

Συνδυαστικός αλγόριθμος μη εποπτευόμενης μάθησης – συσταδοποίησης και αναζήτησης μεταβλητής γειτνίασης για την επίλυση του προβλήματος δρομολόγησης οχημάτων με περιορισμένη χωρητικότητα

Τρανίδης Χρήστος
ΑΕΜ: 19074

Πανεπιστήμιο Μακεδονίας

Θεσσαλονίκη 2020



Συνδυαστική εφαρμογή δύο μεθόδων:

- ενός αλγορίθμου συσταδοποίησης, της κατηγορίας centroid based
- ενός μεθευρετικού αλγορίθμου αναζήτησης μεταβλητής γειτνίασης (VNS)

για την επίλυση του προβλήματος δρομολόγησης οχημάτων με περιορισμένη χωρητικότητα CVRP.

- ελαχιστοποίηση του τελικού συνολικού κόστους διαδρομών σε ικανοποιητικό χρόνο.

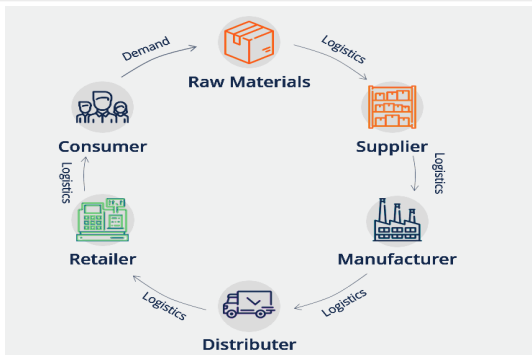
Επαλήθευση ποιότητας λύσης:

- σύγκριση με τις καλύτερες λύσεις, οι οποίες έχουν επιτευχθεί επίσημα για την κάθε περίπτωση δεδομένων CVRP (benchmarks).

Η εφοδιαστική αλυσίδα (supply chain)

Ορισμός εφοδιαστικής αλυσίδας

Στον χώρο των επιχειρήσεων και της οικονομίας, η αλυσίδα εφοδιασμού ορίζεται ως ένα σύστημα οργανισμών, ανθρώπων, δραστηριοτήτων, πληροφοριών και πόρων που σχετίζονται με τη μετακίνηση ενός προϊόντος ή μίας υπηρεσίας από τον προμηθευτή στον πελάτη.



Την σημερινή εποχή, οι αλυσίδες εφοδιασμού καλύπτουν ανάγκες πελατών παγκοσμίως. Ωστόσο, η εξυπηρέτηση των πελατών θα πρέπει να γίνεται με τον καλύτερο δυνατό τρόπο.

- γρήγορος χρόνος εξυπηρέτησης,
- παράλληλα, ελαχιστοποίηση του κόστους δαπανών της αλυσίδας εφοδιασμού.
- **οργάνωση της διαδικασίας μεταφοράς των προϊόντων με τον καλύτερο δυνατό τρόπο.**

Ο τομέας των μεταφορών αντιπροσωπεύει περίπου το ήμισυ των συνολικών δαπανών μίας εφοδιαστικής αλυσίδας.

- περίπου το 7% του ΑΕΠ της Ευρώπης προέρχεται από τις μεταφορές.
- το 60% της παγκόσμιας κατανάλωσης καυσίμου γίνεται εξαιτίας των μεταφορών.

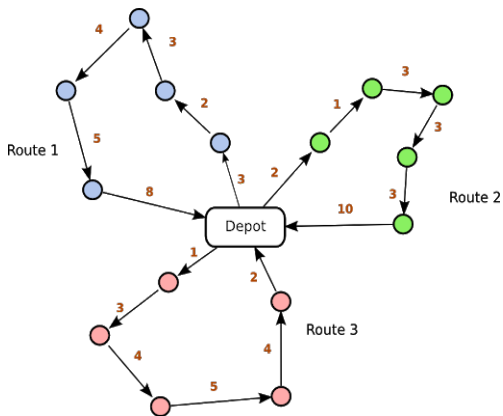
βέλτιστος προγραμματισμός της διαδικασίας μεταφοράς των προϊόντων - δρομολόγηση οχημάτων.

