = 0, \quad k \in K, i \in \Omega_k \setminus \{P, D\}, \quad (4)$$

$$\sum_{j \in \Omega_k} x_{jdk} = 1, \quad k \in K, \quad (5)$$

$$l_{ik} + q_j - l_{jk} \leq Q_k(1 - x_{ijk}), \quad k \in K, j \in D, (i, j) \in A \quad (6)$$

$$l_{ik} + q_j - l_{(n+j)jk} \leq Q_k(1 - x_{i(j+n)k}), \quad k \in K, j \in D, (i, j) \in A \quad (7)$$

$$0 \leq l_{ik} \leq Q_k \quad k \in K, \quad i \in P \quad (8)$$

$$t_{ik} + \delta_{ij}^k - t_{jk} \leq (b_i + \delta_{ij}^k)(1 - x_{jik}), \quad k \in K, (i, j) \in A, \quad (9)$$

$$\sum_{j \in \Omega_k} x_{jik} - \sum_{j \in \Omega_k} x_{(n+i)jk} = 0, \quad k \in K, i \in P, \quad (10)$$

$$t_{ik} + \delta_{i(n+i)}^k - t_{(n+i)k} \leq 0, \quad k \in K, i \in P, \quad (11)$$

$$a_i \leq t_{ik} \leq b_i, \quad k \in K, i \in \Omega_k, \quad (12)$$

$$y_i \in \{0,1\}, \quad i \in \Omega_k, \quad (13)$$

$$x_{jik} \in \{0,1\}, \quad k \in K, (i, j) \in A \quad (14)$$

Η εξίσωση (1) αναπαριστά το άθροισμα των λειτουργικών εξόδων και των επιπρόσθετων κοστών από τη μη εξυπηρέτηση κάποιων φορτίων (ποινές). Όσον αφορά τους περιορισμούς, η εξίσωση (2) καθορίζει ότι κάθε διαθέσιμο φορτίο πρέπει είτε να εξυπηρευτεί από το στόλο των πλοίων είτε αν αυτό δεν είναι εφικτό να προστεθεί η αντίστοιχη «ποινή» στην αντικειμενική συνάρτηση. Οι περιορισμοί (3),(4),(5) είναι υπεύθυνοι για την τήρηση της σωστής αλληλουχίας κόμβων στη διαδρομή κάθε πλοίου  $k$ . Όσον αφορά τους περιορισμούς (6) και (7), είναι υπεύθυνοι για τον έλεγχο του φορτωμένου φορτίου σε κάθε κόμβο παραλαβής ή παράδοσης, ενώ ο (8) εξασφαλίζει ότι το φορτίο που μεταφέρει το πλοίο μετά από κάθε κόμβο δε ξεπερνά τη χωρητικότητα του πλοίου. Στην περίπτωση ωστόσο, της δρομολόγησης πλήρους φορτίου (full load) οι περιορισμοί 6,7 και 8 δε λαμβάνονται υπόψιν, καθώς ο όγκος φορτίου μετά από κάθε κόμβο είναι πάντα σταθερός και μικρότερος της

χωρητικότητας . Η ανισότητα (9) ελέγχει αν η εξυπηρέτηση του πλοίου  $k$  ξεκινάει τη σωστή χρονική στιγμή, δεδομένου του χρόνου που χρειάζεται το πλοίο για να φτάσει (travel time). Από τον περιορισμό (10) εξασφαλίζεται ότι το λιμάνι παραλαβής και το λιμάνι παράδοσης, ενός συγκεκριμένου φορτίου, εξυπηρετούνται από το ίδιο πλοίο  $k$ . Μέσω του περιορισμού (11) τηρείται η σωστή αλληλουχία κόμβων παραλαβής και παράδοσης. Με άλλα λόγια, καθορίζει ότι κάθε πλοίο επισκέπτεται το λιμάνι παραλαβής ενός φορτίου πριν από το λιμάνι παράδοσης. Ο περιορισμός (12) εισάγει στη μοντελοποίηση τα χρονικά παράθυρα κάθε κόμβου. Τέλος, από τους περιορισμούς (13) και (14) ορίζονται οι δυαδικές μεταβλητές που περιγράφηκαν.

Η παραπάνω μοντελοποίηση, χρησιμοποιείται για την επίλυση προβλημάτων δρομολόγησης και χρονικού προγραμματισμού φορτηγών πλοίων βιομηχανικού φορτίου και ελεύθερων φορτηγών πλοίων ξηρού φορτίου. Η εφαρμογή της είναι εφικτή τόσο σε προβλήματα εν πλω μεταφορών, μεγάλων αποστάσεων (deep sea shipping) όσο και σε θαλάσσιες μεταφορές μικρών αποστάσεων (short sea shipping). Εκτός αυτού, με τη προσθήκη κάποιων επιπλέον περιορισμών μπορεί να χρησιμοποιηθεί αποδοτικά και σε προβλήματα δρομολόγησης φορτηγών πλοίων μικτού φορτίου (Mixed load). Ωστόσο, στη παρούσα εργασία εξετάζεται μόνο η περίπτωση δρομολόγησης πλοίων πλήρους φορτίου (full load), ως εκ τούτου έχουν παραληφθεί περιορισμοί που σχετίζονται με τη χωρητικότητα των πλοίων και τον όγκο των φορτίων.

## 4. Αλγόριθμοι Βελτιστοποίησης

### 4.1 Ευρετικοί

Το επόμενο στάδιο, από την ανάλυση και το καθορισμό του προβλήματος και της μοντελοποίησης που το διατρήχει, είναι η επιλογή του σωστού τρόπου επίλυσης. Η δυσκολία της επίλυσης είναι ανάλογη του μεγέθους του προβλήματος, έχοντας επίπτωση στο χρόνο επίλυσης. Ο αλγόριθμος που θα επιλεγεί, θα πρέπει να είναι σε θέση να φτάσει σε μία εφικτή λύση σε λογικό χρόνο. Στη περίπτωση των ευρετικών αλγορίθμων, εξαιτίας της πολυπλοκότητας του προβλήματος, η χρήση τους δεν γίνεται για να βρεθεί η βέλτιστη λύση του προβλήματος, αλλά για να δημιουργηθεί μία αρχική λύση μέτριας απόδοσης, η οποία να σέβεται τους περιορισμούς της μοντελοποίησης. Η αρχική λύση του ευρετικού θα αποτελέσει την είσοδο άλλων αλγορίθμων που θα μπορέσουν να προσεγγίσουν τη λύση πιο αποδοτικά. Οι ευρετικοί είναι δυνατόν να διαχωριστούν σε κατηγορίες ανάλογα με τον τρόπο λειτουργίας τους. Συγκεκριμένα οι κατηγορίες είναι:

- ο Αλγόριθμοι απληστίας (greedy algorithms)
- ο Προσεγγιστικοί αλγόριθμοι (approximation algorithms)
- ο Αλγόριθμοι τοπικής αναζήτησης (local search algorithms)

Κύριο χαρακτηριστικό των δύο πρώτων κατηγοριών είναι πως χρησιμοποιούνται για τη δημιουργία αρχικής λύσης, ενώ οι αλγόριθμοι τοπικής αναζήτησης δέχονται ως είσοδο μία αρχική λύση και τη βελτιώνουν σταδιακά.

#### 4.1.1 Αλγόριθμοι Απληστίας

Οι αλγόριθμοι απληστίας προσπαθούν να οδηγήσουν σε μια εφικτή λύση του προβλήματος, αλλά πολλές φορές χρειάζονται πάρα πολύ μεγάλο χρόνο γιατί είναι μυωπικοί αλγόριθμοι, δηλαδή βλέπουν μόνο μπροστά. Ένας αλγόριθμος απληστίας μπορεί να ξεκινήσει από μία μη-εφικτή λύση και επαναληπτικά να ορίζεται μία μεταβλητή, έως ότου να βρεθεί μία εφικτή αρχική λύση. Σε πολλές περιπτώσεις ωστόσο επιτρέπεται στον αλγόριθμο να κινηθεί σε πεδία μη-εφικτών λύσεων, έτσι ώστε να αποφύγει τοπικά ελάχιστα. Εφικτές είναι μόνο οι λύσεις που δεν παραβιάζουν κανένα από τους περιορισμούς του προβλήματος. Η λειτουργία αυτού του είδους αλγορίθμων είναι επαναληπτική, όπου ξεκινώντας από έναν κόμβο εισάγει στο υποσύνολο της λύσης ένα κόμβο τη φορά. Η διαδικασία συνεχίζεται, λαμβάνοντας υπόψιν τους περιορισμούς μέχρι να προκύψει μία ολοκληρωμένη εφικτή λύση. Οι κόμβοι που εισάγονται σε κάθε βήμα διαλέγονται με συγκεκριμένο τρόπο, ανάλογα με τον αλγόριθμο που χρησιμοποιείται.

### 4.1.2 Προσεγγιστικοί Αλγόριθμοι

Ένας προσεγγιστικός αλγόριθμος στοχεύει στο να λύσει το πρόβλημα, κάνοντας χρήση επιπλέον πληροφορίας. Η ποιότητα των εφικτών λύσεων που προκύπτουν δε μπορεί να προσεγγίσει τη βέλτιστη λύση, καθώς περιορίζεται πάντοτε από το λόγο προσέγγισης του αλγορίθμου. Τέτοιου είδους αλγόριθμοι είναι χρήσιμοι όταν η προσέγγιση της βέλτιστης λύσης δεν είναι εφικτή σε πολυωνυμικό χρόνο, ή ο αλγόριθμος για τη βέλτιστη λύση είναι αρκετά περίπλοκος. Ένας προσεγγιστικός αλγόριθμος αξιολογείται μέσω της χειρότερης περίπτωσης του πιθανώς σχετικού λάθους σε όλα τα πιθανά παραδείγματα του προβλήματος και είναι πάντα πολυωνυμικός. Ένας αλγόριθμος  $A$  μπορεί να θεωρηθεί ως ένας  $\delta$  - προσεγγιστικός αλγόριθμος για ένα πρόβλημα ελαχιστοποίησης  $P$  αν για κάθε παράδειγμα  $I$  του  $P$  δίνει μια λύση η οποία είναι το πολύ  $\delta$  φορές μεγαλύτερη από τη βέλτιστη. Προφανώς,  $\delta > 1$  και όσο πλησιάζει στη μονάδα τόσο καλύτερα. Παρόμοια, για προβλήματα μεγιστοποίησης, ένας  $\delta$  προσεγγιστικός αλγόριθμος δίνει για κάθε παράδειγμα  $I$  μια λύση που είναι τουλάχιστον  $\delta$  φορές μεγαλύτερη από τη μέγιστη. Σε αυτή την περίπτωση  $\delta < 1$ . Το  $\delta$  ορίζεται συχνά ως ρυθμός προσέγγισης, ή ως εγγύηση απόδοσης, ή ως ρυθμός μελέτης χειρότερης περίπτωσης.

Η απόδοση του αλγορίθμου αποτελεί χαρακτηριστικό επιλογής τους αλγορίθμου. Παρ' όλα αυτά υπάρχει μεγάλη ποικιλία πολυωνυμικών αλγορίθμων, όπου μερικοί από αυτούς είναι αρκετά αποτελεσματικοί, αλλά με καμία πρακτική εφαρμογή.

### 4.1.3 Αλγόριθμοι Τοπικής Αναζήτησης

Κάθε αλγόριθμος τοπικής αναζήτησης έχει ως βάση του τη μέθοδο δοκιμής και σφάλματος, μία από τις αρχαιότερες μεθόδους βελτιστοποίησης. Είναι αποδεδειγμένο ότι, η τοπική αναζήτηση βρίσκει εφαρμογή με επιτυχία σε μία πληθώρα προβλημάτων βελτιστοποίησης. Πιο συγκεκριμένα, έστω πρόβλημα βελτιστοποίησης  $(F, c)$  όπου  $F$  είναι ένα εφικτό σύνολο και  $c$  το κόστος, επιλέγεται η γειτονιά στην οποία θα εφαρμοστεί η αναζήτηση για να προκύψει μία βελτιωμένη λύση. Όπως προαναφέρθηκε, κάθε αλγόριθμος τοπικής αναζήτησης δέχεται ως είσοδο μία αρχική λύση και επαναληπτικά προχωράει στη βελτίωση της. Έτσι, αν υπάρχει κάποιο σημείο  $s \in F$  που να έχει κόστος μικρότερο από αυτό της αρχικής λύσης, αυτή η λύση γίνεται η νέα είσοδος του αλγορίθμου. Η διαδικασία αναζήτησης συνεχίζεται έως ότου το σημείο που θα βρεθεί να είναι κάποιο τοπικό ελάχιστο και η λύση δε βελτιώνεται άλλο. Σε γενικές γραμμές, ένας αλγόριθμος τοπικής αναζήτησης ξεκινάει από μία εφικτή λύση  $t \in F$  και γίνεται χρήση μίας υπορουτίνας, ώστε στη γειτονιά της  $t$  να βρεθεί μία λύση με μικρότερο κόστος. Ο αλγόριθμος δεν ολοκληρώνεται, όσο προκύπτει βελτιωμένη λύση. Η διαδικασία σταματάει όταν βρεθεί τοπικό ελάχιστο. Η κίνηση του αλγορίθμου γίνεται σε γείτονες  $s'$  της αρχικής λύσης  $s$ . Κάθε γείτονα  $s'$  δημιουργείται μέσω ενός τελεστή κίνησης  $m$ , ο οποίος μεταβάλλει την αρχική λύση  $s$  ώστε να προκύψει η  $s'$ .

Μια γειτονιά χαρακτηρίζεται από την αρχή της τοπικότητας. Ως τοπικότητα ορίζεται η επίδραση που εμφανίζεται στη λύση, όταν γίνεται κίνηση στην

αναπαράσταση της λύσης. Μία γειτονιά έχει ισχυρή τοπικότητα όταν οι μικρές αλλαγές που γίνονται στην αναπαράσταση της λύσης είναι εμφανείς και στην ίδια τη λύση. Αντιθέτως, μία γειτονιά χαρακτηρίζεται από ασθενή τοπικότητα όταν εμφανίζονται μεγάλες αλλαγές στη λύση.

Συνεπώς, η γειτονιά ενός συνεχούς προβλήματος αποτελείται γύρω από την  $s$  με ακτίνα  $\epsilon$ . Στη περίπτωση ενός διακριτού προβλήματος ωστόσο, η γειτονιά αποτελείται από ένα σύνολο σημείων, όπου κάθε σημείο βρίσκεται σε απόσταση  $d < \epsilon$  με το  $\epsilon$  να είναι μία παράμετρος που εξαρτάται από τη δομή του προβλήματος και τη μέθοδο που έχουμε επιλέξει, για να κινηθούμε από το ένα σημείο στο άλλο.

Τέλος, η επιλογή της σωστής μεθόδου τοπικής αναζήτησης αποτελεί σημαντικό στοιχείο για την επιτυχία του αλγορίθμου. Κρίσιμο για την επιτυχία του αλγορίθμου είναι το στάδιο του σχεδιασμού του αλγορίθμου τοπικής αναζήτησης. Αρχικά, πρέπει να γίνει η σωστή επιλογή γειτονιάς λύσεων, έτσι ώστε να αποφευχθούν μη-εφικτές λύσεις, οι οποίες μπορεί να εισαχθούν στον αλγόριθμο και να οδηγήσουν σε λάθος αποτελέσματα. Ένας άλλος παράγοντας, που επηρεάζει σημαντικά την απόδοση του αλγορίθμου, είναι η ποιότητα της αρχικής λύσης. Αν ο αλγόριθμος ξεκινήσει στη γειτονιά μίας ποιοτικής λύσης, τότε οι πιθανότητες να βελτιώσει τη λύση ευκολότερα και σε μικρότερο χρόνο είναι περισσότερες. Ο τελευταίος σημαντικός παράγοντας, που έχει επίδραση στην απόδοση της τοπικής αναζήτησης, είναι η μέθοδος που θα χρησιμοποιηθεί για την βελτίωση της λύσης. Η πλειοψηφία των μεθόδων, που συναντώνται, παρουσιάζουν προβλήματα όσον αφορά τη ποιότητα των λύσεων καθώς, είναι πολύ πιθανόν να βρεθούν σε τοπικό ελάχιστο και να αδυνατούν να ξεφύγουν από αυτό.

Η επιλογή της γειτονιάς είναι σημαντική καθώς, κάθε γειτονιά αντιστοιχεί σε ένα τοπικό ελάχιστο. Σκοπός της μεθόδου είναι να βρεθεί το καλύτερο τοπικό ελάχιστο, δηλαδή η καλύτερη γειτονιά λύσεων, και να μπορέσει ο αλγόριθμος να οδηγηθεί στη καλύτερη λύση της γειτονιάς. Η αρχική λύση, που θα καθορίσει τη γειτονιά, δημιουργείται είτε τυχαία είτε με τη χρήση κάποιου αλγορίθμου απληστίας. Στη περίπτωση που το πρόβλημα είναι συνεχές, η αρχική λύση δημιουργείται καλύτερα χρησιμοποιώντας τυχαίο τρόπο, ενώ στη περίπτωση διακριτού προβλήματος, προτείνεται η χρήση κάποιου αλγορίθμου απληστίας. Από την άλλη, στη περίπτωση που είναι αναγκαία η δημιουργία πληθυσμού λύσεων, ώστε αυτές να βελτιωθούν επαναληπτικά με κάποια άλλη μέθοδο, οι λύσεις αυτές θα πρέπει να δημιουργηθούν είτε με τυχαίο τρόπο ή με χρήση πολλαπλών αλγορίθμων απληστίας.

Θα μπορούσε να γίνει διαχωρισμός των αλγορίθμων τοπικής αναζήτησης σε δύο κατηγορίες:

- Inter-route: Οι αλλαγές γίνονται αναμεσα σε διαφορετικές διαδρομές. Σε αυτή τη κατηγορία υπάρχει επίδραση σε περιορισμούς, που αφορούν τη διανυθείσα απόσταση, όσο και περιορισμοί που αφορούν τη χωρητικότητα.
- Intra-route: Οι αλλαγές παρουσιάζονται εντός μίας διαδρομής, που κατά πάσα πιθανότητα έχει προέλθει από την προηγούμενη κατηγορία αλγορίθμων. Σε αυτή την περίπτωση, όπως είναι λογικό, δεν εμφανίζεται επίδραση σε περιορισμούς που αφορούν τη χωρητικότητα.



Από τους πιο γνωστούς αλγορίθμους τοπικής αναζήτησης, οι οποίοι θα αναλυθούν στη συνέχεια, καθώς αποτελούν τους αλγόριθμους που χρησιμοποιήθηκαν στη τοπική αναζήτηση του μιμητικού, που αναπτύχθηκε για την επίλυση το προβλήματος που μας απασχολεί, είναι :

- 1-1 Ανταλλαγή (1-1 Exchange)
- 1-0 Επανατοποθέτηση(1-0 Relocate)

## 4.2 Μεθευρετικοί

Στη περίπτωση που ο αλγόριθμος μας κολλήσει σε κάποιο τοπικό ελάχιστο, το οποίο δεν είναι ικανοποιητικό ως λύση, γίνεται χρήση μεθευρετικού αλγορίθμου, ώστε να ξεμπλοκάρει η λύση. Διαχωρισμός ανάμεσα σε μεθευρετικούς αλγορίθμους μπορεί να γίνει με βάση τον αριθμό των λύσεων, που χρειάζονται για την αναζήτηση. Κάποιοι αλγόριθμοι χρησιμοποιούν μόνο μία λύση και η αναζήτηση γίνεται στη γειτονιά της, ενώ άλλοι χρειάζονται πληθυσμό λύσεων και η αναζήτηση γίνεται στο χώρο όλων των λύσεων. Συνδυασμών των δύο κατηγοριών μπορεί να υπάρξει. Αλγόριθμοι που χρησιμοποιούν μία γειτονιά, προχωρούν σε πιο εντατική αναζήτηση γύρω από τη λύση, εξερευνώντας όλο το μέγεθος της συγκεκριμένης περιοχής. Οι αλγόριθμοι, που λειτουργούν με πληθυσμό λύσεων ωστόσο, εστιάζουν την προσοχή τους σε μεγαλύτερο μέρος του χώρου λύσεων, λειτουργώντας διαχυτικά. Υβριδικές περιπτώσεις αλγορίθμων συνδυάζουν πλεονεκτήματα και από τις δύο κατηγορίες (tabu search).

