Τ.Ε.Ι ΠΑΤΡΑΣ
ΣΧΟΛΗ: ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
ΤΜΗΜΑ: ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΥ
ΣΧΕΔΙΑΣΜΟΥ
ΚΑΙ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Η ΕΠΙΛΥΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ ΤΗΣ ΚΑΛΥΨΗΣ ΧΩΡΟΥ (SET COVERING) ΜΕ ΜΕΘΟΔΟΥΣ ΑΚΕΡΑΙΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

ΣΤΕΦΑΝΑΤΟΣ ΦΩΤΗΣ
ΦΙΛΙΠΠΑΣ ΙΩΑΝΝΗΣ

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ:
ΜΑΣΤΡΟΓΙΑΝΝΗΣ ΝΙΚΟΛΑΟΣ

ΠΑΤΡΑ, ΙΑΝΟΥΑΡΙΟΣ 2010

Περιεχόμενα

Πρόλογος

Η παρούσα πτυχιακή εργασία μελετά με το γνωστό πρόβλημα κάλυψης συνόλου.

Το πρόβλημα κάλυψης συνόλου έχει αρκετές εφαρμογές σε πολλούς τομείς όπως στην οικονομία, στην πληροφορική, στην ανάπτυξη δικτύων, στην βιολογία κ.α.

Στο πρόβλημα της κάλυψης συνόλου το αντικείμενο είναι να ελαχιστοποιήσουμε το κόστος της τοποθέτησης του κέντρου έτσι ώστε ένα συγκεκριμένο επίπεδο κάλυψης να επιτευχθεί. Το κέντρο μπορεί να είναι ένα σημείο παροχής υπηρεσιών, ένα κέντρο τοποθέτησης ενός κεντρικού υπολογιστή δικτύου, ένα κέντρο διατροφής βακτηριδίων, ένα οικονομικό κέντρο εξυπηρέτησης μεταφοράς.

Το πρόβλημα αυτό καθορίζει τον ελάχιστο αριθμό και τις θέσεις των κέντρων παροχής υπηρεσιών (σε όποιο τομέα ενδιαφέροντος θέλουμε) που χρειάζονται για την ικανοποίηση (κάλυψη) όλων των απαιτήσεων χρησιμοποιώντας μία ορισμένη μέγιστη αποδεκτή απόσταση υπηρεσίας ή έναν μέγιστο αποδεκτό χρόνο S (coverage distance/time) δηλ. για κάθε σημείο ζήτησης να υπάρχει ένα κέντρο παροχής υπηρεσιών εντός απόστασης ή χρόνου.

Στην πτυχιακή μας δίνεται στο πρώτο κεφάλαιο ο ορισμός αλλά και τα σχετικά προβλήματα που εφαρμόζεται το πρόβλημα κάλυψης συνόλου. Στο δεύτερο κεφάλαιο αναλύονται διάφορες μέθοδοι επίλυσης του προβλήματος κάλυψης συνόλου, στο τρίτο κεφάλαιο αναλύεται ο βασικός αλγόριθμος k-modes που επιλύει το πρόβλημα κάλυψης συνόλου ενώ στο τέταρτο κεφάλαιο αναλύονται και επιλύονται με το

EXCEL μέσω γραμμικού προγραμματισμού διάφορα προβλήματα set covering.

Κεφάλαιο 1-Περιγραφή και μορφοποίηση του προβλήματος κάλυψης συνόλου

1.1 Εισαγωγή

Η βασική αρχή του οικονομικού σχεδιασμού είναι η δημιουργία οικονομικών και κοινωνικών στόχων για το μέλλον, εκφρασμένων σε ποσοτικοποιημένα μεγέθη και η εύρεση του πιο αποτελεσματικού τρόπου, ώστε με βάση τα υπάρχοντα διαθέσιμα οι στόχοι αυτοί να μπορούν να πραγματοποιηθούν. Ένα μεγάλο μέρος από τα μεγέθη αυτά έχουν σχέση με την χωρική ή χρονική σχέση και θέση μεταξύ τους. (Church R., ReVelle C. 1974)

Η θέση στον γεωγραφικό χώρο ή στον χρόνο παίζει σημαντικό και σπουδαίο ρόλο. Το σημείο στο οποίο μια οποιαδήποτε μονάδα χωροθετείται έχει επιπτώσεις:

- Στο κόστος: διαφορετικές θέσεις για την κατασκευή μιας μονάδας αντιπροσωπεύει διαφορετικά κατασκευαστικά και λειτουργικά έξοδα.

- Στην αποδοτικότητα: η θέση επιδρά στο πόσο αποδοτικά, επιτυγχάνονται οι σχεδιαστικοί στόχοι μιας μονάδας.

- Στην χρήση: η θέση μιας μονάδας επηρεάζει το βαθμό χρησιμοποίησής της από τους ανθρώπους τους οποίους η μονάδα αυτή εξυπηρετεί.

- Άλλα κέντρα: η θέση ενός κέντρου επηρεάζει, θετικά ή αρνητικά, το κόστος, την αποδοτικότητα και τη χρησιμοποίηση άλλων κέντρων.

Επειδή για κάθε κέντρο παροχής μιας ορισμένης υπηρεσίας υπάρχει μια δοσμένη και καθορισμένη ακτίνα δράσης και ένα ανώτατο όριο χωρητικότητας. Γι' αυτό η κατανεμημένη στο χώρο ζήτηση για αυτή την υπηρεσία δεν μπορεί να καλυφθεί από ένα και μόνο κέντρο, αλλά από περισσότερα, δηλαδή από ένα σύστημα τέτοιων κέντρων. Αποτέλεσμα αυτού είναι ο ταυτόχρονος καθορισμός τόσο του συνδυασμού των θέσεων που πρέπει να χωροθετηθούν τα κέντρα όσο και του συσχετιζόμενου συνδυασμού των περιοχών που πρέπει να αποτελέσουν τις περιοχές δράσης των κέντρων (Church R., ReVelle C. 1976)

Τα προβλήματα χωροθέτησης διακρίνονται στα παρακάτω

I) Προβλήματα ιδιωτικού τομέα.
- Προβλήματα δημόσιου τομέα.

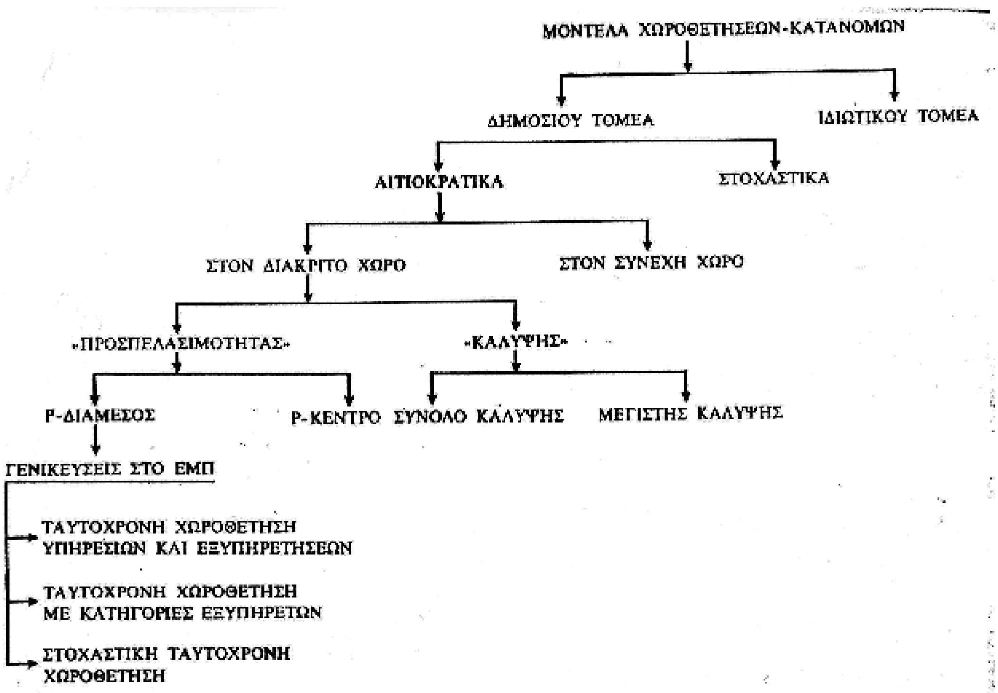
II) • Προβλήματα στον συνεχή χώρο.
- Προβλήματα στον διακριτό χώρο.

III) • Προβλήματα στον αιτιοκρατικό χώρο.
- Προβλήματα στον στοχαστικό χώρο.

IV) Προβλήματα συνόλων κάλυψης

V) Προβλήματα ελαχιστοποίησης της μέγιστης απόστασης (minimax) [p-κέντρα]

VI) Προβλήματα ελαχιστοποίησης της αθροιστικής απόστασης (minisum) [p-διάμεσοι



Όπως βλέπουμε το πρόβλημα κάλυψης συνόλου είναι μία κατηγορία των γενικότερων προβλημάτων χωροθέτησης κατανομών. (V. Vazirani 1st edition). Στόχος είναι να βρεθεί ο ελάχιστος αριθμός των κέντρων παροχής υπηρεσιών και οι θέσεις τους στο χώρο, έτσι ώστε για κάθε σημείο ζήτησης να υπάρχει ένα κέντρο μέσα σε μια απόσταση t_{ij} μονάδων απόστασης ή χρόνου (σε απόσταση ίση ή μικρότερη με t_{ij} μονάδων απόστασης ή χρόνου).

1.2 Γενική Περιγραφή του προβλήματος κάλυψης συνόλου

Σε πολλά προβλήματα χωροθέτησης η παροχή υπηρεσιών στους πελάτες (από τα υπάρχοντα κέντρα εξυπηρέτησης) εξαρτάται από την απόσταση του πελάτη από τοκέντρο παροχής υπηρεσίας. (ReVelle C.S., Eiselt H.A.

2005.) Οι πελάτες γενικά, αν και όχι πάντα, απευθύνονται στο πλησιέστερο κέντρο εξυπηρέτησης. Συχνά, η υπηρεσία θεωρείται ικανοποιητική αν ο πελάτης βρίσκεται μέσα σε μια δεδομένη απόσταση από το κέντρο παροχής της και ανεπαρκής όταν η απόσταση υπερβαίνει μια κρίσιμη τιμή.Αυτό οδηγεί στην έννοια της κάλυψης (coverage).

Με κάθε σημείο ζήτησης i σχετίζεται ένα σύνολο Ni από υποψήφια (εναλλακτικά) κέντρα παροχής υπηρεσίας j τα οποία μπορούν να εξυπηρετήσουν ή να καλύψουν το σημείο ζήτησης.

Το κλειδί για την τοποθέτηση κάποιων κέντρων παροχής υπηρεσιών είναι η «κάλυψη»(«coverage»). Για να χαρακτηρίζεται ένα κέντρο παροχής υπηρεσιών ως επικαλυπτόμενο (covered) πρέπει να χαρακτηρίζεται απαραίτητα και από ένα συγκεκριμένο χρόνο.

Τα προβλήματα κάλυψης χωρίζονται σε δύο κύριες ομάδες. Τα προβλήματα για τα οποία:

(α) η κάλυψη απαιτείται

(β) η κάλυψη βελτιστοποιείται

Δύο τέτοια προβλήματα στα οποία φαίνεται αυτή η διαφορά είναι:

(α) το πρόβλημα Συνόλου Κάλυψης (Locating Set Covering Problem)

(β) το πρόβλημα Μέγιστης Κάλυψης (Maximal Covering Problem).

1.1.1 Μοντέλο Συνόλου Κάλυψης (Location Set Covering Model)

Στο πρόβλημα του Συνόλου Κάλυψης το αντικείμενο είναι να ελαχιστοποιήσουμε το κόστος της τοποθέτησης του κέντρου παροχής υπηρεσιών έτσι ώστε ένα συγκεκριμένο επίπεδο κάλυψης να επιτευχθεί.

Το πρόβλημα αυτό καθορίζει τον ελάχιστο αριθμό και τις θέσεις των κέντρων παροχής υπηρεσιών που χρειάζονται για την ικανοποίηση (κάλυψη) όλων των απαιτήσεων χρησιμοποιώντας μία ορισμένη μέγιστη αποδεκτή απόσταση υπηρεσίας ή έναν μέγιστο αποδεκτό χρόνο S (coverage distance/time) δηλ. για κάθε σημείο ζήτησης να υπάρχει ένα κέντρο παροχής υπηρεσιών εντός απόστασης ή χρόνου tij (σε απόστασημικρότερη ή ίση με tij).

Για τη μαθηματική τυποποίηση αυτού του προβλήματος (Schilling D.A.et al 1993) ορίζουμε τα εξής :

Είσοδοι:

α_{ij} : παίρνει την τιμή 1 αν η υποψήφια θέση j μπορεί να καλύψει την ζήτηση i

f_j : το σταθερό κόστος τοποθέτησης ενός κέντρου παροχής υπηρεσιών στον κόμβο j.

Οι μεταβλητές απόφασης (decision variables) είναι οι εξής:

$$X_j = \begin{cases} 1, & \text{αν τοποθετήσουμε το κέντρο παροχής υπηρεσιών στην πιθανή θέση } j. \\ 0, & \text{διαφορετικά} \end{cases}$$

(1.1)

Έτσι, βάσει των πιο πάνω συμβολισμών, το πρόβλημα του Συνόλου Κάλυψης

τυποποιείται ως εξής:

MINIMIZE

Αντικειμενική συνάρτηση $\sum_j f_j X_j$ (1.2)

υπό τους περιορισμούς: $\sum_j a_{i,j} X_j \geq 1 \quad \forall i$ (1.3)

και **Xj** = 1, 0 $\forall j$ (1.4)

Η αντικειμενική συνάρτηση (1.2) περιγράφει την ελαχιστοποίηση του συνολικού κόστους της τοποθέτησης των κέντρων παροχής υπηρεσιών που έχουν επιλεγεί. Ο περιορισμός (1.3) απαιτεί ότι κάθε ζήτηση i να καλύπτεται από ένα τουλάχιστον κέντρο παροχής υπηρεσιών. Η πιο πάνω τυποποίηση δεν κάνει διαχωρισμό ανάμεσα στους βασικούς κόμβους και το μέγεθος της ζήτησης. Κάθε κόμβος είτε περιλαμβάνει έναν απλό πελάτη είτε ένα μεγάλο τμήμα της ολικής ζήτησης πρέπει να επικαλυφθεί με μία ειδική απόσταση ανεξάρτητα από το κόστος. Αν η επικαλυπτόμενη απόσταση S είναι μικρή, όσον αφορά την διάταξη των κόμβων ζήτησης (demand nodes), η περιγραφή της κάλυψης μπορεί να αναφέρεται σε ένα μεγάλο αριθμό κέντρων παροχής υπηρεσιών οι οποίες θα τοποθετηθούν. Επιπρόσθετα, αν ο πιο απομακρυσμένος κόμβος (outlying node) έχει μικρή ζήτηση, ο λόγος

$$\frac{\text{κόστος}}{\text{όφελος της επικάλυψης που απαιτείται}} \quad (1.5)$$

μπορεί να μεγαλώσει πολύ

1.1.2 Το μοντέλο μέγιστης κάλυψης (Maximum covering location model)

Το μοντέλο συνόλου κάλυψης είναι πραγματικά οριακός σταθμός στην ανάπτυξη των χωροθετικών μοντέλων. (Church R., ReVelle C.1974)

Ένα από τα βασικά προβλήματα που σχετίζονται με το μοντέλο συνόλου κάλυψης είναι ότι ο αριθμός των κέντρων παροχής υπηρεσιών που χρειάζονται για να καλύψουν όλα τα σημεία ζήτησης είναι πιθανό να υπερβαίνει τον αριθμό των κέντρων που πρακτικά μπορούν να τοποθετηθούν (για οικονομικούς και άλλους λόγους).

Επιπλέον, το μοντέλο του συνόλου κάλυψης αντιμετωπίζει όλα τα σημεία ζήτησης με πανομοιότυπο τρόπο.

Τα δύο αυτά θέματα οδήγησαν στη θεώρηση ενός δεδομένου (συγκεκριμένου) αριθμού κέντρων εξυπηρέτησης που πρέπει να τοποθετηθούν, τα οποία μεγιστοποιούν το ποσό της καλυπτόμενης ζήτησης (σε αντίθεση με τον αριθμό των καλυπτόμενων σημείων ζήτησης). Πρόκειται για το μοντέλο Μέγιστης Κάλυψης (maximum covering location model) που πρότειναν οι Church και Revelle (1974).

Το πρόβλημα Μέγιστης Κάλυψης μεγιστοποιεί το ποσό της επικαλυπτόμενης ζήτησης μέσα στην αποδεκτή απόσταση υπηρεσίας (acceptable service distance) τοποθετώντας ένα σταθερό αριθμό από κέντρα παροχής υπηρεσιών. Ουσιαστικά, το πρόβλημα αυτό μεγιστοποιεί τον αριθμό των χρηστών που μπορούν να ικανοποιηθούν από την χωροθέτηση P-κέντρων παροχής υπηρεσιών.

Το πρόβλημα αυτό μπορεί να εκφραστεί ως εξής:

Να χωροθετηθούν P-κέντρα παροχής υπηρεσιών σε σημεία ενός δικτύου, έτσι ώστε το μέγιστο μέρος (όχι το σύνολο) του πληθυσμού / χρηστών να βρίσκεται μέσα σ’ ένα ορισμένο χρόνο ή απόσταση.

Το πρόβλημα αυτό τυποποιείται μαθηματικά ως εξής:

Ορίζουμε:

h_i : η ζήτηση / απαίτηση του πελάτη στον κόμβο i

P : ο αριθμός των κέντρων παροχής υπηρεσιών που θα τοποθετηθούν

Οι μεταβλητές απόφασης (decision variables) είναι οι εξής:

$$Z_I = \begin{cases} 1, & \text{αν ο κόμβος i είναι επικαλυμμένος.} \\ 0, & \text{διαφορετικά.} \end{cases} \quad (1.6)$$

Βάσει των πιο πάνω συμβολισμών, το πρόβλημα Μέγιστης Κάλυψης τυποποιείται ως εξής:

Στόχος

MAXIMIZE $$\sum_j f_j X_j \quad (1.7\ )$$

Ύπο περιορισμούς:

$$\sum_j a_{i,j} X_j \geq Z_i \quad \forall i \quad (1.8)$$

$$\sum_j X_j \leq P \quad (1.9)$$

$$\mathbf{X_j} = 1, 0 \quad \forall j \quad (1.10)$$

$$\mathbf{Z_i} = 1, 0 \quad \forall i \quad (1.11)$$

Η αντικειμενική συνάρτηση (1.7) περιγράφει την μεγιστοποίηση του ποσού απαίτησης (ζήτησης) που καλύπτεται. Ο περιορισμός (1.8) καθορίζει ποιοι κόμβοι ζήτησης (demand nodes) καλύπτονται μέσα στην αποδεκτή απόσταση υπηρεσιών. Κάθε κόμβος i μπορεί να θεωρηθεί επικαλυμμένος (covered) (με Zi = 1) εάν υπάρχει κάποιο κέντρο παροχής υπηρεσιών το οποίο να βρίσκεται επί κάποιου τόπου j που είναι μέσα στο S του κόμβου i. Εάν κανένα τέτοιο κέντρο παροχής υπηρεσιών δεν τοποθετείται επί κάποιου τόπου j S , το δεξί μέλος του περιορισμού (1.9) θα είναι μηδέν, αναγκάζοντας κατά συνέπεια το Zi να μηδενιστεί. Ο περιορισμός (1.9) καθορίζει ότι δεν μπορούμε να τοποθετήσουμε περισσότερα από P κέντρα παροχής υπηρεσιών. Οι περιορισμοί (1.10) και (1.11) είναι οι περιορισμοί για τις μεταβλητές απόφασης.

1.2 Χαρακτηριστικές εφαρμογές για το πρόβλημα κάλυψης συνόλου

Η ανάπτυξη και η απόκτηση ενός νέου κέντρου παροχής υπηρεσιών είναι μια δαπανηρή έρευνα και ευαίσθητη στο χρόνο διαδικασία. Προτού αγοραστεί ή κατασκευαστεί ένα νέο κέντρο παροχής υπηρεσιών πρέπει να οριστούν κάποιες καλές θέσεις, πρέπει να οριστούν οι κατάλληλες προδιαγραφές των δυνατοτήτων των κέντρων παροχής υπηρεσιών και επιπλέον πρέπει να κατανεμηθούν μεγάλα ποσά κεφαλαίου. (Cooper L.1963)

Τα υψηλά κόστη που συνδέονται με τη διαδικασία της Χωροθέτησης Κέντρων Παροχής Υπηρεσιών (Facility Location) μετατρέπουν την οποιαδήποτε έρευνα Χωροθέτησης (location project) σε μια διαδικασία μακράς διάρκειας. Έτσι, όλα τα κέντρα παροχής υπηρεσιών που θα τοποθετήσουμε αναμένεται να παραμείνουν σε εφαρμογή για ένα εκτεταμένο χρόνο.

Κατά τη διάρκεια της επικράτησης ενός κέντρου παροχής υπηρεσιών πιθανών να εμφανιστούν διάφορες περιβαλλοντικές αλλαγές οι οποίες θα αναβαθμίσουν την έκκληση προς μια ειδική πλευρά στρέφοντας την σημερινή κατάλληλη θέση σε αυριανό επενδυτικό λάθος. Ο καθορισμός των καλύτερων θέσεων για κάθε νέο κέντρο παροχής υπηρεσιών αποτελεί μια σημαντική στρατηγική πρόκληση . (Daskin M.S., Owen S.H.1999)

Πολλά από τα προβλήματα χωροθέτησης λύνονται με το πρόβλημα κάλυψης συνόλου. Τα προβλήματα της Χωροθέτησης Κέντρων Παροχής Υπηρεσιών περιλαμβάνουν την τοποθέτηση ενός ή περισσότερων κέντρων παροχής υπηρεσιών σε μια περιοχή όπου υπάρχει ζήτηση ενώ βελτιστοποιούμε την κύρια αντικειμενική συνάρτηση. Μια συχνή απλοποίηση αυτού του μετασχηματισμού είναι η αντικατάσταση της συνεχούς ζήτησης στην περιοχή με ένα διακεκριμένο σύνολο από σημεία ζήτησης (demand points) τα οποία το καθένα αναπαριστάνει μιαν υποπεριοχή.

Η κύρια αιτία που γίνεται η αντικατάσταση αυτή είναι το ότι είναι ευκολότερο να κατασκευάσεις την αντικειμενική συνάρτηση βασιζόμενος σε ένα πεπερασμένο σύνολο από σημεία ζήτησης (demand points) παρά σε μια συνεχή συναρτησιακή ζήτηση (continuous functional demand). Συχνά, η αντικειμενική συνάρτηση συνίσταται σαν ένα σύνολο από όρους, ένα για κάθε σημείο ζήτησης. Στη συνεχή τυποποίηση, αυτό το άθροισμα αντικαθίσταται από ένα διπλό ολοκλήρωμα στην περιοχή.

Οι εφαρμογές στα μοντέλα της Κάλυψης συνόλου και σωστής Χωροθέτησης Κέντρων Παροχής Υπηρεσιών ποικίλουν. Μερικές από αυτές είναι η τοποθέτηση αποθηκών, εργοστασίων, νοσοκομείων, η λιανική αγορά αγαθών (retail outlet) και διάφορα άλλα κλασικά παραδείγματα. Συναντάμε επίσης εφαρμογές της Χωροθέτησης Κέντρων Παροχής Υπηρεσιών στην τοποθέτηση ηλεκτρονικών συνιστωσών (electronic components), σειρήνων συναγερμού, συστημάτων πυρόσβεσης, κεραιών ραντάρ, εξερευνητικών πετρελαιοπηγών κλπ. Αυτά ονομάζονται “facilities” (κέντρα παροχής υπηρεσιών, παροχές, υπηρεσίες). Σκοπός της είναι να βρεθεί η καλύτερη θέση (ή θέσεις) για κάθε κέντρο παροχής υπηρεσιών.