Ορισμός του προβλήματος VRP

Επιδιώκεται ο προσδιορισμός ενός συνόλου διαδρομών για ένα σύνολο οχημάτων τα οποία βρίσκονται σε ένα η περισσότερα σημεία αφετηρίας και πρέπει να εξυπηρετήσουν έναν αριθμό γεωγραφικά διασκορπισμένων πελατών.

Τελικός στόχος:

Η παράδοση αγαθών στους πελάτες, έχοντας διανύσει την μικρότερη δυνατή απόσταση στον λιγότερο χρόνο, ξεκινώντας από την αφετηρία και επιστρέφοντας και πάλι σε αυτήν.

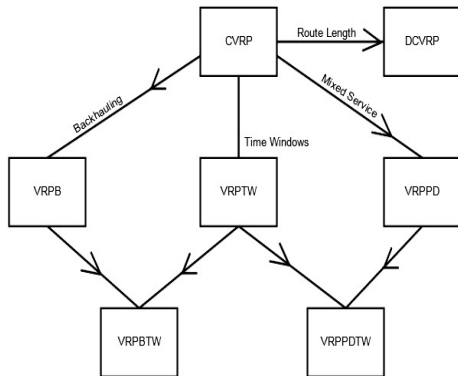


Σχήμα: Γενική μορφή του προβλήματος δρομολόγησης οχημάτων.

- αποτελεί ένα από τα πιο πολυσυζητημένα προβλήματα βελτιστοποίησης.
- ανήκει στην κατηγορία των NP-hard προβλημάτων,
- επομένως, δεν υπάρχει κάποιος αλγόριθμος ο οποίος επιλύει το πρόβλημα σε πολυωνυμικό χρόνο.

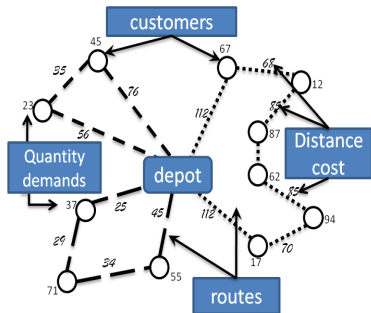
Παραλλαγές του προβλήματος VRP

- ελαχιστοποίηση συνολικού κόστους μεταφοράς
- ελαχιστοποίηση του αριθμού των οχημάτων τα οποία εξυπηρετούν τους πελάτες.
- ελαχιστοποίηση των μεταβολών στον χρόνο ταξιδιού καθώς και στο φορτίο των οχημάτων.



Σχήμα: Κατηγορίες VRP

Ορισμός και ανάλυση του προβλήματος CVRP



Σχήμα: Γράφος προβλήματος CVRP.

- Γράφος $G = (V, E)$
- $V = \{0, \dots, n\}$ ένα σύνολο σημείων
- $E = (i, j) : i, j \in V, i < j$ ένα σύνολο ακμών

$$\min \sum_{k=1}^m \sum_{i=0}^n \sum_{j=0}^n d_{ij} x_{ij}^k \quad (1)$$

$$\sum_{i=1}^n y_i^k \leq Q \quad \forall k = 1, 2, \dots, m \quad (2)$$

$$\sum_{k=1}^m y_i^k = 1 \quad \forall i = 1, 2, \dots, n \quad (3)$$

$$\sum_{k=1}^m y_i^k = m \quad \text{for } i = 0 \quad (4)$$

$$\sum_{i=0}^n x_{ij}^k = y_i^k \quad \forall j = 0, 1, \dots, n; \quad \forall k = 1, 2, \dots, m \quad (5)$$