Η λειτουργία ενός μεθευρετικού αλγορίθμου επικεντρώνεται γύρω από ένα τοπικό ελάχιστο, το οποίο έχει βρεθεί με κάποια διαδικασία τοπικής αναζήτησης. Επιπλέον, διαχωρισμός γίνεται στους αλγορίθμους, που χρησιμοποιούν μία λύση για την αναζήτηση, ανάλογα με τον τρόπο τον οποίο χρησιμοποιούν, για να αποφύγουν το τοπικό ελάχιστο. Συγκεκριμένα χωρίζονται σε :

- Σε διαδικασίες που ξεκινούν από διαφορετικές αρχικές λύσεις και λειτουργούν επαναληπτικά. Στην κατηγορία αυτή ανήκει η μέθοδος της διαδικασίας άπληστης τυχαιοποιημένης προσαρμοστικής αναζήτησης (Greedy Randomized Adaptive Search Procedure - GRASP) που θα περιγραφεί αναλυτικά στη συνέχεια.
- Αλγόριθμους που επιτρέπουν την κίνηση σε κοντινές γειτονίες, που βελτιώνουν τη λύση. Κινήσεις που δε βελτιώνουν τη λύση μπορούν να γίνουν δεκτές σε συγκεκριμένες συνθήκες. Με αυτό τον τρόπο κίνησης, ο αλγόριθμος μπορεί να ξεκολλήσει από το τοπικό ελάχιστο και να βρεθεί σε κάποιο γειτονικό τοπικό ελάχιστο, το οποίο να δίνει καλύτερη λύση από το προηγούμενο τοπικό ελάχιστο.
- Αλγόριθμους που μπορούν να αλλάξουν γειτονιά λύσεων. Τέτοιοι αλγόριθμοι περιλαμβάνουν αλγορίθμους, οι οποίοι όταν κολλήσουν σε κάποιο τοπικό ελάχιστο, αλλάζουν τον αλγόριθμο που χρησιμοποιούν, ώστε να γίνει αναζήτηση σε γειτονικά σημεία του χώρου λύσεων.
- Αλγόριθμους που μεταβάλλουν την αντικειμενική συνάρτηση ή άλλες παραμέτρους του προβλήματος. Δηλαδή αυτό το είδος αλγορίθμων σε

περιπτώσεις που πέφτει σε κάποιο τοπικό ελάχιστο, αλλάζει την αντικειμενική συνάρτηση ή τους περιορισμούς του προβλήματος.

Κινήσεις σε γειτονίες μη εφικτών λύσεων γίνεται πολλές φορές αποδεκτές στη συγκεκριμένη κατηγορία αλγορίθμων, και αυτό γιατί με αυτό τον τρόπο δίνεται η δυνατότητα στον αλγόριθμο να αποφύγει κάποια τοπικά ελάχιστα, και να προσεγγίσει γειτονίες με καλύτερα τοπικά ελάχιστα. Οι μεθευρετικοί αλγόριθμοι πολλές φορές προσομοιάζουν λειτουργίες που συναντώνται στη φύση. Τέτοιου είδους στοιχεία της φύσης, που χρησιμοποιούνται μεταφορικά και κυριολεκτικά σε έναν μεθευρετικό αλγόριθμο είναι ότι :

- ο Γίνονται περιορισμένες δοκιμές επαναληπτικά
- ο Αποτελούνται από έναν ή περισσότερους νευρώνες μόρια κ.λπ.
- ο Χρησιμοποιούν το μηχανισμό συνεργασίας ανταγωνισμού στη περίπτωση πολλών νευρώνων, μορίων κ.λπ.
- ο Κάνουν μετατροπές στις παραμέτρους των ευρετικών αλγορίθμων ή ακόμη και στις παραμέτρους αναπαράστασης του προβλήματος.

#### 4.3 Γενετικοί Αλγόριθμοι

Η έμπνευση για την ανάπτυξη των γενετικών αλγορίθμων προήρθε από τη διαδικασία εξέλιξης της φύσης. Έχουν ως βάση τους μία διαδικασία της βιολογίας, κατά την οποία οι καλύτεροι και οι νεότεροι επιβιώνουν κατά το πέρασμα του χρόνου. Για αυτό το λόγο, ανάμοια με τους περισσότερους ευρετικούς αλγορίθμους, ένας γενετικός αλγόριθμος αντλεί πληροφορία από έναν πληθυσμό λύσεων, που ονομάζεται άτομα (*individuals*). Οι γενετικοί αλγόριθμοι, αποτελούν μία στοχαστική διαδικασία και λειτουργούν επαναληπτικά. Το μέγεθος της γενιάς, δηλαδή του πληθυσμού λύσεων, διατηρείται σταθερό σε κάθε επανάληψη. Κύριος στόχος της διαδικασίας, είναι η σύνδεση των λύσεων, ώστε να προκύψει μία βελτιωμένη λύση. Αυτή η σύνδεση γίνεται με τη χρήση του δυαδικού τελεστή διασταύρωσης και του μοναδιαίου τελεστή μετάλλαξης. Κατά τη διασταύρωση των λύσεων, επιλέγονται δύο άτομα (*Individuals*) και ανταλλάσσουν τμήματα τους, με τρόπο που ορίζει ο τελεστής διασταύρωσης ώστε να προκύψουν άλλα δύο άτομα. Τα δύο αρχικά άτομα, που επιλέχθηκαν, ονομάζονται *γονείς* και τα άτομα που προέκυψαν μετά το ταίριασμα των λύσεων, *απόγονοι* (*offspring*) . Άτομα του πληθυσμού αντιστοιχούν σε λύσεις.

Η χρήση ενός γενετικού αλγορίθμου γίνεται για την βελτίωση του πληθυσμού λύσεων, με σκοπό την προσέγγιση της βέλτιστης λύσης και επιφέρει κάποια πλεονεκτήματα όπως :

- ο Είναι εξίσου αποτελεσματικοί σε προβλήματα με συνεχείς και διακριτές μεταβλητές.
- ο Δεν απαιτούν επιπλέον πληροφορία από παραγωγούς.
- ο Είναι ικανοί να αναζητήσουν παράλληλα στο μεγαλύτερο μέρος του πεδίου λύσεων.
- ο Είναι αποδοτικοί, ανεξάρτητα από το μέγεθος του προβλήματος, δηλαδή το πλήθος των μεταβλητών.
- ο Είναι δυνατή η εφαρμογή τους σε παράλληλα υπολογιστικά συστήματα.
- ο Ξεκollάνε με ευκολία από κάποιο τοπικό ελάχιστο.

- ο Δίνουν ως αποτέλεσμα πληθυσμό τοπικών βέλτιστων λύσεων.

Σε κάθε γενετικό αλγόριθμο, και γενικότερα σε κάθε εξελικτικό, συναντώνται κάποιοι βασικοί παράγοντες, οι οποίοι είναι :

- ο Η αντιστοιχία κάθε λύσης σε κάποιο χρωμόσωμα ή χρωμοσώματα μέσω κάποιας κωδικοποίησης.
- ο Μία συνάρτηση εκτίμησης της ποιότητας – καταλληλότητας του κάθε ατόμου του πληθυσμού.
- ο Αρχικοποίηση του πληθυσμού.
- ο Καθορισμός των τελεστών.
- ο Επιλογή τελεστών αναπαραγωγής.

Κάθε άτομο (λύση) του πληθυσμού αναπαρίσταται από ένα χρωμόσωμα ή γονίδιο. Τα χαρακτηριστικά των ατόμων, που αναπαρίστανται από τα χρωμοσώματα αναφέρονται στις μεταβλητές του προβλήματος βελτιστοποίησης. Τα γονίδια αποτελούν τη μικρότερη ποσότητα πληροφορίας στο πρόβλημα και η ανάθεση μίας τιμής από την επιτρεπόμενη ζώνη της αντίστοιχης μεταβλητής, ορίζεται ως αλληλομορφή (allele).

Ο καθορισμός της κατάλληλης αναπαράστασης της υποψήφιας λύσης, αποτελεί ένα από τα σημαντικότερα μέρη στο σχεδιασμό ενός γενετικού αλγορίθμου. Η αναπαράσταση μπορεί να γίνει ως εξής:

- ο Με διάνυσμα δυαδικών τιμών καθορισμένου μήκους, όπου το διάνυσμα αποτελείται από μηδενικά και άσσους.

$\Delta 1 | 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1$

- ο Σε μορφή διαδρομής.

$\Delta 2 | 1 \ 4 \ 2 \ 6 \ 10 \ 8 \ 4 \ 7 \ 3 \ 5$

- ο Αναπαράσταση με πραγματικούς αριθμούς.

$\Delta 3 | 0,1 \ 0,2 \ 0,24 \ 0,68 \ 0,55 \ 0,7 \ 0,40 \ 0,9 \ 0,41 \ 0,35$

Το μέγεθος του πληθυσμού μπορεί να επηρεάσει την απόδοση του αλγορίθμου. Πολύ μεγάλο μέγεθος πληθυσμού θα αυξήσει το χρόνο σύγκλισης του αλγορίθμου και η αποτελεσματικότητα της αναζήτησης θα μειωθεί. Από την άλλη, αν ο πληθυσμός είναι μικρός, δε θα είναι αντιπροσωπευτικός. Όσον αφορά την αρχικοποίηση του πληθυσμού λύσεων, ένας από τους συνηθέστερους τρόπους αρχικοποίησης είναι επιλογή των τιμών, για κάθε μεταβλητή του ατόμου, τυχαία. Με αυτό τον τρόπο, παίρνουμε υποψήφιας λύσεις σε όλο το εύρος του χώρου λύσεων, καθώς ο πληθυσμός είναι ομοιόμορφα κατανομημένος. Άλλοι πιθανοί τρόποι αρχικοποίησης είναι η μέθοδος της επαναληπτικής διάχυσης των λύσεων (sequential diversification), όπου και σε αυτή τη περίπτωση ο πληθυσμός είναι ομοιόμορφα κατανομημένος στο χώρο λύσεων, αλλά και η χρήση κάποιου ευρετικού ή μεθευρετικού αλγορίθμου. Η τελευταία μέθοδος, μπορεί να αυξάνει αρκετά την ποιότητα των αρχικών λύσεων, αλλά ενδέχεται να συγκεντρώσει τις υποψήφιας λύσεις, σε συγκεκριμένα σημεία του χώρου λύσεων, αφήνοντας περιοχές ανεκμετάλλευτες. Για αυτό το λόγο πρέπει να γίνει σωστή επιλογή αλγορίθμου.

Η ποιότητα του κάθε ατόμου (λύσης) υπολογίζεται με τη χρήση κάποια συνάρτησης καταλληλότητας (fitness function (f)), η οποία εξαρτάται από το πρόβλημα βελτιστοποίησης, στο οποίο εφαρμόζεται ο αλγόριθμος. Συχνά, αντιπροσωπεύεται από κάποια συνάρτηση κόστους και αναλόγως, αν το πρόβλημα είναι μεγιστοποίησης ή ελαχιστοποίησης, διαχωρίζονται οι ποιοτικές και μη ποιοτικές λύσεις. Είναι δυνατή η ύπαρξη παραπάνω από μία αντικειμενικών συναρτήσεων, όπου σε αυτή τη περίπτωση η συνάρτηση καταλληλότητας, μπορεί να αποτελείται από το άθροισμα των επιμέρους συναρτήσεων με τη χρήση κάποιων βαρών. Τέλος, υφίστανται και χρονικά εξαρτώμενες αντικειμενικές συναρτήσεις, που βρίσκουν εφαρμογή σε προβλήματα δυναμικής βελτιστοποίησης, όπου η λύση μεταβάλλεται χρονικά.

Όσον αφορά η επιλογή των γονέων, δηλαδή των λύσεων που θα διασταυρωθούν, για να προκύψει η νέα γενιά λύσεων, αυτή μπορεί να γίνει τυχαία, αλλά η απόδοση του συγκεκριμένου τρόπου είναι περιορισμένη. Η απλούστερη μέθοδος, η διασταύρωση των λύσεων ανά δύο, από τη κορυφή προς το βάθος, ενώ η πιο εξελιγμένη είναι ο κανόνας της ρουλέτας (Roulette wheel selection). Στη τελευταία, οι λύσεις με τη καλύτερη ποιότητα έχουν και περισσότερες πιθανότητες να επιλεγούν για γονείς. Τέλος, μία άλλη διαδικασία επιλογής είναι εκείνη της σύγκρισης ποιότητας μία ομάδας υποψηφίων γονέων και τη επιλογής των ποιοτικότερων λύσεων. Η καλύτερη λύση της τυχαίας επιλεγμένης ομάδας λύσεων, αποτελεί τον ένα γονέα και η διαδικασία επαναλαμβάνεται για τον δεύτερο.

Μετά τη επιλογή των γονέων, η δημιουργία των απογόνων πραγματοποιείται είτε με τη διασταύρωση (crossover) των λύσεων, είτε με μετάλλαξη (Mutation) τους. Η διασταύρωση δημιουργεί δύο ή περισσότερους απογόνους από το γενετικό υλικό των γονέων, κάνοντας χρήση κάποιου τελεστή διασταύρωσης. Οι τελεστές διασταύρωσης χωρίζονται σε τρεις κατηγορίες. Ο τελεστής με τον οποίο ο απόγονος δημιουργείται από ένα μόνο γονέα (asexual), ο τελεστής που με χρήση του οι απόγονοι δημιουργούνται από δύο γονείς και εκείνος όπου χρησιμοποιούνται περισσότεροι από δύο γονείς, για τη δημιουργία δύο ή περισσότερων απογόνων. Οι τρεις πιο γνωστοί τελεστές διασταύρωσης είναι, ο τελεστής διασταύρωσης ενός σημείου (1point crossover) , ο τελεστής δύο σημείων (2point crossover) και ο τελεστής ομοιόμορφης διασταύρωσης (uniform crossover). Στη διασταύρωση ενός σημείου επιλέγεται ένα σημείο του διανύσματος, στο οποίο θα γίνει η ανταλλαγή των στοιχείων, αντίστοιχα, στη διασταύρωση δύο σημείων επιλέγονται δύο σημεία, για να γίνει η διασταύρωση. Στην ομοιόμορφη διασταύρωση επιλέγονται τυχαία τα γονίδια, που θα ανταλλαχθούν μεταξύ των γονέων . Επιπλέον τελεστές έχουν αναπτυχθεί για τη περίπτωση που έχουμε κωδικοποίηση με χρήση πραγματικών αριθμών, όπου εκεί μπορούν να χρησιμοποιηθούν τελεστές, όπως ο γραμμικός τελεστής διασταύρωσης (linear crossover operator) και ο αριθμητικός τελεστής διασταύρωσης (arithmetic crossover operator). Ο τρόπος λειτουργίας των τελεστών παρουσιάζεται συνοπτικά στο πίνακα 1.

Πίνακας 1

| Τελεστής                | Γονείς (P)                                                | Απόγονοι (O)                                              |
|-------------------------|-----------------------------------------------------------|-----------------------------------------------------------|
| <b>1point crossover</b> | P1: 0.35 0.1 0.7    <b>0.74 0.92 0.31 0.22</b>            | O1: 0.35 0.1 0.7    <b>0.98 0.13 0.26 0.84</b>            |
|                         | P2: 0.12 0.15 0.55    <b>0.98 0.13 0.26 0.84</b>          | O2: 0.12 0.15 0.55    <b>0.74 0.92 0.31 0.22</b>          |
| <b>2point crossover</b> | P1: 0.35 0.1    <b>0.7 0.74 0.92</b>    0.31 0.22         | O1: 0.35 0.1    <b>0.55 0.98 0.13</b>    0.31 0.22        |
|                         | P2: 0.12 0.15   <b>0.55 0.98 0.13</b>    0.26 0.84        | O2: 0.12 0.15    <b>0.7 0.74 0.92</b>   0.26 0.84         |
|                         | P1: <b>0.35</b> 0.1 0.7 <b>0.74</b> 0.92 <b>0.31</b> 0.22 | O1: <b>0.12</b> 0.1 0.7 <b>0.98</b> 0.92 <b>0.26</b> 0.22 |

|                                      |                                                             |                                                             |
|--------------------------------------|-------------------------------------------------------------|-------------------------------------------------------------|
| <b>uniform crossover</b>             | P2: <b>0.12</b> 0.15 0.55 <b>0.98</b> 0.13 <b>0.26</b> 0.84 | O2: <b>0.35</b> 0.15 0.55 <b>0.74</b> 0.13 <b>0.31</b> 0.84 |
| <b>Linear crossover operator</b>     | <b>P1:</b> 0.35 0.1 0.7    0.74 0.92 0.31 0.22              | <b>O1(i) = P1(i) + P2(i)</b>                                |
|                                      | <b>P2:</b> 0.12 0.15 0.55    0.98 0.13 0.26 0.84            | <b>O2(i) = 2.7 P1(i) + 3.14 P2(i)</b>                       |
| <b>Arithmetic Crossover operator</b> | <b>P1:</b> 0.35 0.1 0.7    0.74 0.92 0.31 0.22              | <b>O1(i) = (1-γ)P1(i) + γP2(i)</b>                          |
|                                      | <b>P2:</b> 0.12 0.15 0.55    0.98 0.13 0.26 0.84            | <b>O2(i) = γP1(i) + (1-γ)P2(i)</b>                          |

Εισαγωγή νέου γενετικού υλικού στο πληθυσμό είναι δυνατόν να γίνει με τη χρήση κάποιου τελεστή μετάλλαξης. Η επιλογή των γονιδίων, που θα αλλάξουν, γίνεται τυχαία και η εφαρμογή του τελεστή σε αυτά τα γονίδια πρέπει να γίνεται με προσοχή, καθώς υπάρχει κίνδυνος μείωσης της ποιότητας των λύσεων. Αυτό γίνεται γιατί είναι πιθανόν να αλλοιωθούν ποιοτικές λύσεις, για αυτό το λόγο, η μετάλλαξη εφαρμόζεται σε μικρές περιοχές του πληθυσμού. Δύο κλασικές εκδοχές τελεστών μετάλλαξης είναι, ο ομοιόμορφος τελεστής, όπου επιλέγονται τυχαία τα γονίδια, στα οποία θα γίνουν οι αλλαγές και η εντός σειράς μετάλλαξη, όπου μόνο δύο γονίδια επιλέγονται. Σε γενικές γραμμές, κατά τη σχεδίαση ενός τελεστή μετάλλαξης θα πρέπει να δίνεται έμφαση σε σημεία όπως:

- Να επιτρέπει τη κίνηση των ατόμων σε όλο το χώρο λύσεων
- Οι λύσεις που θα παράγονται να είναι εφικτές
- Θα πρέπει να κάνει μικρές αλλαγές στη λύση, χωρίς να την αλλοιώνει εντελώς

Στη συνέχεια μετά τις διαδικασίες διασταύρωσης και μετάλλαξης, θα πρέπει να γίνει η επιλογή της νέας γενιάς λύσεων, επιλέγοντας ανάμεσα στους γονείς και στους απογόνους. Αυτή η επιλογή μπορεί να γίνει με πολλούς τρόπους, αλλά σε κάθε περίπτωση, θα πρέπει να διατηρείται η ποικιλομορφία των χρωμοσωμάτων στα άτομα του πληθυσμού, έτσι ώστε να αποφευχθεί το κόλλημα του αλγορίθμου σε κάποιο μη ποιοτικό τοπικό ελάχιστο. Κάποιες χαρακτηριστικές διαδικασίες επιλογής της γενιάς είναι οι εξής :

- Η αντικατάσταση όλων των γονέων από τους απογόνους (Replace oldest)
- Μία αποδοτικότερη μέθοδος είναι, η κατάταξη όλων των λύσεων (γονείς κ' απογόνους) βάσει της ποιότητας τους, από την συνάρτηση καταλληλότητάς και εν συνεχεία, η επιλογή των καλύτερων.
- Αντικατάσταση των χειρότερων λύσεων του πληθυσμού με ποιοτικούς απογόνους (Replace the worst)
- Η τυχαία αντικατάσταση ατόμων του πληθυσμού από απογόνους.
- Άλλη μέθοδος επιλογής πληθυσμού είναι, η τυχαία επιλογή ατόμων και η αντικατάσταση των χειρότερων με απογόνους (kill tournament).
- Τέλος, είναι δυνατή η επιλογή ενός συνόλου «εκλεκτών» λύσεων, βάσει της τιμής που παίρνουν από την συνάρτηση καταλληλότητάς, ώστε να εξαιρεθούν από την επιλογή, να μην αντικαθίστανται και να προχωρούν πάντα από γενιά σε γενιά.

