ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ

ΘΕΜΑ:

«ΔΙΑΤΗΡΗΣΗ ΙΔΙΩΤΙΚΟΤΗΤΑΣ ΚΑΤΑ ΤΗΝ ΣΥΣΤΑΔΟΠΟΙΗΣΗ ΤΡΟΧΙΩΝ ΚΙΝΟΥΜΕΝΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ»

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ονοματεπώνυμο: Μυστρίδης Παναγιώτης
ΑΕΜ: 113

Επιβλέπων Καθηγητής: Βερύκιος Βασίλειος

Περιεχόμενα

Εικόνες

Πίνακες

Ψευδό-Κώδικες

Κεφάλαιο 1

Εισαγωγή

Η διατήρηση της ιδιωτικότητας των δεδομένων κατά την μοντελοποίηση και εξόρυξη γνώσης αποτελεί μια νέα ερευνητική προσπάθεια η οποία συνεχώς κερδίζει έδαφος σε σχέση με άλλους ερευνητικούς τομείς. Οι βασικοί λόγοι για την ταχεία ανάπτυξη της έρευνας αυτής είναι η ραγδαία και συνεχής πρόοδος στις υπολογιστικές επιστήμες, οι οποίες έχουν πλέον εισχωρήσει σε κάθε πτυχή της καθημερινής μας ζωής, όπως επίσης και στον τεράστιο όγκο δεδομένων τα οποία καθημερινά συσσωρεύονται σε βάσεις δεδομένων. Οι βάσεις περιέχουν δεδομένα διαφορετικού τύπου, καθώς μπορεί να ανήκουν σε οποιοδήποτε πάροχο υπηρεσιών, κάτοχο δεδομένων, όπως νοσοκομεία και να αφορούν καρτέλες ασθενών, τράπεζες με στοιχεία πελατών η ακόμα και καταστήματα με δεδομένα που να αφορούν τις αγοραστικές συνήθειες των πελατών. Επιπλέον η διαρκής αύξηση των δεδομένων, οφείλεται και στην εξάπλωση του διαδικτύου, όπου οι πληροφορίες είναι άμεσα διαθέσιμες και ολοένα και πιο εύκολα προσβάσιμες σε εκατομμύρια χρήστες .Όπως γίνεται αντιληπτό, οι βάσεις δεδομένων κρύβουν σημαντικές πληροφορίες για την προσωπική κατάσταση των πελατών και επομένως πρέπει να προστατευτούν, όμως κρύβουν και χρήσιμες πληροφορίες, οι οποίες αν αξιοποιηθούν κατάλληλα θα αποβούν σε όφελος τόσο των κατόχων όσο και των ίδιων των πελατών. Το πρόβλημα της ιδιωτικότητας των δεδομένων των χρηστών μπορεί να γίνει ακόμα πιο εμφανές σε περιπτώσεις όπου απειλείται η θέση των χρηστών. Η ραγδαία ανάπτυξη της κινητής τηλεφωνίας, η χρήση συστημάτων δορυφορικής πλοήγησης και γενικότερα οι υπηρεσίες ασύρματης επικοινωνίας, είναι μερικές εφαρμογές που ελλοχεύουν τον κίνδυνο αποκάλυψης ανά πάσα στιγμή της θέσης των χρηστών καθώς και τη χρονική στιγμή στην οποία βρίσκονται στη θέση αυτή.

Με τις διαθέσιμες τεχνικές της μοντελοποίησης δεδομένων όπως κατηγοριοποίηση, συσταδοποίηση, δέντρα απόφασης, κανόνες συσχέτισης μπορούμε να εξάγουμε την απαραίτητη πληροφορία από τα δεδομένα. Ωστόσο τα δεδομένα στις περισσότερες περιπτώσεις δεν ανήκουν μόνο σε έναν κάτοχο δεδομένων, αλλά διαμοιράζονται σε περισσότερους του ενός. Το γεγονός αυτό περιπλέκει την κατάσταση αφού για να χρησιμοποιήσουμε τις τεχνικές της μοντελοποίησης πρέπει να γνωρίζουμε ολόκληρη την βάση δεδομένων. Οι κάτοχοι δεδομένων όμως ποτέ δεν θα συμφωνούσαν να μοιραστούν τα δεδομένα τους, ακόμα και αν επωφελούνταν από τα αποτελέσματα της μοντελοποίησης, αφού όπως αναφέραμε τα δεδομένα περιέχουν κρίσιμες πληροφορίες οι οποίες δεν θα ήθελαν να περάσουν σε χέρια ανταγωνιστών.

Ο σκοπός της εργασίας αυτής είναι η υλοποίηση μεθόδων για την ασφαλή ανταλλαγή δεδομένων μεταξύ των κατόχων, με το μικρότερο δυνατό υπολογιστικό κόστος και σαφώς με τη μεγαλύτερη ασφάλεια των δεδομένων.

Πιο συγκεκριμένα στην παρούσα εργασία επικεντρωνόμαστε στην περίπτωση των χώρο-χρονικών δεδομένων, δηλαδή δεδομένων που αφορούν την θέση των αντικειμένων, και πιο συγκεκριμένα στην συσταδοποίηση των τροχιών κινούμενων αντικειμένων. Θεωρούμε ότι τα δεδομένα, δηλαδή οι τροχιές των κινούμενων αντικειμένων, διαμοιράζονται μεταξύ των κατόχων δεδομένων. Αρχικά υλοποιούμε ένα πρωτόκολλο ασφαλούς ανταλλαγής δεδομένων [24] και προτείνουμε ένα δεύτερο πρωτόκολλο το οποίο εγγυάται σε μεγαλύτερο βαθμό την ασφάλεια των δεδομένων. Επιπλέον με το δεύτερο πρωτόκολλο προσπαθούμε να ελαχιστοποιήσουμε τις

υποθέσεις του πρώτου πρωτοκόλλου. Επίσης έχουμε κατασκευάσει μια απλή και εύχρηστη εργαλειοθήκη για την σύγκριση των δύο πρωτοκόλλων, με εναλλακτικές επιλογές για τη βάση δεδομένων με συνθετικά και πραγματικά δεδομένα, τις συναρτήσεις σύγκρισης των τροχιών των κινούμενων αντικειμένων καθώς και για τους αλγόριθμους συσταδοποίησης, ούτως ώστε να μπορέσουμε να εξάγουμε χρήσιμα συμπεράσματα..

Αναλυτικότερα η διάρθρωση της εργασίας έχει ως εξής. Στο κεφάλαιο 2 παρουσιάζεται η σχετική βιβλιογραφία. Στο κεφάλαιο 3 γίνεται μια εισαγωγή στις βασικές έννοιες που θα χρησιμοποιηθούν στην εργασία, όπως στον ορισμό των χώρο-χρονικών δεδομένων, στους πίνακες δεδομένων και ανομοιότητας, στις συναρτήσεις σύγκρισης των τροχιών κινούμενων αντικειμένων καθώς και στις βασικές μεθόδους διατήρησης της ιδιωτικότητας. Επίσης στο κεφάλαιο αυτό γίνεται μία γενική περιγραφή του προβλήματος. Στο κεφάλαιο 4 περιγράφουμε αναλυτικά τις δύο μεθόδους ασφαλούς ανταλλαγής δεδομένων. Η παρουσίαση γίνεται με συγκεκριμένα παραδείγματα, για την ευκολότερη κατανόηση του προβλήματος καθώς επίσης παραθέτουμε και κάθε βήμα των μεθόδων σε μορφή ψευδό-κώδικα. Στο κεφάλαιο 5 παρουσιάζουμε την εργαλειοθήκη καθώς και τις βιβλιοθήκες που χρησιμοποιήσαμε για την υλοποίηση της εργαλειοθήκης. Τέλος στο κεφάλαιο 6 παραθέτουμε τα πειράματα που έχουμε κάνει και τα οποία αφορούν το υπολογιστικό κόστος των μεθόδων και των συναρτήσεων σύγκρισης όπως επίσης και την αξιολόγηση της συσταδοποίησης.

Κεφάλαιο 2

Σχετική βιβλιογραφία

Τα τελευταία χρόνια έχουν αναπτυχθεί πολλές και διαφορετικές τεχνικές για την διατήρηση της ιδιωτικότητας κατά την μοντελοποίηση και εξόρυξη δεδομένων. Κυρίως στην βιβλιογραφία έχουν προταθεί τεχνικές διατήρησης της ιδιωτικότητας με σκοπό είτε την κατηγοριοποίηση των δεδομένων, είτε εξαγωγής κανόνων συσχέτισης και προσφάτως με σκοπό την συσταδοποίηση των δεδομένων. Επιπλέον οι τεχνικές αυτές ποικίλουν ανάλογα με τον τρόπο που διαμοιράζονται τα δεδομένα στους κατόχους, όπως επίσης και με την μέθοδο που ακολουθείται για την τροποποίηση των δεδομένων. Όλες όμως οι τεχνικές που έχουν προταθεί έχουν σαν κύριο στόχο την όσον το δυνατόν μεγαλύτερη εγγύηση ασφάλειας των δεδομένων με το μικρότερο δυνατό υπολογιστικό κόστος.

Στην περίπτωση των κεντρικοποιημένων δεδομένων, δηλαδή όταν ο υπεύθυνος για την μοντελοποίηση έχει γνώση ολόκληρης της βάσης δεδομένων, οι Oliveira και Zaiane στο [1] πρότειναν τον μετασχηματισμό RBT (Rotation Based Transformation) για να επιτύχουν την διατήρηση της ιδιωτικότητας των δεδομένων κατά την συσταδοποίηση. Η μέθοδος τους βασίζεται στους ισομετρικούς μετασχηματισμούς. Το κύριο χαρακτηριστικό των μετασχηματισμών αυτών είναι ότι η απόσταση μεταξύ δύο αντικειμένων παραμένει η ίδια ακόμα και αν τα μετακινήσουμε σε n-διαστατό χώρο, δηλαδή αν T κάποιος μετασχηματισμός στον n-διάστατο χώρο T: $\Re^n \to \Re^n$, λέμε ότι είναι ισομετρικός αν διατηρούνται οι αποστάσεις σύμφωνα με τον ακόλουθο τύπο $|T(p)-T(q)|=|p-q|$ για κάθε $p,q \in \Re^n$. Κατά αυτόν τον τρόπο μετασχηματίζοντας ολόκληρη τη βάση δεδομένων κατάφεραν να επιτύχουν την διατήρηση της ιδιωτικότητας με έγκυρα αποτελέσματα συσταδοποίησης. Στο [2] οι ίδιοι συγγραφείς προτείνουν μία σειρά γεωμετρικών μετασχηματισμών με σκοπό την προστασία της ιδιωτικότητας κατά την συσταδοποίηση.

Παρά το γεγονός ότι αρκετοί αλγόριθμοι μοντελοποίησης είναι σχεδιασμένοι για τις περιπτώσεις κεντρικοποιημένων δεδομένων, στην πραγματικότητα τα δεδομένα διαμοιράζονται σε περισσότερους του ενός κατόχου δεδομένων. Στις περιπτώσεις αυτές η κεντρικοποίηση των δεδομένων πριν την ανάλυση, ίσως αποβεί ανέφικτη τόσο υπολογιστικά όσο και σε θέματα προστασίας της ιδιωτικότητας. Για αυτόν τον λόγο έχουν προταθεί στη βιβλιογραφία τεχνικές σε διαμοιραζόμενα δεδομένα. Για παράδειγμα στο [3] οι Oliveira και Zaiane πρότειναν έναν ενναλακτικό αλγόριθμο με τη χρησιμοποίηση δύο απλών και αποτελεσματικών μετασχηματισμών. Ο αλγόριθμος αυτός μπορεί να εφαρμοστεί και στην περίπτωση όπου τα δεδομένα διαμοιράζονται κάθετα στους κατόχους δεδομένων. Ο πρώτος μετασχηματισμός τον οποίο αναφέρουν ως OSBR (Object Similarity Based Representation), βασίζεται στην ανομοιότητα μεταξύ των αντικειμένων, έχει όμως μεγάλο υπολογιστικό κόστος, για αυτόν τον λόγο χρησιμοποιείται και ο δεύτερος μετασχηματισμός DRBT (Dimensionality Reduction Based Transformation), με τον οποίο μετατρέπουν ένα d-διάστατο πίνακα δεδομένων σε k- διάστατο όπου k<<d.

Στην μείωση των διαστάσεων βασίζεται και η τεχνική που προτείνεται στο [4], στην οποία όμως χρησιμοποιείται η ασαφής λογική. Τα δεδομένα τα οποία πρέπει να προστατευτούν μετατρέπονται σε ασαφή σύνολα και έτσι επιτυγχάνεται η διατήρηση της ιδιωτικότητας.

Στην βιβλιογραφία επίσης έχουν αναφερθεί αρκετές τεχνικές, οι οποίες τροποποιούν τον γνωστό αλγόριθμο συσταδοποίησης κ-μέσων, με σκοπό να λαμβάνεται υπόψη η προστασία των δεδομένων. Οι τεχνικές αυτές αφορούν περιπτώσεις οριζόντια διαχωριζόμενων δεδομένων [5], [6], κάθετα διαχωριζομένων δεδομένων [7], ακόμα και αυθαίρετα διαχωριζόμενων δεδομένων [8], όπου κάθε κάτοχος δεδομένων έχει γνώση είτε κάθετων, είτε οριζόντιων τμημάτων του πίνακα δεδομένων, όπως επίσης δύο κάτοχοι μπορεί να γνωρίζουν το κοινά τμήματα του πίνακα δεδομένων.
Μια παραλλαγή του αλγορίθμου κ-μέσων για κάθετα διαχωριζόμενα δεδομένα προτείνεται στο [9], όπου υλοποιείται ο αλγόριθμος κ-παραθύρων. Η μέθοδος εγγυάται την διατήρηση της ιδιωτικότητας των δεδομένων χωρίς να επηρεάζεται η αποτελεσματικότητα του αλγορίθμου της συσταδοποίησης.
Μια ενδιαφέρουσα τεχνική σε οριζόντια διαχωριζόμενα δεδομένα, με την οποία βελτιώνεται ο αλγόριθμος των κ-μέσων, προτείνεται στο [10]. Κάθε κάτοχος δεδομένων κατασκευάζει ένα γράφο με βάση τα δεδομένα που κατέχει και στη συνέχεια εφαρμόζει έναν αλγόριθμο ελαχιστοποίησης του γράφου αυτού. Στη συνέχεια υλοποιείται ένας αλγόριθμος ημί-συσταδοποίησης (bi-clustering) βάση των γράφων αυτών.
Η τεχνική αυτή υπερισχύει των παραδοσιακών αλγόριθμων συσταδοποίησης ως προς την εύρεση τοπικών προτύπων σε διαμοιραζόμενα δεδομένα.
Στην ίδια κατεύθυνση, δηλαδή σε οριζόντια διαχωριζόμενα δεδομένα τα οποία θα μας απασχολήσουν και στην παρούσα εργασία είναι ο αλγόριθμος που προτείνεται στο [11]. Σε αυτήν την προσέγγιση κάθε κάτοχος δεδομένων κατασκευάζει πιθανοτικά μοντέλα για τα δεδομένα του και στέλνει τις παραμέτρους των μοντέλων αυτών σε μια κεντρική τοποθεσία η οποία ενοποιεί τα μοντέλα αυτά. Με αυτόν τον τρόπο ελαχιστοποιούνται οι αλληλεπιδράσεις μεταξύ των κατόχων και της κεντρικής τοποθεσίας και απλοποιείται το πρόβλημα της συσταδοποίησης σε διαμοιραζόμενα δεδομένα. Οι συγγραφείς πρότειναν την δημιουργία τεχνητών δειγμάτων από τα τοπικά μοντέλα του κάθε κατόχου χρησιμοποιώντας τεχνικές δειγματοληψίας, προσαρμόζοντας τα στη συνέχεια σε ένα καθολικό μοντέλο. Στη συνέχεια χρησιμοποίησαν την ιδέα αυτή για την κατασκευή EM (Estimation-Maximization) αλγορίθμων για το πρόβλημα της συσταδοποίησης, οι οποίοι συγκλίνουν ασυμπτωτικά στο καθολικό μοντέλο.
Στο [12] οι Ali Inan, Yucel Saygin, Erkay Savas κ.α. πρότειναν ένα πρωτόκολλο ασφαλούς ανταλλαγής δεδομένων σε οριζόντια διαχωριζόμενα δεδομένα με σκοπό την κατασκευή ενός πίνακα ανομοιότητας, ο οποίος περιέχει τις αποστάσεις των σημείων και είναι απαραίτητος για την συσταδοποίηση. Σε αυτήν την προσέγγιση, κάθε κάτοχος δεδομένων τροποποιεί τα δεδομένα του με τη χρήση τυχαίων αριθμών, οι οποίοι μοιράζονται μεταξύ των κατόχων και ενός τρίτου συμμετέχοντα. Ο τρίτος συμμετέχοντας είναι υπεύθυνος για την επικοινωνία μεταξύ των κατόχων, συμμετέχει στους υπολογισμούς και ενημερώνει τους κατόχους για τα αποτελέσματα της συσταδοποίησης. Η μέθοδος που χρησιμοποιείται για την προστασία των δεδομένων είναι μια παραλλαγή της τεχνικής του ασφαλούς αθροίσματος [23] και το πρωτόκολλο που προτείνεται αφορά τρεις διαφορετικούς τύπους δεδομένων αριθμητικά, κατηγορικά και αλφαριθμητικά δεδομένα.
Η έρευνα για την προστασία της ιδιωτικότητας σε χώρο-χρονικά δεδομένα βρίσκεται ακόμα σε πρώιμο στάδιο. Στο [24] οι Ali Inan, Yucel Saygin εφάρμοσαν το πρωτόκολλο του [12] για χώρο-χρονικά δεδομένα. Η μέθοδος αυτή είναι η πρώτη μέθοδος που υλοποιούμε στην παρούσα εργασία. Επίσης στο [13] προτάθηκε μια διαφορετική τεχνική για την διατήρηση της ιδιωτικότητας κατά την συσταδοποίηση τροχιών κινούμενων αντικειμένων. Οι Ali Ulvi Kasapoğlu, Mahir Can Doğanay

κατασκεύασαν ένα παρόμοιο πρωτόκολλο τριών συμμετεχόντων με το [12], με την μόνη διαφορά στη μέθοδο που χρησιμοποιήθηκε για την προστασία των δεδομένων. Οι συγγραφείς πρότειναν την «διατάραξη» των δεδομένων, πολλαπλασιάζοντας τα με τυχαίες τιμές θορύβου. Παρά το γεγονός ότι με τη μέθοδο αυτή διατηρούνται οι αποστάσεις μεταξύ των σημείων, η κατασκευή του πίνακα ανομοιότητας είναι υπολογιστικά αρκετά δαπανηρή. Έτσι προτάθηκε να συρρικνωθεί η αρχική βάση δεδομένων σε ένα υποσύνολο το οποίο διατηρεί τα στατιστικά του αρχικού συνόλου δηλαδή τις αποστάσεις μεταξύ των στοιχείων.

Κεφάλαιο 3

Βασικοί Ορισμοί

3. 1 Βασικές έννοιες

Στο κεφάλαιο αυτό γίνεται μια περιγραφή των βασικών εννοιών που είναι απαραίτητες για την ανάλυση που πρόκειται να ακολουθήσει και οι οποίες χρησιμοποιούνται στα επόμενα κεφάλαια. Πιο συγκεκριμένα γίνεται μια εισαγωγή στα χώρο-χρονικά δεδομένα και τις εφαρμογές τους, στους πίνακες δεδομένων και στους πίνακες ανομοιότητας, οι οποίοι αποτελούν ίσως το πιο κρίσιμο τμήμα των αλγορίθμων που θα αναλύσουμε στα επόμενα κεφάλαια, καθώς και στις συναρτήσεις σύγκρισης τροχιών κινούμενων αντικειμένων οι οποίες είναι ιδιαίτερα σημαντικές για την σωστή κατασκευή του πίνακα ανομοιότητας. Επίσης στο τέλος του κεφαλαίου αυτού γίνεται μια διατύπωση του προβλήματος με βάση τις περιγραφές που έχουν γίνει.

3.1. 1 Χώρο-χρονικά δεδομένα

Σε αρκετές εφαρμογές όπως σε Συστήματα Γεωγραφικών Πληροφοριών (GIS), ρομποτικής, βίο-πληροφορικής, φορητών υπολογιστών και ανάλυσης κίνησης (traffic analysis), τεράστιες ποσότητες πληροφορίας δημιουργούνται και αποθηκεύονται στις βάσεις δεδομένων στην μορφή χώρο-χρονικών δεδομένων. Τα χώρο-χρονικά δεδομένα χαρακτηρίζονται από μία χρονική διάσταση και από τουλάχιστον μια χωρική διάσταση.
Τα χώρο-χρονικά δεδομένα κρύβουν χρήσιμες πληροφορίες και επομένως η εξαγωγή γνώσης από τα δεδομένα αυτά είναι ιδιαίτερα σημαντική. Για παράδειγμα στο
[14] υπάρχουν κατηγορίες χώρο-χρονικών δεδομένων που χρησιμοποιούνται για να περιγράψουν τις συμπεριφορές και συνήθειές ζώων, όπως καρχαρίες με παραμέτρους τη θέση στην οποία βρίσκονται κάθε συγκεκριμένη χρονική στιγμή και το πόσο βαθιά κινούνται, φυσικών φαινόμενων όπως τυφώνων με παραμέτρους τη θέση, την ταχύτητα ανέμου τη χρονική στιγμή της μέτρησης καθώς και διαφόρων κινούμενων αντικειμένων όπως τρένων, αυτοκινήτων.
Τα δεδομένα που χρησιμοποιούνται στα επόμενα κεφάλαια, ανήκουν στην κατηγορία των χώρο-χρονικών δεδομένων. Οι βάσεις χώρο-χρονικών δεδομένων προκύπτουν από παρατηρήσεις της θέσης κινούμενων αντικειμένων σε συγκεκριμένα χρονικά διαστήματα, είναι δηλαδή της μορφής (id,t,s), όπου id η μοναδική ταυτότητα του αντικειμένου, t ο χρόνος και s η θέση του αντικειμένου. Η θέση του κάθε αντικειμένου μπορεί να καθοριστεί από μία ή περισσότερες διαστάσεις. Για παράδειγμα όταν τα δεδομένα αφορούν κινούμενα αντικείμενα όπως αυτοκίνητα, η θέση τους μπορεί να περιγραφεί από δύο διαστάσεις, το γεωγραφικό μήκος και πλάτος, ενώ όταν τα χώρο-χρονικά δεδομένα περιγράφουν για παράδειγμα, κινήσεις δορυφόρων η θέση τους καθορίζεται από τρεις διαστάσεις, το γεωγραφικό μήκος και πλάτος καθώς και το ύψος στο οποίο κινούνται τη συγκεκριμένη χρονική στιγμή.

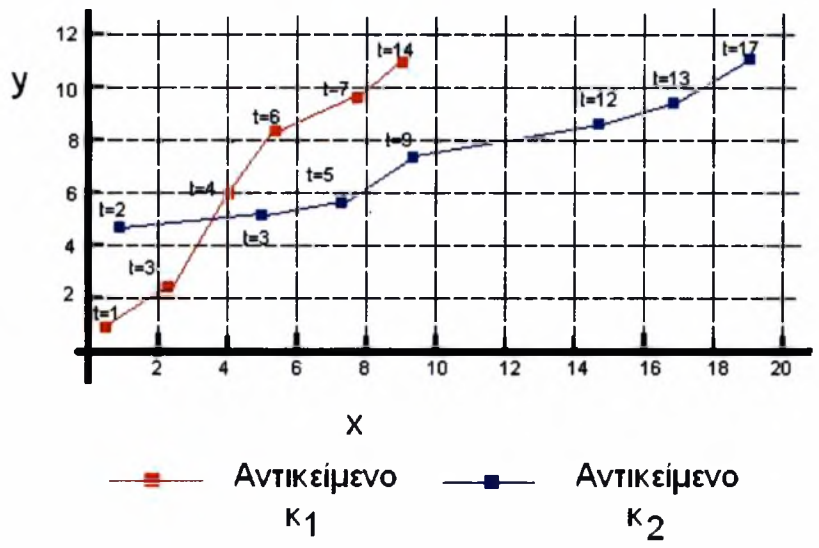
Η βάση χώρο-χρονικών δεδομένων που θα χρησιμοποιήσουμε στα επόμενα κεφάλαια 4,5 για την περιγραφή και υλοποίηση των αλγορίθμων, θεωρούμε ότι αποτελείται από χωρικά δεδομένα δύο διαστάσεων. Δηλαδή η θέση των κινούμενων αντικειμένων θα περιγράφεται από δύο διαστάσεις x, y. Η παραδοχή αυτή δεν είναι δεσμευτική καθώς οι αλγόριθμοι που θα παρουσιαστούν μπορούν να υποστηρίξουν χώρο-χρονικά δεδομένα περισσοτέρων διαστάσεων.
Στον πίνακα 1 φαίνονται τα χώρο-χρονικά δεδομένα για δύο κινούμενα αντικείμενα κ_1,κ_2

Αντικείμενο κ_1	Χρόνος (t)						
	1	3	4	6	7	14	
Μήκος(x)	0.5	2.2	4	5.3	7.7	9	
Πλάτος(y)	1	2	6	8.3	9.9	11	
Αντικείμενο κ_2	Χρόνος (t)						
	2	3	5	9	12	13	17
Μήκος(x)	1	5	7.5	9.5	14.6	17	19
Πλάτος(y)	4.5	5	5.8	7.5	8.5	9.5	11

Πίνακας 1 Χώρο-χρονικά δεδομένα για τα κινούμενα αντικείμενα κ_1,κ_2

3.1. 2 Τροχιές κινούμενων αντικειμένων

Η τροχιά ενός κινούμενου αντικειμένου αποτελείται από ένα σύνολο παρατηρήσεων της θέσης του αντικειμένου ανά χρονική στιγμή. Δηλαδή για το χρονικό διάστημα $T = \{t_1, t_2, ..., t_n\}$, το σύνολο των παρατηρήσεων και επομένως η τροχιά ενός κινούμενου αντικειμένου κ_1 είναι $Traj(\kappa_1) = \{(t_1, x_1, y_1), (t_2, x_2, y_2), ..., (t_n, x_n, y_n)\ \}$
Το μήκος της τροχιάς ενός αντικειμένου είναι ο αριθμός των παρατηρήσεων.
Παρατηρούμε στον Πίνακα 1 ότι οι παρατηρήσεις για το αντικείμενο κ_1 είναι 6 άρα το μήκος της τροχιάς του είναι 6 ($Traj(\kappa_1).length = 6$), ενώ αντίστοιχα για το αντικείμενο κ_2 το μήκος της τροχιάς του είναι 7 ($Traj(\kappa_2).length = 7$).
Στην εικόνα 1 φαίνονται οι τροχιές των δύο κινούμενων αντικειμένων κ_1,κ_2, με βάση τα χώρο-χρονικά δεδομένα του Πίνακα 1



Εικόνα 1 Τροχιές αντικειμένων κ_1,κ_2 με Traj(κ_1).length = 6 και Traj(κ_2).length = 7

3. 2 Βασικοί Πίνακες

3.2. 1 Πίνακας δεδομένων

Ο πίνακας δεδομένων περιέχει ένα σύνολο κινούμενων αντικειμένων $K = (\kappa_1, \kappa_2, \ldots, \kappa_\nu)$ και αντιστοιχίζει τα αντικείμενα με τις παρατηρήσεις τους. Πιο συγκεκριμένα, κάθε γραμμή του πίνακα δεδομένων περιέχει ένα κινούμενο αντικείμενο και κάθε στήλη του πίνακα μια παρατήρηση του αντικειμένου για μια συγκεκριμένη χρονική στιγμή. Δηλαδή όταν γράφουμε για παράδειγμα κ_i^j θα αναφερόμαστε στο i-στό αντικείμενο από το σύνολο των κινούμενων αντικειμένων και στη j-στή παρατήρηση από το σύνολο των παρατηρήσεων του, δηλαδή της τροχιάς του και άρα στο κελί του πίνακα δεδομένων (i, j). Επίσης για να αναφερθούμε στις παραμέτρους μιας παρατήρησης j, $(t, x, y)^j$, δηλαδή στον χρόνο t, στο μήκος x, στο πλάτος y, για ένα συγκεκριμένο κινούμενο αντικείμενο i θα γράφουμε

$\kappa_i^j.t$, $\kappa_i^j.x$, $\kappa_i^j.y$ αντίστοιχα.

Ένα παράδειγμα ενός πίνακα με ν κινούμενα αντικείμενα και μ παρατηρήσεις, όπου i=1, 2, 3....ν , j=1, 2, 3....μ φαίνεται στον πίνακα 2

$$\mathrm{K}=\begin{pmatrix} \kappa_1^1 & \cdots & \kappa_1^j & \cdots & \kappa_1^\mu \\ \vdots & & \vdots & & \vdots \\ \kappa_i^1 & \cdots & \kappa_i^j & \cdots & \kappa_i^\mu \\ \vdots & & \vdots & & \vdots \\ \kappa_\nu^1 & \cdots & \kappa_\nu^j & \cdots & \kappa_\nu^\mu \end{pmatrix}$$

Πίνακας 2 Πίνακας δεδομένων

3.2. 2 Πίνακας Ανομοιότητας

Σε αντίθεση με τον πίνακα δεδομένων, ο πίνακας ανομοιότητας περιέχει σε κάθε γραμμή και κάθε στήλη κινούμενα αντικείμενα. Πιο συγκεκριμένα κάθε κελί του πίνακα περιέχει την απόσταση ή ανομοιότητα μεταξύ δύο κινούμενων αντικειμένων. Για παράδειγμα για το σύνολο $\mathrm{K}=(\kappa_1,\kappa_2,\ldots,\kappa_\nu)$ το κελί του πίνακα $(2,1)$, περιέχει την απόσταση των κινούμενων αντικειμένων κ_1,κ_2 , $dist(\kappa_1,\kappa_2)$.

Η απόσταση δύο κινούμενων αντικειμένων, δηλαδή ο υπολογισμός της απόστασης των τροχιών των κινούμενων αντικειμένων είναι μια περίπλοκη διαδικασία και δεν μπορεί να υπολογιστεί με τις παραδοσιακές τεχνικές σύγκρισης. Στην επόμενη ενότητα 3.4 παρέχουμε τρεις διαφορετικούς αλγόριθμους σύγκρισης, οι οποίοι χρησιμοποιήθηκαν για τον υπολογισμό της απόστασης και μετέπειτα για την κατασκευή του πίνακα ανομοιότητας.