Επίσης, πολλές εφαρμογές της Ανάλυσης Χωροθέτησης αναφέρονται στην τοποθέτηση ενός ή περισσότερων κέντρων παροχής υπηρεσιών με τέτοιο τρόπο ώστε να βελτιστοποιούνται κάποια σταθερά αντικείμενα όπως η ελαχιστοποίηση του κόστους μεταφοράς, η προώθηση ισοδύναμων υπηρεσιών στους πελάτες επιτυγχάνοντας το κέρδος των περισσοτέρων μετοχών της αγοράς. Τα προβλήματα της Χωροθέτησης Κέντρων Παροχής Υπηρεσιών αποτελούν την αφορμή για τη λύση διαφόρων γεωμετρικών και συνδυαστικών προβλημάτων. Η έρευνα των προβλημάτων της Χωροθέτησης Κέντρων Παροχής Υπηρεσιών συνδέει πολλά ερευνητικά πεδία όπως τις εφαρμογές σε Ερευνητικές και Διοικητικές Επιστήμες, τη Μηχανολογία Μηχανικών σε βιομηχανίες, τη Γεωγραφία, τα Οικονομικά, την Επιστήμη των Υπολογιστών, τα Μαθηματικά, το marketing, την Ηλεκτρολογία Μηχανικών, τον Μη Γραμμικό Προγραμματισμό και άλλα σχετικά πεδία.

Επιπλέον, η λύση στο αντίστοιχο πρόβλημα κάλυψης συνόλου είναι ένα σημαντικό στοιχείο για το σχεδιασμό στρατηγικών για ένα γενικό φάσμα των δημόσιων και ιδιωτικών φιρμών. Όταν μια επιχείρηση θέλει να βγάλει στην αγορά κάποια νέα προϊόντα, ο κατασκευαστής θα πρέπει να διαλέξει τον τόπο που θα τοποθετηθεί μια αποθήκη ή ένα city planner το οποίο περιλαμβάνει π.χ τους πυροσβεστικούς σταθμούς. Οι σχεδιαστές στρατηγικών καλούνται συχνά να πάρουν τις ανάλογες χωρικές αποφάσεις κατανομής. Επειδή οι τάσεις της αγοράς εξελίσσονται και πολλοί περιβαλλοντικοί παράγοντες αλλάζουν, η ανάγκη για επανατοποθετήσεις (relocations) εξαπλώνεται και οι προσαρμοσμένες υπηρεσίες εξασφαλίζουν την εξέλιξη των νέων απαιτήσεων του σχεδιασμού .

Οι χωροθετικές αποφάσεις (location decisions) εμφανίζονται και αυτές σε μια ποικιλία προβλημάτων του ιδιωτικού και του δημόσιου τομέα. Στον δημόσιο τομέα ενδεικτικά μπορεί να αναφερθεί η χωροθέτηση πυροσβεστικών σταθμών, αστυνομικών τμημάτων και ασθενοφόρων. Και στις τρεις περιπτώσεις η χωροθέτηση είναι δυνατόν να μεταφραστεί σε αύξηση των πιθανοτήτων καταστροφής περιουσίας ή / και απώλειας ζωής.

Στον ιδιωτικό τομέα, οι βιοτεχνίες και οι βιομηχανίες θα πρέπει να χωροθετήσουν γραφεία, χώρους παραγωγής, κέντρα διανομής και καταστήματα πώλησης. Σε αυτήν την περίπτωση εσφαλμένες χωροθετικές αποφάσεις θα οδηγήσουν σε αύξηση του επενδυτικού κόστους και μείωση της ανταγωνιστικότητας της επιχείρησης.

Επομένως, η επιτυχία ή η αποτυχία λειτουργιών του δημόσιου και του ιδιωτικού τομέα εξαρτάται σε σημαντικό βαθμό από τις θέσεις που θα επιλεγούν για τις συγκεκριμένες λειτουργίες.

Υπάρχουν διάφορες μέθοδοι και τεχνικές που χρησιμοποιούνται για την ανάλυση και την ανεύρεση των βέλτιστων προτύπων και την επίλυση των αντίστοιχων προβλημάτων που λύνονται με ανάπτυξη αλγορίθμων για την αντίστοιχη επίλυση αυτών των προβλημάτων

Πολλά προβλήματα της οικονομίας, των κατασκευών, της πληροφορικής κ.α. επιστημών είναι προβλήματα που ανήκουν στο πρόβλημα κάλυψης συνόλου. Μερικά από αυτά είναι:

1.2.1 Χωροθέτηση Τραπεζικών Καταστημάτων

Τα τελευταία χρόνια έχουν λάβει χώρα σημαντικές αλλαγές στον τραπεζικό τομέα παγκοσμίως. Η χρήση εξειδικευμένης τεχνολογίας από τις τράπεζες έχει δημιουργήσει μία πολύ πιο ευέλικτη δομή όπου απομακρυσμένα τραπεζικά καταστήματα μπορούν να λειτουργούν και να ελέγχονται μέσω υπολογιστών και απασχολώντας λιγότερους υπαλλήλους. (Beraldi 2002)

Το χαμηλότερο κόστος δημιουργίας και υποστήριξης της λειτουργίας τραπεζικών καταστημάτων και οι νέοι τρόποι πραγματοποίησης συναλλαγών (ATMs, phone banking κ.λ.π.) έχουν ενθαρρύνει τη δημιουργία και ανάπτυξη νέων τραπεζών και τραπεζικών καταστημάτων.Οι παράγοντες αυτοί έχουν κάνει το σχεδιασμό και την αξιολόγηση των τραπεζών ζήτημα μεγάλης σημασίας αλλά ταυτόχρονα και πολυπλοκότερο από ποτέ.

Το άνοιγμα ενός νέου τραπεζικού υποκαταστήματος αποτελεί ένα σημαντικό μέρος τουπρογραμματισμού και της διαχείρισης δικτύων κλάδων. Πρόκειται για μια σύνθετη διαδικασία λήψης απόφασης ανάμεσα από διάφορες πιθανές εναλλακτικές λύσεις. Περιλαμβάνει τα συγκρουόμενα ποσοτικά και ποιοτικά κριτήρια και τους διάφορους υπεύθυνους για τη λήψη αποφάσεων. Η διαδικασία λήψης απόφασης μπορεί να ωφεληθεί από τη χρήση των τεχνικών λήψης απόφασης πολλαπλών κριτηρίων (MCDM) που μπορεί να χρησιμοποιηθεί για να διευκολύνει τη διαδικασία λήψης απόφασης με να καταστήσει τη διαδικασία λογική, και αποδοτική .

Ο αυξανόμενος ανταγωνισμός έχει αναγκάσει επίσης το μάρκετινγκ των τραπεζικών υπηρεσιών να οδηγείται κυρίως από τις ανάγκες των πελατών: οι τράπεζες δίνουν τώρα περισσότερη προσοχή στις ανάγκες των πελατών και των απαιτήσεών τους στην παροχή των τραπεζικών υπηρεσιών. Η χωρική διανομή των πελατών και της γεωγραφίας των αγορών επομένως έχει ενισχύσει την άποψη ότι μεγαλύτερη έμφαση πρέπει να δοθεί στις ως προς την τοποθεσία και χωρικές εκτιμήσεις στη λήψη αποφάσεων για να εξασφαλίσει ότι η αιτιολόγηση οδηγεί σε ένα δίκτυο κλάδων ενός κατάλληλων μεγέθους και ενός τύπου με τις εξόδους στη σωστή θέση. Για να επιτύχει αυτό το τέλος υποστηρίζεται ότι οι υπεύθυνοι για τη λήψη αποφάσεων θα μπορούσαν να ωφεληθούν από την υιοθέτηση επίσημων μεθόδων και τεχνικών της πολυκριτηριακής ανάλυσης στη διαδικασίας λήψης αποφάσεων.

Η όλη διαδικασία βασίζεται στην εξέταση τριών βασικών θεμάτων:

- Σε ποια περιοχή θα εγκατασταθεί το νέο κατάστημα

- Πόσα καταστήματα πρέπει να εγκατασταθούν στην κάθε περιοχή
- Ποιες υπηρεσίες πρέπει το κάθε κατάστημα να παρέχει στους πελάτες του;

Παράγοντες που λαμβάνονται υπόψη κατά την ανάλυση του προβλήματος είναι οι ακόλουθοι:

v Οικονομικοί παράγοντες

Ø το ετήσιο οικογενειακό εισόδημα

Ø το ύψος των καταθέσεων

v Δημογραφικοί παράγοντες

Ø πληθυσμός: ο συνολικός πληθυσμός, οι αλλαγές στα χαρακτηριστικά πληθυσμών, και τα ποσοστά αύξησης πληθυσμών μπορούν να χρησιμοποιηθούν ως μέτρα της πιθανής απαίτησης για τις τραπεζικές υπηρεσίες. Εντούτοις, οι συνολικοί δείκτες πληθυσμών που λαμβάνονται μεμονωμένα μπορούν μόνο να θεωρηθούν ως ακατέργαστα μέτρα της πιθανής απαίτησης.

Ø ηλικία πληθυσμού: Η σημασία της ηλικίας ως παράγοντα που έχει επιπτώσεις στην αποδοτικότητα κλάδων επιβεβαιώνεται από το γεγονός ότι τα πρόσωπα πέρα από την ηλικία εκθέτουν αρκετά διαφορετική συμπεριφορά στη χρησιμοποίηση των οικονομικών υπηρεσιών των τραπεζών από τις νεώτερες ομάδες ηλικίας

Ø επάγγελμα

v Ο απασχολούμενος πληθυσμός και ο αριθμός μικρών επιχειρήσεων στις περιοχές ενδιαφέροντος v Η παρουσία ανταγωνιστών: Οι μελέτες έχουν δείξει ότι η θέση των κλάδων των τραπεζών ανταγωνιστών στη στενή εγγύτητα σε εκείνους ενός άλλου οργάνου ασκεί σημαντική επίδραση στην απόδοση.

v Η φυσική θέση των κλάδων είναι επίσης σημαντική.

Ø Η δυνατότητα πρόσβασης

Ø Η ευκολία του ταξιδιού μέσω του δικτύου οδών

Ø Η διαθεσιμότητα των διαστημάτων χώρων στάθμευσης και των δημόσιων συγκοινωνιών

Ø Η στενότητα σε εμπορικά κέντρα και άλλους σημαντικούς εργοδότες όλα διαδραματίζουν έναν ρόλο στον καθορισμό της βιωσιμότητας τραπεζών

1.2.1 Υλοποίηση ενός ασύρματου μη δομημένου δικτύου πολλαπλών αλμάτων

Η σωστή υλοποίηση ενός ασύρματου μη δομημένου δικτύου πολλαπλών αλμάτων απαιτεί τον προσεκτικό σχεδιασμό στο επίπεδο δικτύου (network layer) του μοντέλου OSI (Open System Interconnection) της μεθόδου δρομολόγησης (routing), που είναι υπεύθυνη για την επιλογή των κατάλληλων διαδρομών (paths) που θα ακολουθήσουν τα πακέτα πληροφορίας για να φτάσουν στον τελικό προορισμό τους. Παρά το ότι για πολλά ζεύγη κόμβων η άμεση επικοινωνία δεν είναι δυνατή, μπορούν να βρεθούν διαδρομές οι οποίες θα φέρουν τελικά κάθε ζεύγος αποστολέα-παραλήπτη σε επικοινωνία. (Wendell R.E 1973)

Οι διαδρομές αυτές αποτελούν ακολουθία εφικτών ζεύξεων (links) και προϋποθέτουν ότι οι ενδιάμεσοι κόμβοι θα προωθήσουν το μήνυμα στον τελικό προορισμό του. Στα πλαίσια της οργάνωσης IETF (Internet Engineering Task Force) έχει συσταθεί εδώ και χρόνια η ομάδα εργασίας MANET (Mobile Ad-hoc NETworks), η οποία ασχολείται με θέματα δρομολόγησης αναπτύσσοντας συγκεκριμένα πρωτόκολλα και καθορίζοντας τις προδιαγραφές λειτουργίας τους. Σημαντική κατηγορία των μεθόδων δρομολόγησης στα μη δομημένα δίκτυα αποτελούν οι αλγόριθμοι που αναλαμβάνουν τη μετάδοση της πληροφορίας από έναν κόμβο-πηγή (source node) προς όλους τους υπόλοιπους κόμβους του δικτύου, γνωστή και ως ευρεία μετάδοση δεδομένων (broadcasting).

Όταν η πληροφορία δρομολογείται από έναν κόμβο προς ένα υποσύνολο των μελών του δικτύου, η μετάδοση δεδομένων χαρακτηρίζεται ως πολλαπλή (multicasting) για να διαφοροποιείται από την προηγούμενη περίπτωση. Οι δύο αυτοί τρόποι επικοινωνίας παρουσιάζουν ιδιαίτερο ενδιαφέρον και απαιτούν διαφορετική αντιμετώπιση από την απλή περίπτωση της δρομολόγησης ένα-προς-ένα (από έναν αποστολέα σε ένα μόνο συγκεκριμένο παραλήπτη). Η ευρεία και η πολλαπλή μετάδοση δεδομένων συναντώνται σε πολλές και σημαντικές εφαρμογές, καθώς συχνά δημιουργείται η ανάγκη να ενημερώσει κάποιος κόμβος τους υπόλοιπους για μια μεταβολή που έχει συμβεί στο δίκτυο, ενώ μπορούν να χρησιμοποιηθούν επίσης και για την ανταλλαγή μηνυμάτων ελέγχου και τη μετάδοση κάθε είδους περιεχομένου σε όλο το δίκτυο.

Από την άλλη μεριά, το είδος της επικοινωνίας που είναι περισσότερο χρήσιμο σε ένα δίκτυο αισθητήρων είναι το αντίστροφο της ευρείας μετάδοσης, δηλαδή η δρομολόγηση της πληροφορίας που αποκτά κάθε αισθητήρας του δικτύου προς έναν κόμβο-συλλέκτη (sink node). Ο τελευταίος έχει συνήθως αυξημένες δυνατότητες επεξεργασίας και αποθήκευσης των δεδομένων, ώστε να αξιολογήσει τις πληροφορίες που συλλέγει, να τις αποθηκεύσει, ή ακόμη και να τις προωθήσει σε ένα μεγαλύτερο κέντρο επεξεργασίας όπου λαμβάνονται περαιτέρω δράσεις.

Μία βασική παράμετρος η οποία θα πρέπει να λαμβάνεται υπ' όψη από την εκάστοτε μέθοδο δρομολόγησης είναι το γεγονός ότι οι κόμβοι του δικτύου παίρνουν συνήθως την ενέργειά τους από μπαταρίες, με αποτέλεσμα η κατανάλωση ενέργειας που απαιτείται για την πραγματοποίηση της επικοινωνίας ανάμεσα στους κόμβους να είναι ένας από τους σημαντικότερους περιοριστικούς παράγοντες. Ένα μεγάλο μέρος της ενέργειας που καταναλώνεται οφείλεται στην απαιτούμενη ισχύ μετάδοσης (transmission power) από τον κόμβο-πομπό του μηνύματος, η οποία εξαρτάται κυρίως από την απόστασή του από τον κόμβο-δέκτη και από άλλες παραμέτρους του δικτύου. Ενέργεια επίσης καταναλώνεται για διαδικασίες επεξεργασίας των διακινούμενων πληροφοριών τόσο κατά τη μετάδοσή τους (κωδικοποίηση, διαμόρφωση, κ.τ.λ.), όσο και κατά τη λήψη τους (αποδιαμόρφωση, αποκωδικοποίηση, κ.τ.λ.). Ένας σημαντικός διαχωρισμός ως προς τον τρόπο με τον οποίο πρέπει να εξετάζεται η κατανάλωση ενέργειας κατά τη διάρκεια λειτουργίας του δικτύου, είναι εάν αυτή θεωρείται ως ένα ακριβό (αλλά εύκολα ανανεώσιμο) αγαθό ή ως ένα αγαθό πεπερασμένο (και κατ' επέκταση μη ανανεώσιμο). Με άλλα λόγια, πρέπει να διαφοροποιείται η αντιμετώπιση της ενέργειας ως μία ακόμη συνάρτηση κόστους για το δίκτυο ή ως ένας αυστηρός περιορισμός που υπαγορεύει και τη διάρκεια ζωής (lifetime) του δικτύου.

Στην πρώτη περίπτωση, επιδιώκεται η ελαχιστοποίηση της καταναλισκόμενης ενέργειας για την επίτευξη ενός στόχου επικοινωνίας, όπως για παράδειγμα να διακινηθεί όσο το δυνατόν περισσότερη πληροφορία πριν αντικατασταθούν οι μπαταρίες των συσκευών.

Στη δεύτερη περίπτωση, το πρόβλημα που τίθεται συνήθως είναι η μεγιστοποίηση της διάρκειας ζωής του δικτύου. Αυτή μπορεί να οριστεί ως το χρονικό σημείο κατά το οποίο τα αποθέματα ενέργειας ενός οποιουδήποτε κόμβου μέσα στο δίκτυο εξαντλούνται για πρώτη φορά (ένας ορισμός που χρησιμοποιείται συχνά στη διεθνή βιβλιογραφία) ή όταν ένα συγκεκριμένο ποσοστό των κόμβων μείνει χωρίς ενέργεια και το δίκτυο είναι πλέον μη συνδεδεμένο (disconnected).

Το προβλήμα είναι η μετάδοση της πληροφορίας σε ασύρματα μη δομημένα δίκτυα πολλαπλών αλμάτων, λαμβάνοντας υπ' όψη την κατανάλωση ενέργειας από τους κόμβους του δικτύου. Το είδος της επικοινωνίας που κυρίως μελετάται στη γενική κατηγορία των μη δομημένων δικτύων είναι αυτό της ευρείας (αλλά και της πολλαπλής) μετάδοσης δεδομένων, ενώ στην ειδική κατηγορία των δικτύων αισθητήρων εξετάζεται η δρομολόγηση της πληροφορίας από κάθε αισθητήρα προς τον κόμβο-συλλέκτη.

Όπως είναι φανερό το πρόβλημα είναι ένα σύνθετο πρόβλημα κάλυψης συνόλου με πάρα πολλές παραμέτρους. Οι αλγόριθμοι για την λύση του είναι αρκετοί και πάνω σε αυτό γίνονται αρκετές έρευνες για την βελτιστοποίηση της λύσης του.

1.2.3 Πρόβλημα συμπίεσης συνόλου ακολουθιών

Μπορούμε να διακρίνουμε τις στατικές μεθόδους συμπίεσης σε μεθόδους που εφαρμόζονται σε μια μόνο ακολουθία και σε μεθόδους που εφαρμόζονται σε ένα σύνολο από ακολουθίες ελέγχου. (White J. 1974) A Στις μεθόδους που εφαρμόζονται σε μια ακολουθία ανήκει η μέθοδος της παράλειψης διανυσμάτων (Vector Omission method) που επιτυγχάνει πολύ καλή συμπίεση με επιβάρυνση υψηλούς χρόνους εκτέλεσης (πολλές προσομοιώσεις και μεγάλη λίστα σφαλμάτων). Η μέθοδος της επαναφοράς των διανυσμάτων εισόδου (Vector Restoration method), επιτυγχάνει πολύ υψηλά ποσοστά συμπίεσης και συνήθως εφαρμόζεται σε μια ακολουθία. Στη βιβλιογραφία έχουν προταθεί μέθοδοι συμπίεσης που βασίζονται στην απομάκρυνση των επαναλαμβανόμενων υποακολουθιών (recurrence subsequences), οι

οποίες όμως δίνουν μικρά ποσοστά συμπίεσης στις περιπτώσεις που οι ακολουθίες ελέγχου οδηγούν το κύκλωμα σε πολύ λίγες ή καθόλου επαναλαμβανόμενες καταστάσεις. Υπάρχουν πολλές στατικές μέθοδοι συμπίεσης που εφαρμόζονται σε σύνολο ακολουθιών ελέγχου. Σύνολα από ακολουθίες ελέγχου προκύπτουν συνήθως από ATPG μεθόδους με προσανατολισμό στην ανίχνευση συγκεκριμένων σφαλμάτων (fault oriented methods) Οι ATPG αλγόριθμοι παράγουν κάθε ακολουθία ελέγχου στοχεύοντας στην ανίχνευση ενός συγκεκριμένου σφάλματος και κατόπιν προσομοιώνουν την ακολουθία λαμβάνοντας υπόψη τα υπόλοιπα μη ανιχνεύσιμα σφάλματα για να εξακριβώσουν αν η παραχθείσα ακολουθία ανιχνεύει τυχόν επιπλέον σφάλματα. Το τελικό σύνολο των ακολουθιών που προκύπτει είτε συνενώνεται σε μια ακολουθία (οπότε και μπορούμε να εφαρμόσουμε μεθόδους συμπίεσης που ενεργούν σε μια ακολουθία) εφόσον οι ακολουθίες παρήχθησαν θεωρώντας ότι το κύκλωμα ευρίσκεται σε άγνωστη “X” αρχική κατάσταση ειδάλλως θα πρέπει να κρατηθούν χωριστά δημιουργώντας έτσι ένα σύνολο ακολουθιών ελέγχου. Ειδικά στην περίπτωση που ένα εξωτερικό σήμα reset χρησιμοποιείται για την αρχικοποίηση του υπό έλεγχο κυκλώματος οι ακολουθίες ελέγχου δεν είναι δυνατό να συνενωθούν σε μια. Για τη συμπίεση ακολουθιών εφαρμόζεται η ανάστροφη προσομοίωση σφαλμάτων (Reverse Order Fault Simulation). Επίσης χρησιμοποιούνται οι έννοιες των συμβατών διανυσμάτων (compatible vectors) και συμβατών ακολουθιών (compatible sequences) στις μεθόδους που αναπτύχθηκαν (Simple alignment, Skew, Stretching) όμως οι μέθοδοι αυτές έχουν το μειονέκτημα ότι δεν εγγυώνται ότι η τελική, συμπιεσμένη, ακολουθία θα διατηρεί το αρχικό ποσοστό κάλυψης σφαλμάτων. Η μέθοδος επιλογής διανυσμάτων (Vector Selection) σύμφωνα με την οποία από μια αρχική ακολουθία ελέγχου εξάγονται σύνολα υποακολουθιών ελέγχου για κάθε ανιχνευθέν σφάλμα στα οποία εφαρμόζεται μια μέθοδος κάλυψης του συνόλου των σφαλμάτων (set covering method) για την επιλογή ενός μικρού υποσυνόλου ακολουθιών που καλύπτουν όλα τα σφάλματα. Η μέθοδος της επιλογής διανυσμάτων είναι χρονοβόρα λόγω της εκτεταμένης χρήσης της προσομοίωσης σφαλμάτων (μετά από κάθε επιλογή νέας ακολουθίας) που πραγματοποιεί.