Ο γενετικός τερματίζει στις περιπτώσεις που, έχει φτάσει το μέγιστο όριο επαναλήψεων, έχει βρεθεί ποιοτικά αποδεκτή λύση, δεν παρατηρείται κάποια αλλαγή στον πληθυσμό ή δεν υπάρχει βελτίωση στις λύσεις του πληθυσμού, για συγκεκριμένο αριθμό επαναλήψεων.

## 4.4 Μιμητικοί Αλγόριθμοι

Ένας μιμητικός ή υβριδικός γενετικός αλγόριθμος αποτελείται από έναν γενετικό αλγόριθμο και κάποιον αλγόριθμο τοπικής αναζήτησης. Με αυτό τον τρόπο, δίνεται η δυνατότητα στον αλγόριθμο να χρησιμοποιήσει μια διαδικασία ολικής αναζήτησης, που παρέχεται από τον γενετικό, σε συνδυασμό με μία εντατικοποιημένη αναζήτηση από τη τοπική αναζήτηση. Έτσι, ένας μιμητικός αλγόριθμος, εκμεταλλευόμενος τη δύναμη ενός γενετικού αλγορίθμου και την ταχύτητα της τοπικής αναζήτησης, έχει αυξημένες ικανότητες διασποράς της αναζήτησης, σε όλο το χώρο λύσεων όσο και δυνατότητα εμβάθυνσης της αναζήτησης, σε συγκεκριμένα σημεία του χώρου λύσεων, που βρίσκονται καλές λύσεις. Συγκεκριμένα, ο γενετικός βοηθάει στον εντοπισμό της περιοχής, όπου είναι πιθανόν να βρίσκεται το βέλτιστο και στη συνέχεια με τη χρήση της τοπικής αναζήτησης, προσεγγίζει, όσο είναι δυνατόν, αυτό το βέλτιστο.

Ο συνδυασμός της τοπικής αναζήτησης με το γενετικό, μπορεί να γίνει με διάφορους τρόπους, αναλόγως το πρόβλημα και τους αλγορίθμους, που θα χρησιμοποιήσουμε. Κάποιοι από αυτούς είναι :

- Μετά το τερματισμό του γενετικού, να πάρουμε τη καλύτερη λύση, που μας επιστρέφει, και σε αυτή να εφαρμόσουμε κάποια μέθοδο τοπικής αναζήτησης, ελπίζοντας πως ο γενετικός μας έχει δώσει μία καλή περιοχή λύσεων.
- Να εκτελέσουμε πρώτα την τοπική αναζήτηση στα δεδομένα και τις αρχικές λύσεις, που θα μας δώσει, και να τις χρησιμοποιήσουμε στο γενετικό μας αλγόριθμο.
- Είναι δυνατόν να παρέχουμε στο γενετικό, συνεχώς, λύσεις, οι οποίες θα προέρχονται από μία παράλληλη τοπική αναζήτηση, που θα πραγματοποιείται σε κάθε επανάληψη του γενετικού.
- Να εκτελέσουμε για συγκεκριμένες επαναλήψεις ένα γενετικό αλγόριθμο και στη συνέχεια, τις καλύτερες λύσεις ή τις ελίτ να τις τροφοδοτήσουμε σε κάποια διαδικασία τοπικής αναζήτησης.
- Τέλος, ένας άλλος τρόπος είναι σε κάθε επανάληψη του γενετικού αλγορίθμου να εφαρμόζουμε στη καλύτερη λύση ή στις ελίτ λύσεις ή ακόμη και σε όλες ένα ή περισσότερους αλγορίθμους τοπικής αναζήτησης. Στη περίπτωση που εφαρμόζουμε τοπική αναζήτηση σε κάθε λύση, αυξάνουμε σημαντικά την απόδοση του αλγορίθμου, αλλά αυτό μπορεί να επηρεάσει σε μεγάλο βαθμό το χρόνο εκτέλεσης του αλγορίθμου, ειδικά σε προβλήματα μεγάλου όγκου .
- Εκτός αυτών είναι δυνατόν οι παραπάνω περιπτώσεις να συνδυαστούν.

Η αρχή πίσω από ένα μιμητικό αλγόριθμο είναι ότι, ένα άτομο(λύση) είναι πιθανόν να εξελιχθεί κατά τη διάρκεια της ζωής του. Για αυτό το λόγο ένας μιμητικός αποτελείται από έναν γενετικό ή εξελικτικό αλγόριθμο και έναν άλλο αλγόριθμο ή τεχνική που βοηθάει τον γενετικό στη προσέγγιση καλύτερης λύσης. Συνεπώς, οι κατηγορίες μιμητικών αλγορίθμων, που μπορούν να προκύψουν, είναι οι εξής:

- Συνδυασμός δύο εξελικτικών αλγορίθμων.
- Η χρήση γενετικού με νευρωνικά δίκτυα.
- Ο συνδυασμός γενετικών αλγορίθμων με ασαφή λογική.

- Να χρησιμοποιηθεί ένας γενετικός αλγόριθμος με αλγόριθμους νοημοσύνης σμήνους (αλγόριθμο βελτιστοποίησης σμήνους σωματιδίων, αλγόριθμο βελτιστοποίησης αποικίας μυρμηγκιών).

Τέλος, η πιο συνηθισμένη εκδοχή μιμητικού αλγορίθμου είναι, ο συνδυασμός ενός γενετικού με κάποιο απλό ευρετικό, με αλγόριθμους τοπικής αναζήτησης και μεθευρετικούς αλγόριθμους (αλγόριθμο περιορισμένης αναζήτησης, αλγόριθμο προσομοιωμένης ανόπτησης).

## 5. Τρόπος επίλυσης του προβλήματος

### 5.1 Εισαγωγή

Όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, το πρόβλημα με το οποίο ασχολείται η παρούσα εργασία, είναι το πρόβλημα δρομολόγησης και χρονικού προγραμματισμού φορητών πλοίων βιομηχανικού φορτίου και ελευθέρων πλοίων ξηρού φορτίου. Συγκεκριμένα, το πρόβλημα αντιστοιχεί σε μία εξελιγμένη μορφή ενός VRP προβλήματος, με αυστηρά χρονικά παράθυρα, χρήση ετερογενούς στόλου οχημάτων, με πολλαπλά σημεία εκκίνησης οχημάτων, χρόνο εξυπηρέτησης σε κάθε πελάτη και χρήση μεταβλητής «τιμωρίας» στην αντικειμενική συνάρτηση κόστους για κάθε πελάτη που δε εξυπηρετείται.

Συνεπώς, η λύση του προβλήματος, στοχεύει στη δημιουργία μίας εφικτής διαδρομής για κάθε πλοίο του προβλήματος, έτσι ώστε να εξυπηρετούνται όσο το δυνατόν περισσότερα εμπορεύματα, με το ελάχιστο λειτουργικό κόστος. Η λύση λοιπόν, αποτελείται από έναν αριθμό αλληλουχιών κόμβων, ίσο με τον αριθμό των πλοίων που είναι διαθέσιμα για μεταφορά. Κάθε κόμβος αντιστοιχεί σε κάποιο λιμάνι και ο πρώτος κόμβος κάθε διαδρομής αντιστοιχεί στο σημείο εκκίνησης του πλοίου, στο οποίο αντιστοιχεί η συγκεκριμένη διαδρομή.

Το συγκεκριμένο πρόβλημα επιλύθηκε με τη χρήση μίας διαδικασίας άπληστης τυχαιοποιημένης προσαρμοστικής αναζήτησης (Greedy Randomized Adaptive Search Procedure - GRASP) για τη δημιουργία των αρχικών λύσεων και ενός μιμητικού αλγορίθμου για τη βελτίωσή τους. Ο προγραμματισμός των αλγορίθμων έγινε στο περιβάλλον της Matlab.



## 5.2 Αρχικοποίηση μεταβλητών – Προετοιμασία Δεδομένων

Η εφαρμογή των αλγορίθμων, έγινε σε ένα σύνολο έτοιμων παραδειγμάτων, εμπνευσμένα από πραγματικά προβλήματα. Τα δεδομένα για κάθε παράδειγμα εμφανίζονται σε πέντε πίνακες, των οποίων το μέγεθος εξαρτάται από το μέγεθος του παραδείγματος. Το μέγεθος αυτό κυμαίνεται από 8 έως 130 εμπορεύματα. Οι αλγόριθμοι εφαρμόστηκαν σε μία ποικιλία χαρακτηριστικών παραδειγμάτων και για αυτό το λόγο ήταν αναγκαία η μεταφορά των πινάκων από αρχείο .txt σε φύλλα excel, έτσι ώστε να είναι εύκολη η εναλλαγή παραδειγμάτων, χωρίς να απαιτείται η συνεχής φόρτωση των δεδομένων στο πρόγραμμα. Οι πέντε πίνακες των παραδειγμάτων παρουσιάζουν τα εξής δεδομένα. Τα πλοία και τα εμπορεύματά εμφανίζονται ως ακέραιοι αριθμοί από το 1 έως το σύνολο διαθέσιμων πλοίων και εμπορευμάτων αντίστοιχα. Στην αρχή του αρχείου υπάρχει ο πίνακας, που αντιστοιχεί τα διαθέσιμα πλοία με τα λιμάνια εκκίνησης τους, τους χρόνους που θα είναι διαθέσιμα για μεταφορά και τη χωρητικότητα τους, που στη προκειμένη περίπτωση, είναι σταθερή και ίση με 100. Στη συνέχεια, συναντάται ο πίνακας που αντιστοιχεί κάθε πλοίο με τα εμπορεύματα που μπορεί να μεταφέρει, σύμφωνα με το περιορισμό συμβατότητας πλοίου - φορτίου. Ο επόμενος κατά σειρά πίνακας, είναι εκείνος, στον οποίο εμφανίζονται για κάθε εμπόρευμα, τα λιμάνια παραλαβής και παράδοσης, το μέγεθος του φορτίου, το οποίο είναι και αυτό σταθερό και ίσο με 100, το κόστος στη περίπτωση μη εξυπηρέτησης του συγκεκριμένου εμπορεύματος, το χρονικό παράθυρο του λιμανιού παραλαβής και στο τέλος το χρονικό παράθυρο για το λιμάνι παράδοσης. Στον επόμενο πίνακα, υπάρχει για κάθε πλοίο, η διάρκεια ταξιδιού από και προς οποιοδήποτε λιμάνι, εφόσον είναι εφικτή αυτή η μετακίνηση για το συγκεκριμένο πλοίο, αλλά και το αντίστοιχο κόστος της μετακίνησης στη 4<sup>η</sup> στήλη του πίνακα. Στον τελευταίο πίνακα του αρχείου, βρίσκονται για κάθε πλοίο και εμπόρευμα οι χρόνοι εξυπηρέτησης των λιμανιών παραλαβής και παράδοσης και τα αντίστοιχα κόστη για τις υπηρεσίες του εκάστοτε λιμανιού.

Αυτοί οι πίνακες μεταφέρθηκαν σε ξεχωριστά φύλλα excel, από τα οποία έγινε η αρχικοποίηση των μεταβλητών. Αρχικά, χρειάστηκε η αρχικοποίηση οκτώ μεταβλητών, πέντε για τα δεδομένα που προαναφέρθηκαν και άλλες τρεις για τον αριθμό των διαθέσιμων πλοίων, το σύνολο των εμπορευμάτων προς μεταφορά και το σύνολο των λιμανιών του προβλήματος. Επιπλέον, οι βοηθητικές μεταβλητές, που αρχικοποιήθηκαν, αφορούν το κόστος και το χρόνο σε κάθε σημείο της διαδρομής ενός πλοίου, το συνολικό χρόνο λειτουργίας κάθε πλοίου και τους προσιτούς κόμβους, από κάποιο συγκεκριμένο κόμβο – λιμάνι. Ο μέγιστος χρόνος λειτουργίας των πλοίων, είναι το μεγαλύτερο χρονικό παράθυρο στα δεδομένα, ενώ με τη μεταβλητή προσιτών κόμβων δημιουργείται μία λίστα λιμανιών, στα οποία μπορεί να μεταβεί το εκάστοτε πλοίο ανάλογα με το περιορισμό συμβατότητας και το λιμάνι στο οποίο βρίσκεται εκείνη τη στιγμή. Με άλλα λόγια, έστω ένα πλοίο που ξεκινάει τη διαδρομή του από το λιμάνι εκκίνησης και θέλει να κατευθυνθεί προς το πρώτο λιμάνι παραλαβής. Η επιλογή του λιμανιού, στο οποίο θα κινηθεί, θα γίνει μέσα από τη λίστα προσιτών κόμβων, ανάλογα με τα διαθέσιμα φορτία και το περιορισμό συμβατότητας. Αυτή η μεταβλητή υπήρξε χρήσιμη, στη δημιουργία αρχικής λύση με τη GRASP. Τέλος, στην αρχή του προγράμματος αρχικοποιείται και η μεταβλητή, στην οποία θα εισαχθεί η αρχική λύση του προβλήματος. Αρχικοποίηση μεταβλητών γίνεται σε όλη την έκταση του



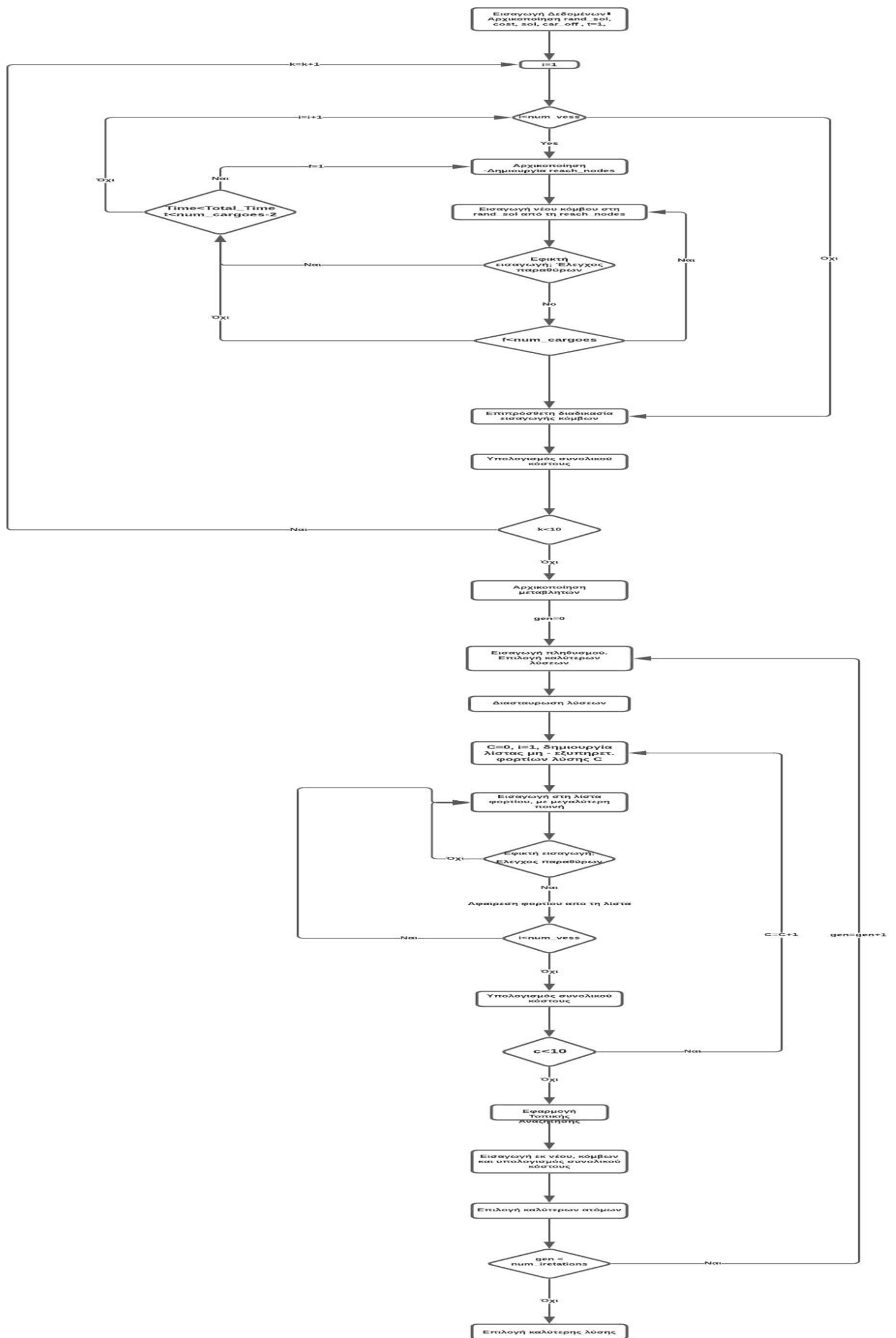
προγράμματος αλλά αυτές είτε θα έχουν παρόμοιες λειτουργίες με αυτές που αναφέρθηκαν ή θα αποτελούν κάποιου είδους μετρητή ή κάποια μεταβλητή προσωρινής αποθήκευσης.

Αξίζει να αναφερθεί πως, δεν ήταν αναγκαία η προ επεξεργασία των δεδομένων, ούτε ο υπολογισμός επιπλέον δεδομένων, εκτός από την επεξεργασία κάποιων κενών και αρνητικών στοιχείων στους πίνακες. Τα αρνητικά στοιχεία είναι τοποθετημένα σε κόστη/χρόνους μετακινήσεων και μετακινήσεις, που δεν είναι εφικτές από το συγκεκριμένο πλοίο, στο οποίο αναφέρεται εκείνη η γραμμή.

