ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Διπλωματική Εργασία

«Μετάδοση πληροφορίας σε ασύρματο δίκτυο αισθητήρων με ομαδοποιημένους κόμβους και με χρήση διευθύνσεων από κανόνες Golomb»

ΧΡΥΣΟΣ ΕΜ. ΓΡΗΓΟΡΙΟΣ

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Δόλλας Απόστολος, Καθηγητής Π.Κ.(Επιβλέπων)
Πνευματικάτος Διονύσιος, Αναπληρωτής Καθηγητής Π.Κ.
Παπαευσταθίου Ιωάννης, Επίκουρος Καθηγητής Π.Κ.

Χανιά – Ιούνιος 2006

ΠΕΡΙΛΗΨΗ

Τα ασύρματα δίκτυα αισθητήρων (Wireless Sensor Network) είναι μία σχετικά νέα τεχνολογία ασύρματων δικτύων. Τα δίκτυα αυτά, αποτελούνται από ένα μεγάλο αριθμό κόμβων που συλλέγουν πληροφορίες και οι οποίες ασύρματα συλλέγονται σε έναν κεντρικό κόμβο (sink) και στην συνέχεια σε ένα κεντρικό server. Οι κόμβοι του δικτύου έχουν μικρή υπολογιστική ισχύ, περιορισμένη μνήμη και ιδιαίτερα περιορισμένη ενέργεια. Το μεγαλύτερο ποσοστό ενέργειας σε κάθε κόμβο δαπανάται κατά την μετάδοση πακέτων.

Οι κανόνες Golomb είναι ένα σύνολο από ακεραίων, που έχουν την ιδιότητα οι αποστάσεις μεταξύ τους κατά απόλυτη τιμή να είναι μοναδικές. Κάθε ακέραιος αριθμός ενός κανόνα Golomb ονομάζεται “σημάδι”(mark).

Ομαδοποίηση (clustering) δεδομένων είναι μία διαδικασία οργάνωσης των δεδομένων σε ομάδες (clusters), τέτοια ώστε τα δεδομένα κάθε ομάδας να έχουν κοινά χαρακτηριστικά. Η ομαδοποίηση των δεδομένων γίνεται με διάφορους αλγόριθμους και τεχνικές.

Σκοπός της εργασίας, αυτής, είναι η έρευνα για το κατά πόσο μπορεί η χρήση διευθύνσεων από κανόνες Golomb σε ένα ασύρματο δίκτυο αισθητήρων και με την βοήθεια των διαφόρων τεχνικών ομαδοποίησης των κόμβων, να μειώσει την δαπανούμενη ενέργεια. Η μείωση της δαπανούμενης ενέργειας οφείλεται στην μείωση των bits ανά μετάδοση αλλά και στην μείωση της μνήμης που χρησιμοποιεί ο κάθε κόμβος.

ΠΕΡΙΕΧΟΜΕΝΑ

Ο κόσμος του κάθε ανθρώπου
στηρίζεται σε δύο κολώνες :
ότι θυμόμαστε και ότι αγαπάμε...

Αφιερωμένη στην οικογένεια μου

ΕΥΧΑΡΙΣΤΙΕΣ

Πρώτα απ'όλους θα ήθελα να ευχαριστήσω τον κ. Δόλλα, για την υποστήριξη και την πολύτιμη βοήθεια του κατά τη διάρκεια υλοποίησης αυτής της διπλωματικής εργασίας καθώς και για το γεγονός ότι μου έδωσε την ευκαιρία να γίνω μέλος του Εργαστηρίου Μικροεπεξεργαστών και Υλικού του Πολυτεχνείου Κρήτης.

Στην συνέχεια, θέλω να ευχαριστήσω :

Την επιτροπή της διπλωματικής μου εργασίας, κ. Δ. Πνευματικάτο και κ. Ι. Παπαευσταθίου για την συμβολή τους στην εργασία αυτή.

Το κ. Μάρκο Κιμιωνή, μέλος ΕΕΔΙΠ και υπεύθυνος του εργαστηρίου Μικροεπεξεργαστών και Υλικού, για την υποστήριξη του σε τεχνικά ζητήματα.

Τους κ. Ευριπίδη Σωτηριάδη και κ. Κυπριανό Παπαδημητρίου, διδακτορικοί φοιτητές, για τις συμβουλές και τις ιδέες τους σε δύσκολα σημεία της εργασίας.

Όλους τους προπτυχιακούς και μεταπτυχιακούς φοιτητές του εργαστηρίου Μικροεπεξεργαστών και Υλικού για την βοήθεια και την στήριξη τους.

Την οικογένεια μου για την υποστήριξη που μου πρόσφερε όλα αυτά τα πέντε χρόνια.

Τους φίλους μου για όλες τις καλές στιγμές που περάσαμε μαζί και ιδιαίτερα τον κ. Κοζανίτη Χρήστο, για την καλή συνεργασία που είχαμε κατά την διάρκεια των σπουδών μας.

Τέλος, ένα πολύ μεγάλο ευχαριστώ στην Φωτεινή, η οποία βρέθηκε στο πλάι μου στις πολύ δύσκολες στιγμές που πέρασα τον τελευταίο καιρό.

ΚΕΦΑΛΑΙΟ 1^{ο}

Εισαγωγή

Οι Golomb κανόνες είναι ένα μαθηματικό πρόβλημα με το οποίο έχουν ασχοληθεί πολλές ομάδες ερευνητών από το 1952. Ο βασικός στόχος αυτής της διπλωματικής εργασίας ήταν η μελέτη του κατά πόσο οι μαθηματικές ιδιότητες των κανόνων Golomb δημιουργούν καλές ιδιότητες για ομαδοποίηση των αριθμών αυτών και πιθανή χρήση των ιδιοτήτων αυτών για ασύρματη μετάδοση σε ένα δίκτυο αισθητήρων. Από την διπλωματική εργασία του κ. Δημητρομανωλάκη [1], "Analysis of the Golomb Ruler and the Sidon Set Problems, and Determination of Large, Near-Optimal Golomb Rulers", έχει αποδειχθεί ότι **για μεγάλους κανόνες Golomb με αριθμό marks n, το μέγεθος του σχεδόν βέλτιστου (near optimum) κανόνα Golomb με n marks είναι μέχρι n^2 ,** ενώ είναι επίσης γνωστό ότι **για έναν κανόνα Golomb με μέγεθος n τότε το μέγεθος του μεγαλύτερου mark αυτού του κανόνα σε bits είναι 2n.**

Οι Golomb κανόνες [1],[8],[9] είναι ένα σύνολο ακέραιων, οι οποίοι ονομάζονται "σημάδια"(marks), των οποίων οι ανά δύο κατά απόλυτη τιμή διαφορά είναι μοναδική. Οι ακέραιοι αυτοί μπορούν να θεωρηθούν και σαν σημάδια σε έναν κανόνα, που έχει κατασκευαστεί με την ιδιότητα ότι κανένα ζευγάρι από τα σημάδια του δεν μπορούν να μετρήσουν την ίδια απόσταση. Ο ακριβής μαθηματικός ορισμός των κανόνων Golomb είναι ο εξής :

Ένας κανόνας Golomb είναι ένα σύνολο από ακέραιους $A = \{α(1), α(2), α(3), ..., α(v)\}$ όπου $α(1) < α(2) < α(3) < ... < α(v)$ και ισχύει ότι $| α(i) – α(j) |$ είναι ***μοναδική*** *για κάθε ζευγάρι (i, j).*

Οι κανόνες Golomb έχουνε πολλές εφαρμογές, όπως είναι η πληροφορική (κατασκευή κωδίκων για την εύρεση και διόρθωση λαθών) και η αστρονομία (για την συλλογή πληροφοριών σε σχέση με την απόσταση μεταξύ των ουρανίων σωμάτων). Ο πρώτος που ασχολήθηκε με τους αριθμούς αυτούς ήταν ο Babock, το 1953. Αυτός που τους μελέτησε συστηματικά και από τον οποίο προέρχεται το όνομα τους είναι ο καθηγητής Solomon Golomb, το 1960.

Βέλτιστος κανόνας Golomb(optimal Golomb Ruler) [8],[9] με n "σημάδια" είναι ο κανόνας Golomb που έχει το μικρότερο μήκος από οποιοδήποτε άλλο με n "σημάδια". Οι κανόνες Golomb που χρησιμοποιήθηκαν σε αυτή την εργασία είναι οι κανόνες που προκύπτουν από τον αλγόριθμο του Ruzsa. Οι κανόνες αυτοί είναι κανόνες "σχεδόν" βέλτιστοι και όπως αποδείχθηκε από τον κ.Δημητρομανωλάκη [1] όλοι έχουν ως άνω όριο το n^2, όπου n είναι ο αριθμός των marks του κανόνα.

"Ομαδοποίηση" (clustering) [6] ονομάζεται η διαδικασία της οργάνωσης των δεδομένων σε ομάδες "clusters", έτσι ώστε σε κάθε ομάδα να υπάρχουν κάποιες κοινές ιδιότητες. Η "ομαδοποίηση" δεδομένων οδηγεί σε δημιουργία ομάδων των οποίων τα αντικείμενα της κάθε ομάδας παρουσιάζουν κάποια ομοιότητα, ενώ τα αντικείμενα διαφορετικών ομάδων παρουσιάζουν διαφορές. Για την ομαδοποίηση των marks των κανόνων Golomb χρησιμοποιήθηκαν διάφοροι αλγόριθμοι και διάφορες τεχνικές, τα οποία παρουσιάζονται στα Κεφάλαια 2 και 3. Γενικά, η "καλή" ομαδοποίηση δεδομένων δεν είναι εύκολη υπόθεση και εξαρτάται καθαρά από τα κριτήρια που θέτει ο χρήστης. Τα αποτελέσματα των μεθόδων ομαδοποίησης που χρησιμοποιήθηκαν καθώς και τα μαθηματικά κριτήρια με τα οποία μία ομαδοποίηση θεωρήθηκε "καλή" αναφέρονται στα Κεφάλαια 4 και 5.

Μία εφαρμογή, στην οποία θεωρήθηκε ότι θα μπορεί να έχει κάποιο όφελος η χρήση των κανόνων Golomb καθώς και η δημιουργία ομάδων των marks τους είναι τα δίκτυα αισθητήρων [3],[4]. Η κεντρική ιδέα ήταν να χρησιμοποιηθούν οι ακέραιοι αριθμοί ενός κανόνα Golomb ως διευθύνσεις κόμβων σε ένα εικονικό δίκτυο αισθητήρων και στη συνέχεια να γίνει εκμετάλλευση κάποιων ιδιοτήτων τους. Ο βασικός στόχος ήταν να βρεθεί κάποιος τρόπος που η διευθυνσιοδότηση με κανόνες Golomb να υπερτερούν έναντι της διευθυνσιοδότησης με weighted binary αριθμούς. Η σύγκριση των δύο αυτών διαφορετικών διευθυνσιοδοτήσεων έγινε από τη σκοπιά της απόδοσης (μικρότερος αριθμός bits για κάθε μετάδοση) και από την μεριά των δομών του υλικού που χρειάζονται για την υποστήριξη των μεθόδων. Οι συγκρίσεις καθώς και τα αποτελέσματα των συγκρίσεων παρουσιάζονται στο Κεφάλαιο 6.

Τέλος, στο Κεφάλαιο 7 περιέχονται όλα τα συμπεράσματα που προέκυψαν από την έρευνα που έγινε για την συγκεκριμένη διπλωματική εργασία, ενώ παρουσιάζονται και κάποιες ιδέες για συνέχιση αυτής της εργασίας.

ΚΕΦΑΛΑΙΟ 2^{ο}

2.1) Εισαγωγή

Ομαδοποίηση (clustering) ονομάζεται η τοποθέτηση των δεδομένων σε ομάδες με κοινά χαρακτηριστικά. Η ομαδοποίηση, αυτή, σίγουρα οδηγεί σε μείωση της λεπτομέρειας των δεδομένων αλλά επίσης και σε απλοποίηση για παραπάνω επεξεργασία. Οι τεχνικές ομαδοποίησης που χρησιμοποιήθηκαν ανήκουν σε δύο μεγάλες κατηγορίες: a) **hierarchical ομαδοποίηση** [5],[6] και b) **partitional ομαδοποίηση**. Εκτός βέβαια από τις δύο αυτές κατηγορίες clustering χρησιμοποιήθηκε και μία άλλη ευρυστική μέθοδος, η οποία περιλαμβάνει την μετατροπή των δεδομένων σε binary μορφή και στην συνέχεια τον διαχωρισμό των clusters με βάση τις τιμές των bits σε συγκεκριμένες θέσεις .

2.2) Τρόποι ομαδοποίησης

2.2.1) Hierarchical Ομαδοποίηση

Η ιεραρχική ομαδοποίηση γίνεται με δύο διαφορετικές μεθόδους :

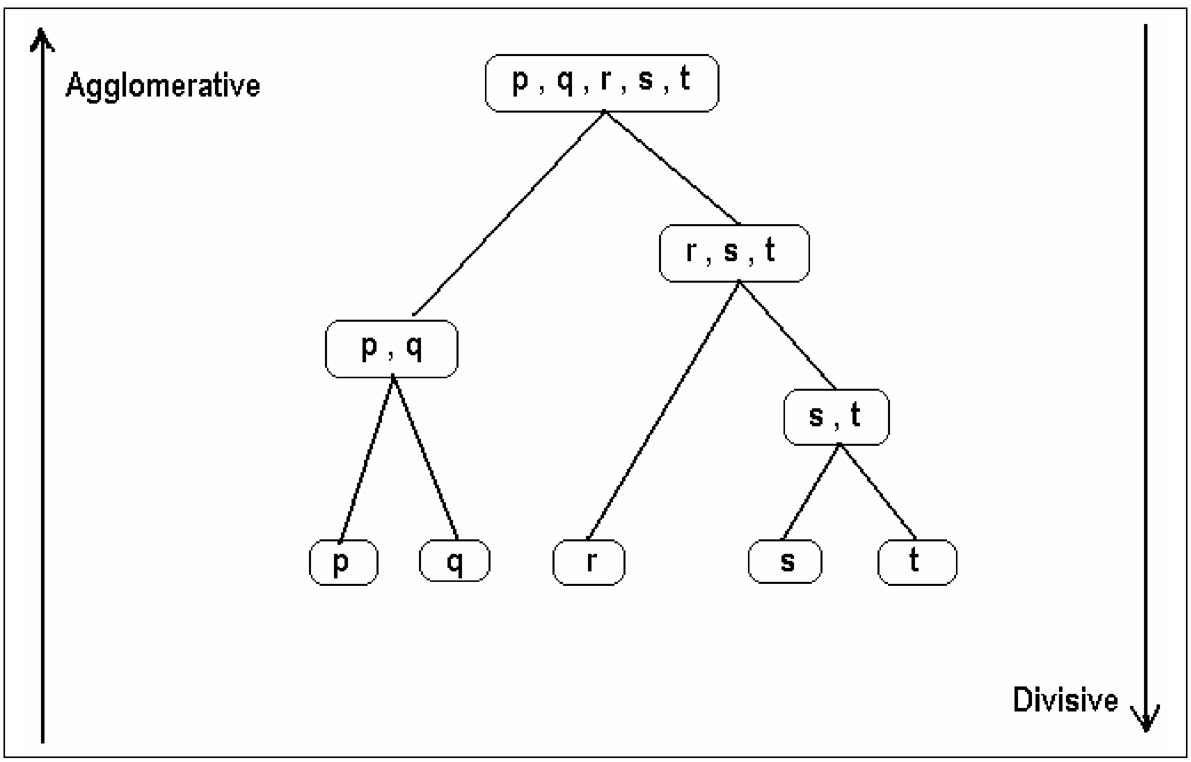
α) Agglomerative hierarchical ομαδοποίηση

Στην αρχή της διαδικασίας κάθε δεδομένο θεωρείται ως ξεχωριστό cluster και σε κάθε βήμα του αλγορίθμου γίνεται η ένωση μικρότερων ομάδων, με βάση κάποιο κριτήριο, απόσταση ή ομοιότητα, σε μεγαλύτερα μέχρι να δημιουργηθεί μία ενιαία ομάδα με όλα τα δεδομένα. Η μέθοδος αυτή απαιτεί την εύρεση μία ποσότητας που ονομάζεται ομοιότητα/απόσταση ομάδων (cluster similarity/distance), και η οποία περιγράφει το μέτρο ομοιότητας ή απόστασης των ομάδων (**Εικόνα 2.1**). Για αυτήν την μέθοδο υπάρχουν διάφοροι αλγόριθμοι, οι διαφορές των οποίων κυρίως βρίσκονται στον τρόπο εύρεσης της απόστασης ή της ομοιότητας των clusters.

b) Divisive hierarchical ομαδοποίηση

Στην αρχή της διαδικασίας όλα τα δεδομένα βρίσκονται σε ένα ενιαίο cluster και στην συνέχεια σε κάθε βήμα με βάση κάποιο κριτήριο, τα δεδομένα "σπάνε" διαδοχικά σε clusters έτσι που τελικά κάθε δεδομένο να αποτελεί και ένα ξεχωριστό cluster (**Εικόνα 2.1**).

Το αποτέλεσμα των αλγόριθμων της hierarchical ομαδοποίησης είναι ένα δένδρο από ομάδες δεδομένων το οποίο ονομάζεται δενδρόγραμμα, και το οποίο δείχνει τον τρόπο που οι ομάδες συνδέονται μεταξύ τους. Κόβοντας το δενδρόγραμμα σε ένα συγκεκριμένο επίπεδο δημιουργείται κάθε φορά ο επιθυμητός αριθμός ομάδων δεδομένων. Οι αλγόριθμοι hierarchical ομαδοποίησης που υλοποιήθηκαν είναι οι εξής: **single linkage (nearest neighbor)**, **complete linkage (furthest neighbor)**, **average linkage**, **centroid linkage** και **ward method**.



Εικόνα 2.1 : Hierarchical ομαδοποίηση

2.2.2) Partitional Ομαδοποίηση

Η partitional ομαδοποίηση είναι μία μέθοδος ομαδοποίησης, στην οποία τα δεδομένα διασπούνται άμεσα σε ένα συγκεκριμένο αριθμό από ομάδες. Αρχικά, στην μέθοδο αυτή τα δεδομένα διασπώνται σε ομάδες με κάποιο τρόπο. Στη συνέχεια μέσω μία επαναληπτικής διαδικασίας τα δεδομένα μετακινούνται από την μία ομάδα στην άλλη με βάση ένα κριτήριο. Όταν η διαδικασία τελειώσει τα δεδομένα θα έχουν διαχωριστεί σε ένα συγκεκριμένο αριθμό ομάδων. Οι τεχνικές partitional ομαδοποιήσεων που χρησιμοποιήθηκαν στην εργασία είναι οι: **k-means** [7] και **"bisecting" k-means** [10] .

2.3) Μέθοδοι Hierarchical ομαδοποίησης

2.3.1) Agglomerative Μέθοδοι ομαδοποίησης

1) Single linkage μέθοδος

Η μέθοδος αυτή χρησιμοποιεί ως κριτήριο ομοιότητας των clusters την ελάχιστη απόσταση μεταξύ τους. Στην μέθοδο αυτή, λοιπόν, αρχικά όλα τα δεδομένα βρίσκονται σε ξεχωριστά clusters και στη συνέχεια σε κάθε βήμα της επαναληπτικής διαδικασίας υπολογίζονται όλες οι αποστάσεις μεταξύ όλων των clusters που υπάρχουν. Απόσταση μεταξύ δύο clusters θεωρείται η απόσταση μεταξύ των δύο πιο «κοντινών» σημείων τους. Τα clusters που βρίσκονται πιο «κοντά» ενώνονται σε ένα ενιαίο cluster και με αυτόν τον τρόπο συνεχίζεται η επαναληπτική διαδικασία. Ο αλγόριθμος single linkage παρουσιάζεται στο παρακάτω διάγραμμα ροής (**Εικόνα 2.2**).

2) Complete linkage μέθοδος

Η μέθοδος αυτή είναι περίπου όμοια με την μέθοδο single linkage. Η μόνη διαφορά που υπάρχει ανάμεσα σε αυτή και την προηγούμενη μέθοδο είναι η μέτρηση της απόστασης ανάμεσα σε δύο clusters. Για την μέθοδο αυτή η απόσταση ανάμεσα σε δύο clusters είναι η απόσταση των πιο απομακρυσμένων σημείων των clusters. Παρακάτω ακολουθεί το διάγραμμα ροής για τον αλγόριθμο complete linkage (**Εικόνα 2.3**) .

3) Average linkage μέθοδος (Unweighted)

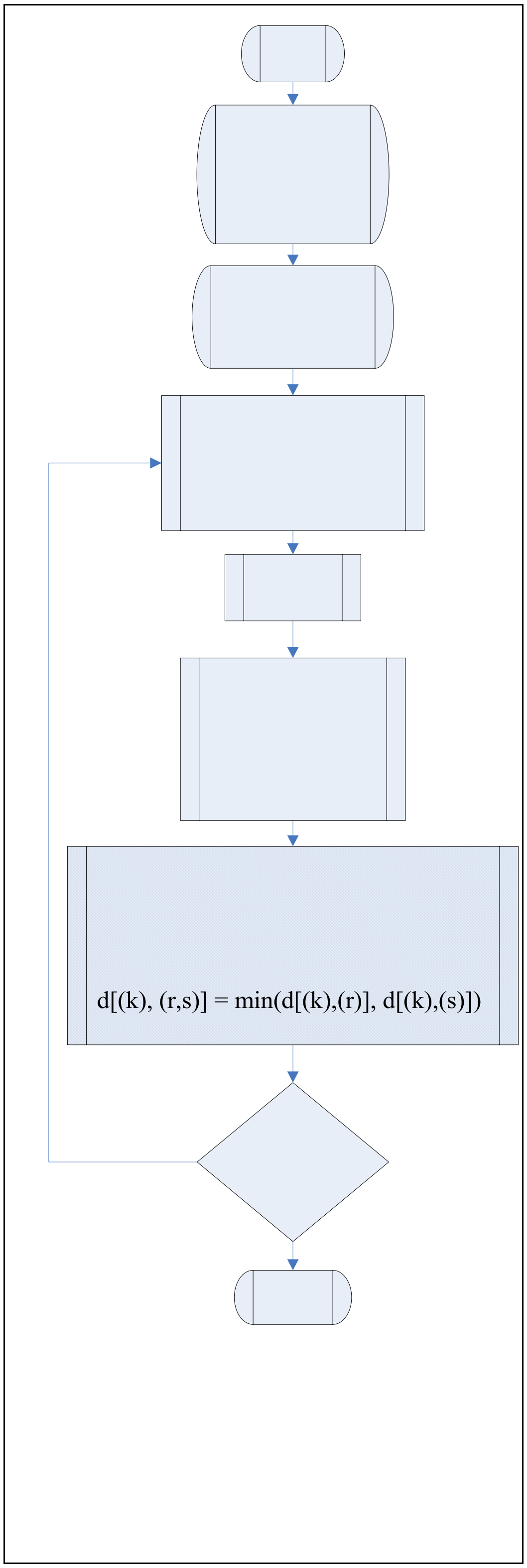
Στην μέθοδο average linkage η απόσταση μεταξύ δύο clusters υπολογίζεται από τον σταθμισμένο μέσο όλων των δυνατών ζευγαριών των clusters. Ο σταθμισμένος μέσος προκύπτει ανάλογα τον αριθμό των κόμβων που υπάρχουν σε κάθε cluster. Μαθηματικά η απόσταση μεταξύ των clusters X και Y υπολογίζεται από τον παρακάτω τύπο :

$$D(X,Y) = \frac{1}{N_X \times N_Y} \sum_{i=1}^{N_X} \sum_{j=1}^{N_Y} d(x_i, y_j);$$
$$x_i \in X, \ y_i \in Y,$$

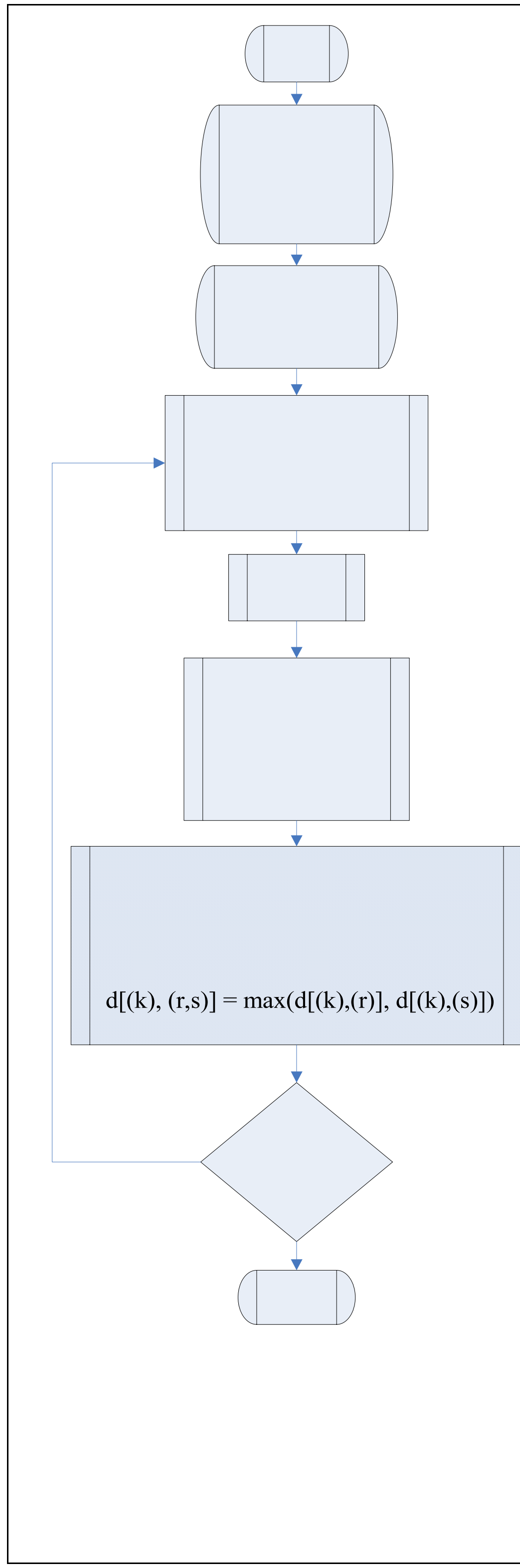
όπου:
d(x,y) είναι η απόσταση μεταξύ των αντικειμένων x, y που ανήκουν στα X και Y clusters αντίστοιχα και N_X, N_Y είναι ο αριθμός των αντικειμένων που υπάρχουν X και Y clusters αντίστοιχα. Στην **Εικόνα 2.4** παρουσιάζεται το διάγραμμα ροής της μεθόδου average linkage.

4) Centroid linkage μέθοδος

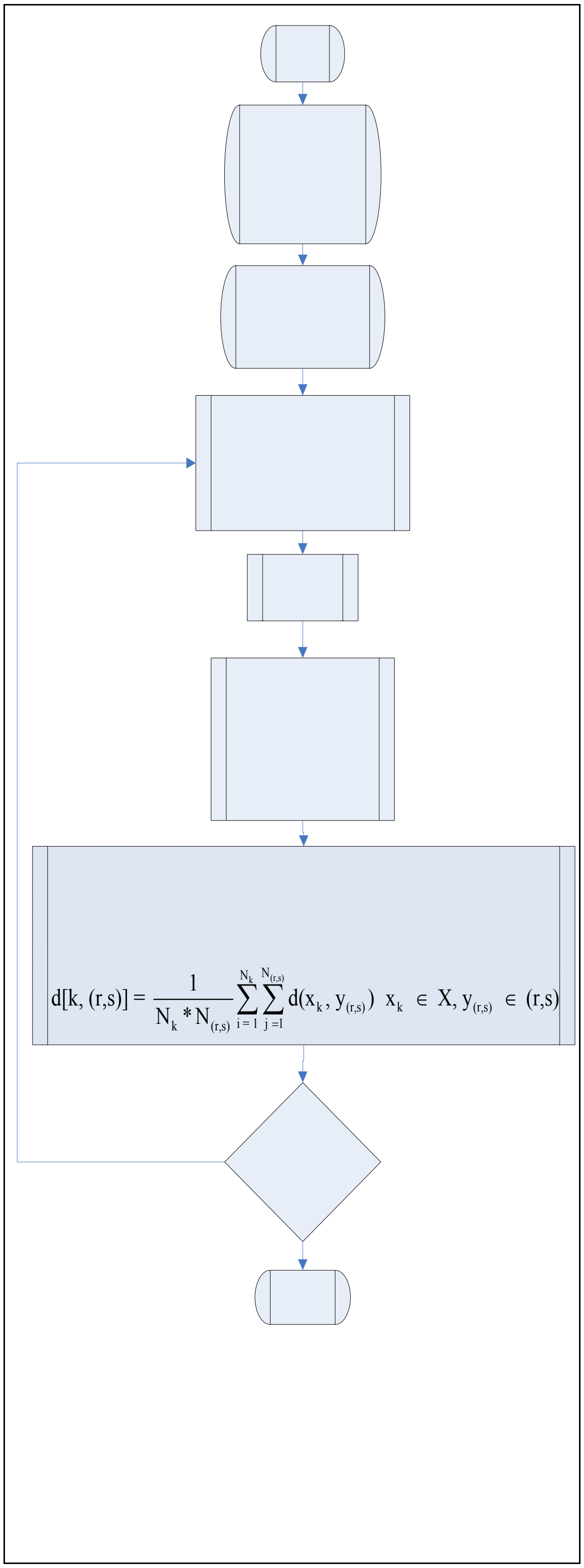
Σε αυτή τη μέθοδο η απόσταση μεταξύ δύο clusters είναι ίση με τη ευκλείδεια απόσταση των κέντρων των clusters. Συγκεκριμένα σε κάθε επαναληπτικό βήμα αυτού του αλγόριθμου ενώνονται δύο clusters σε ένα ενιαίο με βάση την απόσταση των κέντρων τους (τα clusters που απέχουν μικρότερη απόσταση). Μετά από κάθε συνένωση δημιουργείται ένα καινούριο cluster, του οποίου το νέο κέντρο είναι ο μέσος όρος όλων των σημείων που αποτελούν το cluster αυτό. Στη συνέχεια, δημιουργείται ο νέος πίνακας αποστάσεων των clusters και η επαναληπτική διαδικασία του αλγόριθμου συνεχίζεται μέχρι την δημιουργία ενός ενιαίου cluster. Το διάγραμμα ροής του αλγόριθμου παρουσιάζεται στην **Εικόνα 2.5**.



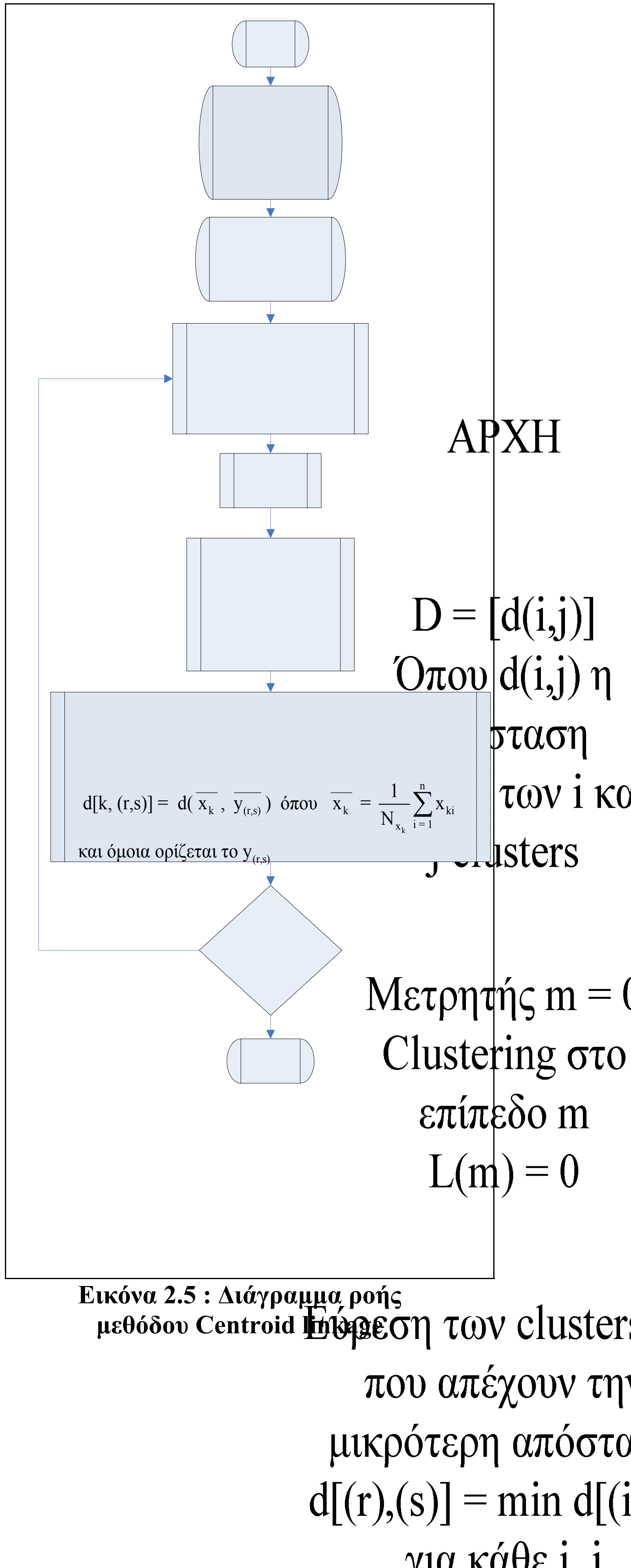
Εικόνα 2.2 : Διάγραμμα ροής Μεθόδου Single linkage



Εικόνα 2.3 : Διάγραμμα ροής Μεθόδου Complete linkage



Εικόνα 2.4 : Διάγραμμα ροής μεθόδου Average linkage



Εικόνα 2.5 : Διάγραμμα ροής μεθόδου Centroid linkage

5) Ward's linkage method

Η τελευταία μέθοδος για agglomerative ομαδοποίηση που χρησιμοποιήθηκε ήταν η Ward μέθοδος. Στην μέθοδο αυτή η απόσταση μεταξύ των clusters δίνεται από των εξής τύπο:

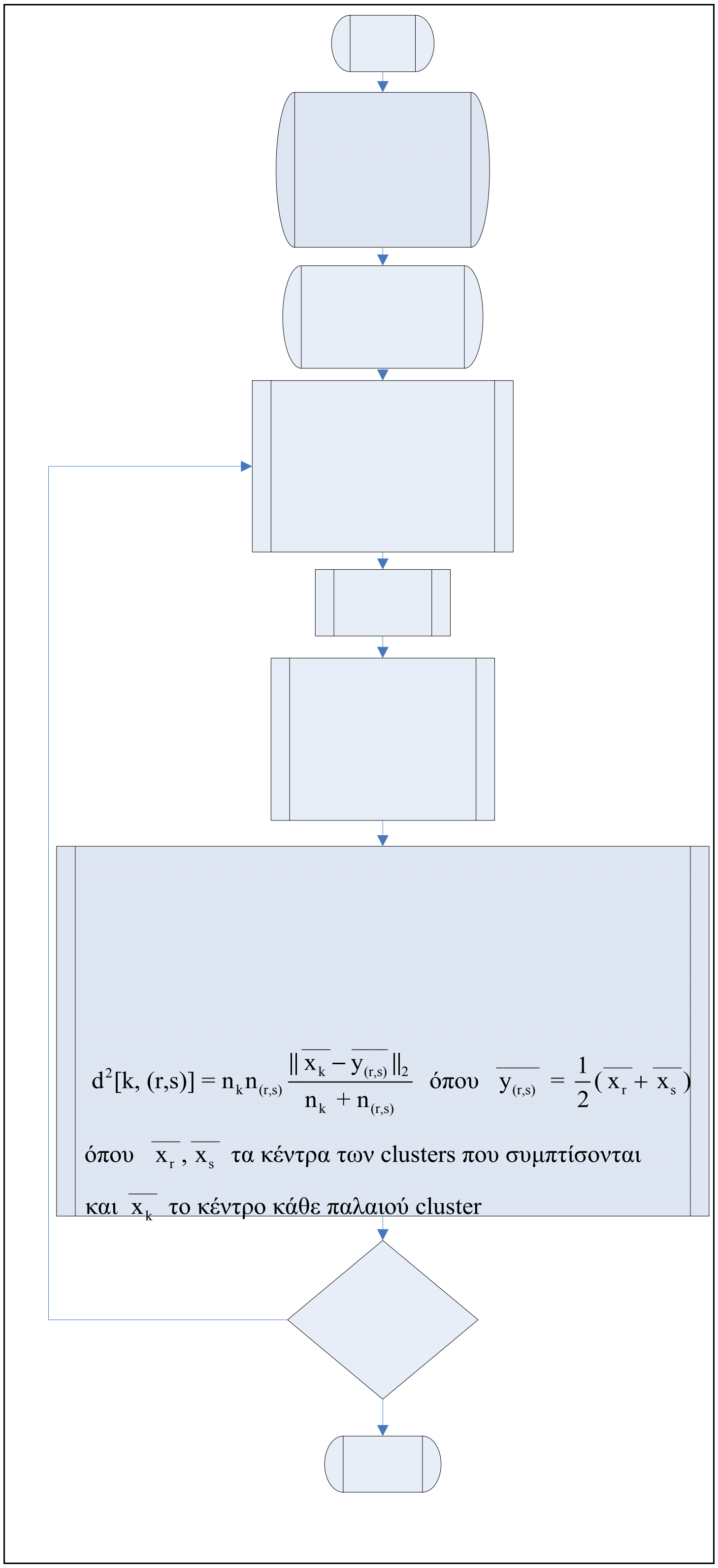
$$d^2(r,s) = n_r n_s \frac{\|\bar{x}_r - \bar{x}_s\|_2^2}{(n_r + n_s)}$$

όπου x_r και x_s είναι τα κέντρα των ομάδων r, s αντίστοιχα ενώ τα n_r, n_s είναι ο αριθμός των στοιχείων που ανήκουν στις r και s ομάδες αντίστοιχα. Η μέθοδος αυτή διαφέρει από όλες τις υπόλοιπες μεθόδους linkage στο κριτήριο της ένωσης των ομάδων. Στην συγκεκριμένη μέθοδο δεν ενώνονται οι ομάδες που απέχουν την μικρότερη απόσταση, αλλά η ένωση των ομάδων γίνεται με στόχο την μικρότερη αύξηση του αθροίσματος των τετραγώνων των αποστάσεων στο εσωτερικού κάθε ομάδας. Δηλαδή, υπολογίζεται για κάθε πιθανή συνένωση δύο clusters το άθροισμα των τετραγώνων των αποστάσεων όλων των στοιχείων από το κέντρο του cluster και στη συνέχεια πραγματοποιείται η συνένωση με το μικρότερο άθροισμα. Το διάγραμμα ροής του αλγόριθμου Ward's linkage παρουσιάζεται στην **Εικόνα 2.6**.