1.3 Προβλήματα σχετιζόμενα με το πρόβλημα κάλυψης συνόλου

1.3.1 Προβλήματα Διακέντρων (Center problems)

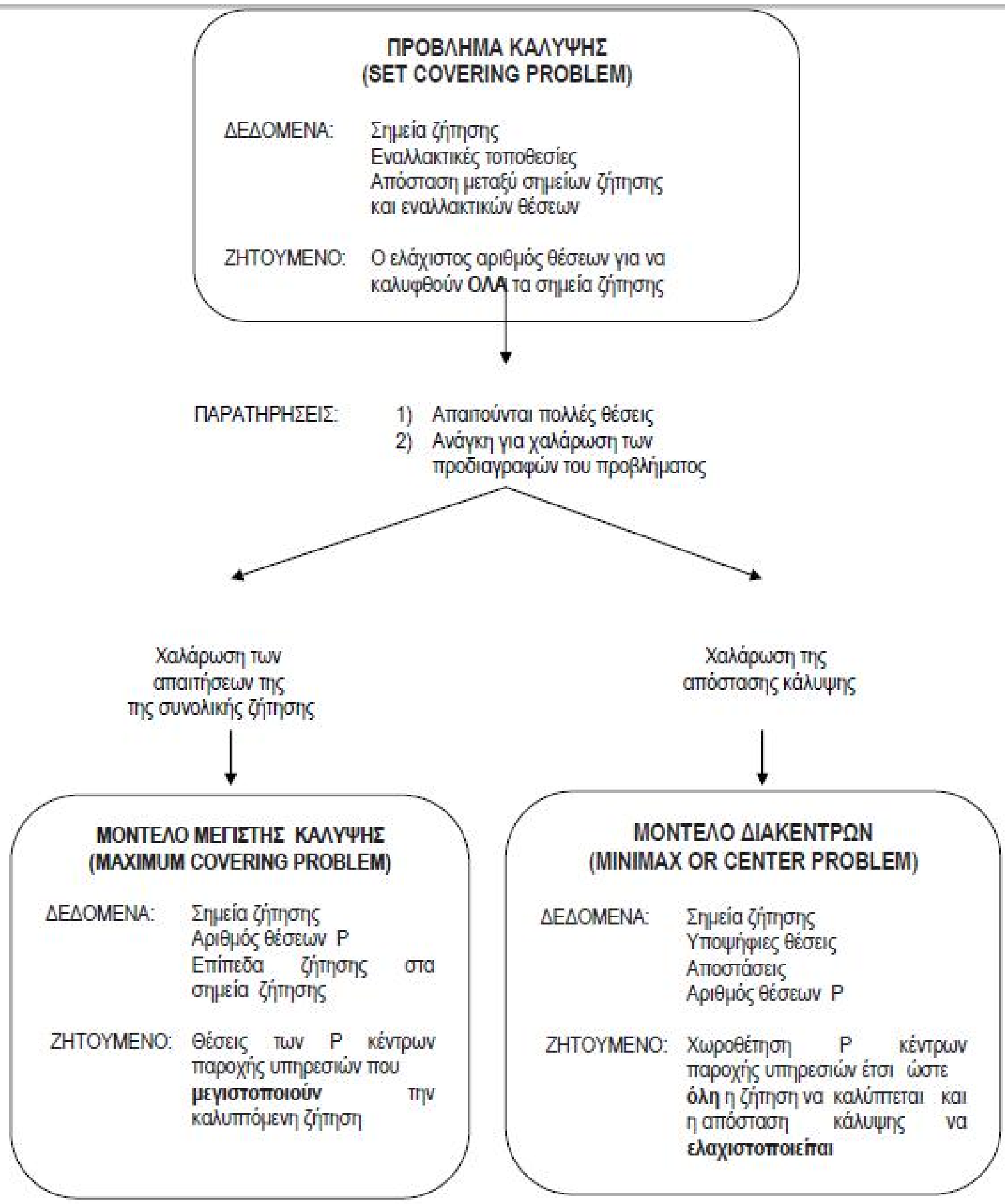
Μια άλλη κατηγορία προβλημάτων χωροθέτησης είναι η κλάση των προβλημάτων P-Διακέντρων (P-center).(Kariv O., Hakimi S.L 1979)

Σε τέτοια προβλήματα με βελτιστοποίηση της αντικειμενικής συνάρτησηςεπιδιώκουμε την τοποθέτηση ενός δεδομένου αριθμού κέντρων παροχής υπηρεσιών (ενός ή P) με τέτοιο τρόπο ώστε ελαχιστοποιείται η μέγιστη απόσταση (ή χρόνος) ενός οποιουδήποτε σημείου από κοντινότερο κέντρο παροχής υπηρεσιών αυτού. Αυτή η απόσταση (ή χρόνος) ονομάζεται απόσταση (ή χρόνος) κάλυψης.

Τέτοια προβλήματα είναι κατάλληλα για τον σχεδιασμό της χωροθέτησης κέντρωνπαροχής υπηρεσιών «επείγουσας βοήθειας», όπως για παράδειγμα Κέντρων Υγείας, Πυροσβεστικών σταθμών, Αστυνομικών τμημάτων κ.λ.π. Δηλαδή, αν παραστήσουμε με δίκτυο ένα οδικό δίκτυο, με τις κορυφές του να παριστάνουν κοινότητες, αντιμετωπίζουμε το πρόβλημα να χωροθετηθούν στην «άριστη» θέση νοσοκομείο, ή ένας πυροσβεστικός σταθμός ή ένα αστυνομικό τμήμα ή τελικά οποιαδήποτε υπηρεσία παροχής κατεπειγουσών υπηρεσιών.

Στην γενική περίπτωση του προβλήματος έχουμε να χωροθετήσουμε ένα μεγάλο αριθμό τέτοιων κέντρων παροχής υπηρεσιών. Σ’ αυτήν την περίπτωση, η πλέον απομακρυσμένη περιοχή του γραφήματος, πρέπει να είναι προσιτή το λιγότερο από ένα κέντρο παροχής υπηρεσιών μέσα σε μια ελάχιστη απόσταση. Αυτά τα προβλήματα που απαιτούν την χωροθέτηση κέντρων παροχής υπηρεσιών και ο αντικειμενικός σκοπός (αντικειμενική συνάρτηση) είναι η ελαχιστοποίηση της μεγαλύτερης απόστασης οποιασδήποτε κορυφής από το πλησιέστερο κέντρο παροχής υπηρεσιών, ονομάζονται minimax προβλήματα χωροθέτησης (Minimax location problems). Οι θέσεις των κέντρων παροχής υπηρεσιών ονομάζονται κέντρα (centers) του γραφήματος.

Στο σχήμα που ακολουθεί απεικονίζονται οι σχέσεις μεταξύ των μοντέλων κάλυψης, μέγιστης κάλυψης και διακέντρων.



Τα προβλήματα διακέντρων (center problems) μπορούν να διακριθούν στις ακόλουθες δύο κατηγορίες:

1. vertex center problems: τα προβλήματα στα οποία ζητείται η χωροθέτηση του κέντρου παροχής υπηρεσιών μόνο στους κόμβους του δικτύου (κορυφές του γραφήματος), και

2. absolute center problems: τα προβλήματα στα οποία ζητείται η χωροθέτηση του κέντρου παροχής υπηρεσιών οπουδήποτε στο δίκτυο (είτε στις κορυφές του γραφήματος είτε πάνω στις ακμές του γραφήματος)

1.3.2 Προβλήματα Διακέντρων κορυφής (vertex center problems)

Εάν οι θέσεις των κέντρων παροχής υπηρεσιών είναι περιορισμένες στους κόμβους του δικτύου, τα προβλήματα αυτά ονομάζονται προβλήματα Διακέντρων κορυφής (vertex center problems και node center problems). (Christofides N., Viola P. 1971)

Το πρόβλημα αυτό τυποποιείται μαθηματικά ως εξής:

Ορίζουμε:

dij : η απόσταση ανάμεσα στους κόμβους κατανομής i και τις πιθανές θέσεις j

hi : η ζήτηση / απαίτηση του πελάτη στον κόμβο i

P : ο αριθμός των κέντρων παροχής υπηρεσιών που θα τοποθετηθούν

Οι μεταβλητές απόφασης (decision variables) είναι οι εξής:

$$X_j = \begin{cases} 1, & \text{αν τοποθετήσουμε το κέντρο παροχής υπηρεσιών στην πιθανή θέση } j. \\ 0, & \text{διαφορετικά} \end{cases}$$

(1.12)

Yij: το ποσοστό της ζήτησης στον κόμβο i που εξυπηρετείται από ένα κέντρο παροχής υπηρεσιών στον κόμβο j

W: η μέγιστη απόσταση μεταξύ του κόμβου ζήτησης (demand node) και του κοντινότερου κέντρου παροχής υπηρεσιών.

Βάσει των πιο πάνω συμβολισμών, το πρόβλημα Διακέντρων κορυφής (vertex P-center problem) τυποποιείται ως εξής:

MINIMIZE W (1.13)

υπό τους περιορισμούς: Yi

$$\sum_j Y_{ij} = 1 \qquad \forall i \qquad (1.14)$$

$$\sum_j X_j = 1 \qquad (1.15)$$

$$X_{ij} \qquad \forall i, j \qquad (1.16)$$

$$W \geq \sum_j d_{ij} \qquad \forall i \qquad (1.17)$$

$$Xj = 0\ ,1 \qquad (1.18)$$

$$Yij \geq 0 \qquad \forall i, j \qquad (1.19)$$

Η αντικειμενική συνάρτηση (Halfin S 1976) (1.13) είναι απλά η ελαχιστοποίηση της μέγιστης απόστασης μεταξύ οποιουδήποτε κόμβου ζήτησης και του κοντινότερου κέντρου παροχής υπηρεσιών. Ο περιορισμός (1.14) δείχνει ότι κάθε ζήτηση προσδιορίζεται σε κάποια θέση του κέντρου παροχής υπηρεσιών. Ο περιορισμός (1.15) απαιτεί ότι, ακριβώς P υπηρεσίες θα τοποθετηθούν ενώ ο περιορισμός (1.16) επιτρέπει την εισχώρηση μόνο των θέσεων στις οποίες έχουν τοποθετηθεί τα κέντρα παροχής υπηρεσιών. Ο περιορισμός (1.17) ορίζει την μέγιστη απόσταση μεταξύ οποιουδήποτε κόμβου απαίτησης i και του κοντινότερου κέντρου παροχής υπηρεσιών του j.

Τέλος, οι περιορισμοί (1.18) και(1.19) είναι περιορισμοί των μεταβλητών απόφασης.

1.3.3 Πρόβλημα p-διασποράς (p- dispersion)

Το πρόβλημα p-διασποράς (p-dispersion) τοποθετεί κέντρα παροχής υπηρεσιών για να μεγιστοποιήσει την ελάχιστη απόσταση ανάμεσα σε οποιοδήποτε ζευγάρι των κέντρων παροχής υπηρεσιών.

1.3.4 Προβλήματα Διαμέσων (Median Problems)

Σε μερικά προβλήματα απαιτείται να γίνει η χωροθέτηση ενός κέντρου παροχής υπηρεσιών σε κορυφή ενός δοσμένου γραφήματος κατά τέτοιο τρόπο ώστε να ελαχιστοποιείται το άθροισμα όλων των συντομότερων αποστάσεων των κορυφών του από αυτή την υπηρεσία. Η άριστη χωροθέτηση του κέντρου παροχής υπηρεσιών ονομάζεται διάμεσος (median) του γραφήματος και λόγω της φύσης της αντικειμενικής συνάρτησης αυτή η κατηγορία προβλημάτων ονομάζεται minsum προβλήματα χωροθέτησης (minsum location problems).

Το πρόβλημα παρουσιάζεται στην πράξη κάτω από διάφορες μορφές: χωροθέτηση κέντρων διακοπτών σε τηλεφωνικά δίκτυα, υποσταθμοί ηλεκτρικής ενέργειας σε δίκτυα, αποθήκες διανομής σε οδικό δίκτυο (όπου οι κορυφές παριστάνουν τους πελάτες). Οι πιο πάνω είναι μερικές από τις περιοχές εφαρμογής του προβλήματος minsum.

Η θεωρητική και ιστορική απαρχή της επίλυσης προβλημάτων χωροθέτησης –

κατανομών στα πλαίσια της θεωρίας γραφημάτων είναι ο αλγόριθμος Hakimi (1964) ο οποίος προσπάθησε να λύσει τη χωροθέτηση κέντρων σε ένα τηλεπικοινωνιακό σύστημα.

Ένα παράδειγμα στο οποίο ζητείται να βρεθεί η διάμεσος του γραφήματος, είναι αυτό της εξυπηρέτησης ενός αριθμού πελατών από μια μόνο αποθήκη. Οι πελάτες μπορούν να ομαδοποιηθούν κατά γειτονιές, έτσι ώστε κάθε ομάδα πελατών να απαιτεί ένα συνολικό αριθμό φορτώσεων.

Έτσι το φορτηγό φορτώνει από την αποθήκη, διανέμει τα εμπορεύματα σε μια ομάδα πελατών και γυρίζει στην αποθήκη. Μπορούμε να παραστήσουμε τις ομάδες πελατών με κορυφές του γραφήματος και το οδικό δίκτυο με τις ακμές του γραφήματος.

Στην πράξη, σε κάθε ομάδα πελατών μπορεί να αντιστοιχηθεί ένα βάρος wi , που να αντιπροσωπεύει τη «σπουδαιότητα» του πελάτη (π.χ. μπορεί να είναι ανάλογο προς την ετήσια ζήτηση αυτού του πελάτη ή να είναι η συχνότητα με την οποία το φορτηγό πρόκειται να επισκεφθεί αυτήν την ομάδα πελατών για να ικανοποιήσει την ζήτηση). Ο αντικειμενικός σκοπός θα είναι να βρεθεί εκείνη η θέση για την χωροθέτηση της αποθήκης ώστε το σύνολο των χιλιομέτρων, που θα καλύψει το φορτηγό να είναι ελάχιστο.

Ένας σημαντικός τρόπος για να μετρήσουμε την αποτελεσματικότητα της χωροθέτησης ενός κέντρου παροχής υπηρεσιών είναι να καθορίσουμε την απόσταση που χρειάζεται να διανύσει κάποιος που βρίσκεται σε ένα σημείο και θέλει να επισκεφτεί το συγκεκριμένο κέντρο παροχής υπηρεσιών. Με τον όρο κόστος της απόστασης εννοούμε τον χρόνο του ταξιδιού και την απόσταση του ταξιδιού από μια τοποθεσία θέση σε άλλη. Όσο η μέση απόσταση ταξιδιού αυξάνεται, η αποδοχή της τοποθέτησης μειώνεται και έτσι η αποτελεσματικότητα της συγκεκριμένης τοποθέτησης μειώνεται.

Αυτή η σχέση ισχύει κυρίως για κέντρα παροχής υπηρεσιών όπως οι βιβλιοθήκες, τα σχολεία και τα κέντρα υπηρεσιών άμεσης ανάγκης.

Ένας ισοδύναμος τρόπος για να μετρήσουμε την αποτελεσματικότητα της χωροθέτησης όταν οι απαιτήσεις δεν επηρεάζονται από το επίπεδο του κέντρου παροχής υπηρεσιών είναι η μέτρηση της απόστασης ανάμεσα στους κόμβους ζήτησης και τα κέντρα παροχής υπηρεσιών. Αυτό γίνεται συνδέοντας την ποσότητα της κάθε απαίτησης και υπολογίζοντας το ολικό βάρος της σταθμικής απόστασης ανάμεσα στη ζήτηση και το συγκεκριμένο κέντρο παροχής υπηρεσιών.

1.3.5 Πρόβλημα P-Διάμεσος (P-median)

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

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

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

Το πρόβλημα αυτό τυποποιείται μαθηματικά ως εξής:

Ορίζουμε:

i : οι κόμβοι κατανομής/ ζήτησης

j : οι ενδεχόμενες (potential) θέσεις του κάθε κέντρου παροχής υπηρεσιών.

hi : η ζήτηση / απαίτηση του πελάτη στον κόμβο i

dij : η απόσταση ανάμεσα στους κόμβους κατανομής i και τις πιθανές θέσεις j

P : ο αριθμός των κέντρων παροχής υπηρεσιών που θα τοποθετηθούν

Οι μεταβλητές απόφασης (decision variables) είναι οι εξής:

$$X_j = \begin{cases} 1, & \text{αν τοποθετήσουμε το κέντρο παροχής υπηρεσιών στην πιθανή θέση } j. \\ 0, & \text{διαφορετικά} \end{cases}$$

$$Y_{ij} = \begin{cases} 1, & \text{αν η ζήτηση στον κόμβο } i \text{ οφείλεται στο κέντρο παροχής υπηρεσιών του κόμβου } j. \\ 0, & \text{διαφορετικά} \end{cases}$$

Έτσι, βάσει των πιο πάνω συμβολισμών, το πρόβλημα P-διάμεσος τυποποιείται ως εξής:

Αντικειμενική Συνάρτηση

MINIMIZE $\sum_i \sum_j h_i d_{ij} Y_{ij}$ (1.20)

Υπό τους περιορισμούς

$\sum_j Y_{ij} = 1 \quad \forall i$ (1.21)

$\sum_j X_j = 1$ (1.22)

$Y_{ij}-X_{ij} <0 \quad \forall i, j$ (1.23)

$Xj = 0 ,1$ (1.24)

$$Yij = 0{,}1 \qquad\qquad \forall i, j \quad (1.25)$$

Η σχέση (1.20) είναι η ελαχιστοποίηση της ολικής σταθμικής απαιτούμενης απόστασης ανάμεσα στους πελάτες και τα κέντρα παροχής υπηρεσιών. Ο περιορισμός (1.21) δείχνει ότι κάθε ζήτηση προσδιορίζεται σε κάποιο facility site. Ο περιορισμός (1.22) απαιτεί ότι ακριβώς P υπηρεσίες θα τοποθετηθούν ενώ ο περιορισμός (1.23) επιτρέπει την εισχώρηση μόνο των θέσεων στις οποίες έχουν τοποθετηθεί τα κέντρα παροχής υπηρεσιών.

Οι περιορισμοί (1.24) και (1.25) είναι οι απαιτήσεις για τις μεταβλητές απόφασης του προβλήματος. Αυτή η τυποποίηση του προβλήματος P-Διάμεσος επιτρέπει στα κέντρα παροχής υπηρεσιών να είναι τοποθετημένα σε ένα πεπερασμένο σύνολο των ενδεχόμενων θέσεων (potential sites). Αυτές οι θέσεις αναπαριστάνουν τους κόμβους του δικτύου. Για οποιονδήποτε αριθμό των κέντρων παροχής υπηρεσιών P υπάρχει το λιγότερο μια κατάλληλη λύση του προβλήματος P-Διάμεσος η οποία τοποθετείται μόνο στους κόμβους του δικτύου. Έτσι, η απλοποιημένη τυποποίηση περιλαμβάνει μόνο τους κόμβους σαν ενδεχόμενη τοποθεσία κάποιου κέντρου παροχής υπηρεσιών και δεν αλλοιώνει την τιμή της αντικειμενικής συνάρτησης.

1.3.6 Fixed Charge Facility Location Model

Στις προηγούμενες κατηγορίες, έχουμε εστιάσει στην απόσταση ταξιδιού ή στον χρόνο του ταξιδιού ως υπόδειξη της τιμής του λειτουργικού κόστους μόλις τοποθετηθεί ένα κέντρο παροχής υπηρεσιών. Αν και αναγνωρίζουμε ότι οι περιορισμένοι πόροι καθορίζουν τον αριθμό των κέντρων παροχής υπηρεσιών που είναι εγκατεστημένα, μόνο σε ένα πρότυπο (το set covering) εξετάσαμε ρητά τις δαπάνες της τοποθέτησης (location costs). Το σύνολο των προβλημάτων χωροθέτησης fixed charge περιλαμβάνει τις περιπτώσεις προβλήματος που έχουν μια σταθερή δαπάνη σε σχέση με την τοποθέτηση σε κάθε πιθανή θέση του κέντρου παροχής υπηρεσιών (facility site).

Στο μοντέλο P-median έχουν γίνει τρεις σημαντικές παραδοχές: Πρώτον, υποθέτει ότι κάθε εναλλακτική θέση έχει το ίδιο σταθερό κόστος εγκατάστασης. Δεύτερον, υποθέτει ότι τα κέντρα παροχής υπηρεσιών που θα τοποθετηθούν δεν εξαρτώνται από τη ζήτηση που θα εξυπηρετήσουν. Τέλος, υποθέτει ότι ο αριθμός των κέντρων παροχής υπηρεσιών που θα εγκαταστήσουμε είναι γνωστός εκ των προτέρων.

Το Fixed Charge Facility Location Model, FCLP αγνοεί αυτές τις παραδοχές. Στόχος του είναι να ελαχιστοποιήσει το συνολικό κόστος τοποθέτησης και το συνολικό κόστος ταξιδιού για την εξυπηρέτηση της ζήτησης, καθορίζοντας το βέλτιστο αριθμό και τοποθεσίες των κέντρων παροχής υπηρεσιών καθώς επίσης και την ανάθεση της ζήτησης σε ένα κέντρο. Δεδομένου ότι τα κέντρα έχουν συγκεκριμένες δυνατότητες εξυπηρέτησης, ή ζήτηση μπορεί να μην ανατεθεί στο κοντινότερο κέντρο όπως στην περίπτωση των μοντέλων που παρουσιάστηκαν πιο πάνω.

Κεφάλαιο 2. Τεχνικές επίλυσης Προβλημάτων κάλυψης συνόλου

2.1 Meta-RAPS Heuristic

2.1.1 Γενική Περιγραφή

Γενικά οι Μέτα Ευρετικές Μεθόδοι είναι μέθοδοι που βελτιώνουν αρχικές λύσεις που έχουν βρεθεί από κάποιον άλλο αλγόριθμο(Chvatal, V., 1979). Έτσι αν έχουμε μια σειρά λύσεων που προκύπτουν με την χρήση κάποιου αλγόριθμου οι Μετα ευρετικές μέθοδοι βελτιώνουν τις λύσεις μας ή επιλέγουν την καλύτερη από αυτές.

Έτσι οι Μέτα Ευρετικές Μέθοδοι βελτιώνουν την αρχική λύση που δημιουργήθηκε στο προηγούμενο βήμα με τη χρήση μετα-ευρετικών τεχνικών. Η Γενική λογική που ακολουθείται είναι η εξής:

Βήμα 1 Δημιούργησε μια αρχική λύση X

Βήμα 2 Αποτίμησε το κόστος της X

Βήμα 3 Δημιούργησε μια νέα λύση X' στην γειτονιά της X

Βήμα 4 Αποτίμησε το κόστος της X'

Βήμα 5 Αν Κόστος(X') < Κόστος(X) τότε θέσε X=X' στο Βήμα 3

Βήμα 6 Τέλος

Οι αλγόριθμοι Meta RaPS (μετα ευρετικοί αλγόριθμοι τυχαίας προτεραιότητας αναζήτησης) είναι μετα ευρετικοί αλγόριθμοι που ακολουθούν την γενική λογική των Μετά ευρετικών αλγορίθμων αλλά επιλέγουν σε δείγμα των λύσεων τυχαίες λύσεις και συγκρίνουν μεταξύ αυτών.

Η διαδικασία της τυχαίας αναζήτησης είναι από τις πιο απλοϊκές. Αρχικά θεωρεί το μηδέν ως λύση και συνεχίζει επιλέγοντας ένα τυχαίο σημείο από το σύνολο των πιθανών λύσεων. Έπειτα υπολογίζει τη τιμή που του δίνει η συνάρτηση f(x) θεωρώντας το x ως λύση και την αποθηκεύει. Εάν μέχρι στιγμής δεν έχει βρεθεί μεγαλύτερη τιμή για το f(x) το πιο πρόσφατο x θεωρείται και το βέλτιστο. Ένας αλγόριθμος είναι μία συγκεκριμένη ακολουθία βημάτων η οποία επιδιώκει να λύσει ένα πρόβλημα. Οι εντολές που συνθέτουν τη τυχαία αναζήτηση είναι οι εξής:

« λύση» = 0.0

Καθώς μια ικανοποιητική λύση δεν έχει βρεθεί, κάνε τα παρακάτω

Διάλεξε ένα τυχαίο σήμείο, x, στο πεδίο ορισμού

Εκτίμησε το f(x) σε αυτό το σημείο. Εάν το f(x) > « λύση», σώσε το x ως τη καλύτερη λύση που έχει βρεθεί μέχρι τώρα

5. Επέστρεψε στο βήμα 2

Όπως εύκολα γίνεται αντιληπτό αρκετά προβλήματα λύνονται εύκολα με αυτήν τη μέθοδο. Είναι εμφανές ότι ο αλγόριθμος αυτός έχει 1/256 πιθανότητα για το f(x) να βρεθεί το μέγιστο από κάθε επανάληψη και για αυτό το λόγο ίσως χρειαστεί να πραγματοποιηθούν επαναλήψεις για σχετικά μεγάλης διάρκειας περιόδους μέχρι να βρεθεί το μέγιστο. Βέβαια, σε αυτό το σημείο, προκύπτει και ένα δεύτερο πρόβλημα. Πως καθορίζεται πότε πρέπει να τερματιστεί η τυχαία αναζήτηση; Προσεκτική ανάλυση της συνάρτησης ίσως θα οδηγούσε σε κάποιες κατευθυντήριες γραμμές όσο αφορά αυτό το θέμα αλλά πιθανόν να μην είναι γνωστή εξαρχής η μορφή της συνάρτησης. Εν τούτοις η μέθοδος της τυχαίας αναζήτησης έχει ένα θετικό προσόν, ότι είναι «εξερευνητική». Συγκεκριμένα, η αναζήτηση θα πραγματοποιηθεί σε πολλές διαφορετικές περιοχές της συνάρτησης δίνοντας ομοιόμορφα κατανεμημένα δείγματα.

	Θετικά	Αρνητικά
Τυχαία αναζήτηση	Εξερεύνηση Αποτελεσματικότερη σε ασυνεχής συναρτήσεις ες	Αποτιμά τις μη κατάλληλες περιοχές πολλές φορές. Δεν έχει «μνήμη». Χρονοβόρα.