## Διάγραμμα Ροής Αλγορίθμου

Στη συνέχεια θα παρουσιαστεί το διάγραμμα ροής της συνολικής μεθόδου που ακολουθείται για την επίλυση του συγκεκριμένου προβλήματος, από την αρχικοποίηση των μεταβλητών, την δημιουργία της αρχικής λύσης και τέλος της βελτίωσης της με το μιμητικό αλγόριθμο. Οι μεταβλητές που παρουσιάζονται αντιστοιχούν σε :

- Rand\_sol αντιστοιχεί στο πίνακα που αποθηκεύεται κάθε αρχική λύση.
- Reach\_nodes είναι ο πίνακας προσιτών κόμβων από τον οποίο επιλέγονται οι υποψήφιοι για εισαγωγή κόμβοι
- Cost είναι ο πίνακας του κόστους κάθε αρχικής λύσης
- Sol είναι ο πίνακας αποθήκευσης του πληθυσμού λύσεων
- Sol\_cost αποτελεί το πίνακα κόστους κάθε μεταβλητής που ανήκει στο πίνακα.
- Car off η λίστα εμπορευμάτων που δεν εξυπηρετήθηκαν.
- i είναι ο μετρητής των εκάστοτε πινάκων λύσεων και αντιπροσωπεύει το πλοίο στο οποίο αναφερόμαστε.
- Num\_vess είναι ο συνολικός αριθμός διαθέσιμων πλοίων.
- Num\_cargoes είναι ο αριθμός εμπορευμάτων προς μεταφορά.
- F και K αποτελούν μετρητές.



### 5.3 Δημιουργία Αρχικής Λύσης με χρήση διαδικασίας άπληστης τυχαιοποιημένης προσαρμοστικής αναζήτησης

Αφού έχουμε αρχικοποιήσει όλες τις αναγκαίες μεταβλητές και έχουμε εισάγει τα απαραίτητα δεδομένα, μπορούμε να προχωρήσουμε στη δημιουργία της αρχικής εφικτής λύσης. Όπως έχει προαναφερθεί σε αυτό το στάδιο του προγράμματος έγινε χρήση της μεθόδου GRASP. Στο γενετικό αλγόριθμο που θα ακολουθήσει θα πρέπει να εισαχθεί πληθυσμός λύσεων και για αυτό το λόγο χρειάζεται ένας αλγόριθμος που λειτουργεί επαναληπτικά και να μας δίνει τη δυνατότητα να έχουμε ποικιλία διαφορετικών λύσεων χωρίς να προκύπτουν συγκεκριμένες λύσεις σε κάθε επανάληψη. Με αυτό τον τρόπο, μπορούμε να εξερευνήσουμε μεγαλύτερη έκταση του πεδίου λύσεων δίνοντας έτσι τη δυνατότητα στο μιμητικό να λειτουργήσει αποδοτικά. Για αυτό το λόγο χρησιμοποιήθηκε μία διαδικασία άπληστης τυχαιοποιημένης προσαρμοστικής αναζήτησης (GRASP)


#### 5.3.1 GRASP

Η διαδικασία άπληστης τυχαιοποιημένης προσαρμοστικής αναζήτησης αποτελεί μια επαναληπτική διαδικασία για την εύρεση προσεγγιστικών λύσεων σε προβλήματα βελτιστοποίησης. Η λειτουργία του συγκεκριμένου αλγορίθμου χωρίζεται σε δύο φάσεις. Στη φάση κατασκευής μίας αρχικής λύσης, κατά την οποία γίνεται χρήση μίας τυχαιοποιημένης συνάρτησης απληστίας ώστε να κατασκευαστεί η αρχική λύση και στη φάση της τοπικής αναζήτησης, η οποία βελτιώνει τη λύση. Στη φάση της κατασκευής της λύσης ο αλγόριθμος λειτουργεί επαναληπτικά προσθέτοντας κάθε φορά ένα κόμβο στη λύση μέχρι να ολοκληρωθεί. Η επιλογή του εκάστοτε στοιχείου γίνεται από μία περιορισμένη λίστα υποψήφιων κόμβων, που ονομάζεται λίστα περιορισμού των υποψηφίων (Restricted Candidate List). Η τελική επιλογή του κόμβου γίνεται τυχαία από τη λίστα. Ο συγκεκριμένος αλγόριθμος αποτελεί έναν προσαρμοστικό αλγόριθμο καθώς η επιλογή του κόμβου προσαρμόζεται έτσι ώστε να επιλέξει κόμβους που δεν έχουν επιλεγεί ήδη. Κρίσιμο κόμματι της διαδικασίας αποτελεί η κατασκευή της περιορισμένης λίστας καθώς από εκεί ελέγχεται η διασπορά των αρχικών λύσεων στο χώρο. Στη περίπτωση που η λίστα είναι αρκετά περιορισμένη, χάνεται η δυναμικότητα του αλγορίθμου και οι λύσεις που προκύπτουν είναι σχεδόν ίδιες. Από την άλλη μία μεγάλη λίστα επιλογής ενισχύει το στοιχείο της τυχαιότητας του αλγορίθμου κάνοντας τον να επιλέγει πλέον τυχαία τον επόμενο κόμβο. Στη κλασσική εκδοχή του αλγορίθμου μετά τη κατασκευή της αρχικής λύσης ορίζεται μία συνάρτηση που θα κάνει αναζήτηση στη γειτονιά της αρχικής λύσης. Ανάλογα με το πρόβλημα ακολουθούνται διαφορετικές διαδικασίες κατασκευής της λίστα, επιλογής υποψηφίων και τοπικής αναζήτησης.

Στη παρούσα περίπτωση, η λίστα περιορισμού των υποψηφίων αποτελείται από τη λίστα προσιτών κόμβων. Οι κόμβοι που μπήκαν στη συγκεκριμένη λίστα είναι λιμάνια παραλαβής εμπορευμάτων που είναι συμβατά με το εκάστοτε πλοίο. Η λίστα ταξινομείται από το κοντινότερο λιμάνι (Σε διάρκεια ταξιδιού) προς το πιο απομακρυσμένο και στη συνέχεια επιλέγεται τυχαία ένα από τα δέκα κοντινότερα


λιμάνια. Αν η λίστα δεν περιέχει δέκα υποψήφια λιμάνια τότε επιλέγουμε από το σύνολο της λίστας. Η διαδικασία επαναλαμβάνεται αλλάζοντας κάθε φορά το λιμάνι από το οποίο συνεχίζει τη πορεία του το πλοίο και κατ' επέκταση μεταβάλλοντας τη λίστα των υποψήφιων λιμανιών παραλαβής. Δηλαδή, κάθε φορά που εισάγεται ένα λιμάνι παραλαβής μπαίνει αυτομάτως δίπλα του το αντίστοιχο λιμάνι παράδοσης, το οποίο στη συνέχεια λαμβάνεται ως σημείο εκκίνησης για τη μέτρηση των νέων αποστάσεων της λίστας και της εισαγωγή των νέων κόμβων σε αυτήν. Αυτό συμβαίνει έως ότου δεν υπάρχει διαθέσιμο λιμάνι να εισαχθεί ή έχει ολοκληρωθεί ο αριθμός επαναλήψεων. Το ίδιο γίνεται και για υπόλοιπα διαθέσιμα πλοία. Για κάθε εισαγωγή ζευγαριού κόμβων ελέγχονται τα αντίστοιχα χρονικά παράθυρα. Ζευγάρια λιμανιών που εισάγονται σε κάποιο διαδρομή προφανώς «αφαιρούνται» από τη λίστα. Μετά τη ολοκλήρωση της διαδικασίας για όλα τα διαθέσιμα πλοία έχει ολοκληρωθεί η δημιουργία μίας αρχικής λύσης και η μορφή της παρουσιάζεται στη εικόνα 2. Συνεπώς, η διαδικασία επαναλαμβάνεται συνολικά για 10 επαναλήψεις έτσι ώστε να προκύψει ο ζητούμενος πληθυσμός λύσεων.

Εικόνα 2. (rand\_sol)

| Λιμάνια εκκίνησης πλοίων                                                          |    | Ζεύγος λιμανιών |    |    |    |    |    |   |
|-----------------------------------------------------------------------------------|----|-----------------|----|----|----|----|----|---|
|  | 8  | 32              | 27 | 31 | 34 | 10 | 36 | 0 |
|                                                                                   | 13 | 31              | 35 | 31 | 37 | 0  | 0  | 0 |
|                                                                                   | 31 | 29              | 34 | 0  | 0  | 0  | 0  | 0 |

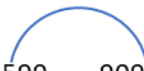
Παράλληλα με την εισαγωγή των κόμβων στην μεταβλητή rand\_sol εισάγεται σε αντίστοιχη μεταβλητή ο χρόνος που φτάνει σε κάθε λιμάνι της διαδρομής το εκάστοτε πλοίο. Το ίδιο συμβαίνει και για τα κόστη των μετακινήσεων που γίνονται. Ο έλεγχος των παραθύρων γίνεται πριν την εισαγωγή τους στις μεταβλητές. Για τον έλεγχο, στο χρόνο ταξιδιού προσθέτουμε και το χρόνο που θα χρειαστεί το λιμάνι για να εξυπηρετήσει το πλοίο. Με άλλα λόγια, το πλοίο θα πρέπει να έχει φτάσει στο εκάστοτε λιμάνι και να έχει ολοκληρώσει την φόρτωση ή την εκφόρτωση πριν το άνω χρονικό παράθυρο του συγκεκριμένου λιμανιού. Στη περίπτωση που ένα πλοίο φτάσει σε κάποιο λιμάνι πριν το κάτω χρονικό παράθυρο, τότε η μετακίνηση δε χαρακτηρίζεται ως μη εφικτή, αλλά το πλοίο περιμένει μέχρι το «άνοιγμα» του λιμανιού. Έτσι ο χρόνος που φτάνει το πλοίο στο συγκεκριμένο λιμάνι ισούται με το κάτω χρονικό παράθυρο. Ο χρόνος ωστόσο που εισάγεται στην αντίστοιχη μεταβλητή είναι η πρόσθεση του κάτω χρονικού ορίου και του χρόνου εξυπηρέτησης. Επιπλέον, κατά των έλεγχων των χρονικών παραθύρων συνηθώς ελέγχεται μόνο το χρονικό παράθυρο του λιμανιού παραλαβής καθώς, αν ένα πλοίο προλάβει να φορτώσει ένα εμπόρευμα είναι πολύ πιθανό να έχει αρκετό χρόνο και για την εκφόρτωση του στο αντίστοιχο λιμάνι. Σε περιπτώσεις όπου το πλοίο φτάνει στο λιμάνι παραλαβής μετά το άνω χρονικό παράθυρο, το συγκεκριμένο ζεύγος λιμανιών αφαιρείται από τη διαδρομή. Οι χρόνοι εμφανίζονται κι αυτοί σε μορφή πίνακα (εικόνα 3), όπου κάθε χρόνος αντιστοιχεί στο χρόνο που το πλοίο έχει ολοκληρώσει την φόρτωση ή την εκφόρτωση του στο λιμάνι, που βρίσκεται στην αντίστοιχη θέση του πίνακα της εικόνας 2. Για παράδειγμα, το πρώτο πλοίο, που ξεκινάει από το λιμάνι 8 (εικόνα 2), κατευθύνεται στο λιμάνι 32 το οποίο είναι το πρώτο λιμάνι παραλαβής. Το συγκεκριμένο πλοίο έχει ολοκληρώσει την φόρτωση του εμπορεύματος από το συγκεκριμένο λιμάνι τη χρονική στιγμή 34 (εικόνα 3) και συνεχίζει τη πορεία του στο λιμάνι 27 (εικόνα 2) στο οποίο θα έχει ολοκληρώσει την εκφόρτωση του φορτίου τη χρονική στιγμή 74 (εικόνα 3). Η πρώτη στήλη του πίνακα της εικόνας 3 αντιστοιχεί στις χρονικές στιγμές από θα είναι διαθέσιμο το κάθε πλοίο. Όμοια εμφανίζονται τα λειτουργικά έξοδα κάθε μετακίνησης στην εικόνα 4.

Εικόνα 3



|    |     |     |     |     |     |     |   |
|----|-----|-----|-----|-----|-----|-----|---|
| 0  | 34  | 74  | 156 | 259 | 751 | 813 | 0 |
| 77 | 111 | 253 | 549 | 709 | 0   | 0   | 0 |
| 90 | 450 | 558 | 0   | 0   | 0   | 0   | 0 |

Εικόνα 4



|   |       |        |        |        |        |        |   |
|---|-------|--------|--------|--------|--------|--------|---|
| 0 | 35599 | 90939  | 176193 | 271323 | 374355 | 447316 | 0 |
| 0 | 52607 | 174937 | 227544 | 364908 | 0      | 0      | 0 |
| 0 | 61503 | 151990 | 0      | 0      | 0      | 0      | 0 |

Στη συγκεκριμένη εκδοχή του GRASP η φάση κατασκευής της αρχικής λύσης δεν ακολουθείται από κάποια κλασσική διαδικασία τοπικής αναζήτησης, καθώς μας ενδιαφέρει κυρίως η δημιουργία του πληθυσμού λύσεων για τον μιμητικό αλγόριθμο. Αντιθέτως, αυτό που ακολουθεί μετά τη δημιουργία της λύσης, είναι μία επιπρόσθετη εισαγωγή κόμβων στην υπάρχουσα λύση. Το συγκεκριμένο κομμάτι του προγράμματος παίρνει τη λύση που δημιουργήθηκε (εικόνα 2) και προσπαθεί να εισάγει στο τέλος κάθε διαδρομής, εμπορεύματα (ζευγάρια λιμανιών) που έμειναν εκτός λύσης κατά την εκτέλεση της GRASP. Εκτός αυτού, συγκρίνει τις ποινές (κόστος για μη εξυπηρέτηση της ζήτησης) των εμπορευμάτων που έχουν μείνει εκτός λύσης με εκείνα εντός κάποιας διαδρομής έτσι ώστε, αν το εμπόρευσμά με τη μεγαλύτερη ποινή είναι εκτός δρομολογίου να εισαχθεί στη λύση, αν αυτό είναι δυνατόν σύμφωνα με τους περιορισμούς. Αυτή η επιπλέον προσπάθεια εισαγωγής κόμβων στη λύση έγινε ώστε να βελτιωθεί η ποιότητα των λύσεων πριν το μιμητικό. Στα προβλήματα δρομολόγησης πλοίων βιομηχανικού φορτίου και ελευθέρων φορτηγών πλοίων, το κόστος που προστίθεται στην αντικειμενική για κάθε εμπόρευμα που δεν εξυπηρετείται αποτελεί μεγάλο ποσοστό του συνολικού κόστους. Για αυτό το λόγο, πέρα από τη μείωση του λειτουργικού κόστους, απώτερος σκοπός είναι η εξυπηρέτηση όσον το δυνατόν περισσότερων φορτίων ώστε να αποφευχθεί η αύξηση κόστος. Εκτός αυτού, όσο λιγότερα φορτία μείνουν εκτός πριν την εισαγωγή των λύσεων στο μιμητικό, δίνεται η δυνατότητα στον αλγόριθμο να πλησιάσει τη βέλτιστη λύση.

Θεωρώντας τα ζευγάρια λιμανιών που μένουν εκτός διαδρομών ως μία μη – εφικτή λύση, θα μπορούσαμε να θεωρήσουμε τις δύο παραπάνω διαδικασίες που περιεγράφηκαν ως παραλλαγές των αλγορίθμων 1-0 επανατοποθέτηση και 1-1 ανταλλαγή μεταξύ της εφικτής λύσης που δημιουργήθηκε από τον GRASP και της μη εφικτής «λύσης» των εμπορευμάτων που δεν εξυπηρετήθηκαν.

Στη συνέχεια υπολογίζεται το συνολικό κόστος βάσει των φορτίων που τελικά έμειναν εκτός και των λειτουργικών εξόδων. Η λύση αποθηκεύεται και η διαδικασία επαναλαμβάνεται ώστε να δημιουργηθεί ο ζητούμενος πληθυσμός δέκα ατόμων.

## 5.4 Βελτίωση Των Λύσεων Με Χρήση Μιμητικού Αλγορίθμου

Στο επόμενο βασικό κομμάτι της επίλυσης του προβλήματος, ο πληθυσμός λύσεων που δημιουργήθηκε στο πρώτο στάδιο πρέπει να εισαχθεί στο γενετικό αλγόριθμο. Οι λύσεις σε αυτό το στάδιο δε θέλουμε να είναι ούτε χαμηλής αλλά ούτε πολύ υψηλής ποιότητας καθώς, θέλουμε να δοθεί η ευκαιρία στο μιμητικό να οδηγηθεί στο ποιοτικότερο τοπικό ελάχιστο χωρίς οι λύσεις που του παρείχαμε να έχουν ήδη προσεγγίσει το τοπικό τους ελάχιστο. Από την άλλη, λύσεις κακής ποιότητας δε θα επιτρέψουν στον αλγόριθμο να βελτιώσει τις λύσεις αρκετά. Η ποιότητα των ατόμων του πληθυσμού στο παρόν πρόβλημα μετρίεται από το συνολικό κόστος της λύσης, δηλαδή το άθροισμα των λειτουργικών εξόδων και των ποινών για τη μη εξυπηρέτηση κάποιων εμπορευμάτων. Συνεπώς τους γονείς της πρώτης γενιάς θα αποτελέσουν αυτές οι λύσεις.

### 5.4.1 Εφαρμογή Γενετικού

Η πρώτη φάση του μιμητικού αλγορίθμου αποτελείται από κάποιο εξελικτικό, νευρωνικά δίκτυα ή κάποιο γενετικό αλγόριθμο. Στην επίλυση του παρόντος προβλήματος για την βελτίωση των λύσεων θα χρησιμοποιηθεί γενετικός αλγόριθμος. Ο πληθυσμός λύσεων που δημιουργήθηκε αποτελείται από ένα πίνακα, όπου κάθε γραμμή του πίνακα αντιπροσωπεύει μία διαδρομή ενός πλοίου. Ξεκινώντας από τη πάνω, μία λύση αποτελείται από έναν αριθμό γραμμών του συγκεκριμένου πίνακα ίσο με τον αριθμό των διαθέσιμων πλοίων. Με άλλα λόγια, αν σε ένα πρόβλημα δρομολόγησης υπάρχουν τρία διαθέσιμα πλοία, τότε οι τρεις πρώτες γραμμές του πίνακα πληθυσμού αντιστοιχούν στη πρώτη λύση (άτομο). Η επόμενη τριάδα στη δεύτερη λύση του πληθυσμού και ούτω καθεξής. Έτσι, ο πίνακας λύσεων θα έχει  $10 \times \text{Αριθμ. Πλοίων}$  γραμμές. Σε κάθε επανάληψη του γενετικού το μέγεθος του πληθυσμού είναι σταθερό και ίσο με δέκα άτομα. Για τη διασταύρωση, σε αυτή την εκδοχή του γενετικού δε χρησιμοποιείται κάποιος συγκεκριμένος τελεστής διασταύρωσης. Αντιθέτως, η διασταύρωση των λύσεων γίνεται παίρνοντας τυχαία μία γραμμή του πίνακα και εκχωρώντας την σε άλλη μεταβλητή. Αυτή η διαδικασία επαναλαμβάνεται για  $10 \times \text{Αριθμ. Πλοίων}$  φορές έως ότου προκύψει ο νέος πληθυσμός λύσεων. Η επιλογή των διαδρομών δε γίνεται τελείως τυχαία αλλά επιλέγεται μία διαδρομή τη φορά από διαφορετικές λύσεις σε κάθε επανάληψη. Με άλλα λόγια κάθε άτομο του πληθυσμού δίνει ένα κομμάτι του για τη δημιουργία του νέου πληθυσμού ατόμων. Με αυτό τον τρόπο προσπαθούμε να πετύχουμε μεγαλύτερη διασπορά στο χώρο λύσεων δίνοντας τη δυνατότητα σε κάθε άτομο να επηρεάσει τη δημιουργία των απογόνων. Η διασταύρωση ωστόσο, θα μπορούσε να γίνει με κάποιο τελεστή ενός ή δύο σημείων, ο οποίος θα χώριζε την εκάστοτε διαδρομή σε κάποιο σημείο και θα την σύνδεε με κομμάτι άλλης διαδρομής από την ίδια λύση ή από άλλη λύση του πληθυσμού. Αυτός ο τρόπος διασταύρωσης εγκαταλείφθηκε νωρίς καθώς, η ικανοποίηση των περιορισμών γινόταν πολύ απαιτητική εξαιτίας του πλήθους των περιορισμών και της πολυπλοκότητας του προβλήματος.