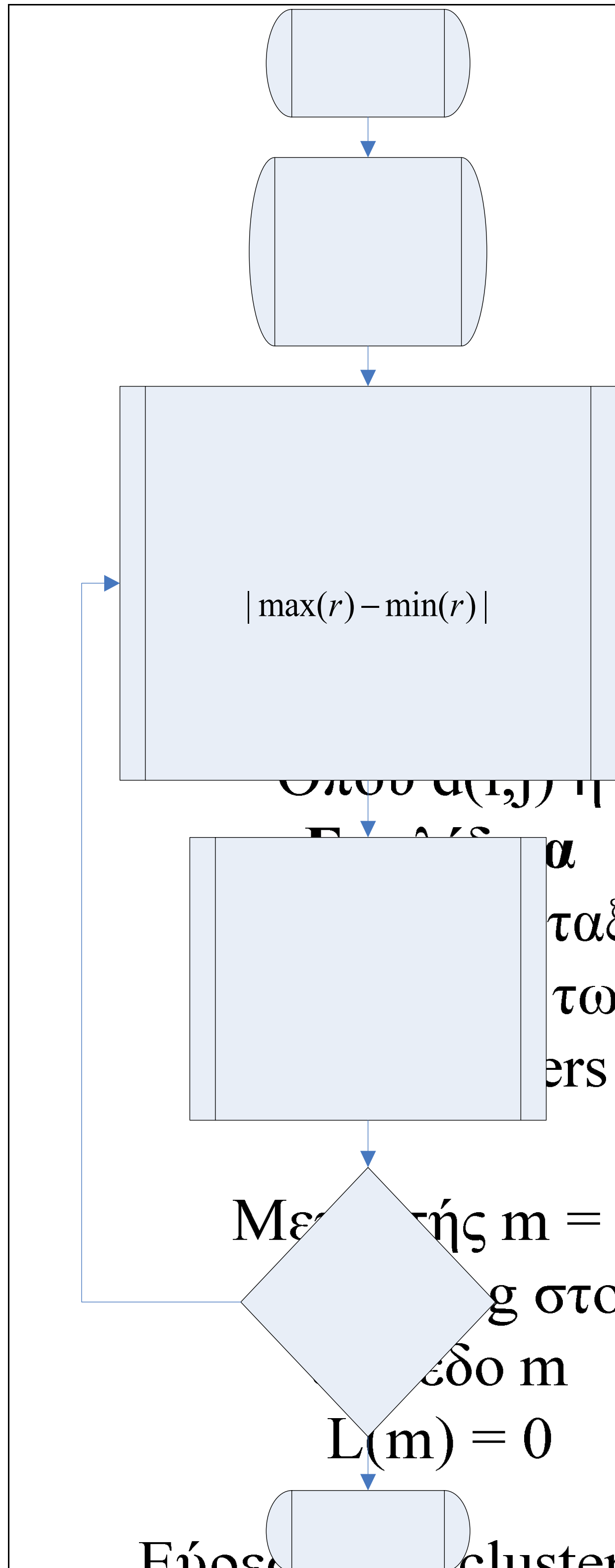
2.3.2) Divisive Μέθοδοι ομαδοποίησης

1) Minimum diameter μέθοδος

Η divisive μέθοδος χρησιμοποιεί τον αλγόριθμο "ελάχιστης διαμέτρου" του Guènoche (1991). Στην μέθοδο αυτή, σε κάθε επαναληπτικό βήμα, επιλέγεται η ομάδα με την μέγιστη διάμετρο, όπου διάμετρος μίας ομάδας ορίζεται η μέγιστη απόσταση μεταξύ δύο οποιοδήποτε στοιχείων του. Η διάσπαση της ομάδας γίνεται σε δύο άλλες ομάδες για τις οποίες ισχύει ότι, η μεγαλύτερη από αυτές τις δύο θα πρέπει να έχει την μικρότερη διάμετρο σε σχέση με τις υπόλοιπες ομάδες, που ήδη έχουν δημιουργηθεί. Η επαναληπτική διαδικασία σταματάει αφού όλα τα δεδομένα αποτελούν μία ξεχωριστή ομάδα. Τα βήματα του αλγόριθμου αυτού φαίνονται στο παρακάτω διάγραμμα ροής (**Εικόνα 2.7**).



Εικόνα 2.6 : Διάγραμμα ροής μεθόδου Ward Linkage



Εικόνα 2.7 : Διάγραμμα ροής μεθόδου Minimum Diameter

2.4) Μέθοδοι Partitional ομαδοποίησης

2.4.1) K -means αλγόριθμος

Ο αλγόριθμος K means είναι ο πιο διαδεδομένος αλγόριθμος ομαδοποίησης. Ο αλγόριθμος K means διασπάει τα δεδομένα σε K διαφορετικά clusters μέσω μίας επαναληπτικής διαδικασίας, όπου K είναι ένας ακέραιος αριθμός. Η επαναληπτική διαδικασία αυτή τερματίζει από τη στιγμή που θα ικανοποιηθεί ένα συγκεκριμένο κριτήριο. Τα κύρια βήματα του αλγορίθμου περιγράφονται παρακάτω:

1. Αρχικά γίνεται η επιλογή K σημείων στο πεδίο των δεδομένων έτσι ώστε τα σημεία αυτά να αποτελούν τα κέντρα των αρχικών ομάδων
2. Ανάθεσε κάθε δεδομένο σε μία ομάδα για το οποίο η απόσταση του από το κέντρο της ομάδας να είναι η μικρότερη από κάθε άλλη από τα κέντρα των υπολοίπων ομάδων.
3. Όταν όλα τα δεδομένα έχουν ανατεθεί στις ομάδες υπολόγισε ξανά τα κέντρα των ομάδων, παίρνοντας τον μέσο όρο των δεδομένων κάθε ομάδας.
4. Επανέλαβε τα βήματα 2,3 μέχρι να επαληθευτεί κάποιο κριτήριο σύγκλισης

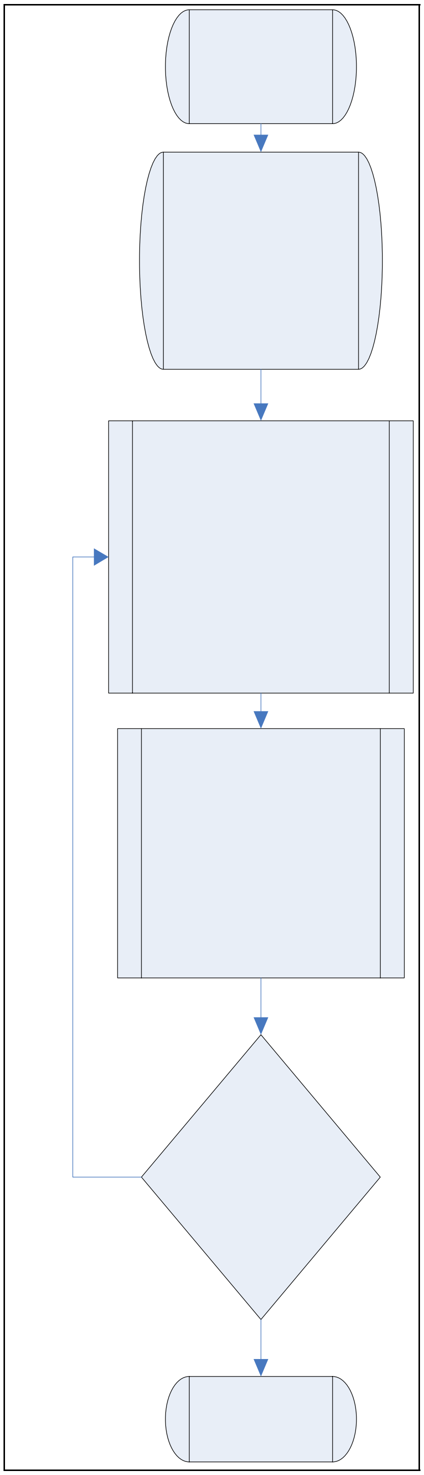
Το κριτήριο σύγκλισης του αλγορίθμου, ο τρόπος μέτρησης της απόστασης των δεδομένων με τα κέντρα των ομάδων καθώς και ο τρόπος ανάδειξης των αρχικών κέντρων τους καθορίζουν σε μεγάλο βαθμό την τελική ομαδοποίηση των δεδομένων και ορίζονται από τον χρήστη. Τα βήματα του αλγορίθμου σε μορφή διαγράμματος ροής ακολουθούν στην **Εικόνα 2.8**.

2.4.2) “Bisecting” K Means Algorithm

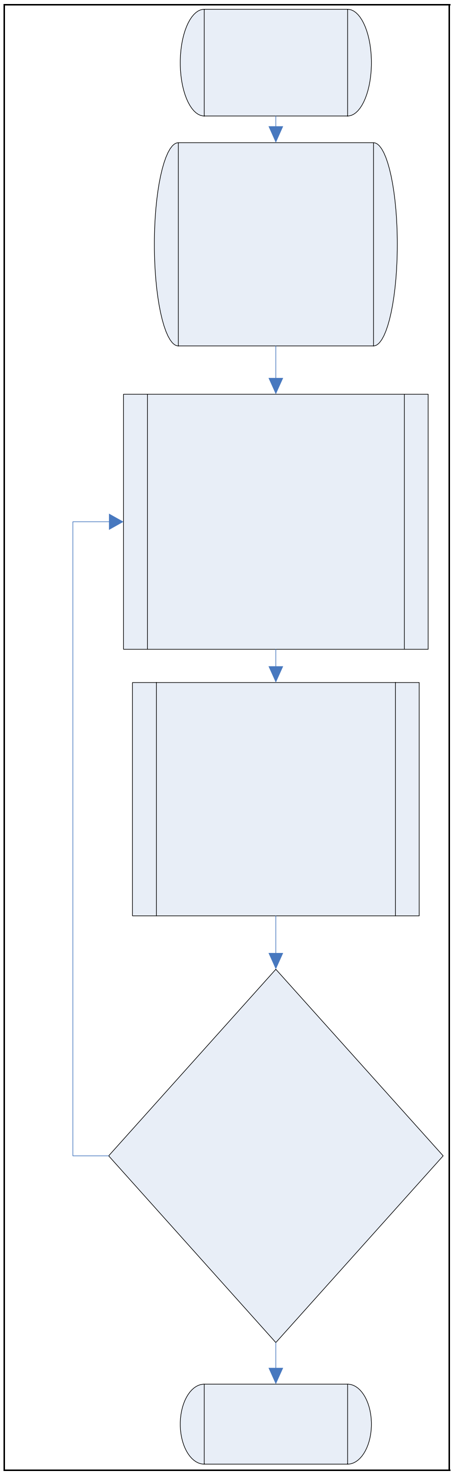
Ο αλγόριθμος “Bisecting” K means χρησιμοποιεί έμμεσα τον K means αλγόριθμο αλλά διαφοροποιείται σε κάποια σημεία από αυτόν. Γενικά θεωρείται ότι λειτουργεί αποδοτικότερα από τον K means αλγόριθμο και μπορεί να παράγει ομάδες ομοιόμορφου μεγέθους. Τα βήματα του αλγόριθμου παρουσιάζονται παρακάτω ενώ πρακτικά η όλη διαδικασία τερματίζει μετά από αρκετές επαναλήψεις :

1. Αρχικά όλα τα δεδομένα θεωρούνται ότι βρίσκονται σε μία ενιαία ομάδα
2. Επιλογή μίας ομάδας για την διάσπαση με βάση κάποιο κριτήριο που καθορίζεται από τον χρήστη
3. Διάσπασε την επιλεγμένη ομάδα σε δύο άλλα clusters χρησιμοποιώντας τον K means αλγόριθμο
4. Επανέλαβε τα βήματα 2,3 μέχρι να δημιουργηθεί ο επιθυμητός αριθμός από ομάδες δεδομένων

Το διάγραμμα ροής του αλγόριθμου έτσι όπως προκύπτει από τα παραπάνω βήματα ακολουθεί και παρουσιάζεται στην **Εικόνα 2.9**.



Εικόνα 2.8 : Διάγραμμα ροής K-Means Αλγορίθμου



Εικόνα 2.9 : Διάγραμμα ροής “Bisecting” K-Means Algorithm

ΑΡ

ΕΠΙΛ
ΣΗΜΕΙΩΝ
ΔΕΔΟΜΕ
ΘΑ ΑΠΟ
ΤΑ ΚΕΝΤ
CLUS

ΑΝΑΘΕΣ

ΚΕΦΑΛΑΙΟ 3ο

3.1) Εισαγωγή

Γενικά, ο τομέας της ανάλυσης δεδομένων σε ομάδες αποτελείται από ένα πλήθος αλγορίθμων που ομαδοποιούν τα δεδομένα με βάση κάποια κριτήρια ,όπως το μέτρο της ομοιότητας ή της “διαφοράς” των δεδομένων μεταξύ τους. Οι αλγόριθμοι που χρησιμοποιήθηκαν για την ομαδοποίηση των “σημαδιών” των κανόνων Golomb περιγράφηκαν στο προηγούμενο κεφάλαιο. Η σύγκριση μεταξύ των διαφόρων ομαδοποιήσεων που υπάρχει γίνεται με την χρήση διαφόρων στατιστικών τεχνικών. Οι στατιστικές αυτές τεχνικές στη συγκεκριμένη εργασία αναπτύχθηκαν στην πλατφόρμα της Matlab [2]. Όλες οι τεχνικές που χρησιμοποιήθηκαν στους αλγορίθμους του προηγούμενου κεφαλαίου καθώς και τα κριτήρια σύγκρισης των ομαδοποιήσεων περιγράφονται στο συγκεκριμένο κεφάλαιο.

3.2) Στατιστικές μέθοδοι στην υλοποίηση αλγορίθμων ομαδοποίησης

Για την υλοποίηση των διαφόρων αλγορίθμων χρειάστηκε αρκετές φορές να γίνει χρήση κάποιων στατιστικών μεθόδων είτε για τον υπολογισμό των αποστάσεων μεταξύ των δεδομένων, είτε για τον υπολογισμό της ομοιότητας μεταξύ των clusters είτε και για την εύρεση της βέλτιστης λύσης όσο αναφορά την ομαδοποίηση των δεδομένων. Όλες αυτές οι στατιστικές μέθοδοι υλοποιήθηκαν στην πλατφόρμα του Matlab παράγοντας κάθε φορά τα αντίστοιχα αποτελέσματα. Συγκεκριμένα για την εφαρμογή της ανάλυσης δεδομένων σε clusters από την Matlab χρησιμοποιήθηκαν συναρτήσεις του στατιστικού πακέτου ενώ τα κριτήρια αξιολόγησης, που περιγράφηκαν παραπάνω, υλοποιήθηκαν και αυτά με την βοήθεια συναρτήσεων του ίδιου πακέτου. Στα παρακάτω εδάφια περιγράφονται στατιστικές μέθοδοι καθώς και οι συναρτήσεις που χρησιμοποιούνται για την υλοποίηση του ιεραρχικού αλλά και του “partitional” clustering.

3.2.1) Στατιστικές μέθοδοι για ιεραρχικό clustering

Η διαδικασία για την υλοποίηση οποιασδήποτε ιεραρχικής ομαδοποίησης αποτελείται από κάποια συγκεκριμένα βήματα και υλοποιείται από συναρτήσεις, οι οποίες ανάλογα με τον αλγόριθμο της ομαδοποίησης παίρνουν και τις κατάλληλες παραμέτρους. Συγκεκριμένα τα βήματα των ιεραρχικών αλγορίθμων είναι τα εξής:

1. **Εύρεση της ομοιότητας ή της ανομοιότητας μεταξύ κάθε αντικειμένου-δεδομένου στο σύνολο των δεδομένων.** Σε αυτό το βήμα υπολογίζεται η απόσταση μεταξύ των αντικειμένων-δεδομένων με την χρήση της συνάρτησης **pdist**. Η συνάρτηση αυτή χρησιμοποιεί διάφορους τρόπους για την εύρεση της απόστασης μεταξύ των δεδομένων. Συγκεκριμένα η συνάρτηση **pdist** παίρνει ως όρισμα το σύνολο των δεδομένων σε έναν διάνυσμα και επιστρέφει ένα διάνυσμα, το οποίο έχει όλες τις δυνατές αποστάσεις μεταξύ των δεδομένων. Έτσι αν θεωρήσουμε ότι το αρχικό διάνυσμα (δεδομένα) έχει m διαφορετικά αντικείμενα τότε το επιστρεφόμενο διάνυσμα έχει μέγεθος m*(m-1)/2. Πολλές φορές είναι πιο εύκολο οι αποστάσεις μεταξύ των δεδομένων να εμφανίζονται σε μορφή πίνακα. Γι'αυτό το λόγο το διάνυσμα που προκύπτει από την συνάρτηση **pdist** χρησιμοποιείται ως όρισμα στην συνάρτηση **squareform**, η οποία μετατρέπει το διάνυσμα αυτό σε πίνακα, ο οποίος έχει την εξής ιδιότητα: το στοιχείο i, j του πίνακα αντιστοιχεί στην απόσταση μεταξύ του αντικειμένου i και j (προφανώς η απόσταση ενός αντικειμένου από τον εαυτό του είναι ίση με 0 και γι'αυτό το λόγο η διαγώνιος του πίνακα είναι ίση με 0). Ο πίνακας που επιστρέφεται από την συνάρτηση αυτή ονομάζεται πίνακας απόστασης ή ανομοιότητας. Η Matlab υποστηρίζει πολλούς διαφορετικούς τρόπους για την μέτρηση της απόστασης μεταξύ δύο αντικειμένων-δεδομένων. Ο τρόπος με τον οποίο μπορεί να υπολογιστεί μία απόσταση ανάμεσα σε δύο δεδομένα περνάει ως όρισμα στην συνάρτηση **pdist**. Οι διαφορετικές τεχνικές μέτρησης της απόστασης περιγράφονται παρακάτω.

2. **Ομαδοποίηση των δεδομένων σε δυαδικό, ιεραρχικό δένδρο.** Σε αυτό το βήμα ομαδοποιούνται τα ζεύγη των δεδομένων που βρίσκονται πιο κοντά χρησιμοποιώντας την συνάρτηση **linkage**. Η συνάρτηση **linkage** χρησιμοποιεί τον πίνακα των αποστάσεων που προέκυψε από το πρώτο βήμα και υπολογίζει την εγγύτητα των δεδομένων μεταξύ τους. Καθώς τα αντικείμενα-δεδομένα ομαδοποιούνται σε ομάδες και στη συνέχεια και αυτά σε μεγαλύτερες

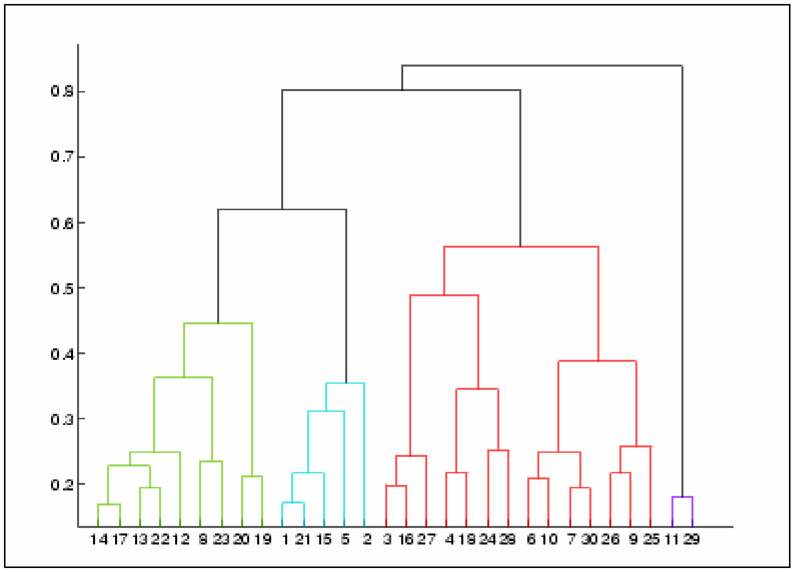
ομάδες τελικά δημιουργείται ένα ιεραρχικό δένδρο(**Εικόνα 3.1**). Η ομαδοποίηση των δεδομένων από την συνάρτηση **linkage** γίνεται μέσω διάφορων τεχνικών που καθορίζει το κριτήριο συνένωσης των clusters. Η τεχνική που καθορίζει το κριτήριο συνένωσης των ομάδων των δεδομένων περνάει σαν όρισμα στην συνάρτηση **linkage** ενώ όλες οι πιθανές τεχνικές που μπορούν να χρησιμοποιηθούν περιγράφονται παρακάτω στο κεφάλαιο. Η συνάρτηση **linkage**, όπως αναφέρθηκε και παραπάνω, δέχεται ως είσοδο το διάνυσμα που προκύπτει από την **pdist** και επιστρέφει έναν πίνακα, που στην ουσία αντιπροσωπεύει ένα ιεραρχικό δένδρο. Ο πίνακας αυτός έχει 3 στήλες και έχει την παρακάτω μορφή:

Πχ.

```
Z = linkage(Y)
Z =
    1.0000    3.0000    1.0000
    4.0000    5.0000    1.0000
    6.0000    7.0000    2.0616
    8.0000    2.0000    2.5000
```

όπου κάθε γραμμή αποτελεί και από μία ξεχωριστή ένωση ομάδων. Οι δύο πρώτες στήλες καθορίζουν ποια αντικείμενα ενώθηκαν, δηλαδή το δεδομένο 1 με το δεδομένο 3 κτλ. Η τρίτη στήλη δείχνει την απόσταση μεταξύ των δύο αυτών αντικειμένων που ενώθηκαν. Δηλαδή, στο παραπάνω παράδειγμα, τα δεδομένα 1 και 3 ενώθηκαν πρώτα έχοντας μεταξύ τους απόσταση ίση με 1 μονάδα, στη συνέχεια ενώθηκαν τα δεδομένα 4 και 5 που είχαν απόσταση ίση με 1 μονάδα κτλ. Στην τρίτη γραμμή περιέχονται η ένωση των δεδομένων 6 και 7, αλλά το δεδομένο 7 δεν υπάρχει εξ'αρχής αλλά είναι στην ουσία η ομάδα που δημιουργήθηκε από την πρώτη ένωση ανάμεσα το 1 και το 3.

3. Εύρεση του βέλτιστου σημείου για διάσπαση του ιεραρχικού δένδρου σε clusters. Αυτό που παράγεται από την χρήση των παραπάνω συναρτήσεων και μεθόδων είναι ένα ιεραρχικό δένδρο. Το σημείο διάσπασης του ιεραρχικού δένδρου καθορίζεται είτε από τον χρήστη ανάλογα με τον αριθμό των ομάδων που επιθυμεί είτε χρησιμοποιώντας την συνάρτηση του Matlab **cluster**, η οποία "κόβει" βέλτιστα το ιεραρχικό δένδρο σε ομάδες δεδομένων. Η βέλτιστη αυτή επιλογή από την συνάρτηση γίνεται είτε με την ανίχνευση της συνάρτησης **cluster** ομάδων από δεδομένα, τα οποία βρίσκονται σχετικά "κοντά", είτε "αυθαίρετα", σε σημεία που τα δεδομένα είναι πιο κοντά από οποιοδήποτε άλλα σημεία. Η παραπάνω συνάρτηση παίρνει ως είσοδο τον τρόπο με τον οποίο επιθυμεί ο χρήστης να γίνει η διάσπαση του δένδρου και αυτές οι επιλογές γίνονται είτε με βάση το κριτήριο “αστάθειας”, είτε με βάση το κριτήριο απόστασης.



Εικόνα 3.1 : Παράδειγμα δενδρογράμματος – Ιεραρχικό δένδρο

3.2.2) Στατιστικές μέθοδοι για partitional ομαδοποίηση

Οι αλγόριθμοι της partitional ομαδοποίησης οργανώνουν τα αποτελέσματα τους σε ξεχωριστές ομάδες. Οι partitional αλγόριθμοι που υλοποιήθηκαν ήταν ο **k-means** [7] και ο **“bisecting” k-means** [10]. Γενικά, οι αλγόριθμοι που υλοποιούν partitional ομαδοποίηση χρησιμοποιούν ένα κριτήριο και υπολογίζουν ποια ομαδοποίηση με συγκεκριμένο αριθμό, K ομάδες, οδηγεί στην ελάχιστη τιμή του κριτηρίου αυτού.

1. Υλοποίηση των αλγόριθμων kmeans και “bisecting” kmeans

Στην πλατφόρμα της Matlab υπάρχει υλοποιημένη η συνάρτηση **kmeans**, η οποία υλοποιεί τον k-means αλγόριθμο παίρνοντας ως παράμετρο τα δεδομένα τα οποία θα πρέπει να ομαδοποιηθούν. Η συνάρτηση **kmeans** χρησιμοποιεί τα δεδομένα σαν σημεία στο χώρο και ομαδοποιεί αυτά που βρίσκονται πιο κοντά, συγκρίνοντας πάντα με 5 διαφορετικούς τρόπους μέτρησης την απόσταση μεταξύ τους. Κάθε ομάδα καθορίζεται από τα μέλη της αλλά και από το κέντρο της. Η ομαδοποίηση των δεδομένων μπορεί να καθοριστεί από κάποιες παραμέτρους που παίρνει η συνάρτηση **kmeans** και καθορίζουν είτε τον μέγιστο αριθμό επαναλήψεων είτε τις αρχικές τιμές των κέντρων των ομάδων. Η συνάρτηση, αυτή, χρησιμοποιεί επαναληπτικά τον αλγόριθμο του k-means και όλη η διαδικασία περιλαμβάνει μετακίνηση των δεδομένων από μία ομάδα σε μία άλλη μέχρι η ομαδοποίηση να ελαχιστοποιεί ένα συγκεκριμένο κριτήριο. Γενικά, πρέπει να αναφερθεί ότι η partitional ομαδοποίηση είναι πιο κατάλληλη για ομαδοποίηση μεγάλου αριθμού δεδομένων από την ιεραρχική.

2. Απεικόνιση της partitional ομαδοποίησης

Η απεικόνιση των ομαδοποιήσεων των δεδομένων στην περίπτωση της partitional ομαδοποίησης γίνεται με τη βοήθεια μίας συνάρτησης της Matlab, της **silhouette** [2]. Η συνάρτηση αυτή παίρνει ως παραμέτρους τα δεδομένα καθώς και έναν πίνακα που καθορίζει οι ομάδες καθώς και τα δεδομένα που ανήκουν σε κάθε ομάδα. Η απεικόνιση της ομαδοποίησης γίνεται με βάση την απόσταση στο εσωτερικό κάθε ομάδας. Η μέτρηση της απόστασης μπορεί να γίνει με διαφορετικές τεχνικές, οι οποίες περιγράφονται παρακάτω και οι οποίες περνάνε ως παράμετροι στην συνάρτηση του Matlab **silhouette**. Γενικά, η επιστρεφόμενη τιμή της συνάρτησης αυτής είναι ένας πίνακας που περιέχει την τιμή "silhouette" για κάθε δεδομένο, το οποίο δείχνει και την εγγύτητα του μέσα στην ομάδα στην οποία ανήκει.

3.3) Κριτήρια αξιολόγησης ομαδοποίησης

Η σύγκριση του κατά πόσο καλή είναι μία ομαδοποίηση ή όχι για κάθε αλγόριθμο έγινε με βάση κάποια συγκεκριμένα κριτήρια. Από αυτά τα κριτήρια, άλλα υλοποιούνται από την πλατφόρμα της MATLAB [2], που χρησιμοποιήθηκε, για την υλοποίηση των αλγορίθμων, και κάποια άλλα κριτήρια υλοποιήθηκαν ξεχωριστά.

3.3.1) Κριτήριο συντελεστή συσχέτισης

Ο συντελεστής συσχέτισης είναι ένας συντελεστής που προκύπτει κυρίως στην περίπτωση του ιεραρχικού clustering. Ο τρόπος εξαγωγής του είναι η σύγκριση της πληροφορίας των αποστάσεων που προκύπτει από την ομαδοποίηση και τις ομάδες που δημιουργούνται με την πληροφορία που προκύπτει από όλες τις δυνατές αποστάσεις μεταξύ των στοιχείων πριν το clustering. Για παράδειγμα, έχοντας μία ομάδα από αντικείμενα (0,1,2,...,m) και εφαρμόζοντας κάποια ιεραρχική μέθοδο ομαδοποίησης πάνω σε αυτά τα στοιχεία προκύπτει κάποιο δενδρόγραμμα, το οποίο περιέχει πληροφορίες για τις αποστάσεις ανάμεσα στις ομάδες αλλά και για τις ομάδες στο εσωτερικό των ομάδων. Στη συνέχεια χρησιμοποιώντας όλες τις πιθανές αποστάσεις όλων των στοιχείων (0,1,...,m) προκύπτει ένας συντελεστής, ο οποίος μετράει την "παραμόρφωση" της ομαδοποίησης, παρουσιάζοντας κατά πόσο ικανοποιητικά τα δεδομένα ταιριάζουν στη δομή που τους έχει δοθεί. Σε μία καλή ομαδοποίηση των δεδομένων η τιμή του συντελεστή συσχέτισης πρέπει να βρίσκεται όσο πιο κοντά στην μονάδα.

Στην πλατφόρμα της Matlab υπάρχει η εντολή **cophenet**, η οποία δίνοντας της ως παραμέτρους την ιεραρχική μέθοδο ομαδοποίησης και όλες τις αποστάσεις ανάμεσα στα στοιχεία δίνει το συντελεστή συσχέτισης. Ο μαθηματικός τύπος από τον οποίο προκύπτει ο συντελεστής συσχέτισης είναι αυτός που παρουσιάζεται παρακάτω:

$$c = \frac{\sum_{i<j}(Y_{ij}-y)(Z_{ij}-z)}{\sqrt{\sum_{i<j}(Y_{ij}-y)^2\sum_{i<j}(Z_{ij}-z)^2}}$$

όπου :

- Y_{ij} είναι η απόσταση μεταξύ των αντικειμένων i και j που προκύπτει από την ομαδοποίηση
- Z_{ij} είναι η απόσταση μεταξύ των αντικειμένων i και j που προκύπτει από τις αποστάσεις που υπήρχαν μεταξύ των αντικειμένων πριν την ομαδοποίηση
- y και z είναι ο μέσος όρος των αποστάσεων μετά την ομαδοποίηση και πριν από αυτό, αντίστοιχα

3.3.2) Κριτήριο συντελεστή "αστάθειας"

Ο συντελεστής "αστάθειας" χρησιμοποιείται για την ιεραρχική ομαδοποίηση και υπολογίζεται από το δένδρο (δενδρόγραμμα). Στο δενδρόγραμμα τα νούμερα στον οριζόντιο άξονα αποτελούν αντίγραφα των αντικειμένων στο πραγματικό σύνολο δεδομένων. Οι ενώσεις μεταξύ των αντικειμένων γίνονται με γραμμές U (ανάποδα τοποθετημένες), ενώ το ύψος των γραμμών καθορίζει την απόσταση μεταξύ των ομάδων, που ενώνονται. Ο συντελεστής "αστάθειας" χαρακτηρίζει κάθε ένωση μεταξύ των ομάδων. Στην ουσία αυτό που γίνεται είναι μία σύγκριση ανάμεσα σε κάθε ένωση δύο ομάδων με το μέσο όρο των ενώσεων που υπάρχουν στο ίδιο επίπεδο. Όσο πιο υψηλός είναι ο συντελεστής για μία ένωση δύο clusters τόσο πιο ανόμοιες ομάδες συνδέονται μεταξύ τους (**Εικόνα 3.1**).

Στην πλατφόρμα της Matlab, η οποία χρησιμοποιήθηκε για την υλοποίηση των διαφόρων αλγορίθμων ομαδοποίησης και συγκεκριμένα των ιεραρχικών υπάρχει υλοποιημένη η εντολή **inconsistent**, η οποία υπολογίζει το συντελεστή “αστάθειας” για κάθε ένωση δύο clusters. Αφού προκύπτει ο συντελεστής “αστάθειας” για κάθε ένωση μεταξύ δύο clusters για την εύρεση του ενιαίου συντελεστή απλά θεωρείται ο μέσος όρος όλων των συντελεστών. Αυτό μπορεί να αιτιολογηθεί από το γεγονός ότι, αν κάποιος αλγόριθμος δεν δίνει καλή ομαδοποίηση των δεδομένων τότε και οι επιμέρους συντελεστές θα έχουν μεγάλη τιμή, όπως και ο μέσος όρος, ενώ ο αλγόριθμος με την καλύτερη ομαδοποίηση θα δίνει συντελεστές μικρότερους συνεπώς και ο μέσος όρος θα έχει μικρή τιμή. Για κάθε ένωση ο συντελεστής αστάθειας υπολογίζεται από τον παρακάτω τύπο :

$$Y(k,4) = (z(k,3) - Y(k,1)) / Y(k,2) \text{ για κάθε k}$$

όπου το Y είναι η έξοδος της εντολής **inconsistent** ενώ το Z είναι ο πίνακας που προκύπτει από την εφαρμογή του κατάλληλου αλγορίθμου πάνω στα δεδομένα και περιγράφει την ομαδοποίηση τους. Αυτά που αντιπροσωπεύει κάθε στήλη του πίνακα Y είναι τα παρακάτω :

Στήλη 1 : Ο μέσος όρος των αποστάσεων (των ενώσεων) που περιλαμβάνονται στους υπολογισμούς

Στήλη 2 : Η απόκλιση από το μέσο όρο όλων των αποστάσεων (των ενώσεων) που περιλαμβάνονται στους υπολογισμούς.

Στήλη 3 : Ο αριθμός των ενώσεων που περιλαμβάνονται στους υπολογισμούς

Στήλη 4 : Ο συντελεστής αστάθειας για κάθε ένωση ανάμεσα σε δύο clusters

3.4) Τεχνικές μέτρησης απόστασης

Κάποιες από τις στατιστικές μεθόδους που χρησιμοποιήθηκαν στο clustering δεδομένων κάνουν χρήση διαφόρων τεχνικών για τον υπολογισμό των αποστάσεων μεταξύ των δεδομένων. Γενικά, οι τεχνικές αυτές διαφέρουν στον τύπο που χρησιμοποιούν για τον υπολογισμό των αποστάσεων μεταξύ των δεδομένων. Παρακάτω αναφέρονται όλοι οι τρόποι υπολογισμών των αποστάσεων που χρησιμοποιήθηκαν στην συγκεκριμένη εργασία.

3.4.1) Ευκλείδεια απόσταση

Η ευκλείδεια απόσταση είναι η πιο συνήθης τεχνική για τον υπολογισμό της απόστασης ανάμεσα σε δύο αντικείμενα. Η ευκλείδεια απόσταση ανάμεσα σε δύο σημεία είναι το μήκος της μικρότερης (σε μήκος) γραμμής που ενώνει τα δύο αυτά σημεία. Γενικά, θεωρώντας δύο διανύσματα x_r και x_S, ο τύπος υπολογισμού της ευκλείδειας απόστασης ανάμεσα στα διανύσματα δίνεται από τον τύπο:

$$d_{rs}^2 = (x_r - x_s)(x_r - x_s)'$$

3.4.2) Standardized ευκλείδεια απόσταση

Η "τυποποιημένη" ευκλείδεια απόσταση είναι περίπου ίδια με την ευκλείδεια απόσταση, με την διαφορά ότι χρησιμοποιεί έναν διαγώνιο πίνακα D, ο οποίος εμφανίζει την διαφορά του διανύσματος Xj πάνω σε όλες τις μεταβλητές. Στην ουσία στην μέτρηση της απόστασης αυτής, γίνεται αρχικά μία κανονικοποίηση των δεδομένων και στην συνέχεια μία μέτρηση της απόστασης με βάση την Ευκλείδεια απόσταση. Ο τύπος που χρησιμοποιείται για τον υπολογισμό της απόστασης είναι ο εξής:

$$d_{rs}^2 = (x_r - x_s)D^{-1}(x_r - x_s)'$$

3.4.3) Απόσταση Mahalanobis

Ο τύπος που χρησιμοποιείται για τον υπολογισμό της απόστασης Mahalanobis είναι ο εξής:

$$d_{rs}^2 = (x_r - x_s)V^{-1}(x_r - x_s)'$$

όπου x_r, x_S διανύσματα και V είναι ο πίνακας συσχετισμών μεταξύ των δεδομένων- αντικειμένων. Η μέθοδος αυτή χρησιμοποιείται ιδιαίτερα για την εύρεση της ομοιότητας μεταξύ των σημείων και θεωρείται ανώτερη από την Ευκλείδεια απόσταση, διότι δεν εξαρτάται μόνο από την μέση τιμή των τιμών αλλά και από τον συσχετισμό ανάμεσα στις μεταβλητές καθώς και από την κατανομή των σημείων.

3.4.4) Απόσταση “City Block”

Ο τύπος που χρησιμοποιείται για τον υπολογισμό την City Block απόσταση κάνει χρήση της απόλυτης τιμής διαφοράς των στοιχείων των διανυσμάτων των δεδομένων. Η απόσταση, αυτή, είναι γνωστή και ως απόσταση Manhattan και αντικατοπτρίζει την απόσταση μεταξύ σημείων σε ένα πλέγμα οδικού δικτύου εντός πόλεως. Ο τύπος υπολογισμού της απόστασης είναι αυτός που παρουσιάζεται παρακάτω ενώ τα x_r και x_S αποτελούν σημεία σε ένα N-διάστατο χώρο:

$$d_{rs} = \sum_{j=1}^{n} \left| x_{rj} - x_{sj} \right|$$

3.4.5) Απόσταση Συσχέτισης

Η απόσταση αυτή υπολογίζει τον συσχετισμό μεταξύ μιας σειράς σημείων. Κατά τον υπολογισμό της απόστασης έχουμε τον υπολογισμό της συσχέτισης για κάθε σημείο και για κάθε επίπεδο ξεχωριστά. Ο τύπος υπολογισμού της απόστασης είναι ο εξής:

$$d_{rs} = 1 - \frac{(x_r - \overline{x_r})(x_s - \overline{x_s})'}{\left[(x_r - \overline{x_r})(x_r - \overline{x_r})' \right]^{\frac{1}{2}} \left[(x_s - \overline{x_s})(x_s - \overline{x_s})' \right]^{\frac{1}{2}}}$$

όπου x_r και x_S διανύσματα σημείων και $\overline{x_r} = \frac{1}{n}\sum_j x_{rj}$ και $\overline{x_s} = \frac{1}{n}\sum_j x_{sj}$.

3.4.6) Απόσταση Hamming

Η Hamming απόσταση υπολογίζει τον αριθμό των διαφορών των αντίστοιχων θέσεων για τα σημεία N-διάστατου επιπέδου. Έτσι για παράδειγμα η διαφορά Hamming ανάμεσα σε δύο νούμερα δυαδικής μορφής, πχ 1001 και 1101 είναι 1, διότι στις αντίστοιχες θέσεις η διαφορά των bits ανάμεσα σε αυτά τα δύο νούμερα είναι 1(μόνο το δεύτερο bit διαφέρει). Η απόσταση αυτή υπολογίζει τον συσχετισμό μεταξύ μιας σειράς σημείων. Κατά τον υπολογισμό της απόστασης έχουμε τον υπολογισμό της συσχέτισης για κάθε σημείο και για κάθε επίπεδο ξεχωριστά. Ο τύπος υπολογισμού της απόστασης είναι ο εξής: $d_{rs} = (\#(x_{rj} \neq x_{sj})/n)$.

3.5) Τεχνικές αρχικοποίησης των κεντρικών σημείων των clusters

Στο partitional clustering και συγκεκριμένα στον αλγόριθμο K-means, όπως αναφέρθηκε και παραπάνω, θα πρέπει αρχικά να γίνει μία επιλογή των κέντρων των clusters. Στην πλατφόρμα του Matlab, όπου και υλοποιήθηκε ο συγκεκριμένος αλγόριθμος, δίνεται η επιλογή της αρχικής θέσης των κέντρων. Η επιλογή αυτή αποτελείται από τρεις διαφορετικές επιλογές οι οποίες είναι οι εξής:

3.5.1) Τυχαίες θέσεις

Με αυτή την επιλογή επιλέγονται τυχαία K θέσεις των κέντρων, όπου βέβαια K είναι ο επιλεγμένος αριθμός των ομάδων στα οποία θα διασπαστούν τα δεδομένα.

3.5.2) Ομοιόμορφη κατανομή θέσεων

Σε αυτή την περίπτωση για την επιλογή των κέντρων των clusters ακολουθείται μία συγκεκριμένη διαδικασία, η οποία είναι η εξής: όλη η σειρά των δεδομένων, διασπάται σε K αριθμό ομάδων και στην συνέχεια επιλέγεται το κέντρο με βάση αυτές τις ομάδες, που δημιουργήθηκαν. Έπειτα, με αρχικά κέντρα, τα κέντρα που βρέθηκαν από την προηγούμενη διαδικασία, ξεκινάει η επαναληπτική διαδικασία εύρεσης των ομάδων με βάση τα βήματα του αλγορίθμου.