$$\sum_{j=0}^n x_{ij}^k = y_i^k \quad \forall i = 0, 1, \dots, n; \quad \forall k = 1, 2, \dots, m \quad (6)$$

$$\sum_{i,j \in S} x_{ij}^k \leq |S| - 1 \quad S \subset \{1, 2, \dots, n\}; \quad |S| \geq 2; \quad \forall k = 1, 2, \dots, m \quad (7)$$

$$x_{ij}^k \in [0, 1] \quad \forall i = 0, 1, \dots, n; \quad \forall j = 0, 1, \dots, n; \quad \forall k = 1, 2, \dots, m \quad (8)$$

$$y_i^k \in \{0, 1\} \quad \forall i = 0, 1, \dots, n; \quad \forall k = 1, 2, \dots, m \quad (9)$$

- construction heuristics
- improvement heuristics
- metaheuristics

Τα construction heuristics χρησιμοποιούνται για να κατασκευαστεί μια αρχική λύση. Η ποιότητα της λύσης που προκύπτει είναι αρκετά χαμηλή.

construction heuristics

- Sequential Insertion Heuristic
- Improved Parallel Insertion
- **Nearest Neighbour**

Procedure Nearest Neighbor Heuristic

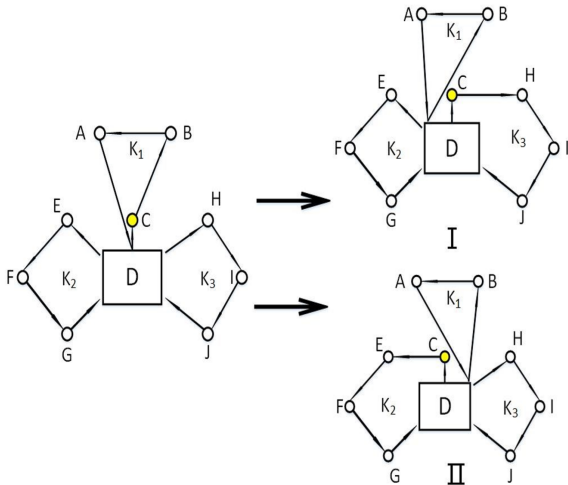
```
10 Select an arbitrary node  $j$ 
20 Set  $l = j$  and  $W = \{1, 2, \dots, n\} \setminus \{j\}$ 
30 While  $W \neq \emptyset$  do
40   Let  $j \in W$  such that  $c_{lj} = \min\{c_{li} \mid i \in W\}$ 
50   Connect  $l$  to  $j$  and set  $W = W \setminus \{j\}$  and  $l = j$ .
60 Connect  $l$  to the node selected in 10 to form a Hamiltonian cycle.
```

Σχήμα: Ψευδοκώδικας αλγορίθμου nearest neighbour.

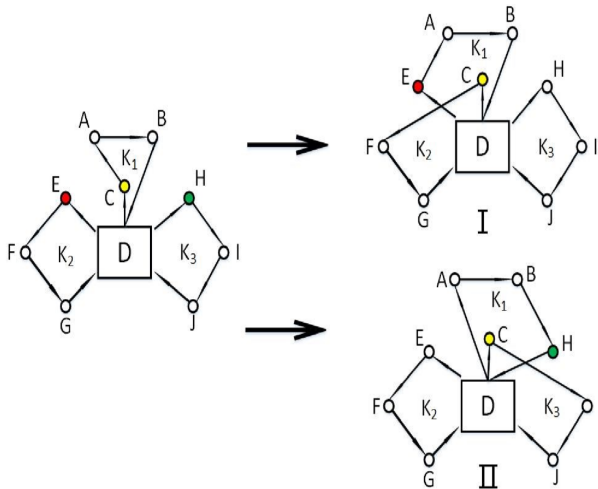
Τα improvement heuristics, χρησιμοποιούνται για να βελτιώσουν την αρχική λύση η οποία κατασκευάστηκε με την χρήση κάποιου construction heuristic.