2.1.2 Meta RAPS στο πρόβλημα κάλυψης συνόλου

Τα βασικά βήματα για να εφαρμοστεί με επιτυχία η Meta-RAPS είναι να γίνει αποτελεσματικός σχεδιασμός στην heuristics συνάρτηση βελτίωσης.

Η πρώτη προσπάθεια είναι να τροποποιηθεί η ευρετική του Chvatal's (1979) με άπληστους αλγόριθμους. Αυτή η ευρετική αξιολογεί κάθε i στήλη από τη συνάρτηση f (Cj, kj) = Cj / kj, όπου Cj είναι το κόστος της στήλης j, και kj είναι ο αριθμός των γραμμών καλύπτονται από την j στήλη, και πάντα προσθέτει σε ένα σύνολο λύση (X), η στήλη με την ελάχιστη αξία Cj / kj. Σε Meta-RAPS, επιλέγουμε ακόμη (Cj, kj) = Cj /

kj ως κανόνας προτεραιότητας, έτσι τους ώστε η βαθμολογία για κάθε στήλη j(PJ) να είναι Cj / kj. Μόνο κατά τη διάρκεια της προτεραιότητας % του χρόνου, η στήλη με την ελάχιστη αξία του Pj θα επιλεγεί, ενώ το υπόλοιπο του χρόνου, μια στήλη από το CL θα επιλέγονται τυχαία. Μετά από μια εφικτή λύση είναι κατασκευασμένη, όλες τις περιττές στήλες (i στήλη είναι περιττή, εάν Xn (i) εξακολουθεί να είναι σε κάλυψη) θα αφαιρεθεί από το σύνολο κατά τρόπο ώστε οι περιττές στήλη με μεγαλύτερο κόστος θα πρέπει να αφαιρεθούν από την πρώτη.

Ο αλγόριθμος Meta-Raps για το πρόβλημα του Set Covering περιγράφεται παρακάτω όπως αυτός αναφέρεται από τους Guanghui Lan et. Al. (2007)

procedure Meta-RaPS-SCP-Construction (I, J, %priority, %restriction)

Set the solution set to be empty: $X = \phi$

Set I^* be the set of the currently uncovered rows: $I^* = I$

while $I^* \neq \phi$

Select $f(c_j, k_j) = c_j / k_j$ as the priority rule, and find the best candidate ω such that

$$f(c_\omega, k_\omega) = \min_{j \in J \setminus X}(f(c_j, k_j))$$

P = RND(1,100)

if P > %priority **then**

Construct the candidate list CL:

$$CL = \{ j : j \in J \setminus X \; and \; c_j / k_j \leq c_\omega / k_\omega \times (1 + \%restriction/100) \}$$

Randomly select an element ω_1 from CL and set $\omega = \omega_1$

end if

Add element ω to the solution X: $X = X \cup \{\omega\}$; $I^* = I^* \setminus I_\omega$

end while

Remove redundant columns from X

return X

end Meta-RaPS-SCP-Construction

Fig. 1. Pseudo-code of Meta-RaPS SCP construction.

for each $i \in I_\omega$

Update the number of selected columns for each row i: $\eta_i = \eta_i + 1$

for each $j \in J_i$

$k_j = k_j - 1$

end for

end for

2.1.3 Μέθοδοι βελτίωσης της ποιότητας λύσης

Για να βελτιωθεί η ποιότητα της λύσης, μια διαδικασία αναζήτησης γειτονιάς μπορεί να εφαρμοστεί μετά από την κατασκευή.

Αυτή η έρευνα καθορίζει τις γειτονικές λύσεις ως εξής: Εάν δύο λύσεις μοιράζονται τουλάχιστον μια στήλη, αυτές οι δύο λύσεις καλούνται γειτονικές λύσεις, δηλ., λαμβάνοντας υπόψη δύο λύσεις X1 και X2, εάν η τομή των X1 και X2 είναι διάφορη του Φ, τότε X1 και X2 είναι γειτονικές λύσεις, διαφορετικά πρέπει και πάλι να χωριστούν οι λύσεις.

Μια γειτονική λύση λαμβάνεται μέσω μίας διαδικασίας δύο βημάτων. Κατ' αρχάς, διάφορες στήλες, όπως καθορίζονται από μια από το χρήστη παράμετρο, αφαιρούνται τυχαία από τη δεδομένη εφικτή λύση, έτσι η λύση θα γίνει ανέφικτη επειδή υπάρχουν μερικές μη καλυμμένες σειρές. Κατόπιν η μερική λύση γίνεται εφικτή με την επίλυση ενός μειωμένου μεγέθους Set Covering προβλήματος που αποτελείται από τις μη καλυμμένες σειρές και τις στήλες που θα μπορούσαν να καλύψουν αυτές τις σειρές.

Ο ψευδοκώδικας της διαδικασίας αναζήτησης γειτόνων παρουσιάζεται παρακάτω όπως αναφέρεται από τους Guanghui Lan et. Al. (2007). Η παράμετρος search_magnitude χρησιμοποιείται για να ελέγξει πόσες στήλες θα αφαιρεθούν από τη λύση. Ο αριθμός αφαιρούμενων στηλών είναι ίσος με το |X| * search_magnitude.

procedure NeighborSearch (I, J, X, %priority, %restriction. search_magnitude, imp_iteration)

  for iter = 1, …, imp_iteration

    Randomly remove columns from X, the maximum number of columns to remove equals to: $|X| \times search\_magnitude$

    Formulate a reduced-size SCP:

$$I' = I \setminus \bigcup_{m \in X} I_m \qquad J' = \bigcup_{i \in I'} J_i$$

    Solve this reduced-size SCP:

      X' = Meta-RaPS-SCP-Construction (I', J', %priority, %restriction)

    Construct the neighboring solution: $X' = X' \cup X$

    Remove redundant columns from X'

    if the objective function value of X' is less than that of X **then**

      X = X'

    end if

  end for

  return X

end NeighborSearch

Pseudo-code of neighbor search procedure.

procedure Meta-RaPS-SCP(I_0, J_0, %priority, %restriction, %improvement, max_iteration, search_magnitude, imp_iteration)

Perform preprocessing and get the reduced row set $I \subseteq I_0$ and column set $J \subseteq J_0$

Set the initial solutions to empty: $X^* = X = \phi$

$Z^* = Z^*_{before\ improvement}$ = LARGE_NUMBER {set to a large number}

Set the initial core problem to empty: $J_c = \phi$

for it = 1, …, max_iteration

    {the construction phase}

    Call Meta-RaPS-SCP_Construction (I, J, %priority, %restriction)

    and update the core problem J_c if desirable

    $Z = \sum_{j \in X} c_j$, where X is the solution from construction phase.

    if $Z < Z^*_{before\ improvement}$ **then** $Z^*_{before\ improvement} = Z$

    {the improvement phase}

    if $Z \leq (1 + \%improvement) \times Z^*_{before\ improvement}$ **then**

        NeighborSearch (I, J_c, X, %priority, %restriction,

            search_magnitude, imp_iteration)

    Penalize the worst elements in X

    {Update of the best solution}

    if ($Z < Z^*$) **then**

        $Z^* = Z$

        $X^* = X$

    end if

end for

return X^*

end Meta-RaPS-SCP

Pseudo-code of final Meta-RaPS SCP algorithm.

2.1.4 Μέθοδοι βελτίωσης της ταχύτητας υπολογισμού

Εκτός από τη βελτίωση της ποιότητας λύσης, παρουσιάζονται και δύο μέθοδοι για την μείωση του χρόνου εκτέλεσης της Meta-RAPs για το πρόβλημα κάλυψης.

Προεπεξεργασία

Η προεπεξεργασία είναι μια δημοφιλής μέθοδος την αύξηση της ταχύτητας του αλγόριθμου. Διάφορες μέθοδοι προεπεξεργασίας για το πρόβλημα κάλυψης έχουν προταθεί στη διεθνή βιβλιογραφία όπως από τον Beasley (1987).

Δύο από τους συχνότερους αλγόριθμους που χρησιμοποιούνται περιγράφονται παρακάτω

• Κυριαρχία στηλών: Οποιαδήποτε στήλη j οι της οποίας σειρές Ij μπορούν να καλυφθούν από άλλες στήλες για ένα κόστος λιγότερο από Cj μπορεί να διαγραφεί από το πρόβλημα. Όπως αναμένεται, αυτή η μείωση είναι ατελέσφορη για τα προβλήματα ενιαίου κόστους

• Συνυπολογισμός στηλών: Εάν μια σειρά καλύπτεται από μόνο μια στήλη μετά από την ανωτέρω κυριαρχία, αυτή η στήλη πρέπει να περιληφθεί στη βέλτιστη λύση.

2.1.5 Καθορισμός του προβλήματος πυρήνων

Δεδομένου ότι η διαδικασία αναζήτησης γειτόνων είναι imp_iteration επαναλήψεων για κάθε κατασκευασμένη λύση, η φάση βελτίωσης είναι πιο υπολογιστικά προσεγγίσιμη από τη φάση κατασκευής. Επομένως, η μείωση του χρόνου εκτέλεσης της διαδικασίας αναζήτησης γειτόνων είναι σημαντικότερη για την μείωση του συνολικού χρόνου εκτέλεσης επιταχύνοντας την κατασκευή της ευρετικής. Προκειμένου να μειωθεί ο χρόνος υπολογισμού, αυτός ο αλγόριθμος ενσωματώνει μια διαδικασία για να καθορίσει το πρόβλημα πυρήνων, που περιλαμβάνει μόνο ένα μικρό υποσύνολο των ενδιαφερουσών στηλών, για τη βελτίωση του ευρετικού για να εργαστεί, έτσι ώστε το μειώσει το μέγεθος του προβλήματος κάλυψης.

Το πρόβλημα πυρήνων ορίζεται ως εκείνες οι στήλες που επιλέγονται τουλάχιστον μια φορά στον κατάλογο υποψηφίων κατά τη διάρκεια της φάσης κατασκευής. Αρχικά το πρόβλημα Jc πυρήνων τίθεται κενό. Κατά τη διάρκεια κάθε βήματος κατασκευής, το πρόβλημα πυρήνων ενημερώνεται σαν Jc = Jc U CL. Τόσο μόνο μια μικρή τροποποίηση στην κατασκευή απαιτείται για να καθορίσει το πρόβλημα πυρήνων.

Μετά από την κατασκευή το νέο μειώνω-μέγεθος SCP για τη διαδικασία αναζήτησης γειτόνων είναι το ακόλουθο

$$\min\left\{\sum_{j\in J'\cap J_c} c_j x_{j'} : \sum_{j\in J'\cap J_c} a_{ij}x'_j \geqslant 1, i\in I'; \quad x'_j = 0 \text{ or } 1,\ j\in J'\cap J_c\right\},$$

όπου

$$I' = I \Big\backslash \bigcup_{m\in X} I_m, \quad J' = \bigcup_{i\in I'} J_i,$$

με Jc να είναι το βασικό πρόβλημα

Προφανώς αυτό το πρόβλημα κάλυψης έχει τουλάχιστον μια εφικτή λύση, δηλ. η εφικτή λύση από την κατασκευή της ευρετικής.

Παρατηρήστε ότι το μέγεθος του προβλήματος πυρήνων θα αυξηθεί δεδομένου ότι περισσότερες επαναλήψεις έχουν οργανωθεί για αυτόν τον αλγόριθμο.

Ενδεχομένως, οι στήλες στο πρόβλημα πυρήνων θα μπουν σε ένα σταθερό κράτος μετά από ορισμένες επαναλήψεις.

Αν και καθορίζοντας τον πυρήνα το πρόβλημα μπορεί πολύ να μειώσει το συνολικό χρόνο εκτέλεσης, έχει επίσης μερικά μειονεκτήματα.

Δεδομένου ότι απομακρύνει μερικές στήλες από την εκτίμηση στη φάση βελτίωσης, η δυνατότητα να βρεθεί μια βέλτιστη λύση να μειωθεί. Σαν αποζημίωση σε αυτό το αρνητικό αποτέλεσμα, διαπιστώνεται ότι η παράμετρος restriction και improvement πρέπει να τεθεί ελάχιστα έναν μεγαλύτερο για να βρει περισσότερες βέλτιστες λύσεις εάν το πρόβλημα πυρήνων καθορίζεται.

2.2 Genetic Algorithm

2.2.1 Γενικά

Γενικά ένας γενετικός αλγόριθμος είναι ένας αλγόριθμος αναζήτησης λύσεων και επίλυσης προβλημάτων. (Melanie Mitchell 1993) Αρχικά «εξελίσσει» ένα σύνολο πιθανών λύσεων μέχρι να βρεθεί εκείνη που ικανοποιεί το ζητούμενο. Η βασική ιδέα είναι ότι μια συγκεκριμένη λύση θα μεταδώσει το περιεχόμενο της, το οποίο είναι μια πληροφορία, στους απογόνους της με την προϋπόθεση ότι αυτή είναι μια « καλή λύση ». Όσο πιο καλή είναι μια λύση, τόσο μεγαλώνει η πιθανότητα να μεταδώσει τη πληροφορία της, δηλαδή με άλλα λόγια επικρατεί η ισχυρότερη. Για να γίνει πιο κατανοητός ο ορισμός των γενετικών αλγόριθμων αναλύεται πως η μέθοδος αυτή μιμείται το μηχανισμό εξέλιξης των οργανισμών στη φύση. Σύμφωνα με αυτήν, οι οργανισμοί που δεν μπορούν επιβιώσουν πεθαίνουν, ενώ οι υπόλοιποι πολλαπλασιάζονται με τη βοήθεια της αναπαραγωγής. Οι απόγονοι τους δεν διαφέρουν πολύ από τους προγόνους, ενώ στη συνέχεια υπερισχύουν εκείνοι που παρουσιάζουν τα καλύτερα χαρακτηριστικά. Στη περίπτωση όπου συμβαίνουν μεταλλάξεις στη φύση, οι μεταλλαγμένοι οργανισμοί συνήθως πεθαίνουν αλλά υπάρχει και μια μικρή πιθανότητα να οδηγήσουν στη δημιουργία οργανισμών με βελτιωμένα χαρακτηριστικά.

Η παρουσίαση των γενετικών αλγορίθμων έγινε από τον Friedberg το 1958, ο οποίος χρησιμοποίησε και την Fortran στην προσπάθειά του να επιλύσει κάποια προβλήματα. (Z.Michalevicz, 1995) Το 1975 ο Holland ενίσχυσε αυτή τη μέθοδο χρησιμοποιώντας σειρές bits για να αναπαραστήσει λειτουργίες, με τρόπο ώστε κάθε συνδυασμός bits να είναι μια έγκυρη λειτουργία. Στο προγραμματισμό παραγωγής η μέθοδος γενετικών αλγορίθμων εφαρμόστηκε από τον Davis μόλις το 1985 ο οποίος κατάφερε να κατασκευάζει μια προτιμημένη διαταγή των διαδικασιών για κάθε μηχανή. Νέα ώθηση στο χώρο έδωσαν οι Falkenauer και Bouffouix το 1991 οι οποίοι οδήγησαν στην εισαγωγή ενός μοντέλου κωδικοποίησης των διαδικασιών που πραγματοποιούνται σε μια μηχανή κατά την λειτουργία της. Ακόμα πιο πρόσφατα μόλις το 1995 ο Kobayashi εισήγαγε μια νέα τεχνική σύμφωνα με την οποία ένα χρωμόσωμα είναι μια σειρά από σύμβολα μήκους n και κάθε σύμβολο αναγνωρίζει μια διαδικασία που πρόκειται να ανατεθεί σε μια μηχανή. Τέλος, πιο πρόσφατα, ο Shi (1997) εφαρμόζει μια τεχνική διασταυρώσεων που διαιρεί τυχαία έναν αυθαίρετα επιλεγμένο σύντροφο σε δύο υποσύνολα, από τα οποία παράγεται ο απόγονος.

2.2.2 Genetic Algorithm για το πρόβλημα κάλυψης συνόλου

Ανάλυση των παραμέτρων του γενετικού αλγόριθμου

- ***Συνάρτηση καταλληλότητας (Fitness function).***

Αρχικά η συνάρτηση η οποία πρέπει να μεγιστοποιηθεί κατά τη διάρκεια μιας διαδικασίας είναι αυτή που ονομάζεται *συνάρτηση καταλληλότητας (Fitness function)* (Z.Michalevicz, 1995). Πιο συγκεκριμένα, αυτή παίρνει κάθε στοιχείο του πεδίου των πιθανών λύσεων και το αποτιμά. Αυτές οι πιθανές λύσεις δεν είναι απαραίτητα οι βέλτιστες, ενώ είναι εύλογο να είναι και μη έγκυρες εφόσον ακόμα δεν έχει γίνει καμία διεργασία, δηλαδή αυτές είναι το αρχικό τυχαίο σύνολο από το οποίο στη συνέχεια θα βρεθούν οι επιθυμητές λύσεις. Η συνάρτηση καταλληλότητας δέχεται ως είσοδο ένα χρωμόσωμα και επιστρέφει έναν αριθμό που υποδηλώνει το πόσο κατάλληλο είναι. Το πεδίο τιμών της συνάρτησης καταλληλότητας είναι συνήθως το διάστημα [0.0,1.0], αν και αυτό ανάλογα την υλοποιήση θα μπορούσε να διαφέρει. Η τιμή 1 υποδηλώνει ότι το συγκεκριμένο χρωμόσωμα είναι τέλειο, δηλαδή ικανοποιεί όλες τις προϋποθέσεις του προβλήματος και αποτελεί αποδεκτή λύση.

Όπως εύλογα έχει ειπωθεί, παράλληλα με την διαδικασία κωδικοποιήσης, η λειτουργία της αποτίμησης είναι η πιο κρίσιμη φάση κάθε γενετικού αλγόριθμου. Η επιτυχία των υπολογισμών που γίνονται με τη βοήθεια των γενετικών αλγόριθμων συνδέεται στενά στη διάκριση με την οποία ένας αλγόριθμος μπορεί να διαλέξει ανάμεσα στις τόσες λύσεις με πολυποίκιλη αποτελεσματικότητα. Αυτή την δυνατότητα την προσφέρει η συνάρτηση καταλληλότητας. Όταν η συνάρτηση αυτή χρησιμοποιείται τα αποτελέσματα μιας μελέτης προσποιητικής διάταξης αποδεικνύουν ότι το πεδίο των τιμών το οποίο παράγεται είναι σχετικά μικρό, ακόμα και μεταξύ των καλύτερων χρωμοσωμάτων, αρκεί η συνάρτηση αυτή να μην έχει πολλά τοπικά μέγιστα ή ένα απομονωμένο ολικό μέγιστο. Τέλος, αξίζει να αναφερθεί ότι σε αρκετά προβλήματα στη θέση της συνάρτησης καταλληλότητας μπορεί να χρησιμοποιηθεί κάποια άλλη καλή προσέγγιση της (approximate fitness function), όταν αυτό που ενδιαφέρει δεν είναι τόσο οι ακριβείς υπολογισμοί όσο ο χρόνος υπολογισμού, εφόσον η ακριβής συνάρτηση καταλληλότητας μπορεί να γίνει αρκετά χρονοβόρα.

Διαδικασία επιλογής χρωμοσωμάτων (selection)

Η διαδικασία επιλογής γονέων (selection) συνίσταται στον καθορισμό των χρωμοσωμάτων από τα οποία θα αναγεννηθεί ο επόμενος πληθυσμός. Η συνάρτηση καταλληλότητας παίζει καθοριστικό ρόλο στη μέθοδο αυτή, αφού ανάλογα τις τιμές της θα προσδιοριστούν ποιοι γονείς είναι αρκετά «υγιείς» ώστε να αναπαράγουν τα μέλη ενός πληθυσμού. Έτσι, χρωμοσώματα με υψηλές τιμές καταλληλότητας έχουν και υψηλή πιθανότητα να επιλεγούν ενώ αντίθετα γονείς με χαμηλή καταλληλότητα έχουν μηδαμινή πιθανότητα. Κατά τη διαδικασία επιλογής αρχικά ο πληθυσμός περιέχει όλες τις δυνατές υποψήφιες συμβολοσειρές αλλά και τις τιμές της συνάρτησης καταλληλότητας. Έπειτα, με διάφορες τεχνικές και με τα παραπάνω δεδομένα γίνεται η επιλογή των μελλοντικών γονέων δίνοντας προτεραιότητα στις πλέον κατάλληλες λύσεις. Μερικές απο τις πιο διαδεδομένες μεθόδους επιλογής είναι η τεχνική Goldberg’s biased wheel και η τεχνική roulette wheel selection που περιγράφονται στη συνέχεια.

Τεχνική Goldberg's biased wheel.

Σύμφωνα με τη μέθοδο αυτή, κάθε τρέχων χρωμόσωμα στο πληθυσμό κατέχει ένα μερίδιο σε μια «ρόδα ρουλέτας» (roulette wheel) σε αναλογία με την τιμή της καταλληλότητας του. Για το κάθε χρωμόσωμα η σχετική τιμή καταλληλότητας Firel υπολογίζεται σύμφωνα με τη σχέση:

$$F_i^{rel} = \frac{F_i}{\sum_{i=1}^{ni} F_i}$$

όπου το ni αναπαριστά το ολικό αριθμό χρωμοσωμάτων και $\sum_{i=1}^{ni} F_i^{rel} = 1$.

Για να δημιουργηθεί ο νέος παράγωγος πληθυσμος , ni τυχαίες επιλογές πραγματοποιούνται με τον ακόλουθο τρόπο. Μια σειρά εντολών υπολογιστή τυχαίας επιλογής επιστρέφει ένα ομοιόμορφο τυχαίο αριθμό R που παίρνει τιμές μεταξύ του 0 και του 1 ($0 \leq R \leq 1$). Το επιλεγμένο χρωμόσωμα θα επαληθεύει το παρακάτω τύπο:

$$\sum_{i=1}^{ni} F_i^{rel} \leq R$$

Χρησιμοποιώντας το συγκεκριμένο πλάνο εργασίας, τα «υγιέστερα» χρωμοσώματα ευνοούνται. Είναι βέβαια φανερό από τα παραπάνω βήματα ότι κάποιες υποψήφιες λύσεις μπορεί να επιλεγούν περισσότερες από μία φορές. Μετά την ολοκλήρωση της διαδικασίας που περιγράφτηκε στο νέο βελτιωμένο πληθυσμό δημιουργούνται ζευγάρια με τυχαίο τρόπο και στη συνέχεια, χρησιμοποιώντας τους μηχανισμούς της διασταύρωσης (cross-over) και της μετάλλαξης (mutation), οι απόγονοι που προκύπτουν αποτελούν ένα τελείως καινούργιο πληθυσμό.

Κατά την αρχική εφαρμογή των γενετικών αλγόριθμων για την επίλυση ενός προβλήματος, είναι αρκετά λογικό να υπάρχουν μερικά μόνο πολύ καλά χρωμοσώματα σε ένα πληθυσμό γεμάτο από μη αποδεκτά χρωμοσώματα. Όταν η τεχνική επιλογής Goldberg's biased wheel χρησιμοποιείται, αυτόματα σημαίνει ότι αυτά τα καλά χρωμοσώματα ή μικρές παραλλαγές τους, θα επικρατήσουν σε μία αρκετά σημαντική ποσοστιαία αναλογία σε μια γενεά, γεγονός που κυρίως οδηγεί στο φαινόμενο της πρόωρης σύγκλισης (premature convergence) του αλγόριθμου. Είναι ευνόητο λοιπόν ότι κατά το φαινόμενο αυτό ο πληθυσμός πολύ γρήγορα συγκλίνει γύρω από κάποιο συγκεκριμένο χρωμόσωμα, το οποίο όμως αποτελεί τοπικό μέγιστο. Άμεσο αποτέλεσμα είναι να μη μπορεί ο γενετικός αλγόριθμος να ξεφύγει από αυτό το τοπικό μέγιστο, παρά μόνο με τη μέθοδο της μετάλλαξης η οποία όμως, όπως και στη φύση, είναι πολύ σπάνιο να πραγματοποιηθεί. Για να αποφευχθεί αυτό το πρόβλημα μπορεί να εισαχθεί στη μέθοδο επιλογής ένα ανώτατο και ένα κατώτατο όριο στον αριθμό των αντιγράφων του ίδιου γονιδίου στον πληθυσμό.