3.5.3) "Δοκιμαστικό" clustering

Με αυτή την παράμετρο γίνεται μία αρχική ομαδοποίηση σε ένα ποσοστό των δεδομένων, 10%, και στην συνέχεια από τις ομάδες, που θα προκύψουν από αυτή την ομαδοποίηση, θα βγουν τα κέντρα των ομάδων που θα δοθούν ως αρχικές θέσεις κέντρων για την ομαδοποίηση όλων των δεδομένων.

3.6) Σύνοψη

Σε αυτό το κεφάλαιο περιγράφηκαν οι διάφορες στατιστικές μέθοδοι που χρησιμοποιήθηκαν είτε στην υλοποίηση των αλγορίθμων, είτε στην σύγκριση των αλγορίθμων, είτε στον υπολογισμό των αποστάσεων μεταξύ τους είτε ακόμα και στην αρχικοποίηση των κεντρικών στοιχείων στους partitional αλγορίθμους.

ΚΕΦΑΛΑΙΟ 4ο

4.1) Εισαγωγή

Στο κεφάλαιο 4 θα παρουσιαστεί μία αναλυτική περιγραφή των διαφόρων τεχνικών, που χρησιμοποιήθηκαν κατά την υλοποίηση των ήδη υπαρχόντων αλγορίθμων, καθώς και κάποιες νέες ιδέες που προέκυψαν κατά την προσπάθεια για βελτίωση των αποτελεσμάτων.

4.2) Αναπαράσταση δεδομένων

Τα δεδομένα, που θα έπρεπε να χρησιμοποιηθούν και να αναλυθούν σε ομάδες είναι τα "σημάδια" στους κανόνες Golomb , τα οποία, τελικώς, θα αποτελέσουν και τις διευθύνσεις σε ένα θεωρητικό δίκτυο αισθητήρων. Κατά την υλοποίηση των αλγορίθμων και την εφαρμογή τους πάνω στα "σημάδια" αυτά υπήρξαν τρεις διαφορετικές μορφές με τις οποίες δόθηκαν τα δεδομένα και οι οποίες περιγράφονται παρακάτω:

4.2.1) Αριθμητική αναπαράσταση

Στην διπλωματική εργασία του κ. Δημητρομανωλάκη [1], Analysis of the Golomb Ruler and the Sidon Set Problems, and Determination of Large, Near-Optimal Golomb Rulers, είχε υλοποιηθεί ένας αλγόριθμος, ο οποίος κατασκευάζει όλους τους **σχεδόν βέλτιστους κανόνες Golomb** για ένα συγκεκριμένο αριθμό από "σημάδια". Στην εργασία αυτή χρησιμοποιήθηκε ο σχεδόν βέλτιστος κανόνας που προέκυπτε από την υλοποίηση του κ. Δημητρομανωλάκη [1] και ο οποίος είχε το μικρότερο μήκος. Στην αρχική προσπάθεια για ομαδοποίηση των "σημαδιών" σε ομάδες χρησιμοποιήθηκε η απλή αναπαράσταση, όπου κάθε "σημάδι" αποτελούνταν από την πληροφορία που είχε μόνο από τον αριθμό, ο οποίος του αντιστοιχούσε σε δεκαδική βάση. Με αυτόν τον τρόπο, για την εύρεση της απόστασης είτε ανάμεσα στα "σημάδια" είτε ανάμεσα στα clusters, τα οποία δημιουργούνται από την συνένωση των "σημαδιών", χρησιμοποιείται απλά η διαφορά των τιμών των "σημαδιών" χρησιμοποιώντας τον κατάλληλο τύπο για τον τρόπο μέτρησης της απόστασης.

4.2.2) Διανυσματική αναπαράσταση

Ένας άλλος τρόπος αναπαράσταση της πληροφορίας της απόστασης των "σημαδιών" των κανόνων Golomb, που χρησιμοποιήθηκε, είναι η διανυσματική αναπαράσταση. Στον τρόπο αυτό το κάθε σημάδι του κανόνα Golomb απεικονίζει ένα σημείο σε N-διάστατο χώρο, όπου N είναι ο αριθμός των σημαδιών των κανόνων Golomb, που χρησιμοποιήθηκε. Η λογική της ιδέας αυτής είναι ότι σε κάθε αντίστοιχη θέση του διανύσματος, το οποίο θα αντιστοιχεί σε ένα σημάδι του κανόνα Golomb, θα βρίσκεται η απόσταση του σημείου αυτού από το αντίστοιχο σημάδι του κανόνα Golomb. Συνεπώς, κάθε διάνυσμα – "σημάδι" έχει ως συντεταγμένες τις αποστάσεις από όλα τα υπόλοιπα σημεία του κανόνα Golomb. Το θετικό με αυτή την αναπαράσταση πληροφορίας είναι ότι για κάθε σημείο συμπεριλαμβάνεται και η πληροφορία της απόστασης των σημείων μεταξύ τους. Φυσικά, η απόσταση μεταξύ των σημαδιών υπολογίζεται ανάλογα με τους τρόπους της απόστασης που αναφέρθηκαν σε προηγούμενο κεφάλαιο.

4.2.3) Δυαδική αναπαράσταση

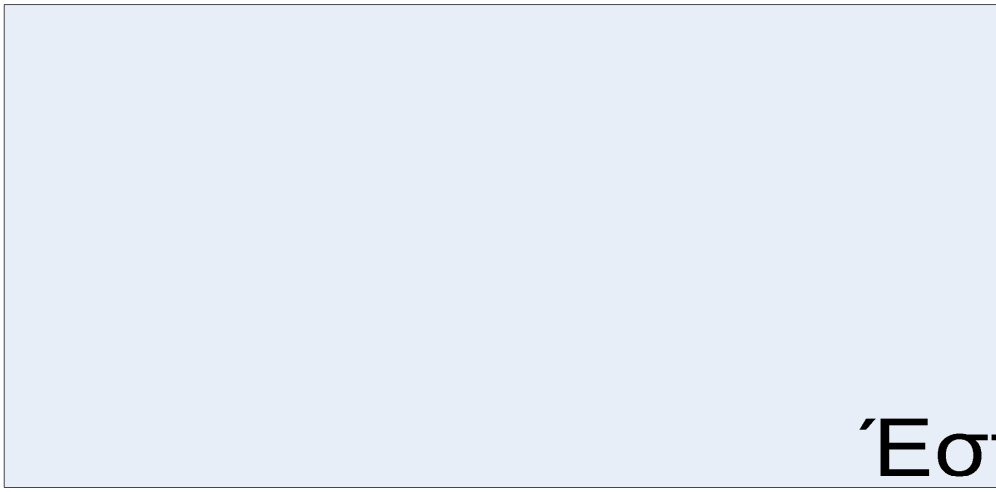
Μία άλλη μορφή αναπαράστασης της πληροφορίας, που χρησιμοποιήθηκε για τους αλγόριθμους clustering είναι η δυαδική αναπαράσταση. Με αυτή την μέθοδο υπολογίζονται οι αποστάσεις ανάμεσα στα σημάδια του κανόνα Golomb και στην συνέχεια η απόσταση αυτή μετατρέπεται σε δυαδική μορφή. Αυτό ήταν ιδιαίτερα χρήσιμο σε κάποιους αλγόριθμους διότι η ομαδοποίηση γινόταν με βάση την διαφορά των bits που υπήρχαν ανάμεσα στις αποστάσεις. Επίσης, υπάρχει μία συγκεκριμένη περίπτωση μέτρησης απόστασης, απόσταση Hamming, για την οποία η απόσταση ανάμεσα στα σημάδια του κανόνα Golomb είναι οι διαφορές στα bits των δυαδικών αναπαραστάσεων των αποστάσεων τους.

4.3) Μέθοδοι ομαδοποίησης

Γενικά στην ανάλυση της ομαδοποίησης των “σημαδιών” των κανόνων Golomb χρησιμοποιήθηκαν έτοιμοι αλγόριθμοι και αλγόριθμοι, που υπακούν σε έναν ευρυστικό κανόνα. Συγκεκριμένα, οι αλγόριθμοι που υπακούν σε έναν ευρυστικό κανόνα υλοποιήθηκαν με τέτοιο τρόπο ώστε να εξυπηρετούν ακριβώς τον λόγο για τον οποίο χρησιμοποιήθηκαν, δηλαδή είτε να δημιουργούν ομοιόμορφη ομαδοποίηση ως προς το μέγεθος των ομάδων, είτε να δημιουργούν τέτοια ομαδοποίηση, η οποία να μειώνει τον αριθμό των bits που χρησιμοποιούνται για την μετάδοση πληροφορίας σε ένα ασύρματο δίκτυο με ομαδοποιημένους κόμβους Παρακάτω περιγράφονται **3** (**τρεις**) ευρυστικοί αλγόριθμοι, οι οποίοι κατασκευάστηκαν με την ιδέα ότι η ομαδοποίηση θα γίνει πάνω σε “σημάδια” κανόνων Golomb και κατά δεύτερο λόγο πάνω στα κριτήρια που αναφέρθηκαν παραπάνω.

4.3.1) Ομαδοποίηση για weighted binary αριθμούς

Στην περίπτωση που η διευθυνσιοδότηση γίνεται με τους weighted binary αριθμούς η κάθε ομάδα καθορίζεται από την πληροφορία ενός bit. Δηλαδή, αν ένας κόμβος ανήκει , πχ στην κόκκινη ομάδα και όχι στην κίτρινη τότε θα υπάρχουν 2 bits που θα καθορίζουν την διεύθυνση του συγκεκριμένου κόμβου ως προς τις ομαδοποιήσεις και θα έχουν την μορφή 10. Έτσι για παράδειγμα, αν υπάρχει μία ομαδοποίηση, η οποία να αποτελείται από **8 διαφορετικές ομάδες** η διεύθυνση κάθε κόμβου θα αποτελείται από **8 bits**, για τα οποία όλοι οι κόμβοι που ανήκουν στην ίδια ομάδα θα έχουν στο bit της ίδιας θέσης την τιμή 1, διαφορετικά αν ένας κόμβος δεν ανήκει σε μία ομάδα στην αντίστοιχη θέση το bit, θα είναι ίσο με 0. Με αυτή την κωδικοποίηση της πληροφορίας δίνεται η δυνατότητα ένας κόμβος να ανήκει σε δύο ή παραπάνω ομάδες**.** Παρακάτω ακολουθεί ένα παράδειγμα, που δείχνει τον ακριβή τρόπο ομαδοποίησης με weighted binary αριθμούς:



Εικόνα 4.1 : Παράδειγμα διευθυνσιοδότησης με weighted binary αριθμούς

4.3.2) 1^{ος} ευρυστικός αλγόριθμος ομαδοποίησης

Η πρώτη ευρυστική μέθοδος ομαδοποίησης υλοποιήθηκε με στόχο να βρεθεί μέσω των bits των “σημαδιών” των κανόνων Golomb, που ανήκουν σε κάθε cluster, κάποια κωδικοποίηση έτσι ώστε να υπάρχει κάποιο πλεονέκτημα στην χρήση για διευθυνσιοδότηση των κόμβων ανάμεσα στους κανόνες Golomb έναντι των weighted binary αριθμών. Στην συγκεκριμένη μέθοδο επειδή υπήρξε ενδιαφέρον και για την ομοιόμορφη κατανομή των “σημαδιών” στα clusters, χρησιμοποιήθηκε ο αλγόριθμος, bisecting k-means, ο οποίος σύμφωνα με τα αποτελέσματα δίνει τα καλύτερα αποτελέσματα ως προς την ομοιομορφία σε μέγεθος των ομάδων. Η περιγραφή των βημάτων των αλγορίθμων που υλοποιήθηκε ήταν η εξής:

1. Επιλογή του κανόνα Golomb στον οποίο θα γίνει η ομαδοποίηση και κατασκευή του κανόνα με τον αλγόριθμο του κ. Δημητρομανωλάκη [1].
2. Εφαρμογή του αλγόριθμου bisecting k-means, και δημιουργία N ομάδων δεδομένων, όπου ο αριθμός N καθορίζεται από τον χρήστη.
3. Μετά την εφαρμογή του αλγορίθμου ακολουθεί η κατασκευή ομάδων, με τέτοιο τρόπο ώστε να έχουν τον ίδιο αριθμό από κόμβους (αφαιρούνται από τα μεγαλύτερα clusters οι κόμβοι που περισσεύουν έναντι της μικρότερης ομάδας).
4. Μετατροπή όλων των “σημαδιών” του κανόνα Golomb που θα χρησιμοποιηθούν στις ομάδες σε δυαδική μορφή.

5. Εύρεση του ελάχιστου αριθμού των bits, έτσι ώστε αν κρατηθεί από την δυαδική μορφή του κάθε "σημαδιού" μόνο αυτά τα bits, τότε οι νέοι αριθμοί που δημιουργούνται είναι μοναδικοί. Δηλαδή, αν υπάρχει η εξής σειρά αριθμών : 0(000), 1(001), 3(011), 6(110) τότε για αυτά τα νούμερα αρκούν μόνο τα δύο bits έτσι ώστε και οι νέοι αριθμοί που προκύπτουν να είναι μοναδικοί μεταξύ τους.
6. Στην συνέχεια για κάθε cluster εφαρμόζω τον αλγόριθμο Quine – McCluskey χρησιμοποιώντας τα νέα νούμερα, που προέκυψαν, ξεχωριστά για κάθε ομάδα. Αυτό που προκύπτει είναι τουλάχιστον μία μάσκα από bits, η οποία περιλαμβάνει μεταβλητό αριθμό bits και τα οποία καθορίζονται από τις διευθύνσεις των κόμβων που ανήκουν στο ομάδα.

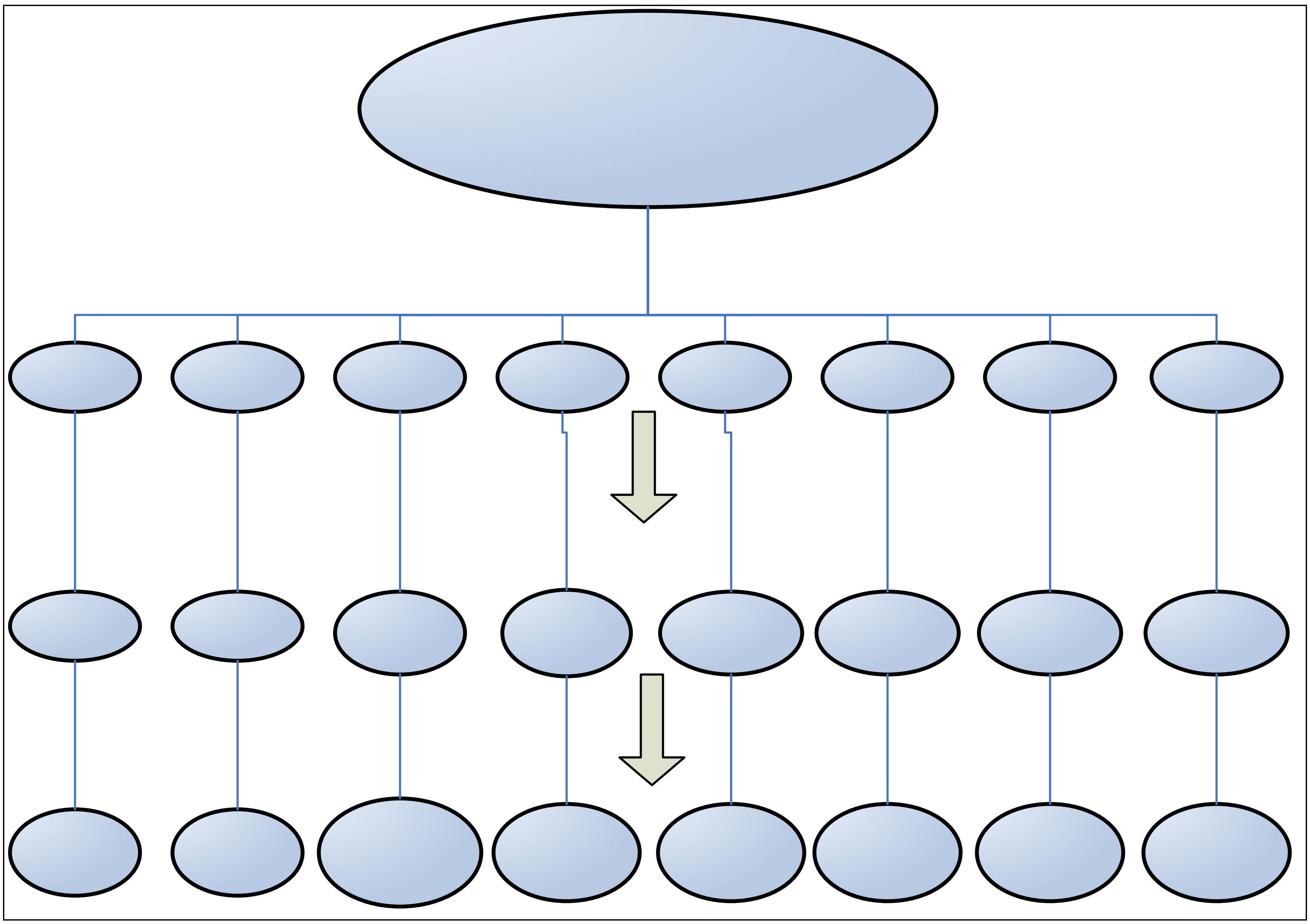
Παρατηρήσεις:

- Ο παραπάνω αλγόριθμος δίνει ομάδες με ομοιόμορφο αριθμό διευθύνσεων, αφού έτσι έχει ορισθεί από την κατασκευή του.
- Ο αλγόριθμος αυτός χρησιμοποιήθηκε με σκοπό για κάθε ομάδα να δημιουργείται τουλάχιστον μία μάσκα από bits, με όσο το δυνατόν λιγότερο αριθμό από bits, και η οποία έχει ορισμένα χαρακτηριστικά:

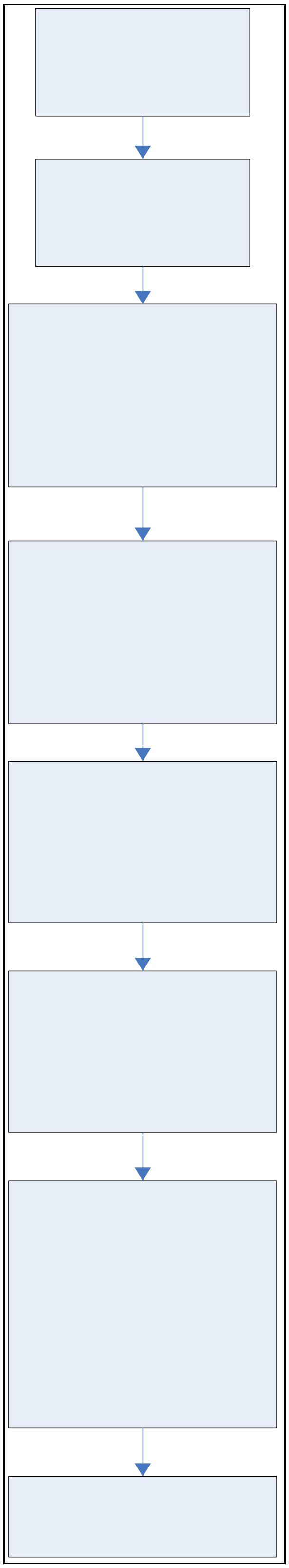
    1. Η μάσκα αυτή των bits προκύπτει από τις διευθύνσεις, και πιο συγκεκριμένα από την δυαδική αναπαράσταση των διευθύνσεων, που ανήκουν σε κάθε ομάδα δεδομένων.
    2. Η μάσκα αποτελείται από ένα μικρό αριθμό bits της διεύθυνσης κάθε κόμβου, τα οποία προκύπτουν από την διαδικασία που περιγράφτηκε παραπάνω και κυρίως από το αποτέλεσμα του αλγορίθμου Quine – McCluskey.
    3. Η μάσκα των bits είναι μοναδική για κάθε ομάδα, με αποτέλεσμα από κάθε μάσκα να προκύπτει μία λογική συνάρτηση στην οποία όταν δοθούν ως είσοδο τα x πρώτα bits της διεύθυνσης του κόμβου, στον οποίο το πακέτο έφτασε, αν από την λογική συνάρτηση αυτή έχουμε αποτέλεσμα 1 τότε ο κόμβος μπορεί να γνωρίζει ότι το πακέτο απευθύνεται στην δική του ομάδα, διαφορετικά ο κόμβος αγνοεί το

περιεχόμενο του πακέτου αφού δεν αφορά την ομάδα των κόμβων στην οποίο ανήκει.

Παρακάτω παρουσιάζεται σε μορφή ψευδογλώσσας ο κώδικας του αλγόριθμου έτσι όπως υλοποιήθηκε στην Matlab καθώς και ένα παράδειγμα υλοποίησης του παραπάνω αλγορίθμου για N = 100 “σημάδια” Golomb(**Εικόνα 4.2**).



Εικόνα 4.2 : Παράδειγμα χρήσης του 1^ου^ ευρυστικού αλγορίθμου



Εικόνα 4.3 : Διάγραμμα ροής του 1ου ευρυστικού αλγορίθμου

Πρωτόκολλο Επικοινωνίας

Όπως φαίνεται και στο παραπάνω παράδειγμα ο αλγόριθμος αυτός καταλήγει σε μία συνάρτηση, η οποία είναι μοναδική για κάθε ομάδα. Αφού, λοιπόν, αρχικά κατασκευαστεί ο κανόνας Golomb και ακολουθήσει η διαδικασία που εμφανίζεται στο παραπάνω παράδειγμα, τελικά καταλήγουμε σε μία συνάρτηση, η οποία καθορίζει εάν ένα πακέτο απευθύνεται σε μία συγκεκριμένη ομάδα ή όχι.

Η βασική ιδέα για το πρωτόκολλο επικοινωνίας για τον συγκεκριμένο αλγόριθμο είχε ως εξής: Αρχικά δίνεται σε κάθε κόμβο από μία διεύθυνση και η λογική συνάρτηση που καθορίζει την ομάδα του. Ας θεωρήσουμε το γεγονός ότι ένας κόμβος της πρώτης ομάδας, από το παράδειγμα, θέλει να στείλει ένα πακέτο πληροφορίας σε όλους τους κόμβους της ίδιας του της ομάδας. Τότε ο κόμβος αυτός στέλνει τα 9 most significant bits **της ίδιας του της διεύθυνσης** ως διεύθυνση στην επικεφαλίδα του πακέτου. Στην συνέχεια σε κάθε κόμβο που θα φτάνει το πακέτο πληροφορίας θα εφαρμόζεται η λογική συνάρτηση της ομάδας του κόμβου και αν το αποτέλεσμα είναι 1 τότε αυτό σημαίνει ότι ο κόμβος αυτός ανήκει στην 1 ομάδα άρα το πακέτο αναφέρεται και σε αυτόν, διαφορετικά αν το αποτέλεσμα της λογικής συνάρτησης είναι 0 τότε το πακέτο δεν αναφέρεται στην ομάδα του συγκεκριμένου κόμβου.

Αν κάποιος κόμβος, που ανήκει στην ομάδα 2 θέλει να στείλει ένα πακέτο πληροφορίας στην ομάδα 1 τότε αρκεί να στείλει ως διεύθυνση μία ποσότητα 9 bits, η οποία να έχει την μορφή 0 0 0 0 X X X X X, όπου X τιμή αδιαφορίας. Εφαρμόζοντας κάθε κόμβος την λογική συνάρτηση της ομάδας, στην οποία ανήκει, πάνω στην ποσότητα παραπάνω οδηγεί στην "κατανόηση" αν το πακέτο με αυτήν την επικεφαλίδα απευθύνεται στην δική του ομάδα ή όχι.

4.3.3) 2ος ευρυστικός αλγόριθμος ομαδοποίησης – Μέθοδος "δένδρου"

Στον δεύτερο ευρυστικό αλγόριθμο που υλοποιήθηκε δόθηκε μεγαλύτερη βαρύτητα στην προσπάθεια με έναν ορισμένο αριθμό bits από τις διευθύνσεις που δίνονται στους κόμβους να προκύψει όσο το δυνατόν μεγαλύτερος αριθμός από ομαδοποιήσεις. Στόχος αυτής της μεθόδου είναι η δημιουργία ομαδοποιήσεων με έναν αριθμό bits όμοιο με αυτό τον συμβατών αριθμών ή έστω με λίγο μεγαλύτερο αλλά με την διαφορά ότι οι αριθμοί Golomb θα υπερτερούσαν στον αριθμό των ομαδοποιήσεων. Παρακάτω περιγράφονται τα βήματα της μεθόδου που υλοποιήθηκαν στον συγκεκριμένο ευρυστικό αλγόριθμο:

1. Επιλογή του κανόνα Golomb στον οποίο θα γίνει η ομαδοποίηση και κατασκευή του κανόνα με τον αλγόριθμο του κ. Δημητρομανωλάκη [1].
2. Εισαγωγή των διευθύνσεων των κόμβων και μετατροπή τους σε δυαδική μορφή.
3. Εύρεση με "εξαντλητική έρευνα" του ενός bit ή των δύο bits που με βάση τις τιμές τους χωρίζουν τα δεδομένα σε δύο ομάδες με ίσο αριθμό διευθύνσεων σε κάθε μία. Πρέπει να αναφερθεί ότι στην περίπτωση των δύο bits για την μία ομάδα οι τιμές των δύο αυτών bits ήταν όμοιες , δηλαδή είτε 00 είτε 11, ενώ για την άλλη ομάδα οι τιμές των bits ήταν διαφορετικές, δηλαδή είτε 01 είτε 10.
4. Για κάθε bit ή bits που βρέθηκαν στο προηγούμενο βήμα δημιουργούνται οι δύο ομάδες με βάση διάσπασης τις τιμές των συγκεκριμένων bits κάθε φορά και στην συνέχεια πάλι με "εξαντλητική έρευνα" βρίσκονται, αν υπάρχουν, είτε ένα bit είτε δυάδα bits τα οποία να σπάνε κάθε νέα ομάδα πάλι σε δύο ισοδύναμες, ως

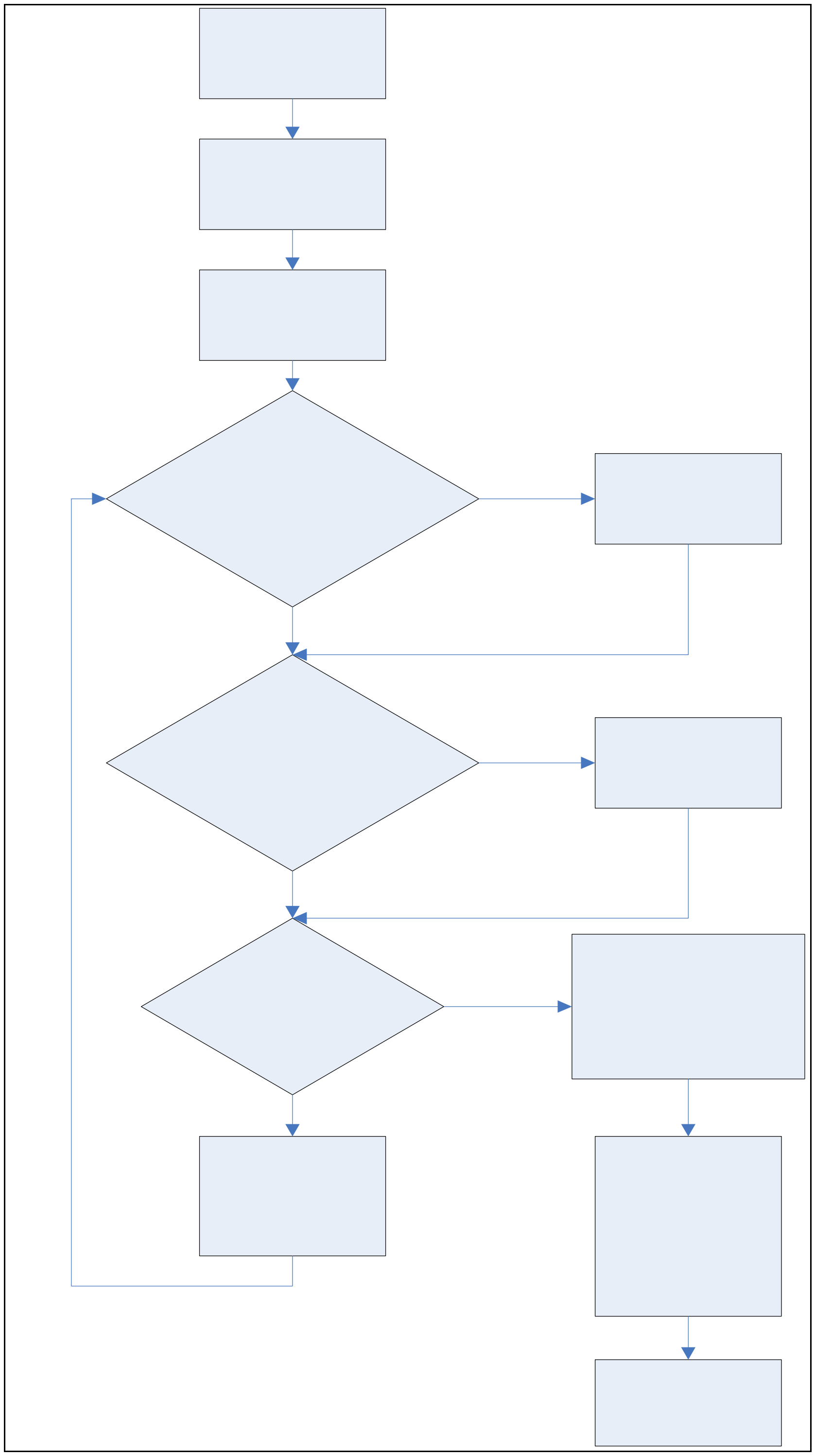
προς τον αριθμό των διευθύνσεων που ανήκουν σε κάθε μία. Μετά το τέλος αυτού του βήματος έχουμε ένα συνδυασμό από τρία μέχρι έξι bits, που να σπάνε τα δεδομένα σε 4 ισοδύναμες ομάδες.

5. Αν θέλουμε να σπάσουμε τον αρχικό αριθμό διευθύνσεων σε μεγαλύτερο αριθμό ομάδων, πχ σε 8 ομάδες, τότε βρίσκουμε την κάθε μία ομάδα από τις τιμές των bits, από τις οποίες καθορίζεται, και στην συνέχεια πάλι με "εξαντλητική έρευνα" γίνεται η εύρεση των bit ή bits τα οποία "σπάνε" κάθε ομάδα από τις τέσσερις σε δύο "ισοδύναμες" ομάδες. Έτσι τελικά προκύπτει μία μάσκα από τρία έως τέσσερα bits τα οποία ανάλογα με την τιμή τους καθορίζουν κάθε μία από τις οκτώ διαφορετικές ομάδες.

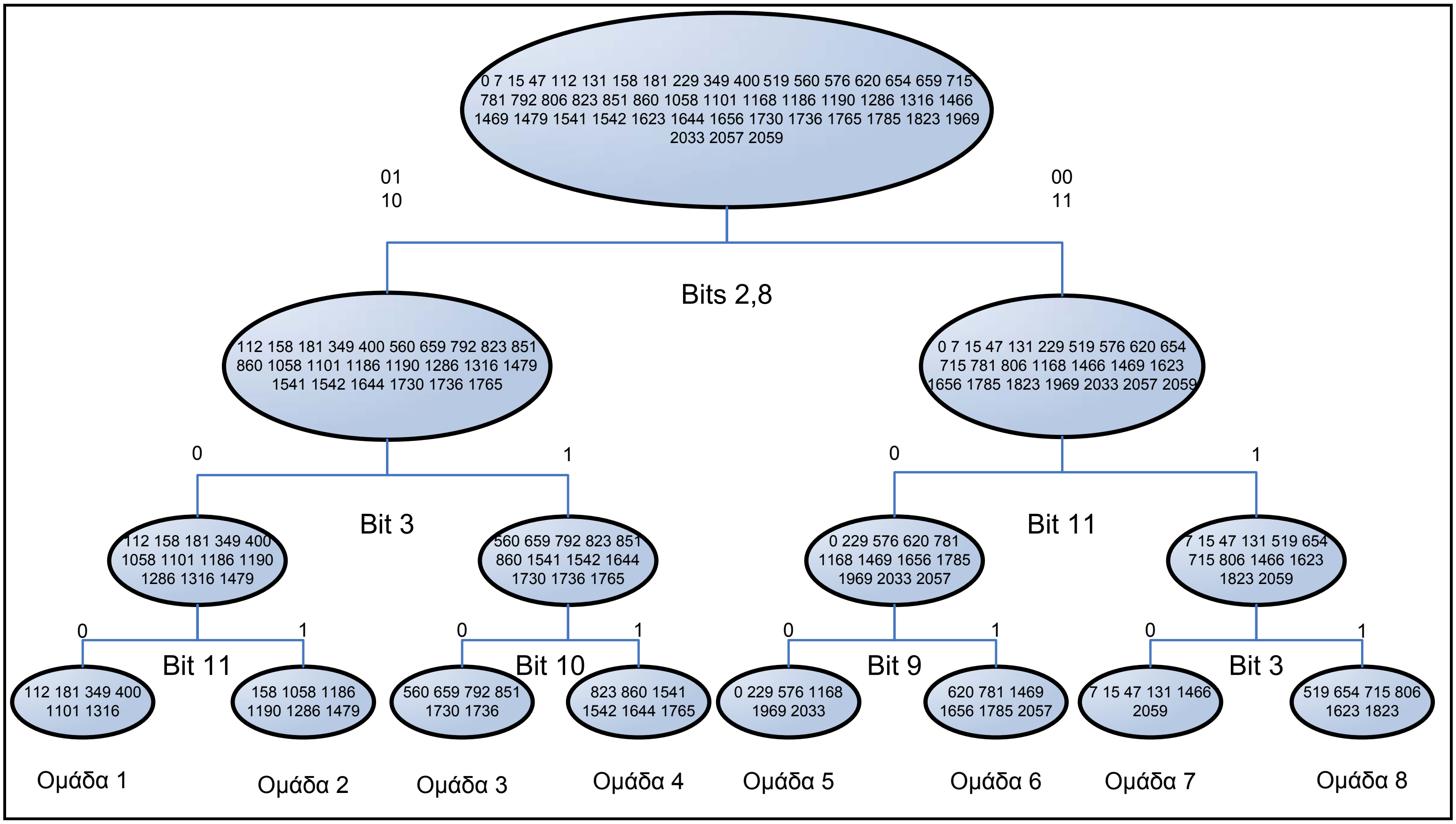
Παρατηρήσεις:

- Ο παραπάνω αλγόριθμος από την κατασκευή του δίνει ισοδύναμες , ως προς το αριθμό τους, ομάδες από διευθύνσεις.
- Οι διασπάσεις των διευθύνσεων μπορούν να παρουσιαστούν σε μορφή δυαδικού δένδρου.
- Η διάσπαση μίας ομάδας σε δύο νέες ομάδες γίνεται πάντα ανάλογα με την τιμή ενός ή δύο bits από την δυαδική αναπαράσταση των διευθύνσεων που ανήκουν στην ομάδα.
- Ο αριθμός των διασπάσεων καθορίζεται από τον επιθυμητό αριθμό των ομάδων στο τέλος της διαδικασίας.
- Γενικά η διάσπαση των ομάδων με βάση την τιμή των bits παρουσιάζεται με την μορφή δένδρου, όπως στο παράδειγμα που ακολουθεί.
- Πρέπει να σημειωθεί ότι, πχ για την διάσπαση των διευθύνσεων σε οκτώ ισοδύναμες ομάδες, κρατήθηκαν μάσκες bits, οι οποίες αποτελούνταν από 4-7 bits, με στόχο ο αριθμός των bits που χρησιμοποιούνται να είναι όσο το δυνατόν πιο κοντά στον αριθμό των συμβατών αριθμών.

Παρακάτω στην **Εικόνα 4.5** περιγράφεται όλη η διαδικασία που ακολουθείται από τον αλγόριθμο με την βοήθεια ενός παραδείγματος, για N = 48 αριθμό διευθύνσεων, και στην συνέχεια ακολουθεί σε μορφή ψευδοκώδικα τα βήματα του αλγορίθμου, όπως αυτός υλοποιήθηκε στο πρόγραμμα της Matlab(**Εικόνα 4.4**).



Εικόνα 4.4 : Διάγραμμα ροής του 2ου ευρυστικού αλγορίθμου



Εικόνα 4.5 : Παράδειγμα χρήσης του 2ου ευρυστικού αλγορίθμου

Στην παραπάνω εικόνα περιγράφεται ένα πλήρες παράδειγμα ομαδοποίησης 48 διευθύνσεων, αρχικά, σε 8 ισοδύναμες ομάδες, όπως προέκυψε από την εφαρμογή του αλγορίθμου που περιγράφηκε παραπάνω. Η διάσπαση των κόμβων έγινε ανάλογα με την τιμή κάποιων bits στην δυαδική αναπαράσταση των διευθύνσεων. Εφαρμόζοντας τον συγκεκριμένο αλγόριθμο στο αρχικό σύνολο αριθμών Golomb, που αποτελούν και τις διευθύνσεις των κόμβων, πήραμε ως αποτέλεσμα 4 διαφορετικούς τρόπους ομαδοποιήσεων των κόμβων. Ο ένας από αυτούς παρουσιάζεται στην παραπάνω εικόνα. Με αυτό τον τρόπο, για την συγκεκριμένη ομαδοποίηση είναι γνωστό ότι όλοι οι κόμβοι που ανήκουν , πχ στην ομάδα 8 θα μπορούν να λάβουν κάποιο πακέτο πληροφορίας με την χρήση μόνο 4 bits. Φυσικά οι άλλες 3 διαφορετικές ομαδοποιήσεις αναδιατάσσουν τις διευθύνσεις στις ομάδες δημιουργώντας ανάλογα με την ομαδοποίηση και διαφορετικές ομάδες. Γενικά για την χρήση του αλγορίθμου αυτού υπήρχαν δύο διαφορετικά πρωτόκολλα επικοινωνίας μεταξύ των κόμβων τα οποία μπορούν να χρησιμοποιηθούν και τα οποία παρουσιάζονται στις επόμενες παραγράφους.

1º Πρωτόκολλο Επικοινωνίας:

Η ιδέα που δημιουργήθηκε από την χρήση του παραπάνω αλγόριθμου ήταν η δημιουργία ενός πρωτοκόλλου επικοινωνίας στο οποίο γίνεται χρήση ορισμένων bits των διευθύνσεων (αριθμοί Golomb) των κόμβων, για την ομαδοποίηση τους σε ομάδες. Στη συνέχεια, με την χρήση των ίδιων bits να μπορεί ένας κόμβος να καταλάβει αν ένα πακέτο απευθύνεται στην δική του ομάδα ή όχι. Όπως αναφέρθηκε παραπάνω με την χρήση του παραπάνω αλγορίθμου από τις διευθύνσεις προκύπτουν διάφορες ομαδοποιήσεις. Οι ομαδοποιήσεις, πχ για 8 διαφορετικές ομάδες όπως στο παράδειγμα, όπου υπάρχουν αρχικά 48 διευθύνσεις, υπάρχουν 4 διαφορετικές ομαδοποιήσεις, διαφέρουν ανάλογα με τον αριθμό των διευθύνσεων αρχικά.

Η βασική ιδέα χρήσης του αλγόριθμου και του πρωτοκόλλου είναι η εξής:

Αφού αρχικά κατασκευαστεί ο επιθυμητός κανόνας Golomb, με τον αλγόριθμο της διπλωματικής εργασίας του κ. Δημητρομανωλάκη [1], στην συνέχεια εφαρμόζεται στους αριθμούς αυτούς ο αλγόριθμος που περιγράφηκε παραπάνω. Το αποτέλεσμα του αλγορίθμου θα είναι ένας αριθμός ομαδοποιήσεων, οι οποίες θα στηρίζονται πάνω στην δυαδική αναπαράσταση των αριθμών Golomb και πιο συγκεκριμένα στις τιμές ορισμένων bits τους. Έστω ότι κάθε ομαδοποίηση γίνεται σε 8 διαφορετικές ομάδες από διευθύνσεις, όπως στο παράδειγμα. Σε μία τέτοια ομαδοποίηση, όπως και στο παράδειγμα, κάθε ομάδα καθορίζεται από τις τιμές 3-4 bits.