Προφανώς ο πίνακας ανομοιότητας είναι συμμετρικός αφού για την απόσταση δύο αντικειμένων ισχύει $dist(\kappa_1,\kappa_2)=dist(\kappa_2,\kappa_1)$ όπως επίσης η διαγώνιος του πίνακα είναι 0 διότι η απόσταση ενός αντικειμένου από τον εαυτό του είναι 0.

Σύμφωνα με τα παραπάνω ο πίνακας ανομοιότητας είναι της μορφής που φαίνεται στον Πίνακα 3.

$$\Pi_{\alpha\nu}=\begin{pmatrix} 0 & & & & \\ dist(\kappa_2,\kappa_1) & 0 & & & \\ dist(\kappa_3,\kappa_1) & dist(\kappa_3,\kappa_2) & 0 & & \\ \vdots & \vdots & \vdots & 0 & \\ dist(\kappa_\nu,\kappa_1) & dist(\kappa_\nu,\kappa_2) & dist(\kappa_\nu,\kappa_3) & \cdots & 0 \end{pmatrix}$$

Πίνακας 3 Πίνακας Ανομοιότητας

Ο πίνακας ανομοιότητας είναι ιδιαίτερα σημαντικός για τον λόγο ότι οι περισσότεροι αλγόριθμοι συσταδοποίησης χρησιμοποιούν τον πίνακα ανομοιότητας σαν είσοδο για να εξάγουν τα αποτελέσματα της συσταδοποίησης, ιεραρχικοί, κ-μέσων και αλγόριθμοι βασισμένοι στην απόσταση των κινούμενων αντικειμένων είναι μερικοί εξ' αυτών. Επίσης και σε άλλες τεχνικές της μοντελοποίησης δεδομένων όπως κατηγοριοποίηση ή σε αλγορίθμους εύρεσης του πλησιέστερου γείτονα, είναι απαραίτητος ο πίνακας ανομοιότητας αφού και σε αυτές τις περιπτώσεις κυρίαρχο ρόλο έχει η απόσταση των αντικειμένων.

3. 3 Διαμέριση του πίνακα δεδομένων

Όπως αναφέραμε και στην εισαγωγή, οι μέθοδοι και οι αλγόριθμοι που μελετάμε στην εργασία αυτή, αφορούν υπολογισμούς και ανταλλαγή πληροφοριών μεταξύ πολλών κατόχων δεδομένων. Ο διαμερισμός των κινούμενων αντικειμένων του πίνακα δεδομένων μπορεί να γίνει με διάφορους τρόπους. Ο κάθε κάτοχος δεδομένων μπορεί να γνωρίζει ένα μέρος μόνο του πίνακα δεδομένων ή και ακόμα να έχει γνώση ολόκληρου του πίνακα δεδομένων.

Στην πρώτη περίπτωση τα δεδομένα του πίνακα δεδομένων διαμοιράζονται μεταξύ των κατόχων δεδομένων (distributed data), ενώ στην δεύτερη περίπτωση οι κάτοχοι δεδομένων έχουν γνώση όλου του πίνακα δεδομένων (centralized data).

Παρακάτω παραθέτουμε αναλυτικά τις κατηγορίες διαμέρισης των δεδομένων και αντιπροσωπευτικά παραδείγματα, τα οποία αναδεικνύουν και το πρόβλημα της ασφάλειας των δεδομένων.

3.3. Κεντρικοποιημένα δεδομένα

Έστω ένας κάτοχος δεδομένων, για παράδειγμα μια μικρή εταιρία, η οποία ενώ έχει στη διάθεση της δεδομένα και έχει καταλάβει τη σημασία της ανάλυσης τους για την περαιτέρω ανάπτυξή της, δεν είναι σε θέση να μοντελοποιήσει τα δεδομένα της λόγω των περιορισμένων δυνατοτήτων της. Επομένως έχει δύο επιλογές είτε να μοντελοποιήσει ένα μόνο μέρος των δεδομένων της με συνέπεια την ανακρίβεια στα αποτελέσματα, είτε να ζητήσει την βοήθεια από άλλη εταιρία με συνέπεια όμως την ασφάλεια των δεδομένων της. Στην δεύτερη περίπτωση η εταιρία που θα αναλάβει να μοντελοποιήσει τα δεδομένα της αρχικής εταιρίας θα πρέπει να γνωρίζει ολόκληρη τη βάση δεδομένων .

3.3. 2 Διαμοιραζόμενα δεδομένα

Έστω τώρα δύο εταιρίες οι οποίες στη βάση δεδομένων τους έχουν δεδομένα για ένα κοινό σύνολο αντικειμένων αλλά για διαφορετικές ιδιότητες, για παράδειγμα για ένα σύνολο ατόμων η πρώτη εταιρία να γνωρίζει το εισόδημα τους, ενώ η δεύτερη εταιρία το επάγγελμά τους και την αγοραστική τους επιθυμία. Οι εταιρίες αυτές επιθυμούν να συνεργαστούν, να ανταλλάξουν δηλαδή τα δεδομένα τους, με σκοπό να αυξήσουν τα κέρδη τους.

Στη περίπτωση που τα δεδομένα διαμοιράζονται, κάθε κάτοχος μπορεί να κατέχει είτε οριζόντια είτε κάθετα τμήματα του πίνακα δεδομένων, η ακόμα και αυθαίρετα τμήματα

3.3.2. 1 Οριζόντια διαχωριζόμενα δεδομένα

Σε αυτήν την περίπτωση κάθε κάτοχος δεδομένων κατέχει οριζόντιες γραμμές του πίνακα δεδομένων. Δηλαδή για ένα σύνολο n κατόχων δεδομένων $\{DH_1, DH_2, ..., DH_n\}$ και για το σύνολο ν κινούμενων αντικειμένων $K = (\kappa_1, \kappa_2, \ldots, \kappa_\nu)$ με $Traj(\kappa_i).length = l_i$ $i, j = 1,2,3.....\nu$, $i \neq j$ και h=1,2,3.....n , και κάθε κάτοχος DH_h έχει το υποσύνολο

$$K^{DH_h} = \left\{ \sum_{m=0}^{l_i} \kappa_i^m \cup ..... \cup \sum_{m=0}^{l_j} \kappa_j^m \right\}.$$

Στην εικόνα 2 φαίνεται η διαμέριση του πίνακα δεδομένων σε οριζόντια τμήματα.

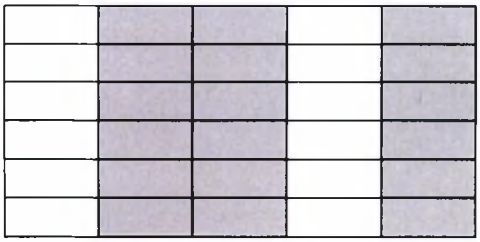
Εικόνα 2 Οριζόντια διαχωριζόμενος πίνακας δεδομένων

3.3.2. 2 Κάθετα διαχωριζόμενα δεδομένα

Στην περίπτωση αυτή κάθε κάτοχος δεδομένων κατέχει στήλες του πίνακα δεδομένων . Ομοίως για ένα σύνολο n κατόχων δεδομένων $\{DH_1, DH_2, ..., DH_n\}$ και για το σύνολο ν κινούμενων αντικειμένων $K = (\kappa_1, \kappa_2, ..., \kappa_\nu)$ με $Traj(\kappa_i).length = l_i$, $i, j = 1,2,3.....\nu$, $i \neq j$ και h=1,2,3.....,n , κάθε κάτοχος δεδομένων DH_h έχει το υποσύνολο

$$K^{DH_h} = \left\{ \sum_{i=1}^{\nu} \kappa_i^{m_1} \cup \kappa_i^{m_2} ..... \cup \kappa_i^{m_l} \right\}$$, όπου $m_1, m_2 \ldots m_l$ παρατηρήσεις των κινούμενων αντικειμένων.

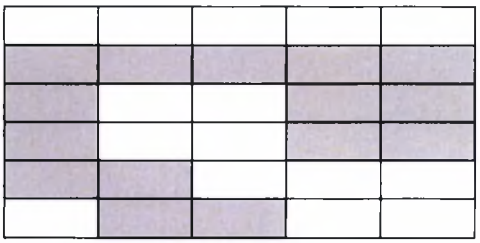
Στην εικόνα 3 φαίνεται η διαμέριση του πίνακα δεδομένων σε κάθετα τμήματα.



Εικόνα 3 Κάθετα διαχωριζόμενος πίνακας δεδομένων

3.3.2. 3 Αυθαίρετα διαχωριζόμενα δεδομένα

Είναι ένας συνδυασμός των δύο παραπάνω τεχνικών όπου κάθε κάτοχος δεδομένων μπορεί να γνωρίζει οποιαδήποτε γραμμή ή στήλη του πίνακα δεδομένων και σαφώς σε αυτή την αυθαίρετη περίπτωση διαμοιρασμού των δεδομένων δύο κάτοχοι μπορούν να κατέχουν κοινά τμήματα του πίνακα δεδομένων.
Στην εικόνα 4 φαίνεται η διαμέριση του πίνακα δεδομένων σε αυθαίρετα τμήματα.



Εικόνα 4 Αυθαίρετα διαχωριζόμενος πίνακας δεδομένων

3. 4 Συναρτήσεις Σύγκρισης

Ο σωστός και ακριβής υπολογισμός της απόστασης των κινούμενων αντικειμένων αποτελεί ένα από τα σημαντικότερα ζητήματα της εργασίας. Όπως αναφέραμε και προηγουμένως, για την κατασκευή του πίνακα ανομοιότητας και επομένως για να συσταδοποιήσουμε τις τροχιές των αντικειμένων, απαιτούνται συναρτήσεις που να δίνουν αξιόπιστα αποτελέσματα σε σχέση με την απόσταση ή την ανομοιότητα των κινούμενων αντικειμένων. Στην ιδανική περίπτωση όπου οι τροχιές των αντικειμένων έχουν τον ίδιο αριθμό παρατηρήσεων για τα ίδια χρονικά διαστήματα η μέτρηση της απόστασης είναι απλή. Η ευκλείδεια απόσταση σε αυτήν την περίπτωση μπορεί να δώσει ξεκάθαρα αποτελέσματα. Στην πραγματικότητα όμως το μήκος των τροχιών των κινούμενων αντικειμένων διαφέρει από αντικείμενο σε αντικείμενο, όπως επίσης και η μέτρηση της θέσης των αντικειμένων γίνεται για διαφορετικά χρονικά διαστήματα.
Για την εύρεση της ομοιότητας σε χρονικές σειρές έχουν προταθεί αρκετές τεχνικές.
Οι Agrawal, Faloutsos στο [15] χρησιμοποίησαν την ευκλείδεια απόσταση σαν το βασικό μέτρο ομοιότητας και πρότειναν την χρησιμοποίηση του διακριτού μετασχηματισμού Fourier (DTF) για την ένωση (matching) χρονικών σειρών. Μια εναλλακτική μέθοδο στην κατεύθυνση αυτή, δηλαδή ένωση χρονικών σειρών, μέσω

μείωσης των διαστάσεων πρότειναν οι Chan και Fu στο [16], χρησιμοποιώντας το διακριτό μετασχηματισμό Wavelet (DTW).
Στο [17] οι Clifford και Berndt χρησιμοποίησαν την τεχνική δυναμικής χρονικής παραμόρφωσης (DTW), η οποία επιτρέπει την επέκταση μιας ακολουθίας στον άξονα του χρόνου ούτως ώστε να ελαχιστοποιηθεί η απόσταση μεταξύ των ακολουθιών και να μπορέσουμε να συγκρίνουμε ακολουθίες διαφορετικού μήκους.
Μια βελτιωμένη έκδοση της παραπάνω τεχνικής η οποία μειώνει το υπολογιστικό κόστος προτείνεται στο [18].
Επίσης πολλές τεχνικές έχουν σαν μέτρο σύγκρισης την Longest Common Sub Sequence (LCSS). Σε αντίθεση με την ευκλείδεια απόσταση και την απόσταση DTW, η LCSS ενώνει δύο ακολουθίες επεκτείνοντας τες χωρίς να αλλάξουν θέση τα στοιχεία και επιτρέποντας ορισμένα στοιχεία να μην ενωθούν.
Στο [19] χρησιμοποιήθηκε η μέθοδος LCSS. Δημιουργήθηκαν συναρτήσεις ομοιότητας βασισμένες στη LCSS, οι οποίες επιτρέπουν επέκταση και παράλληλη μετατόπιση και είναι ανθεκτικές στην παρουσία θορύβου και επιπλέον παρέχουν την ομοιότητα των τροχιών κινούμενων αντικειμένων, δίνοντας περισσότερο βάρος στα όμοια τμήματα των τροχιών.
Στο [20] χρησιμοποιήθηκε η συνάρτηση απόστασης edit distance on real Sequences (EDR). Βασισμένη στην edit distance η συνάρτηση EDR αποδείχθηκε ότι έχει καλύτερα αποτελέσματα σε τροχιές με θόρυβο από τις συναρτήσεις DTW και LCSS. Παρόλα αυτά η EDR υπολογίζει μόνο την χωρική ομοιότητα αγνοώντας τον χρόνο, επομένως η αποτελεσματικότητα και αυτής της μεθόδου μειώνεται όταν οι τροχιές έχουν διαφορετικό ρυθμό δειγματοληψίας.
Στα επόμενα κεφάλαια για τον υπολογισμό της απόστασης των κινούμενων αντικειμένων και για την κατασκευή του πίνακα ανομοιότητας χρησιμοποιήθηκαν οι εξής αλγόριθμοι σύγκρισης :

- Ευκλείδεια Απόσταση
- Αλγόριθμος δυναμικής χρονικής παραμόρφωσης
- Επαναληπτικός αλγόριθμος για τον υπολογισμό της ομοιότητας μεταξύ τροχιών

και οι οποίοι παρουσιάζονται αναλυτικά στις παρακάτω ενότητες.

3.4. 1 Ευκλείδεια Απόσταση

Όπως προαναφέραμε η ευκλείδεια απόσταση έχει νόημα μόνο όταν οι τροχιές των κινούμενων αντικειμένων έχουν το ίδιο μήκος, δηλαδή τον ίδιο αριθμό παρατηρήσεων. Σε οποιαδήποτε άλλη περίπτωση η ευκλείδεια απόσταση δεν μπορεί να μας δώσει αξιόπιστα αποτελέσματα. Για να μπορέσουμε να εφαρμόσουμε την ευκλείδεια απόσταση στους αλγορίθμους του κεφαλαίου 4, για να βγάλουμε χρήσιμα συμπεράσματα, υπάρχουν δύο περιπτώσεις, είτε να συμπληρώσουμε με μηδενικά τις τροχιές των αντικειμένων με μικρότερο μήκος, είτε να επιβάλουμε ένα ελάχιστο μήκος τροχιάς.
Παρακάτω φαίνεται ο ψευδό-κώδικας όπου ο αλγόριθμος δέχεται σαν είσοδο δύο αντικείμενα κ_1,κ_2 .Το μήκος των τροχιών των δύο αντικειμένων είναι διαφορετικό, $Traj(\kappa_1).length \neq Traj(\kappa_2).length$, επομένως επιλέγουμε να επιβάλλουμε ένα ελάχιστο μήκος τροχιάς

Begin
if (Traj(κ_1).length ≤ Traj(κ_2).length)
do
for (i=0 **to** Traj(κ_1).length)

$$dist(\kappa_1,\kappa_2)=\sqrt{\sum_{i=1}^{Traj(\kappa_1).length}\left|\kappa_1^i-\kappa_2^i\right|}$$

end for
else
for (i=0 **to** Traj(κ_2).length)

$$dist(\kappa_1,\kappa_2)=\sqrt{\sum_{i=1}^{Traj(\kappa_2).length}\left|\kappa_1^i-\kappa_2^i\right|}$$

end for
End

Ψευδό-κώδικας 1 Ευκλείδεια Απόσταση

3.4. 2 Δυναμική χρονική παραμόρφωση (Dynamic Time Warping)

Ο αλγόριθμος προτάθηκε στο [21] και χρησιμοποιεί μία συνάρτηση ανομοιότητας χρονικής παραμόρφωσης «time warping», η οποία επιτρέπει τοπικές επιταχύνσεις και επιβραδύνσεις στο ρυθμό του σήματος ή της ακολουθίας.
Ο αλγόριθμος αυτός είναι κατάλληλος για αρκετές εφαρμογές ελέγχου απόστασης σημάτων όπως φωνής, ήχου άλλα και ιατρικών σημάτων (καρδιογραφήματα). Είναι παρόμοιος με τον αλγόριθμο edit distance μόνο που αντί για εκχωρήσεις και διαγραφές χρησιμοποιούνται τα stutters.

3.4.2. 1 Μετασχηματισμός Χρονικής Παραμόρφωσης

Σε αυτή την παράγραφο παραθέτουμε βασικούς ορισμούς του μετασχηματισμού χρονικής παραμόρφωσης. Έστω η τροχιά του κινούμενου αντικειμένου κ_1 με μήκος n
$Traj(\kappa_1) = \{(t_1, x_1, y_1), (t_2, x_2, y_2), ..., (t_n, x_n, y_n)\}$
Ισχύουν τα ακόλουθα

- $head(Traj(\kappa_1)) = (t_1, x_1, y_1)$

- $rest(Traj(\kappa_1)) = \{(t_2, x_2, y_2), ..., (t_n, x_n, y_n)\}$

- $stutter_i(Traj(\kappa_1))$ επαναλαμβάνει το $(t, x, y)^i$ και ολισθαίνει τα στοιχεία δεξιά.

Με βάση τους παραπάνω ορισμούς, η απόσταση δυναμικής χρονικής παραμόρφωσης ορίζεται ως εξής για δύο κινούμενα αντικείμενα κ_1, κ_2 με τροχιές $Traj(\kappa_1)$, $Traj(\kappa_2)$ αντίστοιχα ορίζεται ως εξής

$$D_{warp}(Traj(\kappa_1), <>) = D_{warp}(<>, Traj(\kappa_2)) = \infty$$
$$D_{warp}(Traj(\kappa_1), Traj(\kappa_2)) = D_{base}(head(Traj(\kappa_1)), head(Traj(\kappa_2))) +$$

$$+ \min \begin{Bmatrix} D_{warp}(Traj(\kappa_1), rest(Traj(\kappa_2))) \\ D_{warp}(rest(Traj(\kappa_1)), Traj(\kappa_2)) \\ D_{warp}(rest(Traj(\kappa_1)), rest(Traj(\kappa_2))) \end{Bmatrix}$$

Όπου <> ,η μηδενική τροχιά και D_{base} οποιοδήποτε μέτρο απόστασης που ακολουθεί τον τύπο της L_P νόρμας.

Το σημαντικό σε αυτόν τον ορισμό και επομένως στον αλγόριθμο που χρησιμοποιήσαμε είναι ότι οι δύο ακολουθίες, δηλαδή οι τροχιές δύο αντικειμένων δεν είναι απαραίτητο να είναι του ίδιου μήκους.
Παρακάτω παραθέτουμε τον ψευδό-κώδικα για τον υπολογισμό της απόστασης δύο αντικειμένων με βάση τον αλγόριθμο δυναμικής χρονικής παραμόρφωσης. Ο αλγόριθμος δέχεται ως είσοδο δυο κινούμενα αντικείμενα και έχει ως έξοδο έναν πίνακα M. Ο πίνακας M[i,j] περιέχει το κόστος της ένωσης των πρώτων i παρατηρήσεων του πρώτου αντικειμένου με τις j του δεύτερου αντικειμένου. Έτσι ο πίνακας $M\left[Traj(\kappa_1).length\right]\left[Traj(\kappa_2).length\right]$ περιέχει το επιθυμητό κόστος δηλαδή την απόσταση των δύο κινούμενων αντικειμένων.

Begin

// αρχικοποιήσεις

$m = Traj(\kappa_1.length)$

$n = Traj(\kappa_2.length)$

M[0,0] = 0

M[1.....m, 0] = ∞

M[0,1.....n] = ∞

//

for (i=1 **to** m)

  for (j=1 **to** n)

$$M[i][j] = D_{base}(\kappa_1^i, \kappa_2^j) + \min \begin{Bmatrix} M[i-1, j] \\ M[i, j-1] \\ M[i-1, j-1] \end{Bmatrix}$$

  end for

end for

return M[m][n]

End

Ψευδό-κώδικας 2 Αλγόριθμος Δυναμικής Χρονικής Παραμόρφωσης(DTW)

3.4. 3 Επαναληπτικός αλγόριθμος για τον υπολογισμό της ομοιότητας μεταξύ δύο τροχιών

Οι Laurinen, Siirtola και Roning στο [22] πρότειναν έναν αποτελεσματικό επαναληπτικό αλγόριθμο για τον υπολογισμό της ομοιότητας των τροχιών κινούμενων αντικειμένων, όταν αυτές περιέχουν μία αύξουσα διάσταση όπως για παράδειγμα ο χρόνος.
Ο επαναληπτικός αλγόριθμος ξεκινάει από την πρώτη παρατήρηση της τροχιάς του πρώτου αντικειμένου, διαπερνάει όλες τις παρατηρήσεις της τροχιάς του δεύτερου αντικειμένου και προσθέτει την απόσταση των δυο κοντινότερων παρατηρήσεων στην συνολική απόσταση των δυο αντικειμένων. Στο επόμενο βήμα ο αλγόριθμος ξεκινάει από την δεύτερη παρατήρηση της τροχιάς του πρώτου αντικειμένου και ου το καθεξής μέχρι να διαπεράσει όλες τις παρατηρήσεις της τροχιάς του πρώτου αντικειμένου.
Τελικά η απόσταση διαιρείται με το αριθμό των παρατηρήσεων της τροχιάς του πρώτου αντικειμένου, ούτως ώστε να μην έχει καμία επίδραση στο αποτέλεσμα το διαφορετικό μήκος των τροχιών.
Ο ψευδό-κώδικας για τον συγκεκριμένο αλγόριθμο φαίνεται παρακάτω όπου είσοδος είναι τα δύο κινούμενα αντικείμενα και έξοδος η απόσταση τους.

Begin
// αρχικοποιήσεις
$m = Traj(\kappa_1.length)$
$n = Traj(\kappa_2.length)$
trajectory_distance=0
smallest_distance=∞
//
for (i=1 **to** m)
  for (j=1 **to** n)
    if (dist(κ_1^i, κ_2^j) < smallest_distance)
      1. smallest_distance= dist(κ_1^i, κ_2^j)
      2. trajectory_distance += smallest_distance
      3. smallest_distance=∞
  end for
end for
4. trajectory_distance = trajectory_distance / Traj(κ_1).length
return trajectory_distance
End

Ψευδό-κώδικας 3 Επαναληπτικός Αλγόριθμος

3. 5 Τεχνικές διατήρησης της ιδιωτικότητας

Στην βιβλιογραφία έχουν προταθεί αρκετές τεχνικές για την διατήρηση της ιδιωτικότητας των δεδομένων, στις περιπτώσεις που τα δεδομένα διαμοιράζονται σε κατόχους δεδομένων. Οι τεχνικές που θα μας απασχολήσουν στην παρούσα εργασία και τις υλοποιούμε στις μεθόδους του κεφαλαίου 4 είναι δύο, η τεχνική του ασφαλούς αθροίσματος και η τεχνική του ασφαλούς εσωτερικού γινομένου. Σε αυτήν την παράγραφο θα περιγράψουμε τις δύο αυτές τεχνικές και στο κεφάλαιο 4 θα τις εφαρμόσουμε σε χώρο-χρονικά δεδομένα.

3.5. 1 Ασφαλές άθροισμα (secure sum)

Η τεχνική του ασφαλούς αθροίσματος είναι από τις απλούστερες τεχνικές διατήρησης της ιδιωτικότητας σε υπολογισμούς μεταξύ πολλών συμμετεχόντων. Έστω n κάτοχοι δεδομένων, ο καθένας από τους οποίους έχει μία τιμή v_i , $i = 1, 2, ..., n$ και θέλουν να υπολογίσουν το άθροισμα $\sum_{i=1}^{n} v_i$ χωρίς να αποκαλύψουν τα δεδομένα τους. Η λύση είναι η εξής, ο πρώτος κάτοχος διαλέγει έναν τυχαίο αριθμό R και στέλνει το άθροισμα $v_1' = v_1 + R$ στον δεύτερο κάτοχο. Οι υπόλοιποι κάτοχοι , δηλαδή 2,3….n Λαμβάνουν το v'_{i-1} και στέλνουν το $v_i' = v'_{i-1} + v_i$ στον επόμενο κάτοχο. Ο κάτοχος n στέλνει το αποτέλεσμα πίσω στον πρώτο κάτοχο, ο οποίος απλά αφαιρεί τον τυχαίο αριθμό R και υπολογίζει το άθροισμα επιτυχώς. Η μέθοδος που θα υλοποιήσουμε στο κεφάλαιο 4 είναι μια παραλλαγή της τεχνικής του ασφαλούς αθροίσματος .

3.5. 2 Ασφαλές εσωτερικό γινόμενο (secure scalar product)

Το εσωτερικό γινόμενο είναι μια χρήσιμη ιδιότητα των διανυσμάτων και την οποία μπορούμε να χρησιμοποιήσουμε και στην περίπτωση των χώρο-χρονικών δεδομένων. Υπάρχουν αρκετές τεχνικές ασφαλούς υπολογισμού του εσωτερικού γινομένου. Η τεχνική που χρησιμοποιούμε στο επόμενο κεφάλαιο αφορά τον ασφαλή υπολογισμό του εσωτερικού γινομένου με τη βοήθεια ενός τρίτου συμμετέχοντα (commodity server).

Έστω οι δύο κάτοχοι δεδομένων A, B οι οποίοι κατέχουν δύο σύνολα $X = (x_1, x_2, ..., x_n)$ και $Y = (y_1, y_2, ..., y_n)$ αντίστοιχα.

Ο αλγόριθμος που προτείνεται στο [23] για τον υπολογισμό του εσωτερικού γινομένου $X \square Y = \sum_{i=1}^{n} x_i y_i$ χωρίς να αποκαλυφθούν τα δεδομένα του καθένα φαίνεται παρακάτω

Βήμα 1: Οι δύο κάτοχοι μοιράζονται με τη βοήθεια του commodity server τυχαίους αριθμούς ως εξής: ο κάτοχος A έχει δύο τυχαίους αριθμούς R_A, r_A και ο κάτοχος B R_B, r_B αντίστοιχα τέτοιους ώστε $r_A + r_B = R_A R_B$.

Βήμα 2 : Ο A υπολογίζει το X'=X + R_A και στέλνει το X' στον B

Ο B με την σειρά του υπολογίζει το Y'=Y+ R_B και στέλνει το Y' στον A

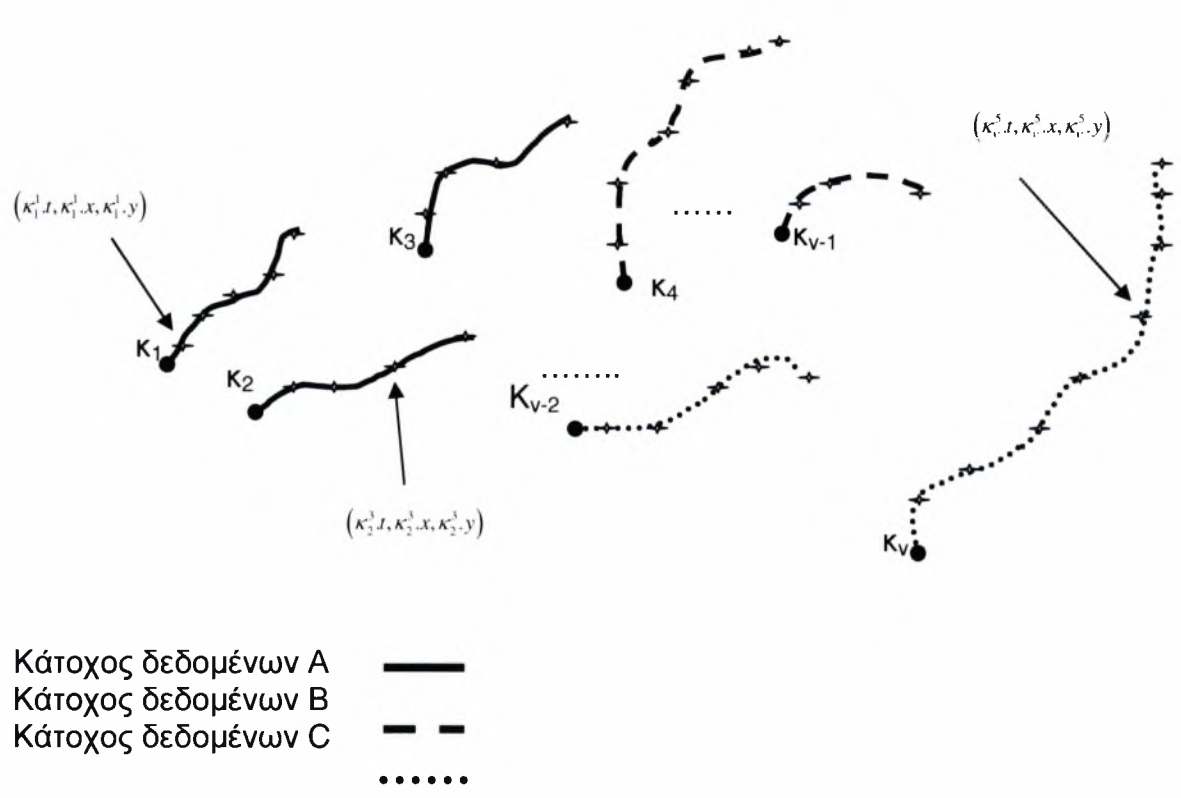
Βήμα 3 : Ο B υπολογίζει το X' Y + r_B και στέλνει το αποτέλεσμα στον A

Βήμα 4 : Ο A υπολογίζει το (X' Y + r_B)- (Y' R_A – r_A)= XY +(r_A – R_A R_B + r_B) = =XY

Με αυτόν τον τρόπο μπορούμε να υπολογίσουμε το γινόμενο $X \cdot Y = \sum_{i=1}^{n} x_i y_i$ με ασφαλή τρόπο.