Μετά τη δημιουργία κάθε γενιάς απογόνων, η μορφή των λύσεων έχει αλλοιωθεί αρκετά και κάθε λύση πρέπει να επιδιορθωθεί ώστε να είναι εφικτή. Ο περιορισμός που πρέπει να ικανοποιηθεί στο πρώτο στάδιο επιδιόρθωσης είναι εκείνος που ορίζει, ότι κάθε εμπόρευμα πρέπει να εξυπηρετείται μόνο μία φορά σε

κάθε λύση. Εξαιτίας της ανταλλαγής διαδρομών (γραμμών) μεταξύ των λύσεων προκύπτουν λύσεις που εξυπηρετούν το ίδιο ζεύγος λιμανιών πολλαπλές φορές. Για αυτό το λόγο, ξεκινώντας από τη πρώτη διαδρομή κάθε λύσης, η οποία είναι πάντα εφικτή, προχωράμε στις υπόλοιπες και αφαιρούμε από αυτές ζευγάρια κόμβων (εμπορεύματα) που έχουν ήδη εξυπηρετηθεί σε κάποια από τις παραπάνω διαδρομές(πλοία). Ολοκληρώνοντας αυτή τη διαδικασία αφαίρεσης κόμβων, σε κάποιες διαδρομές προκύπτουν ενδιάμεσα κενά. Οι λύσεις αυτές μπορεί να είναι κατά κάποιο τρόπο εφικτές, άλλα υπάρχουν μεγάλα χρονικά κενά στην εξυπηρέτηση των φορτίων και μεγάλες αναμονές πλοίων σε λιμάνια. Για αυτό το λόγο, αφαιρούνται τα ενδιάμεσα μηδενικά που προέκυψαν με τη προηγούμενη διαδικασία. Εξαιτίας αυτής της ενέργειας ωστόσο, καταστρατηγούνται τα χρονικά παράθυρα των λιμανιών καθώς, μεταβάλλοντας την αλληλουχία των κόμβων αλλάζουν ουσιαστικά οι χρόνοι άφιξης και αναχωρήσεις των πλοίων από τα λιμάνια. Συνεπώς, στη συνέχεια γίνεται επανέλεγχος των χρόνων που φτάνουν τα πλοία σε κάθε λιμάνι ώστε να τηρούνται τα αυστηρά χρονικά παράθυρα. Ζευγάρια κόμβων στα οποία το πλοίο φτάνει εκτός χρονικών ορίων αφαιρείται εκ νέου από τη διαδρομή.

Τέλος, κατά τη διαδικασία της διασταύρωσης μεταβάλλονται και τα λιμάνια εκκίνησης των πλοίων, που αποτελούν το πρώτο κόμβο σε κάθε διαδρομή. Σε αυτό το σημείο του προγράμματος επιλέχθηκε να γίνει εκ νέου αντιστοίχιση των διαθέσιμων πλοίων σε διαδρομές ανάλογα με τα εμπορεύματα που έχουν μπει ήδη σε αυτές. Ελέγχεται δηλαδή, ποιο πλοίο από τη λίστα πλοίων μπορεί να μεταφέρει όλα τα εμπορεύματα που βρίσκονται ήδη στην εκάστοτε διαδρομή σύμφωνα με το περιορισμό συμβατότητας, ώστε να τοποθετηθεί το αντίστοιχο λιμάνι εκκίνησης του στην αρχή της συγκεκριμένης διαδρομής. Κάτι τέτοιο ωστόσο δεν είναι δυνατό σε όλες τις περιπτώσεις, με αποτέλεσμα να προκύπτουν διαδρομές που να μη μπορούν να εξυπηρετηθούν από κάποιο πλοίο που έχει μείνει διαθέσιμο. Για αυτό το λόγο, περιορίστηκε η δράση του συγκεκριμένου σταδίου, έτσι ώστε να γίνεται ανάθεση πλοίων σε διαδρομές όπου αυτό είναι εφικτό και στις υπόλοιπες, το πλοίο να ξεκινάει από το λιμάνι εκκίνησης του και να κινείται αρχικά στο πρώτο λιμάνι της διαδρομής και έπειτα στο πρώτο λιμάνι παραλαβής. Όπως είναι προφανές το τελευταίο θα μπορούσε να εφαρμοστεί σε όλες τις διαδρομές χωρίς να γίνει η παραπάνω διαδικασία. Με άλλα λόγια, κάθε πλοίο θα μπορούσε να ξεκινάει από το λιμάνι εκκίνησης του, κι αν αυτό δε συμπίπτει με το πρώτο λιμάνι της διαδρομής του, να κάνει μία άσκοπη κίνηση στο πρώτο κόμβο και στη συνέχεια να πηγαίνει στο πρώτο λιμάνι παραλαβής (δεύτερο στη σειρά κόμβο). Συνεπώς, η παραπάνω επιδιόρθωση λύσης έγινε για την αποφυγή περιπτώσεων μετακινήσεων και τη μείωση του λειτουργικού κόστους, αυξάνοντας από τη άλλη τη πολυπλοκότητα των ελέγχων και του κώδικα.

Στο τέλος του συγκεκριμένου κομματιού του αλγορίθμου, γίνεται εισαγωγή εμπορευμάτων στη λύση από τη λίστα των μη – εξυπηρετημένων. Αυτή η εισαγωγή φορτίων ακολουθεί την αφαίρεση λιμανιών από τις λύσεις κατά τη διαδικασία επιδιόρθωσης του πληθυσμού. Με την τελευταία δίνεται η ευκαιρία εισαγωγής νέων φορτίων στη λύση, φορτίων που πολύ πιθανόν να είχαν μείνει εκτός κατά τη διαδικασία δημιουργίας των αρχικώς λύσεων. Με αυτό τον τρόπο, προκύπτουν νέες αλληλουχίες λιμανιών, δίνοντας τη δυνατότητα εξερεύνησης καινούργιων περιοχών του χώρου λύσεων. Η εισαγωγή σε αυτό το κομμάτι δε γίνεται με τον ίδιο τρόπο.

Αρχικά εντοπίζονται ποια εμπορεύματα έχουν μείνει εκτός, ξεκινώντας από τη πρώτη λύση του πληθυσμού. Έτσι, έχοντας τη λίστα των μη-εξυπηρετημένων φορτίων, γίνεται ταξινόμηση τους από εκείνο με τη μεγαλύτερη ποινή για τη μη εξυπηρέτηση, προς εκείνο με τη μικρότερη. Στη συνέχεια, επιλέγοντας από τη κορυφή



της ταξινομημένης λίστα, ελέγχεται αν είναι εφικτή η εισαγωγή του συγκεκριμένου ζεύγους λιμανιών στη λύση, σύμφωνα με το περιορισμό συμβατότητας και τα χρονικά παράθυρα. Στη περίπτωση που αυτή δεν είναι εφικτή, επιλέγεται το επόμενο από τη λίστα. Η διαδικασία λειτουργεί επαναληπτικά έως ότου ελεγχθούν όλα τα φορτία της λίστας ή γίνει εισαγωγή κάποιου. Η διαδικασία συνεχίζεται και για το επόμενο πλοίο (επόμενη γραμμή στο πίνακα πληθυσμού) μέχρι το σύνολο των διαθέσιμων πλοίων. Στη συνέχεια, δημιουργείται εκ νέου λίστα μη – εξυπηρετημένων φορτίων για την επόμενη λύση και η διαδικασία επαναλαμβάνεται για όλη την έκταση του πληθυσμού. Στο τέλος κάθε επανάληψης, εντοπίζονται για κάθε λύση τα φορτία που τελικά έμειναν εκτός και υπολογίζεται το συνολικό της κόστος.

## 5.5 Βελτίωση του Πληθυσμού Λύσεων με Εφαρμογή Τοπικής Αναζήτησης

Η τελευταία κύρια φάση του προγράμματος, με τη οποία ολοκληρώνεται ο μιμητικός αλγόριθμος είναι η διαδικασία της τοπικής αναζήτησης. Σε αυτό το στάδιο γίνεται χρήση του αλγορίθμου 1-0 επανατοποθέτηση και 1-1 ανταλλαγή. Σκοπός των συγκεκριμένων αλγορίθμων είναι να βελτιώσουν το αποτέλεσμα του γενετικού (απόγονοι) σε κάθε επανάληψη μετακινώντας ζευγάρια κόμβων εντός των λύσεων, από τη μία διαδρομή στην άλλη. Συγκεκριμένα, με αυτό το κόμματι κώδικα δίνεται έμφαση στη αναζήτηση καλύτερων λύσεων γύρω από περιοχές των λύσεων του γενετικού αλγορίθμου. Η αναζήτηση θα γίνει σε όλες τις λύσεις του πληθυσμού δίνοντας έτσι ευκαιρία εξέλιξης σε όλα τα άτομα και αναζητώντας σε μεγαλύτερο μέρος του χώρου λύσεων. Αναμενόμενο είναι πως αυτή η διαδικασία θα αυξήσει τη χρονική πολυπλοκότητα του κώδικα, αυξάνοντας το χρόνο εκτέλεσης του.

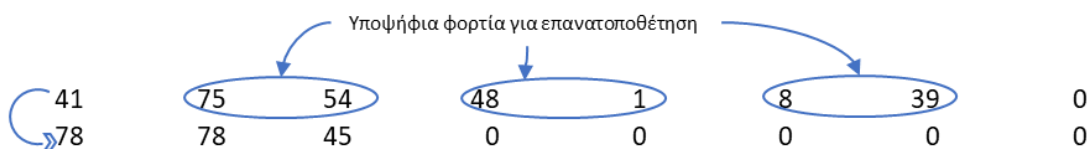
### 5.5.1 1-0 Επανατοποθέτηση

Αρχικά στο πληθυσμό λύσεων εφαρμόζεται ο αλγόριθμος 1-0 ανταλλαγή. Στη συγκεκριμένη μέθοδο τοπικής αναζήτησης επανατοποθετείτε ένας κόμβος από το αρχικό του σημείο σε ένα άλλο. Στη παρούσα φάση, η επανατοποθέτηση γίνεται ανάμεσα σε μία σχετικά γεμάτη διαδρομή προς μία διαδρομή που δεν περνάει από πολλά λιμάνια (εικόνα 5). Βρίσκεται δηλαδή η διαδρομή της λύσης με τους περισσότερους κόμβους και η λύση με τους λιγότερους. Στη συνέχεια, επιλέγεται τυχαία ένα ζεύγος λιμανιών από τη διαδρομή με τους περισσότερους κόμβους και εξετάζεται αν η τοποθέτηση του στην άλλη διαδρομή είναι εφικτή και συμφέρουσα. Η επανατοποθέτηση του φορτίου, για είναι εφικτή θα πρέπει πρώτα το πλοίο, στο οποίο θα ανατεθεί η μεταφορά να μπορεί να εξυπηρετήσει το συγκεκριμένο εμπόρευμα. Σύμφωνα με τον περιορισμό συμβατότητας αλλά και να μπορεί να φτάσει στο λιμάνι παραλαβής πριν το άνω χρονικό παράθυρο, δεδομένων των φορτίων που εξυπηρετεί ήδη. Εκτός αυτού, η επανατοποθέτηση ενός φορτίου θεωρείται συμφέρουσα όταν το άθροισμα του κόστους μεταφοράς και των κοστών εξυπηρέτησης στα λιμάνια, του πλοίου στο οποίο μεταφέρεται, είναι μικρότερο από το αντίστοιχο άθροισμα για το πλοίο στο οποίο ήταν πριν την επανατοποθέτηση. Συνεπώς, αφού ελεγχθούν οι υπάρχοντες περιορισμοί, καθορίζεται αν η μετακίνηση του φορτίου είναι συμφέρουσα.



Μόνο συμφέρουσες μετακινήσεις γίνονται. Στο τέλος της μεθόδους, για τα εμπλεκόμενα πλοία υπολογίζονται εκ νέου το λειτουργικό κόστος και οι χρόνοι άφιξης - αναχώρησης από τα λιμάνια.

Εικόνα 5



Η συγκεκριμένη διαδικασία στοχεύει στη μείωση των λειτουργικών εξόδων κάθε λύσης αλλά και στο ξεμπλοκάρισμα κάποιων διαδρομών, έτσι ώστε να δημιουργηθεί χώρος για εισαγωγή φορτίων που έχουν μείνει εκτός λύσης, κατά τις διαδικασίες που προηγήθηκαν. Η μέθοδος λειτουργεί επαναληπτικά επιχειρώντας την εκτέλεση της παραπάνω διαδικασίας σε όλες τις λύσεις του πληθυσμού. Με αυτό τον τρόπο ολοκληρώνεται η πρώτη φάση της τοπικής αναζήτησης.

### 5.5.2 1 -1 Ανταλλαγή

Στη συνέχεια, ακολουθεί ο αλγόριθμος τοπικής αναζήτησης 1-1 Ανταλλαγή με σκοπό την εντατικοποίηση της αναζήτησης γύρω από τη περιοχή των λύσεων του πληθυσμού. Οι διαδρομές στις οποίες θα γίνει η ανταλλαγή κόμβων επιλέγεται τυχαία για κάθε λύση. Συνεπώς, έχοντας τις δύο διαδρομές ελέγχεται κάθε πιθανή ανταλλαγή ζευγαριών. Συγκεκριμένα, ελέγχεται αν το πρώτο ζευγάρι της πρώτης διαδρομής μπορεί να αλλάξει θέση με οποιοδήποτε άλλο ζευγάρι της δεύτερης διαδρομής. Αν καμία από αυτές δεν γίνουν τότε ελέγχεται το δεύτερο ζευγάρι της πρώτης διαδρομής. Η διαδικασία επαναλαμβάνεται μέχρι να ελεγχθούν όλες οι πιθανές ανταλλαγές (εικόνα 6) ή βρεθεί κάποια που να γίνεται. Όπως και στον προηγούμενο αλγόριθμο τοπικής αναζήτησης, μία ανταλλαγή γίνεται μόνο όταν ικανοποιεί τους περιορισμούς και για τα δύο πλοία και παράλληλα είναι οικονομικά συμφέρουσα. Σε αυτή τη περίπτωση, οικονομικά συμφέρουσα ανταλλαγή θεωρείται, όταν το άθροισμα των λειτουργικών εξόδων για τη μεταφορά των συγκεκριμένων φορτίων στις αρχικές τους θέσεις είναι μεγαλύτερο από τα αντίστοιχα κόστη μετά την ανταλλαγή. Στο τέλος του αλγορίθμου υπολογίζονται ξανά τα κόστη στις διαδρομές που έγινε κάποια ανταλλαγή. Οι χρόνοι υπολογίζονται και εκχωρούνται κατά τον έλεγχο των χρονικών παραθύρων.

Εικόνα 6



Πιθανές ανταλλαγές που ελέγχονται.

## 5.6 Εισαγωγή φορτίων στη λύση

Κατά την ολοκλήρωση της διαδικασίας έχουν δημιουργηθεί διαθέσιμες θέσεις σε διαδρομές που προηγουμένως ήταν γεμάτες. Αυτό συμβαίνει πιο έντονα στη μέθοδο τοπικής αναζήτησης 1-0 επανατοποθέτηση. Για αυτό το λόγο οι αλγόριθμοι τοπικής αναζήτησης ακολουθούνται από μία επιπλέον εισαγωγή εμπορευμάτων (ζυγαριών κόμβων) σε κάθε λύση. Ομοίως με την τελευταία εισαγωγή, κι αυτή λειτουργεί με τις ίδιες αρχές. Για κάθε λύση του πληθυσμού δημιουργείται η λίστα των φορτίων που έχουν μείνει εκτός λύσης(δεν εξυπηρετούνται). Ακολουθεί η ταξινόμηση της λίστας από το φορτίο με τη μεγαλύτερη ποινή προς εκείνο με τη μικρότερη. Το πρώτο υποψήφιο για εισαγωγή φορτίο επιλέγεται από τη κορυφή της λίστας, αν η εισαγωγή του είναι εφικτή, σύμφωνα με του περιορισμούς, τότε ελέγχεται η εισαγωγή του επόμενου στη λίστα φορτίου. Η διαδικασία επαναλαμβάνεται για την ίδια διαδρομή μέχρι να ελεγχθεί η εισαγωγή όλων των φορτίων της λίστας. Στη συνέχεια ακολουθείται η ίδια μέθοδος και για τις υπόλοιπες διαδρομές της λύσης. Ομοίως και για τις υπόλοιπες λύσεις του πληθυσμού επιχειρείται η εισαγωγή επιπλέον φορτίων. Στο τέλος της εισαγωγής, εντοπίζονται τα φορτία που τελικά μένουν εκτός για κάθε λύση και υπολογίζεται το συνολικό κόστος κάθε λύσης, δηλαδή το άθροισμα των εξόδων μεταφοράς και εξυπηρέτησης σε κάθε λιμάνι και των ποινών για την μη εξυπηρέτηση ζήτησης. Συνεπώς, έχοντας το συνολικό κόστος κάθε ατόμου του πληθυσμού, στο τέλος του μιμητικού επιλέγονται τα δέκα ποιοτικότερα άτομα για να γίνουν οι νέοι γονείς. Ο μιμητικός επαναλαμβάνεται με τους νέους γονείς και η διαδικασία ξεκινάει πάλι από τη διασταύρωση των λύσεων. Η μέθοδος λειτουργεί για συγκεκριμένο αριθμό επαναλήψεων.

Όπως είναι προφανές, με αυτό τον τρόπο αυξάνεται η χρονική πολυπλοκότητα του κώδικα, αυξάνοντας παράλληλα και το χρόνο εκτέλεσης του προγράμματος. Παρ' όλα αυτά, η συγκεκριμένη διαδικασία είναι απαραίτητη καθώς, με αυτό τον τρόπο όχι μόνο στοχεύουμε στη κάλυψη του απώτερου σκοπού του προβλήματος, να εξυπηρετηθούν δηλαδή όσο το δυνατόν περισσότερα εμπορεύματα, αλλά αποφεύγονται τα επιπλέον κόστη που προσθέτονται στην αντικειμενική για τη μη εξυπηρέτηση φορτίων. Οι συγκεκριμένες ποινές αποτελούν μεγάλη χρηματική επιβάρυνση σε σχέση με τα υπόλοιπα κόστη του προβλήματος, αφού συχνά επιβαρύνουν την εταιρία με επιπλέον κόστος εκατομμυρίων ευρώ.

Έτσι ολοκληρώνεται ο μιμητικός αλγόριθμος και η διαδικασία βελτίωσης του πληθυσμού λύσεων. Συνεπώς, επιλέγεται η ποιοτικότερη λύση που προέκυψε από την όλη διαδικασία που περιεγράφηκε. Η λύση που επιλέγεται είναι αυτή με μικρότερο συνολικό κόστος και τα λιγότερα μη εξυπηρετημένα φορτία.