Παρατηρώντας την εικόνα του παραδείγματος, για να στείλει κάποιος κόμβος κάποια πληροφορία σε όλους τους κόμβους της ομάδας 1 θα πρέπει στην επικεφαλίδα του πακέτου, αντί για την διεύθυνση του παραλήπτη θα υπάρχει ένας αριθμός 12 bits, ο οποίος θα πρέπει να έχει την μορφή X0XX1XXXX00X είτε X0XX0XXXX01X, όπου X κατάσταση αδιαφορίας (στην συγκεκριμένη περίπτωση ο αριθμός διευθυνσιοδότησης είναι 12 bits διότι η μεγαλύτερη διεύθυνση του κανόνα Golomb που χρησιμοποιήθηκε είναι 12 bits, συνεπώς και όλοι οι κόμβοι θα πρέπει να έχουν διευθύνσεις 12 bits). Φυσικά θα πρέπει να προσεχθεί το γεγονός ότι αφού η ποσότητα που θα αποστέλλεται ως διεύθυνση των ομάδων θα είναι **12 bits**, ίση με αυτή της διεύθυνσης των κόμβων, η τιμή στα bits αδιαφορίας να μην συμπίπτει με κάποια διεύθυνση κόμβου είτε με κάποια απόσταση μεταξύ των δύο κόμβων. Επίσης, ένα άλλο σημαντικό σημείο που θα πρέπει να προσεχθεί είναι ότι επειδή για μία ομαδοποίηση ένας κόμβος μπορεί να ανήκει, πχ στην ομάδα 1 ενώ σε μία άλλη ομαδοποίηση να ανήκει στην ομάδα 5, θα πρέπει με κάποιο τρόπο πάνω στην επικεφαλίδα του πακέτου να γίνεται σαφές σε ποια κωδικοποίηση αναφέρεται η διευθυνσιοδότηση της ομάδας.

Πρέπει τέλος να αναφερθεί ότι στην 12 bit ποσότητα υπάρχουν κάποια bits για τα οποία δεν μας απασχολεί η τιμή τους και αυτό δεν πρόκειται να δημιουργήσει ποτέ κάποια σύγχυση στην συγκεκριμένη κάθε φορά ομαδοποίηση ανάμεσα διότι αυτό καλύπτεται από τον τρόπο με το οποίο βγάζει τα αποτελέσματα ο αλγόριθμος που κατασκευάστηκε και δημιουργεί τις ομαδοποιήσεις. Πιο συγκεκριμένα στον υλοποιημένο αλγόριθμο όταν δημιουργείται μία ομαδοποίηση, αυτή κατασκευάζεται σε μορφή δυαδικού δένδρου, κατά συνέπεια οι τιμές των bits καθορίζουν μοναδικά την κάθε ομάδα. Για παράδειγμα, ένα πακέτο που προορίζεται για την ομάδα 2 δεν πρόκειται να γίνει αποδεκτό από την ομάδα 7 (επειδή έχουν την ίδια τιμή για τα bits 3 και 11) λόγω του γεγονότος ότι τα bits 2 και 8 θα είναι όμοια (00 ή 11) και όχι διαφορετικά όπως στην περίπτωση που το πακέτο προοριζόταν για την ομάδα 7.

<u>2^{o} Πρωτόκολλο Επικοινωνίας</u>:

Πάνω στον ίδιο αλγόριθμο υπήρξε η ιδέα ότι θα μπορούσαμε χρησιμοποιήσουμε ένα άλλο πρωτόκολλο, το οποίο ίσως να έδινε και καλύτερα αποτελέσματα από το πρώτο. Η ιδέα για το δεύτερο πρωτόκολλο επικοινωνίας ήταν παρόμοια με το πρώτο πρωτόκολλο με την μόνη διαφορά στον αριθμό των bits, που θα χρησιμοποιούνταν στην επικεφαλίδα του πακέτου και θα καθόριζαν σε ποια ομάδα αντιστοιχεί το συγκεκριμένο πακέτο.

Πιο συγκεκριμένα, σε αυτό το πρωτόκολλο επικοινωνίας η διεύθυνση της ομάδας στην οποία θα άνηκε κάθε πακέτο δεν θα αποτελούνταν από τον αριθμό των bits της μεγαλύτερης διεύθυνσης των κόμβων αλλά από τον συνολικό αριθμό των bits που χρησιμοποιούνται για τον καθορισμό όλων των ομάδων της συγκεκριμένης κάθε φορά ομαδοποίησης. Πιο συγκεκριμένα για το παράδειγμα που περιγράφεται παραπάνω στην **Εικόνα 4.5**, για να σταλεί κάποιο πακέτο στην ομάδα 1 θα πρέπει στην επικεφαλίδα του πακέτου ως διεύθυνση παραλήπτη να μπουν **6 bits**, αφού τα bits που χρησιμοποιούνται για την συγκεκριμένη ομαδοποίηση είναι τα bits 11,10,9,8,3,2 των διευθύνσεων, τα οποία θα πρέπει να έχουν την εξής τιμή 0XX100 είτε 0XX001, όπου Χ κατάσταση αδιαφορίας.

Φυσικά, επειδή στην συγκεκριμένη περίπτωση έχουμε 4 διαφορετικές ομαδοποιήσεις θα πρέπει με κάποιο τρόπο να αναφερθεί σε ποια ομαδοποίηση αναφέρονται αυτά τα bits κάθε φορά. Για τον λόγο αυτό μπροστά από τα 6 αυτά bits θα πρέπει να υπάρχουν άλλα δύο bits που θα αναφέρονται στην ομαδοποίηση. Με αυτό τον τρόπο θα πρέπει για αποστολή πακέτου στην πρώτη ομάδα(θεωρώντας ότι την κωδικοποίηση που περιγράφεται παραπάνω την αναφέρουμε ως πρώτη κωδικοποίηση) η επικεφαλίδα του πακέτου θα πρέπει να περιλαμβάνει ως διεύθυνση τα εξής 8 bits, είτε 00 0XX100 είτε 00 0XX001, όπου Χ τιμή αδιαφορίας.

4.3.4) 3^{ος} ευρυστικός αλγόριθμος clustering – Μέθοδος “αρχηγών”

Σε αυτή τη μέθοδο ακολουθήθηκε μία διαφορετική τακτική από τις δύο προηγούμενες μεθόδους. Σημαντικό ρόλο σ’αυτόν τον ευρυστικό αλγόριθμο παίζουν οι λεγόμενοι αρχηγοί των ομάδων. Η ιδέα του αλγορίθμου αυτού έχει να κάνει με την προσπάθεια να ελαχιστοποιήσουμε τον αριθμό των bits που θα στέλνονται ως επικεφαλίδα σε κάθε πακέτο, το οποίο θα έχει ως προορισμό μία ολόκληρη ομάδα κόμβων, χρησιμοποιώντας μόνο τις διευθύνσεις ορισμένων κόμβων των ομάδων. Συγκεκριμένα οι κόμβοι αυτοί θα ονομάζονται αρχηγοί των ομάδων και ο τρόπος εύρεσης τους που περιγράφεται παρακάτω έχει ως στόχο όσο τον δυνατόν την μείωση των αποστάσεων ανάμεσα στους αρχηγούς. Τα βήματα του συγκεκριμένου αλγορίθμου περιγράφονται παρακάτω:

1. Γνώση του συνολικού αριθμού των κόμβων του δικτύου και εξαγωγή του κανόνα Golomb για αυτό τον αριθμό “σημαδιών”
2. Επιλογή του αριθμού των ομάδων που θα πρέπει να διασπαστούν οι διευθύνσεις-“σημάδια” του κανόνα Golomb.
3. Ομαδοποίηση των διευθύνσεων με όλους τους πιθανούς τρόπους σε ομάδες με μέγεθος όσο ο αριθμός των ομάδων, που επιλέχθηκε στο βήμα 2. Επειδή, το

άθροισμα των αποστάσεων στο εσωτερικό των ομάδων που δεν περιέχουν συνεχόμενα "σημάδια" του κανόνα Golomb είναι μεγαλύτερο από ότι αυτές που έχουν, απορρίπτονται όλες οι ομάδες με τα μη συνεχόμενα "σημάδια" του κανόνα Golomb. Κάθε "σημάδι" μίας ομάδας από τις παραπάνω θα είναι ο αρχηγός κάθε ομάδας των "σημαδιών", που θα δημιουργηθούν.

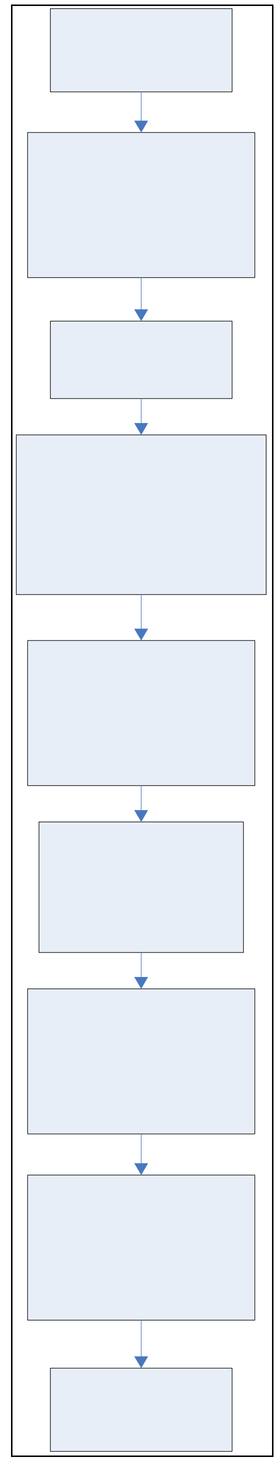
4. Εύρεση όλων των δυνατών αποστάσεων ανάμεσα στα σημάδια που ανήκουν στην ίδια ομάδα. Η διαδικασία επαναλαμβάνεται για όλες τις ομάδες που βρέθηκαν στο βήμα 3.
5. Μετατροπή των αποστάσεων στο εσωτερικό κάθε ομάδας σε δυαδική αναπαράσταση.
6. Υπολογισμός του μέσου όρου των bits που μπορούν να αναπαρασταθούν οι αποστάσεις στο εσωτερικό της κάθε ομάδας.
7. Εύρεση της ομάδας, η οποία χρησιμοποιεί τον μικρότερο αριθμό bits για τις αποστάσεις μεταξύ των διευθύνσεων που περιλαμβάνει.
8. Διαχωρισμός των κόμβων σε ισοδύναμες ομάδες και χρήση κάθε διεύθυνσης της ομάδας, που βρέθηκε στο βήμα 7, ως ενιαία διεύθυνση κάθε ομάδας κόμβων.

Παρατηρήσεις:

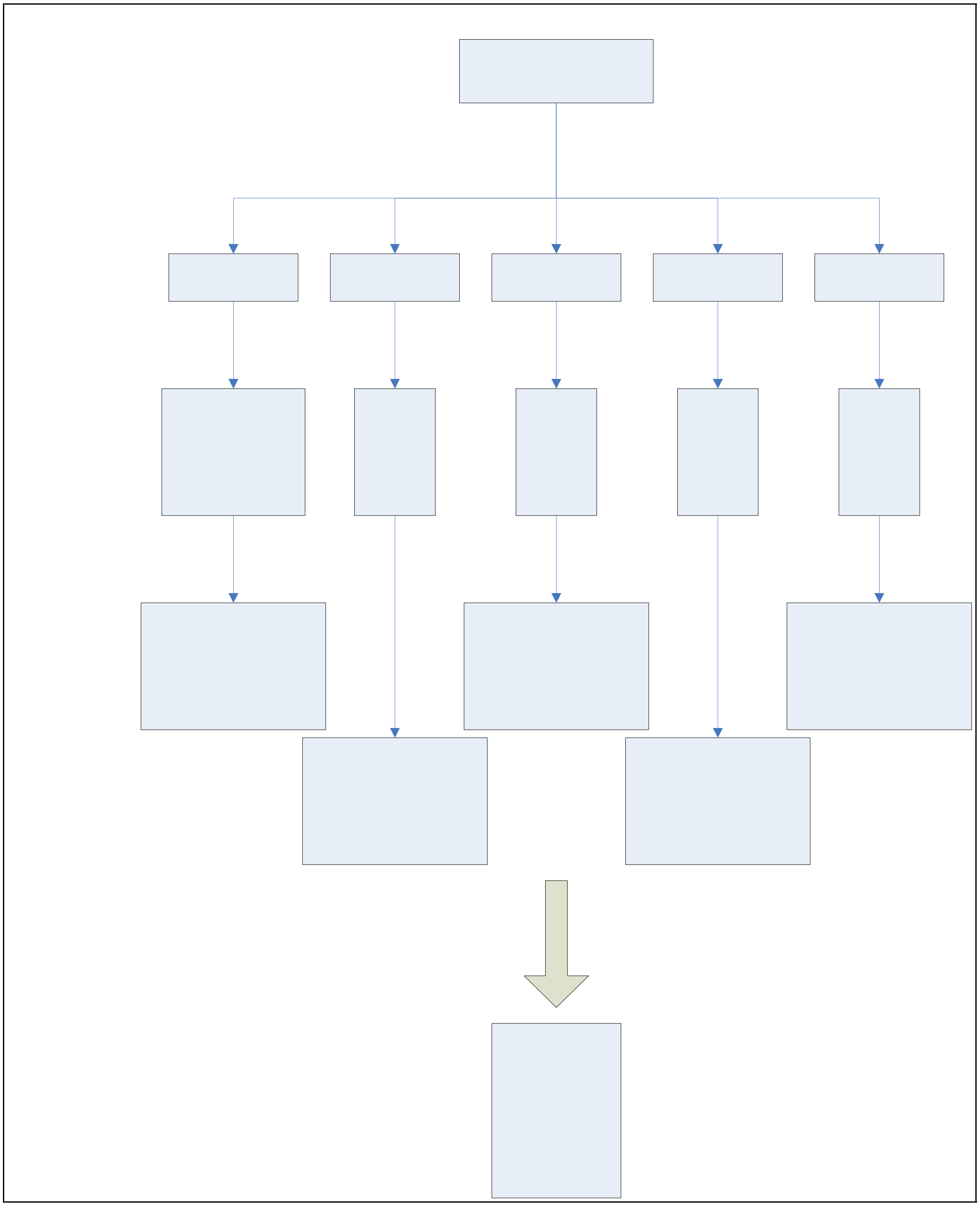
- Σε αυτόν τον αλγόριθμο δεν λαμβάνονται υπ'όψιν οι διευθύνσεις όλων των κόμβων αλλά μόνο οι διευθύνσεις που δίνονται ενιαία στην ομάδα.
- Επίσης, για να επικοινωνήσει ένας κόμβος της μίας ομάδας με όλους τους κόμβους μίας άλλης ομάδας αρκεί να στείλει ως διεύθυνση την διαφορά ανάμεσα στις διευθύνσεις των ομάδων, η οποία απόσταση αφού οι διευθύνσεις προέρχονται και από τους κανόνες Golomb είναι μοναδική.
- Στο βήμα 6 του αλγορίθμου υπολογίζεται ο μέσος όρος των bits των αποστάσεων μεταξύ των διευθύνσεων των ομάδων και ο λόγος είναι η παραδοχή του γεγονότος ότι η κάθε ομάδα να επικοινωνεί ισοπίθανα με κάθε άλλη ομάδα.

- Πρέπει να σημειωθεί ότι στην περίπτωση αυτή οι διευθύνσεις, οι οποίες επιλέγονται, δεν ανήκουν σε κόμβους αλλά μόνο συνολικά στις ομάδες των κόμβων.
- Από την στιγμή που θα γίνει η επιλογή των διευθύνσεων των ομάδων στην συνέχεια δεν ενδιαφέρει ο τρόπος που θα γίνει η κατανομή των κόμβων, διότι οι διευθύνσεις των κόμβων της κάθε ομάδας σε αυτόν τον αλγόριθμο δεν λαμβάνονται υπ'όψιν στην επικοινωνία μεταξύ των ομάδων των κόμβων.

Στην παρακάτω εικόνα (**Εικόνα 4.7**) περιγράφεται ο τρίτος αλγόριθμος με την βοήθεια ενός παραδείγματος για N = 8 σημάδια ενός κανόνα Golomb ενώ στην **Εικόνα 4.6** υπάρχει το διάγραμμα ροής του αλγορίθμου.



Εικόνα 4.6 : Διάγραμμα ροής του 3^{ου} ευρυστικού αλγορίθμου



Εικόνα 4.7 : Παράδειγμα χρήσης του 3ου ευρυστικού αλγορίθμου

ΚΕΦΑΛΑΙΟ 5^{ο}

5.1) Εισαγωγή

Σε αυτό το κεφάλαιο θα παρουσιαστεί μία συγκριτική ανάλυση μεταξύ των διαφόρων μεθόδων που υλοποιήθηκαν και περιγράφηκαν σε προηγούμενα κεφάλαια. Η σύγκριση των μεθόδων αυτών θα γίνει πάνω σε πολλούς και διαφορετικούς τομείς. Στα παραπάνω κεφάλαια περιγράφονται διάφοροι αλγόριθμοι, οι οποίοι αφορούν είτε τους τρόπους ομαδοποιήσης των "σημαδιών" των κανόνων Golomb, είτε τους τρόπους επικοινωνίας των κόμβων.

5.2) Σύγκριση μεθόδων

Αρχικά, θα πρέπει να αναφερθεί ότι υπήρξαν πολλές διαφορετικές μέθοδοι, οι οποίες χρησιμοποιήθηκαν για την ομαδοποίηση των "σημαδιών" των Golomb. Για κάθε μέθοδο χρησιμοποιήθηκε, επίσης, μεγάλος αριθμός παραμέτρων, οι οποίες άλλαζαν ανάλογα τον τρόπο λειτουργίας του αλγορίθμου. Οι παράμετροι αυτοί αναφέρονται πιο αναλυτικά στο Κεφάλαιο 2. Στην παρακάτω ανάλυση που θα ακολουθήσει, έχει γίνει μία γενικότερη έρευνα και ως προς τις τεχνικές και τους αλγορίθμους αλλά και ως προς τις παραμέτρους που κάνουν τον αλγόριθμο να βγάζει καλύτερα αποτελέσματα.

Ένα σημαντικό στοιχείο για την ανάλυση που ακολουθεί είναι ότι η σύγκριση των μεθόδων και των τεχνικών έγινε είτε με την βοήθεια κάποιων συντελεστών, οι οποίοι περιγράφονται στο Κεφάλαιο 2, είτε με βάση το αποτέλεσμα ως προς τον σκοπό της εργασίας αυτής. Συγκεκριμένα, αυτό που ενδιαφέρει στην εργασία αυτή, όπως αναφέρθηκε και στην Εισαγωγή, είναι η εύρεση της μεθόδου-αλγόριθμου, που να δίνει τα καλύτερα αποτελέσματα είτε ως προς την δημιουργία ισοδύναμων ομάδων από τα "σημάδια" του κανόνα Golomb, είτε ως προς την μικρότερη πληροφορία που θα πρέπει να μεταφερθεί στην επικεφαλίδα του κάθε πακέτου όταν σε ένα ασύρματο δίκτυο έχουμε ομαδοποιημένους κόμβους.

Τέλος, πρέπει να σημειωθεί ότι από όλη αυτή την έρευνα προέκυψε, ότι σημαντικό ρόλο παίζει και το μέγεθος αλλά και ο κανόνας Golomb, που χρησιμοποιείται. Γενικά, σε όλα τα πειράματα εφαρμόστηκε μία ενιαία τακτική να χρησιμοποιούνται κανόνες Golomb από 100 έως 2000 "σημάδια" και από τα αποτελέσματα των οποίων, μπορούν να εξαχθούν τα συμπεράσματα που περιγράφονται στο τέλος του Κεφαλαίου.

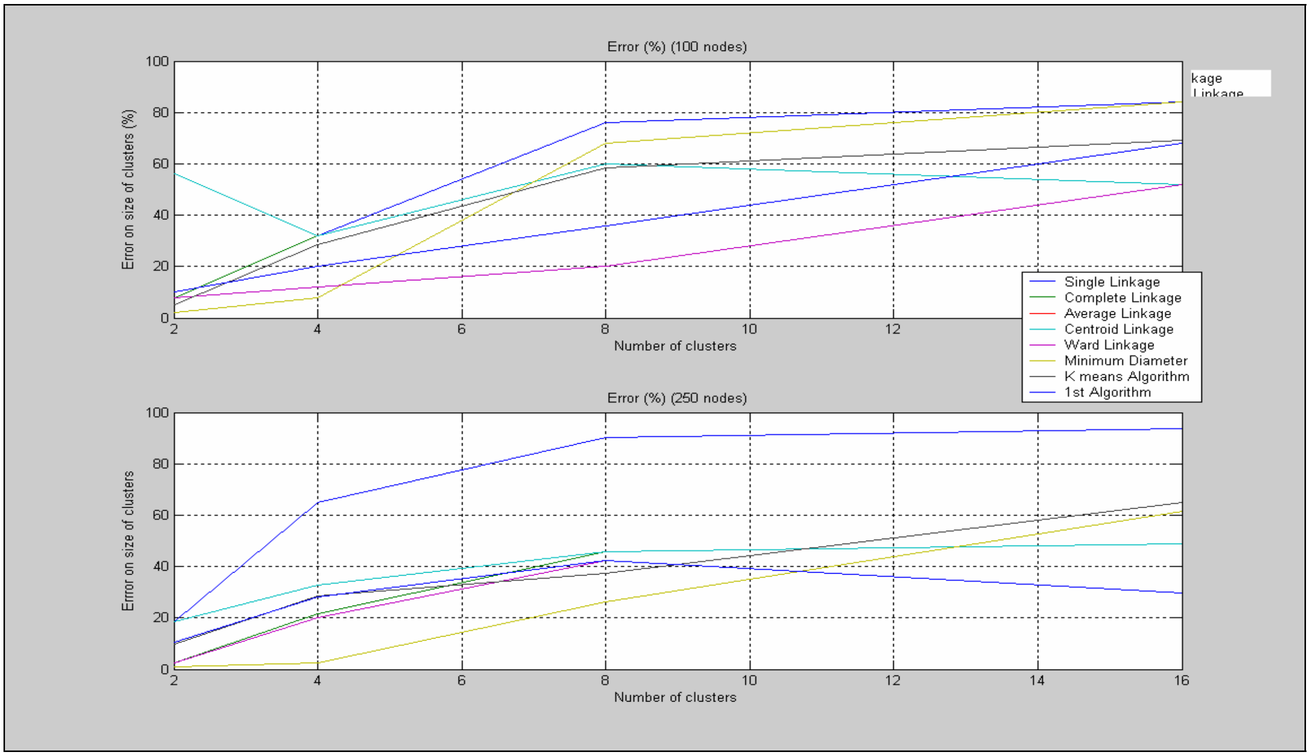
5.2.1) 1^o Κριτήριο σύγκρισης : Ομοιόμορφη κατανομή "σημαδιών" στις ομάδες

Ένα από τα πιο σημαντικά θέματα της εργασίας είναι γενικά η εύρεση ενός αλγορίθμου που να δίνει όσο το δυνατόν πιο ομοιόμορφη κατανομή "σημαδιών" στις ομάδες, ανεξάρτητα από τον αριθμό των "σημαδιών" του κανόνα Golomb και επίσης ανεξάρτητα του αριθμού των ομάδων, στις οποίες διασπούνται τα "σημάδια" του κανόνα Golomb.

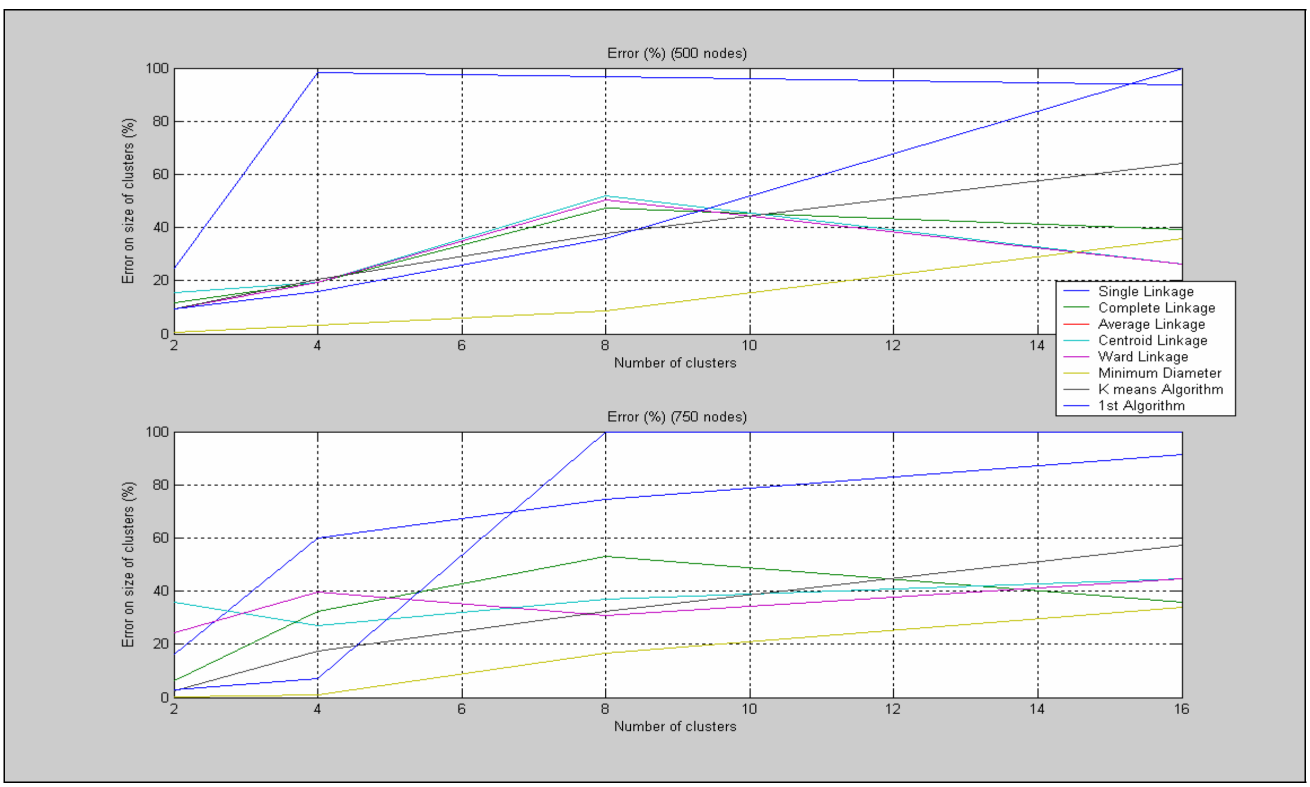
Η σύγκριση ως προς την ομοιόμορφη κατανομή έγινε μόνο ως προς τις μεθόδους partitional, hierarchical και τον 1^o ευρυστικό αλγόριθμο, που υλοποιήθηκε και οι οποίες εφαρμόστηκαν πάνω σε "σημάδια" κανόνων Golomb. Αυτό οφείλεται στο γεγονός ότι οι υπόλοιποι αλγόριθμοι ομαδοποίησης, από τον τρόπο κατασκευής τους δημιουργούν **απόλυτα ομοιόμορφες ομαδοποιήσεις**. Για την σύγκριση των αλγορίθμων ως προς την ομοιομορφία της ομαδοποίησης χρησιμοποιήθηκε μία ποσότητα (σφάλμα %), η οποία αν έχουμε ένα σύνολο από N διευθύνσεις και πρέπει να "σπάσουν" σε X ομάδες τότε ισούται με:

$$\Sigma\varphi\acute{\alpha}\lambda\mu\alpha = \frac{\mathrm{N/X} - \mathrm{min}\_\textit{ομάδα\_του\_αλγόριθμου}}{\mathrm{N/X}} * 100\%$$

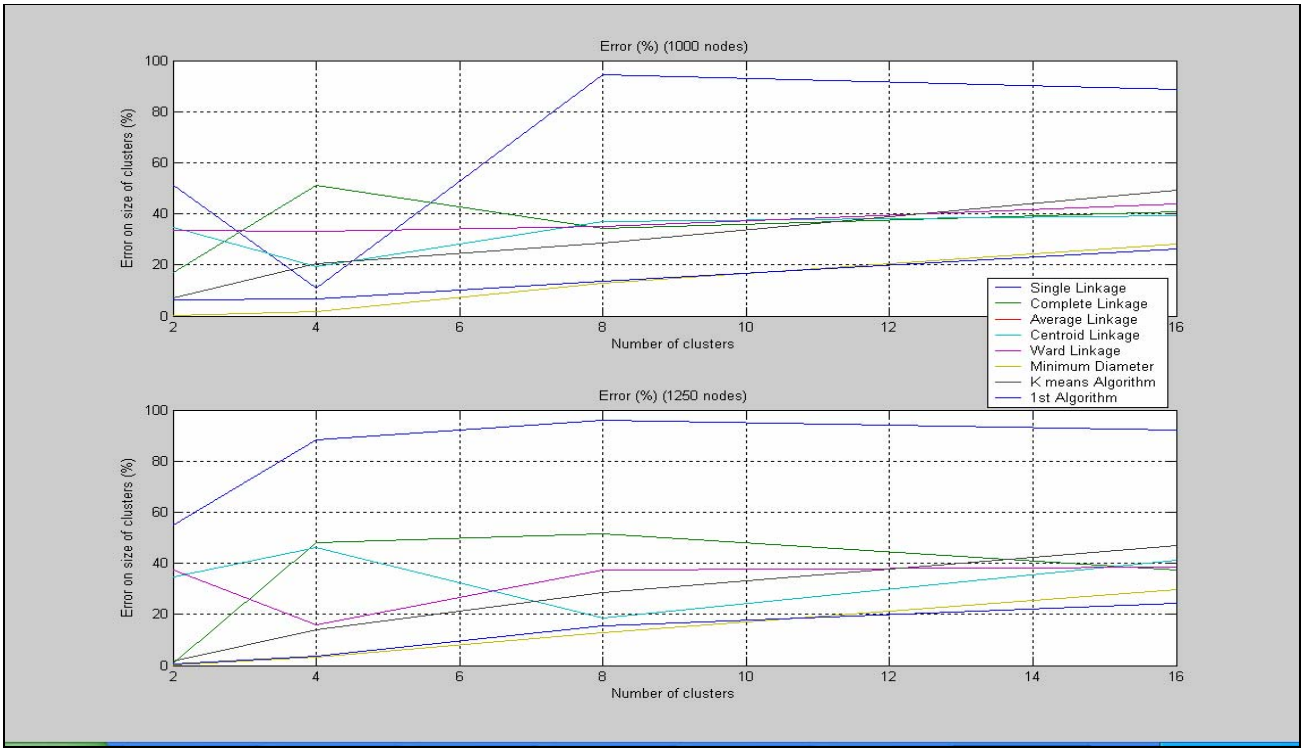
Παρακάτω ακολουθούν γραφικές παραστάσεις του σφάλματος των αλγορίθμων για κάθε διαφορετικό αριθμό ομάδων και διευθύνσεων.



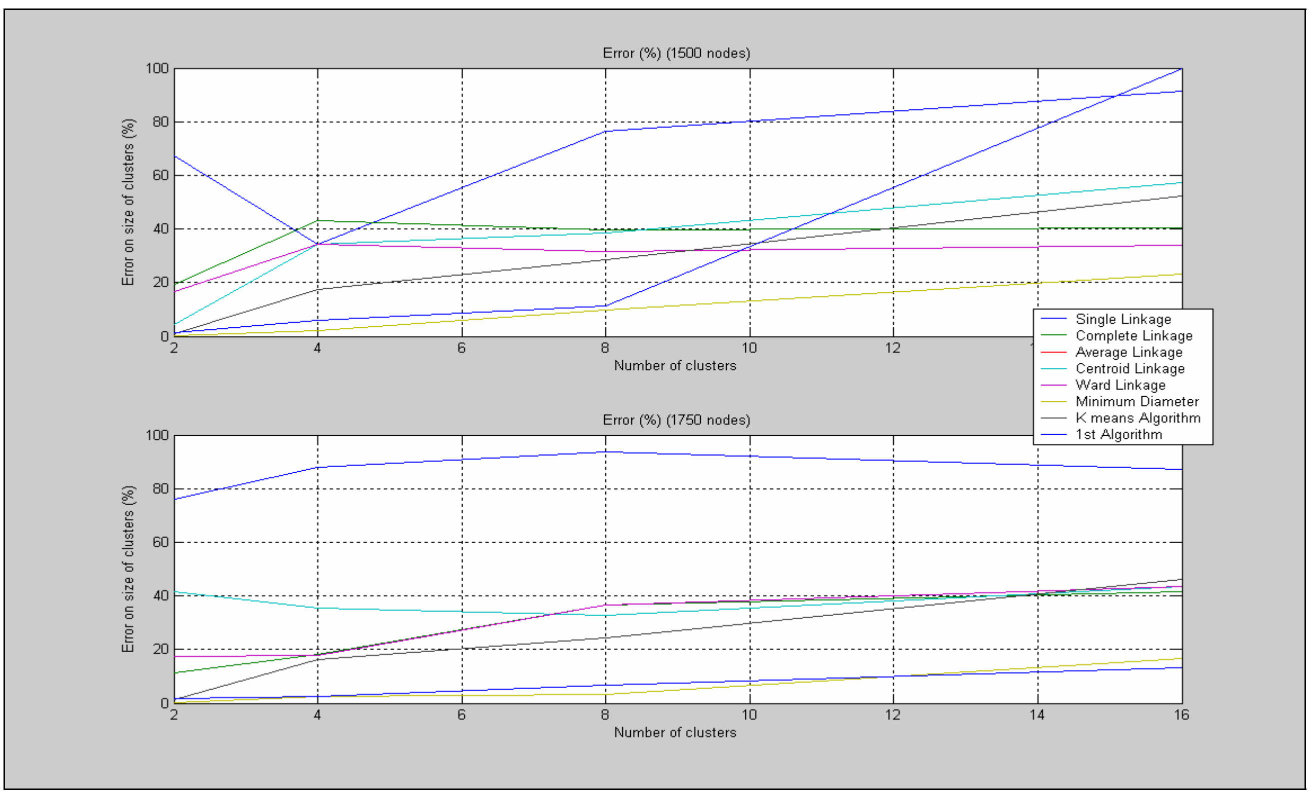
Εικόνα 5.1 : Γραφική παράσταση του σφάλματος των αλγορίθμων ως προς την ομοιόμορφη κατανομή (100 κόμβοι – 250 κόμβοι)



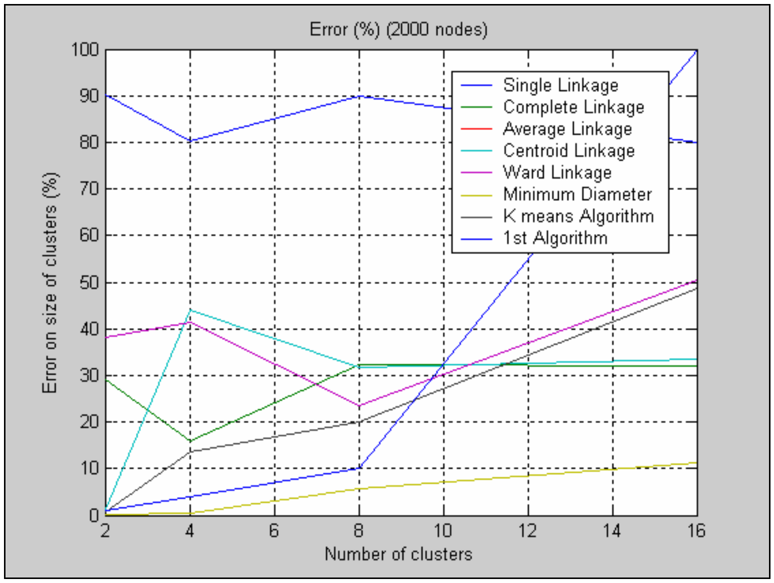
Εικόνα 5.2 : Γραφική παράσταση του σφάλματος των αλγορίθμων ως προς την ομοιόμορφη κατανομή (500 κόμβοι – 750 κόμβοι)



Εικόνα 5.3 : Γραφική παράσταση του σφάλματος των αλγορίθμων ως προς την ομοιόμορφη κατανομή (1000 κόμβοι – 1250 κόμβοι)



Εικόνα 5.4 : Γραφική παράσταση του σφάλματος των αλγορίθμων ως προς την ομοιόμορφη κατανομή (1500 κόμβοι – 1750 κόμβοι)



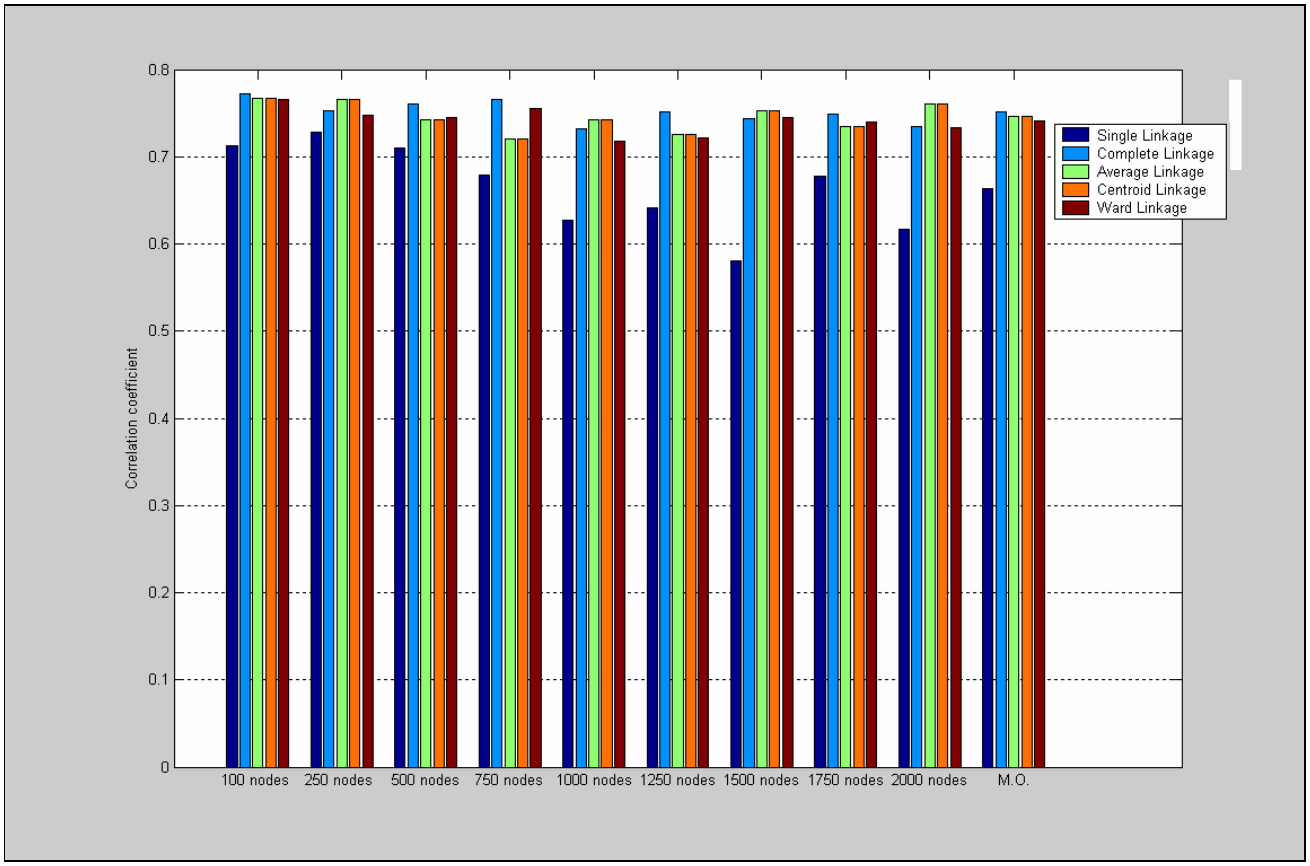
Εικόνα 5.5 : Γραφική παράσταση του σφάλματος των αλγορίθμων ως προς την ομοιόμορφη κατανομή (2000 κόμβοι)

Παρατηρήσεις:

Αυτό που μπορεί να εξαχθεί ως συμπέρασμα, παρατηρώντας τις παραπάνω γραφικές παραστάσεις, είναι ότι η μέθοδος με την **καλύτερη ομοιομορφία (ως προς τον αριθμό των κόμβων σε κάθε ομάδα) είναι η Minimum Diameter** (Μέθοδος Ελάχιστης Διαμέτρου) ενώ αμέσως καλύτερη είναι η μέθοδος **Single Linkage.** Τέλος, πρέπει να παρατηρηθεί ότι η μέθοδος με τα **χειρότερα αποτελέσματα είναι ο 1^{ος} Αλγόριθμος** ενώ οι υπόλοιπες μέθοδοι δίνουν περίπου τα ίδια αποτελέσματα.