3. 6 Διατύπωση του προβλήματος

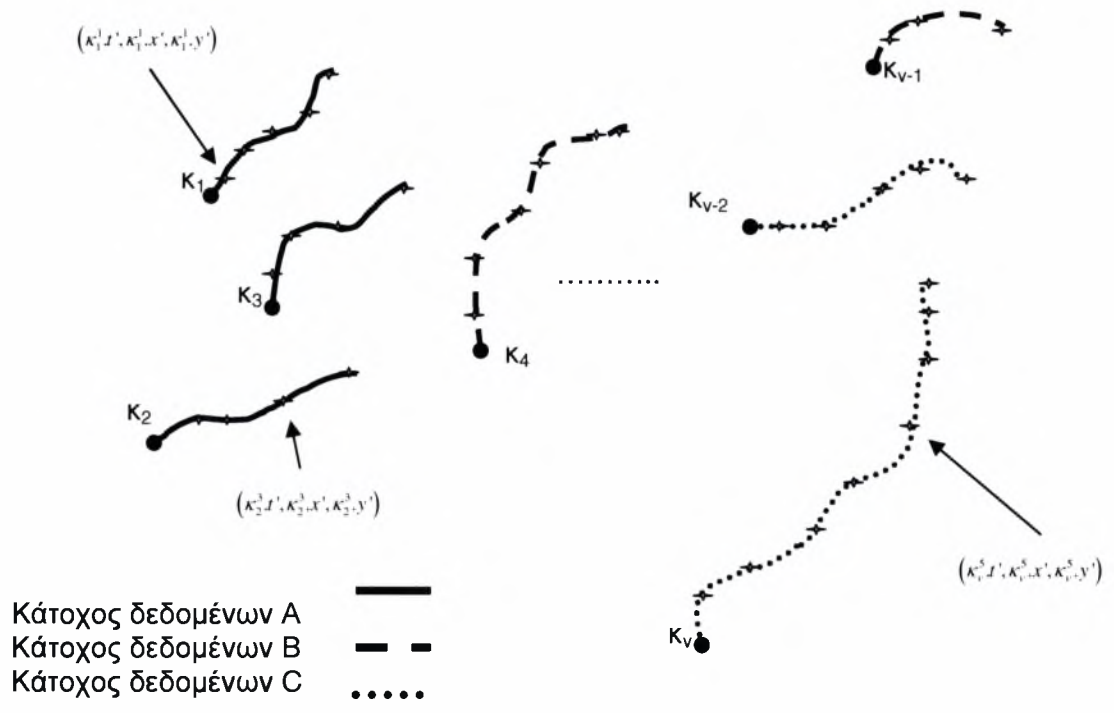
Στην παράγραφο αυτήν θα παρουσιάσουμε το πρόβλημα της συσταδοποίησης κινούμενων αντικειμένων, τα οποία διαμοιράζονται σε διαφορετικούς κατόχους δεδομένων, το πρόβλημα της ιδιωτικότητας των δεδομένων και τον ρόλο της εργασίας στην επίλυση των προβλημάτων αυτών.
Για να γίνει κατανοητό παραθέτουμε το παρακάτω σενάριο, έστω ότι θέλουμε να συσταδοποιήσουμε τα δρομολόγια των σχολικών λεωφορείων μιας πόλης για να ελαχιστοποιηθεί ο χρόνος αναμονής και τα άσκοπα δρομολόγια κάποιων λεωφορείων. Τα δεδομένα, οι τροχιές δηλαδή των λεωφορείων, συλλέγονται από GSM χειριστές, οι οποίοι όμως ίσως να μην είναι πρόθυμοι να μοιραστούν τα δεδομένα τους. Επομένως για να μπορέσουμε να βγάλουμε χρήσιμα και ακριβή συμπεράσματα για τα δρομολόγια των λεωφορείων πρέπει να εφαρμόσουμε ένα πρωτόκολλο διατήρησης της ιδιωτικότητας, ούτως ώστε να γίνει η ανταλλαγή των δεδομένων χωρίς να αποκαλυφθούν τα δεδομένα του κάθε χειριστή GSM.
Έστω ένα σύνολο ν κινούμενων αντικειμένων, λεωφορείων , $K = (\kappa_1, \kappa_2, \ldots, \kappa_\nu)$, και τρείς GSM χειριστές, κάτοχοι δεδομένων A,B,C , στους οποίους διαμοιράζονται τα δεδομένα. Τα κινούμενα αντικείμενα έχουν τροχιές διαφορετικού μήκους και απεικονίζονται στην εικόνα 5 .
Τα δεδομένα διαμοιράζονται οριζοντίως, επομένως κάθε κάτοχος δεδομένων έχει ένα σύνολο αντικειμένων και γνωρίζει τις τροχιές των αντικειμένων που κατέχει δηλαδή όπως φαίνεται στην εικόνα 5 ο κάτοχος A έχει το υποσύνολο $K^A = (\kappa_1, \kappa_2, \kappa_3)$ και άρα τις τροχιές Traj (κ_1), Traj (κ_2), Traj (κ_3). Ομοίως οι κάτοχοι B,C κατέχουν τα υποσύνολα $K^B = (\kappa_4, \kappa_{\nu-1})$ και $K^C = (\kappa_{\nu-2}, \kappa_\nu)$ αντίστοιχα.



Εικόνα 5 Ανάθεση αντικειμένων στους κατόχους δεδομένων

Οι τρεις κάτοχοι επιθυμούν να συσταδοποιήσουν τα αντικείμενα τους χωρίς όμως να αποκαλύψουν τα δεδομένα τους, δηλαδή τις τροχιές των αντικειμένων τους. Όπως αναφέραμε η συσταδοποίηση μπορεί να γίνει αρκετά πιο εύκολη κατασκευάζοντας τον πίνακα ανομοιότητας.

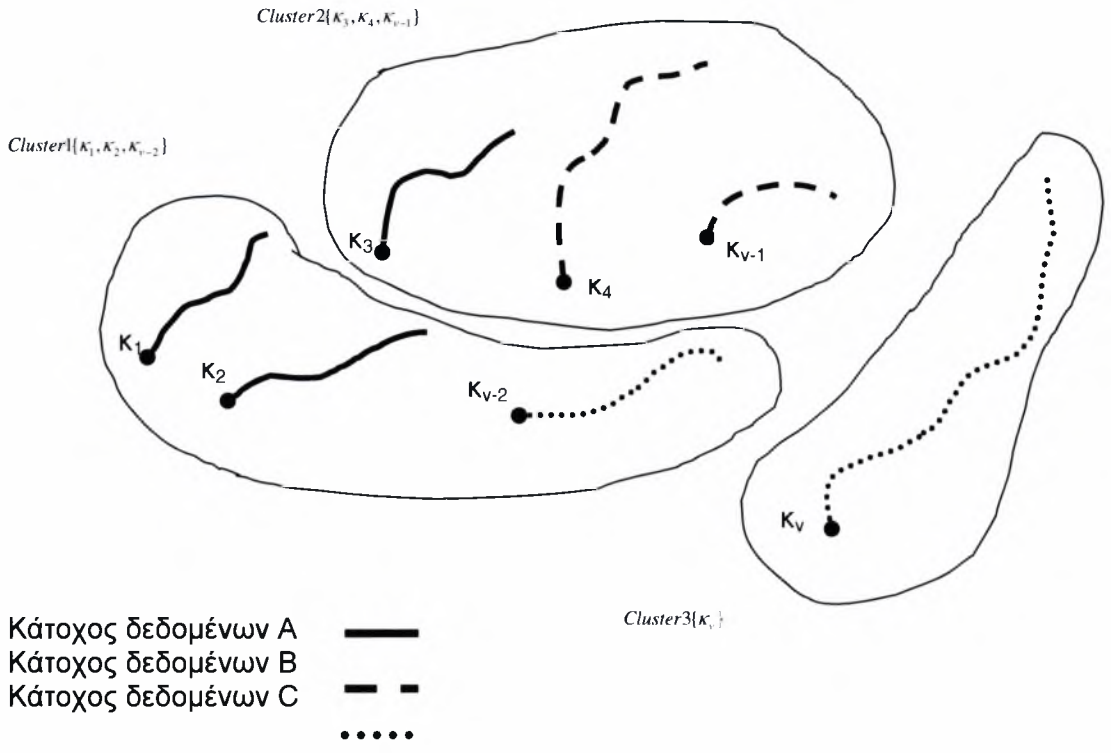
Προφανώς για να συσταδοποιήσουμε τα αντικείμενα θα πρέπει οι κάτοχοι να συγκρίνουν και επομένως να εμφανίσουν τις τροχιές των αντικειμένων τους. Στο επόμενο κεφάλαιο 4 παρουσιάζουμε δύο μεθόδους, με τις οποίες επιτυγχάνεται ασφαλής ανταλλαγή δεδομένων ανάμεσα στους κατόχους. Κάθε κάτοχος δεδομένων τροποποιεί τις παρατηρήσεις των τροχιών των αντικειμένων που του ανήκουν και τις στέλνει στους υπόλοιπους κατόχους. Όπως φαίνεται και στην εικόνα 6 ο κάτοχος Α τροποποιεί τις παρατηρήσεις των αντικειμένων κ_1,κ_2,κ_3 και σχηματίζονται καινούργιες τροχιές, οι οποίες δεν μπορούν να αποκαλύψουν την τιμή των αρχικών δεδομένων. Ομοίως και οι άλλοι δύο κάτοχοι τα δικά τους δεδομένα. Ο στόχος και των δύο μεθόδων είναι η κατασκευή ενός προσωρινού πίνακα, ο οποίος περιέχει τις αποστάσεις μεταξύ όλων των πραγματικών παρατηρήσεων της τροχιάς ενός αντικειμένου με όλες τις αντίστοιχες παρατηρήσεις ενός αντικειμένου διαφορετικού κατόχου μέσω ενός ασφαλούς πρωτοκόλλου ανταλλαγής δεδομένων. Προφανώς στους υπολογισμούς πρέπει να λάβει μέρος και ένας τρίτος συμμετέχοντας, αφού δεν έχει νόημα η ασφαλής ανταλλαγή δεδομένων μεταξύ δύο κατόχων όταν ο ένας από τους δύο μαθαίνει την απόσταση των δεδομένων. Ο τρίτος συμμετέχοντας, τον οποίο ονομάζουμε TP (trusted party), λειτουργεί σαν ένα υπολογιστικό και αποθηκευτικό μέσο και ο ρόλος του θα αναλυθεί εκτενέστερα στα επόμενα κεφάλαια.



Εικόνα 6 Τροποποίηση τροχιών κινούμενων αντικειμένων

Στην συνέχεια χρησιμοποιούμε τον προσωρινό πίνακα σαν είσοδο για τις συναρτήσεις σύγκρισης που αναφέραμε στην ενότητα 3.4 και κατασκευάζεται ο πίνακας ανομοιότητας. Πλέον είναι αρκετά εύκολο με κάποιον αλγόριθμο συσταδοποίησης να λάβουμε το τελικό αποτέλεσμα (εικόνα 7) το οποίο είναι ίδιο με αυτό που θα παίρναμε αν οι κάτοχοι αντάλλασαν τα δεδομένα τους χωρίς να τους ενδιέφερε η διατήρηση της ιδιωτικότητας.

Ο σκοπός της εργασίας αυτής είναι η σύγκριση δύο μεθόδων διατήρησης της ιδιωτικότητας. Αυτό που θα μας απασχολήσει είναι κυρίως ποια μέθοδος είναι ασφαλέστερη και σαφώς το υπολογιστικό κόστος των δύο αυτών μεθόδων μέσα από ένα σύνολο πειραμάτων σε διάφορα σύνολα δεδομένων.



Εικόνα 7 Αποτέλεσμα συσταδοποίησης

Κεφάλαιο 4

Μέθοδοι διατήρησης της ιδιωτικότητας

4. 1 Εισαγωγή

Το κεφάλαιο αυτό αποτελεί ουσιαστικά το σημαντικότερο κομμάτι της εργασίας αυτής. Σε αυτό το κεφάλαιο παρουσιάζουμε δύο μεθόδους για την διατήρηση της ιδιωτικότητας των τροχιών κινούμενων αντικειμένων καθώς και τους αλγορίθμους για την υλοποίηση τους. Η πρώτη μέθοδος είναι από το [24] των Ali Inan, Yusel Saygin είναι μια παραλλαγή της τεχνικής ασφαλούς αθροίσματος (secure sum) που αναφέραμε στην ενότητα 3.5 .1, ενώ η δεύτερη μέθοδος βασίζεται στον συνδυασμό της ευκλείδειας απόστασης και του ασφαλούς εσωτερικού γινομένου διανυσμάτων (secure scalar product ,ενότητα 3.5.2). Οι δύο μέθοδοι αφορούν την διατήρηση της ιδιωτικότητας σε οριζόντια διαχωριζόμενα δεδομένα, επομένως το πρωτόκολλο που θα ακολουθήσουμε για την επίλυση είναι το ίδιο και για τις δύο μεθόδους. Η διαφορά των δύο μεθόδων είναι οι τεχνικές διατήρησης της ιδιωτικότητας και αυτό που θα μας απασχολήσει είναι ποια μέθοδος εγγυάται μεγαλύτερη ασφάλεια με τον λιγότερο υπολογιστικό κόστος.

4. 2 Πρωτόκολλο

Έστω ότι υπάρχουν n κάτοχοι δεδομένων, $n \geq 2$, οι οποίοι κατέχουν οριζόντια τμήματα του πίνακα δεδομένων. Οι κάτοχοι δεδομένων έχουν ένα συγκεκριμένο αριθμό αντικειμένων και θέλουν να συσταδοποιήσουν τις τροχιές τους, χωρίς όμως να αποκαλύψουν πληροφορίες για την θέση των αντικειμένων.
Σύμφωνα με το πρωτόκολλο εκτός από τους κατόχους υπάρχει και μια τρίτη οντότητα TP, η οποία χρησιμοποιείται ως μέσο υπολογισμού και αποθηκευτικού χώρου. Ο ρόλος του TP στο πρωτόκολλο είναι ο εξής

1) διευθύνει την επικοινωνία μεταξύ των κατόχων δεδομένων
2) κατασκευάζει τον καθολικό πίνακα ανομοιότητας
3) συσταδοποιεί τις τροχιές χρησιμοποιώντας πίνακα ανομοιότητας
4) γνωστοποιεί τα αποτελέσματα στους κατόχους

Όσον αφορά την πρώτη μέθοδο των Ali Inan, Yusel Saygin έχουν γίνει οι εξής παραδοχές, οι κάτοχοι όπως και το TP είναι semi-honest δηλαδή ακολουθούν το πρωτόκολλο όπως ορίζεται όμως μπορούν να χρησιμοποιήσουν οποιαδήποτε πληροφορία τους είναι διαθέσιμη για προσωπικό τους όφελος. Επίσης καμία οντότητα δεν είναι συνεργάσιμη με μια άλλη δηλαδή δεν αποκαλύπτει ιδιωτικές πληροφορίες σε άλλες οντότητες. Όμως στην πραγματικότητα κάτι τέτοιο δεν μπορεί να ισχύει, αφού η συνεργασία μεταξύ κατόχων δεν μπορεί να αποφευχθεί, όπως επίσης είναι σχεδόν αδύνατο να βρεθεί ένας τρίτος έμπιστος συμμετέχοντας (trusted party). Αυτό που πετυχαίνουμε με την δεύτερη μέθοδο είναι να περιορίσουμε τον

ρόλο του τρίτου συμμετέχοντα, ούτως ώστε να γνωρίζει όσο το δυνατόν λιγότερες πληροφορίες.
Όπως αναφέραμε και προηγουμένως στη μέθοδο συμμετέχουν οι κάτοχοι δεδομένων και το ΤΡ. Στην περίπτωση που τα αντικείμενα ανήκουν σε έναν κάτοχο δεδομένων η συμμετοχή του ΤΡ στους υπολογισμούς δεν είναι απαραίτητη. Ο κάτοχός υπολογίζει την απόσταση των αντικειμένων και στέλνει τα αποτελέσματα στο ΤΡ.
Στην περίπτωση όμως που τα αντικείμενα ανήκουν σε διαφορετικούς κατόχους πρέπει να εφαρμόσουμε μια από τις δύο μεθόδους διατήρησης της ιδιωτικότητας ανάμεσα στους κατόχους. Από τα προηγούμενα συνεπάγεται ότι για την κατασκευή του καθολικού πίνακα ανομοιότητας κάθε κάτοχος πρέπει να υπολογίσει τον τοπικό πίνακα ανομοιότητας και να τον στείλει στο ΤΡ, όπως επίσης και να «τρέξει» το πρωτόκολλο με όλους τους άλλους κατόχους. Επομένως αν υπάρχουν n κάτοχοι δεδομένων το πρωτόκολλο πρέπει να επαναληφθεί C(n,2) φορές για κάθε ζεύγος κατόχων ούτως ώστε το ΤΡ να κατασκευάσει τον καθολικό πίνακα ανομοιότητας και να επιστρέψει τα αποτελέσματα της συσταδοποίησης.
Στις επόμενες ενότητες 4.3, 4.4 παραθέτουμε τις δύο μεθόδους για την διατήρηση της ιδιωτικότητας των δεδομένων. Η παρουσίαση θα γίνει με δύο κατόχους δεδομένων έστω Α και Β και τον τρίτο συμμετέχοντα ΤΡ. Αρχικά θα θεωρήσουμε ότι οι δύο κάτοχοι δεδομένων έχουν στην κατοχή τους δύο ακεραίους x_A,x_B, αντίστοιχα και όχι χώρο-χρονικά δεδομένα και στην συνέχεια παραθέτουμε σε μορφή ψευδό- κώδικα την μεταφορά των μεθόδων σε χώρο-χρονικά δεδομένα.

4. 3 Μέθοδος πρώτη

Για την διατήρηση της ιδιωτικότητας των δεδομένων η μέθοδος των Ali Inan, Yusel Saygin ορίζει τη χρήση δύο τυχαίων αριθμών (κλειδιών) R_{AB}, R_{AT}. Το πρώτο κλειδί μοιράζονται οι δύο κάτοχοι Α, Β ενώ το δεύτερο ο κάτοχος Α και το ΤΡ.
Οι κάτοχοι Α, Β θέλουν να υπολογίσουν την απόσταση των δύο ακεραίων x_A,x_B, η οποία δίνεται από τον τύπο

$$dist(x_A, x_B) = |x_A - x_B|$$

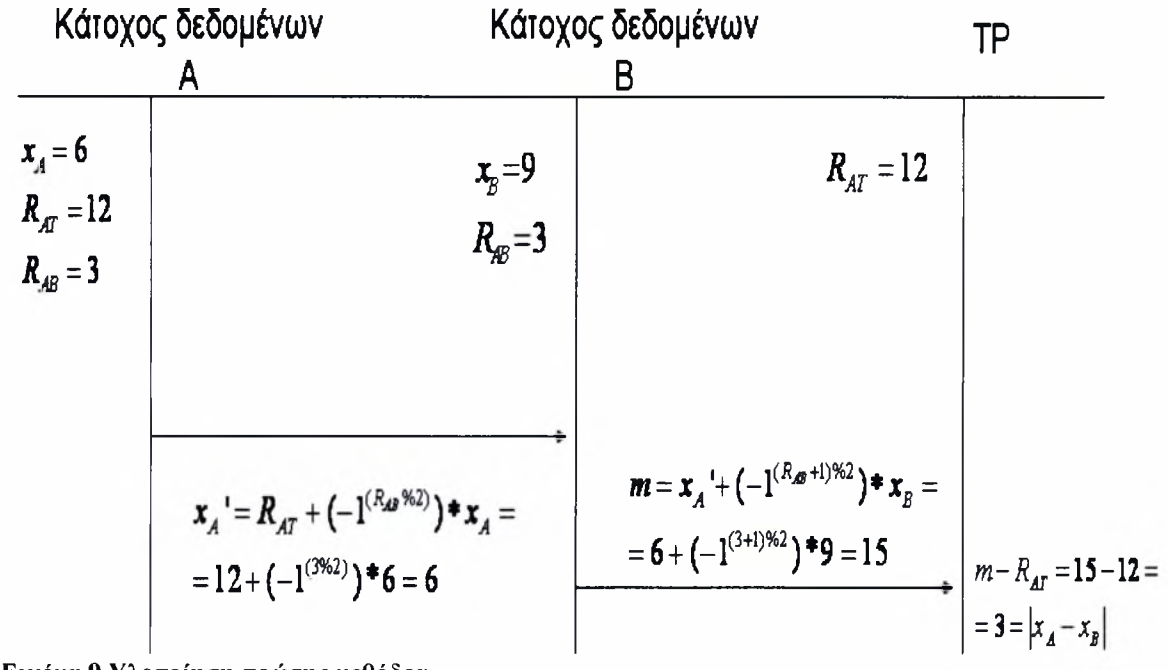
Αρχικά θα εξηγήσουμε το προφανές, ότι δηλαδή χωρίς τη χρήση των δύο κλειδιών υπάρχει απώλεια πληροφορίας και στην συνέχεια θα εξηγήσουμε την αναγκαιότητα χρήσης και των δυο κλειδιών για την διατήρηση της ιδιωτικότητας των δεδομένων, δείχνοντας ότι με τη χρήση μόνο ενός εκ των δύο κλειδιών μπορεί να έχουμε απώλεια πληροφορίας.
Χωρίς τη χρήση κλειδιών ο κάτοχος Α μπορεί να υπολογίσει την τιμή του ακεραίου x_B. Αυτό μπορεί να το πετύχει απλά μαθαίνοντας την απόλυτη τιμή της απόστασης των δύο ακεραίων, δηλαδή να υπολογίσει $x_B = x_A + | x_A - x_B |$ αν ο ακέραιος x_A είναι μεγαλύτερος από τον x_B, είτε $x_B = x_A - | x_A - x_B |$ αν ισχύει το αντίστροφο. Σε περιπτώσεις δεδομένων των οποίων οι τιμές έχουν ελάχιστο και μέγιστο όριο, όπως για παράδειγμα ηλικία, βάρος, ο Α μπορεί να υπολογίσει την τιμή του x_B απορρίπτοντας την τιμή που είναι εκτός ορίων.
Έστω τώρα ότι το πρωτόκολλο ορίζει μόνο την χρήση του κλειδιού R_{AT}. Ο Α προσθέτει στον ακέραιο x_A το κλειδί R_{AT} και στέλνει το $x_A' = x_A + R_{AT}$ στον Β. Μέχρι αυτό το σημείο δεν υπάρχει απώλεια πληροφορίας αφού το κλειδί R_{AT}

μεταβάλλει την τιμή του ακεραίου x_A. Ο Β με την σειρά του αφαιρεί το x_B από το $x_A' = x_A + R_{AT}$ και στέλνει το αποτέλεσμα $m = x_A' - x_B = x_A + R_{AT} - x_B$ στο ΤΡ.

Ο ΤΡ γνωρίζει την τιμή του κλειδιού R_{AT} και έτσι αφαιρώντας το κλειδί υπολογίζει την απόσταση $| x_A - x_B |$. Όμως τώρα ο ΤΡ γνωρίζει αν ο ακέραιος x_A είναι μεγαλύτερος από τον x_B ή το αντίθετο, απλά παρατηρώντας το πρόσημο της απόστασης. Μπορεί ο ΤΡ να μην γνωρίζει την ακριβή τιμή των δυο ακεραίων, ωστόσο γνωρίζοντας ποιος ακέραιος είναι μεγαλύτερος, προκύπτει απώλεια πληροφορίας καθώς για παράδειγμα οι δύο ακέραιοι μπορεί να αφορούν τα κέρδη ή τα έσοδα δύο εταιριών.
Για να λυθεί αυτό το πρόβλημα η μέθοδος ορίζει την χρήση του κλειδιού R_{AB}. Αν το κλειδί R_{AB} είναι περιττός ο Α βάζει το πρόσημο «-» στην είσοδο του, αρνητικοποιεί δηλαδή το x_A , διαφορετικά αν δηλαδή το κλειδί R_{AB} είναι άρτιος, ο κάτοχος Β αρνητικοποιεί την είσοδο του x_B.
Με αυτόν τον τρόπο ο ΤΡ δεν μπορεί να γνωρίζει ποιος ακέραιος είναι μεγαλύτερος και δεν υπάρχει απώλεια πληροφορίας.
Στην εικόνα 8 βλέπουμε την υλοποίηση της μεθόδου με τη χρήση και των δύο κλειδιών

Εικόνα 8 Μέθοδος Ali Inan ,Yucel Saygin

Επομένως η χρήση και των δύο κλειδιών είναι απαραίτητη για την διατήρηση της ιδιωτικότητας. Παρακάτω στην εικόνα 9 φαίνεται η υλοποίηση της μεθόδου για τους δύο κατόχους Α, Β. Για να γίνει κατανοητή η βασική ιδέα της μεθόδου υποθέσαμε ότι οι δύο ακέραιοι x_A,x_B έχουν τις τιμές 6,9 αντίστοιχα και τα κλειδιά R_{AB} και R_{AT} τις τιμές 3 και 12 αντίστοιχα.



Εικόνα 9 Υλοποίηση πρώτης μεθόδου

4.3. 1 Υλοποίηση της μεθόδου

Σε αυτήν την ενότητα θα δείξουμε πως υλοποιείται η μέθοδος που αναλύσαμε σε χώρο-χρονικά δεδομένα Αρχικά ο κάτοχος Α δημιουργεί δυο ψευδό-τυχαίους αριθμούς r_{AB}, r_{AT} με βάση τα κλειδιά R_{AB} R_{AT} που μοιράζεται με τον κάτοχο Β και το ΤΡ αντίστοιχα. Στην συνέχεια μετατρέπει κάθε διάσταση των χώρο-χρονικών δεδομένων δηλαδή, τον χρόνο t και το διάνυσμα της θέσης x , y , ως εξής :
Αν ο ψευδό-τυχαίος αριθμός r_{AB} είναι περιττός ο κάτοχος Α αρνητικοποιεί την είσοδο του και προσθέτει τον ψευδό-τυχαίο αριθμό r_{AT}. Τέλος στέλνει τις αλλαγμένες τιμές των διαστάσεων στον κάτοχο Β .
Παρακάτω φαίνεται ο ψευδό-κώδικας για τον κάτοχο Α.

INPUT : K^A, R_{AB}, R_{AT}
OUTPUT: πίνακας ΠΑ
Begin
1.Αρχικοποίηση πίνακα ΠΑ= {K^A.length }
for (i=0 **to** K^A.length)
2. Αρχικοποίηση πίνακα ΠΑ[i]={Traj(κ_i).length }
3. Αρχικοποίηση ψευδό-τυχαίου αριθμού r_{AB} με βάση τα κλειδιά R_{AB}
4. Αρχικοποίηση ψευδό-τυχαίου αριθμού r_{AT} με βάση τα κλειδιά R_{AT}
for (j=0 **to** Traj(κ_i).length)

5.1 $\Pi A[i][j].t = r_{AT} + K^{A}[i][j].t * (-1^{(r_{AB} \% 2)})$

5.2 $\Pi A[i][j].x = r_{AT} + K^{A}[i][j].x * (-1^{(r_{AB} \% 2)})$

5.3 $\Pi A[i][j].y = r_{AT} + K^{A}[i][j].y * (-1^{(r_{AB} \% 2)})$

6.Αποστολή πίνακα ΠΑ στο ΤΡ

End

Ψευδό-κώδικας 4 Πρώτη Μέθοδος-Κάτοχος δεδομένων Α

Ο κάτοχος Β αφού λάβει τον πίνακα ΠΑ με τις αλλαγμένες τιμές των δεδομένων του κατόχου Α, αρχικοποιεί έναν πίνακα ΠΒ και δημιουργεί ένα ψευδό-τυχαίο αριθμό r_{AB} με το κλειδί R_{AB}, δηλαδή τον ίδιο ψευδό-τυχαίο αριθμό που δημιούργησε και ο κάτοχος Α. Επομένως ο Β αλλάζει το πρόσημο της εισόδου του με τρόπο παρόμοιο με τον Α, μόνο που σε αυτή την περίπτωση την αρνητικοποιεί την είσοδο του αν ο r_{AB} είναι άρτιος. Έπειτα ο Β απλά προσθέτει την είσοδο του με αυτήν που έλαβε από τον κάτοχο Α. Σύμφωνα με τα παραπάνω κάθε στοιχείο του πίνακα ΠΒ π.χ. το ΠΒ[2][3] θα περιέχει την απόσταση του δεύτερου αντικειμένου του συνόλου K^B από αυτήν του τρίτου αντικειμένου του συνόλου K^B. Προφανώς η απόσταση αυτή δεν είναι η πραγματική καθώς τα δεδομένα του Α έχουν αλλαχθεί.
Παρακάτω φαίνεται ο ψευδό-κώδικας για τον κάτοχο δεδομένων Β.

INPUT : K^B, R_{AB}, ΠΑ

OUTPUT: πίνακας ΠΒ

Begin

1.Αρχικοποίηση πίνακα ΠΒ= {K^B.length x ΠΑ.length }

for (i=0 **to** K^B.length)

for (j=0 **to** ΠΑ.length)

2. Αρχικοποίηση πίνακα ΠΒ[i][j]={Traj(κ_i).length x ΠΑ[j].length }

for (m=0 **to** Traj(κ_i).length)

3. Αρχικοποίηση ψευδό-τυχαίου αριθμού r_{AB} με βάση τα κλειδιά R_{AB}

for (n=0 **to** ΠΑ[j].length)

4.1 $\Pi B[i][j][m][n].t = \Pi A[j][n].t + K^{B}[i][m].t * (-1^{((r_{AB}+1) \% 2)})$

4.2 $\Pi B[i][j][m][n].x = \Pi A[j][n].x + K^{B}[i][m].x * (-1^{((r_{AB}+1) \% 2)})$

4.3 $\Pi B[i][j][m][n].y = \Pi A[j][n].y + K^{B}[i][m].y * (-1^{((r_{AB}+1) \% 2)})$

5.Αποστολή πίνακα ΠΒ στο ΤΡ

End

Ψευδό-κώδικας 5 Πρώτη Μέθοδος - Κάτοχος δεδομένων Β

Ο TP λαμβάνει τον πίνακα ΠΒ από τον κάτοχο Β και αρχικοποιεί τον ψευδό-τυχαίο αριθμό r_{AT} με το κλειδί R_{AT} που μοιράζεται με τον κάτοχο Α. Στην συνέχεια αφαιρεί τον r_{AT} και δημιουργεί έναν πίνακα Τ ο οποίος περιέχει σε απόλυτη τιμή τις αποστάσεις μεταξύ όλων των παρατηρήσεων όλων των αντικειμένων. Για να γίνει πιο κατανοητό, το στοιχείο του πίνακα T[2][3] [1][1] θα περιέχει την απόλυτη τιμή της απόστασης της πρώτης παρατήρησης του δεύτερου αντικειμένου του Β από την πρώτη παρατήρηση του τρίτου αντικειμένου του κατόχου Α.
Η απόσταση αυτή είναι ότι χρειάζεται μια συνάρτηση σύγκρισης, όπως αναφέραμε στο τρίτο κεφάλαιο, για να υπολογίσει την απόσταση της τροχιάς ενός αντικειμένου από ενός άλλου.
Παρακάτω φαίνεται ο ψευδό-κώδικας για τον TP.