## 6. Παρουσίαση Αποτελεσμάτων

### 6.1 Παραδείγματα εφαρμογής αλγορίθμου

Στο συγκεκριμένο κεφάλαιο θα γίνει παρουσίαση των αποτελεσμάτων. Η εφαρμογή του αλγορίθμου έγινε σε παραδείγματα βασισμένα σε πραγματικά σενάρια προβλημάτων δρομολόγησης, όμοια με αυτά που καλείται να αντιμετωπίσει μία εταιρία που δραστηριοποιείται στο συγκεκριμένο χώρο. Αυτό έγινε εκμεταλλευόμενοι τη πρόσφατη δημιουργία των συγκεκριμένων παραδειγμάτων από τους Hemmati, Hvattum, Fagerholt, and Norstad (2014) οι οποίοι δημιούργησαν ένα σύνολο ποικίλων παραδειγμάτων μεγέθους από 8 έως 130 εμπορεύματα. Το μέγεθος του κάθε παραδείγματος καθορίζεται από τα φορτία (cargoes) που χρήζουν μεταφοράς και τον αριθμό των διαθέσιμων πλοίων. Εκτός αυτού, τα παραδείγματα χωρίζονται σε δύο μεγάλες κατηγορίες, στα προβλήματα μεγάλων αποστάσεων (deep sea shipping) και στα προβλήματα κοντινών αποστάσεων. Κάθε κατηγορία διαχωρίζεται σε δύο υποκατηγορίες, στα προβλήματα πλήρους φορτίου(FUN) και στα προβλήματα μικτού φορτίου (mixed load). Τα αποτελέσματα που θα παρουσιαστούν προέρχονται και από τις δύο κατηγορίες προβλημάτων πλήρους φορτίου (FUN) και το μέγεθος τους ποικίλει. Η σύγκριση των αποτελεσμάτων γίνεται σε σχέση με τις βέλτιστες τιμές που πάρθηκαν από σχετικό άρθρο “ *HOMSI, Gabriel, et al. Industrial and tramp ship routing problems: Closing the gap for real-scale instances. European Journal of Operational Research*”. Η σύγκριση με αυτό το άρθρο είναι έγκυρη, καθώς ακολουθήθηκε ίδια μοντελοποίηση. Τα παραδείγματα που επιλέχθηκαν για παρουσίαση είναι εκείνα με τα καλύτερα αποτέλεσμα. Συγκεκριμένα, για κάθε παράδειγμα παρουσιάζεται :

- Ο τύπος του προβλήματος (Deep sea / Short sea / FUN)
- Το μέγεθος του προβλήματος
- Ο αριθμός των λιμανιών (κόμβων)
- Η καλύτερη λύση. Τα αποτελέσματα σε κάθε στάδιο του αλγορίθμου. (Σχετικοί Πίνακες)
- Η απόκλιση της από τις βέλτιστες τιμές.
- Και ο χρόνος εκτέλεσης του προγράμματος

## Συγκεντρωτικός πίνακας αποτελεσμάτων

| Παράδειγμα                | GRASP        | Γενετικό     | Βελτίωση Γενετικού | Μιμητικός    | Βελτίωση τοπικής | Συνολική Βελτίωση Μιμητικού | Απόκλιση     |
|---------------------------|--------------|--------------|--------------------|--------------|------------------|-----------------------------|--------------|
| DEEPSEA_FUN_C8_V3_HE_3    | 8.830.798€   | 4.596.681 €  | 92,11%             | 4.596.681 €  | 0,00%            | 92,11%                      | 0,00%        |
| SHORTSEA_FUN_C8_V3_HE_3   | 2.104.562 €  | 1.941.726 €  | 8,39%              | 1.698.102 €  | 14,3%            | 23,94%                      | 0,00%        |
| DEEPSEA_FUN_C8_V3_HE_1    | 11.625.314 € | 9.584.863 €  | 21,29%             | 9.584.863 €  | 0,00%            | 21,29%                      | 0,00%        |
| DEEPSEA_FUN_C11_V4_HE_2   | 47.793.723 € | 27.504.715 € | 73,77%             | 27.491.364 € | 0,05%            | 73,85%                      | 6,15%        |
| DEEPSEA_FUN_C16_V6_HE_1   | 66.529.203 € | 52.496.185 € | 26,73%             | 52.496.185 € | 0,00%            | 26,72%                      | 2,68%        |
| SHORTSEA_FUN_C17_V13_HE_4 | 3.410.295 €  | 2.947.380 €  | 15,71%             | 2.841.687 €  | 3,77%            | 20,01%                      | 1,26%        |
| DEEPSEA_FUN_C25_V7_HE_1   | 34.737.782 € | 26.059.469   | 33,30%             | 24.037.588 € | 8,88%            | 44,51%                      | 5,55%        |
| DEEPSEA_FUN_C90_V40_HE_1  | 301.612.274€ | 197.018.896  | 53,09%             | 197.018.896€ | 0,00%            | 51,49%                      | 3,35%        |
| DEEPSEA_FUN_C100_V50_HE_4 | 322.998.375€ | 242.875.490€ | 32,99%             | 232.272.899€ | 4,80%            | 39,06%                      | 5,16%        |
|                           |              |              |                    |              |                  | <b>M.O</b>                  | <b>2,68%</b> |

## 6.2 Συμπέρασμα

Στον συγκεντρωτικό πίνακα αποτελεσμάτων παρουσιάζεται μία γενική εικόνα της απόδοσης του μιμητικού αλγορίθμου σε συνδυασμό με τις αρχικές λύσεις του GRASP. Σε σύγκριση με τις βέλτιστες τιμές συνολικού κόστους της βιβλιογραφίας, τα αποτελέσματα που παρουσιάστηκαν έχουν κατά μέσο όρο απόκλιση 2,68% με κάποια από αυτά να ταυτίζονται με τις βέλτιστες τιμές. Εκτός αυτού, στον εν λόγω πίνακα φαίνεται και η συνεισφορά του κάθε σταδίου του αλγόριθμου επίλυσης, στη βελτίωση της λύσης μέχρι το τελικό αποτέλεσμα. Αυτό που παρατηρείται είναι ότι σε κάποιες περιπτώσεις η τοπική αναζήτηση στο τέλος του μιμητικού δε βελτιώνει το αποτέλεσμα. Αυτό συμβαίνει είτε γιατί η βέλτιστη τιμή έχει ήδη βρεθεί από τον γενετικό αλγόριθμο ή ο αλγόριθμος έχει «πέσει» σε κάποιο τοπικό ελάχιστο του χώρου λύσεων. Επιπλέον, παρατηρείται ότι η μεγαλύτερη βελτίωση των αρχικών λύσεων γίνεται στη φάση του γενετικού αλγορίθμου ενώ η τοπική αναζήτηση έχει μικρή συνεισφορά. Παρ' όλα αυτά κάτι τέτοιο είναι αναμενόμενο καθώς, πρωταρχικός σκοπός του γενετικού είναι να εντοπίζει την περιοχή στο χώρο λύσεων όπου υπάρχει ένα ποιοτικό τοπικό ελάχιστο, ενώ της τοπικής είναι να προσεγγίσει αυτό το ελάχιστο. Συνεπώς, αναμένουμε μεγαλύτερη βελτίωση στο στάδιο του γενετικού, όπου τα ποσοστά καλυτέρευσης της λύσης φτάνουν το 92,11%. Όπως παρατηρείται, όσο αυξάνεται ο αριθμός των φορτίων προς μεταφορά και των διαθέσιμων πλοίων, τόσο τείνει να αυξάνεται και η απόκλιση από τις βέλτιστες τιμές. Αυτό συμβαίνει εξαιτίας της μεγάλης πολυπλοκότητας που έχουν τα παραδείγματα με πάνω από 50 φορτία. Οι πιθανές διαδρομές που καλείται να εξετάσει ο αλγόριθμος είναι πάρα πολλές και κατ' επέκταση το μέγεθος του χώρου λύσεων τεράστιο. Γι' αυτό το λόγο, είναι όχι μόνο αρκετά απαιτητικό για τον μιμητικό να συγκλίνει σε βέλτιστες τιμές αλλά και πολύ χρονοβόρο. Παρ' όλα αυτά, εντοπίστηκαν πολύ ποιοτικές λύσεις και σε προβλήματα μεγάλα διαστάσεων. Όπως φαίνεται και στις δύο τελευταίες γραμμές του συγκεντρωτικού πίνακα στα δύο απ' τα μεγαλύτερα προβλήματα του συνόλου παραδειγμάτων βρέθηκαν λύσεις με 2 και 5% απόκλιση από τις βέλτιστες. Αυτό επιτεύχθηκε καθώς, ο μιμητικός αλγόριθμος έχει τόσο καλές ικανότητες διασποράς της αναζήτησης στο πεδίο λύσεων, όσο και δυνατότητες εμβάθυνσης της αναζήτησης του γύρω από ποιοτικές περιοχές λύσεων.

### 6.3 Τροποποιήσεις στο Τρόπο Επίλυσης

Στο πρώτο εγχείρημα δημιουργίας της λύσης χρησιμοποιήθηκε μία εκδοχή του αλγόριθμου πλησιέστερου γείτονα. Συγκεκριμένα, τοποθετώντας στην αρχή κάθε διαδρομής το λιμάνι εκκίνησης του εκάστοτε πλοίου και με τη χρήση της μεταβλητής προσιτών κόμβων εντοπίζονται τα λιμάνια παραλαβής στα οποία μπορεί να κινηθεί το πλοίο. Από αυτά τα λιμάνια επιλέγεται εκείνο με το μικρότερο χρόνο ταξιδιού και στη συνέχεια τοποθετείται αυτομάτως το αντίστοιχο λιμάνι παράδοσης για τη συγκεκριμένη ζήτηση. Το συγκεκριμένο λιμάνι «αφαιρείται» από τον πίνακα και τη θέση, που προηγουμένως είχε το λιμάνι εκκίνησης, παίρνει ο τελευταίος κόμβος της διαδρομής. Η διαδικασία συνεχίζεται μέχρι να μην υπάρχει κάποιο εμπόρευμα (ζευγάρι κόμβων) που να μπορεί να εισαχθεί στη λύση. Σε κάθε εισαγωγή λιμανιού παραλαβής ελέγχονται τα χρονικά παράθυρα του λιμανιού έτσι ώστε η λύση που θα προκύψει να είναι εφικτή. Στη λίστα προσιτών κόμβων εισάγονται μόνο φορτία που ήταν συμβατά με το συγκεκριμένο πλοίο. Η διαδικασία επαναλαμβάνεται για κάθε πλοίο.

Η συγκεκριμένη εκδοχή του αλγορίθμου του πλησιέστερου γείτονα δε εξυπηρετούσε επαρκώς το υπόλοιπο μέρος του προγράμματος, δηλαδή το μιμητικό. Ο μιμητικός απαιτούσε ως εισόδου του πληθυσμού λύσεων κάτι που ήταν δύσκολο να γίνει με τη χρήση του αλγόριθμου του πλησιέστερου γείτονα. Γι' αυτό το λόγο έγινε μετατροπή του εν λόγω αλγορίθμου σε GRASP που ήταν πιο βολικός για τη δημιουργία πληθυσμού αλλά και για να πετύχουμε τη κατάλληλη ποικιλία λύσεων στον πληθυσμό, κάτι που είναι πολύ σημαντικό για την συνέχεια του προγράμματος

Επιπλέον, τροποποιήσεις από την αρχική εκδοχή έγιναν και στο στάδιο της επιδιόρθωσης των ατόμων του πληθυσμού μετά τη διασταύρωση. Συγκεκριμένα, η εκ νέου αντιστοίχιση πλοίων (λιμανιών εκκίνησης) σε διαδρομές ήταν δύσκολο να γίνει σε κάθε περίπτωση δεδομένων. Για αυτό το λόγο όπου ήταν ανέφικτο να γίνει, τα πλοία έμεναν όπως ήταν αρχικά και έκανα μία άσκοπη μετακίνηση στο πρώτο λιμάνι της διαδρομής. Αυτό μείωσε τη πολυπλοκότητα του αλγορίθμου, περιόρισε τις περιπτώσεις λαθών και μείωσε τον αριθμό των ελέγχων.

### 6.4 Αποτελέσματα Παραδειγμάτων

#### Παράδειγμα#1 : DEEPSEA\_FUN\_C8\_V3\_HE\_3

*Αριθμός λιμανιών: 86*

*Αριθμός φορτίων : 8*

*Αριθμός πλοίων : 3*

*Ενδεικτικός χρόνος εκτέλεσης: 380,15sec*

*Βέλτιστη τιμή συνολικού κόστους: 4.596.681 €*

Πίνακα 2

|                 |        |                |                          |         |                      |                  |           |             |
|-----------------|--------|----------------|--------------------------|---------|----------------------|------------------|-----------|-------------|
|                 |        | Λύση μιμητικού |                          |         |                      | Χρόνος εκτέλεσης | 380,15sec | Απόκλιση 0% |
| Συνολικό κόστος |        | 4.596.681 €    |                          |         | 0 φορτία εκτός λύσης |                  |           |             |
|                 |        |                | <b>Διαδρομές</b>         |         |                      |                  |           |             |
| 78              | 79     | 33             | 33                       | 16      | 4                    | 51               | 0         | 0           |
| 13              | 62     | 32             | 36                       | 59      | 15                   | 77               | 72        | 58          |
| 73              | 72     | 32             | 0                        | 0       | 0                    | 0                | 0         | 0           |
|                 |        |                | <b>Χρόνοι Αναχώρησης</b> |         |                      |                  |           |             |
| 310             | 759    | 1336           | 1539                     | 1937    | 3361                 | 4166             | 0         | 0           |
| 0               | 98     | 485            | 599                      | 886     | 1759                 | 2458             | 2847      | 3184        |
| 0               | 111    | 748            | 0                        | 0       | 0                    | 0                | 0         | 0           |
|                 |        |                |                          |         |                      |                  |           |             |
|                 |        |                | <b>Λειτουργικά κόστη</b> |         |                      |                  |           |             |
| 0               | 76238  | 588115         | 660936                   | 1022669 | 1142369              | 1827223          | 0         | 0           |
| 0               | 109960 | 453954         | 593209                   | 831885  | 958866               | 1511301          | 1585626   | 1865837     |
| 0               | 160576 | 903621         | 0                        | 0       | 0                    | 0                | 0         | 0           |

## Παράδειγμα #2 : SHORTSEA\_FUN\_C8\_V3\_HE\_3

Αριθμός λιμανιών: 39

Αριθμός φορτίων : 8

Αριθμός πλοίων : 3

Ενδεικτικός χρόνος εκτέλεσης: 147.46sec

Βέλτιστη τιμή συνολικού κόστους: 1.698.102 €

Πίνακας 3

|                 |       | Λύση μιμητικού           |        |        |                      |        |   |  |  | Απόκλιση<br>0% |
|-----------------|-------|--------------------------|--------|--------|----------------------|--------|---|--|--|----------------|
| Συνολικό κόστος |       | 1.698.102€               |        |        | 1 φορτίο εκτός λύσης |        |   |  |  |                |
|                 |       | <b>Διαδρομές</b>         |        |        |                      |        |   |  |  |                |
| 31              | 31    | 34                       | 29     | 34     | 10                   | 36     | 0 |  |  |                |
| 8               | 32    | 27                       | 26     | 25     | 31                   | 37     | 0 |  |  |                |
| 13              | 31    | 35                       | 0      | 0      | 0                    | 0      | 0 |  |  |                |
|                 |       | <b>Χρόνοι Αναχώρησης</b> |        |        |                      |        |   |  |  |                |
| 90              | 126   | 222                      | 450    | 558    | 751                  | 808    | 0 |  |  |                |
| 0               | 34    | 74                       | 255    | 332    | 549                  | 709    | 0 |  |  |                |
| 0               | 99    | 241                      | 0      | 0      | 0                    | 0      | 0 |  |  |                |
|                 |       |                          |        |        |                      |        |   |  |  |                |
|                 |       | <b>Λειτουργικά κόστη</b> |        |        |                      |        |   |  |  |                |
| 0               | 31147 | 114557                   | 202406 | 292893 | 352267               | 418897 | 0 |  |  |                |
| 0               | 35599 | 90939                    | 161394 | 241936 | 285730               | 423094 | 0 |  |  |                |
| 0               | 52607 | 174937                   | 0      | 0      | 0                    | 0      | 0 |  |  |                |
|                 |       |                          |        |        |                      |        |   |  |  |                |

### Παράδειγμα #3 : DEEPSEA\_FUN\_C8\_V3\_HE\_1

Αριθμός λιμανιών: 86

Αριθμός φορτίων : 8

Αριθμός πλοίων : 3

Βέλτιστη τιμή συνολικού κόστους: 9.584.863 €

|                    |        | Λύση μιμητικού |                          |                      |   |   |  |  | Απόκλιση<br>0% |
|--------------------|--------|----------------|--------------------------|----------------------|---|---|--|--|----------------|
| Συνολικό<br>κόστος |        | 9.584.863 €    |                          | 2 φορτία εκτός λύσης |   |   |  |  |                |
|                    |        |                | <b>Διαδρομές</b>         |                      |   |   |  |  |                |
| 13                 | 8      | 39             | 78                       | 36                   | 0 | 0 |  |  |                |
| 78                 | 76     | 35             | 5                        | 54                   | 0 | 0 |  |  |                |
| 73                 | 79     | 40             | 81                       | 69                   | 0 | 0 |  |  |                |
|                    |        |                | <b>Χρόνοι Αναχώρησης</b> |                      |   |   |  |  |                |
| 0                  | 2031   | 2442           | 3402                     | 3990                 | 0 | 0 |  |  |                |
| 363                | 1519   | 2081           | 2673                     | 3503                 | 0 | 0 |  |  |                |
| 0                  | 54     | 456            | 1073                     | 1594                 | 0 | 0 |  |  |                |
|                    |        |                |                          |                      |   |   |  |  |                |
|                    |        |                | <b>Λειτουργικά κόστη</b> |                      |   |   |  |  |                |
| 0                  | 176494 | 533256         | 1071603                  | 1595887              | 0 | 0 |  |  |                |
| 0                  | 79471  | 572091         | 895093                   | 1605918              | 0 | 0 |  |  |                |
| 0                  | 70357  | 558507         | 1017034                  | 1614295              | 0 | 0 |  |  |                |



## Παράδειγμα #4 DEEPSEA\_FUN\_C11\_V4\_HE\_2

Αριθμός λιμανιών: 86

Αριθμός φορτίων : 11

Αριθμός πλοίων : 4

Ενδεικτικός χρόνος εκτέλεσης: 401.29sec

Βέλτιστη τιμή συνολικού κόστους: 25.454.434 €

|                   |                 |         |                |             |         |         |                      |         |                   |           |
|-------------------|-----------------|---------|----------------|-------------|---------|---------|----------------------|---------|-------------------|-----------|
|                   |                 |         | Λύση μιμητικού |             |         |         |                      |         | Απόκλιση<br>6,15% |           |
|                   | Συνολικό Κόστος |         |                | 27.018.993€ |         |         | 1 φορτίο εκτός λύσης |         |                   |           |
| Διαδρομές         |                 |         |                |             |         |         |                      |         |                   |           |
| 78                | 76              | 41      | 42             | 86          | 64      | 77      | 76                   | 34      | 84                | 60        |
| 41                | 72              | 54      | 41             | 7           | 0       | 0       | 0                    | 0       | 0                 | 0         |
| 13                | 72              | 47      | 48             | 68          | 0       | 0       | 0                    | 0       | 0                 | 0         |
| 73                | 78              | 57      | 0              | 0           | 0       | 0       | 0                    | 0       | 0                 | 0         |
| Χρόνοι Αναχώρησης |                 |         |                |             |         |         |                      |         |                   |           |
| 3                 | 89              | 324     | 392            | 651         | 816     | 1222    | 1264                 | 1621    | 1803              | 2030      |
| 463               | 1007            | 1208    | 1509           | 2007        | 0       | 0       | 0                    | 0       | 0                 | 0         |
| 0                 | 444             | 649     | 1455           | 1865        | 0       | 0       | 0                    | 0       | 0                 | 0         |
| 0                 | 79              | 306     | 0              | 0           | 0       | 0       | 0                    | 0       | 0                 | 0         |
| Λειτουργικά Κόστη |                 |         |                |             |         |         |                      |         |                   |           |
| 0                 | 126318          | 1178631 | 1474968        | 2937082     | 3460411 | 5346532 | 5459346              | 7272284 | 8066106           | 9.263.861 |
| 0                 | 1318003         | 2633486 | 2710483        | 5597846     | 0       | 0       | 0                    | 0       | 0                 | 0         |
| 0                 | 1823124         | 2909422 | 3176118        | 5256636     | 0       | 0       | 0                    | 0       | 0                 | 0         |
| 0                 | 149934          | 1373525 | 0              | 0           | 0       | 0       | 0                    | 0       | 0                 | 0         |