Βέβαια εκτός από το φαινόμενο της πρόωρης σύγκλισης μπορεί να παρατηρηθεί και το ακριβώς αντίθετο φαινόμενο της αργής σύγκλισης (slow convergence). Σύμφωνα με αυτό το πρόβλημα υπάρχει περίπτωση μετά από ένα αρκετά μεγάλο κύκλο επαναλήψεων, ο πληθυσμός να μην έχει «συγκλίνει». Ως «σύγκλιση» χαρακτηρίζεται η επικράτηση ενός χρωμόσώματος σε μεγάλο ποσοστό πληθυσμού, που μπορεί να φθάσει και στο 95% των χρωμοσωμάτων. Το φαινόμενο της αργής σύγκλισης συνήθως οφείλεται στο ότι η συνάρτηση καταλληλότητας μπορεί περιέχει ελάχιστες απότομες μεταβολές, με αποτέλεσμα τα μέγιστα και τα ελάχιστα της να έχουν μικρές διαφορές. Σε αυτή την περίπτωση μία λύση είναι η αντικατάσταση της συνάρτησης από κάποια προσεγγιστική συνάρτηση με μεγαλύτερες κλίσεις.

- *Τεχνική rulette wheel.*

Μία ακόμα πολύ συνηθισμένη τεχνική επιλογής των χρωμοσωμάτων από τα οποία θα προκύψουν οι νέοι απόγονοι του καινούργιου πληθυσμού είναι η μέθοδος roulette wheel selection [Dav91]. Ακολουθούν τα βήματα του μηχανισμού που χρησιμοποιεί από τα οποία θα γίνει αντιληπτό ότι μερικές υποψήφιες λύσεις μπορεί να επιλεχθούν περισσότερες από μία φορές. Δίνεται μεγάλη σημασία στον τρόπο επιλογής εφόσον αυτός επηρεάζει σημαντικά την απόδοση των γενετικών αλγόριθμων.

Αθροίζονται οι τιμές καταλληλότητας όλων των χρωμοσωμάτων του πληθυσμού. Ονομάζεται αυτό το άθροισμα ολικής καταλληλότητας (total fitness).

Παράγεται ένας τυχαίος αριθμός n, που παίρνει τιμές από το 0 μέχρι την ολική τιμή καταλληλότητας.

Επιστρέφεται το πρώτο χρωμόσωμα του πληθυσμού του οποίου η καταλληλότητα, αν προστεθεί στις καταλληλότητες των προηγούμενων μελών του πληθυσμού, είναι μεγαλύτερη ή ίση του n.

Αν ο αριθμός επιλεγμένων υποψήφιων λύσεων δεν είναι ικανοποιητικός εκτελείται το βήμα 2, διαφορετικά ο αλγόριθμος τερματίζεται.

Το αποτέλεσμα των παραπάνω βημάτων είναι κάθε χρωμόσωμα να έχει πιθανότητα να επιλεγεί για τη διαδικασία της διασταύρωσης ίση με:

$$\frac{\textit{Τιμή Καταλληλότητας}}{\textbf{Συνολική τιμή καταλληλότητας του πληθυσμού}}$$

Με αυτό τον τρόπο θα ευνοηθούν τα χρωμοσώματα με υψηλές καταλληλότητες για να συμμετέχουν στη λειτουργία της διασταύρωσης. Ως παράδειγμα, έστω ότι ένας πληθυσμός των 10 χρωμοσωμάτων με υπολογισμένες τιμές καταλληλότητας και πιθανότητες για επιλογή υπολογισμένες σύμφωνα με τον παραπάνω τύπο. Ακολουθεί ένας πίνακας που περιλαμβάνει τις δύο αυτές παραμέτρους.

Χρωμόσωμα	**Καταλληλότητα**	**Πιθ. επιλογής**	**Χρωμόσωμα**	**Καταλληλότητα**	**Πιθ. επιλογ**
1	0.54	0.12	6	0.81	0.18
2	0.10	0.02	7	0.11	0.02
3	0.70	0.15	8	0.24	0.05
4	0.56	0.12	9	0.16	0.04
5	0.48	0.11	10	0.87	0.19
Ολικό	**2.37**	**0.52**		**2.19**	**0.48**

Πληθυσμός 10 χρωμοσωμάτων

Όπως παρατηρείται στο παραπάνω πίνακα το χρωμόσωμα 10, με τιμή καταλληλότητας 0.87, είναι το πιο «υγιή» χρωμόσωμα στο πληθυσμό. Συγχρόνως το χρωμόσωμα αυτό έχει τη μεγαλύτερη πιθανότητα επιλογής για διασταύρωση με τιμή 0.19. Στην περίπτωση που έπρεπε να επιλεγούν δέκα από αυτά για διασταύρωση, τότε το χρωμόσωμα 10 μπορεί να επιλεγόταν 2 φορές ενώ το χρωμόσωμα 5 μπορεί μία φορά. Επομένως το καθένα απο αυτά μπορεί να επιλεγεί πάνω από μία φορά, ενώ ο αριθμός των φορών που επιλέγονται καθορίζεται από τη πιθανότητα επιλογής, η οποία βέβαια υπολογίζεται από την τιμή καταλληλότητας. Αξίζει να αναφερθεί, ότι τα χρωμοσώματα 2 και 7 μπορούν επίσης να επιλεχθούν για ‘ζευγάρωμα’ και να μεταδώσουν σε επόμενες γενιές τα γονίδια τους και ας έχουν σχετικά χαμηλές τιμές καταλληλότητας. Εάν οι απόγονοι που παράγονται είναι χαμηλής ποιότητας, τότε θα έχουν χαμηλή καταλληλότητα και είναι πιθανόν να παραμείνουν αζευγάρωτοι στις επόμενες γενεές. Καθώς οι επόμενοι καινούργιοι πληθυσμοί περνάνε, τα γονίδια τα οποία μεταδίδονται από γενεά σε γενεά είναι εκείνα που κατα μία έννοια κάνουν τα χρωμοσώματα πιο υγιή. Έτσι όλο και περισσότερα χρωμοσώματα παράγονται με μεγαλύτερη τιμή καταλληλότητας και επομένως η μέση καταλληλότητα του πληθυσμού σταθερά αυξάνεται. Αυτή η συνεχώς αυξανόμενη καταλληλότητα σε κάποιο σημείο ισορροπεί, αλλά οι γενετικοί αλγόριθμοι βρίσκουν μία ικανοποιητική λύση πριν αυτό συμβεί.

Πληθυσμός (Population) και Γονίδια (genes)

Στους γενετικούς αλγόριθμους, τα στοιχεία στο πεδίο ορισμού αναπαρίσταται με συμβολοσειρές (strings), ενώ κάθε στοιχείο της συμβολοσειράς ονομάζεται άτομο (individual) ή γονίδιο (gene). Βεβαίως, όπως αναφέρθηκε παραπάνω, κάθε string αναφέρεται και σαν χρωμόσωμα, δανειζόμενη την ορολογία από τη βιολογία. Οι κυριότεροι τρόποι κωδικοποίησης είναι η δυαδική κωδικοποιήση, όπου κάθε χρωμόσωμα αποτελείται από μία σειρά ψηφίων 0 ή 1, και η κωδικοποιήση δένδρου, όπου το χρωμόσωμα είναι ένα δέντρο. Ως κωδικοποιήση ονομάζουμε το τρόπο με τον οποίο κάθε χρωμόσωμα του πληθυσμού θα εμπεριέχει πληροφορία για τη λύση που περιγράφει. Κάθε συμβολοσειρά έχει κάποια συγκεκριμένη τιμή καταλληλότητας όπου είναι η τιμή που επιστρέφει η συνάρτηση καταλληλότητας για αυτήν. Για παράδειγμα, εάν η συνάρτηση καταλληλότητας ήταν αυτή στο γράφημα 1 παραπάνω και '01000000' (6410) ήταν το string, τότε η τιμή καταλληλότητας θα ήταν το 0 εφόσον f(64)=0. Τέλος, σειρές απο συμβολοσειρές ή γονίδια ονομάζονται «πληθυσμοί» στους γενετικούς αλγόριθμους. Κάθε ζευγάρι του πληθυσμού ζευγαρώνει (mates) προσφέρωντας δύο απογόνους (offsprings), επιδιώκοντας με αυτόν τον τρόπο τη δημιουργία ενός νέου πληθυσμού ο οποίος αποτελείται από τους καινούργιους απογόνους και αποτελεί βελτίωση του προηγούμενου πληθυσμού. Όπως και η εξελιγκτική διαδικασία στη φύση, ο πληθυσμός συνήθως παραμένει σταθερός, χάρη στον αποκλεισμό των πιο αδύναμων λύσεων που παράγονται (weaker child solutions), στην αναπαραγωγή ή τη μετάλλαξη των μελών του πληθυσμού της προηγούμενης γενιάς και στο γενικό κανόνα της επιβίωσης του υγιέστερου (survival of the fittest).

Ακολουθεί παράδειγμα πληθυσμού (population) χρωμοσωμάτων.

Υποψήφιες λύσεις	String(σε bits)	Τιμή καταλληλότητας
Α	000110010110	0,8
Β	111010101100	0,6
Γ	001110101001	0,7
Δ	111011011100	0,5

Πίνακας 2.3 : Παράδειγμα πληθυσμού

Διασταύρωση (Cross-over)

Εκτός από τη συνάρτηση καταλληλότητας και τη κωδικοποίηση συμβολοσειράς, οι γενετικοί αλγόριθμοι χρησιμοποιούν και δύο ακόμα λειτουργίες που ο συνδυασμός τους αποτελεί τεχνική αναπαραγωγής. Οι πολύπλοκες αυτές διαδικασίες είναι η διασταύρωση (cross-over) [MIT96] και η μετάλλαξη (mutation). Αναπαραγωγή είναι η μέθοδος δημιουργίας δύο απογόνων από δύο γονείς-χρωμοσώματα. Η διασταύρωση (crossover operator) είναι η βασική γενετική λειτουργία των γενετικών αλγορίθμων. Από την απόδοσή της εξαρτάται η απόδοση της ίδιας της μεθόδου. Κατά την διαστάυρωση, επιχειρείται να «ζευγαρώσουν» δύο χρωμοσώματα του πληθυσμού, για αυτό ακριβώς το λόγο τυχαία επιλέγεται ένα σημείο στη συμβολοσειρά, το οποίο είναι ένας αριθμός και ονομάζεται σημείο διαστάυρωσης (cross-point). Στη συνέχεια, κάθε «γονέας» χωρίζεται στα δύο σε αυτό ακριβώς το σημείο, ανταλλάσουν μεταξύ τους τα δεύτερα μισά τους και ξαναενώνονται. Το αποτέλεσμα είναι δύο απόγονοι από καθε ζευγάρι γονέων. Αυτό το είδος διασταύρωσης ονομάζεται διασταύρωση ενός σημείου (single point crossover). Βέβαια, αυτό το είδος είναι αποτελεσματικό όταν έχουμε δυαδική απεικόνιση λύσεων. Λόγω της φύσης ενός προβλήματος όμως, μπορεί να χρειάζεται να χρησιμοποιουθούν πιο πολύπλοκες τεχνικές διασταυρώσεως. Γι’αυτό το λόγο έχουν δημιουργηθεί κάποιες «παραλλαγές» της απλής διασταύρωσης που εξασφαλίζουν το εφικτό των λύσεων (feasibility).

Ακολουθεί ένα παράδειγμα διασταύρωσης ενός σημείου (single point crossover) δύο χρωμοσωμάτων τα οποία επιλέχθηκαν τυχαία για να ζευγαρώσουν και να «γεννήσουν» δύο «παιδιά» (offsprings).

Αρχικά Χρωμοσώματα			
Γονέας Α:	10110000	Γονέας Β:	10011100

Τώρα επιλέγεται τυχαία το σημείο διασταύρωσης και έστω ότι αυτό είναι η τρίτη θέση (cross-point =3) , οπότε κάθε γονέας χωρίζεται στα δύο σε αυτό το σημείο:

Αρχικά Χρωμοσώματα			
Γονέας Α:	101 10000	Γονέας Β:	100 11100

Στη συνέχεια αντιμετατίθονται τα δεύτερα μισά από κάθε γονέα ώστε να δημιουργηθούν δύο απόγονοι:

Απόγονοι	
Απόγονος A: 101 11100	Απόγονος B: 100 10000

Όπως γίνεται αντιληπτό από το παραπάνω παράδειγμα, οι απόγονοι περιέχουν τμήματα και απο τους δύο γονείς, δηλαδή κατά μία έννοια τα παιδιά έχουν εξασφαλίσει να κληρονομήσουν «γονίδια» από τα αρχικά χρωμοσώματα. Στο παραπάνω παράδειγμα και δύο αρχικές συμβολοσειρές ξεκινούν με ‘10’ , ομοίως και οι απογόνοι τους. Είναι πολύ πιθανό ότι αυτό το κύριο πρόθεμα να είναι ένα από τα χαρακτηριστικά το οποίο αυξάνει τη τιμή της καταλληλότητας. Υποθέτωντας ότι το ‘10’ στην αρχή του χρωμοσώματος είναι ένα καλό κληρονομημένο χαρακτηριστικό, τότε η το φαινόμενο της μεταβίβασης αυτής της πληροφορίας στον απόγονο πολυ πιθανόν να του αποδόσει μεγαλύτερη τιμή καταλληλότητας. Γενικεύωντας, η μεταβίβαση γενετικού υλικού μετατρέπει τους γενετικούς αλγόριθμους σε πολύτιμο απόκτημα της βιομηχανικής εκμετάλλευσης, εφόσον προωθούν το γεγονός ότι κάποιο θετικό γνώρισμα κληρονομείται από τους απογόνους, όπου στη περίπτωση αυτή είναι τα αρχικά δύο bits.

Τέλος , από το γράφημα 1 είναι εμφανές ότι το μέγιστο της f(x) βρίσκεται στο σημείο x=224, το οποίο αναπαρίσταται απο τη συμβολοσειρά 11100000. Το συγκεκριμένο χρωμόσωμα είναι σαφώς αρκετά κοντά στον απόγονο Β, αλλά όπως γίνεται αντιληπτό κανένας συνδυασμός των γονέων ή των απογόνων τους, όπως δίνονται στο παραπάνω παράδειγμα, δεν είναι δυνατόν ποτέ να οδηγήσει στη δημιουργία δύο αρχικών bits ‘11’, πόσο μάλλον στη δημιουργία τριών αρχικών συνεχόμενων ‘111’. Το φαινόμενο αυτό κυρίως δικαιολογείται από το γεγονός ότι κανένα από τα χρωμοσώματα δεν περιέχει ‘1’ στη έκτη θέση, οπότε κανένας γόνος δεν είναι πιθανόν να κληρονομήσει αυτό το χαρακτηριστικό γνώρισμα. Επομένως, ο γενετικός αλγόριθμος δε μπορεί να αποφύγει το αρχικό ‘10’ και για αυτό ακριβώς το λόγο δε θα μπορέσει ποτέ να βρει τη μέγιστη τιμή της f(x). Με σκοπό να επιλυθεί αυτό το πρόβλημα και να βρεθεί η βέλτιστη λύση εισάγεται μία δεύτερη λειτουργία, η μετάλλαξη (mutation).

Μετάλλαξη (Mutation)

Η μετάλλαξη στους γενετικούς αλγόριθμους ακολουθεί τη διαδικασία της διασταύρωσης και εκτελείται στους απογόνους της. Είναι ανάλογη με την μετάλλαξη στη φύση από την οποία δανείστηκε και την ορολογία. Στο παράδειγμα που ακολουθεί θα επιλεχθεί τυχαίως η θέση ενός bit και θα αλλαχθεί η τιμή του. Με μία μικρή πιθανότητα, κάθε σύμβολο των bit-strings μπορεί να αλλάξει τιμή. Εάν επιλεχθεί η δεύτερη θέση και αντιστραφεί τότε το χρωμόσωμα θα καταλήξει να έχει ‘11’ για τα δύο πρώτα bit αντί για ‘10’. Σε αυτήν τη περίπτωση είναι για την ακρίβεια ευεργετική εφόσον καταφέρνει να μετακινήσει τη συμβολοσειρά από το τμήμα που δεν είναι το ολικό μέγιστο στο τμήμα με το πραγματικό ολικό μέγιστο. Έτσι από το γράφημα 1 παρατηρείται ότι όλα τα χρωμοσώματα που ξεκινούν με ‘11’ είναι μεγαλύτερα ή ίσα με 19210 εφόσον 110000002=19210. Με αυτόν το τρόπο λοιπόν αποφεύγεται ο αλγόριθμος να παγιδευτεί μέσα σε τοπικά ακρότατα του προβλήματος. Η μετάλλαξη συνήθως εφαρμόζεται πενιχρά αφού έχει πολύ μικρή πιθανότητα να συμβεί. Για παράδειγμα, η διαδικασία της μετάλλαξης μπορεί να εφαρμοστεί με πιθανότητα 0.001 [Mit96]. Βέβαια, στη περίπτωση που γίνει κάποια προσπάθεια να αυξηθεί η πιθανότητα μετάλλαξης (probability of mutation) πάρα πολύ, τότε ο γενετικός αλγόριθμος κλίνει να μετατραπεί σε τυχαία αναζήτηση. Επιπρόσθετα, επειδή η μετάλλαξη καταφέρνει να υπερπηδήσει τα εμπόδια στο πεδίο της συνάρτησης, όπως απεδείχθει παραπάνω, κατ'ουσίαν καταφέρνει να ξεπεράσει την αδύναμη φύση της αναζήτησης Αναρρίχησης Λόφων. Οπότε, η μετάλλαξη χρησιμοποιείται ως μία τεχνική ώστε να αποφύγει ένας γενετικός αλγόριθμος να «κολλήσει» σε ένα τοπικό μέγιστο και συνήθως εφαρμόζεται κατά τη διάρκεια της λειτουργίας της διασταύρωσης σαν ένα «λάθος» στη διαδικασία αντιγραφής, δηλαδή σαν ένα τυπογραφικό λάθος.

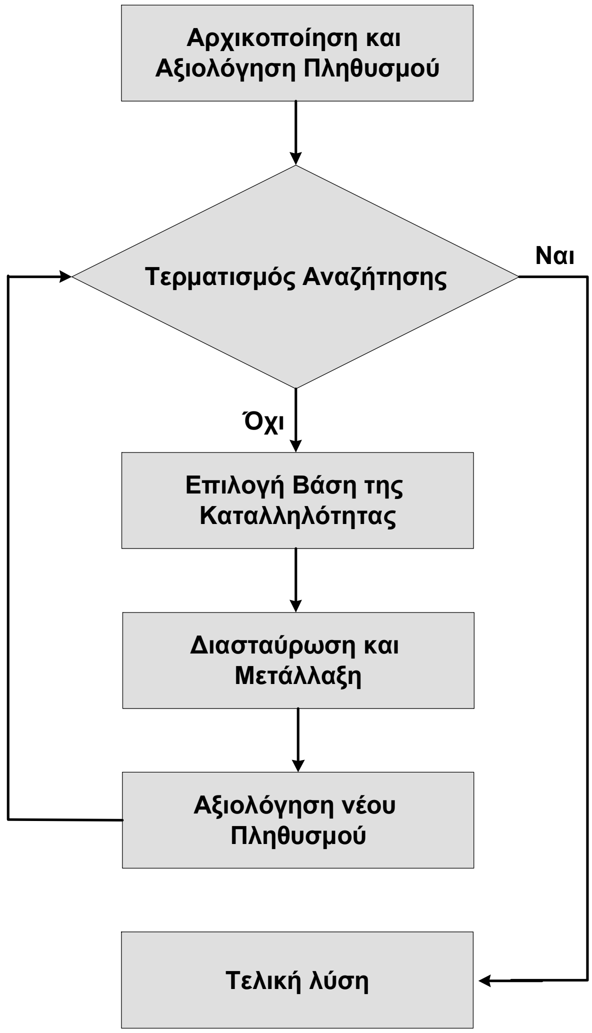
Ακολουθεί παράδειγμα αναπαραγωγής από δύο γονείς-χρωμοσώματα. Τα έντονα τυπωμένα νούμερα στην πέμπτη σειρά είναι αυτά που επηρεάστηκαν από τη μετάλλαξη ενώ σαν σημείο διασταύρωσης επιλέγεται η τρίτη θέση από τα αριστερά.

Υποψήφιοι	**Γονέας Α**	**Γονέας Β**
Γονείς	101 10000	100 11100
	Απόγονος Α	**Απόγονος Β**
Διασταύρωση	101 11100	100 10000
Μετάλλαξη	1**1**1 11100	100 1000**1**

Πίνακας 2.4 : Παράδειγμα Αναπαραγωγής

Ανάλυση της λειτουργίας των γενετικών αλγόριθμων

Αφού ορίστηκαν παραπάνω οι παράμετροι των γενετικών αλγόριθμων αλγόριθμων ακολουθεί ένα διάγραμμα ροής όπου αναπαρίσταται η βασική λειτουργία του:



Εικόνα 2.2 Ο Βασικός Γενετικός Αλγόριθμος

Με τον παρακάτω ψευδοκώδικα γίνεται πιο κατανοητή η βασική λειτουργία των γενετικών αλγορίθμων.

1. Παράγεται αρχικός τυχαίος πληθυσμός χρωμοσωμάτων

2. Καθώς μια ικανοποιητική λύση δεν βρίσκεται στο τρέχων πληθυσμό, κάνε

3. Γίνεται εκτίμηση της καταλληλότητας των χρωμοσωμάτων στο τρέχων πληθυσμό

4. Επιλέγονται τα πιο υγιή από αυτά για διασταύρωση και μετάλλαξη αν είναι απαραίτητη

5. Δημιουργία του νέου βελτιωμένου πληθυσμού από τους απογόνους της διαδικασίας διασταύρωσης.

6. Ανάτρεξε στο 2_0 βήμα.

Ακολουθεί μια λεπτομερής περιγραφή της κάθε γραμμής του ψευδοκώδικα για να σχηματιστεί μία ολική εικόνα των γενετικών αλγόριθμων ως υπολογιστική μέθοδος η οποία επιδιώκει να επιλέξει τις βέλτιστες λύσεις σε ένα πρόβλημα μέσα από ένα σύνολο υποψήφιων λύσεων.

Βήμα (1)

Όπως έχει παραπάνω αναλυθεί η μέθοδος της τυχαίας αναζήτησης επιλέγει ένα τυχαίο σημείο, το εκτιμά, και συνεχίζει μέχρις ότου να βρεθεί μία ικανοποιητική λύση ή μέχρι να τη διακόψει ο προγραμματιστής. Υποθέτοντας ότι η μέθοδος αυτή επιτρέπεται να πραγματοποιήσει 100 επαναλήψεις και τελικά δεν βρήκε το ολικό μέγιστο, τότε υπάρχει ένα σύνολο 100 σημείων στο πεδίο ορισμού τα οποία ελέγχτηκαν με δειγματοληψία, ένα για κάθε επανάληψη. Έστω τώρα ότι αυτά τα σημεία αποτελούν το σύνολο των σημείων το οποίο χρησιμοποιείται ως αρχικός πληθυσμός για το γενετικό αλγόριθμο. Από αυτόν τον αρχικό πληθυσμό χρωμοσωμάτων παίρνουν οι γενετικοί αλγόριθμοι τις περισσότερες από τις εξερευνητικές ιδιότητες. Κατά μία έννοια λοιπόν όταν οι γενετικοί αλγόριθμοι παράγουν τον αρχικό πληθυσμό, απλά ολοκληρώνεται ένα σύνολο τυχαίων αναζητήσεων.

Βήμα (2)

Κάθε χρωμόσωμα στον πληθυσμό αναπαριστά μία λύση στο πρόβλημα το οποίο οι γενετικοί αλγόριθμοι επικαλούνται να επιλύσουν. Αυτό κυρίως εξηγείται εφόσον κάθε χρωμόσωμα απεικονίζει ένα σημείο στο πεδίο ορισμού και αφού κάθε σημείο του πεδίου αυτού είναι λύση τότε όλα τα χρωμοσώματα είναι λύσεις. Βέβαια αυτό δεν σημαίνει ότι πρέπει να παραβλέπουμε τη συνάρτηση καταλληλότητας. Για παράδειγμα, στο γράφημα 1 για τη συνάρτηση f(x) τα χρωμοσώματα 11100000 (22410) και 10000000 (12810) είναι και τα δύο λύσεις αλλά το 11100000 είναι καλύτερη. Ο ορισμός του τι είναι ικανοποιητική λύση εξαρτάται από το πόσο κοντά είναι στη βέλτιστη λύση (optimal solution) και στο πόσο κόντα είναι σε αυτό που αναζητά ο χρήστης των γενετικών αλγορίθμων.