INPUT : πίνακας ΠΒ, R_{AT}

OUTPUT: πίνακας Τ

Begin

1.Αρχικοποίηση πίνακα T= { ΠΒ.length }

for (i=0 **to** ΠΒ.length)

2. Αρχικοποίηση πίνακα T[i]={ ΠΒ[i].length }

for (j=0 **to** ΠΒ[i].length)

3. Αρχικοποίηση πίνακα T[i][j]={ ΠΒ[i][j].length }

for(m=0 **to** ΠΒ[i][j].length)

4. Αρχικοποίηση πίνακα T[i][j][m]={ ΠΒ[i][j][m].length }

5. Αρχικοποίηση ψευδό-τυχαίου αριθμού r_{AT} με βάση τα κλειδιά R_{AT}

for(n=0 **to** ΠΒ[i][j][m].length)

6.1 $T[i][j][m][n].t = |\Pi B[i][j][m][n].t - r_{AT}|$

6.2 $T[i][j][m][n].x = |\Pi B[i][j][m][n].x - r_{AT}|$

6.3 $T[i][j][m][n].y = |\Pi B[i][j][m][n].y - r_{AT}|$

End

Ψευδό-κώδικας 6 Πρώτη Μέθοδος -Trusted Party

Στην περίπτωση που ο αριθμός των κατόχων δεδομένων είναι μεγαλύτερος του 2 τότε όπως αναφέραμε και προηγουμένως το πρωτόκολλο θα πρέπει να επαναληφθεί για όλα τα ζευγάρια των κατόχων. Πιο συγκεκριμένα, για κάθε ζεύγος κατόχων, ένας θα παίρνει τον ρόλο του Α και ο άλλος τον ρόλο του Β.

4. 4 Δεύτερη Μέθοδος

Η δεύτερη μέθοδος που υλοποιούμε στην παρούσα εργασία βασίζεται στην τεχνική του ασφαλούς εσωτερικού γινομένου και στον τύπο της ευκλείδειας απόστασης. Όπως όμως αναφέραμε σε υπολογισμούς μεταξύ πολλών συμμετεχόντων, όταν αυτοί αφορούν την απόσταση χρειαζόμαστε και την συμμετοχή ενός τρίτου συμμετέχοντα TP. Επομένως θα διαφοροποιήσουμε την τεχνική του ασφαλούς εσωτερικού γινομένου της ενότητας 3.5.2 με την παρουσία του TP.

4.4. 1 Ασφαλές εσωτερικό γινόμενο τριών συμμετεχόντων

Έστω οι κάτοχοι δεδομένων A, B οι οποίοι κατέχουν δύο σύνολα $X = (x_1, x_2, ..., x_n)$ και $Y = (y_1, y_2, ..., y_n)$ και μοιράζονται τα κλειδιά R_A, r_A και R_B, r_B αντίστοιχα. Ο αλγόριθμος για τον υπολογισμό του εσωτερικού γινομένου $X \cdot Y = \sum_{i=1}^{n} x_i y_i$ με την παρουσία τρίτου συμμετέχοντα φαίνεται παρακάτω

Βήμα 1 : Ο A υπολογίζει το X'=X + R_A και στέλνει το X' στον B

Ο B με την σειρά του υπολογίζει το Y'=Y+ R_B και στέλνει το Y' στον A

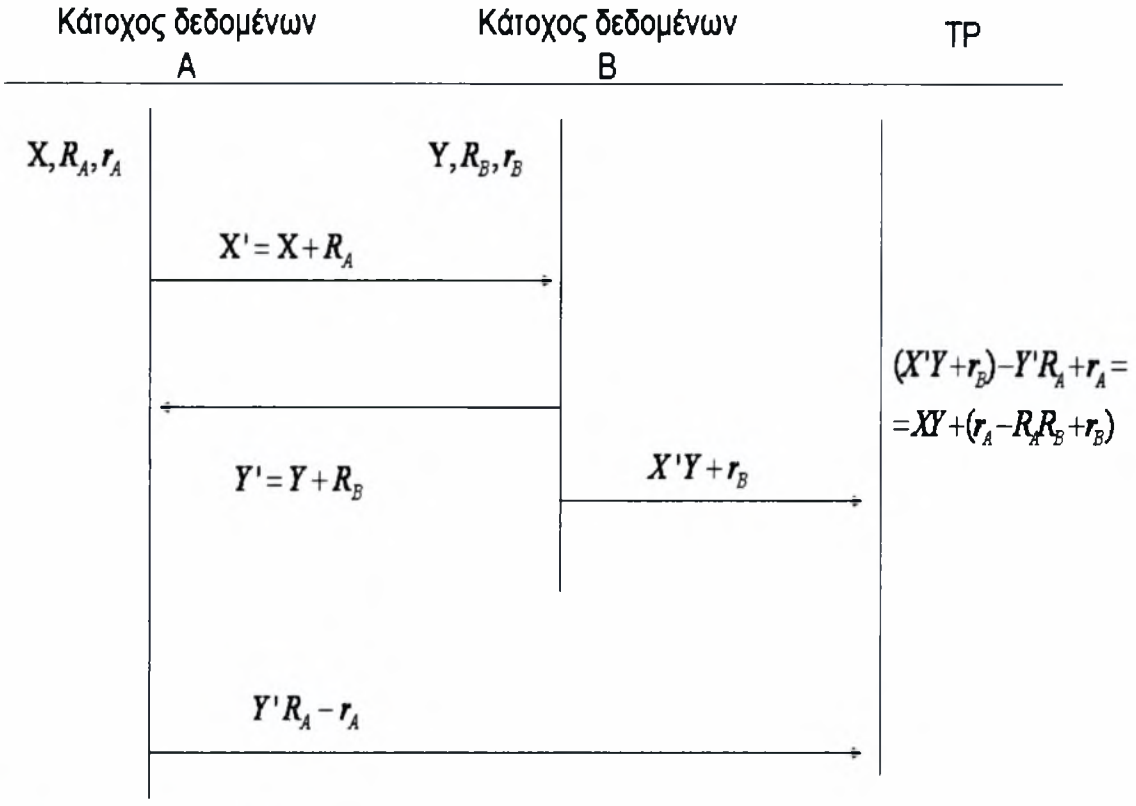
Βήμα 2 : Ο B υπολογίζει το X' Y + r_B και στέλνει το αποτέλεσμα στον τρίτο συμμετέχοντα TP

Βήμα 3 : Ο A υπολογίζει το Y' R_A – r_A και στέλνει το αποτέλεσμα στο TP

Βήμα 4: Το TP υπολογίζει απλά (X' Y + r_B)– Y' R_A + r_A =

=XY +(r_A – R_A R_B + r_B) = XY

Αρχικά στο βήμα 1 οι δύο κάτοχοι ανταλλάσουν τα δεδομένα τους τροποποιημένα με τα κλειδιά R_A, R_B, επομένως δεν υπάρχει κάποια απώλεια πληροφορίας. Στην συνέχεια του αλγορίθμου και στο βήμα 2, ο κάτοχος B υπολογίζει το γινόμενο X'Y, προσθέτει το κλειδί r_B και το στέλνει στο TP, το οποίο δεν μπορεί να ανακτήσει τις αρχικές τιμές των διανυσμάτων, αφού δεν έχει κάποια γνώση για τα κλειδιά. Ομοίως ο κάτοχος A, στέλνει το Y' τροποποιημένο με τα κλειδιά που κατέχει R_A,r_A στο TP. Μέχρι αυτό το σημείο η μέθοδός 2 εγγυάται μεγαλύτερη ασφάλεια από την πρώτη μέθοδο των Ali Inan, Yucel Saygin, αφού η τροποποίηση των δεδομένων γίνεται με περισσότερα κλειδιά. Τέλος στο βήμα 4 του αλγορίθμου το TP υπολογίζει την διαφορά δύο αριθμών άγνωστων προς αυτό.
Στην παρακάτω εικόνα 10 φαίνεται η μέθοδος του ασφαλούς εσωτερικού γινομένου.



Εικόνα 10 Δεύτερη Μέθοδος

Η διαφορά αυτής της μεθόδου με την προηγούμενη είναι ότι τα κλειδιά μοιράζονται μόνο μεταξύ των κατόχων και όχι μεταξύ των κατόχων και του τρίτου συμμετέχοντα ΤΡ. Η διαφορά αυτή είναι σημαντική αν αναλογιστούμε ότι ακόμα και στην περίπτωση που συνεργαστεί ένας κάτοχος με το ΤΡ δεν θα υπάρξει απώλεια πληροφορίας. Ενώ για παράδειγμα στην προηγούμενη μέθοδο σε πιθανή συνεργασία του δεύτερου κατόχου με το ΤΡ, δηλαδή να αποκαλύψει ο ΤΡ το κλειδί R_{AT} στον Β έχουμε απώλεια πληροφορίας. Επιπλέον το ΤΡ δεν κρατάει κάποια πληροφορία όπως στην μέθοδο των Ali Inan,Yusel Saygin, το κλειδί R_{AT}. Επίσης είναι σημαντική διαφορά καθώς στην πραγματικότητα δεν μπορεί να υπάρξει έμπιστος τρίτος συμμετέχοντας. Σε αυτήν τη μέθοδο όπως δείξαμε παραπάνω ο ρόλος του ΤΡ είναι απλά ο υπολογισμός της αφαίρεσης δύο αριθμών που δέχεται από τους κατόχους, χωρίς να έχει στην κατοχή του κάποιο κλειδί ή κάποια γνώση για τα δεδομένα .

4.4. 2 Διατήρηση της ιδιωτικότητας συνδυάζοντας ευκλείδεια απόσταση και εσωτερικό γινόμενο.

Έστω τώρα ότι οι δύο κάτοχοι θέλουν να υπολογίσουν την απόσταση δύο ακεραίων, x_A τον οποίο κατέχει ο Α και x_B τον οποίο κατέχει ο Β. Πιο συγκεκριμένα την ευκλείδεια απόσταση των δύο ακεραίων

$$dist(x_A, x_B) = \sqrt{(x_A - x_B)^2} = \sqrt{x_A^2 - 2x_A x_B + x_B^2}$$

Η μέθοδος που αναλύουμε είναι απλή, ο κάτοχος Α κατασκευάζει διάνυσμα $X_A = (x_A^2, -2x_A, -1)$ ενώ ο Β διάνυσμα $X_B = (1, x_B, -x_B^2)$

Στην περίπτωση των χώρο-χρονικών δεδομένων οι κάτοχοι δημιουργούν διανύσματα $(x_A^2, -2x_A, -1)$, $(y_A^2, -2y_A, -1)$ και $(t_A^2, -2t_A, -1)$ ο πρώτος και $(1, x_B, -x_B^2)$, $(1, y_B, -y_B^2)$ και $(1, t_B, -t_B^2)$ ο δεύτερος ούτως ώστε να σχηματιστούν οι επιθυμητές αποστάσεις dist(x_A, x_B) , dist(y_A, y_B) , dist(t_A, t_B). Ωστόσο και σε αυτή τη μέθοδο για λόγους απλότητας θα δείξουμε την υλοποίηση για δύο ακεραίους x_A,x_B.

Επομένως υπολογίζοντας το εσωτερικό γινόμενο $X_A \square X_B$ προκύπτει η ευκλείδεια απόσταση των δύο αριθμών στο τετράγωνο. Δηλαδή

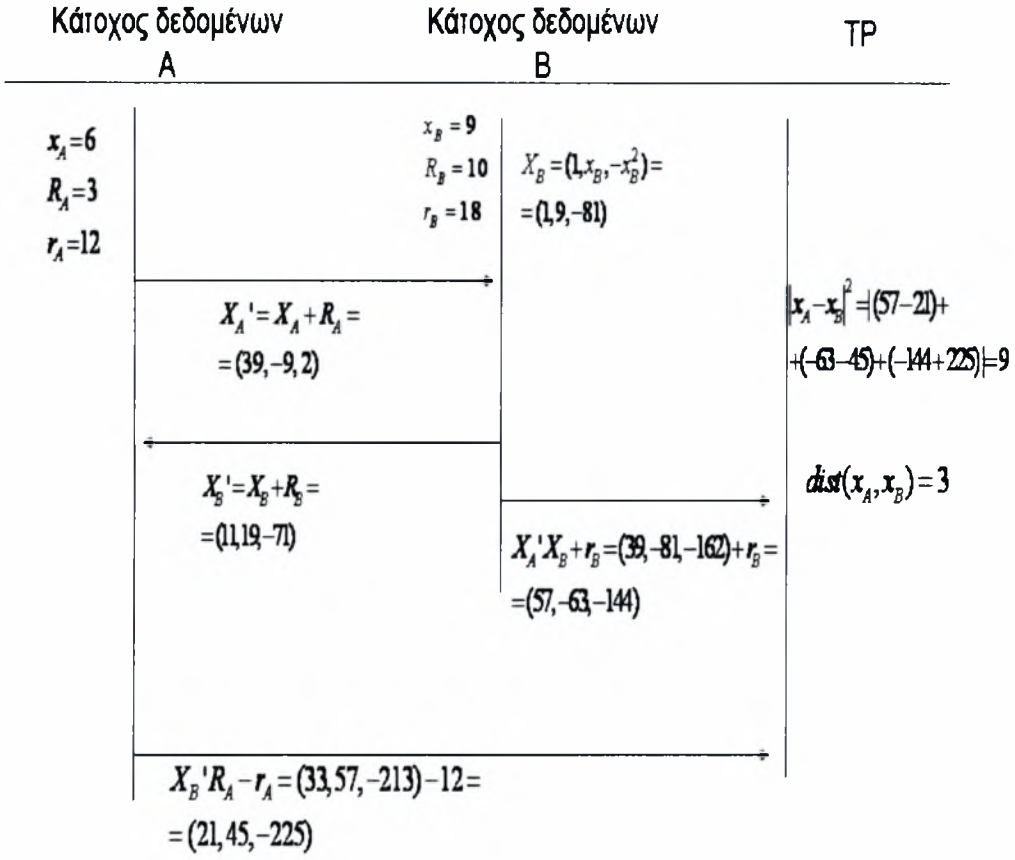
$$X_A \square X_B = (x_A^2, -2x_A, -1) \square (1, x_B, -x_B^2) =$$
$$= x_A^2 - 2x_A x_B + x_B^2 = dist(x_A, x_B)^2$$

Επιπλέον σε αυτή τη μέθοδο το TP δεν μπορεί να γνωρίζει ποιος ακέραιος είναι μεγαλύτερος αφού η απόσταση των δύο ακεραίων είναι στο τετράγωνο και έτσι δεν είναι απαραίτητη η χρήση ενός επιπλέον κλειδιού, όπως στη πρώτη μέθοδο το κλειδί R_{AB} .

Επομένως για να υπολογίσουμε ασφαλώς την απόσταση των δύο ακεραίων πρέπει να υπολογίσουμε με ασφαλή τρόπο όπως δείξαμε στην ενότητα 4.4.1 το εσωτερικό γινόμενο των δύο διανυσμάτων

Παρακάτω στην εικόνα 11 φαίνεται η υλοποίηση του πρωτοκόλλου για τους δύο κατόχους οι οποίοι κατέχουν τους ακεραίους x_A,x_B αντίστοιχα καθώς και τα κατάλληλα κλειδιά που αναφέραμε παραπάνω.

Για να γίνει κατανοητή η βασική ιδέα του πρωτοκόλλου υποθέσαμε ότι οι δύο ακέραιοι x_A,x_B έχουν τις τιμές 6,9 αντίστοιχα και τα κλειδιά $R_A = 3$και $r_A = 12$ και $R_B = 10$ και $r_B = 18$ ($r_A + r_B = R_A R_B$ =>18+12 =3*10) αντίστοιχα.



Εικόνα 11 Υλοποίηση δεύτερης μεθόδου

4.4. 3 Υλοποίηση της μεθόδου

Σε αυτήν την ενότητα θα δείξουμε πως υλοποιείται η μέθοδος που αναλύσαμε σε χώρο-χρονικά δεδομένα Αρχικά ο κάτοχος Α δημιουργεί τα διανύσματα που αναφέραμε για κάθε διάσταση των χώρο-χρονικών δεδομένων. Στην συνέχεια προσθέτει σε κάθε διάσταση δηλαδή, τον χρόνο t και το διάνυσμα της θέσης x, y, το κλειδί R_A. Ο πίνακας ΠΑ με τις τροποποιημένες τιμές των διανυσμάτων στέλνεται στον κάτοχο Β. Στην συνέχεια ο Α, όταν λάβει από τον κάτοχο Β τα τροποποιημένα δεδομένα του πολλαπλασιάζει τα δεδομένα αυτά με το κλειδί R_A και αφαιρεί το κλειδί r_A και αποθηκεύει τις τιμές στον πίνακα $ΠΤ_Α$ τον οποίο στέλνει στο ΤΡ.
Παρακάτω φαίνεται ο ψευδό-κώδικας για τον κάτοχο Α.

INPUT : K^A, ΠΒ, r_A, R_A

OUTPUT: πίνακες ΠΑ, $ΠΤ_A$

Begin

1.Αρχικοποίηση πίνακα ΠΑ= { K^A.length }

for (i=0 **to** K^A.length)

2. Αρχικοποίηση πίνακα ΠΑ [i]={ Traj($κ_i$).length }

for (j=0 **to** Traj($κ_i$).length)

3. Κατασκευή διανυσμάτων $\vec{t} = (t^2, -2t, -1), \vec{x} = (x^2, -2x, -1), \vec{y} = (y^2, -2y, -1)$

4.1 $ΠΑ[i][j].\vec{t} = R_A + K^A[i][j].\vec{t}$

4.2 $ΠΑ[i][j].\vec{x} = R_A + K^A[i][j].\vec{x}$

4.3 $ΠΑ[i][j].\vec{y} = R_A + K^A[i][j].\vec{y}$

end for

end for

for (i=0 **to** ΠΒ.length)

for (j=0 **to** ΠΒ[i].length)

5.1 $ΠΤ[i][j].\vec{t} = ΠΒ[i][j].\vec{t} * R_A - r_A$

5.2 $ΠΤ[i][j].\vec{x} = ΠΒ[i][j].\vec{x} * R_A - r_A$

5.3 $ΠΤ[i][j].\vec{y} = ΠΒ[i][j].\vec{y} * R_A - r_A$

6.Αποστολή πίνακα ΠΑ στον κάτοχο Β

7.Αποστολή πίνακα $ΠΤ_A$ στον ΤΡ

End

Ψευδό-κώδικας 7 Δεύτερη Μέθοδος-Κάτοχος δεδομένων Α

Ο κάτοχος Β αρχικά κατασκευάζει τα αντίστοιχα διανύσματα, ούτως ώστε να σχηματιστεί ο τύπος της ευκλείδειας απόστασης και αρχικοποιεί έναν πίνακα ΠΒ, ο οποίος περιέχει το σύνολο K^B τροποποιημένο με το κλειδί R_B. Ο Β στέλνει τον πίνακα ΠΒ στον κάτοχο Α.
Όταν αντίστοιχα ο κάτοχος Β λάβει τον πίνακα ΠΑ, αρχικοποιεί έναν πίνακα $ΠΤ_B$ τον οποίο στέλνει στο ΤΡ.

Παρακάτω φαίνεται ο ψευδό-κώδικας για τον κάτοχο δεδομένων Β.

INPUT : K^B, ΠΑ, r_B, R_B

OUTPUT: πίνακες ΠΒ, $ΠT_B$

Begin

1.Αρχικοποίηση πίνακα ΠΒ= { K^B.length }

for (i=0 **to** K^B.length)

2. Αρχικοποίηση πίνακα ΠΒ [i]={ Traj(κ_i).length }

for (j=0 **to** Traj(κ_i).length)

3. Κατασκευή διανυσμάτων $\vec{t} = (1, t, -t^2), \vec{x} = (1, x, -x^2), \vec{y} = (1, y, -y^2)$

4.1 $ΠB[i][j].\vec{t} = R_B + K^B[i][j].\vec{t}$

4.2 $ΠB[i][j].\vec{x} = R_B + K^B[i][j].\vec{x}$

4.3 $ΠB[i][j].\vec{y} = R_B + K^B[i][j].\vec{y}$

end for

end for

5.Αρχικοποίηση πίνακα $ΠT_B$ ={ΠΒ.length x ΠΑ.length }

for (i=0 **to** ΠΒ.length)

for (j=0 **to** ΠΑ.length)

6. Αρχικοποίηση πίνακα $ΠT_B$ [i][j]={K^B[i].length x ΠΑ[j].length}

for (i=0 **to** K^B[i].length)

for (j=0 **to** ΠΑ[j].length)

7.1 $ΠT_B[i][j][m][n].\vec{t} = ΠA[j][n].\vec{t} * K^B[i][m].\vec{t} + r_B$

7.2 $ΠT_B[i][j][m][n].\vec{x} = ΠA[j][n].\vec{x} * K^B[i][m].\vec{x} + r_B$

7.3 $ΠT_B[i][j][m][n].\vec{y} = ΠA[j][n].\vec{y} * K^B[i][m].\vec{y} + r_B$

8.Αποστολή πίνακα ΠΒ στον κάτοχο Α

9.Αποστολή πίνακα $ΠT_B$ στο ΤΡ

End

Ψευδό-κώδικας 8 Δεύτερη Μέθοδος- Κάτοχος δεδομένων Β

Ο ΤΡ λαμβάνει τους πίνακες $ΠT_B$ και $ΠT_A$ από τους κατόχους Β και Α αντίστοιχα και αρχικοποιεί έναν πίνακα Τ ο οποίος περιέχει τις αποστάσεις μεταξύ όλων των παρατηρήσεων όλων των αντικειμένων στο τετράγωνο. Για να γίνει πιο κατανοητό,

το στοιχείο του πίνακα T[2][3] [1][1] θα περιέχει το τετράγωνο της απόστασης της πρώτης παρατήρησης του δεύτερου αντικειμένου του Β από την πρώτη παρατήρηση του τρίτου αντικειμένου του κατόχου Α.
Παρακάτω φαίνεται ο ψευδό-κώδικας για τον TP.

INPUT : πίνακες ΠT_A, ΠT_B

OUTPUT: πίνακας T

Begin

1.Αρχικοποίηση πίνακα T= { ΠT_B.length }

for (i=0 **to** ΠT_B.length)

2. Αρχικοποίηση πίνακα T[i]={ ΠT_B[i].length }

for (j=0 **to** ΠT_B[i].length)

3. Αρχικοποίηση πίνακα T[i][j]={ ΠT_B[i][j].length }

for (m=0 **to** ΠT_B[i][j].length)

4. Αρχικοποίηση πίνακα T[i][j][m]={ ΠT_B[i][j][m].length }

for (n=0 **to** ΠT_B[i][j][m].length)

5.1 $T[i][j][m][n].\vec{t} = \Pi T_B[i][j][m][n].\vec{t} - \Pi T_A[j][n].\vec{t}$

5.2 $T[i][j][m][n].\vec{x} = \Pi T_B[i][j][m][n].\vec{x} - \Pi T_A[j][n].\vec{x}$

5.3 $T[i][j][m][n].\vec{y} = \Pi T_B[i][j][m][n].\vec{y} - \Pi T_A[j][n].\vec{y}$

End

Ψευδό-κώδικας 9 Δεύτερη Μέθοδος -Trusted Party

Όπως και στην πρώτη μέθοδο, στην περίπτωση που ο αριθμός των κατόχων δεδομένων είναι μεγαλύτερος του 2 το πρωτόκολλο θα πρέπει να επαναληφθεί για όλα τα ζευγάρια των κατόχων.

4. 5 Κατασκευή πινάκων ανομοιότητας

Κάθε κάτοχος δεδομένων κατασκευάζει τον τοπικό πίνακα ανομοιότητας για το σύνολο των κινούμενων αντικειμένων που του ανήκει. Σε αυτήν την περίπτωση όπως αναφέραμε δεν είναι απαραίτητο να χρησιμοποιήσουμε κάποια από τις μεθόδους διατήρησης της ιδιωτικότητας, αφού οι κάτοχοι δεν ανταλλάσουν δεδομένα και άρα δεν υπάρχει απώλεια πληροφορίας.
Παρακάτω φαίνεται ο ψευδό-κώδικας για την κατασκευή του τοπικού πίνακα ανομοιότητας από τον κάτοχο δεδομένων Α

INPUT : K^A , συνάρτηση σύγκρισης dist()

OUTPUT: τοπικός πίνακας ανομοιότητας $ΤΠ^A_{αν}$

Begin

1.Αρχικοποίηση πίνακα $ΤΠ^A_{αν}$ = { K^A.length x K^A.length }

for (i=0 **to** K^A.length)

  for (j=0 **to** K^A.length)

    2. $ΤΠ^A_{αν}[i][j] = dist(\kappa_i, \kappa_j)$

End

Ψευδό-κώδικας 10 Κατασκευή τοπικού πίνακα ανομοιότητας

Η κατασκευή του καθολικού πίνακα ανομοιότητας για ένα σύνολο n κατόχων δεδομένων γίνεται ως εξής: Κάθε κάτοχος δεδομένων κατασκευάζει τον τοπικό του πίνακα ανομοιότητας $ΤΠ^i_{αν}, i = 1, 2....n$ όπως δείξαμε παραπάνω και τον στέλνει στο ΤΡ. Στην συνέχεια για κάθε ζεύγος κατόχων εφαρμόζεται το πρωτόκολλο για την ασφαλή ανταλλαγή των δεδομένων και την εξαγωγή των αποτελεσμάτων. Έτσι το ΤΡ έχοντας τους τοπικούς πίνακες και τον πίνακα με τις αποστάσεις των αντικειμένων που ανήκουν σε διαφορετικούς κατόχους κατασκευάζει τον καθολικό πίνακα ανομοιότητας.
Ο ψευδό-κώδικας για την κατασκευή του καθολικού πίνακα ανομοιότητας φαίνεται παρακάτω

INPUT : σύνολο κατόχων δεδομένων $DH = \{DH_1, DH_2, ..., DH_n\}$

OUTPUT: καθολικός πίνακας ανομοιότητας $Π_{αν}$

Begin

for (i=0 **to** n)

  1. Ζήτα τους τοπικούς πίνακες ανομοιότητας $ΤΠ^i_{αν}$

  for (j=i+1 **to** n)

    2. Τρέξε το πρωτόκολλο μεταξύ DH_i, DH_j

3.Κατασκεύασε πίνακα ανομοιότητας $Π_{αν}$

End

Ψευδό-κώδικας 11 Κατασκευή καθολικού πίνακα ανομοιότητας

Ο καθολικός πίνακας ανομοιότητας μπορεί πλέον να χρησιμοποιηθεί για την συσταδοποίηση των τροχιών των αντικειμένων ως είσοδος σε οποιονδήποτε αλγόριθμο συσταδοποίησης.

4. 6 Αλγόριθμοι Συσταδοποίησης

Για την εξαγωγή των αποτελεσμάτων χρησιμοποιήσαμε τους παρακάτω αλγόριθμους συσταδοποίησης, οι οποίοι δέχονται σαν είσοδο τον πίνακα ανομοιότητας.

4.6. 1 Ιεραρχικός συσωρευτικός αλγόριθμος με το κριτήριο ελάχιστης διακύμανσης

Οι ιεραρχικοί αλγόριθμοι δέχονται σαν είσοδο τον πίνακα ανομοιότητας, επομένως ο συγκεκριμένος συσωρευτικός αλγόριθμος ήταν μια επιλογή για την εξαγωγή των αποτελεσμάτων. Αρχικά ο αλγόριθμος χωρίζει το σύνολο των ν αντικειμένων σε ν συστάδες, δηλαδή κάθε αντικείμενο και μία συστάδα και στην συνέχεια με το κριτήριο της ελάχιστης διακύμανσης προχωράει σταδιακά μέχρι όλα τα αντικείμενα να ανήκουν σε μία συστάδα.