5.2.2) 2^{ο} Κριτήριο σύγκρισης : Συντελεστής συσχέτισης

Ο συντελεστή συσχέτισης είναι ένας συντελεστής, όπως περιγράφεται και στο Κεφάλαιο 2, που μετράει την συσχέτιση μεταξύ των ομάδων των κόμβων του δικτύου. Γενικά, όσο η τιμή του συντελεστή αυτού είναι πιο κοντά στην μονάδα τόσο καλύτερα αποτελέσματα ομαδοποίησης δίνει ο αλγόριθμος. Πρέπει να σημειωθεί ότι ο συντελεστής συσχέτισης μπορεί να υπολογιστεί μόνο για ιεραρχικούς(hierarchical) αλγορίθμους. Παρακάτω ακολουθεί μία απεικόνιση των τιμών του συντελεστή για του ιεραρχικούς αλγορίθμους για διαφορετικό αριθμό κόμβων:



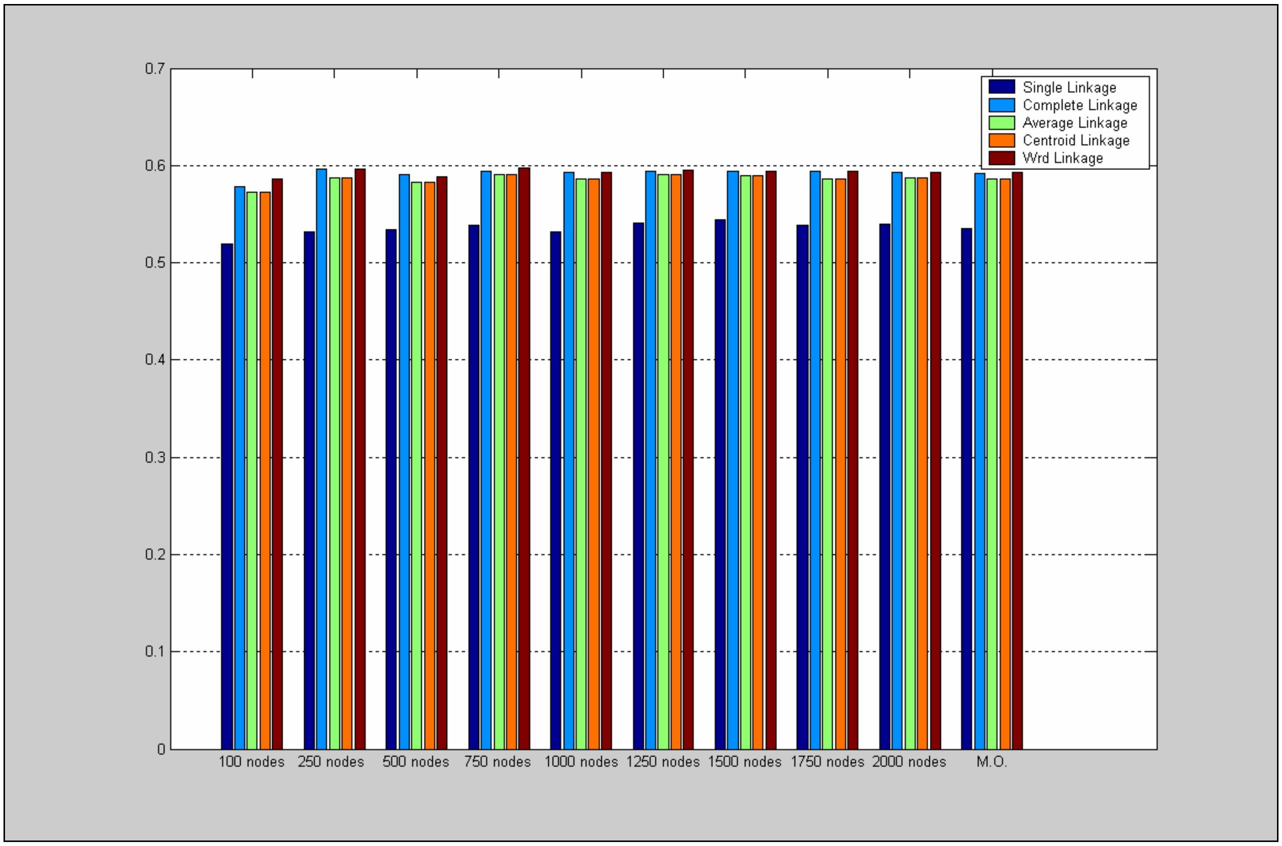
Εικόνα 5.6 : Τιμές του Συντελεστή Συσχέτισης για διάφορους αριθμούς κόμβων

Παρατηρήσεις:

Μετά από την σύγκριση που γίνεται από το παραπάνω ραβδόγραμμα προκύπτει η παρακάτω αξιολόγηση των μεθόδων (από την καλύτερη προς τη χειρότερη):
1) Complete Linkage, 2) Average Linkage, 3)Centroid Linkage, 4)Ward Linkage, 5) Single Linkage.

5.2.3) 3^{o} Κριτήριο σύγκρισης : Συντελεστής "αστάθειας"

Ο συντελεστής αστάθειας είναι ένα μέτρο, που δείχνει την ομοιότητα μεταξύ των ομάδων που δημιουργούνται. Όσο πιο υψηλή είναι η τιμή του συντελεστή, τόσο πιο πολύ διαφέρουν τα "σημάδια" κανόνων Golomb που βρίσκονται στην ίδια ομάδα, άρα τόσο πιο κακή ποιοτικά είναι η ομαδοποίηση λαμβάνοντας υπ'όψιν την εσωτερική ομαδοποίηση σε κάθε κόμβο. Παρακάτω ακολουθεί ένα ραβδόγραμμα που δείχνει την ποιότητα των ομαδοποιήσεων εσωτερικά στους κόμβους για διάφορους αριθμούς κόμβων:



Εικόνα 5.7 : Τιμές του Συντελεστή Αστάθειας για διάφορους αριθμούς κόμβων

Παρατηρήσεις:

Μετά από την σύγκριση που γίνεται από το παραπάνω ραβδόγραμμα προκύπτει η παρακάτω αξιολόγηση των μεθόδων (από την καλύτερη προς τη χειρότερη):

1) Single Linkage, 2) Centroid Linkage, 3)Average Linkage, 4)Complete Linkage, 5) Ward Linkage .

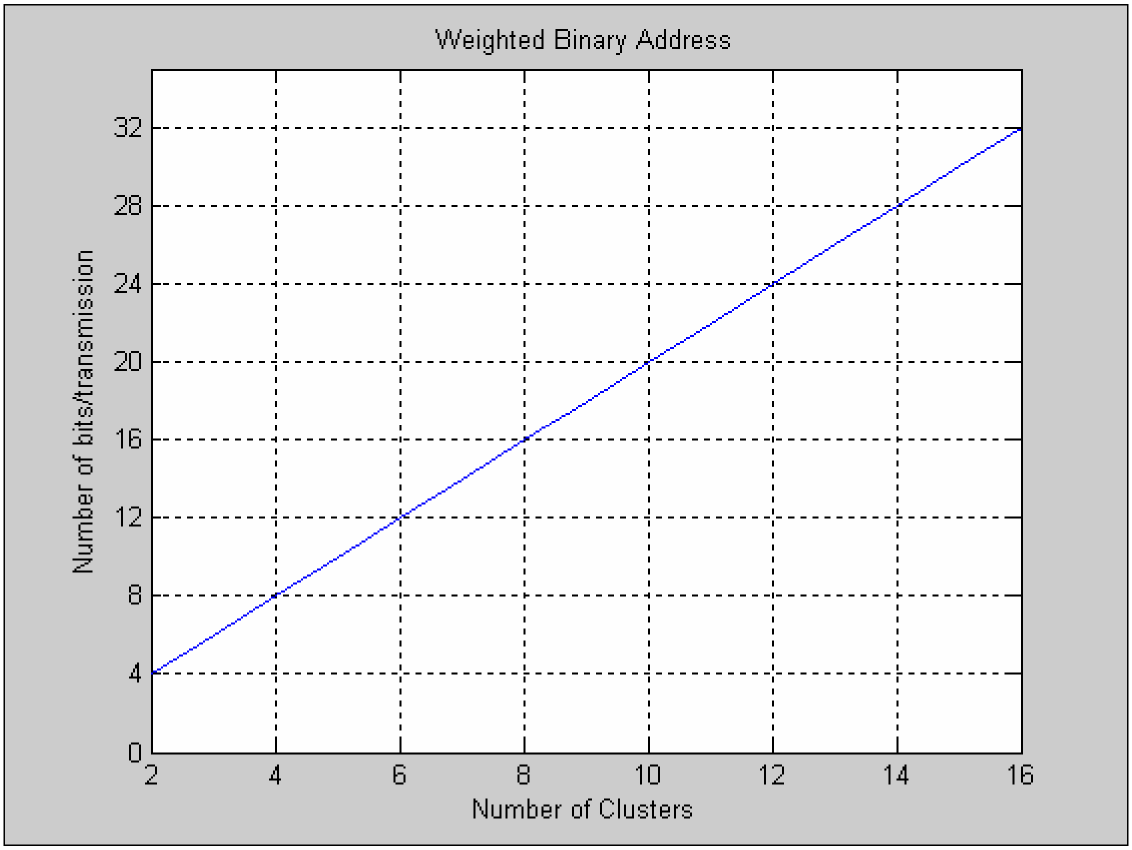
5.2.4) 4ο Κριτήριο σύγκρισης : Αριθμός bits/μετάδοση

Όπως είχε αναφερθεί και στην εισαγωγή της εργασίας, για ένα ασύρματο δίκτυο η μετάδοση πληροφορίας είναι αυτή που καταναλώνει το μεγαλύτερο ποσοστό ενέργειας. Ο σκοπός της εργασίας ήταν η εύρεση, μέσω της διευθυνσιοδότησης με "σημάδια" κανόνων Golomb, τρόπων ώστε να μειωθεί όσο το δυνατόν ο αριθμός των bits που μεταδίδονται από κόμβο σε κόμβο. Παρακάτω ακολουθεί μία ανάλυση ως προς τον μέσο αριθμό των bits που χρησιμοποιούνται σε κάθε μετάδοση ανάμεσα στις ομάδες των κόμβων για κάθε μέθοδο ομαδοποίησης που υλοποιήθηκε παραπάνω.

A) Weighted binary διευθύνσεις

Όπως αναφέρθηκε σε προηγούμενο κεφάλαιο, η διευθυνσιοδότηση των ομάδων των κόμβων με weighted binary αριθμούς οδηγεί στο αποτέλεσμα ότι όσος είναι ο αριθμός των ομάδων τόσος είναι και ο αριθμός των bits για τις διευθύνσεις των ομάδων. Συνεπώς, αν υπάρχει ένα δίκτυο με 8 διαφορετικές ομάδες κόμβων, και θεωρώντας ισοπίθανο το γεγονός να επικοινωνήσει μία ομάδα με οποιαδήποτε άλλη ομάδα, τότε τα Bits σε κάθε πακέτο για τις διευθύνσεις είναι 8 bits(για την διεύθυνση του αποστολέα) + 8 bits (για την διεύθυνση του παραλήπτη) = 16 bits.

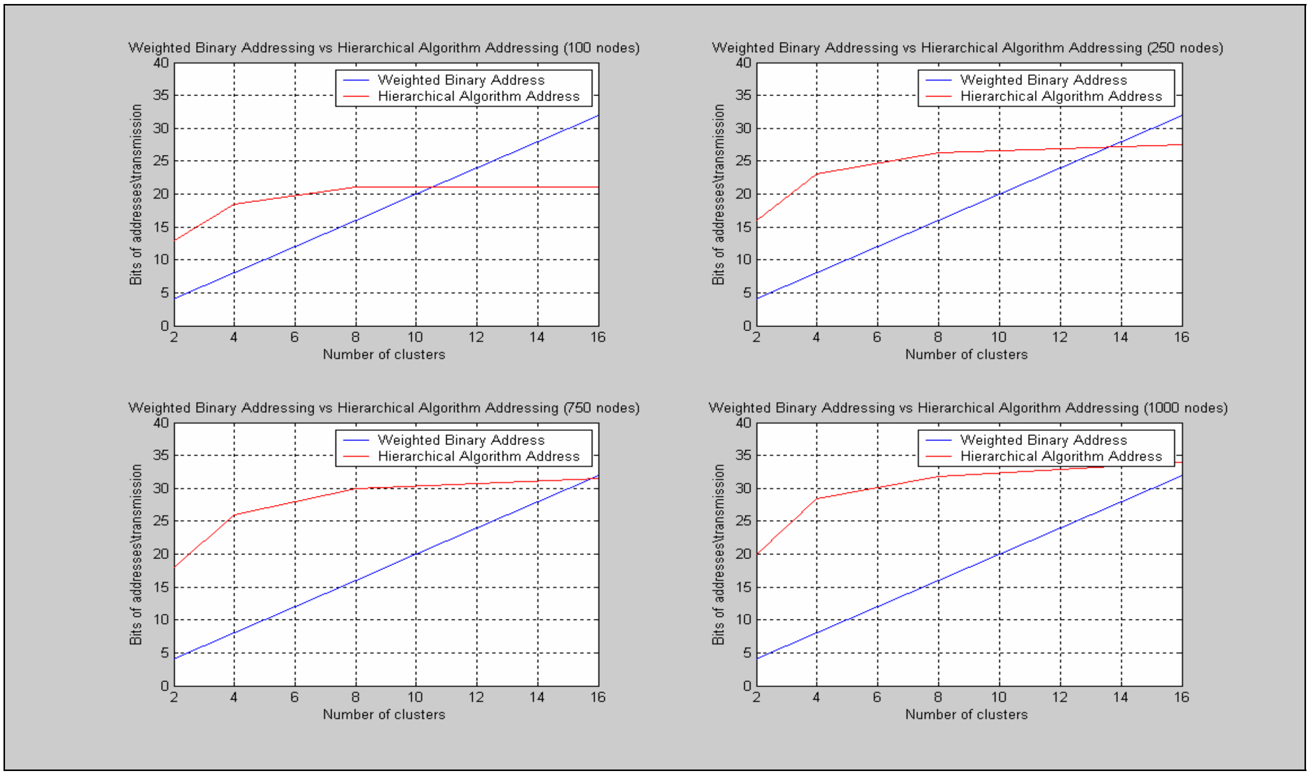
Θεωρώντας την ίδια περίπτωση, γενικά για ένα δίκτυο με **N ομάδες**, προκύπτει ότι σε κάθε πακέτο που μεταδίδεται για τις διευθύνσεις των κόμβων τα bits που μεταδίδονται είναι ίσα με **2 * N bits/μετάδοση**. Η γραφική παράσταση που απεικονίζει τον αριθμό των bits ανά μετάδοση σε σχέση με τον αριθμό των ομάδων, σε ένα δίκτυο αισθητήρων είναι αυτή που φαίνεται παρακάτω(**Εικόνα 5.8**):



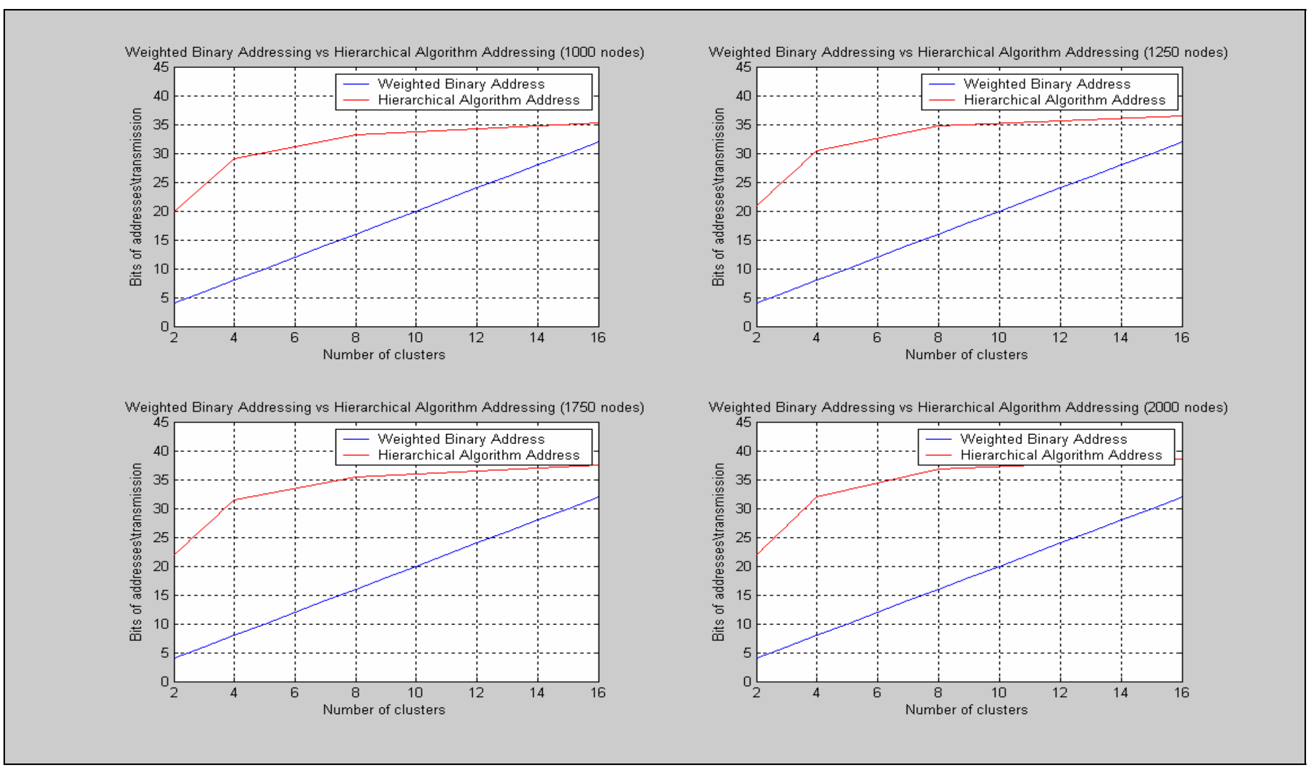
Εικόνα 5.8 : Γραφική παράσταση του αριθμού των Bits/μετάδοση με διευθυνσιοδότηση με weighted binary αριθμούς

B) Διευθύνσεις για partitional και hierarchical ομαδοποιήσεις

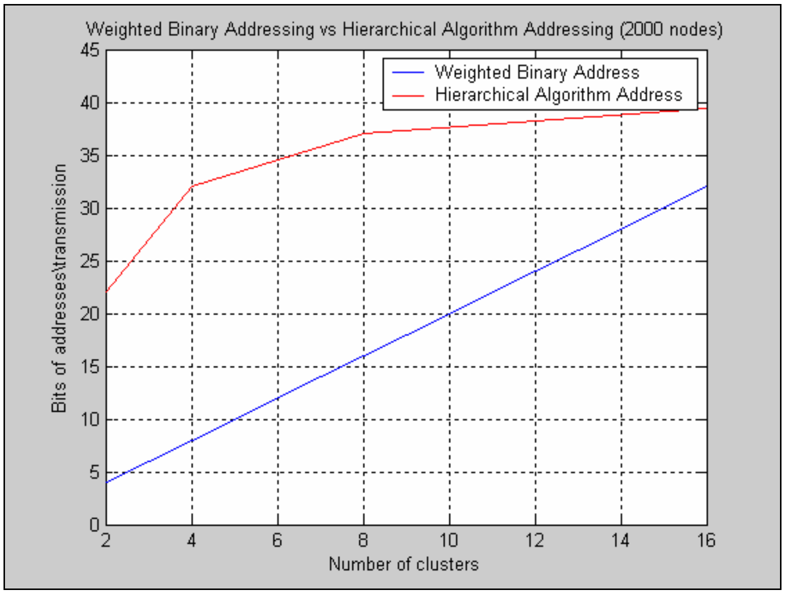
Επειδή οι αλγόριθμοι για partitional και hierarchical ομαδοποιήσεις, που υλοποιήθηκαν ήταν πολλοί, η σύγκριση ως προς τον αριθμό των bits/μετάδοση έγινε με κριτήριο το σφάλμα, όπως περιγράφηκε στην πρώτη παράγραφο του κεφαλαίου και γι'αυτό τον λόγο χρησιμοποιήθηκε κάθε φορά ο αριθμός των bits που χρησιμοποιεί ο partitional και hierarchical αλγόριθμος με το μικρότερο σφάλμα. Παρακάτω ακολουθεί μία γραφική αναπαράσταση του καλύτερου hierarchical αλγορίθμου για τις διάφορες τιμές των κόμβων αλλά και για τις διάφορες τιμές του αριθμού των ομάδων:



Εικόνα 5.9 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για hierarchical αλγόριθμο(100, 250, 500, 750 κόμβοι)



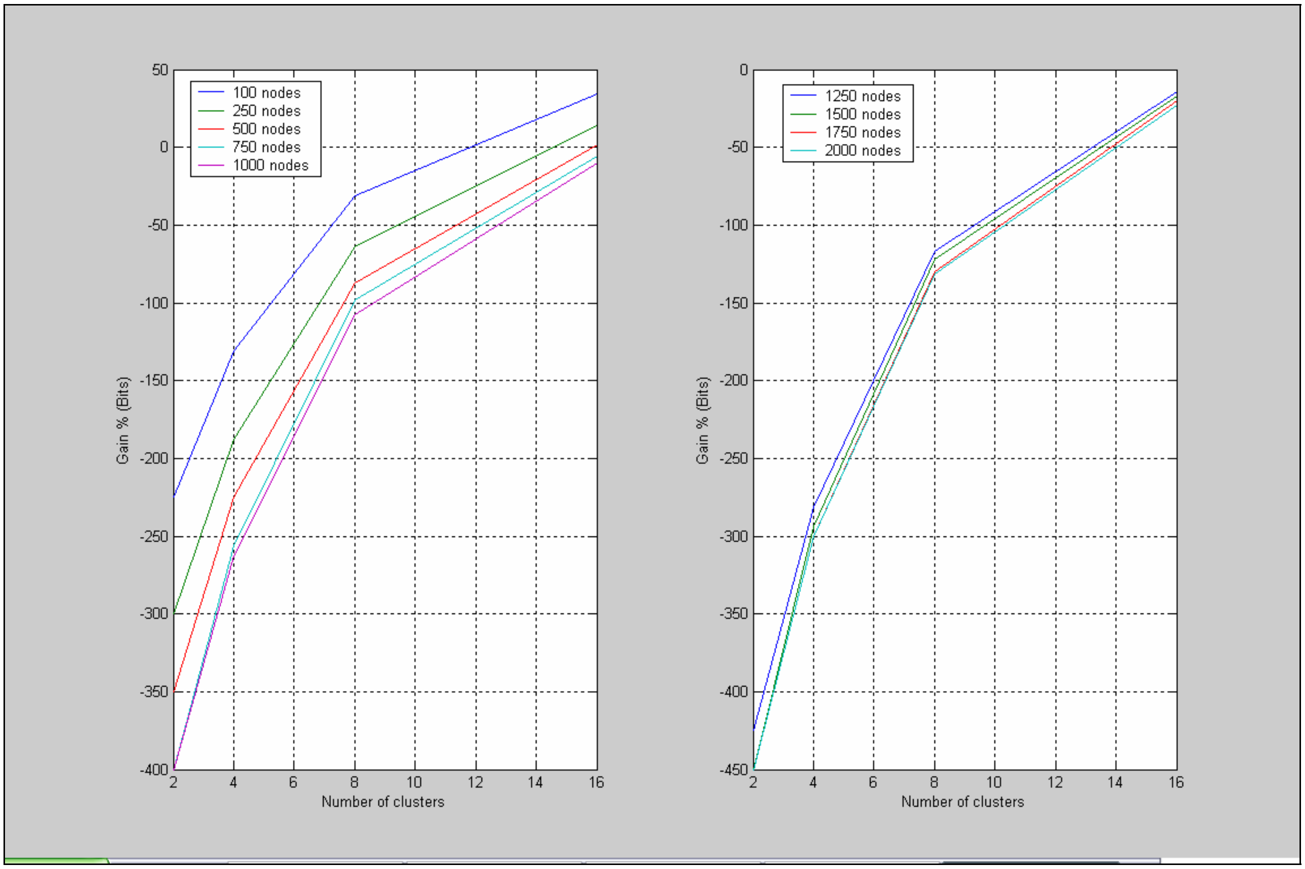
Εικόνα 5.10 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για hierarchical αλγόριθμο(1000, 1250, 1500, 1750 κόμβοι)



Εικόνα 5.11 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για hierarchical αλγόριθμο(2000 κόμβοι)

Όπως, μπορεί να παρατηρηθεί από την παραπάνω γραφική παράσταση αλλά και από την γραφική παράσταση του κέρδους , που ακολουθεί, **η διευθυνσιοδότηση με αριθμούς Golomb και με ομαδοποίηση με partitional και hierarchical αλγορίθμους είναι χειρότερη από την ομαδοποίηση με weighted binary αριθμούς σχεδόν για όλους τους αριθμούς των διευθύνσεων.** Επίσης, ένα άλλο σημαντικό στοιχείο, που προκύπτει από την παραπάνω γραφική παράσταση είναι ότι ο αριθμός των bits/μετάδοση καθορίζεται από τον αριθμό των ομάδων και όσο αυξάνεται αυτός ο αριθμός τόσο αυξάνεται και ο αριθμός των bits. Η αύξηση αυτή στην περίπτωση των partitional και hierarchical αλγορίθμων όπως και στις υπόλοιπες μεθόδους μειώνεται από έναν ορισμένο αριθμό ομάδων και πάνω.

Παρακάτω ακολουθεί μία άλλη γραφική παράσταση που δείχνει το κέρδος της διευθυνσιοδότησης με κανόνες Golomb και με χρήση αλγορίθμων partitional και hierarchical σε σχέση με την διευθυνσιοδότηση με weighted binary αριθμούς.

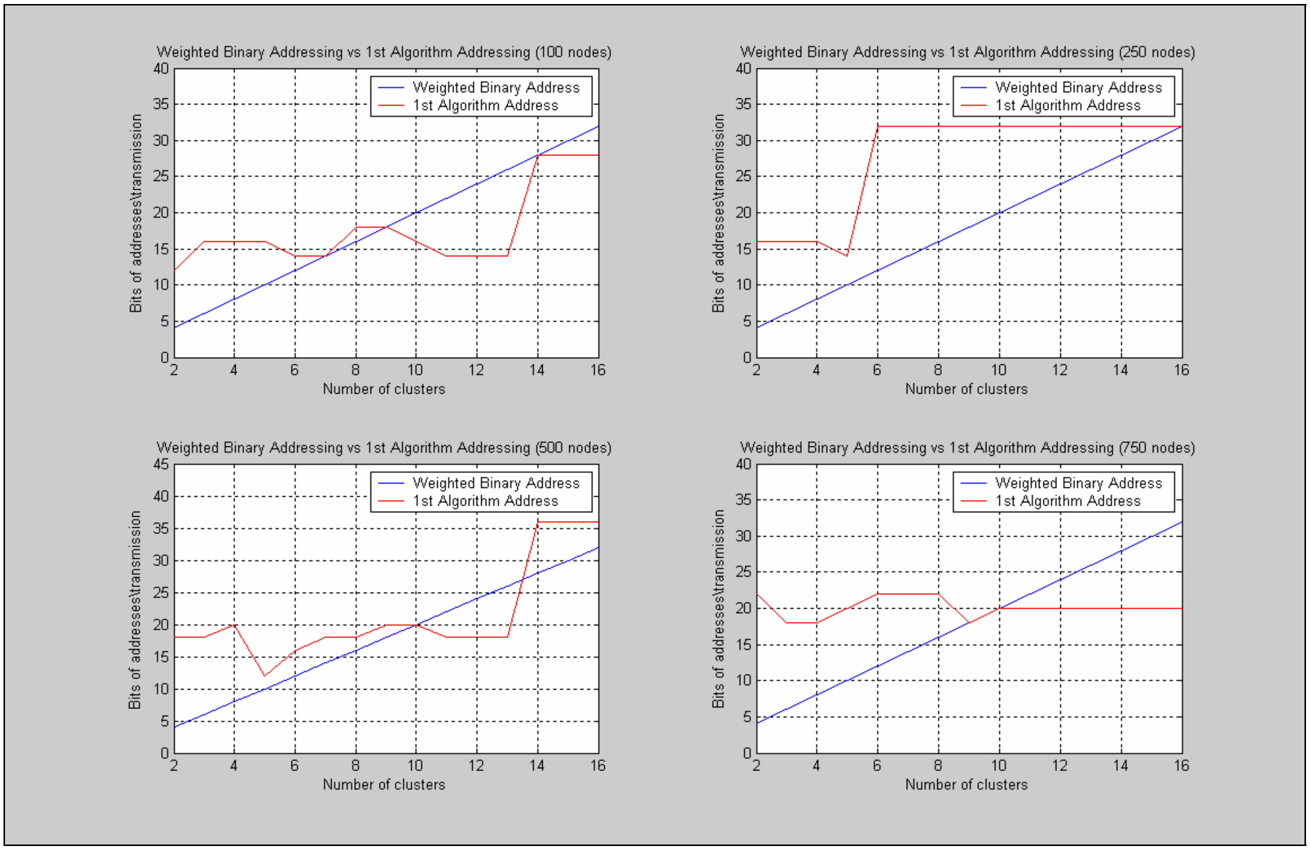


Εικόνα 5.12 : Κέρδος % σε bits (Weighted binary vs. Partitional & Hierarchical Algorithm)

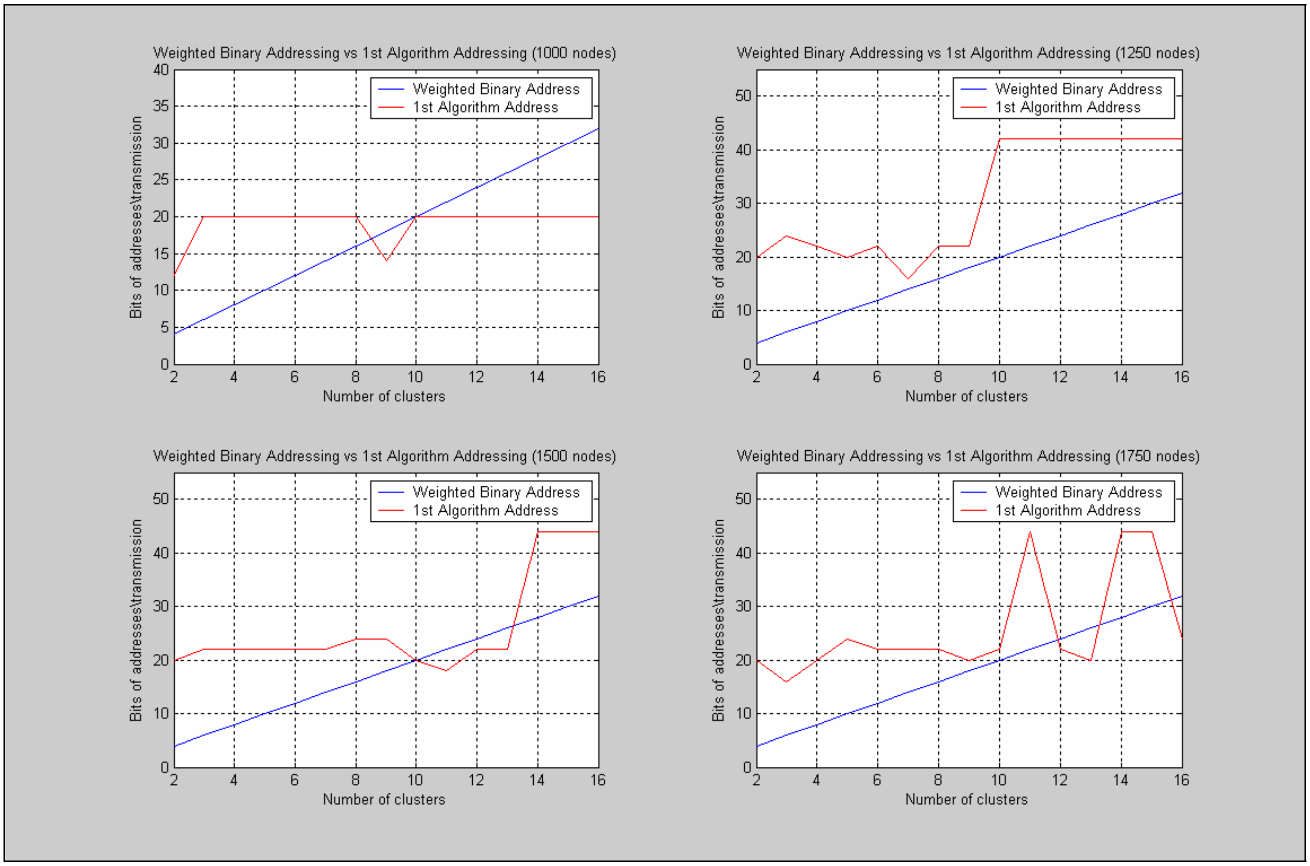
Γ) Διευθυνσιοδότηση με βάση τον 1ο ευρυστικό αλγόριθμο ομαδοποίησης

Ο επόμενος αλγόριθμος με τον οποίο συγκρίθηκαν τα αποτελέσματα της διευθυνσιοδότησης με weighted binary αριθμούς ήταν ο 1ος ευρυστικός αλγόριθμος. Λόγω της ιδιομορφίας που έχει αυτός ο αλγόριθμος, τα bits που χρησιμοποιούνται για τις διευθυνσιοδοτήσεις των πακέτων δεν αυξάνουν μονότονα καθώς αυξάνουν και οι ομάδες των κόμβων. Αυτό είναι λογικό, αφού όπως περιγράφεται και στην παράγραφο 4.3.2, οι ομάδες προκύπτουν μετά από εφαρμογή του αλγορίθμου bisecting k-means και στην συνέχεια ανάλογα με τις ομαδοποιήσεις των διευθύνσεων επιλέγεται ένα μέρος της κάθε διεύθυνσης, έτσι ώστε να μην επαναλαμβάνεται ίδιο κομμάτι της διεύθυνσης σε διαφορετικές ομάδες.

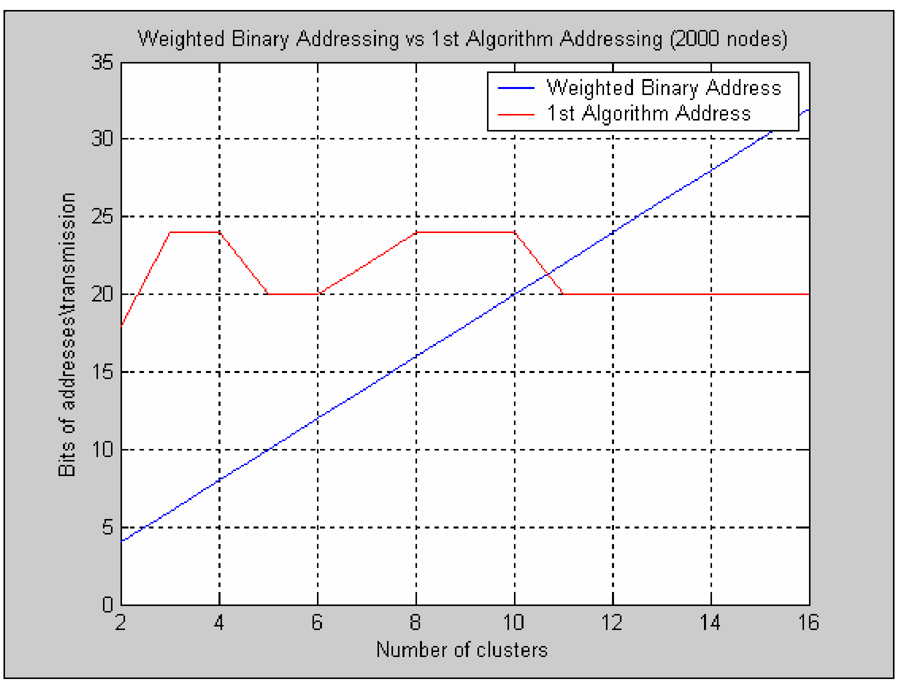
Όπως είναι φανερό, ο αριθμός των bits αυτών, διαφέρει ανάλογα με τις διευθύνσεις που ανήκουν σε κάθε ομάδα και όχι τόσο πολύ από τον αριθμό των διευθύνσεων. Αυτό μπορεί να γίνει αντιληπτό από τις παρακάτω γραφικές παραστάσεις όπου παρουσιάζονται τα αποτελέσματα του $1^{ου}$ ευρυστικού αλγορίθμου για διάφορους αριθμούς κόμβων και για διάφορο αριθμό ομάδων σε σύγκριση με την διευθυνσιοδότηση με weighted binary αριθμούς. Αμέσως μετά ακολουθεί το κέρδος σε bits που θα υπάρξει αν σε ένα ασύρματο δίκτυο χρησιμοποιηθεί διευθυνσιοδότηση και ομαδοποίηση με βάση αυτόν τον αλγόριθμο ως προς τις weighted binary διευθύνσεις.