Βήμα (3)

Σε αυτό το σημείο εφαρμόζεται η συνάρτηση καταλληλότητας σε κάθε χρωμόσωμα του τρέχοντος πληθυσμού. Η συνάρτηση καταλληλότητας εξαρτάται από τη συγκεκριμένη γνώση του προβλήματος. Από τη στιγμή όπου η συνάρτηση αυτή καθορίζει τη μορφή που θα αναζητηθεί, αποτελεί κρίσιμο τμήμα των γενετικών αλγορίθμων. Και βέβαια η συνάρτηση αυτή δεν επιβάλεται μόνο σε ακέραιους αλλά και σε χρονοπρογράμματα όπως θα δούμε σε επόμενα κεφάλαια.

Βήμα (4)

Σε αυτό το βήμα εισάγεται η λειτουργία της επιλογής (selection). Όπως έχει ήδη αναφερθεί, όταν στα χρωμοσώματα απονέμονται κάποιες τιμές καταλληλότητας στη συνέχεια αυτά κατατάσσονται σύμφωνα με τις τιμές αυτές. Τα χρωμοσώματα με τις υψηλότερες τιμές καταλληλότητας έχουν και τις μεγαλύτερες πιθανότητες να 'ζευγαρώσουν'. Αυτό το φαινόμενο είναι ανάλογο με τα ζωντανά πλάσματα της φύσης. Τα πιο 'υγιή' από αυτά , είναι αυτά που είναι πιο πιθανό να επιζήσουν και να προσελκύσουν ζευγάρι, και επομένως να μεταδώσουν τα γονίδιά τους. Μια προσομοίωση στους γενετικούς αλγόριθμους πραγματοποιείται με τη τεχνική 'roulette wheel' που περιγράφτηκε παραπάνω.

Βήμα (5)

Καθώς κάποια χρωμοσώματα στο πληθυσμό επιλέχθηκαν για ζευγάρωμα, στη συνέχεια εκτελείται διασταύρωση μεταξύ τους. Όπως ήδη έχει αναφερθεί, η διαδικασία της επιλογής ευνοεί τα χρωμοσώματα εκείνα με υψηλή τιμή καταλληλότητας και η διασταύρωση δεν πρέπει να προκαλεί μεγάλη αλλοίωση στους απόγονους. Μία από τις βασικές ιδέες στους γενετικούς αλγόριθμους είναι αυτή του 'building block' ή 'schemata' τα οποία συμβάλουν στην αύξηση της τιμής καταλληλότητας. Ακολουθεί ανάλυση του μοτίβου αυτού:

- *Το μοτίβο του* ***'building block'*** *ή* ***'schemata'***

Έστω η συμβολοσειρά 111001102 η οποία αντιστοιχεί στο σημείο 23010 και η συνάρτηση f(x) του γραφήματος 1. Το χρωμόσωμα αυτό είναι υψηλής καταλληλότητας και η ιδιότητα του αυτή οφείλεται στα αρχικά του bits 1110, που αποτελεί και ένα συγκεκριμένο μοτίβο, εφόσον η συμβολοσειρά 111000002 αναπαριστά το σημείο 22410 που είναι το ολικό μέγιστο, όπως φαίνεται από τη γραφική παράσταση. Αυτό που απομένει από την συμβολοσειρά αν αφαιρέσουμε το μοτίβο 1110 μπορεί ελάχιστα τα απομακρύνει το χρωμόσωμα από τη βέλτιστη τιμή, αν δηλαδή κάποιο από τα εναπομένοντα ψηφία 0 γίνει 1 η τιμή της καταλληλότητας θα μειωθεί στο ελάχιστο. Προκύπτει, λοιπόν, ότι το μοτίβο των αρχικών 1110 είναι αυτό που κάνει ένα χρωμόσωμα υγιή. Εάν τώρα η διαδικασία της διασταύρωσης σε κάποιο σημείο αλλοιώσει αυτό το μοτίβο, με το να αλλάξει κάποιο ψηφίο, τότε ενδεχομένως θα χαθούν και τα πλεονεκτήματα που αυτή προσφέρει. Ακολουθεί ένα παράδειγμα για περισσότερη κατανόηση:

Έστω τα παρακάτω αρχικά χρωμοσώματα:

Αρχικά Χρωμοσώματα			
Γονέας Α:	11100100	Γονέας Β:	10011000

Τώρα επιλέγεται τυχαία το σημείο διασταύρωσης και έστω ότι αυτό είναι η τρίτη θέση (cross-point =3) , οπότε κάθε γονέας χωρίζεται στα δύο σε αυτό το σημείο:

Αρχικά Χρωμοσώματα	
Γονέας Α: 111 00100	Γονέας Β: 100 11000

Στη συνέχεια αντιμετατίθονται τα δεύτερα μισά από κάθε γονέα ώστε να δημιουργηθούν δύο απόγονοι:

Απόγονοι	
Απόγονος Α: 111 11000	Απόγονος Β: 100 00100

Γίνεται άμεσα αντιληπτό από το παραπάνω παράδειγμα ότι η επιλογή του σημείου διασταυρώσεως επηρεάζει τον 'οικοδομικό λίθο' (building block) του υγιή χρωμοσώματος που στη περίπτωση αυτή είναι το 1110. Αρχικά, ο γονέας Α περιείχε αυτό το μοτίβο αλλά τελικά κανένα από τους απογόνους δεν το έχει κληρονομήσει. Αν όμως είχε επιλεχθεί κάποιο διαφορετικό σημείο διασταυρώσεως τότε θα δημιουργούνταν παιδιά με υψηλότερη καταλληλότητα. Έτσι αν στο παραπάνω παράδειγμα επιλεγόταν ως σημείο διασταυρώσεως η πέμπτη θέση τότε δημιουργούνται οι εξής απόγονοι:

Αρχικά Χρωμοσώματα	
Γονέας Α: 11100 100	Γονέας Β: 10011

000

Στη συνέχεια αντιμετατίθονται τα δεύτερα μισά από κάθε γονέα ώστε να δημιουργηθούν δύο απόγονοι:

Απόγονοι			
Απόγονος Α:	11100 000	Απόγονος Β:	10011 100

Όπου ο απόγονος Α αναπαριστά το σημείο 22410 που είναι το ολικό μέγιστο της συνάρτησης f(x) του γραφήματος 1 ή αλλιώς, η βέλτιστη λύση του προβλήματος.

Τέλος αξίζει να αναφερθεί ότι μέχρις στιγμής έχει περιγραφτεί η διασταύρωση μοναδιαίου σημείου ενώ στη πραγματικότητα υπάρχει και η πολλαπλή διασταύρωση. Σε αυτή την περίπτωση, μπορούν να επιλεχθούν περισσότερα από ένα σημεία για να διαχωριστούν οι αρχικές χρωμοσειρές. Στο παραπάνω παράδειγμα, αν επιλεχθεί μία από τις τρεις πρώτες θέσεις ως σημεία διασταυρώσεως τότε χάνεται το ευεργετικό μοτίβο του αρχικού ‘1110’ το οποίο βέβαια αυξάνει την τιμή της καταλληλότητας. Έτσι υπάρχει μία πιθανότητα 3/7 να εκτελεστεί η διαδικασία της διασταυρώσεως στον ‘οικοδομικό λίθο’ χρησιμοποιώντας τη διασταύρωση μοναδιαίου σημείου. Είναι φανερό ότι επιλέγοντας περισσότερα σημεία διασταυρώσεως, στην περίπτωση δηλαδή της πολλαπλής διασταύρωσης, η πιθανότητα της διασταυρώσεως σε αυτό το αρχικό μοτίβο αυξάνεται. Για αυτό ακριβώς το λόγο, η πολλαπλή διασταύρωση δεν είναι τόσο διαδεδομένη, αφού μειώνεται η πιθανότητα του ‘οικοδομικού λίθου’ να διατηρηθεί από γενεά σε γενεά .

2.3 Clustering - Ομαδοποίηση

Ομαδοποίηση δεδομένων (data clustering) είναι μια τεχνική στατιστικής ανάλυσης δεδομένων και χρησιμοποιείται σε πολλούς τομείς όπως: η μηχανική μάθηση, η εξόρυξη δεδομένων, η αναγνώριση προτύπων, η ανάλυση εικόνων και η βιοπληροφορική. Ομαδοποίηση είναι η ταξινόμηση όμοιων αντικειμένων σε διαφορετικές ομάδες ή αλλιώς ο καταμερισμός των δεδομένων σε υποσύνολα (clusters), έτσι ώστε τα δεδομένα να μοιράζονται κοινά χαρακτηριστικά, τα οποία συνήθως σχετίζονται με κάποια μετρική αποστάσεων. (Renata Krystyna Kwatera 1993)

Ένα συχνό δίλημμα που προκύπτει στην ομαδοποίηση είναι ποιος αλγόριθμος είναι ο κατάλληλος για τα διαθέσιμα δεδομένα. Το σίγουρο είναι ότι δεν υπάρχουν «καλοί» και «κακοί» αλγόριθμοι, αλλά η ποιότητα του καθενός μπορεί να κριθεί από τα αποτελέσματα που δίνει σε μια συγκεκριμένη εφαρμογή. Για παράδειγμα, αν το θεμιτό είναι να διεξαχθούν συμπεράσματα για τα ρυθμιστικά μοτίβα των ομάδων, τότε η σύγκριση των μεθόδων ομαδοποίησης μπορεί να γίνει με βάση το P-value των μοτίβων. Παρόμοια, για λειτουργική ταξινόμηση, μπορεί κανείς να συγκρίνει τα p-values που δείχνουν σε ποιες λειτουργικές κατηγορίες είναι εμπλουτισμένες οι ομάδες.

Οι αλγόριθμοι ομαδοποίησης μπορεί να είναι ιεραρχικοί (hierarchical) ή μη ιεραρχικοί (non-hierarchical/partitional). Οι ιεραρχικοί αλγόριθμοι βρίσκουν διαδοχικές ομάδες, χρησιμοποιώντας κάθε φορά ήδη καθιερωμένες ομάδες, ενώ οι μη ιεραρχικοί καθορίζουν τις ομάδες αμέσως. Οι ιεραρχικοί αλγόριθμοι χωρίζονται στους συσσωρευτικούς (agglomerative) και στους διαχωριστικούς (divisive). Οι πρώτοι αντιμετωπίζουν κάθε στοιχείο σαν μια ομάδα από μόνο του και στη συνέχεια συγχωνεύεται σε μεγαλύτερες ομάδες. Οι δεύτεροι ξεκινούν με ολόκληρο το σύνολο και το διασπούν σε μικρότερες ομάδες.

Συν-ομαδοποίηση (co-clustering, two-way clustering, bi-clustering) είναι η ομαδοποίηση, όπου εκτός από τα αντικείμενα ομαδοποιούνται και τα χαρακτηριστικά των αντικειμένων. Αν δηλαδή τα δεδομένα απεικονίζονται σε ένα πίνακα δεδομένων, τότε ομαδοποιούνται και οι γραμμές και οι στήλες.

Μια άλλη σημαντική διάκριση είναι αν η ομαδοποίηση χρησιμοποιεί συμμετρικές ή ασύμμετρες αποστάσεις. Π.χ μια ιδιότητα του Ευκλείδειου χώρου είναι ότι οι αποστάσεις είναι συμμετρικές, κάτι που δεν είναι εφικτό σε όλες τις εφαρμογές.

2.3.1 Clustering Heuristic - Γενική Περιγραφή

Το πρόβλημα κάλυψης συνόλου μπορεί να τεθεί μαθηματικά όπως παρακάτω (Renata Krystyna Kwatera 1993):

Mininize cx

$$Ax>e,$$

$$x \in B_n$$

όπου B το σύνολο {0,1}, A ένας διάδικος πίνακας m x n, e ένας μονοδιάστατος πίνακας m όπου οι τιμές του είναι όλες 1 και c ένας θετικός πραγματικός μονοδιάστατος πίνακας n τιμών ο λεγόμενος πίνακας κόστους.

Θέτουμε τους παρακάτω ορισμούς και παρατηρήσεις:

Δίνοντας μας ένα πρόβλημα κάλυψης όπως το παραπάνω θεωρούμε ότι η στήλη j επικαλύπτει την γραμμή i αν το a_{ij} = 1. Επίσης θεωρούμε τα σύνολα M={1..m} και N={1..n}. Ένα υποσύνολο J $\subseteq$ N είναι σε κάλυψη αν κάθε γραμμή είναι σε κάλυψη με κάποια στήλη j $\in$ J ή ισοδύναμα αν υπάρχει κάποιο εφικτό x έτσι ώστε J={j: x_j=1}. Ένα επικαλυπτόμενο υποσύνολο J λέγεται πρωταρχικό αν δεν υπάρχει κανένα υποσύνολο του J που είναι επικαλυπτόμενο.

Είναι προφανές ότι μία ιδανική λύση του προβλήματος κάλυψης αντιστοιχεί σε ένα πρωταρχικό σύνολο κάλυψης.

Συχνά χρειάζεται μια αρχική επεξεργασία του προβλήματος έτσι ώστε να ελαχιστοποιήσουμε τις γραμμές και στήλες του προβλήματος. Με αυτό τον τρόπο το μέγεθος του προβλήματος προσαρμόζεται καλύτερα στα προβλήματα της καθημερινής ζωής που συνήθως περιέχουν αρκετές στήλες και γραμμές.

Έτσι ένα σύνολο από κανόνες ελαχιστοποίησης εφαρμόζονται. Ένας τρόπος για την παραγωγή τέτοιον κανόνων είναι ο παρακάτω:

Κανόνας: Αν δύο στήλες επικαλύπτουν ακριβώς τις ίδιες γραμμές τότε κρατάμε μόνο την στήλη με το μικρότερο κόστος (c)

Από τον κανόνα κάποιος μπορεί να παράγει τον παρακάτω προσεγγιστικό ή ευρεστικό κανόνα:

Ευρετικός Κανόνας: Αν δύο στήλες επικαλύπτουν κατά προσέγγιση την ίδια γραμμή τότε κρατάμε μόνο την στήλη με το μικρότερο κόστος. Το κατά προσέγγιση στον κανόνα αυτό σημαίνει να προσεγγίζει καλύτερα κάποια ακρίβεια (με χρήση ενός τύπου)

Ο παραπάνω ευρετικός κανόνας οδηγεί σε μία γενική τάξη από ευρετικών κανόνων για το πρόβλημα κάλυψης.

2.3.2 Clustering Heuristic στο πρόβλημα κάλυψης συνόλου

Ευρετική Ομαδοποίηση (Renata Krystyna Kwatera 1993)μπορούμε να πούμε ότι είναι τα παρακάτω.

Χωρίζουμε ένα σύνολο από στήλες σε ομάδες (clusters) οι οποίες ακολουθούν μια κοινή συμπεριφορά (π.χ. επικαλύπτουν τις ίδιες γραμμές).

Επιλέγουμε από κάθε ομάδα την καλύτερη στήλη (π.χ. την στήλη με το μικρότερο κόστος).

Μπορούμε λοιπόν αν περιγράψουμε έναν αλγόριθμο ευρετικής ομαδοποίηση όπως παρακάτω:

Σε ένα σύνολο J από όλες τις επιλεγμένες στήλες σαν επικαλυπτόμενες παράγετε ένα πρωταρχικό σύνολο J*. Αν παραχθεί ένα σύνολο J* σταματάμε αλλιώς αλλάζουμε τον αρχικό διαχωρισμό (αυξάνουμε το πλήθος των ομάδων) και επαναλαμβάνουμε την διαδικασία.

Με στόχο να ορίσουμε ένα συγκεκριμένο ευρετικό τρόπο ομαδοποίησης κάποιος πρέπει να ορίσει:

1. Μία μέθοδο για διαχωρισμό των στηλών σε ομογενείς ομάδες

Για αυτό τον σκοπό μια ποικιλία από μεθόδους ομαδοποίησης είναι διαθέσιμες στην βιβλιογραφία (Anderberg, Hartigan, Gordon, Mulvey , Crowder, Sahni, Gonzales). Αυτές οι μέθοδοι προσφέρουν μια σειρά επιλογών από τρόπους ταξινόμησης , κριτήρια ομαδοποίησης , μεθόδους ομαδοποίησης.

2. Ένα κανόνα φ για μετασχηματισμό ενός διαχωρισμού P σε ένα άλλο διαχωρισμό P΄.

Γενικά ένας νέος διαχωρισμός P΄ πρέπει να παράγει ένα καλύτερο επικαλυπτόμενο σύνολο J από το παλιότερο. Αυτή η απαίτηση συναντάτε αν για παράδειγμα ο διαχωρισμός P΄ είναι ένας «καθαρισμός» του συνόλου P (π.χ. κάθε τάξη του P' είναι υποσύνολο από κάποια ομάδα του P). Έτσι κάθε ιεραρχικός αλγόριθμος ομαδοποίησης μπορεί να χρησιμοποιηθεί σαν οδηγός για ευρετική ομαδοποίηση για το πρόγραμμα κάλυψης συνόλου.

3. Ένα κανόνα σ για επιλογή της καλύτερης στήλης μιας ομάδας

Τυπικά σ είναι μια απεικόνιση 2^N σε N έτσι ώστε σ(S) $\in$ S για όλα τα S $\subseteq$N. Κάθε τέτοια απεικόνιση θα λέγεται συλλέκτης. Συνήθως, το σ διαλέγεται έτσι ώστε

$$s\,(s) = s \Rightarrow \; f_x = \min_{j \in S} f_j \,,$$

με $f_j = f(c_j, | N_j |)$, όπου $N_j = \{ i, a_{ij} = 1\}$ και f κάποια απλή συνάρτηση του cj με |Nj|.

Για παράδειγμα

$f_j = c_j$ (επιλογή ελάχιστου κόστους)

$f_j = c_j / |N_j|$ (επιλογή Chvatal's)

Η απεικόνιση σ πρέπει να ενσωματώνει και ένα σπάσιμο δεσμών κριτήριο. Για παράδειγμα, κάποιος μπορεί να ορίσει σ(S) σε ένα δείκτη του οποίου το $(c_j, m-|N_j|,j)$ είναι αλφαριθμητικά μικρότερο.

Έτσι η προβολή της προηγούμενης περιγραφής μπορεί να μορφοποιηθεί και να οριστεί σαν ευρετική ομαδοποίηση τεσσάρων κανόνων H=(P,P0,φ,σ) όπου:

Π το σύνολο των διαχωρισμών του N={1…n}

P0 ∈ Π ο αρχικός διαχωρισμός

φ Μια απεικόνιση από το P στον εαυτό του. Με στόχο να διασφαλίσουμε ότι η ευρετική ομαδοποίηση μας θα σταματά μετά από ένα πεπερασμένο πλήθος από επαναλήψεις χρειάζεται για κάποια k επανάληψη φ_k, $\varphi_k(P0)$ είναι ο επιθυμητός διαχωρισμός, π.χ. ο διαχωρισμός με όλες τις ομάδες μοναδικά ομαδοποιημένες.

σ 2^N -> N ο επιλογέας

Ορίζουμε ένα δοσμένο διαχωρισμό P={ C_1,…., Cp } ∈ Π.

$J_\sigma(P) = \{ \sigma(C_1),\ldots\ldots, \sigma(Cp)\}$

Μία τυποποιημένη περιγραφή του αλγόριθμου που ακολουθεί την λογική της Ευρετικής Ομαδοποίησης H είναι η παρακάτω:

```
Begin
        P=P0
        Repeat
                J:=Jσ(P)
                If J is not a cover then P:=φ(P)
        Until J is a cover
        Extract from J a prime cover J*
        Output J*
End
```

2.3.3 Single Clustering Heuristic

Για κάθε δύο στήλες a_i και a_j του πίνακα A ορίζουμε την μη ομοιότητα τους σαν

$$d_{ij} = \sum_{k=1}^{m} | a_{ki} - a_{kj} |$$

Δηλαδή την απόσταση Hamming μεταξύ ai και aj.

Ο αλγόριθμος Single Linkage είναι ένας ιεραρχικός συσσωρευτικός αλγόριθμος ομαδοποίησης ο οποίος ξεκινά από το διαχωρισμό Pn, γεννά μια σειρά διαχωρισμών $\{P_n, P_{n-1}, P_{n-2}, \ldots P_1\}$ από διαχωρισμούς του N. Για κάθε t = n, n-1 ,….,2 ο διαχωρισμός P_{t-1} προκύπτει από τον διαχωρισμό P_t.

Αρχικά βρίσκουμε ένα ζευγάρι (i,j) έτσι ώστε

d_{ij}=min { d_{rs}: όπου r και s ανήκουν σε διαφορετικές ομάδες στον διαχωρισμό P_t}

Τότε οι δύο ομάδες που περιλαμβάνουν τα i,j αντίστοιχα ενσωματώνονται σε μία ομάδα για τον επόμενο διαχωρισμό P_{t-1}.

Ορίζουμε ευρετική ομαδοποίηση Η={Π,P0,φ,σ} παίρνοντας σαν $Π=\{P_1, P_2,\ldots,P_n\}$ και P0=P1 , φ(Pt-1)=Pt με t=2…n και σ τον επιλογέα μικρότερου κόστους.

Οι Gower και Ross έδειξαν ότι ο Single Linked αλγόριθμος μπορεί να υλοποιηθεί μέσω του υπολογισμού του ελάχιστου δέντρου σε ένα γράφο Kn

Έτσι μπορούμε να περιγράψουμε τον αλγόριθμο όπως παρακάτω. Δίνουμε ένα γράφο Kn. Υπολογίζουμε με βάση τη dij Hamming απόσταση και βρίσκουμε το ελάχιστο δέντρο που προκύπτει από το Kn , έστω T.

Θέτουμε F το δάσος που προκύπτει από όλα τα υποδέντρα που προκύπτουν από το T. Για όλα τα δέντρα που προκύπτουν από το F παίρνουμε τις έστω οι ρίζες r1..rt. Για κάθε κόμβο i του Ft υπολογίζουμε το γ(t) που είναι το ελάχιστο κόστος κάθε κόμβου i . Τότε το γ(rh) είναι το ελάχιστο κόστος του rh.

Ο αλγόριθμος αναλυτικά του Single Linkege είναι ο παρακάτω:

Begin

Assign to each edge (i,j) of Kn a length d_{ij} according Hamming Distance

Find a minimum spanning tree T of K_n,

F:=T

Root T an arbitay vertex r1

For each vertex i of T do

    Let p(i) be a cheapest descendant of i in T

    $\gamma(i) := c_p(i)$

end for

t:=1

J:={ p(r1)}

While J is not a cover do

t:=t+1

    let(i,j) be an edge of F such dij is maximum

    let T_h be the connected component of F containing (i, j)

    let r_h be the root of T_h

    F:= F – T_h

    J:= J – {p(rh)}

    Cut Th into two trees T’ ∍ I, T’’ ∍ j by removing edge (i,j)

    T_h:=T’ ; Tt=T’’;

Make j the root r_t of T_t

F:=f $\cup$Th $\cup$Tt

for each u along the backpath from i to rh, update γ(u), p(u)

end while

extract from J a prime cover J*

output J*

end

Παραλλαγές του παραπάνω αλγόριθμου μπορούν να προκύψουν αν για κάθε επανάληψη t μπορεί να αφαιρέσει (αντί για την μεγαλύτερη κορυφή του Ft) μια κορυφή (i,j) όπου

(α) ο λόγος dij/γ(j) να είναι μέγιστος ή

(β) ο λόγος dij/cj να είναι μέγιστος

Συνήθως η περίπτωση (β) δίνει πιο καλά αποτελέσματα

Ο χρόνος εκτέλεσης του παραπάνω αλγόριθμου είναι O(α m n2) όπου α είναι ο λόγος των μη μηδενικών στοιχείων του A σε σχέση με το πλήθος των στοιχείων του.

2.3.4 Leader Heuristic

Θεωρούμε το σύνολο N με στοιχεία αριθμημένα από 1 μέχρι n με τυχαίο τρόπο. Αρχικά το στοιχείο 1 αντιστοιχείται στην πρώτη ομάδα C1 και γίνεται “Leader” το C1.