4.6. 2 Αλγόριθμος CLARANS

Ο αλγόριθμος συσταδοποίησης CLARANS που υλοποιούμε προτείνεται στο [25].Ο συγκεκριμένος αλγόριθμος αποτελεί μια βελτιωμένη έκδοση του αλγορίθμου PAM, διότι δεν υπολογίζει όλες τις πιθανές συσταδοποιήσεις, παρά μόνο ένα μικρό αριθμό και οι οποίες καθορίζονται τυχαία. Επομένως είναι πιο αποτελεσματικός για μεγάλες βάσεις δεδομένων. Παρακάτω παραθέτουμε τον ψευδό-κώδικα για τον αλγόριθμο CLARANS ο οποίος δέχεται σαν είσοδο τον αριθμό των συστάδων k, τον πίνακα ανομοιότητας και το σύνολο n αντικειμένων. Επιπλέον πρέπει να καθορίσουμε και τις παραμέτρους numlocal και maxneighbor. Η πρώτη παράμετρος αφορά τον αριθμό των βέλτιστων τοπικών συσταδοποιήσεων που πρέπει να γίνουν και η οποία έχει προταθεί ίση με 2 και η δεύτερη παράμετρος αφορά τον αριθμό των ανταλλαγών μέσων (medoids) και μη-μέσων και η οποία έχει μέγιστη τιμή μεταξύ 1.25 % *k(n-k) και 250.
Ο μέσος κάθε συστάδας ορίζεται ως εξής
$\text{medoid}(o) = m_i,\ m_i \in M,\ \forall\ m_j \in M: \text{dist}(o, m_i) \leq \text{dist}(o, m_j)$
όπου M το σύνολο των μέσων και ο αντικείμενο από το σύνολο των αντικειμένων K

INPUT : αριθμός συστάδων k,$\Pi_{\alpha\nu}$, numlocal ,maxneighbor,σύνολο n αντικειμένων K
OUTPUT: cluster_set
Begin
for (i=1 **to** numlocal)
1. Δημιούργησε ένα τυχαίο αρχικό σύνολο από k μέσους
do
2.Διάλεξε τυχαία ένα μέσο από το σύνολο των k μέσων
και ένα μη-μέσο από το σύνολο των n-k μη-μέσων
3.Υπολόγισε την διαφορά της μέσης απόστασης(average_dist),η οποία προκύπτει

```
      από την εναλλαγή των δύο παραπάνω αντικειμένων του βήματος 2.
    if( average_dist < 0)
      4.Κάνε τον μέσο μη-μέσο και τον μη-μέσο μέσο.
  while (maxneighbor)
  5.Υπολόγισε την μέση απόσταση της τρέχουσας συσταδοποίησης (current_dist)
  if(current_dist < best_distance)
    6.Θέσε την τρέχουσα συσταδοποίηση σαν την καλύτερη συσταδοποίηση
end for
End
```

Ψευδό-κώδικας 12 Αλγόριθμος CLARANS

Η μέση απόσταση μίας συσταδοποίησης δίνεται από τον παρακάτω τύπο

$$average\_dist = \sum_{m_i \in M} \sum_{o \in cluster(m_i)} dist(o, m_i) / n$$

Κεφάλαιο 5

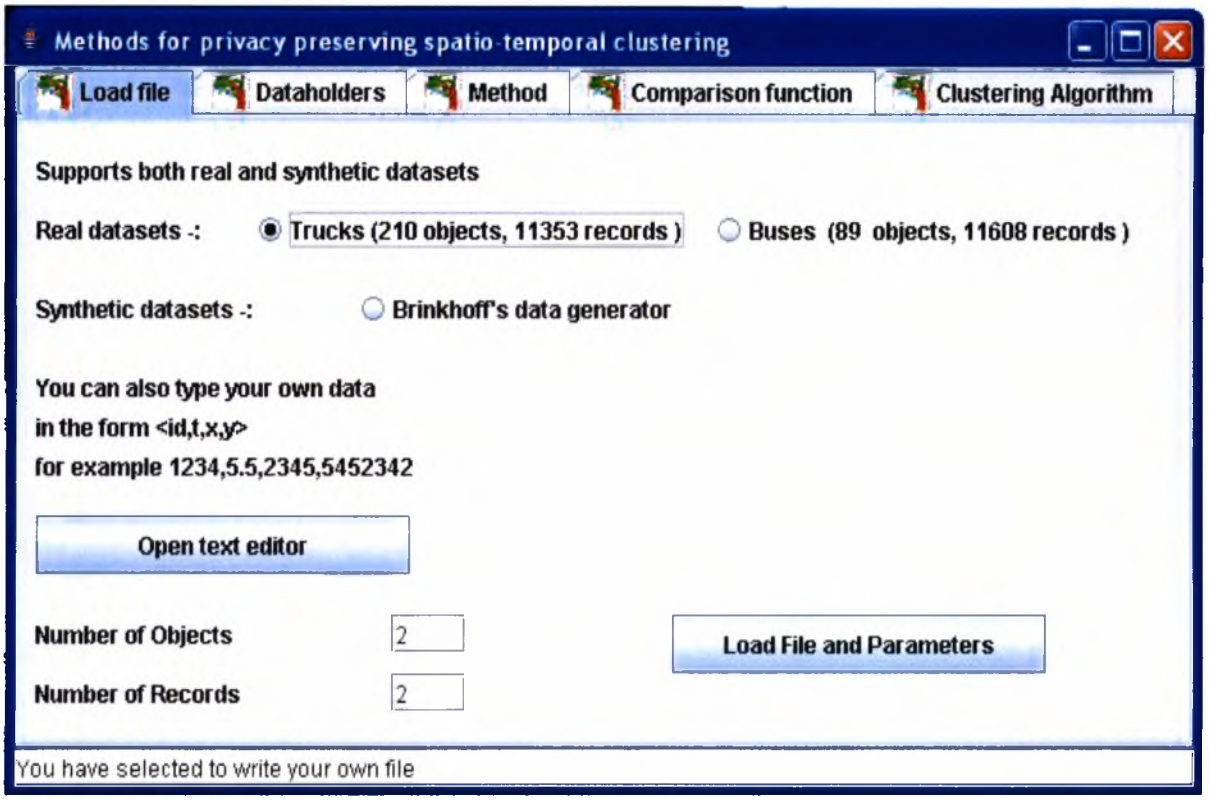
Σύστημα

5. 1 Εισαγωγή

Σε αυτό το κεφάλαιο περιγράφουμε τον τρόπο λειτουργίας της εργαλειοθήκης που έχουμε κατασκευάσει καθώς και την αρχιτεκτονική στην οποία έχουμε βασιστεί. Η παρουσίαση γίνεται με εικόνες, όπου επεξηγούμε σε κάθε βήμα τις κινήσεις που πρέπει να κάνει ένας απλός χρήστης. Η εργαλειοθήκη είναι μια απλή και εύχρηστη εφαρμογή και η οποία περιέχει όλα όσα έχουμε αναφέρει στα προηγούμενα κεφάλαια. Όπως θα δείξουμε και στην ενότητα 5.2 η εργαλειοθήκη είναι χωρισμένη σε καρτέλες με τις οποίες ο χρήστης μπορεί να επιλέξει το αρχείο που θα χρησιμοποιήσει, τον αριθμό των κατόχων δεδομένων, την μέθοδο υλοποίησης, την συνάρτηση σύγκρισης καθώς και τον αλγόριθμο συσταδοποίησης.

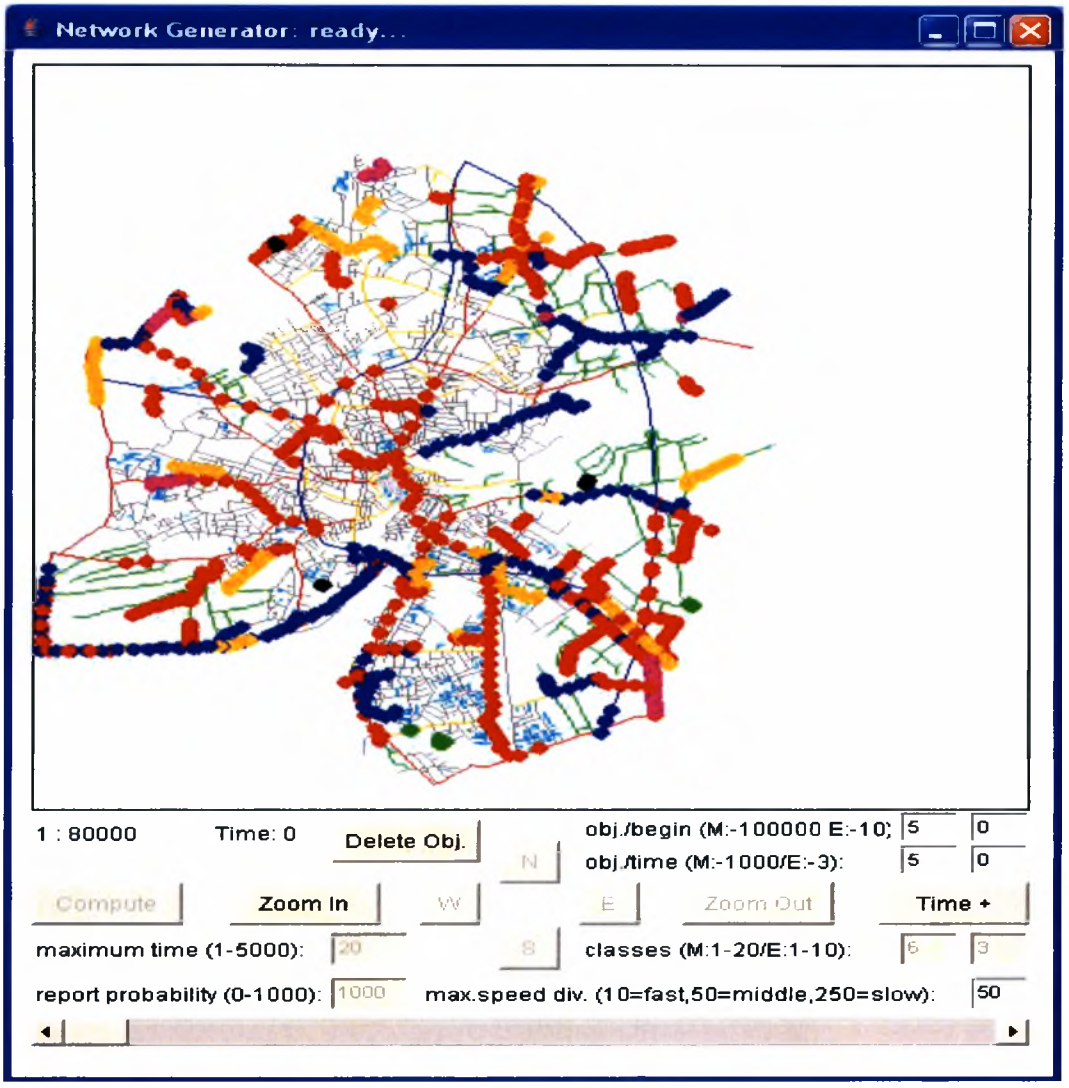
5. 2 Εργαλειοθήκη

Η αρχική καρτέλα «Load File» φαίνεται στην εικόνα 12, όπου μπορούμε να επιλέξουμε το επιθυμητό αρχείο με τις τροχιές των κινούμενων αντικειμένων. Οι τύποι των δεδομένων που υποστηρίζονται είναι τόσο πραγματικά όσο και συνθετικά δεδομένα. Στην πρώτη περίπτωση ο χρήστης μπορεί να επιλέξει είτε το αρχείο trucks, το οποίο περιέχει πραγματικές τροχιές αυτοκινήτων και πιο συγκεκριμένα 210 αυτοκινήτων με 11353 εγγραφές, είτε το αρχείο buses με τροχιές 89 λεωφορείων με 11608 εγγραφές. Τα παραπάνω αρχεία είναι από την ιστοσελίδα [26] η οποία περιέχει συλλογές πραγματικών χωρικών και χώρο-χρονικών δεδομένων.



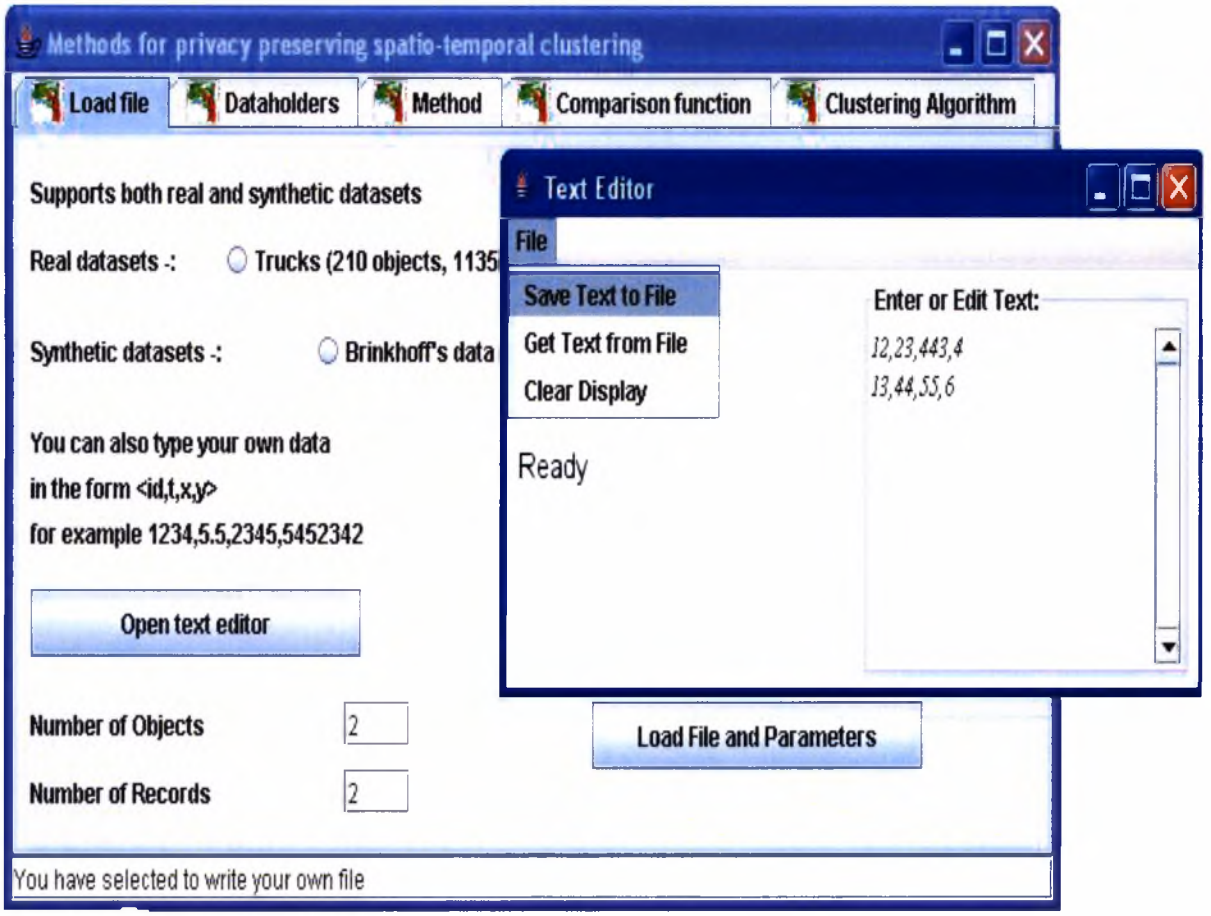
Εικόνα 12 Εργαλειοθήκη 1ης καρτέλα

Στην δεύτερη περίπτωση, δηλαδή των συνθετικών δεδομένων, ο χρήστης μπορεί να δημιουργήσει με την βοήθεια της γεννήτριας δεδομένων του Brinkhoff το δικό του αρχείο, με τον επιθυμητό αριθμό αντικειμένων και εγγραφών. Στην εικόνα 13 φαίνεται η γεννήτρια δεδομένων όπου έχουμε δημιουργήσει ένα αρχικό σύνολο 5 κινούμενων αντικειμένων και όπου κάθε χρονικό διάστημα προστίθενται 5 αντικείμενα. Στην εικόνα 13 βλέπουμε την τελική αναπαράσταση των αντικειμένων μετά από 20 χρονικά διαστήματα.



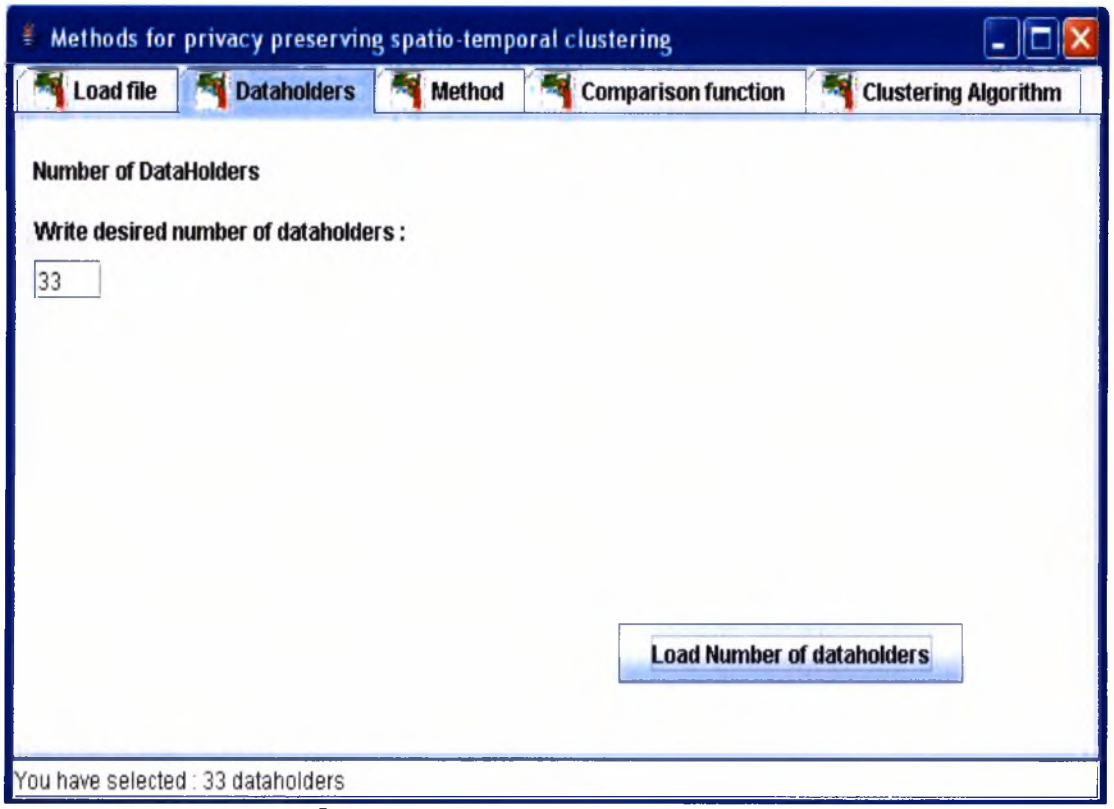
Εικόνα 13 Γεννήτρια δεδομένων Brinkhoff

Επιπλέον ο χρήστης μπορεί να γράψει το δικό του αρχείο δεδομένων. Πατώντας το κουμπί «Open text Editor», όπως διακρίνεται και στην εικόνα 14, εμφανίζεται στην οθόνη ο κειμενογράφος, όπου ο χρήστης γράφει τα δεδομένα του στη μορφή <id , t x ,y> και αποθηκεύει το αρχείο. Στην συνέχεια εισάγει τον αριθμό των αντικειμένων και των εγγραφών που έχει γράψει και φορτώνει το αρχείο που έχει κατασκευάσει πατώντας το κουμπί «Load File and Parameters».



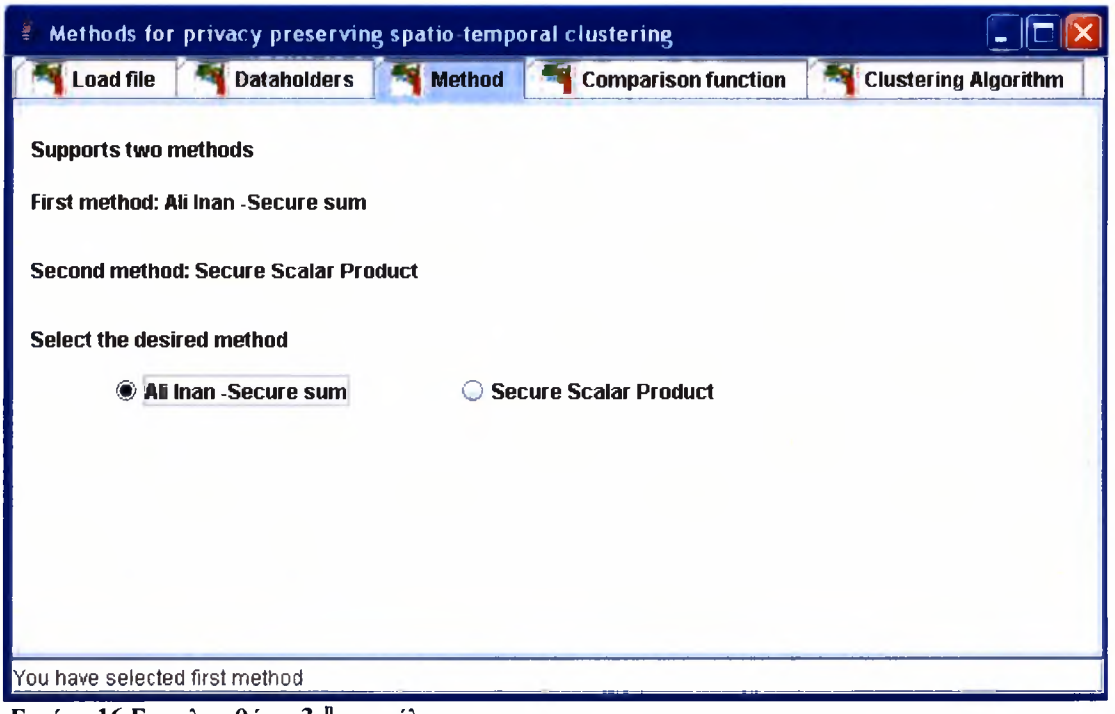
Εικόνα 14 Επιλογή κειμενογράφου

Στην δεύτερη καρτέλα «Dataholders», ο χρήστης εισάγει τον αριθμό των κατόχων δεδομένων στο πλαίσιο όπως φαίνεται και στην εικόνα 15 και εν συνεχεία φορτώνει τον αριθμό των κατόχων, πατώντας το κουμπί «Load Number of dataholders»



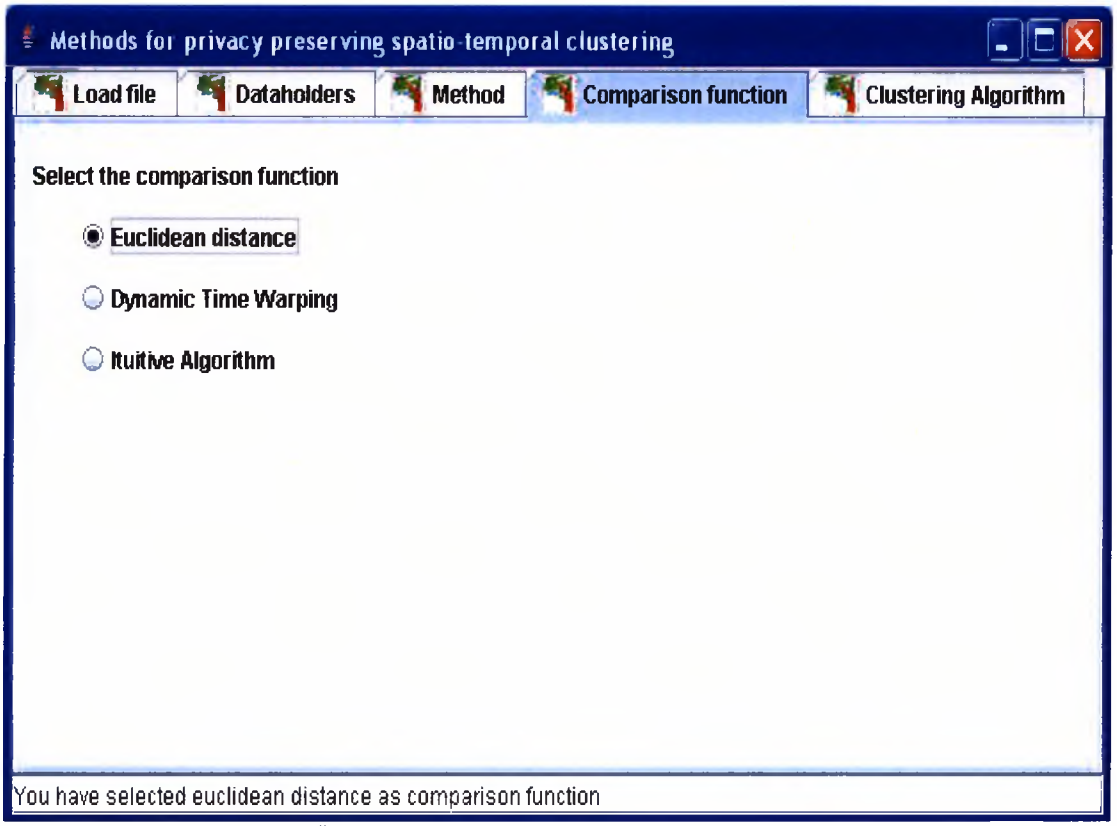
Εικόνα 15 Εργαλειοθήκη 2ης καρτέλα

Στην τρίτη κατά σειρά καρτέλα «Method», ο χρήστης επιλέγει μία από τις δύο μεθόδους διατήρησης της ιδιωτικότητας που αναφέραμε στο κεφάλαιο 4. Η τρίτη καρτέλα φαίνεται παρακάτω στην εικόνα 16.



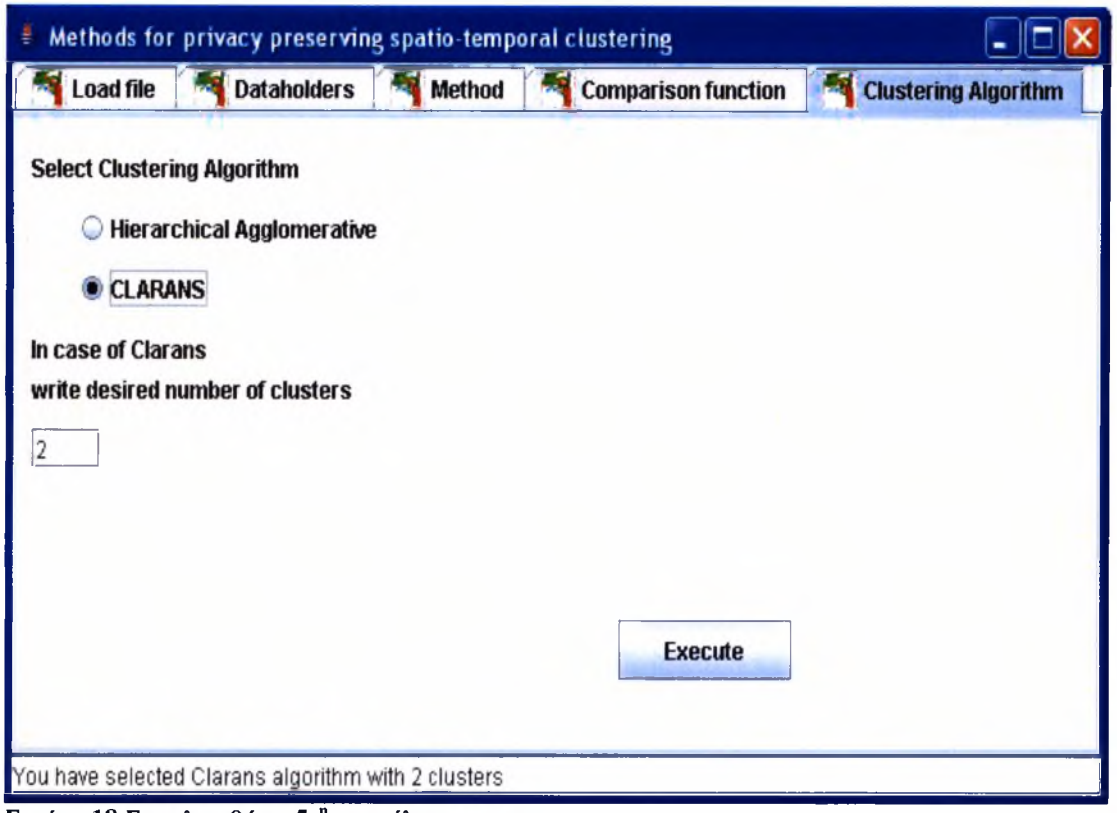
Εικόνα 16 Εργαλειοθήκη 3ης καρτέλα

Στην τέταρτη καρτέλα «Comparison function», ο χρήστης επιλέγει την συνάρτηση σύγκρισης με την οποία επιθυμεί να υπολογίσει την ανομοιότητα των τροχιών. Οι διαθέσιμες συναρτήσεις σύγκρισης είναι οι τρεις που αναφέραμε και στο κεφάλαιο 3, δηλαδή η ευκλείδεια απόσταση, ο αλγόριθμος δυναμικής χρονικής παραμόρφωσης και ο επαναληπτικός αλγόριθμος.
Στην παρακάτω εικόνα 17 φαίνεται η τέταρτη καρτέλα.



Εικόνα 17 Εργαλειοθήκη 4_η καρτέλα

Τέλος στην πέμπτη κατά σειρά και τελευταία καρτέλα, ο χρήστης επιλέγει τον αλγόριθμο συσταδοποίησης. Οι δύο διαθέσιμοι αλγόριθμοι τους οποίους περιγράψαμε και στην ενότητα 4.6 είναι ο ιεραρχικός συσωρευτικός αλγόριθμος με το κριτήριο της ελάχιστης διακύμανσης και ο αλγόριθμος CLARANS. Στην περίπτωση που ο χρήστης θα επιλέξει τον αλγόριθμο CLARANS, θα πρέπει να εισάγει στο πλαίσιο, όπως διακρίνεται και στην εικόνα 18, τον επιθυμητό αριθμό των συστάδων.
Στην συνέχεια πατώντας το κουμπί «Execute», φορτώνει όλα τα δεδομένα που έχει επιλέξει και εκτελείται το πρόγραμμα.



Εικόνα 18 Εργαλειοθήκη 5_η καρτέλα

5. 3 Αρχιτεκτονική

Ο κώδικας της εργαλειοθήκης είναι γραμμένος σε Java και έχουν χρησιμοποιηθεί οι εξής βιβλιοθήκες.

5.3. 1 Βιβλιοθήκες

Για την έξοδο των αποτελεσμάτων δηλαδή για την απεικόνιση των συστάδων χρησιμοποιήσαμε την βιβλιοθήκη Visad [27]. Η Visad είναι μία Java βιβλιοθήκη για την αναπαράσταση και ανάλυση αριθμητικών δεδομένων. Η δυνατότητα της βιβλιοθήκης να υποστηρίζει τρισδιάστατα δεδομένα, φάνηκε αρκετά χρήσιμη για την γραφική αναπαράσταση των τροχιών των κινούμενων αντικειμένων, όπως επίσης και για την συσταδοποίηση τους. Επίσης για την έξοδο και συγκεκριμένα για την περίπτωση του ιεραρχικού αλγορίθμου, όπου απαιτείται η απεικόνιση των συστάδων σε μορφή δενδρογράμματος, χρησιμοποιήσαμε την βιβλιοθήκη Ddraw [28].
Η συγκεκριμένη Java βιβλιοθήκη δέχεται σαν είσοδο ένα αρχείο με τα αποτελέσματα του ιεραρχικού αλγορίθμου κατάλληλα τροποποιημένο και εξάγει τα αποτελέσματα σε μορφή δενδρογράμματος.
Τέλος για την κατασκευή των γράφων των πειραμάτων όπως επίσης και την αναπαράσταση των τροχιών σε δυσδιάστατο επίπεδο, χωρίς να λαμβάνεται υπόψη ο χρόνος, χρησιμοποιήσαμε την Java βιβλιοθήκη jahuwaldt [29].

.

Κεφάλαιο 6

Πειράματα

6. 1 Εισαγωγή

Στο κεφάλαιο αυτό παραθέτουμε τα πειράματα που έχουν γίνει με βάση όλα όσα αναφέραμε στα προηγούμενα κεφάλαια. Οι μετρήσεις και τα πειράματα που έχουν γίνει αφορούν τόσο συνθετικά όσο και πραγματικά δεδομένα. Επίσης τρεις είναι οι παράμετροι που θα μας απασχολήσουν σε αυτό το κεφάλαιο σε ότι αφορά τα πειράματα και αυτές είναι ο αριθμός των κατόχων δεδομένων, ο αριθμός των αντικειμένων του συνόλου της βάσης δεδομένων καθώς και ο αριθμός των παρατηρήσεων. Στις ενότητες που ακολουθούν συγκρίνουμε τις δύο μεθόδους διατήρησης της ιδιωτικότητας και τις τρεις συναρτήσεις σύγκρισης ως προς το υπολογιστικό κόστος, καθώς επίσης αξιολογούμε την αποτελεσματικότητα των αλγορίθμων συσταδοποίησης όπως και των συναρτήσεων σύγκρισης.