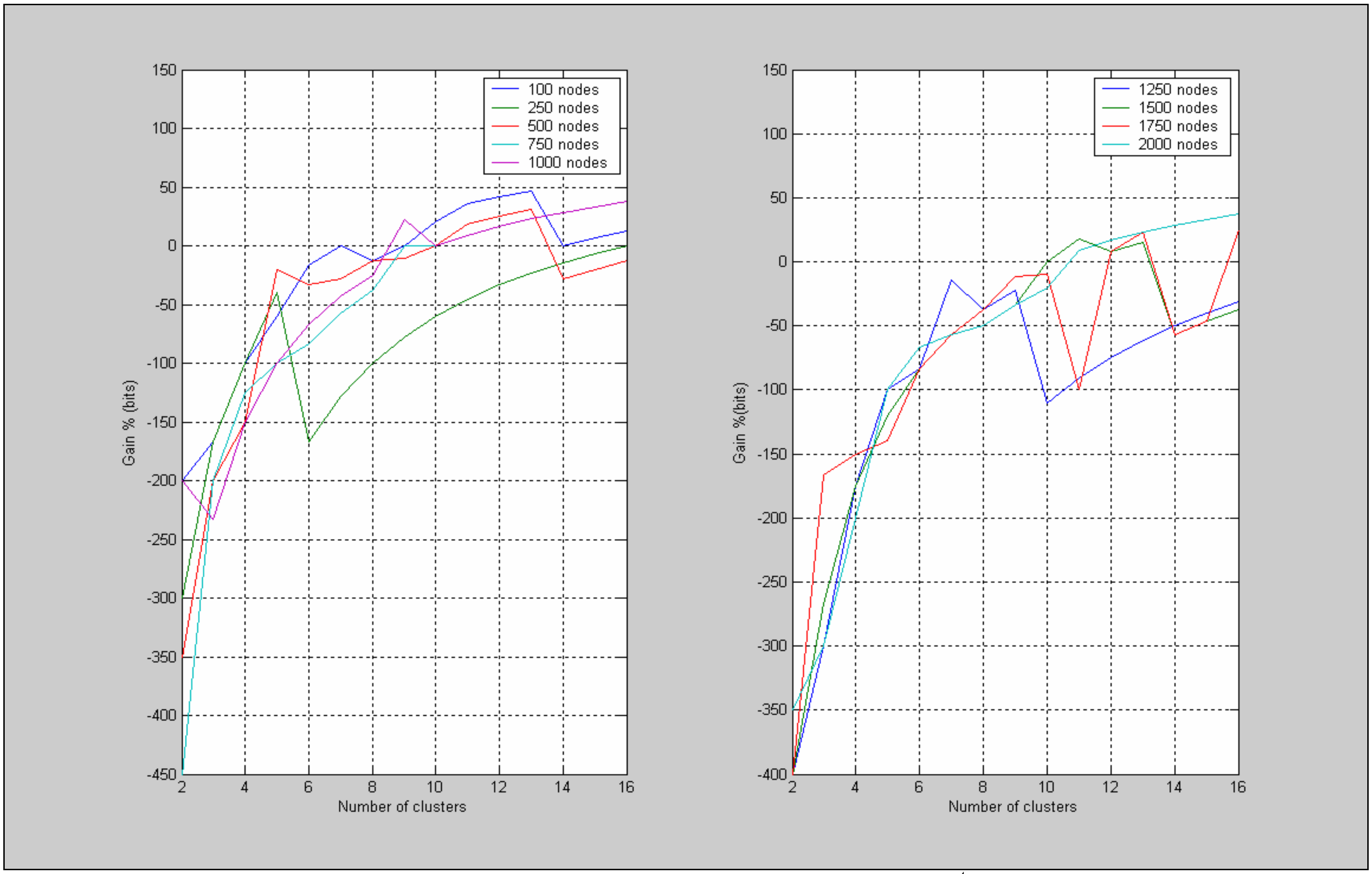
Εικόνα 5.13 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 1^{ο} ευρυστικό αλγόριθμο(100, 250, 500, 750 κόμβοι)



Εικόνα 5.14 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 1ο ευρυστικό αλγόριθμο(1000, 1250, 1500, 1750 κόμβοι)



Εικόνα 5.15 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 1ο ευρυστικό αλγόριθμο(2000 κόμβοι)

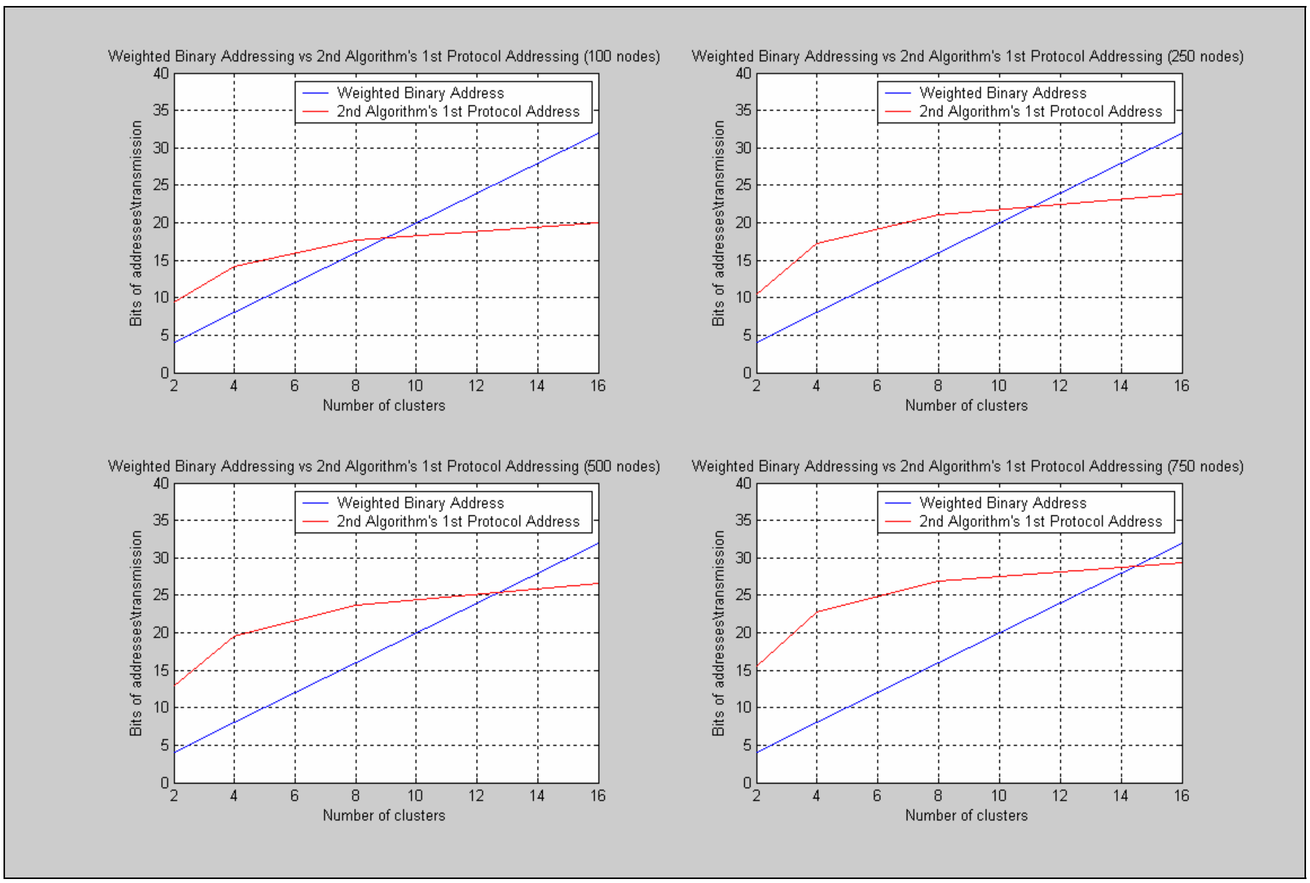


Εικόνα 5.16 : Κέρδος % σε bits (Weighted binary vs. 1st Algorithm)

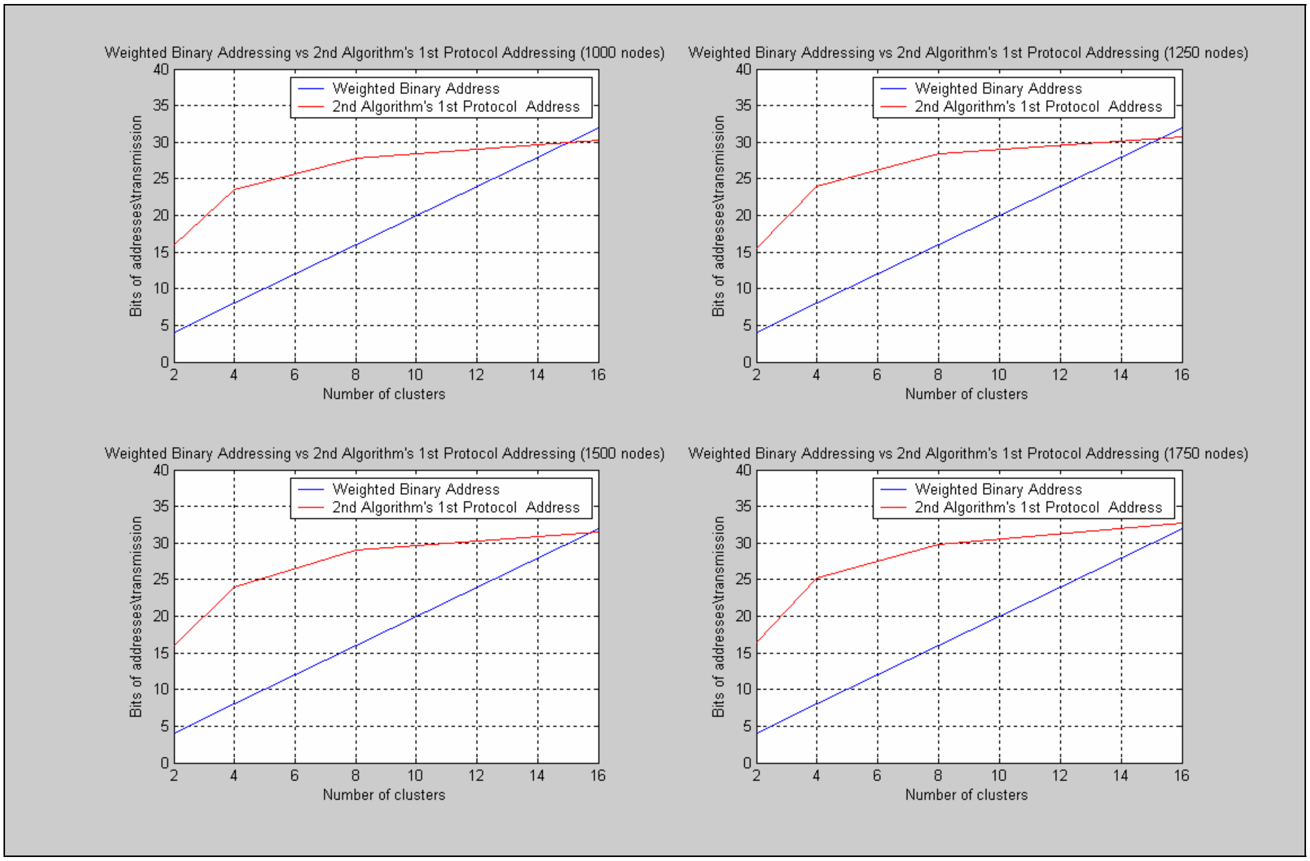
Παρατηρώντας τις παραπάνω γραφικές παραστάσεις μπορεί να βγει το συμπέρασμα ότι η διευθυνσιοδότηση με αριθμούς Golomb με την εφαρμογή του 1ου ευρυστικού αλγορίθμου έχει πολύ χειρότερα αποτελέσματα από την διευθυνσιοδότηση με weighted binary αριθμούς (μέχρι και -450%!). Γενικά, **ο 1ος ευρυστικός αλγόριθμος δεν φαίνεται να υπερτερεί απέναντι στην χρήση weighted binary διευθύνσεων ως προς τον αριθμό των bits/μετάδοση ενώ παρουσιάζει μία βελτίωση στις περιπτώσεις με μεγαλύτερο αριθμό κόμβων αλλά και πάλι αυτό δεν ισχύει για όλες τις περιπτώσεις** (βλέπε την γραφική του κέρδους για 1250 κόμβους).

Γ) Διευθυνσιοδότηση με βάση τον 2ο ευρυστικό αλγόριθμο ομαδοποίησης-Μέθοδος "δένδρου"

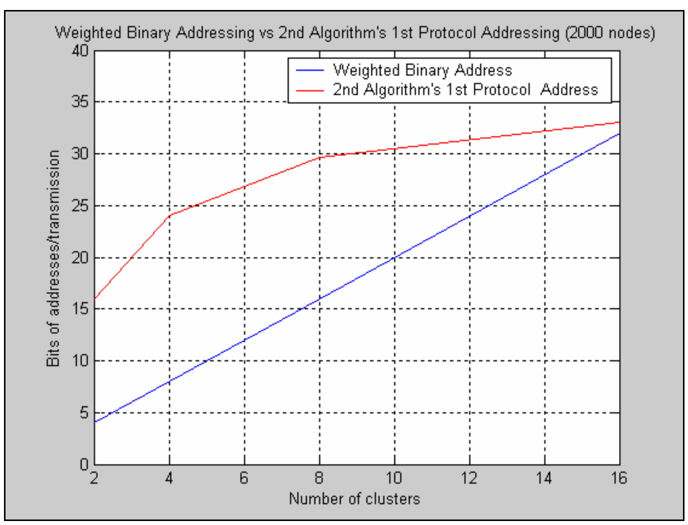
Στην περίπτωση του δεύτερου ευρυστικού αλγορίθμου, που υλοποιήθηκε, χρησιμοποιήθηκαν δύο διαφορετικά πρωτόκολλα επικοινωνίας. Για το πρώτο πρωτόκολλο (Παράγραφος 4.3.3) είναι φανερό, από τον τρόπο λειτουργίας του αλγορίθμου, ότι **ο αριθμός των bits που αποστέλλονται θα εξαρτάται από τον αριθμό των bits των διευθύνσεων που προκύπτουν από τον αλγόριθμο.** Εφαρμόζοντας τον αλγόριθμο στους διάφορους κανόνες Golomb και για διάφορο αριθμό ομάδων προκύπτουν οι παρακάτω γραφικές παραστάσεις:



Εικόνα 5.17 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 2ο ευρυστικό αλγόριθμο(1ο Πρωτόκολλο Επικοινωνίας) (100, 250, 500, 750 κόμβοι)

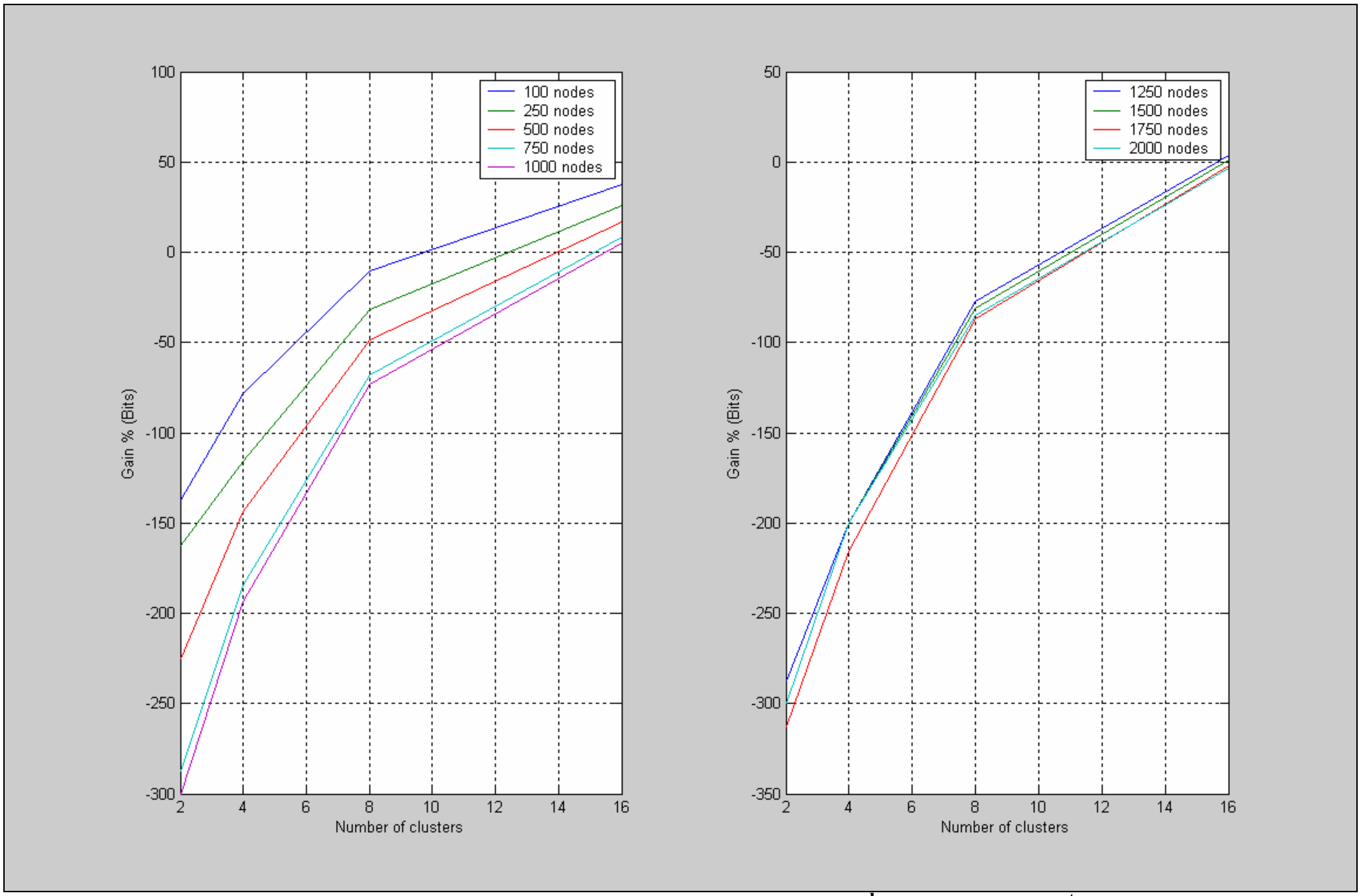


Εικόνα 5.18 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 2^{o} ευρυστικό αλγόριθμο(1^{o} Πρωτόκολλο Επικοινωνίας) (1000, 1250, 1500, 1750 κόμβοι)



Εικόνα 5.19 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 2^{o} ευρυστικό αλγόριθμο(1^{o} Πρωτόκολλο Επικοινωνίας) (2000 κόμβοι)

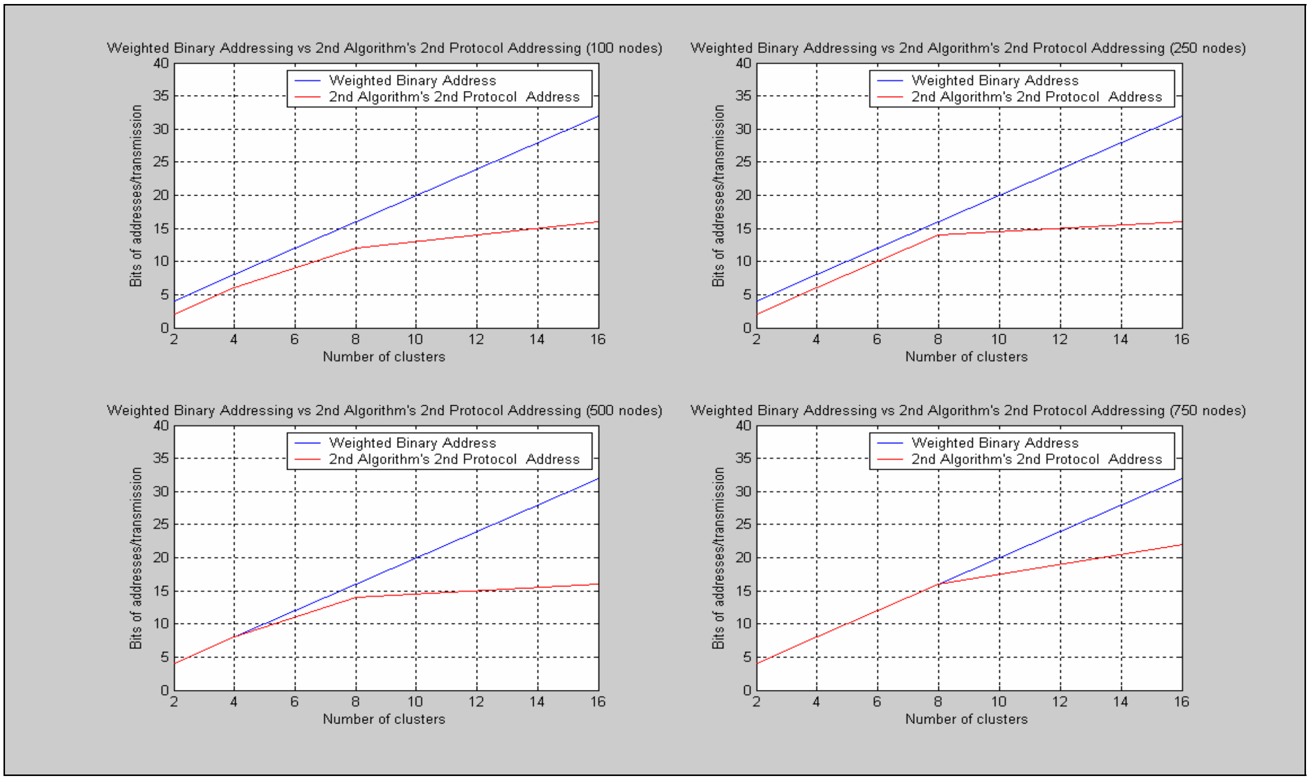
Στην παρακάτω γραφική παράσταση παρουσιάζεται το κέρδος του συγκεκριμένου αλγορίθμου σε σχέση με τις weighted binary διευθύνσεις.



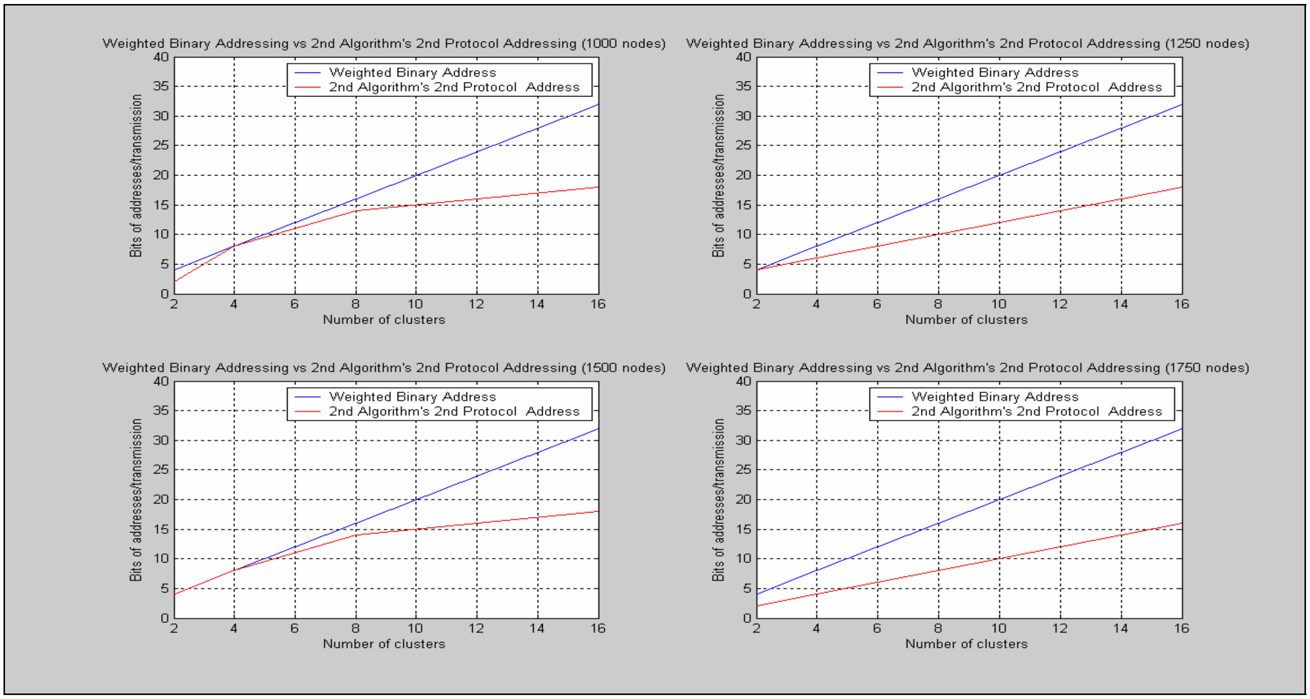
Εικόνα 5.20 : Κέρδος % σε bits (Weighted binary vs. 2nd Algorithm's 1st Protocol)

Για το 1^{ο} πρωτόκολλο επικοινωνίας του 2^{ου} αλγόριθμου ισχύουν ότι και για τον 1^{ο} αλγόριθμο που περιγράφηκε στην προηγούμενη παράγραφο. **Γενικά, τα αποτελέσματα της μεθόδου αυτής είναι πολύ χειρότερα από της μεθόδου με weighted binary αριθμούς (μέχρι και -300%) ενώ για μικρό αριθμό κόμβων και μεγάλο αριθμό ομάδων φαίνεται να βελτιώνεται αλλά και πάλι με μικρούς ρυθμούς.**

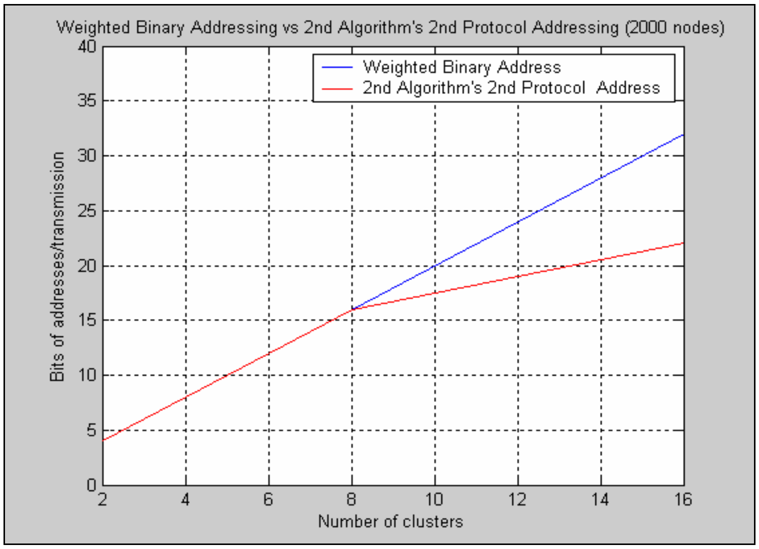
Στο δεύτερο πρωτόκολλο του 2^{ου} αλγορίθμου ο αριθμός των bits εξαρτάται αποκλειστικά και είναι ίσος με τα bits που δημιουργούν τις ομαδοποιήσεις. Εφαρμόζοντας τον παραπάνω αλγόριθμο στα “σημάδια” των κανόνων Golomb, που έχουν κατασκευαστεί εξαγάγονται τα παρακάτω αποτελέσματα:



Εικόνα 5.21 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 2^{ο} ευρυστικό αλγόριθμο(2^{ο} Πρωτόκολλο Επικοινωνίας) (100, 250, 500, 750 κόμβοι)

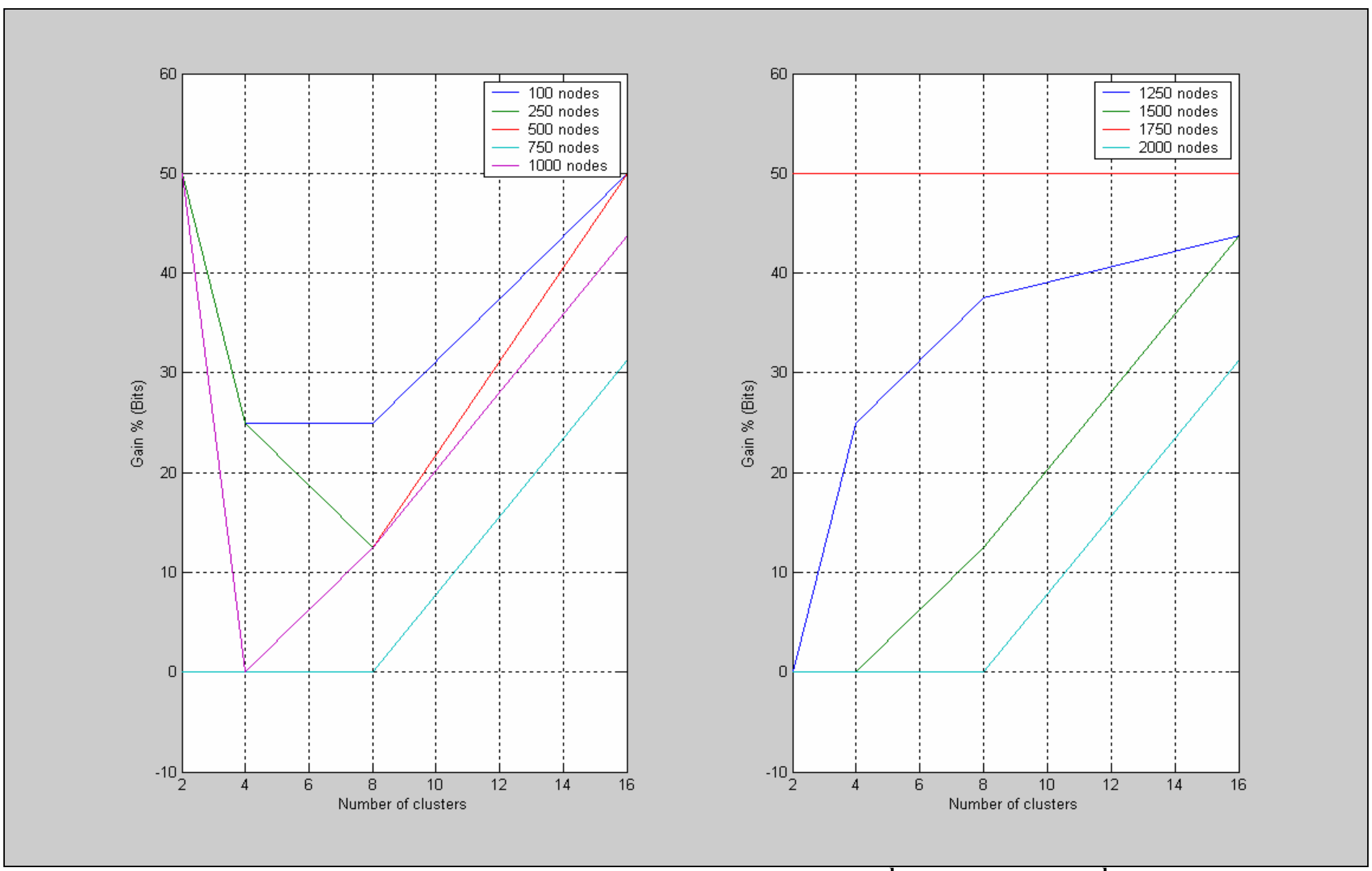


Εικόνα 5.22 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 2^{ο} ευρυστικό αλγόριθμο(2^{ο} Πρωτόκολλο Επικοινωνίας) (1000, 1250, 1500, 1750 κόμβοι)



Εικόνα 5.23 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 2ο ευρυστικό αλγόριθμο(2ο Πρωτόκολλο Επικοινωνίας) (2000 κόμβοι)

Το κέρδος για διάφορο αριθμό κόμβων και με την χρήση του 2ου πρωτοκόλλου του δεύτερου αλγόριθμου παρουσιάζεται στις παρακάτω γραφικές παραστάσεις:

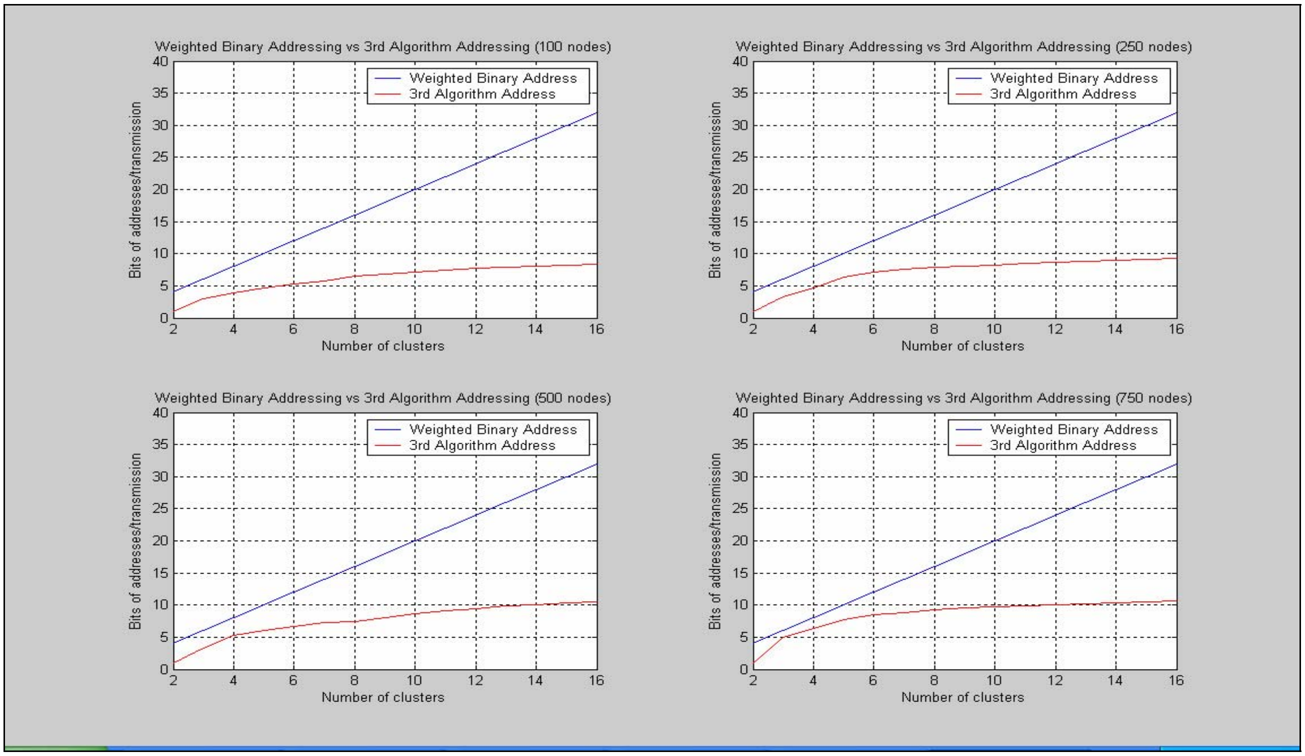


Εικόνα 5.24 : Κέρδος % σε bits (Weighted binary vs. 2nd Algorithm's 2nd Protocol)

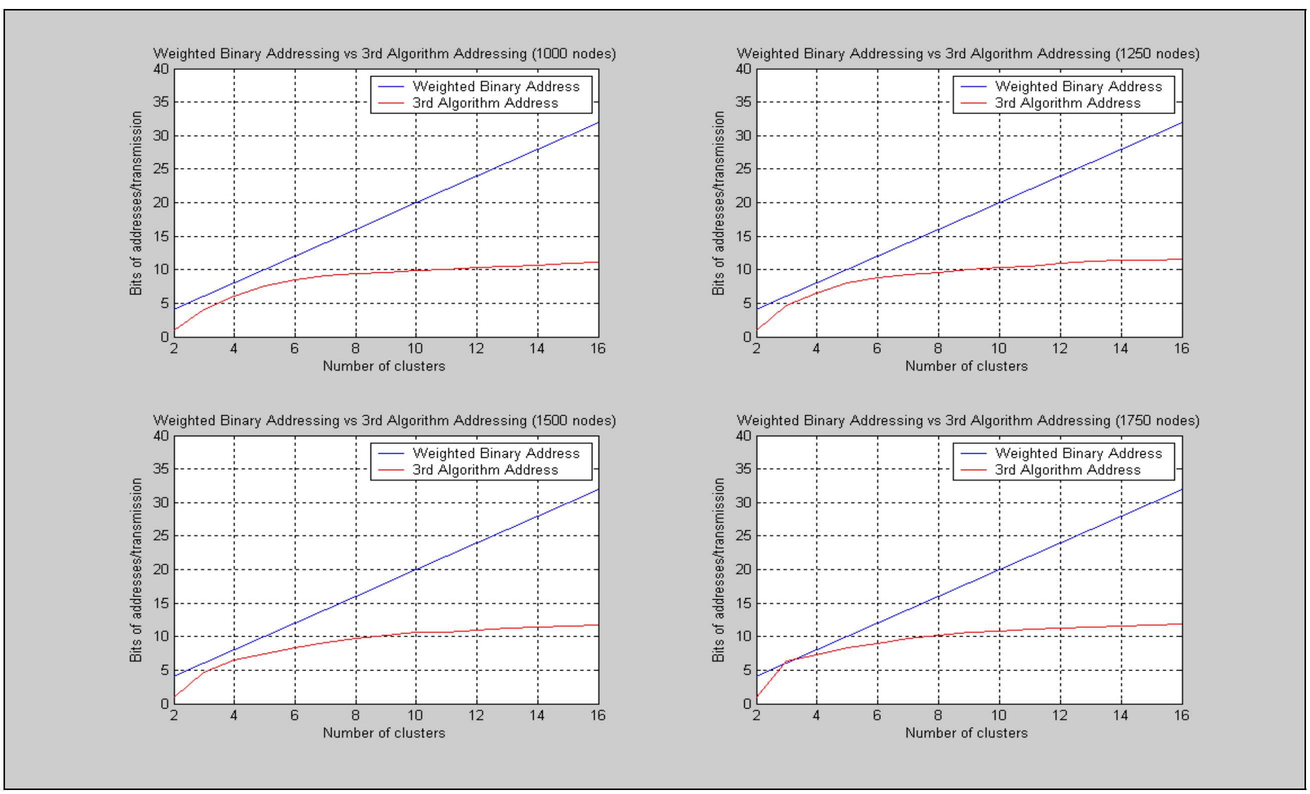
Το 2ο πρωτόκολλο του αλγορίθμου, όπως φαίνεται και παραπάνω, σε αντίθεση με την πρώτη μέθοδο δίνει αρκετά καλά αποτελέσματα και με βάση την γραφική αναπαράσταση του κέρδους καλύτερα αποτελέσματα από την μέθοδο με weighted binary αριθμούς. Το κέρδος έναντι της μεθόδου με weighted binary διευθύνσεις μπορεί να φτάσει μέχρι και το 50% σε αριθμό bits.