Τρία από τα πιο σημαντικά και ευρέως χρησιμοποιούμενα improvement heuristics είναι τα εξής:

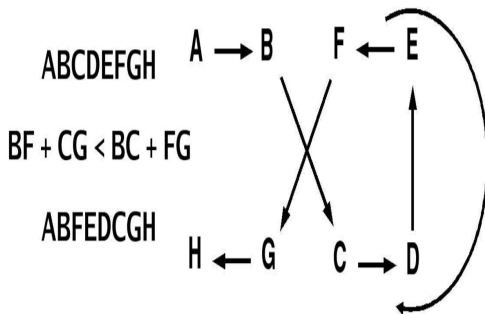
- relocate
- exchange
- 2 – Opt



Σχήμα: Αναπαράσταση αλγορίθμου relocate.



Σχήμα: Αναπαράσταση αλγορίθμου exchange.

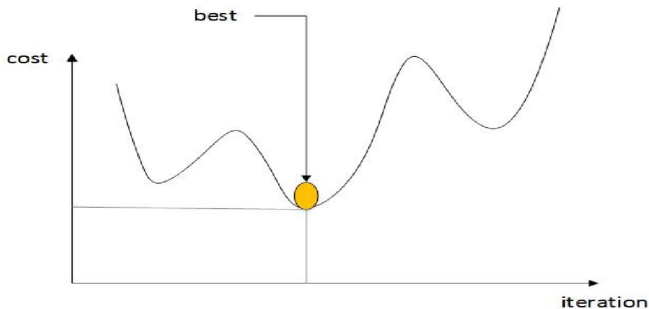


Σχήμα: Αναπαράσταση αλγορίθμου 2 – Opt.

Metaheuristics

Οι συγκεκριμένοι αλγόριθμοι πολλές φορές δέχονται λύσεις κατώτερης ποιότητας.

Πού εξυπηρετεί αυτό;



Σχήμα: Παρουσία τοπικών ελαχίστων και ολικού ελάχιστου αντικειμενικής συνάρτησης.

- 1 Local search
 - Tabu search
 - Simulated annealing
 - VNS
- 2 Population based
 - Genetic algorithms
- 3 Learning mechanisms
 - Neural networks
 - Ant colony optimisation

Variable neighborhood search (VNS)

Initialization. Select the set of neighborhood structures \mathcal{N}_k , for $k = 1, \dots, k_{max}$, that will be used in the search; find an initial solution x ; choose a stopping condition;

Repeat the following sequence until the stopping condition is met:

(1) Set $k \leftarrow 1$;

(2) Repeat the following steps until $k = k_{max}$:

(a) Shaking. Generate a point x' at random from the k^{th} neighborhood of x ($x' \in \mathcal{N}_k(x)$);

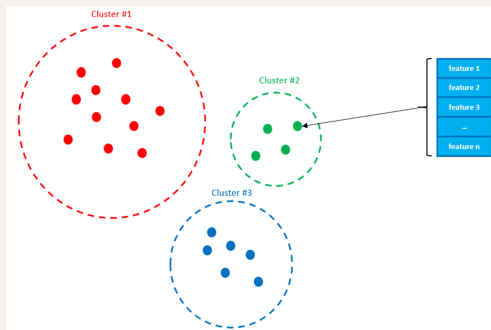
(b) Local search. Apply some local search method with x' as initial solution; denote with x'' the so obtained local optimum;

(c) Move or not. If the local optimum x'' is better than the incumbent x , move there ($x \leftarrow x''$), and continue the search with \mathcal{N}_1 ($k \leftarrow 1$); otherwise, set $k \leftarrow k + 1$;