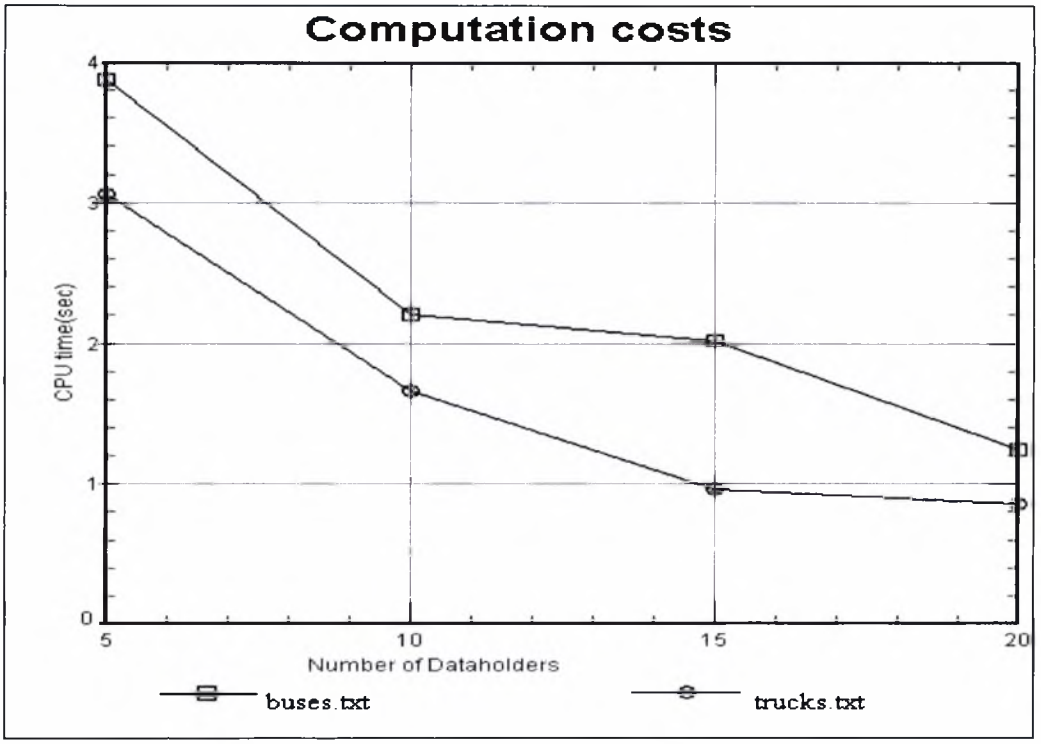
6. 2 Ανάλυση υπολογιστικού κόστους

Τα πειράματα για τα πραγματικά δεδομένα έγιναν σε επεξεργαστή AMD Athlon(tm) 64 με ταχύτητα 2.2 GHz και μνήμη RAM 512 MB ενώ για τα συνθετικά δεδομένα σε επεξεργαστή Intel(r) Core(TM)2 με ταχύτητα 2.16 GHz και μνήμη RAM 2 GB.

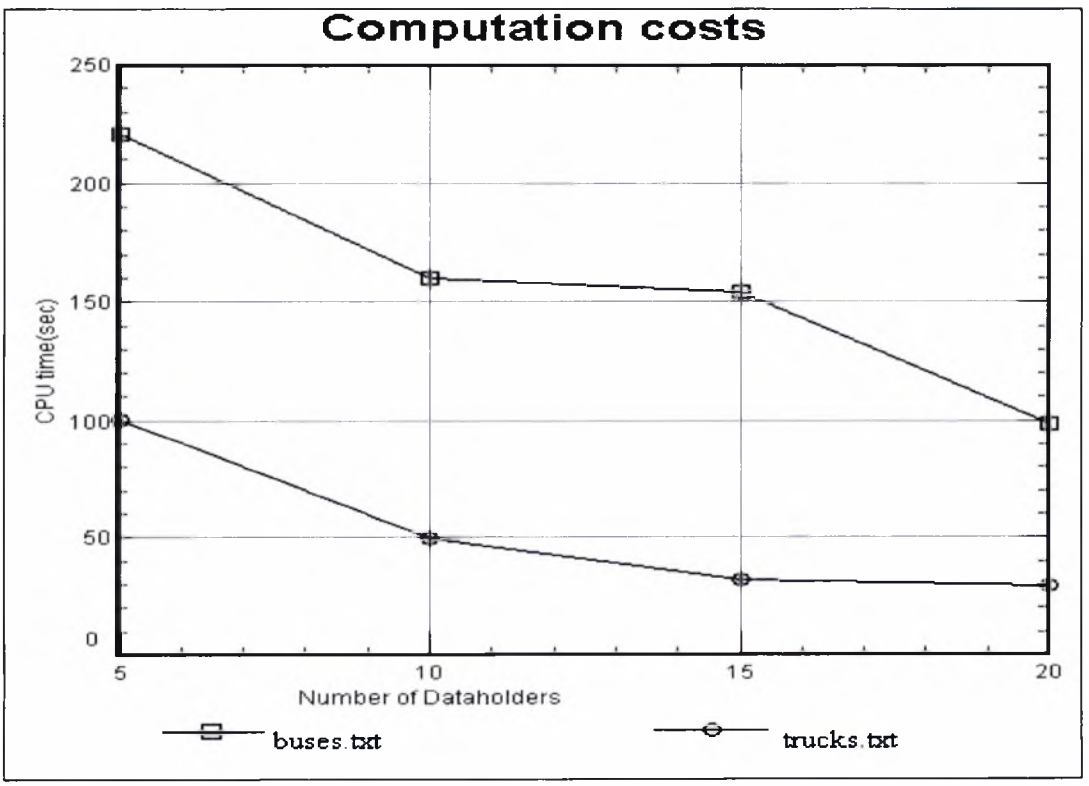
6.2. 1 Πραγματικά δεδομένα

Αρχικά αναλύουμε το υπολογιστικό κόστος σε πραγματικά δεδομένα και πιο συγκεκριμένα στα υπάρχοντα αρχεία της εργαλειοθήκης (trucks,buses). Όπως αναφέραμε και στο κεφάλαιο 5 τα δεδομένα προέρχονται από μετρήσεις πραγματικών τροχιών αυτοκινήτων και σχολικών λεωφορείων αντίστοιχα [26]. Το πρώτο αρχείο trucks περιέχει τροχιές 210 αυτοκινήτων με συνολικά 11353 παρατηρήσεις ενώ το αρχείο buses τροχιές 89 λεωφορείων με 11608 συνολικά παρατηρήσεις. Σε κάθε αρχείο η πληροφορία που καταγράφεται για κάθε αντικείμενο είναι η ταυτότητα του αντικειμένου, ο χρόνος και η θέση του αντικειμένου η οποία περιγράφεται από δύο διαστάσεις το γεωγραφικό μήκος και πλάτος.

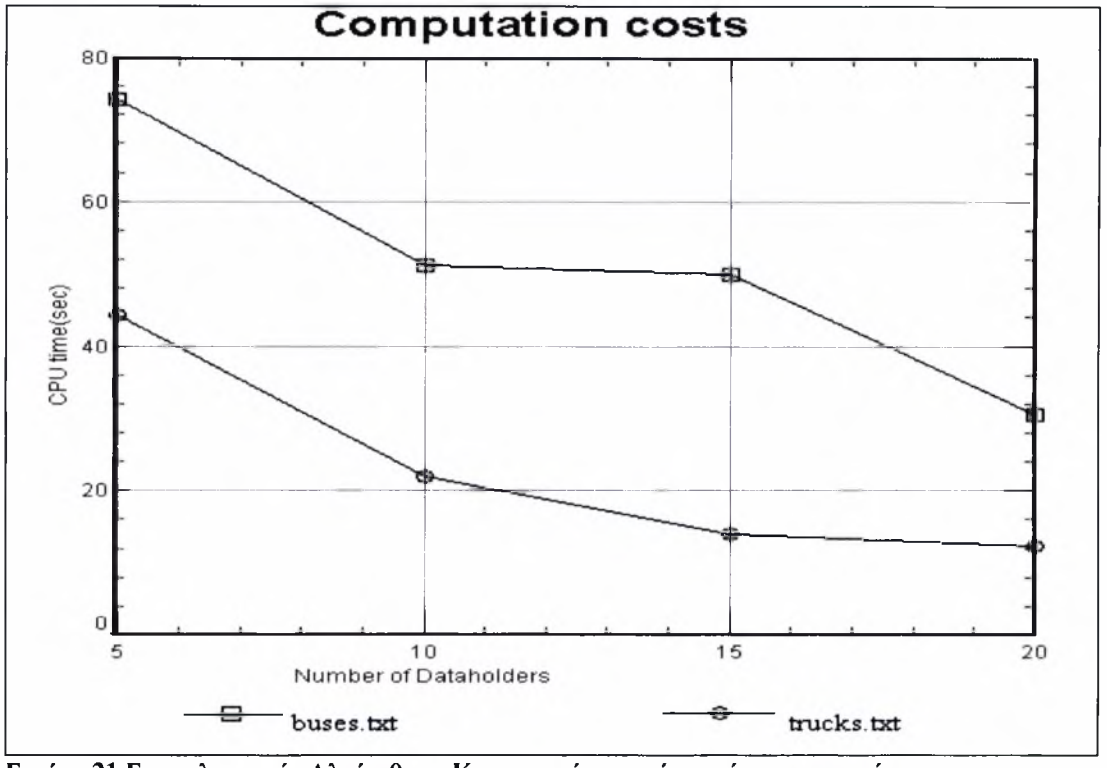
Στην περίπτωση των πραγματικών δεδομένων η μόνη παράμετρος που μπορούμε να χρησιμοποιήσουμε και να τροποποιήσουμε είναι ο αριθμός των κατόχων δεδομένων. Στις παρακάτω εικόνες παραθέτουμε το υπολογιστικό κόστος για την κατασκευή των τοπικών πινάκων ανομοιότητας. Σε κάθε γραφική παράσταση συγκρίνουμε τους χρόνους για τα δύο υπάρχοντα αρχεία πραγματικών δεδομένων. Η διαφορά των δύο αρχείων που είναι και ιδιαίτερα σημαντική είναι ο αριθμός των παρατηρήσεων ανά αντικείμενό.



Εικόνα 19 Ευκλείδεια απόσταση-Κατασκευή τοπικών πινάκων ανομοιότητας



Εικόνα 20 DTW-Κατασκευή τοπικών πινάκων ανομοιότητας



Εικόνα 21 Επαναληπτικός Αλγόριθμος-Κατασκευή τοπικών πινάκων ανομοιότητας

Η ανάθεση των αντικειμένων στους κατόχους δεδομένων προκύπτει διαιρώντας τον αριθμό των αντικειμένων με τον αριθμό των κατόχων. Αν το υπόλοιπο της διαίρεσης είναι διάφορο του μηδενός τότε το υπόλοιπο προστίθεται στον αριθμό των αντικειμένων του πρώτου κατόχου δεδομένων. Επομένως πρέπει να λάβουμε υπ'όψιν την παράμετρο αυτή για τον αριθμό των υπολογισμών που απαιτούνται. Όπως αναφέραμε στο κεφάλαιο 4 για την κατασκευή του τοπικού πίνακα ανομοιότητας κάθε κάτοχος πρέπει να υπολογίσει ανά δύο τις αποστάσεις μεταξύ των αντικειμένων του. Για παράδειγμα αν ένας κάτοχος έχει 5 αντικείμενα πρέπει να κάνει (5,2) =10 υπολογισμούς. Στην περίπτωση όμως που οι κάτοχοι έχουν διαφορετικό αριθμό αντικειμένων, όπως αναφέραμε παραπάνω ο αριθμός των υπολογισμών είναι ίσος με τους υπολογισμούς που πρέπει να κάνει ο πρώτος κάτοχος δεδομένων, ο οποίος έχει και τον μεγαλύτερο αριθμό αντικειμένων, αν θεωρήσουμε ότι οι υπολογισμοί των τοπικών πινάκων γίνονται παράλληλα.
Στους παρακάτω πίνακες 4,5 παραθέτουμε τους υπολογισμούς που πρέπει να γίνουν για τα δύο αρχεία.

Κάτοχοι δεδομένων	**5**	**10**	**15**	**20**
Μέγιστος Αριθμός Αντικειμένων	42	21	14	20
Υπολογισμοί	861	210	91	190

Πίνακας 4 Αριθμός υπολογισμών για το αρχείο trucks

Κάτοχοι δεδομένων	**5**	**10**	**15**	**20**
Μέγιστος Αριθμός Αντικειμένων	21	17	19	13
Υπολογισμοί	210	136	171	78

Πίνακας 5 Αριθμός υπολογισμών για το αρχείο buses

Όπως γίνεται αντιληπτό από τις εικόνες 19,20,21 το υπολογιστικό κόστος μειώνεται και στις τρεις συναρτήσεις σύγκρισης όσο αυξάνεται ο αριθμός των κατόχων δεδομένων. Η μείωση αυτή όπως δείχνουν και οι πίνακες 4,5 είναι φανερή όταν ο αριθμός των κατόχων αυξάνεται από 5 σε 10, αφού εκτελούνται οι μισοί υπολογισμοί. Επίσης μείωση του υπολογιστικού κόστους για το αρχείο trucks υπάρχει όταν ο αριθμός των κατόχων αυξάνεται σε 15, ενώ για το αρχείο buses οι χρόνοι απόκρισης είναι περίπου στα ίδια για 10 και 15 κατόχους και μειώνεται κατακόρυφα για 20 κατόχους αφού εκτελούνται οι μισοί υπολογισμοί 78 (πίνακας 5). Επίσης ο μεγαλύτερος αριθμός παρατηρήσεων του αρχείου buses ανά αντικείμενο δικαιολογεί κα το μεγαλύτερο υπολογιστικό κόστος.
Στον παρακάτω πίνακα 6 παραθέτουμε για το αρχείο trucks τους χρόνους για την κατασκευή των τοπικών πινάκων ανομοιότητας για κάθε μία από τις τρεις συναρτήσεις σύγκρισης. Όπως μπορούμε να διαπιστώσουμε η ευκλείδεια απόσταση έχει το μικρότερο υπολογιστικό κόστος όπως ήταν αναμενόμενο για τον λόγο ότι οι υπολογισμοί γίνονται για ένα συγκεκριμένο αριθμό παρατηρήσεων κάθε αντικειμένου (ψευδό –κώδικας 1), ενώ το μεγαλύτερο υπολογιστικό κόστος αφορά τον αλγόριθμο δυναμικής χρονικής παραμόρφωσης.

Συνάρτηση Σύγκρισης/Αριθμός κατόχων Δεδομένων	**5**	**10**	**15**	**20**
Ευκλείδεια Απόσταση	3,1 sec	1,6 sec	0,9sec	0,8sec
Αλγόριθμος χρονικής Παραμόρφωσης (DTW)	100,6 sec	49,8 sec	31,9 sec	29,5 sec
Επαναληπτικός Αλγόριθμος	44,3 sec	21,9 sec	13,9 sec	12,3 sec

Πίνακας 6 Χρόνοι απόκρισης συναρτήσεων σύγκρισης για την κατασκευή τοπικών πινάκων ανομοιότητας

Αφού έχουμε πλέον μία ξεκάθαρη εικόνα για το ποια συνάρτηση σύγκρισης έχει το μικρότερο και αντίστοιχα μεγαλύτερο υπολογιστικό κόστος, θα εξετάσουμε παρακάτω τις δύο μεθόδους ως προς τους χρόνους απόκρισης.
Στους πίνακες 7,8 παραθέτουμε το υπολογιστικό κόστος των δύο μεθόδων για την ανταλλαγή των δεδομένων και την κατασκευή του προσωρινού πίνακα, ο οποίος όπως αναφέραμε και στο κεφάλαιο 3, περιέχει τις αποστάσεις μεταξύ όλων των

πραγματικών παρατηρήσεων της τροχιάς ενός αντικειμένου με όλες τις αντίστοιχες παρατηρήσεις ενός αντικειμένου διαφορετικού κατόχου. Για το αρχείο trucks επιλέξαμε οι μετρήσεις να γίνουν για 5,10,15,20 κατόχους δεδομένων, ενώ για το αρχείο buses για 8,11,14,21 κατόχους, ούτως ώστε να υπάρξει μια πιο ορθολογική ανάθεση των αντικειμένων και να μπορέσουμε να βγάλουμε χρήσιμα συμπεράσματα.

Μέθοδος Διατήρησης Ιδιωτικότητας /Αριθμός κατόχων Δεδομένων	**5**	**10**	**15**	**20**
1^{H} Μέθοδος	258,8 sec	56,8 sec	60,3 sec	63,8 sec
2^{H} Μέθοδος	1300,6 sec	801,3 sec	65,7 sec	68,8 sec

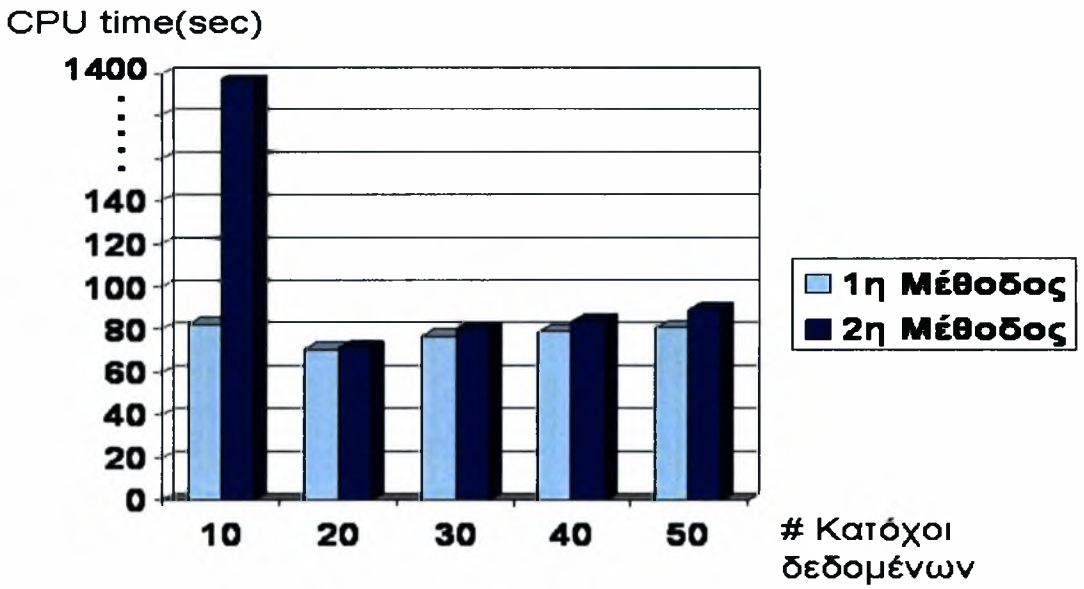
Πίνακας 7 Χρόνοι απόκρισης μεθόδων διατήρησης της ιδιωτικότητας –αρχείο trucks

Όπως μπορούμε να διαπιστώσουμε η πρώτη μέθοδος του ασφαλούς αθροίσματος έχει μικρότερο χρόνο απόκρισης από την δεύτερη μέθοδο. Αυτό γίνεται ιδιαίτερα εμφανές για μικρό αριθμό κατόχων δεδομένων, ενώ όσο αυξάνεται ο αριθμός των κατόχων η διαφορά των χρόνων απόκρισης μικραίνει. Το γεγονός αυτό οφείλεται στην ανάθεση των αντικειμένων καθώς και στον αριθμό των παρατηρήσεων ανά αντικείμενο, ωστόσο με πραγματικά δεδομένα δεν μπορούμε να βγάλουμε ασφαλή συμπεράσματα.

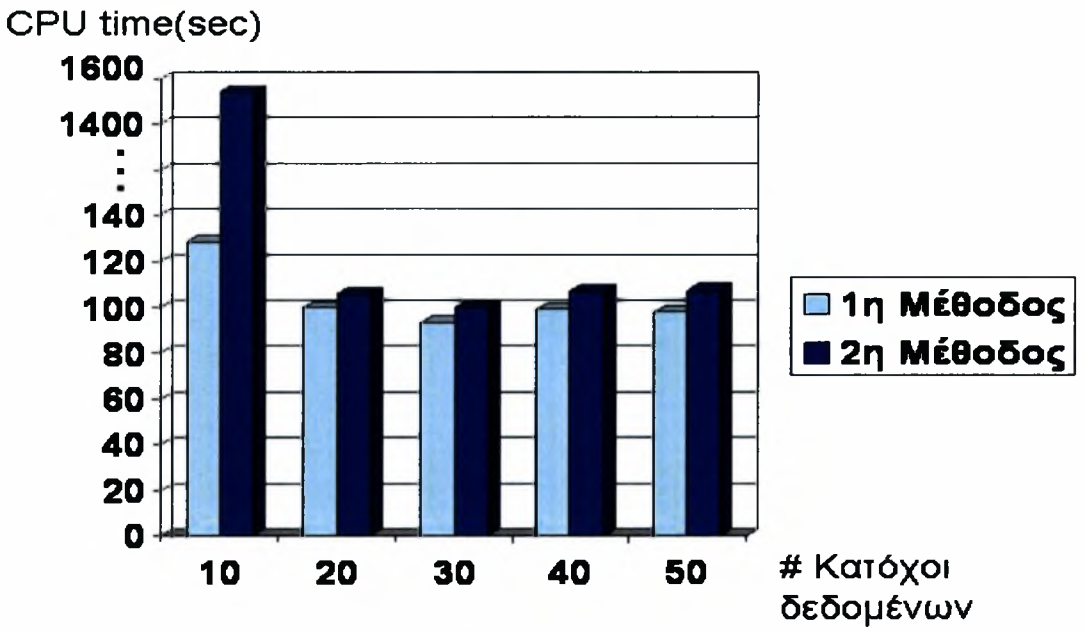
Μέθοδος Διατήρησης Ιδιωτικότητας /Αριθμός κατόχων Δεδομένων	**8**	**11**	**14**	**21**
1^{H} Μέθοδος	91,7 sec	61,3 sec	63,1 sec	66,2 sec
2^{H} Μέθοδος	1147,6 sec	451,8 sec	65,3 sec	72,8 sec

Πίνακας 8 Χρόνοι απόκρισης μεθόδων διατήρησης της ιδιωτικότητας –αρχείο buses

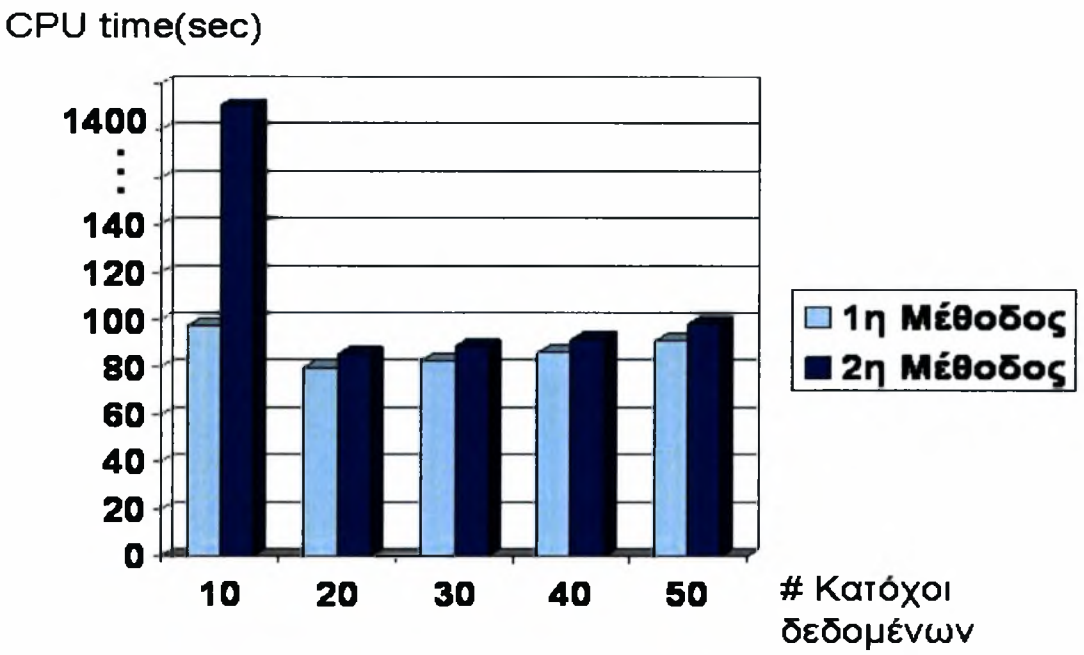
Παρά το γεγονός ότι όσο αυξάνεται ο αριθμός των κατόχων δεδομένων και άρα κάθε κάτοχος δεδομένων έχει μικρότερο αριθμό αντικειμένων, παρατηρούμε ότι ο χρόνος απόκρισης αυξάνεται. Αυτό οφείλεται στο ότι πρέπει να 'τρέξουμε' το πρωτόκολλο περισσότερες φορές και πιο συγκεκριμένα όπως είχαμε αναφέρει στο κεφάλαιο 4 C(n,2) φορές, όπου n ο αριθμός των κατόχων δεδομένων.
Στους γράφους που ακολουθούν εικόνες 22,23,24 παραθέτουμε τους χρόνους απόκρισης των δύο μεθόδων από την αρχή των πρωτοκόλλων μέχρι και την κατασκευή του πίνακα ανομοιότητας. Οι μετρήσεις έχουν γίνει για το αρχείο trucks και για τις τρεις συναρτήσεις σύγκρισης, ευκλείδεια απόσταση (εικόνα 22), αλγόριθμος δυναμικής χρονικής παραμόρφωσης (εικόνα 23), επαναληπτικός αλγόριθμος (εικόνα 24).



Εικόνα 22 Υπολογιστικό κόστος των δύο μεθόδων με την ευκλείδεια απόσταση



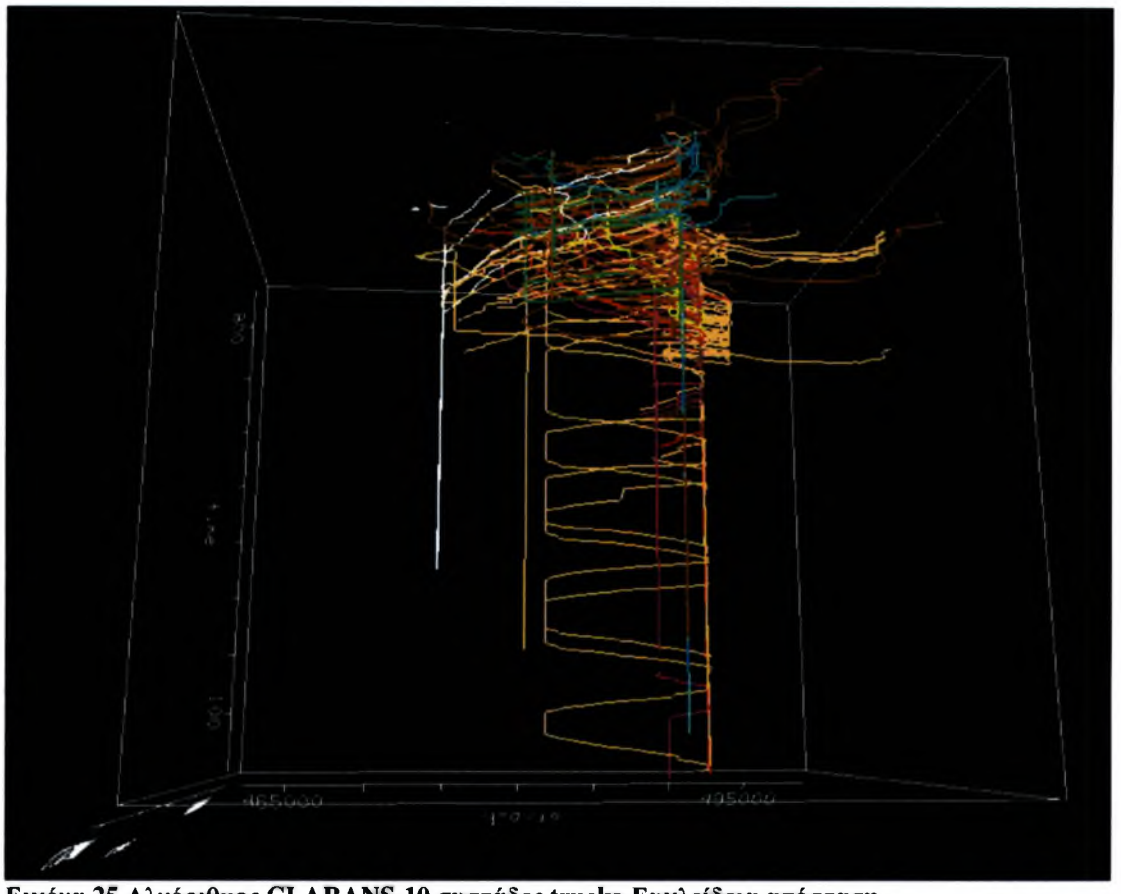
Εικόνα 23 Υπολογιστικό κόστος των δύο μεθόδων με τον αλγόριθμο χρονικής παραμόρφωσης



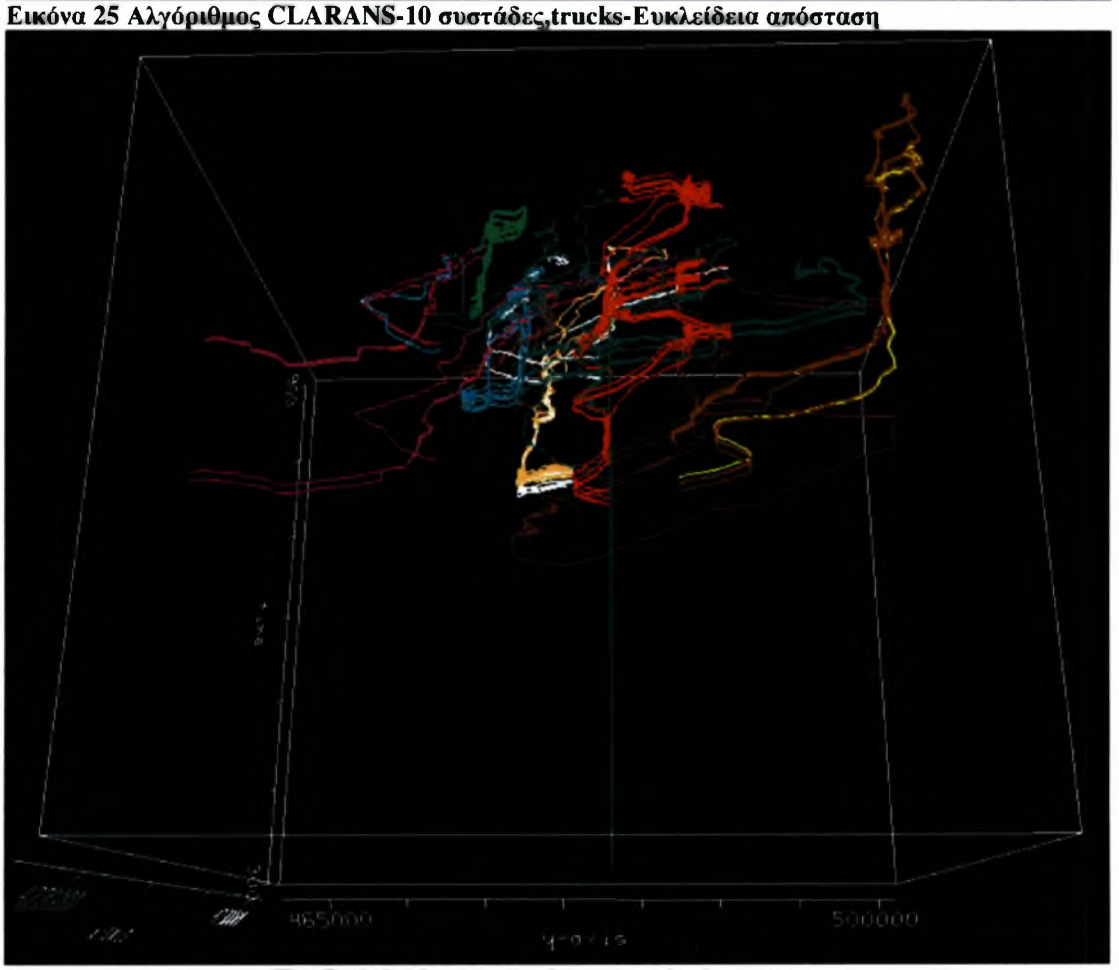
Εικόνα 24 Υπολογιστικό κόστος των δύο μεθόδων με τον επαναληπτικό αλγόριθμο

Οι παραπάνω εικόνες επιβεβαιώνουν το γεγονός ότι η πρώτη μέθοδος έχει μικρότερο υπολογιστικό κόστος, όπως και η ευκλείδεια απόσταση σε σχέση με την δεύτερη μέθοδο και τις υπόλοιπες συναρτήσεις σύγκρισης αντίστοιχα.