Δ) Διευθυνσιοδότηση με βάση τον 3ο ευρυστικό αλγόριθμο clustering – Μέθοδος "αρχηγών"

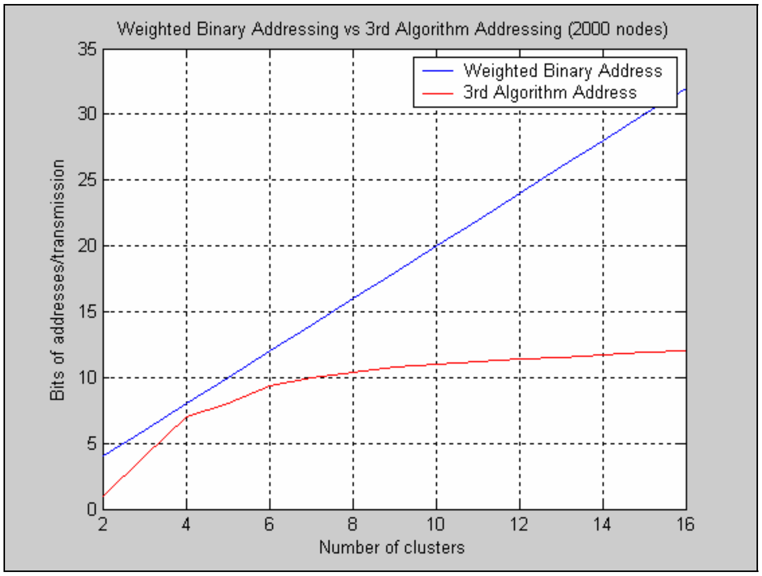
Στην μέθοδο αυτή πρωταρχικό ρόλο παίζουν οι αρχηγοί των ομάδων και πιο συγκεκριμένα οι αποστάσεις μεταξύ των αρχηγών. Οι αρχηγοί, αυτοί επιλέγονται με τέτοιο τρόπο ώστε η ομαδοποίηση με αυτή την μέθοδο να έχει τον μικρότερο αριθμό bits/ μετάδοση, δηλαδή οι αποστάσεις μεταξύ τους να είναι όσο το δυνατόν μικρότερες. Παρακάτω ακολουθούν κάποιες γραφικές παραστάσεις, που δείχνουν και την σύγκριση, ως προς τον αριθμό των Bits χρησιμοποιούνται για την διευθυνσιοδότηση σε κάθε μετάδοση, της μεθόδου αυτής με την διευθυνσιοδότηση με τους weighted binary αριθμούς:



Εικόνα 5.25 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 3ο ευρυστικό αλγόριθμο (100, 250, 500, 750 κόμβοι)

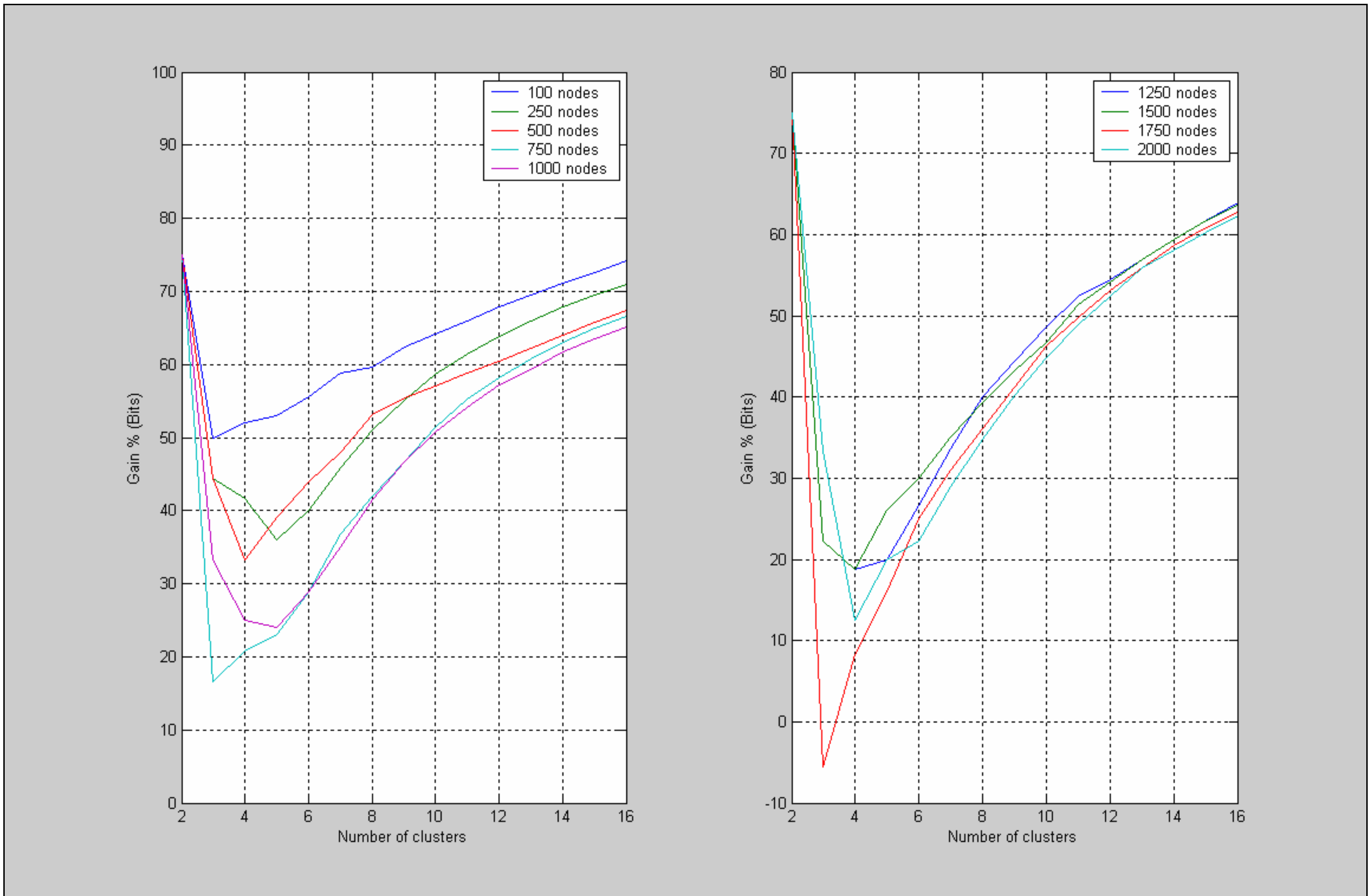


Εικόνα 5.26 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 3ο ευρυστικό αλγόριθμο (1000, 1250, 1500, 1750 κόμβοι)



Εικόνα 5.27 : Γραφική παράσταση του αριθμού των bits της διεύθυνσης του αποστολέα ανά αποστολή για τον 3^{ο} ευρυστικό αλγόριθμο (2000 κόμβοι)

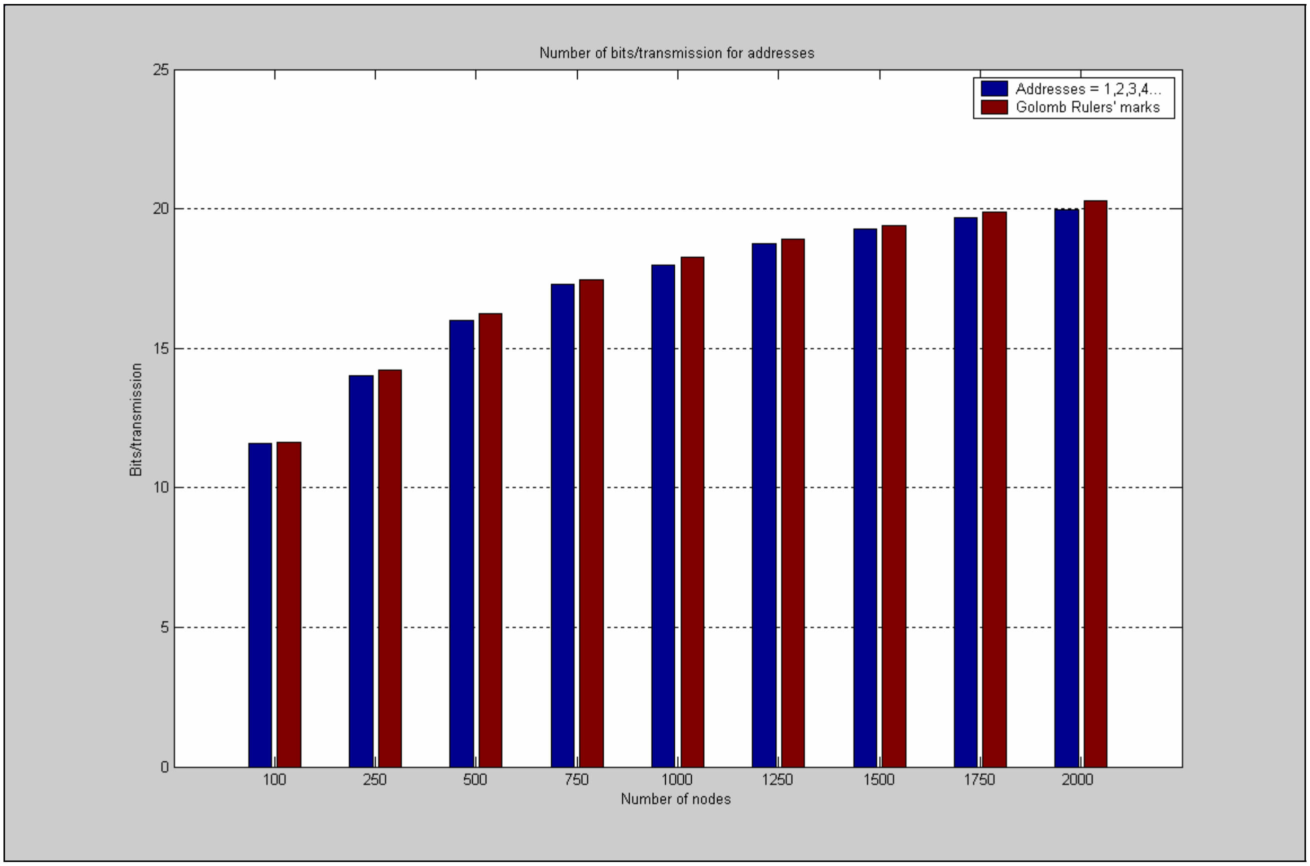
Παρατηρώντας, τις παραπάνω γραφικές παραστάσεις του αριθμού των bits/μετάδοση για κάθε αλγόριθμο μπορεί να εξαχθεί το συμπέρασμα ότι **ο 3ος ευρυστικός αλγόριθμος δίνει πολύ καλύτερα αποτελέσματα από την αντίστοιχη διευθυνσιοδότηση με weighted binary αριθμούς καθώς και από όλους του υπόλοιπους αλγορίθμους. Αυτό γίνεται φανερό ακόμα περισσότερο στην γραφική παράσταση, που ακολουθεί και περιγράφει το κέρδος της ομαδοποίησης με βάση τον 3^{ο} ευρυστικό αλγόριθμο σε σχέση με την χρήση Weighted Binary αριθμούς, όπου το κέρδος για τις περιπτώσεις που εξετάστηκαν φτάνει μέχρι και 80%**.



Εικόνα 5.28 : Κέρδος % σε bits (Weighted binary vs. 3rd Algorithm)

Εκτός από τις παραπάνω περιπτώσεις αλγορίθμων θα πρέπει να γίνει και μία γενικότερη ανάλυση και σύγκριση στην διευθυνσιοδότηση των κόμβων ενός ασύρματου δικτύου με τους κανόνες Golomb σε σύγκριση με την χρήση απλών ακεραίων αριθμών. Όπως έχει αναφερθεί και σε προηγούμενο Κεφάλαιο, στο κάθε πακέτο πληροφορίας που αποστέλλεται υπάρχει η διεύθυνση του παραλήπτη και του αποστολέα. Στην περίπτωση, όμως των κανόνων Golomb κάτι τέτοιο δεν είναι αναγκαίο διότι αντί για τις δύο διευθύνσεις μπορεί να υπάρξει η διαφορά των διευθύνσεων των κόμβων, η οποία είναι και μοναδική! Παρακάτω γίνεται μία ανάλυση για διάφορους αριθμούς κόμβων ως προς το μέσο όρο των bits που θα χρειαστούν αν σε ένα δίκτυο επιθυμούν όλοι οι κόμβοι να μιλήσουν με όλους τους υπόλοιπους. Όπως φαίνεται από το ραβδόγραμμα, που ακολουθεί, σίγουρα **η απλή διευθυνσιοδότηση από κανόνες Golomb δεν είναι**

χειρότερη σε καμία περίπτωση από την διευθυνσιοδότηση με weighted binary αριθμούς(Εικόνα 5.29).



Εικόνα 5.29 : Αριθμός bits/αποστολή (Weighted binary addresses vs. Golomb Addressing)

ΚΕΦΑΛΑΙΟ 6^{ο}

6.1) Εισαγωγή

Στο 6^{ο} Κεφάλαιο θα γίνει μία σύγκριση των μεθόδων, που υλοποιήθηκαν ως προς μία διαφορετική προσέγγιση και συγκεκριμένα ως προς τις δομές υλικού που θα πρέπει να χρησιμοποιεί κάθε κόμβος για κάθε διαφορετική ομαδοποίηση. Όλοι οι διαφορετικοί αλγόριθμοι, που παρουσιάστηκαν σε προηγούμενα Κεφάλαια, είχαν ως στόχο την ομαδοποίηση των "σημαδιών" του κανόνα Golomb και την καταχώρηση στην συνέχεια, του κάθε "σημαδιού" σε κόμβους ασύρματου δικτύου αισθητήρων ως διεύθυνση τους. Με βάση την κωδικοποίηση που είχε γίνει δημιουργήθηκαν και τα κατάλληλα πρωτόκολλα επικοινωνίας. Σε όλα τα δίκτυα σημαντικό ρόλο παίζει ο τρόπος με τον οποίο κάποιος κόμβος καταλαβαίνει αν κάποιο πακέτο πληροφορίας αναφέρεται σε αυτόν και στην ομάδα του ή όχι. Αυτό το γεγονός είναι ακόμα πιο σημαντικό στα ασύρματα δίκτυα στα οποία περιοριστικός παράγοντας είναι η ενέργεια. Γενικά, έχοντας γνώση πόσο σημαντική είναι η ενέργεια για τα ασύρματα δίκτυα στο παρακάτω Κεφάλαιο, θα παρουσιαστεί μία γενική εικόνα για τις δομές υλικού που χρησιμοποιούνται στον τρόπο εξαγωγής της πληροφορίας, αν κάποιο πακέτο αναφέρεται στην ομάδα του συγκεκριμένου κόμβου ή όχι.

6.2) Δομή πακέτου πληροφορίας

Ένα πολύ σημαντικό ρόλο στα δίκτυα παίζει η μορφή που θα έχει η επικεφαλίδα του κάθε πακέτου καθώς και τις πληροφορίες που θα περιέχει. Για να γίνει μία ανάλυση για τις δομές που χρειάζονται από τους κόμβους για την αναγνώριση ενός πακέτου θα πρέπει να δοθεί έστω και μια γενική μορφή της επικεφαλίδας του πακέτου ή τουλάχιστον της πληροφορίας που είναι απαραίτητη.

Στην περίπτωση που εξετάζεται θα πρέπει να προσεχθεί το γεγονός ότι γενικά στο δίκτυο μπορούν να υπάρχουν δύο διαφορετικές ομάδες πακέτων. Η μία ομάδα πακέτων είναι αυτή που αποστέλλεται από έναν κόμβο και έχει παραλήπτη έναν άλλο κόμβο και η άλλη ομάδα πακέτων είναι αυτή που αποστέλλεται από έναν κόμβο προς μία ομάδα κόμβων.

6.2.1) Πακέτο από κόμβο σε κόμβο

Το πακέτο πληροφορίας σε ένα δίκτυο, που αποστολέας του είναι ένας κόμβος και παραλήπτης του ένας άλλος κόμβος, στα περισσότερα δίκτυα έχει μία συγκεκριμένη δομή. Γενικά, η πληροφορία που πρέπει να μεταφέρει κάθε πακέτο είναι η εξής:

- 1 Bit που θα καθορίζει αν το πακέτο έχει παραλήπτη έναν κόμβο ή μία ολόκληρη ομάδα κόμβων
- Α Bits (όπου Α είναι ο αριθμός των bits που αντιστοιχούν στην διεύθυνση του αποστολέα)
- Π Bits (όπου Π είναι ο αριθμός των Bits που αντιστοιχούν στην διεύθυνση του παραλήπτη του πακέτου)
- Χ Bits (όπου Χ είναι ο συνολικός αριθμός bits που περιλαμβάνει όλη την χρήσιμη πληροφορία του πακέτου)

Γενικά πρέπει να αναφερθεί ότι υπάρχουν πρωτόκολλα που σε κάθε πακέτο υπάρχουν και άλλα bits με πληροφορία, αλλά στην συγκεκριμένη εργασία κάτι τέτοιο δεν μας αφορά, αφού τα παραπάνω bits αρκούν για την αναπαράσταση της πληροφορίας που χρησιμοποιείται.

6.2.2) Πακέτο από κόμβο σε ομάδα κόμβων

Ένα πακέτο πληροφορίας, που έχει ως παραλήπτη μία ομάδα κόμβων έχει περίπου την ίδια δομή με το πακέτο που περιγράφηκε παραπάνω. Η κύρια διαφορά είναι ότι λόγο της ύπαρξης παραπάνω από μία ομαδοποιήσεων των κόμβων αυτό οδηγεί στην εισαγωγή περισσοτέρων bits που να καθορίζουν και την συγκεκριμένη ομαδοποίηση. Πιο αναλυτικά τα τμήματα της επικεφαλίδας του πακέτου έχουν ως εξής:

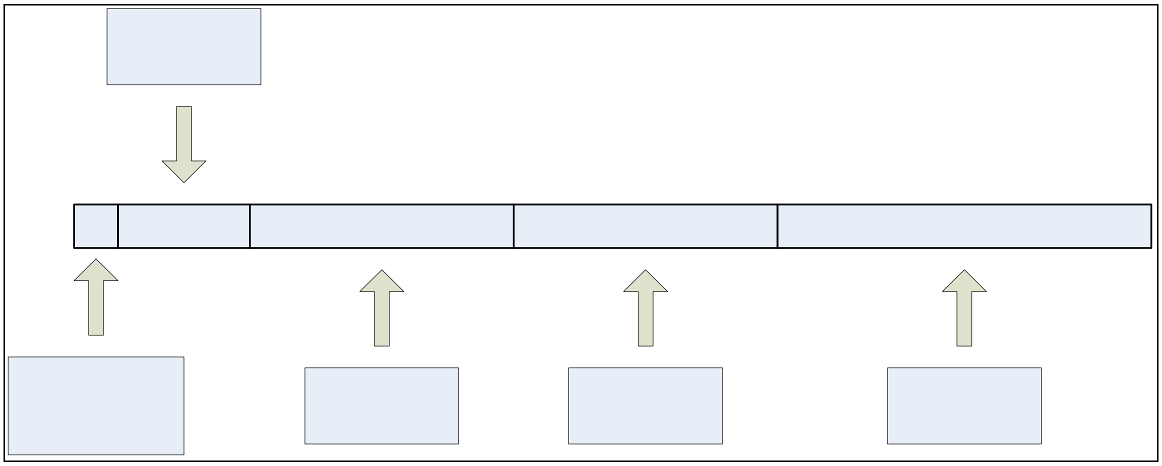
- 1 Bit που θα καθορίζει αν το πακέτο έχει παραλήπτη έναν κόμβο ή μία ολόκληρη ομάδα κόμβων
- N Bits, όπου θα καθορίζεται σε ποια κωδικοποίηση αναφέρεται η διεύθυνση της ομάδας των κόμβων, που είναι οι παραλήπτες.
- A Bits (όπου A είναι ο αριθμός των bits που αντιστοιχούν στην διεύθυνση του αποστολέα)
- Π Bits (όπου Π είναι ο αριθμός των Bits που αντιστοιχούν στην διεύθυνση της ομάδας του παραλήπτη του πακέτου)
- X Bits (όπου X είναι ο συνολικός αριθμός bits που περιλαμβάνει όλη την χρήσιμη πληροφορία του πακέτου)

6.2.3) Μορφή ενιαίας επικεφαλίδας πακέτου

Επειδή σε ένα δίκτυο δεν είναι δυνατόν να υπάρχουν διαφορετικές μορφές πακέτων, οι δύο προηγούμενες μορφές συμπτύχθηκαν σε μία ενιαία και έτσι τελικά η ενιαία μορφή του πακέτου αποτελείται από τα εξής μέρη:

- 1 Bit που θα καθορίζει αν το πακέτο έχει παραλήπτη έναν κόμβο ή μία ολόκληρη ομάδα κόμβων
- N Bits, όπου θα καθορίζεται σε ποια ομαδοποίηση αναφέρεται η διεύθυνση της ομάδας των κόμβων, που είναι οι παραλήπτες.
- A Bits (όπου A είναι ο αριθμός των bits που αντιστοιχούν στην διεύθυνση του αποστολέα)
- Π Bits (όπου Π είναι ο αριθμός των Bits που αντιστοιχούν στην διεύθυνση της ομάδας του παραλήπτη ή του κόμβου που είναι παραλήπτης του πακέτου)
- X Bits (όπου X είναι ο συνολικός αριθμός bits που περιλαμβάνει όλη την χρήσιμη πληροφορία του πακέτου)

Η μορφή του πακέτου παρουσιάζεται στην παρακάτω εικόνα (**Εικόνα 6.1**).



Εικόνα 6.1 : Δομή πακέτου πληροφορίας

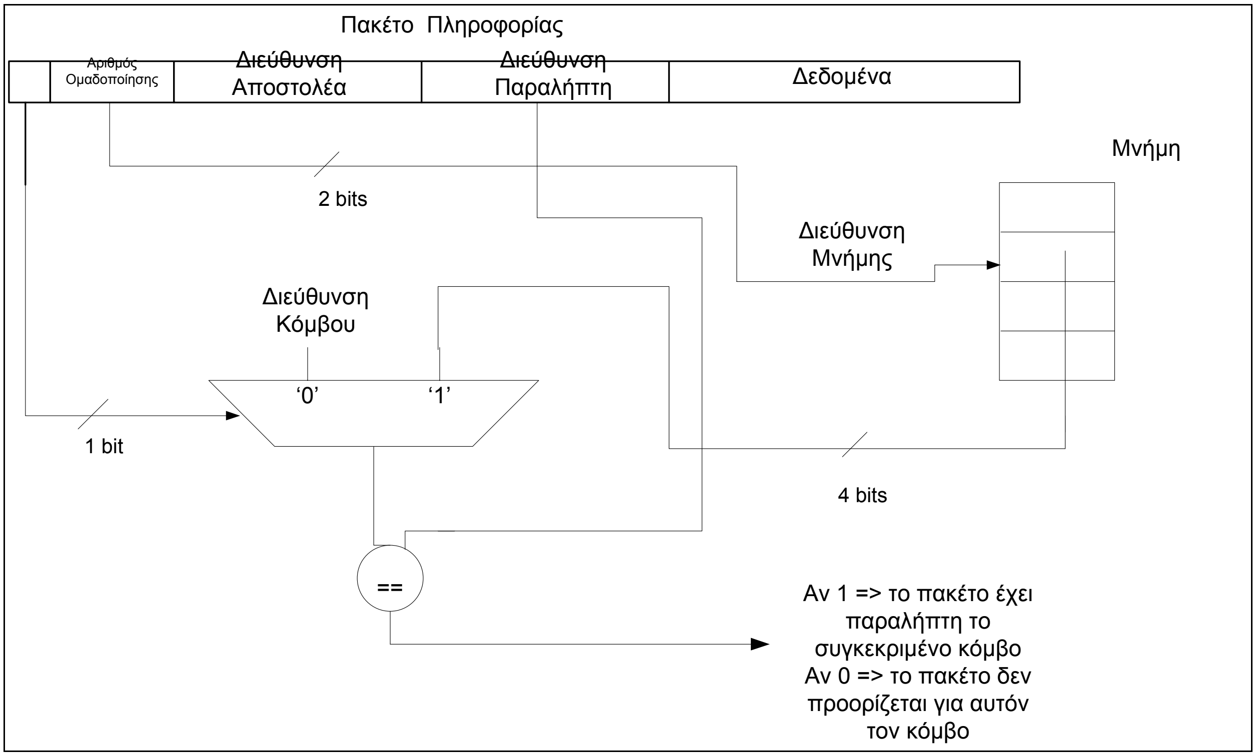
6.3) Δομές υλικού για την υποστήριξη των αλγορίθμων

Όπως αναφέρθηκε και στην εισαγωγή, σημαντικό ρόλο στην υλοποίηση ενός δικτύου ασύρματου παίζει η ενέργεια. Η κατανάλωση της ενέργειας καθορίζεται, κυρίως, από τις δομές που χρησιμοποιούνται για την κατασκευή του δικτύου. Στην συγκεκριμένη εργασία χρησιμοποιήθηκαν διάφορες κωδικοποιήσεις και ομαδοποιήσεις των διευθύνσεων των κόμβων καθώς και διάφορα πρωτόκολλα επικοινωνίας. Σημαντικό ρόλο στο είδος καθώς και στον αριθμό των δομών, που χρησιμοποιούνται από κάθε κόμβο παίζει το μέγεθος της διεύθυνσης του κόμβου. Για αυτό το λόγο θα θεωρήσουμε ότι για όλες τις περιπτώσεις έχουμε ένα ενιαίο παράδειγμα για το οποίο ισχύουν τα παρακάτω: πχ **υπάρχει ένα σύνολο Ν αριθμών κόμβων σε ένα ασύρματο δίκτυο στο οποίο υπάρχουν 4 διαφορετικές ομαδοποιήσεις όπου η κάθε ομαδοποίηση χωρίζει τους κόμβους σε 4 διαφορετικές ομάδες**. Παρακάτω περιγράφονται για κάθε διαφορετικό τρόπο κωδικοποίησης της πληροφορίας των ομάδων οι διάφορες δομές που θα πρέπει να χρησιμοποιηθούν.

6.3.1) Δομές υλικού για weighted binary διευθύνσεις

Όπως αναφέρθηκε σε προηγούμενο Κεφάλαιο, το μέγεθος των διευθύνσεων για ομαδοποιήσεις, που χρησιμοποιούν weighted binary αριθμούς εξαρτώνται από τον αριθμό των ομαδοποιήσεων. Για την συγκεκριμένη περίπτωση, λοιπόν, σε κάθε κόμβο θα χρειαστεί η ύπαρξη μίας μνήμης της οποίας το μέγεθος θα είναι αριθμός ομαδοποιήσεων x αριθμός ομάδων /ομαδοποίηση, αφού όπως έχει ήδη αναφερθεί η διεύθυνση κάθε κόμβου για τις ομαδοποιήσεις αποτελείται από τόσα bits, όσα και ο αριθμός των ομάδων για κάθε ομαδοποίηση.

Αν το πρώτο bit του πακέτου είναι 0 (το πακέτο απευθύνεται σε ένα κόμβο) τότε αυτό σημαίνει ότι θα πρέπει να ελεγχθεί αν η διεύθυνση του κόμβου είναι ίδια με αυτή που υπάρχει στο αντίστοιχο πεδίο στην επικεφαλίδα του πακέτου. Αν, αντίθετα, το bit, αυτό, είναι 1 (το πακέτο απευθύνεται σε ομάδα κόμβων)τότε θα πρέπει να γίνει η σύγκριση μεταξύ των bits της κωδικοποίησης που υπάρχει στην μνήμη του κόμβου και η θέση της οποίας καθορίζεται από τον αριθμό ομαδοποίησης στο αντίστοιχο πεδίο στην επικεφαλίδα του πακέτου. Άρα από τα παραπάνω είναι φανερό ότι χρειάζονται άλλες δύο δομές, ένας συγκριτής των διευθύνσεων και ένας πολυπλέκτης, που θα καθορίζει την τιμή σύγκρισης ανάμεσα στην διεύθυνση του κόμβου και στην διεύθυνση της ομάδας του κόμβου. Μία ενδεικτική συνδεσμολογία των δομών, που θα χρησιμοποιηθούν για την υλοποίηση της διευθυνσιοδότησης των κόμβων με weighted binary αριθμούς είναι αυτή που εμφανίζεται παρακάτω(στην Εικόνα 6.2 έχουν κρατηθεί οι τιμές του παραπάνω παραδείγματος γι’αυτό τον λόγο και η μνήμη του κόμβου έχει μέγεθος 4x4) :



Εικόνα 6.2 : Δομές υλικού και η συνδεσμολογία τους για διευθυνσιοδότηση με weighted binary αριθμούς

6.3.2) Δομές υλικού για διευθυνσιοδοτήσεις των partitional και hierarchical αλγορίθμων

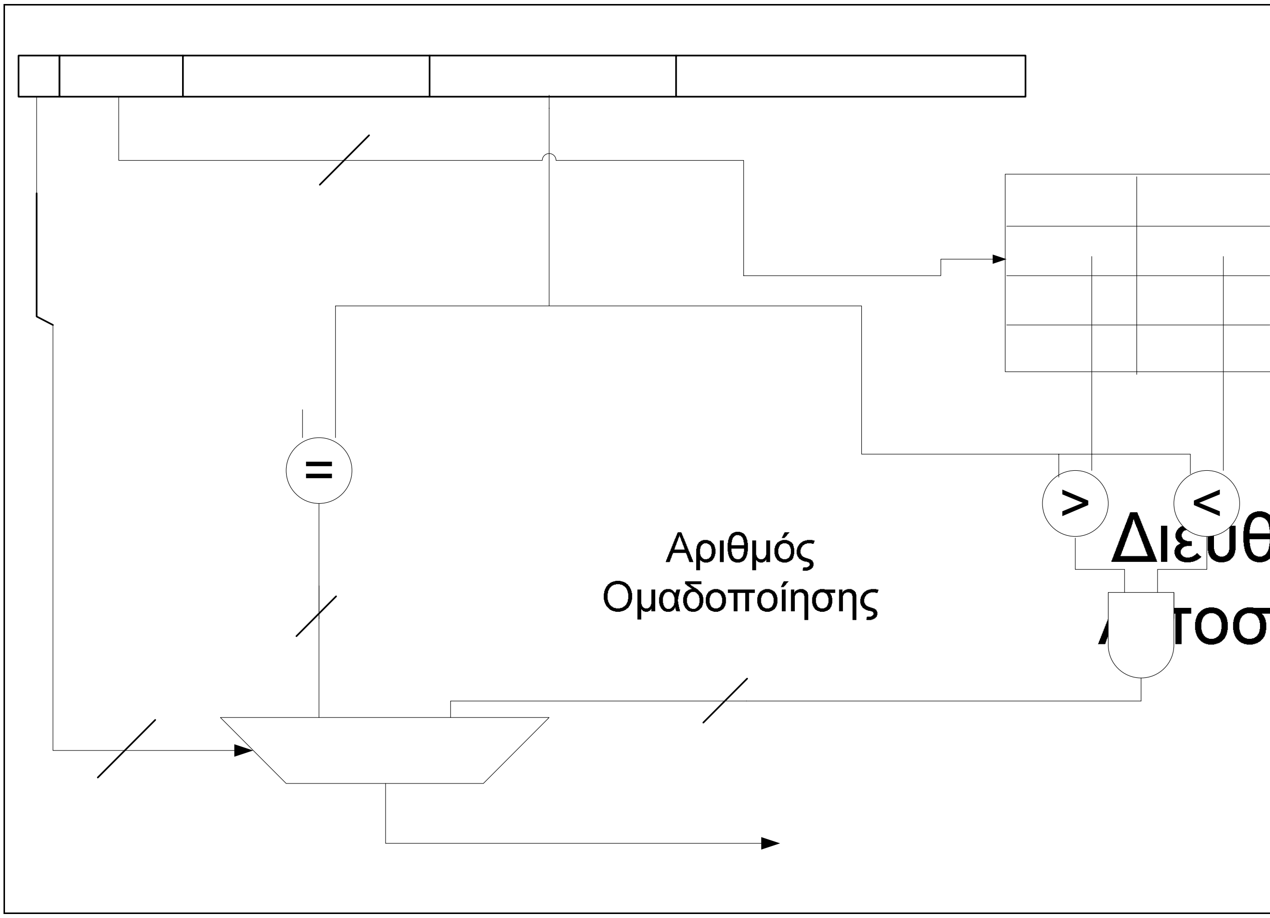
Από τα αποτελέσματα ομαδοποίησης των hierarchical (Agglomerative και Divisive) και των partitional (K-means) μεθόδων μπορεί κάποιος να εξαγάγει το συμπέρασμα ότι αυτές οι μέθοδοι δημιουργούν ομάδες κόμβων, η κάθε μία από τις οποίες περιλαμβάνει διαδοχικά "σημάδια" – διευθύνσεις του κανόνα Golomb. Αυτό το χαρακτηριστικό είναι πολύ σημαντικό αφού μπορεί να γίνει εκμετάλλευση του έτσι ώστε να μειώσει αρκετά τις δομές, που χρειάζονται σε κάθε κόμβο για την αναγνώριση του παραλήπτη του πακέτου.

Μία σημαντική δομή για αυτό τον τρόπο διευθυνσιοδότησης είναι μία μνήμη, η οποία θα έχει μήκος όσος ο αριθμός των διαφορετικών ομαδοποιήσεων αλλά και πλάτος **δύο φορές την μέγιστη διεύθυνση του κανόνα Golomb.**

Αυτό που είναι σημαντικό να τονιστεί είναι ότι η μνήμη αυτή θα έχει στην ουσία διπλάσιο πλάτος αφού στην πρώτη θέση κάθε θέσης μνήμης θα περιλαμβάνεται η μικρότερη διεύθυνση κόμβου, που ανήκει στην ίδια ομάδα με τον συγκεκριμένο κόμβο για την ίδια ομαδοποίηση, ενώ στην δεύτερη θέση θα περιλαμβάνεται η μέγιστη διεύθυνση κόμβου της ομάδας.

Μέγ

Σε περίπτωση που το πακέτο προορίζεται για ομάδα κόμβων θα πρέπει να υπάρχουν δύο συγκριτές και να ελέγχουν αν η διεύθυνση του παραλήπτη είναι ανάμεσα στην μικρότερη και στην μεγαλύτερη διεύθυνση την ομάδας κόμβων που ανήκει ο κόμβος. Επίσης, όπως και παραπάνω, θα πρέπει να υπάρχει και ένας συγκριτής για την περίπτωση που το είχε αποδέκτη κόμβο και όχι ομάδα κόμβων. Τέλος, θα πρέπει να υπάρχει και ένας πολυπλέκτης που να καθορίζει την έξοδο του ελέγχου πάντα ανάλογα με την τιμή του bit, το οποίο καθορίζει τον προορισμό του πακέτου. Μία συνδεσμολογία που θα μπορούσε να χρησιμοποιηθεί για αυτή την περίπτωση διευθυνσιοδότησης είναι αυτή που φαίνεται παρακάτω(**Εικόνα 6.3**)(πρέπει να τονιστεί ότι το σχήμα έγινε με βάση το παραπάνω παράδειγμα, παρ’όλα αυτά δεν είναι γνωστό το πλάτος της μνήμης διότι αυτό καθορίζεται από την μεγαλύτερη διεύθυνση όλων των κόμβων των δικτύων) :



Εικόνα 6.3: Δομές υλικού και η συνδεσμολογία τους για διευθυνσιοδότηση με partitional και hierarchical clustering αλγορίθμους

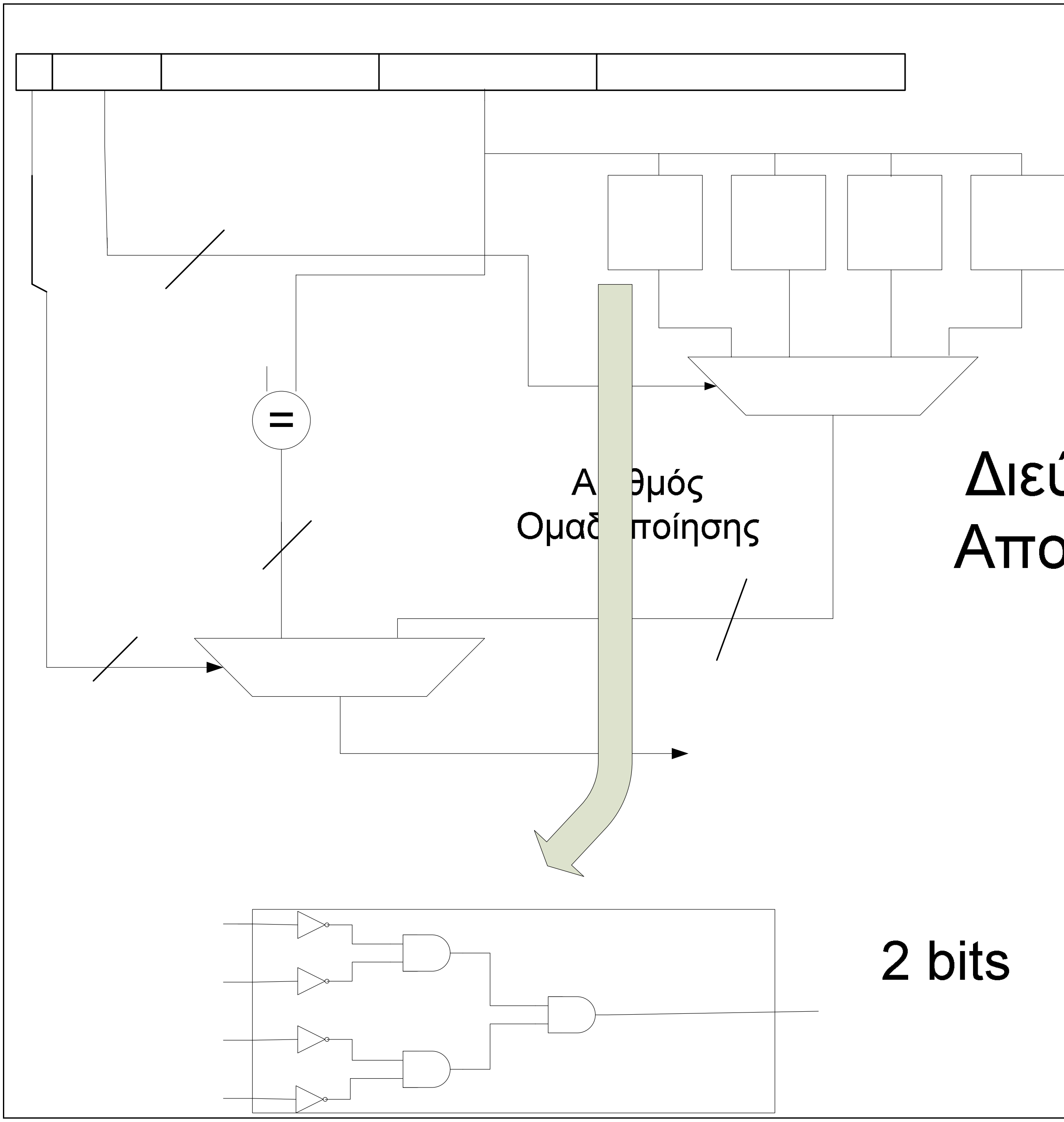
6.3.3) Δομές υλικού για διευθυνσιοδοτήσεις με βάση τον 1ο ευρυστικό αλγόριθμο

Όπως είχε αναφερθεί στην παράγραφο 4.3.2, όπου είχε περιγραφεί η λειτουργία του 1ου ευριστικού αλγορίθμου, ο αλγόριθμος αυτός καταλήγει σε κάθε ομάδα να αντιστοιχίζεται και από μια λογική συνάρτηση. Η λογική συνάρτηση, αυτή, παίρνει ως είσοδο κάποια από τα bits της διεύθυνσης του παραλήπτη από το πακέτο πληροφορίας και στην συνέχεια αν το αποτέλεσμα της είναι λογικό 1 τότε αυτό σημαίνει ότι το πακέτο προορίζεται για την ομάδα, στην οποία ανήκει και ο συγκεκριμένος κόμβος, διαφορετικά το πακέτο δεν αναφέρεται στον συγκεκριμένο κόμβο.

Από τα παραπάνω, και με βάση όλα αυτά που αναφέρθηκαν σε παραπάνω μεθόδους οι δομές που θα χρησιμοποιηθούν είναι περίπου οι ίδιες, ένας πολυπλέκτης 2x1, ένας συγκριτής της διεύθυνσης του κόμβου με την διεύθυνση του αποστολέα αλλά και η υλοποίηση της λογικής συνάρτησης, η οποία καθορίζει αν το πακέτο που έφθασε στον κόμβο προοριζόταν για την ομάδα του συγκεκριμένου κόμβου ή όχι.

Ένα σημαντικό στοιχείο, που δεν αναλύθηκε, είναι η περίπτωση οι ομαδοποιήσεις των κόμβων να είναι παραπάνω από μια. Σε αυτή την περίπτωση, λόγω της απουσίας της μνήμης, θα πρέπει να υπάρχει άλλος ένας πολυπλέκτης, του οποίου οι είσοδοι του θα είναι όσος και ο αριθμός των διαφορετικών ομαδοποιήσεων και η έξοδος του μία, ενώ το σήμα ελέγχου του θα αποτελείται από τον αριθμό της ομαδοποίησης, που βρίσκεται στο αντίστοιχο τμήμα του πακέτου.

Συνεπώς, μία πιθανή συνδεσμολογία σε κάθε κόμβο είναι αυτή που φαίνεται παρακάτω. Πρέπει να σημειωθεί ότι για την υλοποίηση της λογικής συνάρτησης με πύλες θα πρέπει η λογική συνάρτηση να είναι γνωστή και ακολουθεί ένα παράδειγμα υλοποίησης από την **Εικόνα 6.4**,της πρώτης ομάδας.



Εικόνα 6.4 Δομές υλικού και η συνδεσμολογία τους για διευθυνσιοδότηση με βάση τον 1ο ευρυστικό αλγόριθμο

Διεύθυνση
Κόμβου

6.3.4) Δομές υλικού για διευθυνσιοδοτήσεις με βάση τον 2^{ο} ευρυστικό αλγόριθμο

Στον δεύτερο ευρυστικό αλγόριθμο, όπως αναφέρθηκε και στην παράγραφο 3.3.2, μπορούν να χρησιμοποιηθούν δύο διαφορετικά πρωτόκολλα επικοινωνίας. Τα πρωτόκολλα, αυτά, διαφέρουν μόνο ως προς τον αριθμό των bits που θα υπάρχουν στο αντίστοιχο τμήμα του πακέτου όταν αυτό προορίζεται για ομάδα κόμβων. Αυτό, φυσικά, διαφοροποιεί έστω και λίγο τις δομές που χρησιμοποιούνται από τους κόμβους για το πρώτο πρωτόκολλο από εκείνες που χρησιμοποιούνται για το δεύτερο πρωτόκολλο. Ουσιαστικά, όμως, και στις δύο περιπτώσεις που αναφέρθηκαν παραπάνω οι δομές είναι ίδιες ενώ αυτό που αλλάζει είναι κυρίως το μέγεθος της μνήμης καθώς και το μέγεθος των εισόδων ορισμένων πυλών, αφού στην περίπτωση του δεύτερου πρωτοκόλλου χρησιμοποιούνται λιγότερα bits για την αναπαράσταση, στην ουσία, της ίδιας πληροφορίας.