Σχήμα: Basic VNS

Ορισμός συσταδοποίησης

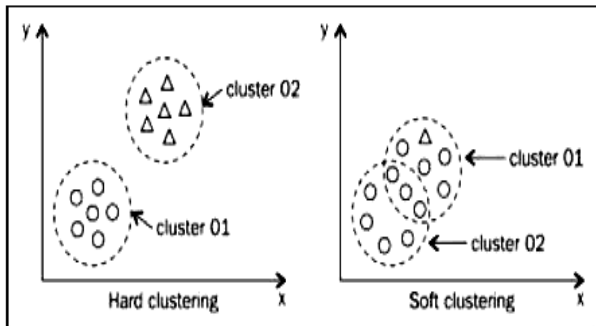
Συσταδοποίηση, ονομάζεται η διαδικασία ομαδοποίησης ενός συνόλου αντικειμένων, με τέτοιον τρόπο ώστε τα αντικείμενα της ίδιας ομάδας να είναι πιο παρόμοια μεταξύ τους σε σχέση με τα αντικείμενα που ανήκουν σε άλλες ομάδες.



Σχήμα: Συσταδοποίηση ομάδας δεδομένων.

Η συσταδοποίηση χωρίζεται σε δύο κατηγορίες:

- Hard clustering
- Soft clustering

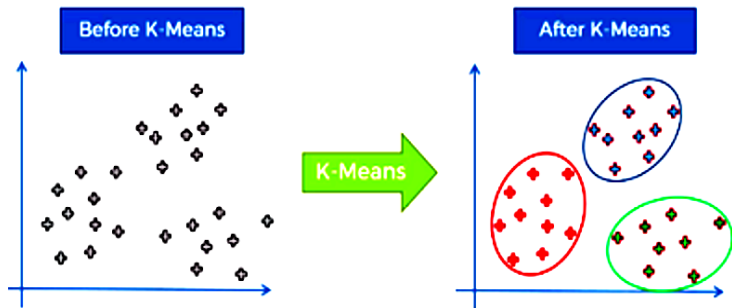


Σχήμα: Παράδειγμα hard - soft clustering.

Ως προς τον τρόπο λειτουργίας και την μεθοδολογία συσταδοποίησης, διακρίνονται 4 κατηγορίες αλγορίθμων συσταδοποίησης:

- Centroid based clustering
- Density based clustering
- Hierarchical based clustering
- Grid based clustering

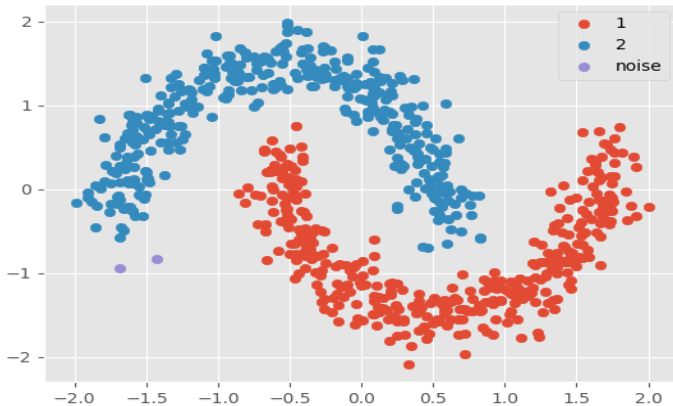
- 1 αρχικά, επιλέγεται μια σειρά κλάσεων / ομάδων για να χρησιμοποιηθεί και να οριστούν τυχαία τα κεντρώα αυτών. Ο αριθμός των συστάδων επιλέγεται στην αρχή με βάση την κατανομή των σημείων στον χώρο.
- 2 για κάθε σημείο, υπολογίζεται η απόσταση αυτού από τα κεντρώα που ορίστηκαν στο πρώτο βήμα και εισάγεται στην συστάδα του κεντρώου που βρίσκεται πλησιέστερα.
- 3 έπειτα υπολογίζεται η νέα θέση των κεντρώων, με βάση την μέση τιμή των δεδομένων που ανήκουν στην συστάδα αυτών.
- 4 επανέλαβε τα παραπάνω βήματα μέχρι να μην παρατηρείται μετατόπιση των κεντρώων.