Στην παρακάτω εικόνα 25 φαίνεται η έξοδος του προγράμματος για τον αλγόριθμο CLARANS με 10 συστάδες για το αρχείο trucks και την ευκλείδεια απόσταση, ενώ στην εικόνα 26 η έξοδος του προγράμματος για τον αλγόριθμο CLARANS με 10 συστάδες για το αρχείο buses. Κάθε συστάδα απεικονίζεται με διαφορετικό χρώμα, για αυτόν τον λόγο οι τροχιές των αντικειμένων έχουν χρωματιστεί με το χρώμα της συστάδας στην οποία ανήκουν.

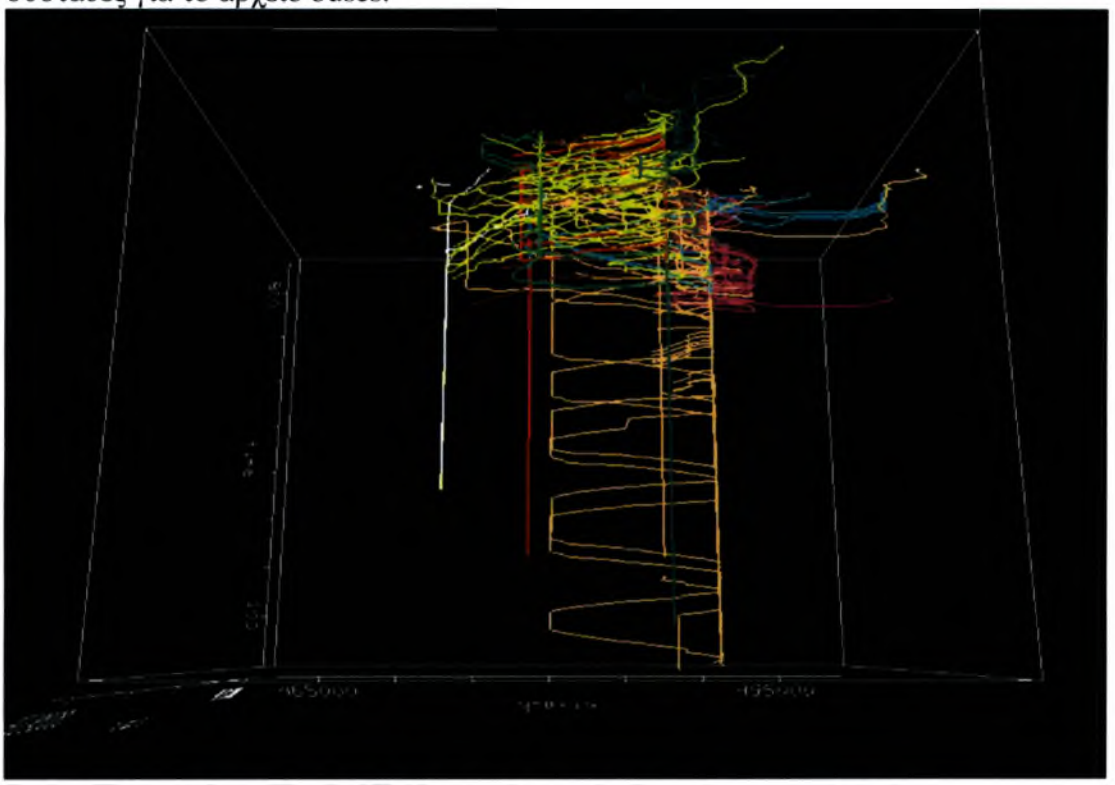


Εικόνα 25 Αλγόριθμος CLARANS-10 συστάδες,trucks-Ευκλείδεια απόσταση

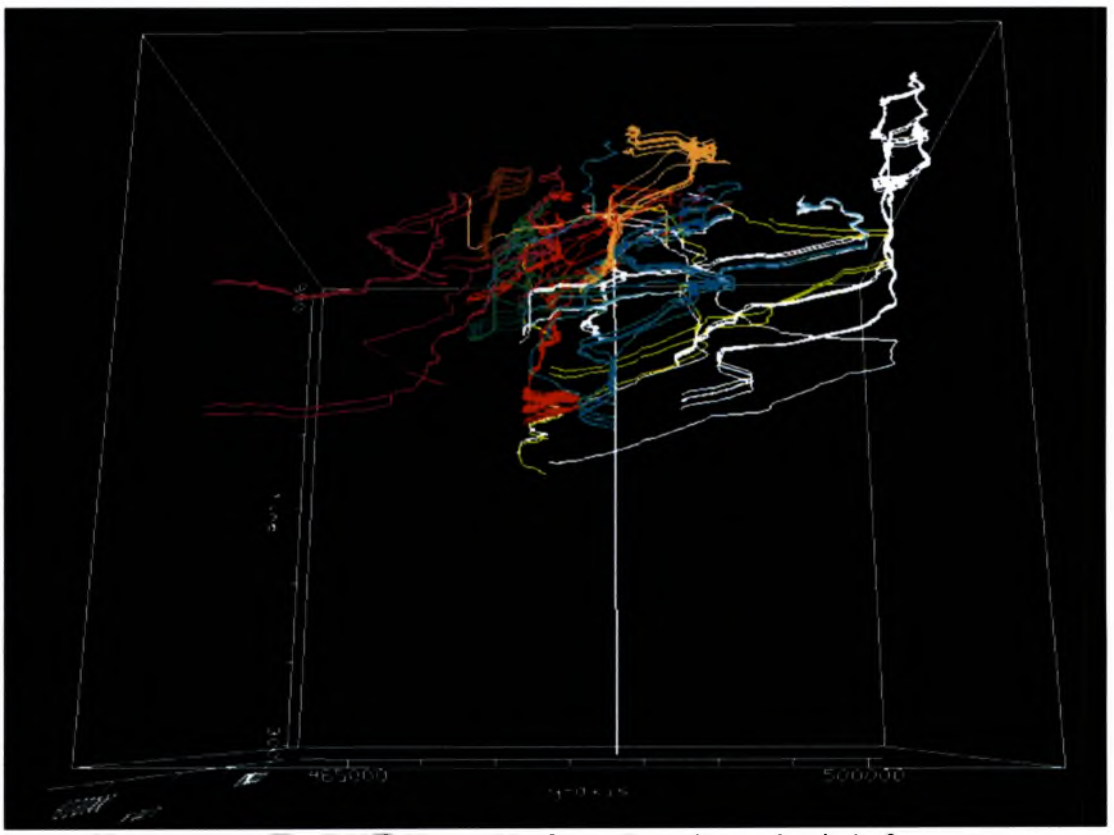


Εικόνα 26 Αλγόριθμος CLARANS-10 συστάδες,buses-Ευκλείδεια απόσταση

Στην παρακάτω εικόνα 25 φαίνεται η έξοδος του προγράμματος για τον αλγόριθμο CLARANS με 10 συστάδες για το αρχείο trucks και επαναληπτικό αλγόριθμο, ενώ στην εικόνα 26 η έξοδος του προγράμματος για τον αλγόριθμο CLARANS με 10 συστάδες για το αρχείο buses.

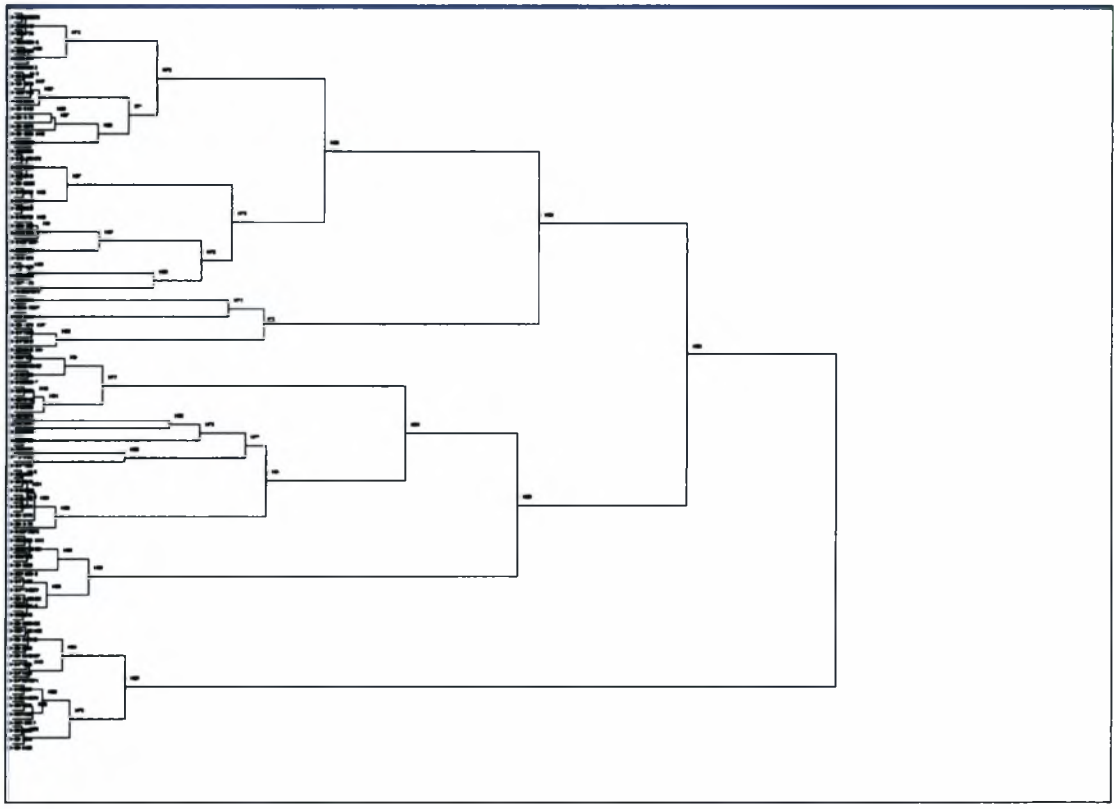


Εικόνα 27 Αλγόριθμος CLARANS-10 συστάδες,trucks-Επαναληπτικός αλγόριθμος



Εικόνα 28 Αλγόριθμος CLARANS-10 συστάδες,buses-Επαναληπτικός Αλγόριθμος

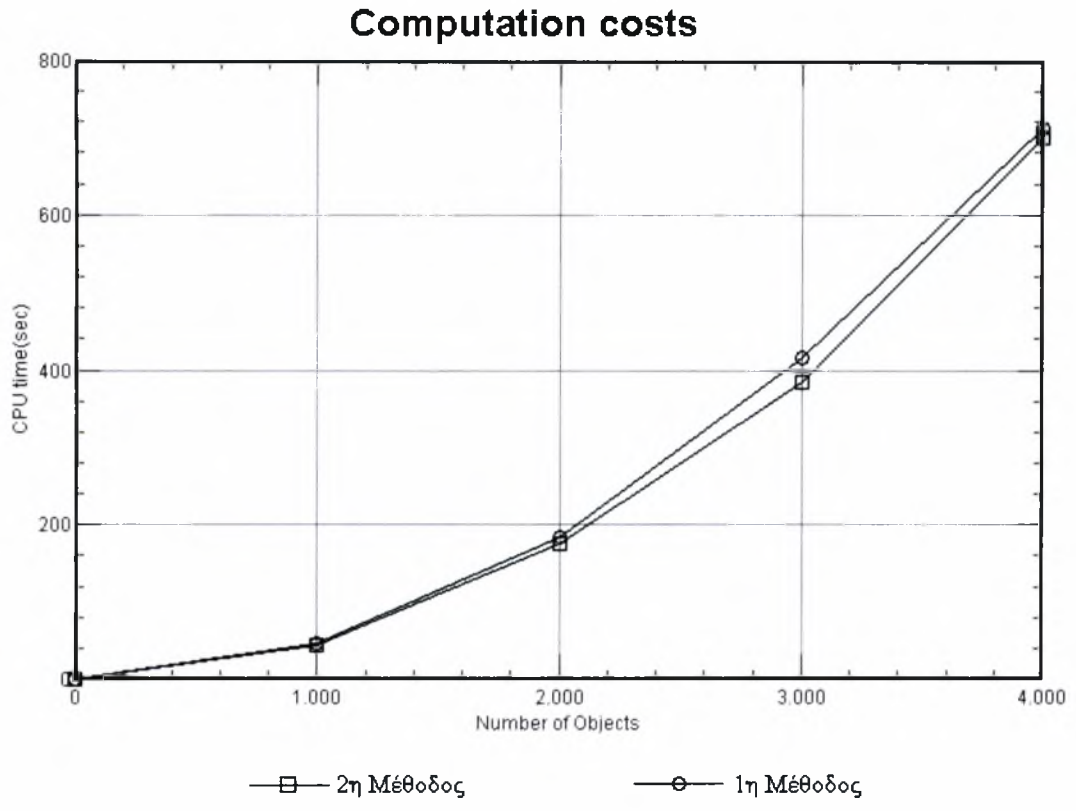
Τέλος στην εικόνα 29 παραθέτουμε την έξοδο του προγράμματος για τον ιεραρχικό αλγόριθμο σε μορφή δενδρογράμματος.



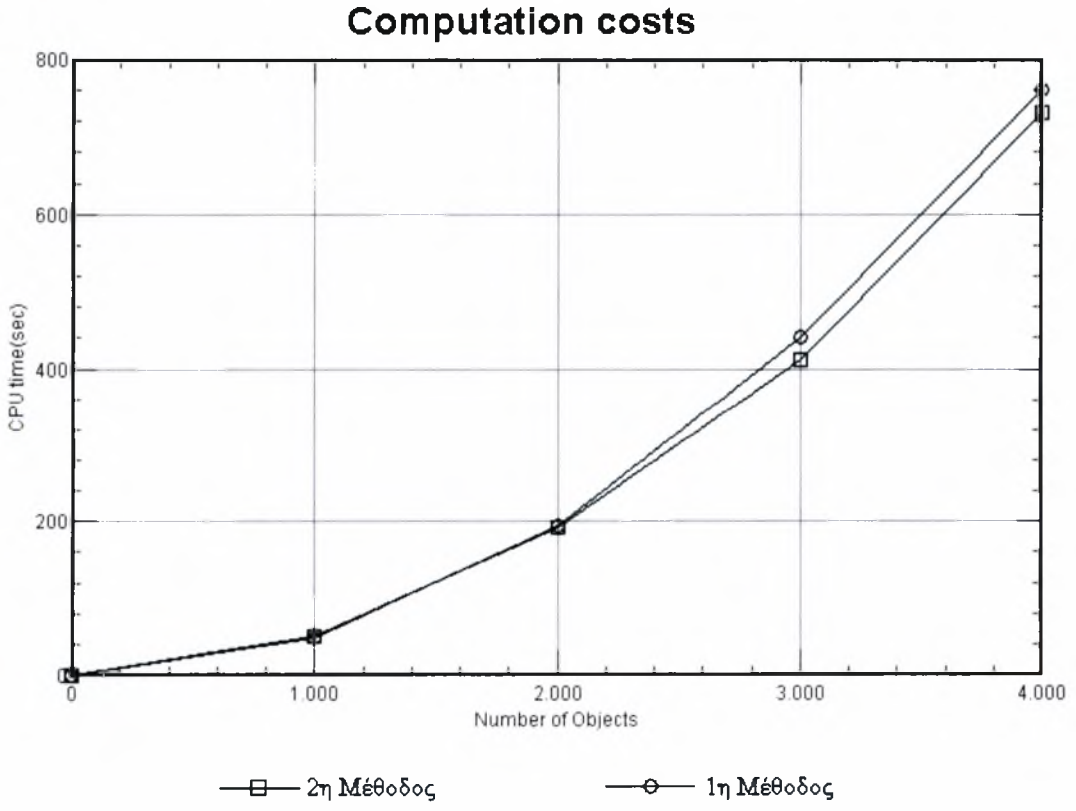
Εικόνα 29 Ιεραρχικός Αλγόριθμος-αρχείο buses, αλγόριθμος δυναμικής παραμόρφωσηςDTW

6.2. 2 Συνθετικά δεδομένα

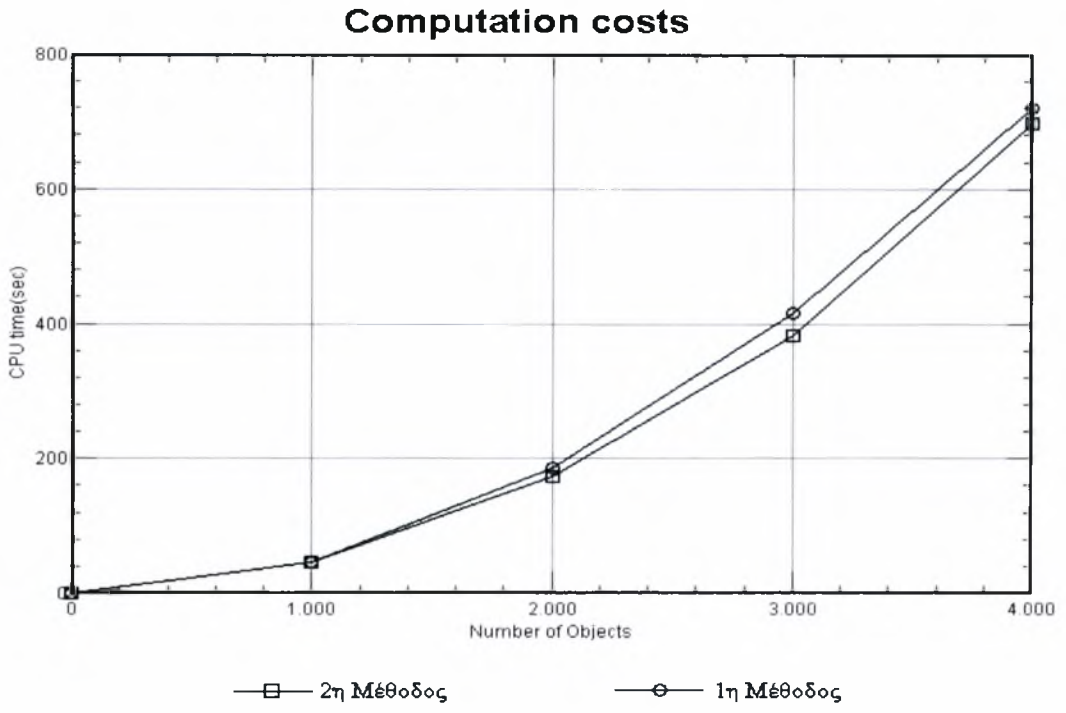
Με την βοήθεια της γεννήτριας δεδομένων του Brinkhoff, έχουμε την δυνατότητα να δημιουργήσουμε τροχιές αντικειμένων με τις παραμέτρους που επιθυμούμε ούτως ώστε να βγάλουμε χρήσιμα συμπεράσματα.
Αρχικά θα μελετήσουμε την συμπεριφορά των δύο μεθόδων, ως προς το υπολογιστικό κόστος στην περίπτωση που μεταβάλλεται ο αριθμός των τροχιών, δηλαδή των αντικειμένων. Στις παρακάτω εικόνες 30,31,32, παρατηρούμε την συμπεριφορά των δύο μεθόδων για τροχιές μήκους 10 παρατηρήσεων. Επίσης έχουμε θέσει τον αριθμό των κατόχων για όλες τις μετρήσεις ίσο με 10. Πιο συγκεκριμένα στην εικόνα 30 φαίνεται η συμπεριφορά των μεθόδων για την ευκλείδεια απόσταση, ενώ στις εικόνες 31,32 για τον αλγόριθμο χρονικής παραμόρφωσης και τον επαναληπτικό αλγόριθμο αντίστοιχα.
Όπως μπορούμε να παρατηρήσουμε οι δύο μέθοδοι έχουν το ίδιο περίπου υπολογιστικό κόστος, με την δεύτερη μέθοδο να έχει οριακά μικρότερο χρόνο. Επίσης όσο αυξάνεται ο αριθμός των αντικειμένων αυξάνεται και το υπολογιστικό κόστος των δύο μεθόδων



Εικόνα 30 Υπολογιστικό κόστος των δύο μεθόδων – ευκλείδεια απόσταση για μεταβαλλόμενο αριθμό αντικειμένων

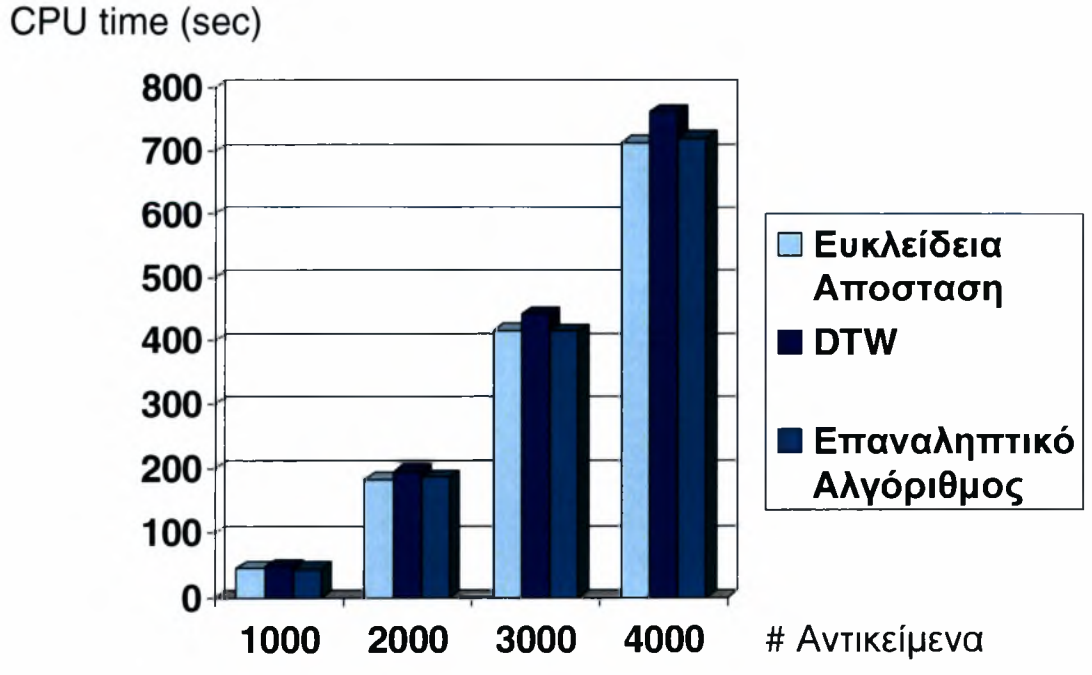


Εικόνα 31 Υπολογιστικό κόστος των δύο μεθόδων – DTW για μεταβαλλόμενο αριθμό αντικειμένων



Εικόνα 32 Υπολογιστικό κόστος των δύο μεθόδων – Επαναληπτικός Αλγόριθμος για μεταβαλλόμενο αριθμό αντικειμένων

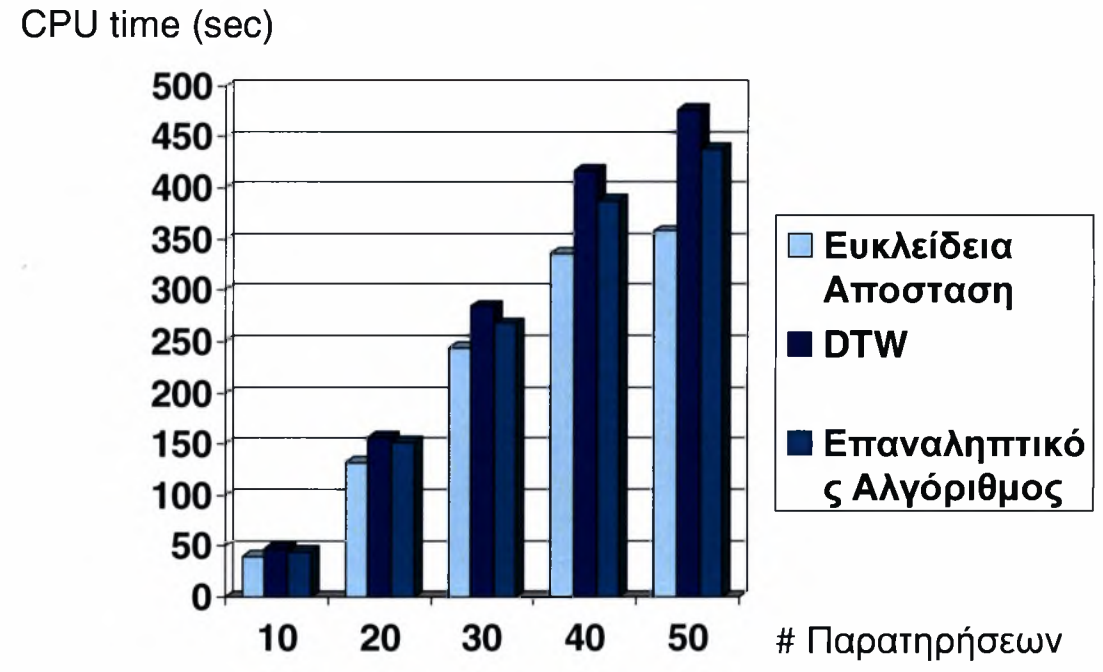
Επίσης στην εικόνα 33 βλέπουμε συγκριτικά τους χρόνους απόκρισης των τριών συναρτήσεων σύγκρισης για την πρώτη μέθοδο. Όπως μπορούμε να διακρίνουμε, η ευκλείδεια απόσταση και ο επαναληπτικός αλγόριθμος έχουν σχεδόν ίσους χρόνους απόκρισης ενώ το μεγαλύτερο υπολογιστικό κόστος, όπως και στην περίπτωση των πραγματικών δεδομένων έχει ο αλγόριθμος της δυναμικής χρονικής παραμόρφωσης.



Εικόνα 33 Χρόνοι απόκρισης συναρτήσεων σύγκρισης για μεταβαλλόμενο αριθμό αντικειμένων

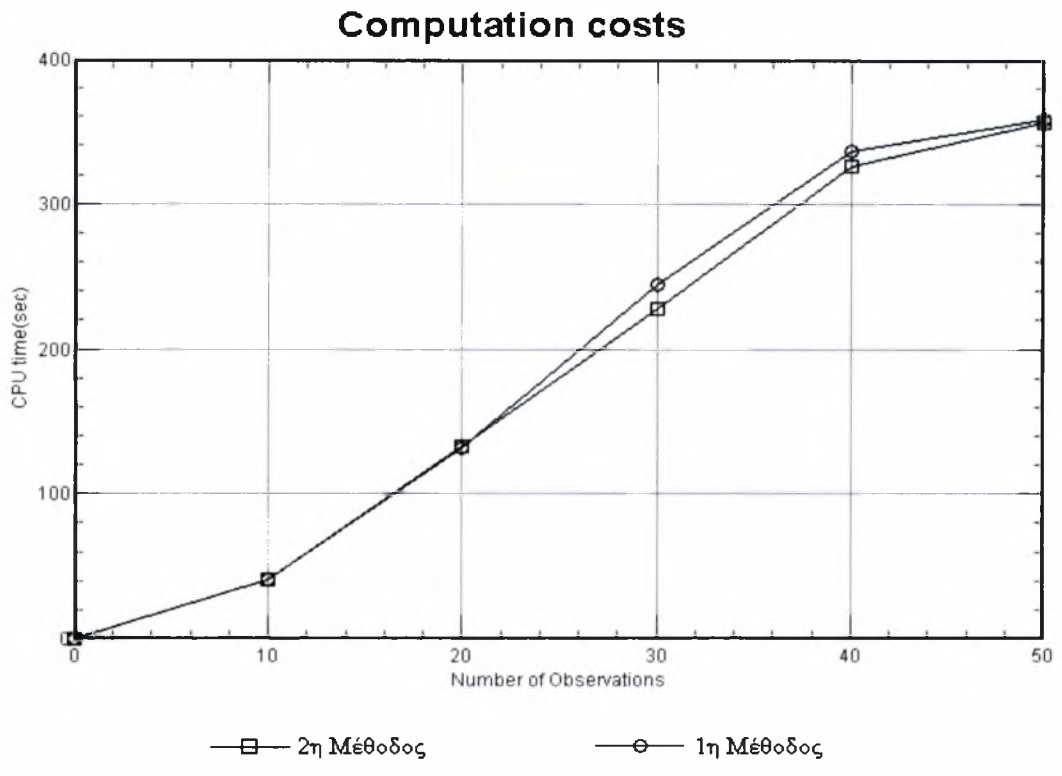
Στη συνέχεια θα μελετήσουμε το υπολογιστικό κόστος των δύο μεθόδων στην περίπτωση που μεταβάλλεται ο αριθμός των παρατηρήσεων, δηλαδή το μήκος των τροχιών. Με τη γεννήτρια δεδομένων δημιουργούμε ένα σύνολο 1000 αντικειμένων με μεταβαλλόμενα μήκη. Οι μετρήσεις έχουν γίνει για 10 κατόχους δεδομένων.