**Παράδειγμα #5 : DEEPSEA\_FUN\_C16\_V6\_HE\_1**

Αριθμός λιμανιών: 86

Αριθμός φορτίων : 16

Αριθμός πλοίων : 6

Βέλτιστη τιμή συνολικού κόστους: 51.127.590 €

|                   |         |                |         |         |                      |         |   |   |                   |
|-------------------|---------|----------------|---------|---------|----------------------|---------|---|---|-------------------|
|                   |         | Λύση μιμητικού |         |         |                      |         |   |   | Απόκλιση<br>2,68% |
| Συνολικό Κόστος   |         | 52.521.777€    |         |         | 3 φορτία εκτός λύσης |         |   |   |                   |
| Διαδρομές         |         |                |         |         |                      |         |   |   |                   |
| 78                | 73      | 51             | 79      | 41      | 78                   | 36      | 0 | 0 |                   |
| 13                | 8       | 39             | 21      | 79      | 79                   | 6       | 0 | 0 |                   |
| 13                | 76      | 35             | 0       | 0       | 0                    | 0       | 0 | 0 |                   |
| 41                | 80      | 57             | 43      | 14      | 79                   | 40      | 0 | 0 |                   |
| 73                | 78      | 42             | 81      | 69      | 0                    | 0       | 0 | 0 |                   |
| 45                | 48      | 1              | 0       | 0       | 0                    | 0       | 0 | 0 |                   |
| Χρόνοι Αναχώρησης |         |                |         |         |                      |         |   |   |                   |
| 0                 | 71      | 318            | 562     | 784     | 1501                 | 1840    | 0 | 0 |                   |
| 0                 | 166     | 391            | 796     | 1142    | 1613                 | 1977    | 0 | 0 |                   |
| 0                 | 1136    | 1461           | 0       | 0       | 0                    | 0       | 0 | 0 |                   |
| 0                 | 249     | 431            | 760     | 1184    | 1554                 | 1810    | 0 | 0 |                   |
| 0                 | 294     | 529            | 892     | 1213    | 0                    | 0       | 0 | 0 |                   |
| 0                 | 87      | 339            | 0       | 0       | 0                    | 0       | 0 | 0 |                   |
| Λειτουργικά Κόστη |         |                |         |         |                      |         |   |   |                   |
| 0                 | 236207  | 1353549        | 2379934 | 3521942 | 4623070              | 6365821 | 0 | 0 |                   |
| 0                 | 515284  | 1671770        | 2624879 | 4383692 | 4428492              | 6272191 | 0 | 0 |                   |
| 0                 | 1802773 | 3469139        | 0       | 0       | 0                    | 0       | 0 | 0 |                   |
| 0                 | 1297229 | 2488220        | 2642442 | 5303567 | 7400525              | 8801477 | 0 | 0 |                   |
| 0                 | 266027  | 1486427        | 2712882 | 4189480 | 0                    | 0       | 0 | 0 |                   |
| 0                 | 361690  | 1732430        | 0       | 0       | 0                    | 0       | 0 | 0 |                   |

**Παράδειγμα #6 : SHORTSEA\_FUN\_C17\_V13\_HE\_4**

Αριθμός λιμανιών: 39

Αριθμός φορτίων : 17

Αριθμός πλοίων : 13

Χρόνος εκτέλεσης: ???

Βέλτιστη τιμή συνολικού κόστους: 2.806.231 €€

|                   |        |        |        |                 |   |   |                   |     |     |                   |     |   |   |
|-------------------|--------|--------|--------|-----------------|---|---|-------------------|-----|-----|-------------------|-----|---|---|
|                   |        |        |        | Λύσης μιμητικού |   |   |                   |     |     | Απόκλιση<br>1,26% |     |   |   |
| Συνολικό κόστος   |        |        |        | 2.841.687 €     |   |   |                   |     |     |                   |     |   |   |
| Διαδρομές         |        |        |        |                 |   |   | Χρόνοι Αναχώρησης |     |     |                   |     |   |   |
| 14                | 29     | 27     | 0      | 0               | 0 | 0 | 0                 | 433 | 610 | 0                 | 0   | 0 | 0 |
| 24                | 11     | 35     | 29     | 27              | 0 | 0 | 0                 | 117 | 221 | 433               | 598 | 0 | 0 |
| 14                | 29     | 36     | 0      | 0               | 0 | 0 | 0                 | 343 | 456 | 0                 | 0   | 0 | 0 |
| 32                | 30     | 36     | 33     | 36              | 0 | 0 | 0                 | 421 | 534 | 608               | 670 | 0 | 0 |
| 17                | 17     | 39     | 0      | 0               | 0 | 0 | 0                 | 37  | 229 | 0                 | 0   | 0 | 0 |
| 31                | 11     | 34     | 34     | 8               | 0 | 0 | 43                | 184 | 257 | 291               | 474 | 0 | 0 |
| 19                | 10     | 14     | 29     | 17              | 0 | 0 | 43                | 163 | 250 | 520               | 681 | 0 | 0 |
| 31                | 33     | 34     | 0      | 0               | 0 | 0 | 43                | 90  | 159 | 0                 | 0   | 0 | 0 |
| 31                | 4      | 34     | 0      | 0               | 0 | 0 | 43                | 138 | 185 | 0                 | 0   | 0 | 0 |
| 31                | 30     | 34     | 0      | 0               | 0 | 0 | 0                 | 34  | 159 | 0                 | 0   | 0 | 0 |
| 32                | 31     | 36     | 0      | 0               | 0 | 0 | 0                 | 406 | 499 | 0                 | 0   | 0 | 0 |
| 13                | 10     | 38     | 0      | 0               | 0 | 0 | 0                 | 138 | 253 | 0                 | 0   | 0 | 0 |
| 8                 | 1      | 17     | 0      | 0               | 0 | 0 | 0                 | 35  | 67  | 0                 | 0   | 0 | 0 |
|                   |        |        |        |                 |   |   |                   |     |     |                   |     |   |   |
| Λειτουργικά Κόστη |        |        |        |                 |   |   |                   |     |     |                   |     |   |   |
| 0                 | 127369 | 275771 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 33522  | 122344 | 234774 | 361873          | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 105027 | 213004 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 65841  | 165550 | 227447 | 296662          | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 25418  | 166358 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 23025             | 72440  | 146810 | 171828 | 322442          | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 94559  | 177539 | 265388 | 385598          | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 48427  | 120481 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 65658  | 122756 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 44607  | 154341 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 72014  | 159904 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 70385  | 180094 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 36609  | 82403  | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 127369 | 275771 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 33522  | 122344 | 234774 | 361873          | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 105027 | 213004 | 0      | 0               | 0 | 0 |                   |     |     |                   |     |   |   |
| 0                 | 65841  | 165550 | 227447 | 296662          | 0 | 0 |                   |     |     |                   |     |   |   |
|                   |        |        |        |                 |   |   |                   |     |     |                   |     |   |   |

**Παράδειγμα #7 DEEPSEA\_FUN\_C25\_V7\_HE\_1**

Αριθμός λιμανιών: 86

Αριθμός φορτίων : 7

Αριθμός πλοίων : 25

Βέλτιστη τιμή συνολικού κόστους: 22.773.158 €

|                   |        |         |                |         |         |                      |         |                   |   |
|-------------------|--------|---------|----------------|---------|---------|----------------------|---------|-------------------|---|
|                   |        |         | Λύση μιμητικού |         |         |                      |         | Απόκλιση<br>5,55% |   |
| Συνολικό Κόστος   |        |         | 24.037.588 €   |         |         | 5 φορτία εκτός λύσης |         |                   |   |
| Διαδρομές         |        |         |                |         |         |                      |         |                   |   |
| 33                | 5      | 54      | 78             | 42      | 0       | 0                    | 0       | 0                 | 0 |
| 13                | 21     | 79      | 78             | 40      | 48      | 1                    | 78      | 36                | 0 |
| 41                | 75     | 49      | 75             | 54      | 81      | 52                   | 78      | 32                | 0 |
| 41                | 43     | 14      | 81             | 69      | 0       | 0                    | 0       | 0                 | 0 |
| 45                | 78     | 45      | 0              | 0       | 0       | 0                    | 0       | 0                 | 0 |
| 73                | 80     | 57      | 79             | 6       | 63      | 39                   | 8       | 39                | 0 |
| 78                | 79     | 40      | 79             | 41      | 76      | 35                   | 0       | 0                 | 0 |
| Χρόνοι Αναχώρησης |        |         |                |         |         |                      |         |                   |   |
| 84                | 604    | 1378    | 2994           | 3392    | 0       | 0                    | 0       | 0                 | 0 |
| 0                 | 118    | 750     | 827            | 1233    | 1629    | 2026                 | 2775    | 3392              | 0 |
| 0                 | 401    | 811     | 1268           | 1641    | 2049    | 2419                 | 3215    | 3881              | 0 |
| 0                 | 956    | 1720    | 2380           | 2934    | 0       | 0                    | 0       | 0                 | 0 |
| 0                 | 1324   | 1739    | 0              | 0       | 0       | 0                    | 0       | 0                 | 0 |
| 0                 | 113    | 477     | 1080           | 1709    | 1939    | 2292                 | 2729    | 3166              | 0 |
| 0                 | 74     | 500     | 922            | 1351    | 3014    | 3576                 | 0       | 0                 | 0 |
| Λειτουργικά κόστη |        |         |                |         |         |                      |         |                   |   |
| 0                 | 477051 | 1415069 | 1876917        | 2382760 | 0       | 0                    | 0       | 0                 | 0 |
| 0                 | 131502 | 634882  | 738080         | 1101650 | 1199664 | 1541113              | 2159396 | 2.683.680         | 0 |
| 0                 | 344903 | 688774  | 1041725        | 1400962 | 1758369 | 2091399              | 2418818 | 2.976.054         | 0 |
| 0                 | 68774  | 702235  | 1252487        | 1705407 | 0       | 0                    | 0       | 0                 | 0 |
| 0                 | 494769 | 990778  | 0              | 0       | 0       | 0                    | 0       | 0                 | 0 |
| 0                 | 118345 | 445244  | 763399         | 1312677 | 1484469 | 1797154              | 2156396 | 2.513.158         | 0 |
| 0                 | 76238  | 455139  | 797840         | 1158688 | 1503830 | 1996450              | 0       | 0                 | 0 |

# Παράδειγμα #8 DEEPSEA\_FUN\_C90\_V40\_HE\_1

Αριθμός λιμανιών: 86

Αριθμός φορτίων : 90

Αριθμός πλοίων : 40

Βέλτιστη τιμή συνολικού κόστους: 190.627.186 € €

|                 |    |    |    |    |    |    | Λύση μιμητικού |      |                      |      |      |      | Απόκλιση<br>3,35% |
|-----------------|----|----|----|----|----|----|----------------|------|----------------------|------|------|------|-------------------|
| Συνολικό Κόστος |    |    |    |    |    |    | 197.018.896    |      | 5 φορτία εκτός λύσης |      |      |      |                   |
| Διαδρομές       |    |    |    |    |    |    |                |      | Χρόνοι Αναχώρησης    |      |      |      |                   |
| 78              | 44 | 11 | 78 | 36 | 0  | 0  | 0              | 1898 | 2929                 | 3985 | 4710 | 0    | 0                 |
| 55              | 52 | 59 | 78 | 35 | 14 | 70 | 0              | 234  | 1170                 | 2441 | 3179 | 3583 | 3961              |
| 68              | 79 | 22 | 73 | 51 | 0  | 0  | 0              | 1547 | 2446                 | 3546 | 3975 | 0    | 0                 |
| 30              | 30 | 78 | 79 | 6  | 0  | 0  | 0              | 100  | 1110                 | 2079 | 3047 | 0    | 0                 |
| 58              | 55 | 62 | 75 | 35 | 0  | 0  | 0              | 104  | 1075                 | 2565 | 3322 | 0    | 0                 |
| 41              | 74 | 33 | 74 | 52 | 72 | 15 | 13             | 1001 | 1636                 | 2411 | 2844 | 3234 | 4250              |
| 30              | 80 | 55 | 41 | 14 | 69 | 64 | 0              | 1265 | 1634                 | 2004 | 2922 | 3755 | 4541              |
| 72              | 81 | 34 | 81 | 23 | 0  | 0  | 3              | 212  | 907                  | 2866 | 3901 | 0    | 0                 |
| 62              | 66 | 49 | 75 | 49 | 75 | 42 | 0              | 378  | 1408                 | 2325 | 2757 | 3303 | 3767              |
| 72              | 79 | 40 | 77 | 37 | 0  | 0  | 0              | 149  | 595                  | 2528 | 3210 | 0    | 0                 |
| 76              | 75 | 33 | 78 | 38 | 0  | 0  | 0              | 450  | 1148                 | 2116 | 2927 | 0    | 0                 |
| 19              | 34 | 12 | 5  | 54 | 79 | 34 | 0              | 656  | 1046                 | 1202 | 2202 | 2686 | 3341              |
| 30              | 21 | 79 | 79 | 34 | 75 | 35 | 0              | 143  | 1080                 | 1212 | 1867 | 3015 | 3772              |
| 76              | 78 | 48 | 78 | 40 | 75 | 50 | 0              | 848  | 1226                 | 1606 | 2027 | 2439 | 2855              |
| 85              | 63 | 74 | 74 | 35 | 0  | 0  | 0              | 164  | 1097                 | 2077 | 2756 | 0    | 0                 |
| 55              | 43 | 14 | 63 | 39 | 38 | 84 | 0              | 189  | 1121                 | 2067 | 2465 | 2771 | 2992              |
| 41              | 45 | 71 | 8  | 39 | 0  | 0  | 0              | 141  | 1134                 | 1905 | 2360 | 0    | 0                 |
| 47              | 76 | 35 | 74 | 36 | 0  | 0  | 0              | 414  | 1158                 | 1903 | 2562 | 0    | 0                 |
| 76              | 80 | 38 | 49 | 1  | 0  | 0  | 0              | 965  | 1788                 | 2662 | 3070 | 0    | 0                 |
| 78              | 81 | 36 | 78 | 32 | 0  | 0  | 0              | 146  | 871                  | 1716 | 2370 | 0    | 0                 |
| 30              | 3  | 33 | 75 | 36 | 0  | 0  | 0              | 103  | 525                  | 1390 | 2127 | 0    | 0                 |
| 55              | 80 | 57 | 75 | 42 | 78 | 37 | 0              | 555  | 939                  | 1323 | 1780 | 2246 | 2908              |
| 45              | 72 | 36 | 73 | 40 | 0  | 0  | 0              | 635  | 1370                 | 2030 | 2495 | 0    | 0                 |
| 30              | 85 | 41 | 81 | 69 | 0  | 0  | 0              | 166  | 1084                 | 1582 | 2139 | 0    | 0                 |
| 68              | 85 | 36 | 76 | 50 | 75 | 54 | 0              | 164  | 487                  | 1762 | 2174 | 2592 | 3009              |
| 40              | 78 | 51 | 75 | 41 | 0  | 0  | 255            | 1116 | 1508                 | 2307 | 2723 | 0    | 0                 |
| 73              | 75 | 54 | 72 | 42 | 0  | 0  | 0              | 1459 | 1883                 | 2347 | 2811 | 0    | 0                 |
| 47              | 81 | 69 | 48 | 1  | 0  | 0  | 0              | 615  | 1165                 | 1935 | 2371 | 0    | 0                 |
| 72              | 81 | 44 | 78 | 42 | 0  | 0  | 0              | 216  | 651                  | 1236 | 1708 | 0    | 0                 |
| 45              | 76 | 50 | 79 | 41 | 0  | 0  | 0              | 596  | 1016                 | 1483 | 1931 | 0    | 0                 |
| 30              | 5  | 34 | 74 | 39 | 0  | 0  | 0              | 120  | 549                  | 1247 | 1880 | 0    | 0                 |
| 62              | 78 | 45 | 0  | 0  | 0  | 0  | 0              | 1297 | 1755                 | 0    | 0    | 0    | 0                 |
| 76              | 81 | 69 | 80 | 41 | 0  | 0  | 0              | 113  | 663                  | 1317 | 1732 | 0    | 0                 |
| 72              | 75 | 56 | 79 | 43 | 0  | 0  | 0              | 603  | 1026                 | 1499 | 1939 | 0    | 0                 |
| 24              | 7  | 34 | 0  | 0  | 0  | 0  | 0              | 1558 | 1983                 | 0    | 0    | 0    | 0                 |
| 72              | 76 | 47 | 0  | 0  | 0  | 0  | 0              | 1241 | 1664                 | 0    | 0    | 0    | 0                 |
| 76              | 72 | 52 | 73 | 48 | 0  | 0  | 0              | 104  | 486                  | 906  | 1319 | 0    | 0                 |
| 5               | 16 | 35 | 36 | 22 | 0  | 0  | 0              | 108  | 494                  | 1015 | 1334 | 0    | 0                 |

|    |    |    |   |   |   |   |   |      |      |   |   |   |   |
|----|----|----|---|---|---|---|---|------|------|---|---|---|---|
| 40 | 77 | 51 | 0 | 0 | 0 | 0 | 0 | 549  | 933  | 0 | 0 | 0 | 0 |
| 47 | 81 | 52 | 0 | 0 | 0 | 0 | 0 | 1249 | 1639 | 0 | 0 | 0 | 0 |