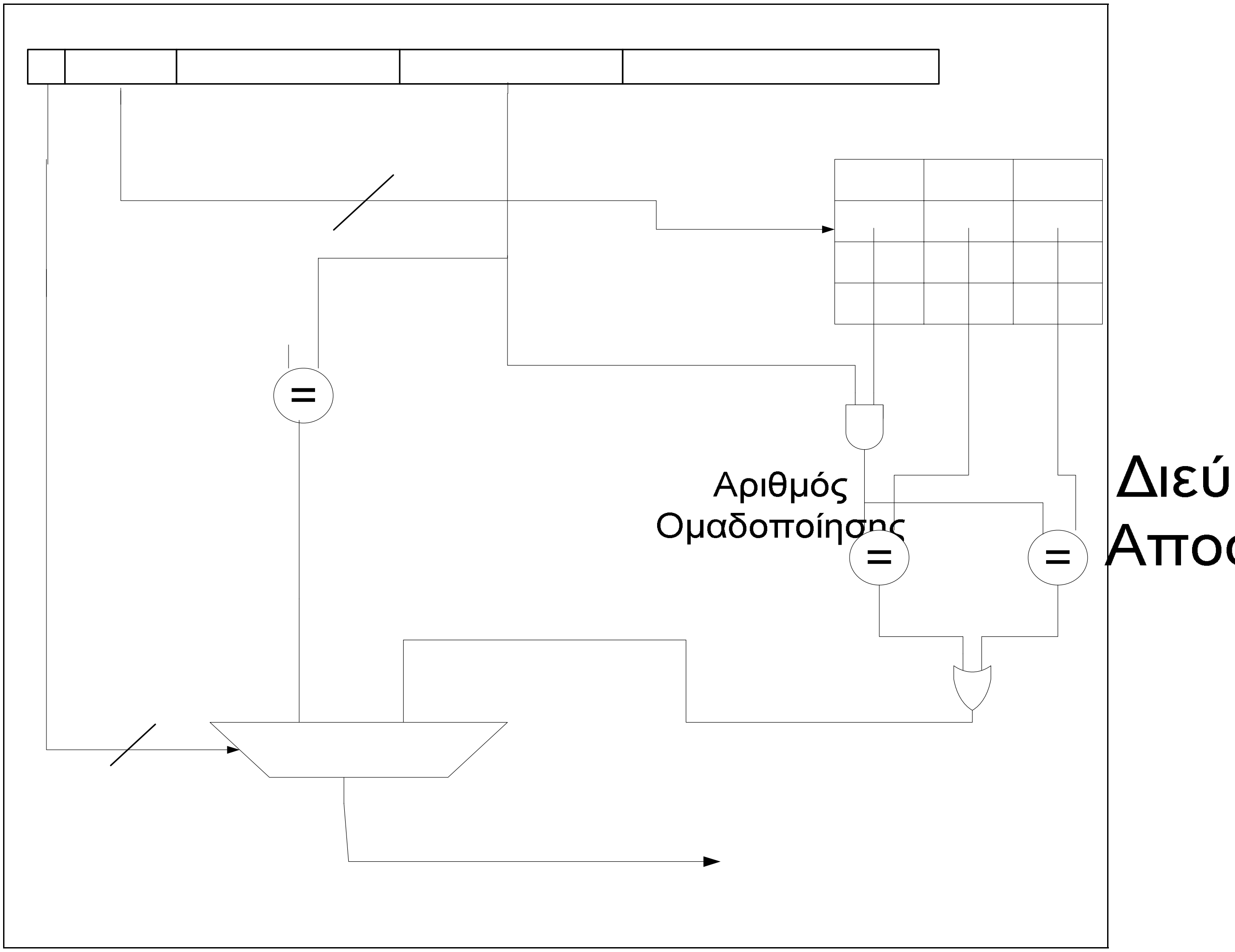
Η βασική δομή που χρειάζεται και σε αυτήν την περίπτωση είναι μία μνήμη, η οποία θα έχει μήκος όσο ο αριθμός των διαφορετικών ομαδοποιήσεων των κόμβων ενώ το πλάτος της, διαφέρει ανάλογα με το πρωτόκολλο που θα χρησιμοποιηθεί αλλά και τον αριθμό των ομάδων που θα πρέπει να χωριστούν οι κόμβοι.

Η κάθε διεύθυνση μνήμης θα αποτελείται από τρεις θέσεις, τα bits της κάθε μία θα είναι όσο το μήκος σε bits της μέγιστης διεύθυνσης των κόμβων. Δηλαδή, με βάση το παράδειγμα της παραγράφου 4.3.3 η μνήμη θα πρέπει να έχει πλάτος 12 bits x 3 = 36 bits. Πρέπει να σημειωθεί ότι από τον τρόπο λειτουργίας του αλγορίθμου, όπως είχε αναφερθεί και παραπάνω, η κάθε ομάδα θα μπορεί να καθορίζεται **πάντα από 1, το πολύ 2,** διαφορετικές διευθυνσιοδοτήσεις. Αυτό είναι ξεκάθαρο, αν δει κάποιος ότι από την λειτουργία του αλγόριθμου κάθε διάσπαση μία ομάδας σε δύο άλλες μπορεί να γίνει με το πολύ δύο bits, ενώ και για κάθε ομάδα που δημιουργείται στο δένδρο (**Εικόνα 4.5**) μπορεί να εξαρτάται από την τιμή μίας και μοναδικής, το πολύ, 2 bits ποσότητας. Με αυτό τον τρόπο για κάθε διεύθυνση μνήμης στην δεύτερη και την τρίτη θέση θα υπάρχουν οι δύο(πιθανόν και μία) διευθυνσιοδοτήσεις της ομάδας, ενώ στην πρώτη θέση θα υπάρχει μία μάσκα από bits, μεγάλη όσο και οι διευθύνσεις, η οποία θα έχει λογικό ‘1’ στα bits με τα οποία γίνεται η διευθυνσιοδότηση και ‘0’ στα υπόλοιπα.

Επίσης, εκτός από την μνήμη μία ακόμη δομή, που θα χρειαστεί είναι μία πύλη AND, η οποία θα έχει ως εισόδους την μάσκα των bits που χρησιμοποιούνται από την πρώτη στήλη και την διεύθυνση του παραλήπτη του πακέτου, έτσι ώστε στην έξοδο της πύλης αυτής να εμφανίζονται μόνο τα bits, που χρειάζονται και με τα οποία θα συγκριθούν οι διευθύνσεις της δεύτερης και της τρίτης στήλης της μνήμης.

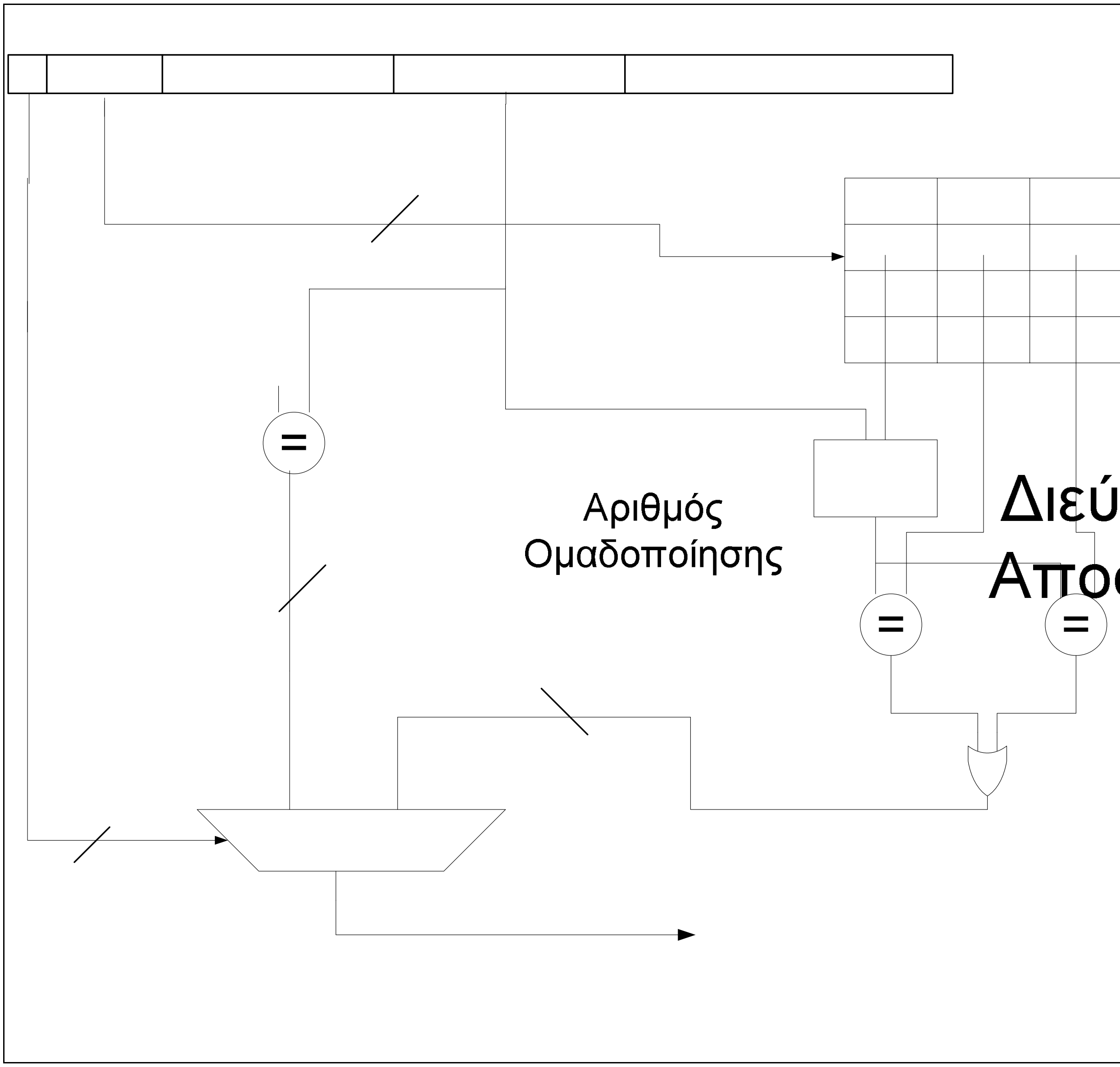
Τέλος, εκτός από δύο συγκριτές, οι οποίοι χρειάζονται για την σύγκριση της παραγόμενης διεύθυνσης με τις διευθύνσεις, που υπάρχουν στην μνήμη, επίσης, χρειάζονται και μία πύλη OR, η οποία θα έχει ως είσοδο τα αποτελέσματα των συγκριτών και στην έξοδο το bit, που θα δείχνει αν το πακέτο αυτό αναφέρεται στην ομάδα του κόμβου που το παρέλαβε.

Η συνδεσμολογία καθώς και οι δομές που χρησιμοποιούνται εμφανίζονται στην παρακάτω εικόνα(**Εικόνα 6.5**):



Εικόνα 6.5 : Δομές υλικού και η συνδεσμολογία τους για διευθυνσιοδότηση με βάση τον 2ο ευρυστικό αλγόριθμο (1ο πρωτόκολλο επικοινωνίας)

Για το δεύτερο πρωτόκολλο επικοινωνίας, η συνδεσμολογία δεν είναι πολύ διαφορετική. Η μόνη, διαφορά που υπάρχει ανάμεσα στις δύο συνδεσμολογίες των δύο πρωτοκόλλων είναι ότι για το δεύτερο πρωτόκολλο λόγο της μείωσης των bits, από τον αριθμό των bits της μεγαλύτερης διεύθυνσης στα bits που είναι απαραίτητα, αντί για την πύλη AND υπάρχει μία δομή η οποία για τις θέσεις των bits, που δεν υπάρχουν στην διεύθυνση του αποδέκτη που αποστέλλεται, τις γεμίζει με λογικό ‘0’. Έτσι η συνδεσμολογία που χρησιμοποιείται για το δεύτερο πακέτο είναι αυτή που φαίνεται παρακάτω(**Εικόνα 6.6**):



Εικόνα 6.6 : Δομές υλικού και η συνδεσμολογία τους για διευθυνσιοδότηση με βάση τον 2^{ο} ευρυστικό αλγόριθμο (2^{ο} πρωτόκολλο επικοινωνίας)

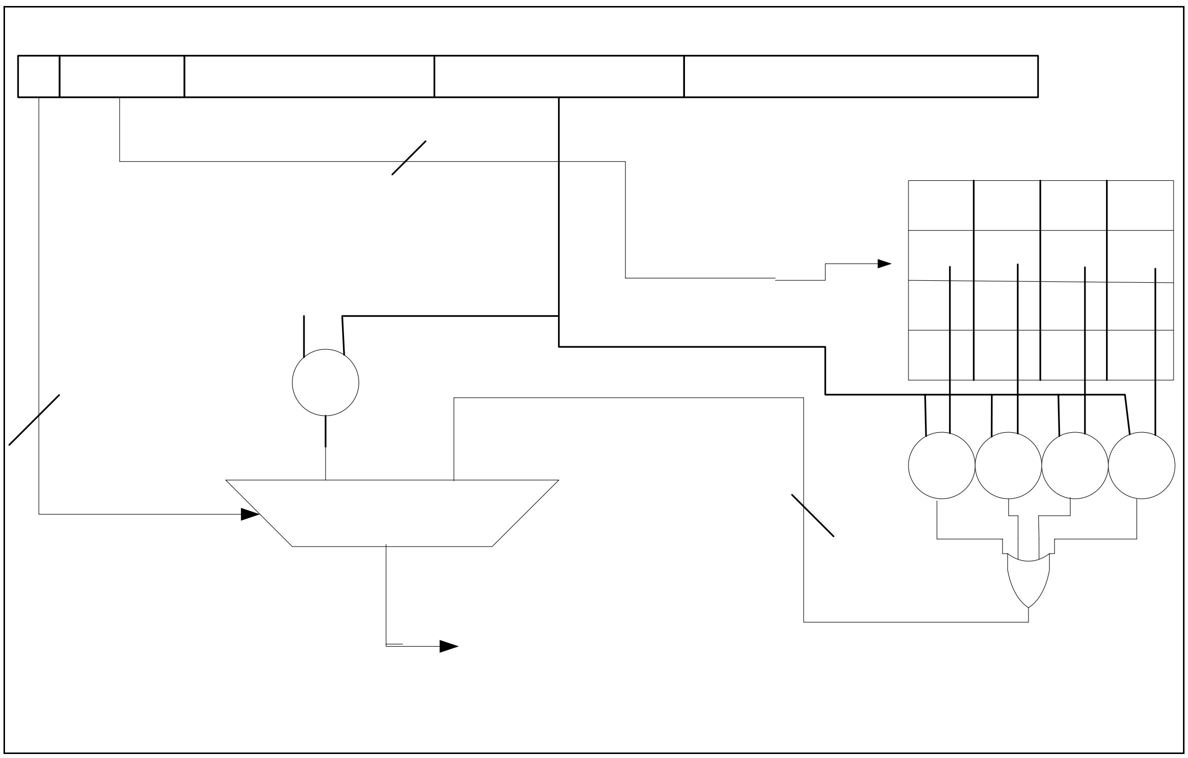
6.3.5) Δομές υλικού για διευθυνσιοδοτήσεις με βάση τον 3^{ο} ευρυστικό αλγόριθμο

Στον 3^{ο} ευρυστικό αλγόριθμο, που υλοποιήθηκε και περιγράφτηκε στο Κεφάλαιο 4, σημαντικό ρόλο παίζουν **μόνο οι διευθύνσεις των αρχηγών των ομάδων**. Λόγω του γεγονότος, λοιπόν, ότι μία ολόκληρη ομάδα ξεχωρίζεται από τις υπόλοιπες μέσω των

Διεύθυνση
Κόμβου

διευθύνσεων των αρχηγών τους, η συνδεσμολογία που χρειάζεται σε κάθε κόμβο είναι πολύ απλή.

Πιο συγκεκριμένα, για αυτή την μέθοδο, η κύρια δομή είναι μία μνήμη στην οποία ο κάθε κόμβος θα κρατάει τις αποστάσεις των αρχηγών των υπόλοιπων ομάδων από τον αρχηγό της ομάδας στην οποία ανήκει ο κόμβος. Προφανώς, η μνήμη θα έχει μήκος τόσες θέσεις όσες και οι διαφορετικές ομαδοποιήσεις που υπάρχουν ενώ το πλάτος της θα είναι όσο ο αριθμός των bits αποστάσεων μεταξύ των αρχηγών. Η γραφική απεικόνιση των δομών και οι συνδεσμολογίες περιγράφονται στην παρακάτω εικόνα(**Εικόνα 6.7**):



Εικόνα 6.7 : Δομές υλικού και η συνδεσμολογία τους για διευθυνσιοδότηση με βάση τον 3^{ο} ευρυστικό αλγόριθμο

6.4) Σύγκριση Αλγορίθμων με βάση τις δομές υλικού κατασκευής τους

Στην παράγραφο αυτή θα γίνει μία σύγκριση ως προς τις δομές, που θα πρέπει να υπάρχουν σε κάθε κόμβο του δικτύου έτσι ώστε να γίνεται επιβεβαίωση από κάθε κόμβο, αν το πακέτο που έχει φτάσει είχε προορισμό αυτόν και την ομάδα στην οποία ανήκει, ή όχι. Η σύγκριση θα γίνει κυρίως ως προς το είδος των δομών και το μέγεθος τους καθώς και ως προς τον χρόνο λειτουργίας του κυκλώματος.

Η κύρια δομή για τις περισσότερες κωδικοποιήσεις, που αναφέρθηκαν παραπάνω, είναι η μνήμη. Οι παραπάνω κωδικοποιήσεις διαφέρουν κυρίως ως προς το μέγεθος της μνήμης που χρησιμοποιούν, ενώ υπάρχουν και κάποιες που δεν χρησιμοποιούν καθόλου μνήμη. Παρακάτω, ακολουθεί μία ανάλυση για το μέγεθος της μνήμης σε κάθε περίπτωση και μία σύγκριση μεταξύ τους.

- **Μέθοδος των weighted binary αριθμών**

  Μέγεθος μνήμης = αριθμός ομαδοποιήσεων * αριθμό ομάδων ανά ομαδοποίηση (bits)

- **Patirtional και hierarchical αλγόριθμοι**

  Μέγεθος μνήμης = αριθμός ομαδοποιήσεων * (2 * αριθμό bits της μεγαλύτερης διεύθυνσης των κόμβων) (bits)

- **1^{ος} Ευρυστικός Αλγόριθμος**

  Ο αλγόριθμος αυτός δεν κάνει χρήση μνήμης.

- **2^{ος} Ευρυστικός Αλγόριθμος (1^{ο} Πρωτόκολλο Επικοινωνίας)**

  Μέγεθος μνήμης = αριθμός ομαδοποιήσεων * (3 * αριθμό bits της μεγαλύτερης διεύθυνσης των κόμβων) (bits)

- **2^{ος} Ευρυστικός Αλγόριθμος (2^{ο} Πρωτόκολλο Επικοινωνίας)**

  Μέγεθος μνήμης = αριθμός ομαδοποιήσεων * (2 * αριθμό bits της μεγαλύτερης διεύθυνσης των κόμβων + αριθμό bits που χρησιμοποιούνται για την ομαδοποίηση των ομάδων) (bits)

- **3^{ος} Ευρυστικός Αλγόριθμος**

  Μέγεθος μνήμης = αριθμός ομαδοποιήσεων * (αριθμός ομάδων * αριθμό bits της μεγαλύτερης απόστασης μεταξύ των διευθύνσεων των αρχηγών των ομάδων) (bits)

Πρέπει να αναφερθεί, ότι στον υπολογισμό των μνημών δεν συμπεριλαμβάνονται οι μνήμες που περιλαμβάνουν τις αποστάσεις των κόμβων μεταξύ τους, λόγω του γεγονότος ότι μία τέτοια μνήμη θα ήταν αναγκαία για όλες τις περιπτώσεις των αλγορίθμων.

Αν πάρουμε για παράδειγμα, ένα δίκτυο αισθητήρων που έχει 16 κόμβους και για την διευθυνσιοδότηση του έναν κανόνα Golomb, που έχει μέγεθος N = 16 “σημάδια” (0 13 14 23 42 53 86 89 107 111 113 148 156 163 168 194). Έστω ότι αυτοί οι κόμβοι πρέπει να χωριστούν σε 4 διαφορετικές ομάδες για κάθε ομαδοποίηση ενώ γενικά υπάρχουν 4 διαφορετικές ομαδοποιήσεις για του κόμβους. Έτσι τα μεγέθη των μνημών, που θα χρησιμοποιηθούν ανάλογα με τον αλγόριθμο κωδικοποίησης παρουσιάζονται στον παρακάτω πίνακα :

Αλγόριθμος	**Μέγεθος Μνήμης**
Διευθυνσιοδότηση με weighted binary αριθμούς	4 * 4 = **16 bits**
Patirtional και hierarchical αλγόριθμοι	4 * (2 * 8) = **64 bits**
1^{ος} Ευρυστικός Αλγόριθμος	**-**
2^{ος} Ευρυστικός Αλγόριθμος (1^{ο} Πρωτόκολλο Επικοινωνίας)	4 * (3 * 8) = **96 bits**
2^{ος} Ευρυστικός Αλγόριθμος (2^{ο} Πρωτόκολλο Επικοινωνίας)	4 * (2 * 8 + 3) = **76 bits**
3^{ος} Ευρυστικός Αλγόριθμος	4 * (4 * 5) = **80 bits**

Εικόνα 6.8 : Πίνακας του μεγέθους της απαιτούμενης μνήμης για όλους τους αλγορίθμους για το παράδειγμα του κεφαλαίου

Εκτός από τις μνήμες στα κυκλώματα, που περιγράφηκαν στις παραπάνω παραγράφους, υπάρχουν και κάποια άλλα κυκλώματα που αποτελούνται κυρίως από λογικές πύλες. Πιο συγκεκριμένα υπάρχουν κυκλώματα συγκριτών, πολυπλεκτών καθώς και κάποιες μεμονομένες λογικές πύλες. Ο αριθμός των κάθε μία μονάδων από αυτές που αναφέρθηκαν, περιγράφονται στον παρακάτω πίνακα :

Αλγόριθμος	**Συγκριτές**	**Πολυπλέκτες**	**Λογικές Πύλες**
Διευθυνσιοδότηση με weighted binary αριθμούς	1	1	-
Patirtional και hierarchical αλγόριθμοι	3	1	1 x AND
1ος Ευρυστικός Αλγόριθμος	1	2	Υλοποίηση Λογικών Συναρτήσεων
2ος Ευρυστικός Αλγόριθμος (1ο Πρωτόκολλο Επικοινωνίας)	3	1	1 x AND 1 x OR
2ος Ευρυστικός Αλγόριθμος (2ο Πρωτόκολλο Επικοινωνίας)	3	1	1 x OR
3ος Ευρυστικός Αλγόριθμος	5	1	-

Εικόνα 6.9 : Πίνακας των απαιτούμενων δομών για τους αλγορίθμους

Εκτός από την σύγκριση των δομών καθώς και το μέγεθος τους, μία άλλη σύγκριση είναι αυτή του χρόνου που χρειάζεται κάθε ένα από τα παραπάνω κυκλώματα να αποκριθεί από την στιγμή που θα φτάσει σε ένα κόμβο ένα πακέτο πληροφορίας. Ο χρόνος, που υπολογίζεται δεν είναι δυνατόν να είναι ακριβής διότι και οι δομές, που χρησιμοποιήθηκαν δεν είναι σταθερές αλλά εξαρτώνται από τον αριθμό των κόμβων καθώς και το μέγεθος των διευθύνσεων. Για τον λόγο, αυτό, οι χρόνοι στους οποίους αναφερόμαστε παρακάτω είναι παραμετρικοί :

- **Μέθοδος των weighted binary αριθμών**

  Χρόνος απόκρισης = $T_{RMEM} + T_{MUX} + T_{COMP}$

- **Patirtional και hierarchical αλγόριθμοι**

  Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{AND} + T_{MUX}$

- **1ος Ευρυστικός Αλγόριθμος**

  Χρόνος απόκρισης = $T_{Logic\_Function} + 2 * T_{MUX}$

- **$2^{ος}$ Ευρυστικός Αλγόριθμος ($1^{ο}$ Πρωτόκολλο Επικοινωνίας)**
  Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{MUX} + T_{AND} + T_{OR}$

- **$2^{ος}$ Ευρυστικός Αλγόριθμος ($2^{ο}$ Πρωτόκολλο Επικοινωνίας)**
  Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{MUX} + T_{OR}$

- **$3^{ος}$ Ευρυστικός Αλγόριθμος**
  Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{MUX} + T_{OR}$

Από τα αποτελέσματα που προέκυψαν από τις παραπάνω αναλύσεις μπορεί κανείς να παρατηρήσει ότι οι διευθυνσιοδοτήσεις με βάση τους κανόνες Golomb δεν δίνουν πολύ χειρότερα αποτελέσματα ως προς την διευθυνσιοδότηση με weighted binary αριθμούς. Πιο συγκεκριμένα παρακάτω ακολουθεί κάποια σύγκριση ανάμεσα στις διάφορες μεθόδους διευθυνσιοδοτήσεων:

Πρώτα απ'όλα παρατηρείται ότι καλύτερη μέθοδος ως προς τις δομές που χρησιμοποιούνται είναι σίγουρα η διευθυνσιοδότηση με weighted binary αριθμούς, η οποία όμως δεν απέχει πάρα πολύ από τις υπόλοιπες μεθόδους. Στην συνέχεια, αμέσως καλύτερη μέθοδος, περισσότερο ως προς το μέγεθος της μνήμης, είναι οι partitional και hierarchical αλγόριθμοι ενώ αμέσως καλύτερος αλγόριθμος είναι ο $2^{ος}$ ευρυστικός αλγόριθμος με το $2^{ο}$ πρωτόκολλο επικοινωνίας. Ο επόμενος αλγόριθμος είναι, επίσης, ο $2^{ος}$ ευρυστικός αλλά με το $1^{ο}$ πρωτόκολλο επικοινωνίας ενώ με βάση τις δομές σε κάθε κόμβο χειρότεροι θεωρούνται οι αλγόριθμοι, $3^{ος}$ και $1^{ος}$ ευρυστικός, από τους οποίους ο μεν πρώτος, για το συγκεκριμένο παράδειγμα χρησιμοποιεί την μεγαλύτερη μνήμη από όλους αλλά και αρκετές λογικές πύλες λόγω των πολλών συγκριτών, ενώ ο δεύτερος χρησιμοποιεί μεγάλο αριθμό πυλών για την υλοποίηση των λογικών συναρτήσεων, κερδίζοντας βέβαια ως προς το γεγονός ότι δεν χρησιμοποιεί καθόλου μνήμη.

Εκτός φυσικά από τον ρόλο των δομών, σημαντικό ρόλο παίζει και ο χρόνος απόκρισης του κυκλώματος. Στην περίπτωση των χρόνων ο αλγόριθμος που φαίνεται να έχει τον μικρότερο χρόνο απόκρισης είναι ο $1^{ος}$ ευριστικός, λόγω της μη ανάγκης για ανάγνωση από μνήμη, η οποία είναι αρκετά χρονοβόρα διαδικασία. Αμέσως μετά ακολουθεί η μέθοδος διευθυνσιοδότησης με weighted binary αριθμούς και αμέσως μετά με τον ίδιο χρόνο απόκρισης είναι οι αλγόριθμοι, ο $2^{ος}$ ευρυστικός(με το $2^{ο}$ πρωτόκολλο επικοινωνίας), ο $3^{ος}$ ευρυστικός αλγόριθμος και τέλος οι partitional και hierarchical αλγόριθμοι. Τελευταίος, από μεριάς χρόνου απόκρισης, είναι ο $2^{ος}$ ευρυστικός αλγόριθμος διευθυνσιοδότησης (με το $1^{ο}$ πρωτόκολλο επικοινωνίας). Παρακάτω ακολουθεί πίνακας που περιλαμβάνει τους αλγόριθμους(από τον καλύτερο προς τον χειρότερο) με βάση τα συγκεντρωτικά στοιχεία και των δομών αλλά και των χρόνων απόκρισης:

Α/Α	**Αλγόριθμος**	**Μνήμη(bits)** (για το συγκεκριμένο παράδειγμα)	**Συγκριτές**	**Πολυπλέκτες**	**Λογικές Πύλες**	**Χρόνος Απόκρισης**
1	Διευθυνσιοδότηση με weighted binary αριθμούς	16	1	1	-	Χρόνος απόκρισης = $T_{RMEM} + T_{MUX} + T_{COMP}$
2	Patirtional και hierarchical αλγόριθμοι	64	3	1	1 x AND	Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{AND} + T_{MUX}$
3	2ος Ευρυστικός Αλγόριθμος (2ο Πρωτόκολλο Επικοινωνίας)	76	3	1	1 x OR	Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{MUX} + T_{OR}$
4	1ος Ευρυστικός Αλγόριθμος	-	1	2	Υλοποίηση Λογικών Συναρτήσεων	Χρόνος απόκρισης = $T_{Logic\_Function} + 2 * T_{MUX}$
5	2ος Ευρυστικός Αλγόριθμος (1ο Πρωτόκολλο Επικοινωνίας)	96	3	1	1 x AND 1 x OR	Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{MUX} + T_{AND} + T_{OR}$
6	3ος Ευρυστικός Αλγόριθμος	80	5	1	-	Χρόνος απόκρισης = $T_{RMEM} + T_{COMP} + T_{MUX} + T_{OR}$

Πίνακας 6.10 : Συγκεντρωτικός πίνακας με το σύνολο των δομών και τον χρόνο απόκρισης του κυκλώματος για όλους τους αλγορίθμους

ΚΕΦΑΛΑΙΟ 7^{ο}

Συμπεράσματα

Η εργασία είχε ως στόχο την μελέτη της χρήσης των "σημαδιών" των κανόνων Golomb, έτσι όπως προέκυπταν από την διπλωματική του κ. Δημητρομανωλάκη [1], ως διευθύνσεις σε ασύρματα δίκτυα αισθητήρων. Επίσης, ένας σημαντικός στόχος ήταν η εύρεση αλγορίθμων και τρόπων ομαδοποίησης, έτσι ώστε η ομαδοποίηση των κόμβων με διευθύνσεις τα "σημάδια" των κανόνων Golomb να είναι καλύτερη από την ομαδοποίηση με weighted binary αριθμούς.

Γενικά, όλη η προσπάθεια κινήθηκε γύρω από δύο κεντρικούς άξονες: ο πρώτος άξονας είναι ότι στα ασύρματα δίκτυα σημαντικό ρόλο κατέχει η κατανάλωση της ενέργειας, η οποία οφείλεται κυρίως στην ασύρματη μετάδοση, επομένως πρωτεύων σκοπός ήταν χωρίς να χαθεί πληροφορία να μειωθεί ο αριθμός των bits σε κάθε μετάδοση. Ο δεύτερος άξονας, στον οποίο στηρίχθηκε η εργασία ήταν η τρόπος κατασκευής των κανόνων Golomb και η κύρια ιδιότητα τους, που είναι η μοναδική απόσταση ανάμεσα στα σημάδια.

Από τους αλγόριθμους, που χρησιμοποιήθηκαν στην συγκεκριμένη εργασία, κάποιοι ήταν έτοιμοι ενώ κάποιοι κατασκευάστηκαν με βάση την μείωση του αριθμού των bits. Γενικά, από τα αποτελέσματα των συγκρίσεων μεταξύ των αλγορίθμων, όπως αυτά παρουσιάζονται σε προηγούμενα κεφάλαια, μπορεί να παρατηρηθεί ότι ορισμένοι αλγόριθμοι δίνουν καλύτερα αποτελέσματα από τους weighted binary αριθμούς ενώ για κάποιους άλλους αυτό δεν ισχύει. Επίσης, ένα άλλο σημαντικό στοιχείο της έρευνας που προέκυψε είναι ότι μπορεί κάποιοι αλγόριθμοι να είναι καλοί ως προς τον αριθμό των bits/ μετάδοση αλλά οι δομές, που χρειάζονται για την υλοποίηση ή την λειτουργία τους να είναι περισσότερες από ότι άλλοι αλγόριθμοι.

Γενικά, λοιπόν, η εργασία αυτή δίνει σε γενικές γραμμές την συμπεριφορά κάποιων αλγορίθμων ομαδοποίησης ως προς τους κανόνες Golomb, την σύγκριση μεταξύ των διαφόρων αλγορίθμων ως προς τον αριθμό των bits ανά μετάδοση αλλά και την χρήση των δομών για κάθε έναν από τους αλγόριθμους.

Μελλοντική Εργασία

Γενικά, υπάρχουν τμήματα της παραπάνω εργασίας, που μπορούν να αναλυθούν περισσότερο ή ακόμα και να μελετηθούν. Παρακάτω αναφέρονται κάποιες ιδέες, οι οποίες μπορούν να στηριχτούν στην παραπάνω ανάλυση και με βάση αυτή είτε να επεκταθούν σε άλλους τομείς, κυρίως των ασύρματων δικτύων, είτε να κάνουν μία μελέτη των κανόνων Golomb, ως διευθύνσεις των κόμβων ασύρματων δικτύων από κάποια άλλη άποψη:

- Μία ιδέα είναι η έρευνα της συμπεριφοράς των παραπάνω αλγορίθμων με τις οποίες ασχολήθηκε αυτή η εργασία με μεγαλύτερο αριθμό κόμβων. Αυτή η εργασία περιορίστηκε κυρίως σε αριθμό κόμβων δικτύου **μέχρι 2000**, το οποίο οφείλεται στο γεγονός των πολύ χρονοβόρων αλγορίθμων αλλά και στην πλατφόρμα πάνω στην οποία υλοποιήθηκαν(MATLAB). Επίσης, υπάρχουν ακόμα αρκετοί αλγόριθμοι, οι οποίοι δεν εξετάσθηκαν σε αυτή την εργασία, και μία ευρύτερη συμπεριφορά των αλγορίθμων θα ήταν χρήσιμη.

- Ένα άλλο θέμα που έχει μεγάλη σημασία για τα ασύρματα δίκτυα είναι η ενέργεια. Έτσι, μία καλή ιδέα που θα μπορούσε να εμπλουτίσει αυτή την εργασία είναι η αναλυτική έρευνα πάνω στην κατανάλωση έρευνας σε κάθε κόμβο κατά την εφαρμογή κάθε αλγόριθμου.

- Μία άλλη ιδέα για επέκταση της εργασίας, είναι η κατασκευή ενός ασύρματου δικτύου και η εφαρμογή των αλγορίθμων που υλοποιήθηκαν σε αυτό. Αυτό μπορεί να οδηγήσει σε μία σύγκριση μεταξύ των θεωρητικών και πραγματικών αποτελεσμάτων.

- Τέλος, μία άλλη ενδιαφέρουσα πρόταση είναι ο συνδυασμός της θεωρίας για χρωματισμό γράφων (Graph coloring) με την έρευνα πάνω σε ασύρματα δίκτυα με διευθύνσεις από κανόνες Golomb και πως μπορεί αυτή να βοηθήσει στην καλύτερη απόδοση του δικτύου.

ΑΝΑΦΟΡΕΣ

Πηγές Internet

[2] *The official Mathworks Company site, www.mathworks.com*

[5] *Hierarchical Clustering*
http://www.resample.com/xlminer/help/HClst/HClst_intro.htm

[6] *A tutorial on Clustering Algorithms*
http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/index.html

[9] Golomb Ruler http://en.wikipedia.org/wiki/Golomb_ruler

Βιβλιογραφία

[1] *Απόστολος Δημητρομανωλάκης, Διπλωματική Εργασία, "Analysis of the Golomb Ruler and the Sidon Set Problems, and Determination of Large, Near-Optimal Golomb Rulers ", Πολυτεχνείο Κρήτης, Τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Ηλεκτρονικών Υπολογιστών, Εργαστήριο Μικροεπεξεργαστών και Υλικού*

[3] *Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A Survey on Sensor Networks. Georgia Institute of Technology*

[4] *Praveen Rentala, Ravi Musunuri, Shashidhar Gandham, Udit Saxena. Survey on Sensor Networks. Department of Computer Science University of Texas at Dallas*

[7] *Cheong Hee Park "Clustering: k means and Hierarchical Clustering ", παρουσίαση για το μάθημα "Multimedia Application", Dept. of Computer Science and Engineering Chungnam National University*

[8] *Apostolos Dollas, William T. Rankin, David McCracken. New Algorithms for Golomb Ruler Derivation and Proof of the 19 Mark Ruler. Dept_ of Electrical Engineering Duke University, Durham*

[10] *Sergio M. Savaresi and Daniel L. Boley "On the performance of bisecting K-means and PDDP". Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milan, Italy, Department of Computer Science and Engineering, University of Minnesota, USA*

[11] *Deepak Ganesan, Alberto Cerpa, Yan Yu, Deborah Estrin, Wei Ye and Jerry Zhao. Networking Issues in Wireless Sensor Networks. Journal of Parallel and*

Distributed Computing (JPDC), Special issue on Frontiers in Distributed Sensor Networks, Elsevier Publishers.

[12] *Gyula Simon, Miklós Maróti, Ákos Lédeczi, György Balogh, Branislav Kusy, András Nádas, Gábor Pap, János Sallai, Ken Frampton. Sensor Network-Based Countersniper System. Institute for Software Integrated Systems Vanderbilt University*

[13] *Emil Jovanov, Dejan Raskovic, John Price, John Chapman, Anthony Moore, Abhishek Krishnamurthy. Patient Monitoring Using Personal Area Networks of Wireless Intelligent Sensors. Electrical and Computer Engineering Department, University of Alabama in Huntsville*

[14] *Bjørn Thorstensen, Tore Syversen, Trond-Are Bjørnvold, Tron Walseth. Electronic Shepherd – A Low-Cost, Low-Bandwidth, Wireless Network System. Telenor R&D*

[15] *Athanasios Boulis, Paul Lettieri, Mani Srivastava. Active Base Stations and Nodes forWireless Networks. UCLA-EE Department, Los Angeles, USA, Broadcom Corporation, El Segundo, USA*

[16] *J. M. Kahn, R. H. Katz (ACM Fellow), K. S. J. Pister. Next Century Challenges: Mobile Networking for "Smart Dust". Department of Electrical Engineering and Computer Sciences, University of California, Berkeley*

[17] *Michael J. Dong, K. Geoffrey Yung, and William J. Kaiser. Low Power Signal Processing Architectures for Network Microsensors. University of California, Los Angeles - Los Angeles, California*

[18] *Dr Alexander Strehl. "Relationship-based Clustering and Cluster Ensembles for High-dimensional Data Mining"*

[19] *Vlasios Tsiatsis, Scott A. Zimbeck, Mani B. Srivastava. Architecture Strategies for Energy-Efficient Packet Forwarding in Wireless Sensor Networks. Networked & Embedded Systems Laboratory, E.E. Dept, UCLA, Los Angeles, CA*

[20] *Kafil M. Razeeb, Stephen Bellis, Brendan O'Flynn, John Barton, Kieran Delaney and Cian O'Mathuna. A Hybrid Network of Autonomous Sensor Nodes. NMRC, University College Cork, Lee Maltings, Prospect Row, Ireland*

[21] *Basil Etefia. "Routing Protocols for Wireless Sensor Networks". Electrical Engineering Loyola Marymount University*

[22] *Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, Kristofer Pister. System Architecture Directions for Networked Sensors. Department of Electrical Engineering and Computer Sciences University of California, Berkeley, CA*