Θεωρούμε δ>0 σαν ένα κεντρικό κάτω όριο. Υποθέτουμε ότι όταν το στοιχείο j (1<j<=n) βρεθεί υπάρχουν ακόμα k ομάδες C1… Ck. Αν υπάρχει κάποιο Ch 1<h<k τέτοιο ώστε η ανομοιότητα μεταξύ του j και του Leader Ch να μην υπερβαίνει το δ τότε το j αντιστοιχείται σαν το ελάχιστο αριθμημένο στο Ch, αλλιώς τοποθετείται μέσα σε μία νέα ομάδα Ck+1 και γίνεται Leader to Ck+1. Σταματάμε όταν όλα τα στοιχεία του N εξεταστούν.

Θεωρώντας dij είναι ο αριθμός των επικαλυπτώμενων στηλών για κάθε στήλη i ή στήλη j ή και των δύο

$$d_{ij} = \sum_{k=1}^{m} (a_{ri} + a_{rj} - a_{ri}a_{rj})$$

μπορούμε να περιγράψουμε τον αλγόριθμο όπως παρακάτω:

Begin

Sort N so that $c_1/|N_1| >= c_2/|N_2| >= \ldots >= c_n/|N_n|$

J= ∅

δ := α m

while J is not a cover do

    δ:= |δ/2|

```
    k := 1
    leader(1) := 1
    for each j:= 1 to n do
        member := false;
        h:=1
        repeat
            if (dj,leader(h))<= δ then member:= true else h:=h+1
        until h>k or member =true
        last(h)=j
        if h>k then k:=k+1 , leader(k):=j
    end for
    J:= { last(1), …., last(k)}
End While
Extract from J a prime cover J*
Output J*
End
```

Ο χρόνος της χειρότερης περίπτωσης του LEADER είναι $O(\alpha\ m\ n_2 \log2(\alpha\ m))$

Κεφάλαιο 3. Ένας ευρετικός αλγόριθμος

3.1 Ευρετικός αλγόριθμος ομαδοποίησης για την κάλυψη συνόλου.

Όπως είδαμε και στην παράγραφο για 2... για τους ευρεστικούς αλγορίθμους ομαδοποίησης (clustering heuristics), υπάρχει μια συγκεκριμένη διαδικασία που αφορά την δόμηση ενός τέτοιου αλγορίθμου (βλέπει παράγραφο 2.1.2 – άλλαξε το νούμερο βάσει της παρατήρησης 8). Ως πυρήνα του ευρεστικού αλγορίθμου ομαδοποίησης θα χρησιμοποιήσουμε τον αλγόριθμο ομαδοποίησης k-modes.

Ο αλγόριθμος k-modes (Huang, 1998; Μαστρογιάννης, 2009) επεκτείνει την χρήση του κλασσικού αλγορίθμου k-means (MacQueen, 1967) για την διαχείριση αριθμητικών δεδομένων, στον τομέα των κατηγορικών δεδομένων. Συγκεκριμένα, εισάγει ένα νέο μέτρο ανομοιότητας, αντικαθιστώντας τα αριθμητικά κέντρα (means) με αντίστοιχα κατηγορικά (modes). Ειδικότερα, αν x, y είναι 2 κατηγορικά αντικείμενα που περιγράφονται από n χαρακτηριστικά, και n_{x_j}, n_{y_j} είναι το πλήθος των αντικειμένων της βάσης δεδομένων που περιλαμβάνουν τις τιμές x_j, y_j στο j χαρακτηριστικό τους, το μέτρο ανομοιότητας του k-modes τυπικά ορίζεται βάσει του ακόλουθου τύπου (Μαστρογιάννης, 2009):

$$Dh(x,y) = \sum_{j=1}^{n} \frac{(n_{x_j} + n_{y_j})}{(n_{x_j} \cdot n_{y_j})} d(x_j, y_j)$$

(3.1)

όπου $d(x_j, y_j) = \begin{cases} 0, \text{ αν } x_j = y_j \\ \\ 1, \text{ αν } x_j \neq y_j \end{cases}$.

Ταυτόχρονα, ο αλγόριθμος χρησιμοποιεί μια τεχνική βελτιστοποίησης για την συνεχή ανανέωση των κέντρων στην διάρκεια της επαναληπτικής διαδικασίας ομαδοποίησης, η οποία βασίζεται στην συχνότητα εμφάνισης των τιμών των χαρακτηριστικών στο σύνολο των δεδομένων του προβλήματος. Τα βήματα του αλγορίθμου συνοψίζονται ως ακολούθως:

1. Επιλογή των *k* αρχικών κέντρων (modes), ένα για κάθε ομάδα.

2. Ανάθεση κάθε αντικειμένου στην κατάλληλη ομάδα με βάση το μέτρο ανομοιότητας *Dh*. Μετά από κάθε ανάθεση, το κέντρο της παραπάνω ομάδας επανυπολογίζεται.

3. Όταν όλα τα αντικείμενα της βάσης έχουν ανατεθεί σε μια ομάδα, η ομοιότητα/ανομοιότητα των αντικειμένων σε σχέση με τα νέα κέντρα των ομάδων ελέγχεται ξανά. Αν ένα αντικείμενο βρεθεί να είναι πιο κοντά στο κέντρο μιας άλλης ομάδας παρά σε αυτό της ομάδας που ήδη ανήκει, τότε το αντικείμενο αυτό ανατίθεται στην νέα ομάδα. Στην συνέχεια τα κέντρα των δύο ομάδων (παλιά και νέα ομάδα του αντικειμένου), επανυπολογίζονται.

4. Η φάση 3 επαναλαμβάνεται έως ότου κανένα αντικείμενο δεν αλλάζει ομάδα μετά από ένα πλήρη κύκλο ελέγχου ολόκληρης της βάσης των προς ομαδοποίηση δεδομένων.

Προφανώς στην θέση των αντικειμένων του αλγορίθμου, στην περίπτωσή μας βρίσκονται οι στήλες του συνόλου που χρησιμοποιούμε. Όταν ολοκληρωθεί η διαδικασία των τρεξιμάτων του k-modes θα έχουν προκύψει μια σειρά από ομάδες, κάθε μια εκ των οποίων θα ορίζεται από ένα κέντρο-στήλη. Το σύνολο αυτών των κέντρων ορίζουν μια πιθανή κάλυψη (cover) για το εκάστοτε σύνολο που θέλουμε να καλύψουμε. Για να οριστικοποιήσουμε ωστόσο την λύση του προβλήματός μας, θα πρέπει να αποδείξουμε ότι η παραπάνω πιθανή κάλυψη, είναι η βέλτιστη δυνατή (prime cover).

Το τελευταίο θα γίνει μέσα από την χρήση του ακέραιου προγραμματισμού, στα πλαίσια του γραμμικού προγραμματισμού.

3.2 Ακέραιος προγραμματισμός

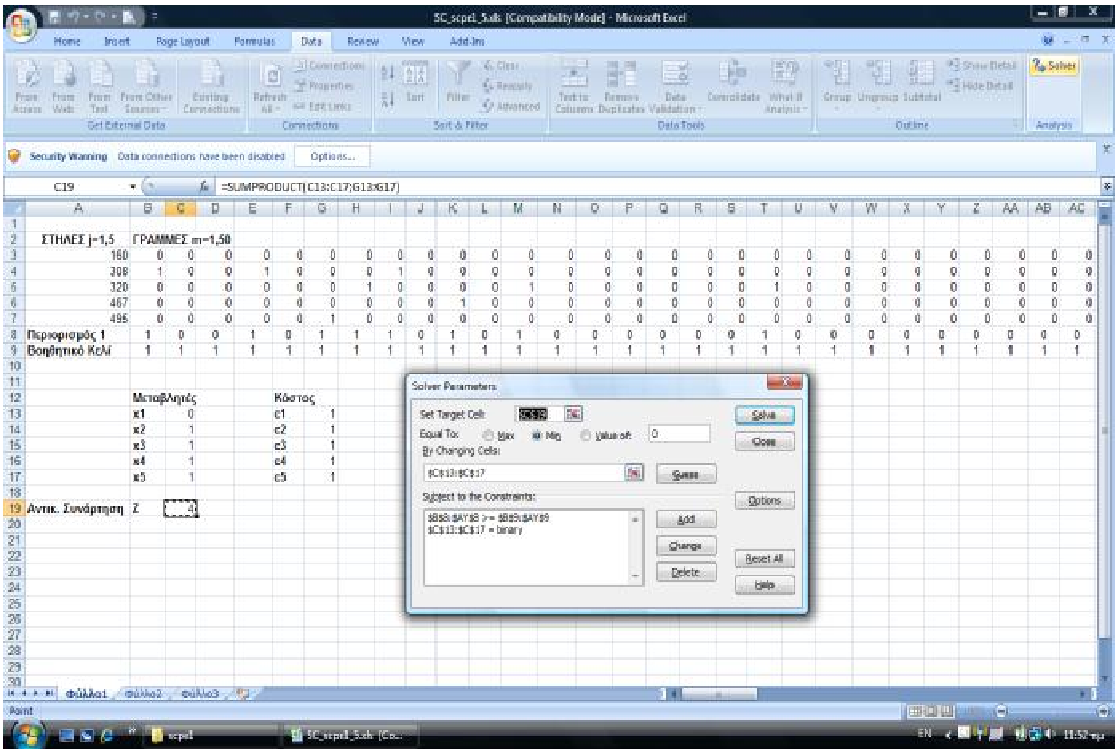
Ο Γραμμικός Προγραμματισμός είναι ουσιαστικά το πρόβλημα βελτιστοποίησης μιας συνάρτησης που υπόκειται σε κάποιους ανισοτικούς περιορισμούς. Η αντικειμενική συνάρτηση πρέπει να είναι πάντα γραμμική. Η ιδιαιτερότητα του ακέραιου προγραμματισμού βρίσκεται στο ότι οι παράμετροι του προβλήματος πρέπει να παίρνουν ακέραιες τιμές (Πραστάκος, 2000). Για παράδειγμα, ζητάμε την ελαχιστοποίηση της ποσότητας Z=7x1 + x2 +5x3, υπό την προϋπόθεση ότι ισχύουν οι κάτωθι περιορισμοί:

$$x_1 - x_2 + 3x_3 \geq 10$$

$$5x_1 + 2x_2 - x_3 \geq 6$$

ενώ, επίσης, απαιτούμε οι 3 μεταβλητές να είναι αφενός μη αρνητικές και αφετέρου ακέραιες.

Εκτός ωστόσο από τα συνήθη αυτά προβλήματα γραμμικού προγραμματισμού, όπου οι μεταβλητές είναι γενικά ακέραιες και η λύση σχετικά απλή, υπάρχει μια ιδιαίτερα σημαντική κατηγορία προβλημάτων αποφάσεων, στα οποία η απόφαση μπορεί να είναι της μορφής 0 ή 1. Η κατηγορία αυτών των προβλημάτων ακέραιου προγραμματισμού ονομάζεται «κατηγορία προβλημάτων 0/1», όπου στην περίπτωση της κάλυψης συνόλο, το 0 σημαίνει «δεν καλύπτει» και το 1 σημαίνει «καλύπτει».



Εικόνα 3.1

Στην εικόνα 3.1 φαίνεται η γενική μορφή του μοντέλου του ακέραιου προγραμματισμού. Ειδικότερα:

- Οι στήλες 160, 308, 320, 467, 495 (στήλη A του Excel) είναι οι στήλες-κέντρα των ισάριθμων ομάδων στις οποίες ο k-modes χώρισε τα δεδομένα του πρώτου συνόλου.
- Οι μεταβλητές x1, x2, x3, x4, x5 πρέπει να είναι ακέραιες.

Στην εικόνα 3.1 φαίνεται ένα ενδεικτικό μοντέλο εφαρμογής του ακέραιου προγραμματισμού, μέσω του Excel Solver. Παρατηρήστε ότι στην στήλη A, φαίνονται τα 5 κέντρα-στήλες (στήλη 160, στήλη 308, στήλη 320, στήλη 467, στήλη 495) από τις ισάριθμες ομάδες στις οποίες χώρισε τα δεδομένα του προβλήματος ο αλγόριθμος k-modes. Στο κελί C19 βρίσκεται η αντικειμενική συνάρτηση (βλέπε ορισμό κάλυψης συνόλο στο κεφάλαιο 1).

Από όλα τα τρεξίματα του αλγορίθμου k-modes, όπως αυτά φαίνονται παρακάτω, θα κρατήσουμε εκείνα που, μέσω του ακέραιου προγραμματισμού, θα μας δώσουν την βέλτιστη κάλυψη.

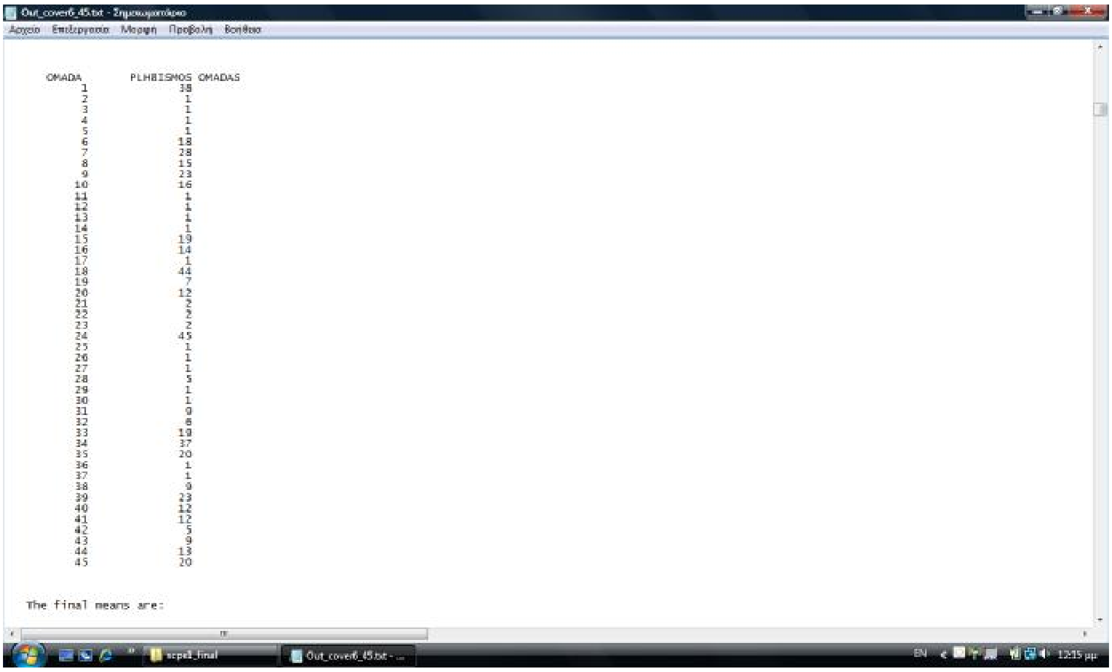
3.3. Εφαρμογή

Εφαρμόσαμε τον k-modes σε 2 σύνολα..

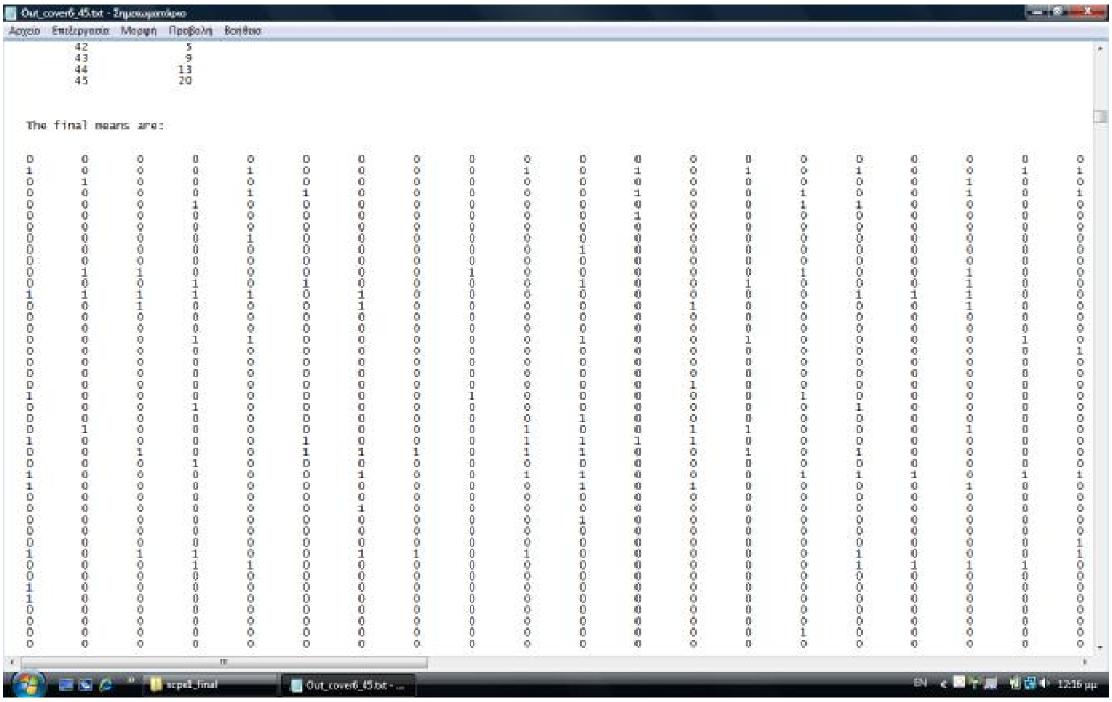
Τα σύνολα αυτά περιέχουν 500 σημεία με 50 διαφορετικές συντεταγμένες με τιμές 0,1 για κάθε σημείο.

Για να μπορέσουμε να τρέξουμε τα σύνολα αυτά, χρειάστηκε να γίνουν κάποιες τροποποιήσεις στην αρχική τους μορφή. Συγκεκριμένα, κατασκευάστηκε κώδικας VBA για να μπορέσουμε να περάσουμε τα στοιχεία των συνόλων σε αρχείο Excel υπό την μορφή γραμμών-στηλών με 0 ή 1 (Βλέπε Παράρτημα Β).

Επειδή, στον αλγόριθμο k-modes, το πλήθος των k ομάδων ορίζεται από την χρήστη, έγιναν μια σειρά από τρεξίματα, ορίζοντας το k στις τιμές 40, 45, 50, 55, 60, 65, 70, 75, 80. Τα τρεξίματα αυτά έγιναν σε ειδικό λογισμικό το οποίο κατασκευάστηκε από τον επιβλέποντα καθηγητή κ. Νικόλαο Μαστρογιάννη. Ενδεικτικά, στις εικόνες 3.2 και 3.3, και στα πλαίσια των αποτελεσμάτων των τρεξιμάτων, φαίνονται το πλήθος των στηλών του συνόλου που ανήκει σε κάθε ομάδα όταν το k=45, καθώς και τα τελικά κέντρα-στήλες των ισάριθμων ομάδων, αντίστοιχα.



Εικόνα 3.2



Εικόνα 3.3

Η εφαρμογή του ακέραιου προγραμματισμού στο Excel Solver για τα παραπάνω αποτελέσματα της ομαδοποίησης οδήγησε σε μια σειρά από αποτελέσματα. Ειδικότερα:

Στο πρώτο σύνολο του Beasley, βέλτιστη κάλυψη εμφανίζεται όταν το πλήθος των ομάδων, και κατ' επέκταση των κέντρων-στηλών είναι 40 και 45, με τιμή αντικειμενικής συνάρτησης 7 και 6 αντίστοιχα. Σε όλες τις υπόλοιπες περιπτώσεις δεν βρέθηκε βέλτιστη κάλυψη.

Στο δεύτερο σύνολο του Beasley, βέλτιστη κάλυψη εμφανίζεται όταν το πλήθος των ομάδων, και κατ' επέκταση των κέντρων-στηλών είναι 45 και 50, με τιμή αντικειμενικής συνάρτησης 7 και 6 αντίστοιχα. Σε όλες τις υπόλοιπες περιπτώσεις δεν βρέθηκε βέλτιστη κάλυψη.

Σε γενικές γραμμές, φαίνεται ότι ο αλγόριθμος k-modes, ενώ είναι ένας από τους καλύτερους αλγορίθμους ομαδοποίησης, δεν ανταποκρίνεται ιδιαίτερα θετικά στο πρόβλημα της κάλυψης συνόλου. Ενδεχομένως, πιο εκτεταμένα τρεξίματα να ενδυναμώσουν τα παραπάνω αποτελέσματα, ωστόσο είναι προφανές ότι ένας αλγόριθμος ομαδοποίησης, δεν μπορεί να λύσει ικανοποιητικά το πρόβλημα της κάλυψης συνόλων, αν δεν προσαρμοστεί πρώτα κατάλληλα. Για παράδειγμα, θα μπορούσε να χρησιμοποιηθεί ένας διαφορετικός επιλογέας (επιλογέας Chvatal) για τον καθορισμό της κάλυψης ή/και μια ευρεστική διαδικασία για να καθοριστεί αν η παραπάνω κάλυψη είναι βέλτιστη.

Κεφάλαιο 4 – Συμπεράσματα

Το πρόβλημα κάλυψης συνόλου στόχο έχει να βρεθεί ο ελάχιστος αριθμός των κέντρων και οι θέσεις τους στο χώρο, έτσι ώστε για κάθε σημείο να υπάρχει ένα κέντρο μέσα σε μια απόσταση tij μονάδων απόστασης ή χρόνου (σε απόσταση ίση ή μικρότερη με tij μονάδων απόστασης ή χρόνου).

Το πρόβλημα έχει μελετηθεί και εξεταστεί από πολλές σκοπιές όπως την λύση με γενετικούς αλγόριθμους, με μετα-ευρετικές μεθόδους καθώς και με μεθόδους κλασικής ομαδοποίησης όπου αναφέρονται αναλυτικά στο κεφάλαιο 2.

Ένας κλασικός αλγόριθμος για την λύση του προβλήματος είναι ο k-modes όπου και χρησιμοποιήθηκε για την επίλυση σε συγκεκριμένα σετ δεδομένων (Beasley sets).

Ο αλγόριθμος k-modes κάνει χρήση του γραμμικού ακέραιου προγραμματισμού όπου αναφερόμαστε σχετικά στο κεφάλαιο 3.

Ο Γραμμικός Προγραμματισμός είναι ουσιαστικά το πρόβλημα βελτιστοποίησης μιας συνάρτησης που υπόκειται σε κάποιους ανισοτικούς περιορισμούς. Η αντικειμενική συνάρτηση πρέπει να είναι πάντα γραμμική. Η ιδιαιτερότητα του ακέραιου προγραμματισμού βρίσκεται στο ότι οι παράμετροι του προβλήματος πρέπει να παίρνουν ακέραιες τιμές. Τέτοιο πρόβλημα είναι και το πρόβλημα κάλυψης συνόλου και ο αλγόριθμος k-modes χρησιμοποιεί επαναληπτικά γραμμικό ακέραιο προγραμματισμό

Με χρήση VBA και εργαλεία επίλυσης ακέραιου προγραμματισμού όπως το SOLVER μπορούμε να χρησιμοποιήσουμε το MICROSOFT EXCEL από την έκδοση 2002 και μετά για την επίλυση του Set Covering μέσω του k-modes

Η εφαρμογή του με χρήση του και του εργαλείου SOLVER του EXCEL και της VBA έδωσε άμεσα αποτελέσματα και γρήγορα αποτελέσματα.

Παράρτημα Α

Επίλυση Προβλημάτων Γραμμικού Προγραμματισμού με το EXCEL (Το εργαλείο Solver)

Για να εξηγήσουμε το εργαλείο SOLVER θα χρησιμοποιήσουμε ένα απλό παράδειγμα.

Ένας προμηθευτής τροφοδοτεί μια αυτοκινητοβιομηχανία με ανταλλακτικά τα οποία χρησιμοποιούνται στη συνέχεια για την παραγωγή των οχημάτων που διαθέτει η αυτοκινητοβιομηχανία. Ο προμηθευτής μόλις έλαβε μια παραγγελία συνολικής αξίας 750.000 Ευρώ για τρεις τύπους ανταλλακτικών. Τα ανταλλακτικά αυτά ο προμηθευτής μπορεί να τα κατασκευάσει ο ίδιος τοποθετώντας με ένα ειδικό μηχάνημα τα απαραίτητα καλώδια και συναρμολογώντας στη συνέχεια τα ανταλλακτικά. Ο χρόνος καλωδίωσης και ο χρόνος συναρμολόγησης ανά τεμάχια δίνονται στον παρακάτω πίνακα. Εναλλακτικά, ο προμηθευτής μπορεί να αγοράσει τα ανταλλακτικά από άλλους προμηθευτές και στη συνέχεια να τα προωθήσει στην αυτοκινητοβιομηχανία. Το κόστος κατασκευής και αγοράς για κάθε τύπο ανταλλακτικού δίνεται επίσης στον παρακάτω πίνακα.