### Παράδειγμα #9 DEEPSEA\_FUN\_C100\_V50\_HE\_4

Αριθμός λιμανιών: 86

Αριθμός φορτίων : 100

Αριθμός πλοίων : 50

Βέλτιστη τιμή συνολικού κόστους: 220.879.632 €

|                 |    |    |    |               |                |    |  |                   |                      |      |      |                   |      |      |
|-----------------|----|----|----|---------------|----------------|----|--|-------------------|----------------------|------|------|-------------------|------|------|
|                 |    |    |    |               | Λύση μιμητικού |    |  |                   |                      |      |      | Απόκλιση<br>5,16% |      |      |
| Συνολικό Κόστος |    |    |    | 232.272.899 € |                |    |  |                   | 9 φορτία εκτός λύσης |      |      |                   |      |      |
| Διαδρομές       |    |    |    |               |                |    |  | Χρόνοι Αναχώρησης |                      |      |      |                   |      |      |
| 41              | 78 | 29 | 0  | 0             | 0              | 0  |  | 86                | 2267                 | 3275 | 0    | 0                 | 0    | 0    |
| 72              | 57 | 23 | 0  | 0             | 0              | 0  |  | 323               | 3337                 | 4365 | 0    | 0                 | 0    | 0    |
| 13              | 17 | 44 | 46 | 25            | 0              | 0  |  | 138               | 406                  | 1452 | 1947 | 2958              | 0    | 0    |
| 6               | 10 | 72 | 45 | 83            | 0              | 0  |  | 0                 | 249                  | 1294 | 1822 | 2739              | 0    | 0    |
| 6               | 60 | 79 | 75 | 47            | 73             | 37 |  | 0                 | 1067                 | 2064 | 2358 | 2778              | 3263 | 3865 |
| 41              | 66 | 49 | 78 | 34            | 0              | 0  |  | 43                | 1934                 | 2964 | 3630 | 4325              | 0    | 0    |
| 78              | 11 | 79 | 74 | 2             | 0              | 0  |  | 24                | 1411                 | 2425 | 2642 | 3473              | 0    | 0    |
| 76              | 77 | 39 | 0  | 0             | 0              | 0  |  | 0                 | 2243                 | 2942 | 0    | 0                 | 0    | 0    |
| 75              | 60 | 81 | 0  | 0             | 0              | 0  |  | 0                 | 2696                 | 3743 | 0    | 0                 | 0    | 0    |
| 72              | 74 | 42 | 75 | 69            | 0              | 0  |  | 0                 | 2205                 | 2715 | 3414 | 3964              | 0    | 0    |
| 24              | 25 | 52 | 75 | 42            | 0              | 0  |  | 192               | 381                  | 1407 | 2127 | 2591              | 0    | 0    |
| 50              | 49 | 59 | 71 | 20            | 0              | 0  |  | 372               | 1210                 | 2133 | 2884 | 3235              | 0    | 0    |
| 72              | 81 | 45 | 81 | 56            | 74             | 39 |  | 0                 | 589                  | 1047 | 1661 | 2092              | 2629 | 3262 |
| 30              | 5  | 51 | 72 | 53            | 0              | 0  |  | 0                 | 983                  | 2003 | 2405 | 2828              | 0    | 0    |
| 62              | 59 | 42 | 72 | 54            | 0              | 0  |  | 68                | 180                  | 1048 | 2195 | 2619              | 0    | 0    |
| 73              | 74 | 57 | 75 | 38            | 0              | 0  |  | 157               | 703                  | 1141 | 2280 | 3110              | 0    | 0    |
| 76              | 73 | 38 | 0  | 0             | 0              | 0  |  | 917               | 1548                 | 2299 | 0    | 0                 | 0    | 0    |
| 19              | 6  | 46 | 76 | 53            | 0              | 0  |  | 0                 | 438                  | 1432 | 2359 | 2779              | 0    | 0    |
| 73              | 75 | 61 | 0  | 0             | 0              | 0  |  | 471               | 1474                 | 2430 | 0    | 0                 | 0    | 0    |
| 76              | 73 | 35 | 0  | 0             | 0              | 0  |  | 917               | 1547                 | 2225 | 0    | 0                 | 0    | 0    |
| 46              | 70 | 59 | 72 | 34            | 0              | 0  |  | 0                 | 1115                 | 1505 | 2581 | 3279              | 0    | 0    |
| 58              | 75 | 48 | 74 | 57            | 80             | 45 |  | 0                 | 479                  | 857  | 1277 | 1715              | 2242 | 2692 |
| 12              | 10 | 48 | 73 | 34            | 0              | 0  |  | 0                 | 166                  | 1214 | 1632 | 2267              | 0    | 0    |
| 72              | 77 | 32 | 74 | 36            | 77             | 44 |  | 0                 | 100                  | 768  | 1421 | 2087              | 2820 | 3246 |
| 62              | 36 | 69 | 81 | 32            | 0              | 0  |  | 0                 | 485                  | 1158 | 1717 | 2371              | 0    | 0    |
| 71              | 37 | 44 | 77 | 37            | 0              | 0  |  | 0                 | 803                  | 1611 | 2293 | 2975              | 0    | 0    |
| 33              | 37 | 45 | 76 | 39            | 0              | 0  |  | 0                 | 125                  | 891  | 1926 | 2625              | 0    | 0    |
| 76              | 77 | 39 | 81 | 46            | 0              | 0  |  | 917               | 1339                 | 2038 | 2726 | 3134              | 0    | 0    |
| 41              | 74 | 51 | 80 | 42            | 0              | 0  |  | 0                 | 556                  | 993  | 2343 | 2806              | 0    | 0    |
| 72              | 78 | 54 | 80 | 69            | 75             | 69 |  | 0                 | 123                  | 554  | 1186 | 1735              | 2337 | 2887 |

|    |    |    |    |    |   |   |     |      |      |      |      |   |   |
|----|----|----|----|----|---|---|-----|------|------|------|------|---|---|
| 5  | 79 | 69 | 77 | 54 | 0 | 0 | 0   | 1051 | 1617 | 2425 | 2841 | 0 | 0 |
| 62 | 81 | 37 | 75 | 41 | 0 | 0 | 68  | 1324 | 1993 | 2843 | 3266 | 0 | 0 |
| 75 | 80 | 39 | 76 | 50 | 0 | 0 | 0   | 188  | 893  | 1922 | 2342 | 0 | 0 |
| 62 | 6  | 84 | 74 | 49 | 0 | 0 | 68  | 310  | 695  | 1937 | 2414 | 0 | 0 |
| 75 | 75 | 40 | 78 | 45 | 0 | 0 | 0   | 249  | 670  | 1150 | 1608 | 0 | 0 |
| 45 | 81 | 45 | 79 | 55 | 0 | 0 | 0   | 700  | 1158 | 1780 | 2173 | 0 | 0 |
| 71 | 73 | 53 | 73 | 47 | 0 | 0 | 0   | 915  | 1383 | 1910 | 2381 | 0 | 0 |
| 62 | 34 | 1  | 74 | 49 | 0 | 0 | 0   | 540  | 1143 | 1903 | 2380 | 0 | 0 |
| 72 | 77 | 32 | 76 | 46 | 0 | 0 | 0   | 100  | 768  | 1790 | 2195 | 0 | 0 |
| 72 | 79 | 42 | 74 | 48 | 0 | 0 | 0   | 221  | 710  | 1498 | 1919 | 0 | 0 |
| 8  | 73 | 54 | 0  | 0  | 0 | 0 | 364 | 1345 | 1806 | 0    | 0    | 0 | 0 |
| 33 | 35 | 62 | 81 | 54 | 0 | 0 | 0   | 153  | 469  | 1813 | 2244 | 0 | 0 |
| 35 | 72 | 49 | 0  | 0  | 0 | 0 | 0   | 1052 | 1483 | 0    | 0    | 0 | 0 |
| 72 | 72 | 39 | 78 | 46 | 0 | 0 | 0   | 101  | 804  | 1499 | 1915 | 0 | 0 |
| 58 | 76 | 46 | 0  | 0  | 0 | 0 | 0   | 608  | 1013 | 0    | 0    | 0 | 0 |
| 41 | 73 | 44 | 0  | 0  | 0 | 0 | 0   | 498  | 970  | 0    | 0    | 0 | 0 |
| 44 | 74 | 46 | 0  | 0  | 0 | 0 | 0   | 1201 | 1647 | 0    | 0    | 0 | 0 |
| 41 | 77 | 69 | 0  | 0  | 0 | 0 | 0   | 771  | 1320 | 0    | 0    | 0 | 0 |
| 75 | 81 | 55 | 81 | 57 | 0 | 0 | 0   | 124  | 501  | 1319 | 1719 | 0 | 0 |
| 72 | 75 | 42 | 0  | 0  | 0 | 0 | 0   | 102  | 566  | 0    | 0    | 0 | 0 |

## 7. Γενικός Επίλογος

Συμπερασματικά, η επίλυση του συγκεκριμένου προβλήματος και ο εντοπισμός ικανοποιητικής λύσης υπήρξε απαιτητικό, εξαιτίας της πολυπλοκότητας του σε πολλά σημεία. Ο αυξημένος αριθμός περιορισμών δυσκόλεψαν αρκετά τη δημιουργία της αρχικής λύσης. Ο GRASP έπρεπε για κάθε σύνολο δεδομένων να δημιουργεί εφικτές λύσεις. Συνεπώς, κάθε εισαγωγή φορτίου έπρεπε να σέβεται όλους τους περιορισμούς. Εκτός αυτού, στα αρχικά στάδια ανάπτυξης του κώδικα παρατηρήθηκε πως η μεγάλη ποικιλία παραδειγμάτων έκαναν το πρόγραμμα να αντιδρά διαφορετικά σε σύνολα δεδομένων με ιδιαιτερότητες. Για αυτό το λόγο μικρές τροποποιήσεις έγιναν και μετά την ολοκλήρωση του κώδικα. Τροποποιήσεις που δεν επηρεάζουν τόσο τη δομή του αλγορίθμου αλλά τον τρόπο διαχείρισης των δεδομένων. Η διαχείριση των δεδομένων αποτέλεσε ένα απαιτητικό αλλά και σημαντικό κομμάτι της διαδικασίας επίλυσης του προβλήματος. Ο αλγόριθμος θα έπρεπε να κατασκευαστεί με τέτοιο τρόπο ώστε να έχει τη δυνατότητα προσαρμογής στις διάφορες παραλλαγές των δεδομένων. Το μέγεθος των παραδειγμάτων επίσης ποικίλει σε αυτού του είδους προβλήματα, ως εκ τούτου το πρόγραμμα όφειλε να είναι δυναμικό ως προς το μέγεθος των μεταβλητών και τη διαχείριση της μνήμης, ώστε να είναι εφικτή η επίλυση οποιοδήποτε προβλήματος ανεξαρτήτως μεγέθους. Η εφαρμογή της μεθοδολογίας επίλυσης έπρεπε να γίνει σε παραδείγματα πολλών μεγεθών, ώστε να ελεγχθούν οι δυνατότητες του σε απαιτητικά δεδομένα. Η δυσκολία εντοπισμού ποιοτικής λύσης αυξάνεται ανάλογα με το μέγεθος και τη πολυπλοκότητα του προβλήματος.

Βοηθητικό υπήρξε το γεγονός ότι η συγκεκριμένη διπλωματική εργασία αφορούσε μόνο προβλήματα πλήρους φορτίου (FUN). Συνεπώς, κάποιοι περιορισμοί της μοντελοποίησης δε λήφθηκαν υπόψιν κατά την επίλυση, μειώνοντας έτσι τη πολυπλοκότητα του προβλήματος και τον όγκο των περιορισμών. Σε περίπτωση που επιχειρούσαμε την κάλυψη και προβλημάτων μικτού φορτίου η ήδη αυξημένη δυσκολία επίλυσης θα ξεπερνούσε τα όρια μίας διπλωματικής εργασίας και σε χρόνο ολοκλήρωσης αλλά και σε πολυπλοκότητα. Για αυτό το λόγο επιλέχθηκε να γίνει επίλυση μόνο προβλημάτων πλήρους φορτίου.

Επιπλέον δυσκολία βρέθηκε στην αντιμετώπιση κάποιων περιορισμών και στην επισήμανση κάποιων λεπτομερειών του προβλήματος. Συγκεκριμένα, η ύπαρξη χρονικών παραθύρων στα λιμάνια παραλαβής και στα λιμάνια παράδοσης αύξανε τον αριθμό των απαιτούμενων ελέγχων. Για κάθε εισαγωγή κόμβου σε κάποια διαδρομή θεωρητικά απαιτούνταν ο έλεγχος τεσσάρων χρονικών ορίων. Παρ' όλα αυτά μετά από δοκιμές παρατηρήθηκε πως εφόσον ένα πλοίο έχει ολοκληρώσει την φόρτωση ενός εμπορεύματος πριν το άνω χρονικό όριο του εκάστοτε λιμανιού παραλαβής, τότε θα είχε αρκετό χρόνο ώστε να παραδώσει το φορτίο στο αντίστοιχο λιμάνι. Αυτή η παρατήρηση βοήθησε στη τροποποίηση του προγράμματος, ώστε να είναι μειωθεί η πολυπλοκότητα του, να γίνουν λιγότεροι έλεγχοι και με αυτό τον τρόπο να φτάσουμε στη μείωση του χρόνου εκτέλεσης. Ένας άλλος περιορισμός, ο οποίος δεν ήταν τόσο δύσκολος στη διαχείριση του αλλά περιόριζε τον αλγόριθμο στο να φτάσει σε ποιοτικές λύσεις, ήταν ο περιορισμός συμβατότητας φορτίου – πλοίου. Ο συγκεκριμένος αλγόριθμος έκανε τα πράγματα δύσκολα κατά τη φάση της επιδιόρθωσης των λύσεων, κάνοντας πολλές διαδρομές που προκύπταν κατά τη διασταύρωση των λύσεων να είναι μη εφικτές. Για την αντιμετώπιση αυτού του εμποδίου έπρεπε μη συμβατά φορτία να αφαιρεθούν από τη λύση. Εκτός αυτού, προσεκτική διαχείριση απαιτούσαν και επιπλέον λεπτομέρειες που υπήρχαν στα πρόβληματα, όπως η σωστή διαχείριση των χρόνων εξυπηρέτησης στα λιμάνια αλλά και οι χρόνοι που κάθε πλοίο είναι διαθέσιμό. Τέτοιου είδους λεπτομέρειες επηρέασαν σημαντικά τα αποτελέσματα του αλγορίθμου, οδηγώντας τον σε μη εφικτές λύσεις. Ο εντοπισμός λογικών λαθών και η αντιμετώπιση τους λοιπόν, ήταν ένα άλλο κομμάτι της ανάπτυξης του κώδικα που ήταν απαιτητικό εξαιτίας των παραπάνω χαρακτηριστικών του προβλήματος που αναφέρθηκαν.

Πέραν των δυσκολιών που υπήρξαν, οι οποίες είναι αναμενόμενες σε ένα τέτοιου είδους πρόβλημα δρομολόγησης. Ο συνδυασμός του αλγορίθμου δημιουργίας αρχικών λύσεων και του αλγορίθμου βελτίωσης του πληθυσμού λύσεων είναι αρκετά αποδοτικός, σύμφωνα με τα αποτελέσματα. Αυτό συμβαίνει καθώς, ο GRASP κατασκευάζει ποικίλες αρχικές λύσεις μέτριας ποιότητας, κάτι που δίνει τη δυνατότητα στο μιμητικό να μην ξεκινήσει την αναζήτηση του στο χώρο λύσεων, ούτε από μη ποιοτικές περιοχές (κακές αρχικός πληθυσμός) λύσεων αλλά ούτε από περιοχές που δε θα του επιτρέψουν να διευρύνει την αναζήτηση του (πολύ ποιοτικός αρχικός πληθυσμός). Συνεπώς, ο μιμητικός μπόρεσε να διασπείρει την αναζήτηση με πολλές ποιοτικές περιοχές λύσεων και με τη βοήθεια της τοπικής αναζήτησης μπόρεσε να δώσει ικανοποιητικά αποτελέσματα.

Επιπλέον, παρ' όλο που η συγκεκριμένη μεθοδολογία απέδωσε, υπάρχουν κάποιες βελτιώσεις που θα μπορούσαν να γίνουν. Οι βελτιώσεις αφορούν κυρίως τους αλγορίθμους επίλυσης και όχι τον προγραμματιστικό τρόπο ανάπτυξης του αλγορίθμου. Συγκεκριμένα, κατά τη διασταύρωση των λύσεων, μπορεί να δοκιμασθεί η συστηματική επιλογή των διαδρομών που θα διασταυρωθούν. Με αυτό τον τρόπο είναι δυνατό να μειωθούν οι απαιτούμενοι έλεγχοι και επιδιορθώσεις άλλα και να μειωθεί ο χρόνος σύγκλισης του γενετικού. Εκτός αυτού, στον αλγόριθμο τοπικής



αναζήτησης 1-0 επανατοποθέτηση, θα μπορούσε να γίνει μετατροπή, ώστε να ελέγχονται όλες οι πιθανές ανατοποθετήσεις των ζευγαριών. Με άλλα λόγια, ο αλγόριθμος να ελέγχει την επανατοποθέτηση όλων των ζευγαριών από μία γεμάτη διαδρομή σε οποιαδήποτε διαδρομή της λύσης. Η αύξηση του χρόνου εκτέλεσης που συνεπάγεται με αυτή τη μετατροπή, δεν αποτελεί σημαντικό μειονέκτημα για το συγκεκριμένο τύπο προβλημάτων. Τα συγκεκριμένα αποτελέσματα είναι χρήσιμα στη φάση σχεδιασμού και χρονικού προγραμματισμού μία εταιρίας και όχι σε πραγματικό χρόνο λειτουργίας της ΕΑ. Για αυτό το λόγο, θέματα που αφορούν το χρόνο εκτέλεσης δεν είναι τόσο σημαντικά.

Η επιλογή επίλυσης του συγκεκριμένου προβλήματος δρομολόγησης και χρονικού προγραμματισμού φορηγών πλοίων έγινε μετά τη παρακολούθηση του προπτυχιακού μαθήματος *Σχεδιασμός και Βελτιστοποίηση της Εφοδιαστικής Αλυσίδας του Δρ. Ιωάννη Μαρινάκη*, στο 9<sup>ο</sup> εξάμηνο της σχολής Μηχανικών Παραγωγής και Διοίκησης. Σκοπός ήταν η επιλογή ενός προβλήματος δρομολόγησης με ιδιαίτερα χαρακτηριστικά και αυξημένη δυσκολία, που να είναι άμεσα συνδεδεμένο με την αγορά εργασίας και να προσομοιάζει σε μεγάλο βαθμό προβλήματα πραγματικών συνθηκών. Έτσι, σε συνδυασμό με την αύξηση της δραστηριότητας στον συγκεκριμένο τομέα τα τελευταία χρόνια, επιλέχθηκε η ενασχόληση με το συγκεκριμένο τύπο προβλήματος.

## Βιβλιογραφία

Προβλήματα Δρομολόγησης Οχημάτων στη Διαχείριση Εφοδιαστικής Αλυσίδας, Ιωάννης Μαρινάκης, Μαγδαληνή Μαρινάκη, Αθανάσιος Μυγδαλάς, Εκδόσεις Νέων Τεχνολογιών, Οκτώβριος 2019.

Σημειώσεις Προπτυχιακού Μαθήματος: Σχεδιασμός και Βελτιστοποίηση της Εφοδιαστικής Αλυσίδας, Ιωάννης Μαρινάκης, 2020.

HEMMATI, Ahmad, et al. Benchmark suite for industrial and tramp ship routing and scheduling problems. *INFOR: Information Systems and Operational Research*, 2014, 52.1: 28-38.

HOMSI, Gabriel, et al. Industrial and tramp ship routing problems: Closing the gap for real-scale instances. *European Journal of Operational Research*, 2020, 283.3: 972-990.

NORSTAD, Inge; FAGERHOLT, Kjetil; LAPORTE, Gilbert. Tramp ship routing and scheduling with speed optimization. *Transportation Research Part C: Emerging Technologies*, 2011, 19.5: 853-865.

TIRADO, Gregorio, et al. Heuristics for dynamic and stochastic routing in industrial shipping. *Computers & Operations Research*, 2013, 40.1: 253-263.

FRÉMONT, Antoine. Shipping lines and logistics. *Transport Reviews*, 2009, 29.4: 537-554.

CHRISTIANSEN, Marielle, et al. Maritime transportation. *Handbooks in operations research and management science*, 2007, 14: 189-284. DOI: 10.1016/S0927-0507(06)14004-9