K-Means Clustering

Σχήμα: Παράδειγμα συσταδοποίησης K-Means.

- 1 ο DBSCAN αρχίζει με ένα αυθαίρετο σημείο εκκίνησης που δεν έχει επισκεφθεί. Η γειτονιά αυτού του σημείου εξάγεται χρησιμοποιώντας μια απόσταση ϵ (όλα τα σημεία που βρίσκονται εντός της απόστασης ϵ θεωρούνται σημεία κοινής γειτονιάς).
- 2 εάν υπάρχει επαρκής αριθμός σημείων (σύμφωνα με έναν αριθμό `minPoints`) εντός αυτής της γειτονιάς τότε ξεκινά η διαδικασία ομαδοποίησης και το πρώτο σημείο δεδομένων γίνεται το πρώτο σημείο της συστάδας. Διαφορετικά, το σημείο θα χαρακτηριστεί ως θόρυβος (αργότερα αυτό το θορυβώδες σημείο μπορεί να γίνει σημείο μίας συστάδας). Και στις δύο περιπτώσεις το σημείο αυτό επισημαίνεται ως "επισκέφθηκε".
- 3 Για αυτό το πρώτο σημείο της νέας συστάδας, τα σημεία εντός της απόστασης ϵ εισάγονται επίσης στην ίδια συστάδα. Αυτή η διαδικασία ελέγχου πραγματοποιείται για όλα τα σημεία εντός της απόστασης ϵ .



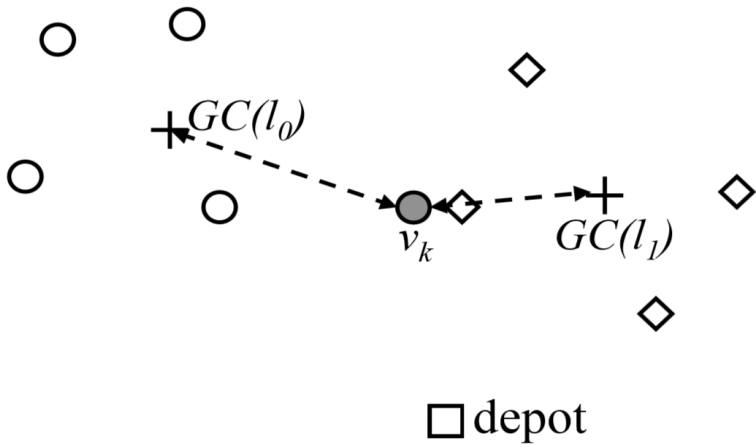
Σχήμα: Παράδειγμα συσταδοποίησης DBSCAN.

Ανήκει στην κατηγορία των centroid based αλγορίθμων συσταδοποίησης και αποτελείται από δύο φάσεις:

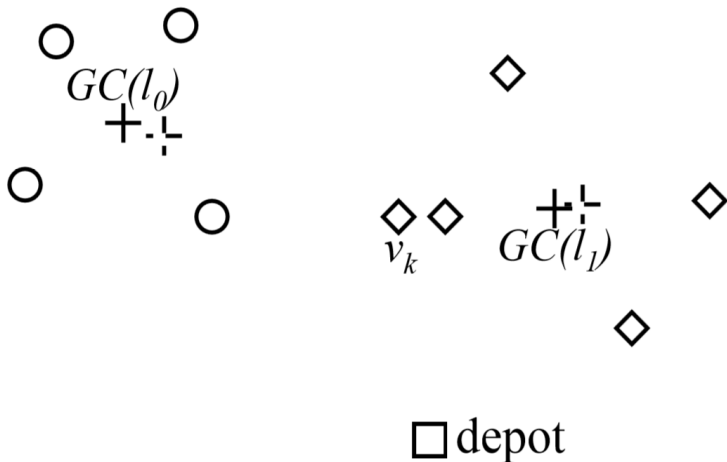
- δημιουργία συστάδων
- βελτιστοποίηση συστάδων