	Τύπος 1	**Τύπος 2**	**Τύπος 3**
Μέγεθος παραγγελίας (τεμ.)	3000	2000	900
Κόστος κατασκευής (Ευρώ/τεμ)	50	83	130
Κόστος αγοράς (Ευρώ/τεμ)	61	97	145

Χρόνος καλωδίωσης (ώρες/τεμ)	2	1,5	3
Χρόνος συναρμολ. (ώρες/τεμ)	1	2	1

Δεδομένα παραδείγματος

Ο προμηθευτής έχει συνολικά διαθέσιμες 10.000 ώρες στο μηχάνημα καλωδίωσης και 5.000 ώρες στο μηχάνημα συναρμολόγησης ανά εβδομάδα. Πώς θα εκτελέσει την παραγγελία κατά τον καλύτερο δυνατό τρόπο;

Σχεδιασμός μοντέλου στο Excel

Υπάρχουν πολλοί τρόποι με τους οποίους μπορεί να σχεδιάσουμε ένα μοντέλο στο Excel. Μία καλή πρακτική είναι να δίνουμε ονόματα σε κελιά η περιοχές κελιών με βάση το περιεχόμενό τους, ώστε να είναι καλύτερα κατανοητό τι προσπαθούμε να κάνουμε με το συγκεκριμένο μοντέλο. Επιλέξτε τα κελιά που μας αφορούν και στη συνέχεια επιλέξτε από το μενού «Εισαγωγή»→«Όνομα»→«Δημιουργία» (Insert→Name→Create) ώστε να ονομάσετε το κελί B34 με το όνομα «Συνολικό Κόστος (Ευρώ)».

Γενικά, τα στοιχεία που πρέπει να υπάρχουν σε κάθε μοντέλο που σχεδιάζουμε είναι τα ακόλουθα:

Δεδομένα: Όλα τα αριθμητικά δεδομένα που απαιτούνται για τον υπολογισμό της αντικειμενικής συνάρτησης και των περιορισμών πρέπει να εμφανίζονται σε μία περιοχή του φύλλου. Συνήθως επιδιώκουμε να εμφανίζουμε όλα τα δεδομένα επάνω και αριστερά στο φύλλο και να τα σκιάζουμε, ώστε να ξεχωρίζουν.

Μεταβλητές: Οι τιμές των μεταβλητών, που θα δώσουν την απάντηση στο πρόβλημά μας, είναι αποθηκευμένες σε μια περιοχή του φύλλου. Τα κελιά της περιοχής αυτής λέγονται κελιά που αλλάζουν (changing cells). Συνηθίζεται να τα σημειώνουμε με πλαίσιο.

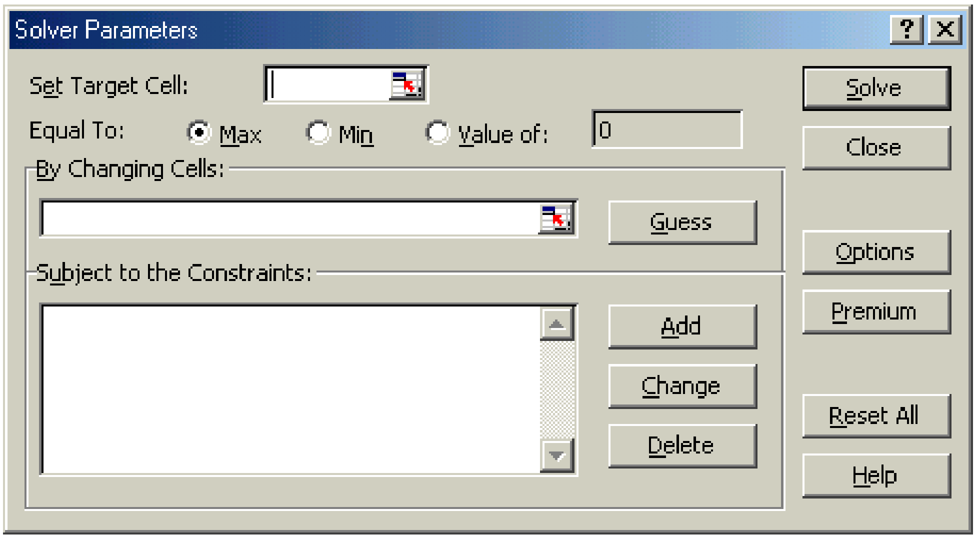
Αντικειμενική συνάρτηση: Η αντικειμενική συνάρτηση υπολογίζεται σε ένα κελί που λέγεται κελί προορισμού (target cell). Συνηθίζεται το κελί αυτό να το σημειώνουμε με διπλό πλαίσιο.

Περιορισμοί: Ο μόνος τρόπος να εμφανίσουμε τους περιορισμούς στο Excel είναι να υπολογίσουμε το αριστερό μέρος κάθε περιορισμού σε ένα κελί και να το συγκρίνουμε με το δεξιό μέρος (σταθερό όρο) που θα είναι αποθηκευμένο σε κάποιο άλλο κελί.

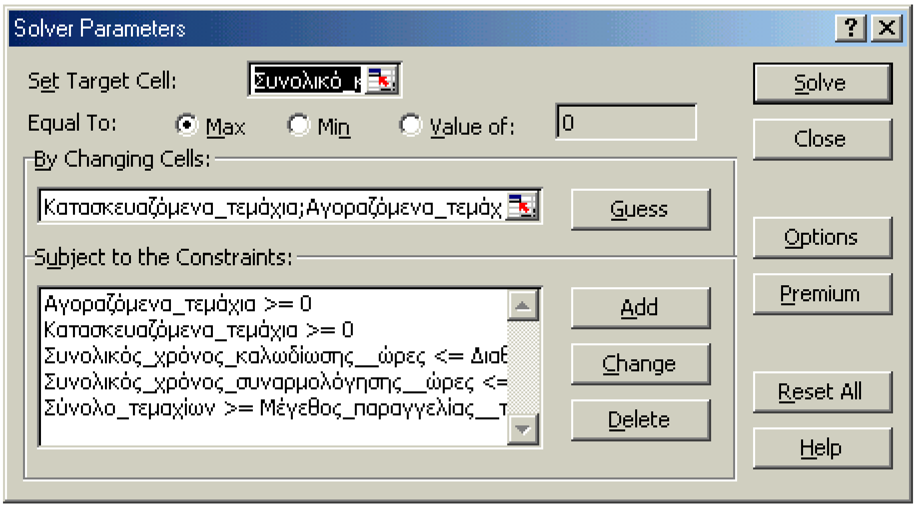
Το επόμενο στάδιο είναι να συνδέσουμε τα περιεχόμενα των κελιών με κατάλληλους τύπους ώστε να υπολογίζεται η τιμή της αντικειμενικής συνάρτησης και το αριστερό μέρος κάθε περιορισμού. Στο σχετικό αρχείο του παραδείγματος φαίνονται οι τύποι που συνδέουν τις τιμές των κελιών μεταξύ τους.

Επίλυση Προβλήματος

Για να λύσουμε το πρόβλημα επιλέξτε «Επίλυση» (Solver) από το μενού «Εργαλεία» (Tools) και θα σας εμφανιστεί το παρακάτω παράθυρο διαλόγου:

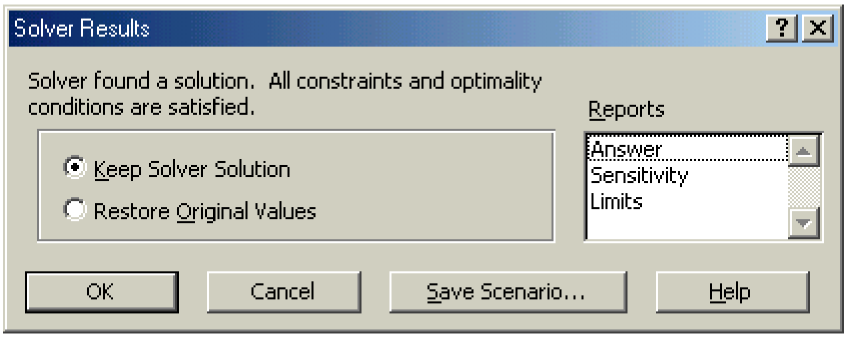


Στο πεδίο «Κελί Προορισμού» (Set Target Cell) επιλέξτε το κελί της αντικειμενική συνάρτησης (Συνολικό κόστος). Στη συνέχεια επιλέξτε «Ελαχιστοποίηση» (Min) και στο πεδίο «Με αλλαγή των κελιών» (By Changing Cells) επιλέξτε τα κελιά των μεταβλητών. Τέλος, στο πεδίο «Με τους περιορισμούς» (Subject to the Constraints) προσθέστε διαδοχικά τους περιορισμούς του προβλήματος. Μην παραλείψετε να προσθέσετε τις συνθήκες μη αρνητικότητας των μεταβλητών. Στο τέλος της διαδικασίας, το παράθυρο του Solver θα είναι όπως φαίνεται παρακάτω:



Στη συνέχεια, επιλέξτε «Επιλογές» (Options), πατήστε την επιλογή «Υπόθεση Γραμμικού Μοντέλου» (Assume Linear Model) και OK.

Τέλος, πατήστε «Επίλυση» (Solve) για να λύσετε το πρόβλημα. και αν όλα έχουν γίνει σωστά, θα πρέπει να εμφανιστεί το παρακάτω παράθυρο διαλόγου



Επιλέγουμε «Απάντηση», πατήστε «ΟΚ» και το πρόγραμμα θα δημιουργήσει ένα νέο φύλλο, στο οποίο θα φαίνεται αναλυτικά η λύση του προβλήματος (Πίνακας 2). Παρατηρούμε ότι για κάθε περιορισμό αναφέρεται η «Κατάστασή» του (Status), δηλαδή κατά πόσο ο περιορισμός είναι αποτελεσματικός (Binding) ή όχι (Non binding). Θυμίζουμε ότι αποτελεσματικοί είναι οι περιορισμοί που ικανοποιούνται ως ισότητες στη βέλτιστη λύση. Για κάθε περιορισμό η στήλη «Απόκλιση» (Slack) δίνει τη διαφορά ανάμεσα στο αριστερό μέρος και το σταθερό όρο. Όπως είναι αναμενόμενο, η απόκλιση για τους αποτελεσματικούς περιορισμούς είναι μηδέν.

Αναφορά Ευαισθησίας

Επιλύουμε ξανά το πρόβλημα και στο παράθυρο διαλόγου που εμφανίζεται μετά την ολοκλήρωση της λύσης επιλέξτε «Ευαισθησία».

Τότε το Excel δημιουργεί ένα νέο φύλλο, ένα τμήμα του οποίου φαίνεται παρακάτω.

Παρατηρούμε ότι στη βέλτιστη λύση ορισμένες μεταβλητές (Ρυθμιζόμενα κελιά) έχουν τιμή μηδέν (αγορά Τύπου 1), ενώ άλλες έχουν μη μηδενική τιμή (παραγωγή Τύπου 1, 2 και 3). Οι μεταβλητές της δεύτερης κατηγορίας, δηλαδή αυτές που έχουν τιμή ανάμεσα στο μηδέν και το άνω όριό τους (εφόσον αυτό υπάρχει) λέγονται βασικές και οι υπόλοιπες μη βασικές.

Στη στήλη «Μειωμένο Κόστος» (Reduced Cost) εμφανίζεται για κάθε μεταβλητή πόσο θα μεταβληθεί η αντικειμενική συνάρτηση αν εμείς αυξήσουμε την τιμή της μεταβλητής κατά μία μονάδα, χωρίς να παραβιάζαμε τις συνθήκες εφικτότητας. Όπως είναι φυσικό, στη βέλτιστη λύση τα μειωμένα κόστη για τις βασικές μεταβλητές θα πρέπει να είναι υποχρεωτικά μηδέν γιατί διαφορετικά θα μπορούσαμε να αυξήσουμε κι άλλο την τιμή τους και να έχουμε καλύτερη λύση. Για τον ίδιο λόγο τα μειωμένα κόστη των μεταβλητών που έχουν τιμή μηδέν είναι αρνητικά (αν έχουμε πρόβλημα μεγιστοποίησης) ή θετικά (αν έχουμε πρόβλημα ελαχιστοποίησης, όπως στο παράδειγμά μας). Πράγματι, αν ο προμηθευτής επιμείνει να αγοράσει ένα τεμάχιο Τύπου 1, το νέο ελάχιστο κόστος θα είναι 453.304 χρηματικές μονάδες.

Στα πραγματικά προβλήματα, εκτός από τις τιμές των μεταβλητών που βελτιστοποιούν την αντικειμενική συνάρτηση, είναι πολύ χρήσιμο να ξέρουμε πώς θα επηρεαστεί η λύση μας αν αλλάξουν τα δεδομένα του προβλήματος.

Παράρτημα Β

Κώδικας Μετατροπής Αρχείων txt που δίνονταν σε αρχεία EXCEL

Για να εξετάσουμε το πρόβλημα Set Cover μας δόθηκαν αρχεία TXT που είχαν την παρακάτω μορφή.

Αρχικά δίνονταν 2 αριθμοί όπου αποτελούσαν των αριθμό των στηλών M και των γραμμών N. Μετά ακολουθούν αντίστοιχα N μονάδες που αντιστοιχούν στα βάρη κάθε παραμέτρου. Στη συνέχεια ακολουθούσε ένας αριθμός K όπου δήλωνε τον αριθμό των μονάδων που υπήρχαν στην πρώτη στήλη. Στην συνέχεια ακολουθούσαν K νούμερα που δήλωναν την γραμμή που θα έπρεπε να αντιστοιχηθεί μονάδα. Το ίδιο ακολουθούσε και για την επόμενη στήλη και την επόμενη μέχρι να τελειώσουν οι M στήλες.

Για να μετατρέψουμε αυτά τα αρχεία σε αρχεία EXCEL δημιουργήσαμε κώδικα VBA ο κώδικάς φαίνεται παρακάτω:

```
Sub readfile()
Dim g As Range
Set g = Range("a1:aa10000")
s = InputBox("give file")
Open s For Input As #1
```

```
i = 1
Input #1, cols
Input #1, lns
For i = 1 To lns
Input #1, c
g(i, 1) = c
Next

For i = 1 To lns
For j = 1 To cols
g(i, j + 1) = 0
Next
Next
k = 1
Do While Not EOF(1)    ' Loop until end of file.
Input #1, cols

   For j = 1 To cols
   Input #1, c
   g(c, k + 1) = 1
   Next
   k = k + 1
```

Loop

Close #1

End Sub

Παράρτημα Γ

Παράδειγμα αρχείου TXT

Δίνεται ένα παράδειγμα αρχείου TXT που μας δόθηκε δίνοντας μερικές από της στήλες του αρχείου (το αρχείο είναι τεράστιο σε όγκο οπότε παρουσιάζεται μόνο ένα μέρος του)

```
50 500
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1

97

1 2 4 7 9 11 12 15 16 20 22 23 24 28 33

34 36 37 38 39 48 51 52 55 59 60 68 73 78 80

82 83 86 91 94 95 96 98 99 111 119 121 122 132 135

139 144 148 150 171 174 175 189 191 192 204 210 214 217 226

232 237 241 243 246 251 254 255 261 263 265 266 268 285 303

314 329 330 350 354 359 366 374 379 384 394 397 402 414 416

422 430 444 453 466 483 492

77

2 11 15 21 26 30 36 37 38 43 47 50 51 56 60

66 70 78 89 91 93 96 97 100 113 119 131 133 151 152

153 154 155 169 170 172 175 178 188 193 210 227 233 260 268

275 279 280 281 286 287 290 292 294 309 317 327 329 339 358

362 364 365 372 374 380 381 383 413 429 430 432 436 444 449

466 474

105

2 3 4 7 9 10 12 15 23 24 25 27 28 34 35

41 42 43 44 45 55 58 59 64 67 69 70 82 83 85

87 90 92 94 100 101 113 118 121 142 143 161 162 163 168

169 174 177 183 190 191 193 200 204 207 208 222 223 227 232

242 243 248 249 250 256 259 265 271 273 278 282 287 290 306

309 313 320 324 331 339 358 359 360 365 369 370 374 376 381

386 389 396 441 443 449 450 451 459 468 469 479 482 491 493

91

2 4 5 6 19 26 37 41 46 49 52 54 61 65 74

75 79 86 91 95 101 102 105 112 116 124 125 126 134 140

141 142 145 146 149 155 156 158 159 160 170 181 182 187 190

199 200 201 203 205 217 221 223 234 239 242 244 247 252 257

266 271 272 273 276 284 288 295 298 303 313 332 341 351 352

363 367 379 384 404 411 416 427 438 446 460 462 479 480 483

492

108

2 5 6 15 16 18 22 31 32 35 39 40 43 50 54

59 62 63 64 69 72 75 77 81 84 87 89 94 95 96

98 106 108 122 126 127 128 131 136 143 152 153 159 160 165

166 168 170 174 176 178 179 180 183 186 187 189 191 201 205

208 211 220 229 232 235 245 246 251 257 260 261 264 269 272

281 282 288 303 304 314 316 319 326 334 338 339 344 353 354

359 370 372 373 377 378 390 404 410 429 439 451 454 461 464

476 484 496

93

3 7 17 18 22 25 27 31 32 35 36 39 43 44 46

48 61 62 75 77 83 98 102 110 112 113 117 122 131 137

138 139 140 143 145 152 153 154 156 159 173 178 190 196 197

200 203 204 208 224 227 230 231 238 240 243 247 270 273 275

276 277 284 285 286 298 306 307 316 319 328 333 347 351 367

372 392 415 419 420 433 435 438 457 458 461 470 475 489 494

495 498 500

115

1 2 3 4 7 9 14 17 24 25 26 27 28 29 31

34 38 41 42 43 45 46 50 56 57 59 61 63 66 69

71 72 76 82 86 88 89 91 93 97 99 101 103 106 109

116 118 121 129 134 139 146 147 152 153 158 166 169 176 177

178 180 183 188 190 194 198 201 216 221 222 232 244 248 251

274 277 280 283 284 286 308 312 320 326 342 344 357 360 364

369 370 375 378 382 383 386 392 393 402 413 415 416 418 431

434 440 451 462 465 477 479 485 493 497

102

3 4 5 11 12 14 15 27 30 31 33 38 42 44 46

49 50 51 52 59 63 68 74 77 79 85 87 93 94 97

109 118 119 121 122 125 126 127 135 143 144 156 169 176 189

191 194 195 197 198 199 202 204 209 211 216 219 221 226 228

232 233 239 241 244 246 251 256 258 281 287 296 306 308 310

320 321 324 334 335 342 345 346 357 365 373 387 389 391 400

404 408 420 431 442 457 471 473 477 487 492 496

92

5 14 17 20 22 25 26 30 33 34 35 44 47 52 53

57 62 67 68 71 73 74 79 89 108 115 116 122 128 130

138 143 146 157 158 162 165 184 193 196 197 198 203 218 219

220 225 233 241 242 246 259 262 283 290 293 297 299 301 312

316 322 323 326 328 334 340 345 349 350 355 357 367 373 380

385 387 394 403 405 407 408 411 412 413 429 434 447 451 462

473 484

92

1 3 4 5 10 16 21 24 26 27 28 31 34 35 37

41 47 48 50 51 60 64 73 79 81 95 100 101 104 108

109 117 120 123 125 126 127 129 134 136 137 141 146 150 154

169 176 182 183 185 191 194 200 202 205 209 222 223 228 231

236 260 263 274 277 286 292 298 302 309 312 313 317 348 351

355 371 375 388 389 390 396 397 399 414 423 467 468 471 474

475 486

Παράρτημα Δ

Παράδειγμα αρχείου EXCEL

Δίνεται ένα παράδειγμα αρχείου EXCEL που μας δόθηκε δίνοντας μερικές από της στήλες του αρχείου (το αρχείο είναι τεράστιο σε όγκο οπότε παρουσιάζεται μόνο ένα μέρος του)

1	1	0	0	0	0	0	1	0
1	1	1	1	1	1	0	1	0
1	0	0	1	0	0	1	1	1
1	1	0	1	1	0	0	1	1
1	0	0	0	1	1	0	0	1
1	0	0	0	1	1	0	0	0
1	1	0	1	0	0	1	1	0
1	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	1	0
1	0	0	1	0	0	0	0	0
1	1	1	0	0	0	0	0	1
1	1	0	1	0	0	0	0	1
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	1
1	1	1	1	0	1	0	0	1
1	1	0	0	0	1	0	0	0
1	0	0	0	0	0	1	1	0
1	0	0	0	0	1	1	0	0
1	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	1	0	0	0	1	1	0	0
1	1	0	1	0	0	0	0	0
1	1	0	1	0	0	0	1	0
1	0	0	1	0	0	1	1	0
1	0	1	0	1	0	0	1	0
1	0	0	1	0	0	1	1	1
1	1	0	1	0	0	0	1	0
1	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	1

1	0	0	0	0	1	1	1	1
1	0	0	0	0	1	1	0	0
1	1	0	0	0	0	0	0	1
1	1	0	1	0	0	0	1	0
1	0	0	1	0	1	1	0	0
1	1	1	0	0	0	1	0	0
1	1	1	0	1	0	0	0	0
1	1	1	0	0	0	0	1	1

Βιβλιογραφία

Church R., ReVelle C. “The Maximal Covering Location Problem”. Papers of the Regional Science Association, 32, 101-118, 1974.

Church R., ReVelle C. “Theoretical and Computational Links Between the P-Median Location Set Covering and the Maximal Covering Location Problem”. Geographical Analysis, 8, 406-415, 1976.

Halfin S. “On Finding the Absolute and Vertex Centers of a Tree with Distances”. Transportation Science, Vol. 8, 75-77, 1974.

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

ReVelle C.S., Eiselt H.A. “Location Analysis: A Synthesis and Survey”. European Journal of Operational Research 165, 1-19, 2005.

Schilling D.A., Jayaraman V., Barkhi R.. “A Review of Covering Problems in Facility Location”. Location Science I, 1, 25-55, 1993.

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

White J. A., Case K.E. “On Covering Problems and the Central Facility Location Problem”. Geographical Analysis 6, 281-293, 1974.
Beraldi Patrizia, Ruszczynski Andrzej, “The probabilistic Set Covering Problem”, Operations Research, 50, 956-967, 2002

Renata Krystyna Kwatera, “Clustering heuristics for set covering”, Annals of Operations Research 43, 295-308, 1993

Melanie Mitchell. An Introduction to Genetic Algorithms. MIT Press, 1996.

Z.Michalevicz, Genetic Algorithms + Data structures = Evolution Programs, Springer, 1995.

Beasley, J.E (1987), An algorithm for set covering problem, European Journal of Operational Research, 31, pp.85-93

Huang, Z. (1998), *Extensions to the k-means Algorithm for Clustering Large Data Sets with categorical Values*, Data Mining and Knowledge Discovery, vol.2, no.3, pp.283-304

Chvatal, V., 1979. A greedy heuristic for the set-covering problem. Mathematics of Operations Research 4, 233–235.

Guanghui Lan, Gail W. DePuy, Gary E. Whitehouse. An effective and simple heuristic for the set covering problem, European Journal of Operational Research 176 (2007) 1387–1403.

Beasley, J.E., 1987. An algorithm for set covering problem. European Journal of Operational Research 31, 85–93.

Μαστρογιάννης Νικόλαος, Διδακτορική Διατριβή «Μεθοδολογικό Πλαίσιο υποστήριξης της Εξόρυξης Γνώσης από Δεδομένα με την χρήση αρχών της Πολυκριτήριας Ανάλυσης Αποφάσεων», Τμήμα Διοίκησης Επιχειρήσεων, Πανεπιστήμιο Πατρών, 2009.

Πραστάκος Γρηγόρης, Διοικητική Επιστήμη – Λήψη Επιχειρηματικών Αποφάσεων στην κοινωνία της πληροφορίας, Εκδόσεις Σταμούλης, 2000.