Set $i = 0$ and $Q = \text{truck_capacity}$
While (unvisited node exist)
 $v_j =$ the farthest node among un-clustered nodes from the depot
 Generate cluster l_i with v_j
 Set capacity of $l_i = Q$
 While (demand of v_j does not exceed available capacity of l_i)
 Add v_j to l_i
 Reduce available capacity of l_i by demand of v_j
 Calculate $\text{GC}(l_i)$
 $v_j =$ the closest node among un-clustered nodes from $\text{GC}(l_i)$
 End while
 $i = i+1$
End while

Σχήμα: Ψευδοκώδικας δημιουργίας συστάδων.



Σχήμα: Σημείο συστάδας το οποίο πρέπει να μετακινηθεί σε άλλη.



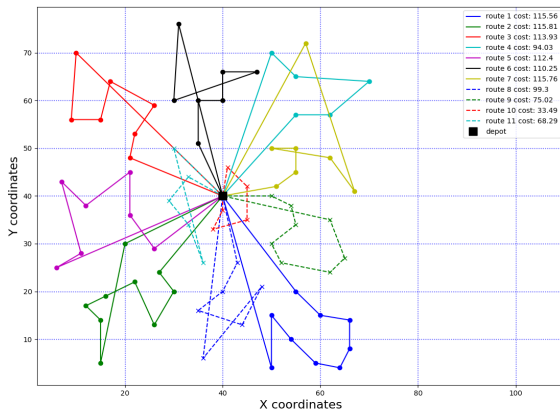
Σχήμα: Συστάδες μετά την βελτιστοποίηση.

Επίλυση του προβλήματος σε δύο στάδια:

- εφαρμογή αλγορίθμου συσταδοποίησης για την ομαδοποίηση των δεδομένων. Βελτιστοποίηση των αποστάσεων της κάθε συστάδας ξεχωριστά, πολλαπλά προβλήματα TSP.
- εφαρμογή αλγορίθμου μεταβλητής γειτονιάς αναζήτησης (VNS). Τα δεδομένα των συστάδων αλληλεπιδρούν, επιτυγχάνοντας βελτιστοποίηση στο συνολικό κόστος των αποστάσεων (CVRP).

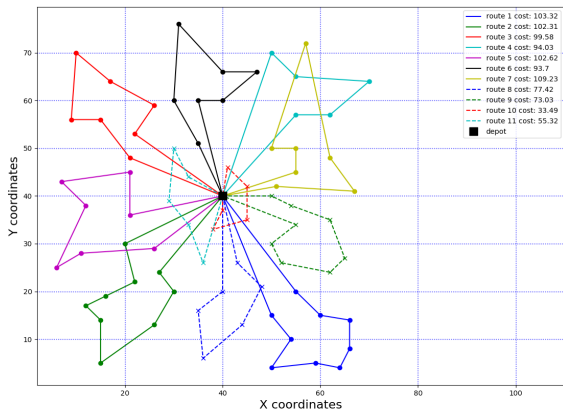
- ο αριθμός των επαναλήψεων του local search κατά την εκτέλεση της VNS.
- στο shaking phase, το οποίο αποτελεί τμήμα της VNS για την αλλαγή γειτονιάς, χρησιμοποιήθηκε ο τελεστής relocate.
- Τέλος, ένας πολύ σημαντικός παράγοντας όσον αφορά την εκτέλεση της VNS, είναι η επιλογή του αριθμού k_{max} στην VNS.
- ο χρόνος εκτέλεσης της VNS αποτελεί επίσης έναν παράγοντα ο οποίος ορίζεται αρχικά, αφού μαζί με την παράμετρο k_{max} , αποτελούν τις δύο συνθήκες τερματισμού του αλγορίθμου VNS.

Συσταδοποίηση δεδομένων και εφαρμογή του nearest neighbour construction heuristic.



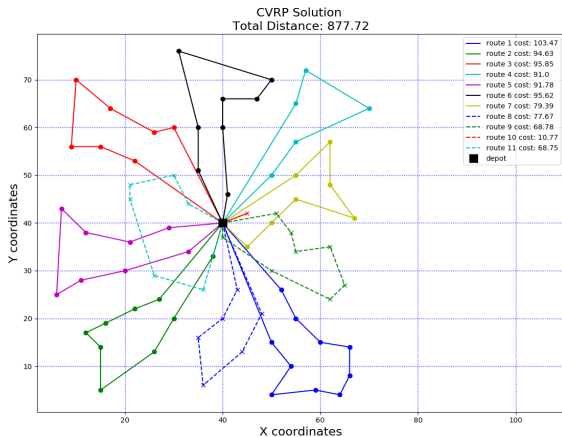
Σχήμα: Συσταδοποίηση δεδομένων και κατασκευή αρχικής λύσης.

Εφαρμογή του τελεστή 2 – Opt για την βελτιστοποίηση της διαδρομής στην κάθε συστάδα ξεχωριστά.



Σχήμα: Αποτελέσματα μετά την εφαρμογή του τελεστή 2 – Opt.

Εφαρμογή τελεστών relocate - exchange.



Σχήμα: Τελικό αποτέλεσμα βελτιστοποίησης μετά την εφαρμογή της VNS.

Πίνακας: Σετ δεδομένων A

Instance	__ BKS __	__ Best __	Cpu Time (s)	% Dev
A-n60-k9	1354	1380.86	167	1.98
A-n61-k9	1034	1045.01	171	1.06
A-n62-k8	1288	1323.65	153	2.77
A-n63-k9	1616	1642.99	103	1.67
A-n63-k10	1314	1332.16	125	1.38
A-n64-k9	1401	1481.33	86	5.73
A-n65-k9	1174	1191.27	176	1.47
A-n69-k9	1159	1205.58	354	4.02
A-n80-k10	1763	1793.72	212	1.74

Πίνακας: Σετ δεδομένων B

Instance	BKS	Best	Cpu Time (s)	% Dev
B-n31-k5	672	686.99	85	2.23
B-n34-k5	788	796.72	96	1.11
B-n35-k5	955	956.3	94	0.14
B-n38-k6	805	811.15	137	0.76
B-n39-k5	549	563.55	201	2.65
B-n41-k6	829	833.82	135	0.58
B-n50-k7	741	745.17	91	0.56
B-n50-k8	1312	1326.84	112	1.13
B-n51-k7	1032	1040.27	202	0.8

Πίνακας: Σετ δεδομένων E

Instance	-- BKS --	-- Best --	Cpu Time (s)	% Dev
E-n22-k4	375	384.06	19	2.41
E-n23-k3	569	569.75	76	0.13
E-n33-k4	835	837.67	88	0.32
E-n51-k5	521	537.37	129	3.14
E-n76-k7	682	707.23	561	3.7
E-n76-k8	735	751.07	531	2.19
E-n76-k10	830	864.75	137	4.19
E-n76-k14	1021	1058.23	99	3.6
E-n101-k8	815	846.55	625	3.87

Ο μέσος όρος απόκλισης όλων των λύσεων υπολογίστηκε ότι είναι:

- 2.14 % για το σετ δεδομένων A
- 2.25 % για το σετ δεδομένων B
- 2.99 % για το σετ δεδομένων E

Ορισμένες προτάσεις, οι οποίες θα μπορούσαν να οδηγήσουν σε περαιτέρω βελτίωση των αποτελεσμάτων του αλγορίθμου, θα ήταν:

- η χρήση επιπλέον τελεστών τοπικής αναζήτησης (local search πχ 3 – opt).
- ή επιπλέον τελεστών στο shaking phase της VNS.

Ευχαριστώ για την προσοχή σας !!!