Στην παρακάτω εικόνα 34 βλέπουμε συγκριτικά τους χρόνους απόκρισης των τριών συναρτήσεων σύγκρισης για την πρώτη μέθοδο. Παρατηρούμε ότι η ευκλείδεια απόσταση έχει μικρότερο χρόνο απόκρισης σε σχέση με τους άλλους δύο αλγορίθμους όσο αυξάνεται το μήκος των τροχιών. Αυτό οφείλεται στη γεννήτρια δεδομένων και στο γεγονός ότι δεν μπορεί αν δημιουργήσει τροχιές ακριβώς ίδιου μήκους. Για αυτόν τον λόγο επιλέξαμε ο αριθμός των παρατηρήσεων στα πειράματα που ακολουθούν να είναι 10,20,30,40,50 ούτως ώστε να επιτύχουμε όσο το δυνατόν ίδιο αριθμό παρατηρήσεων ανά αντικείμενο.

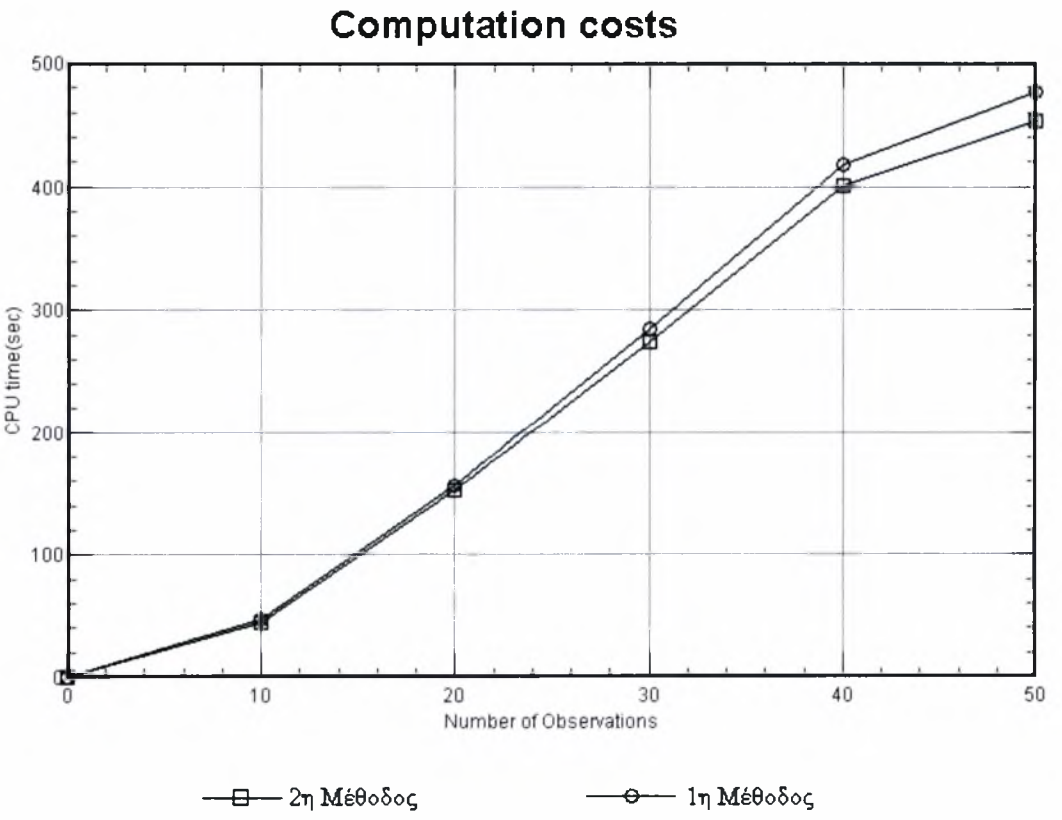


Εικόνα 34 Χρόνοι απόκρισης συναρτήσεων σύγκρισης για μεταβαλλόμενο αριθμό παρατηρήσεων

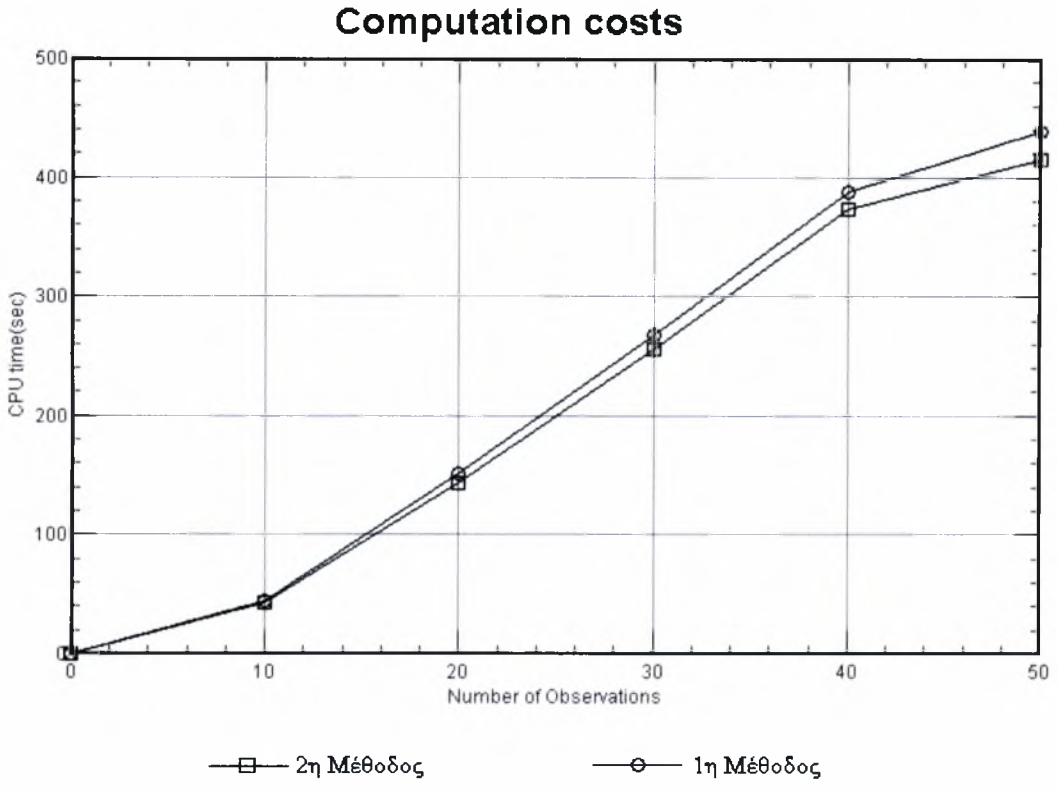
Στις παρακάτω εικόνες 35,36,37 παραθέτουμε το υπολογιστικό κόστος δύο μεθόδων για την ευκλείδεια απόσταση, τον αλγόριθμο χρονικής παραμόρφωσης και τον επαναληπτικό αλγόριθμο αντίστοιχα με μεταβαλλόμενα μήκη τροχιών. Ομοίως και σε αυτήν την περίπτωση οι δύο μέθοδοι έχουν παραπλήσιους χρόνους απόκρισης, με την δεύτερη μέθοδο να έχει το μικρότερο υπολογιστικό κόστος.



Εικόνα 35 Υπολογιστικό κόστος των δύο μεθόδων – ευκλείδεια απόσταση για μεταβαλλόμενο αριθμό παρατηρήσεων



Εικόνα 36 Υπολογιστικό κόστος των δύο μεθόδων – DTW για μεταβαλλόμενο αριθμό παρατηρήσεων

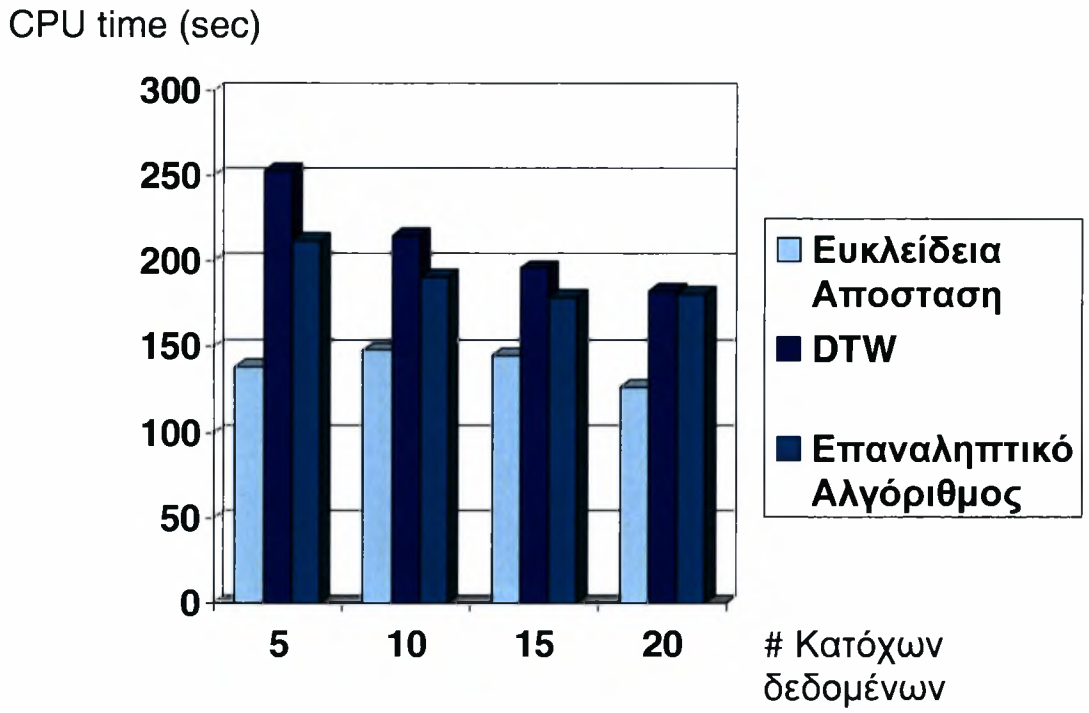


Εικόνα 37 Υπολογιστικό κόστος των δύο μεθόδων – επαναληπτικός αλγόριθμος για μεταβαλλόμενο αριθμό παρατηρήσεων

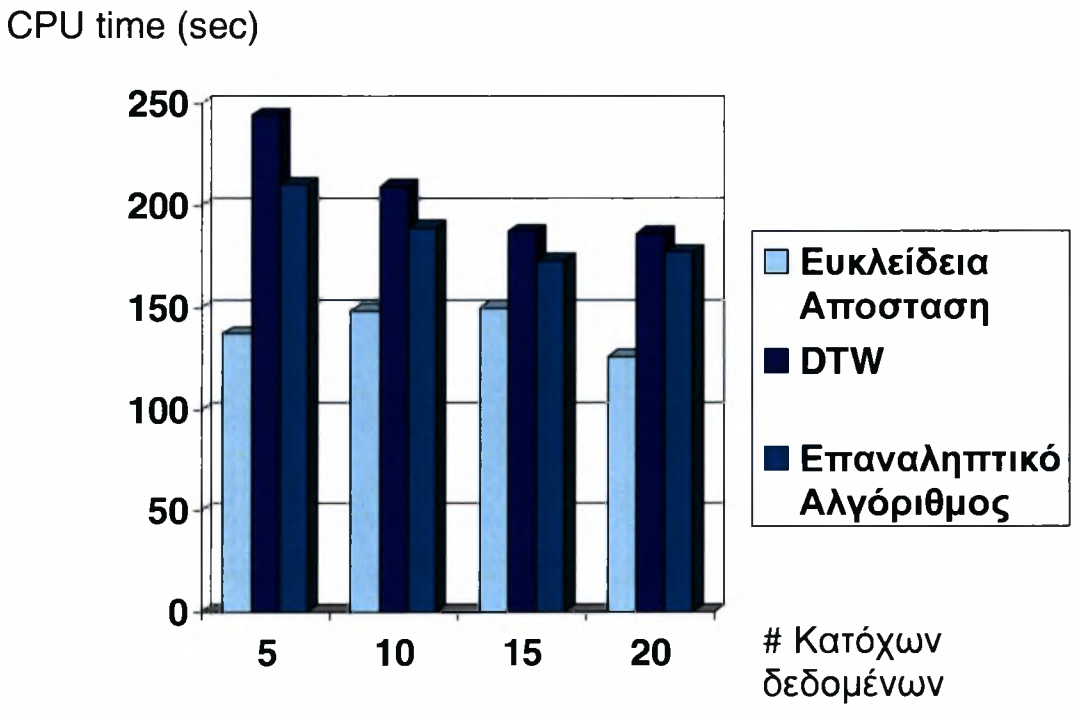
Για να ολοκληρώσουμε τις μετρήσεις θα εξετάσουμε την συμπεριφορά των μεθόδων όταν μεταβάλλεται ο αριθμός των κατόχων δεδομένων. Δημιουργούμε ένα σύνολο 500 αντικειμένων με 100 παρατηρήσεις ανά αντικείμενο. Στον παρακάτω πίνακα 9 βλέπουμε το υπολογιστικό κόστος των δύο μεθόδων για μεταβαλλόμενο αριθμό κατόχων δεδομένων και για όλες τις συναρτήσεις σύγκρισης. Όπως και στις περιπτώσεις όπου μεταβάλλαμε τον αριθμό των αντικειμένων, παρατηρήσεων ,οι δύο μέθοδοι εμφανίζουν παρόμοιους χρόνους απόκρισης. Επίσης παρατηρούμε ότι το υπολογιστικό κόστος μειώνεται όσο αυξάνεται ο αριθμός των κατόχων δεδομένων.

Κάτοχοι Δεδομένων	**5**		**10**		**15**		**20**	
Μέθοδος	**1^{H}**	**2^{H}**	**1^{H}**	**2^{H}**	**1^{H}**	**2^{H}**	**1^{H}**	**2^{H}**
Ευκλείδεια Απόσταση	138,3	137,2	148,1	148,61	145,3	149,88	126,6	125,95
DTW	252,9	244,13	215,1	209,31	196,1	187,2	183,1	186,07
Επαναληπτικός Αλγόριθμος	212,1	210,3	190,8	189,22	179,2	173,09	181,1	177,33

Πίνακας 9 Υπολογιστικό κόστος των δύο μεθόδων για μεταβαλλόμενο αριθμό κατόχων δεδομένων



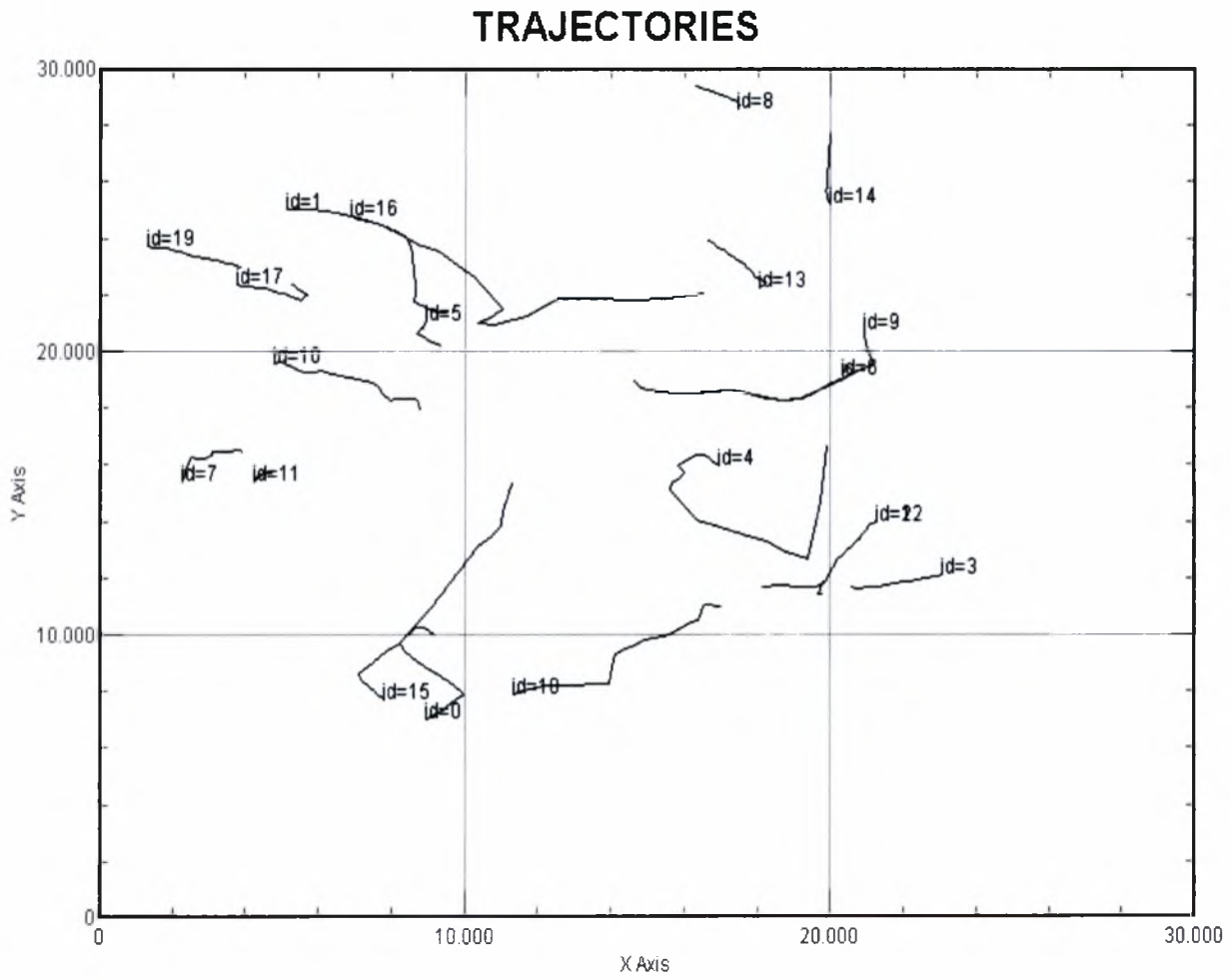
Εικόνα 38 Υπολογιστικό κόστος 1^H μεθόδου για μεταβαλλόμενο αριθμό κατόχων δεδομένων



Εικόνα 39 Υπολογιστικό κόστος 2^H μεθόδου για μεταβαλλόμενο αριθμό κατόχων δεδομένων

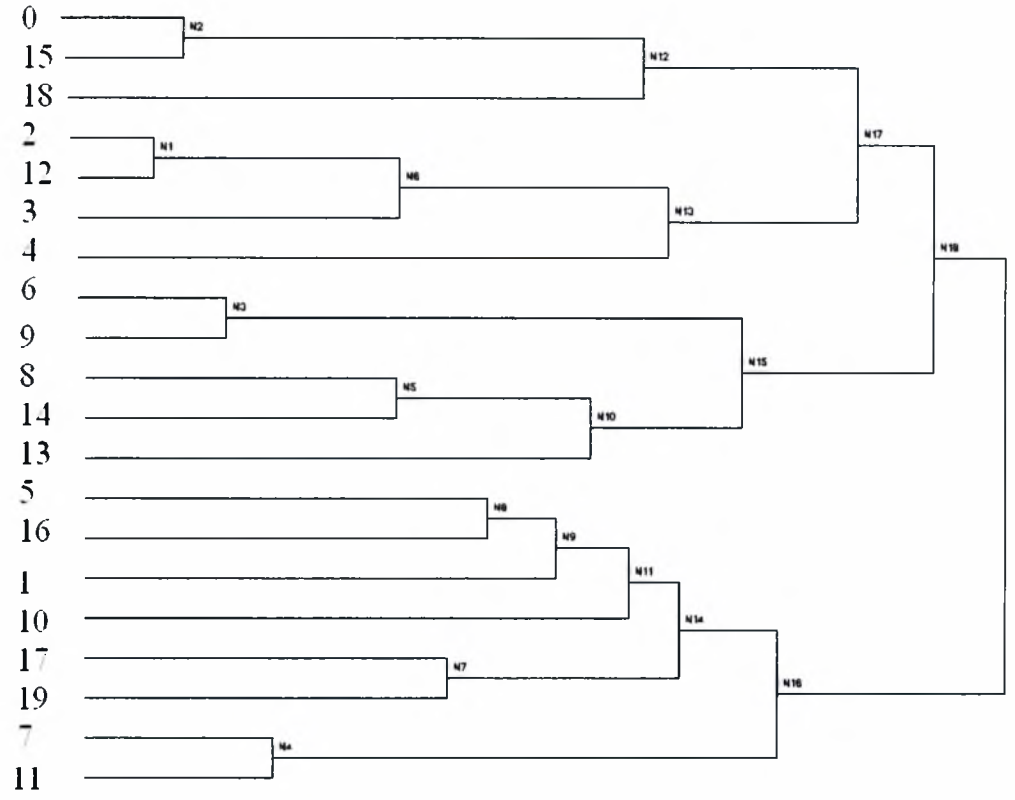
6. 3 Αξιολόγηση συσταδοποίησης

Για να μπορέσουμε να αξιολογήσουμε τα αποτελέσματα της συσταδοποίησης, θα δημιουργήσουμε με την γεννήτρια δεδομένων του Brinkhoff, ένα μικρό σύνολο κινούμενων αντικειμένων και αντίστοιχα παρατηρήσεων και στο οποίο μπορούμε να ελέγξουμε τις αποστάσεις μεταξύ των τροχιών.
Στην παρακάτω εικόνα 40 φαίνεται ένα σύνολο 20 τροχιών σε δυσδιάστατο επίπεδο το οποίο έχουμε δημιουργήσει με την γεννήτρια δεδομένων. Οι παρατηρήσεις των αντικειμένων έχουν καταγραφεί για τα ίδια χρονικά διαστήματα, πιο συγκεκριμένα για 20 χρονικά διαστήματα, επομένως ο χρόνος δεν επηρεάζει την συσταδοποίηση. Επίσης στην παρακάτω εικόνα βλέπουμε και τις ταυτότητες των αντικειμένων επάνω στις τροχιές τους.



Εικόνα 40 Τροχιές 20 αντικειμένων σε δυσδιάστατο χώρο

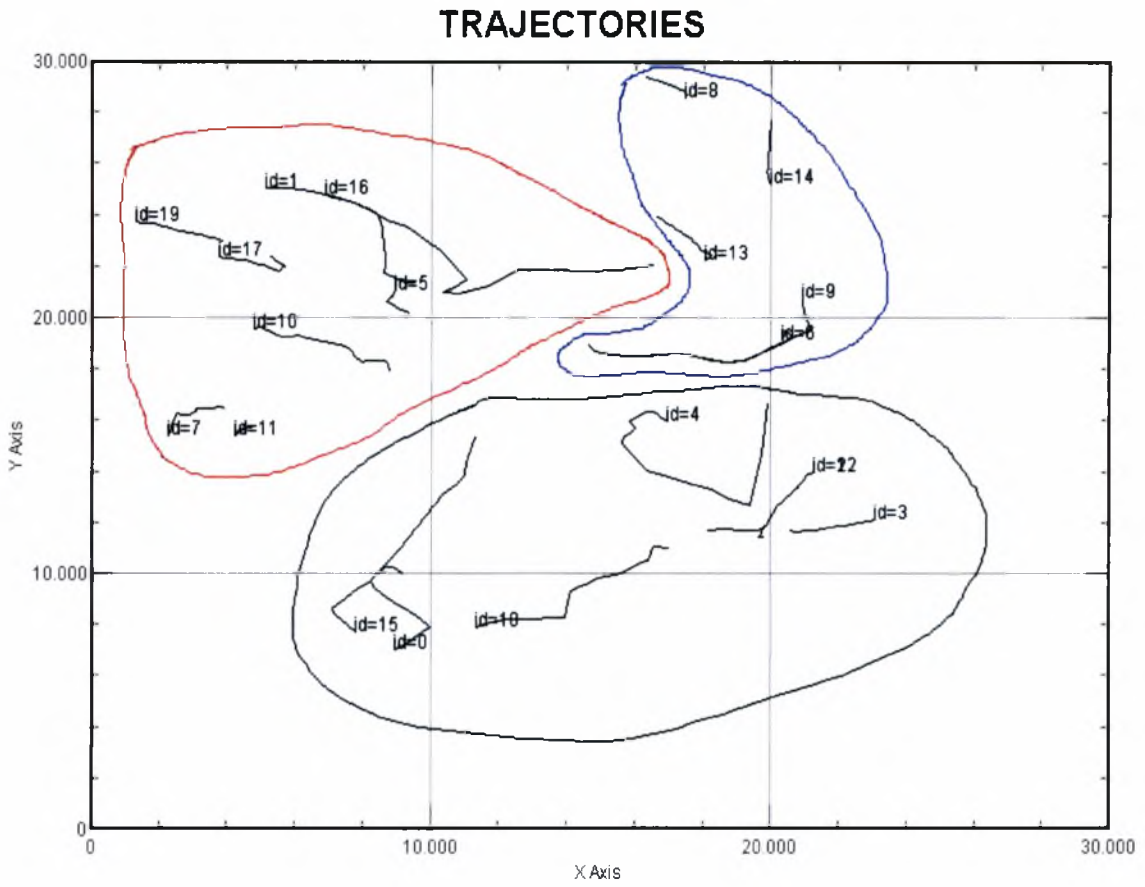
Στις παρακάτω εικόνες 41,43,45, βλέπουμε την έξοδο του ιεραρχικού αλγορίθμου σε μορφή δενδρογράμματος για την ευκλείδεια απόσταση ,τον DTW αλγόριθμο και τον επαναληπτικό αλγόριθμο αντίστοιχα. Στα αριστερά κάθε εικόνας φαίνονται οι ταυτότητες των αντικειμένων.



Εικόνα 41 Ιεραρχικός Αλγόριθμος-Ευκλείδεια Απόσταση

Με βάση την ευκλείδεια απόσταση ο ιεραρχικός αλγόριθμος δημιουργεί τους δύο πρώτους κόμβους N1(2,12) και N2(0,15) το οποίο είναι αναμενόμενο αφού οι τροχιές των αντικειμένων 2,12 και 0,15 συμπίπτουν (εικόνα 40). Στη συνέχεια δημιουργούνται οι κόμβοι N3 (6,9) και N4 (7,11). Στο επόμενο βήμα του αλγορίθμου ενώνονται τα αντικείμενα 8,14 στον κόμβο N5. Ο κόμβος N1 ενώνεται με το αντικείμενο 3 (κόμβος N6) και στη συνέχεια με το αντικείμενο 4 (κόμβος N13). Στον κόμβο N7 ενώνονται τα αντικείμενα 17,19. Στη συνέχεια ο αλγόριθμος δημιουργεί τον κόμβο N8 (5,16) και τον κόμβο N9 (N8,1). Όπως αναφέραμε η ευκλείδεια απόσταση δημιουργεί και ελέγχει τροχιές ίδιου μήκους για αυτόν τον λόγο ενώθηκαν πρώτα τα αντικείμενα 5 και 16. Το ίδιο ισχύει και στην περίπτωση του κόμβου N10 (N5, 13). Το αντικείμενο 10 ενώνεται με τον κόμβο N9 (κόμβος N11) και στη συνέχεια με τα αντικείμενα 17,19 (κόμβος N14) και 7,11 (κόμβος N16). Ο κόμβος N15 δημιουργείται από την ένωση των κόμβων N3,N10 και ο κόμβος N17 από την ένωση των κόμβων N12, N13. Σε αυτό το σημείο ο αλγόριθμος έχει δημιουργήσει τρεις συστάδες εικόνα 42.

1^H (5,16,1,10,17,19,7,11)
2^H (6,9,8,14,13)
3^H (0,15,18,2,12,3,4)

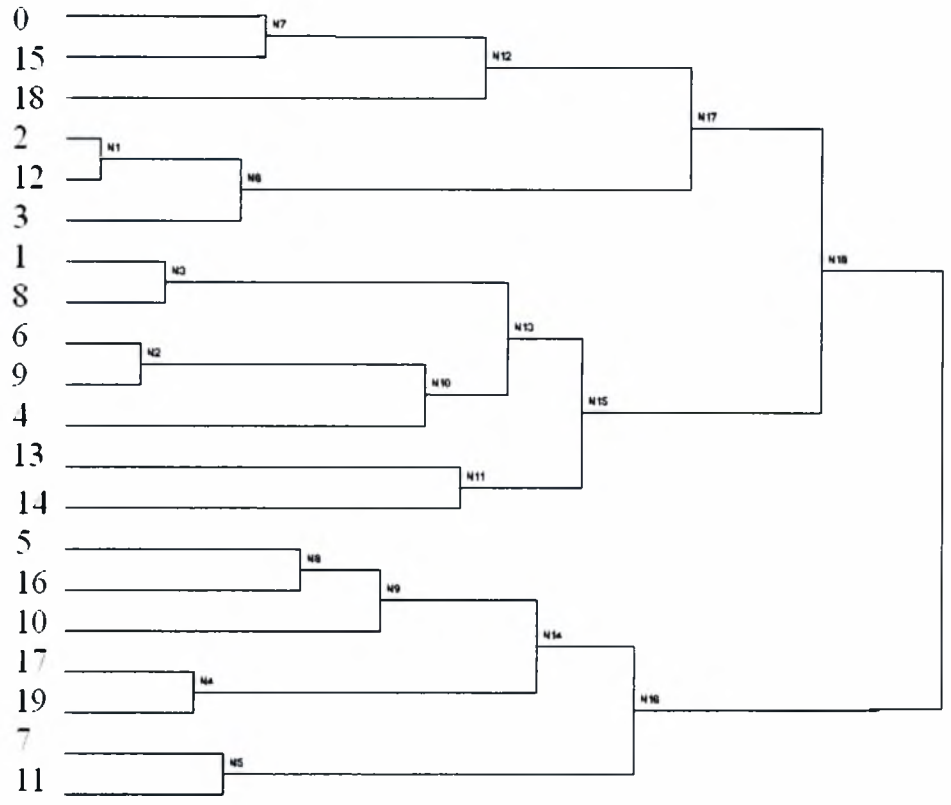


Εικόνα 42 Ευκλείδεια Απόσταση -3 συστάδες

Στο τελευταίο βήμα του αλγορίθμου ενώνονται οι κόμβοι N15,N17 δηλαδή η δεύτερη και η τρίτη συστάδα ,στον κόμβο N19 και στη συνέχεια η τρίτη συστάδα με τον κόμβο N19. Τα αποτελέσματα μας δείχνουν ότι η ευκλείδεια απόσταση μπορεί να αποδειχθεί σωστό κριτήριο σύγκρισης μόνο στην περίπτωση τροχιών ίδιου μήκους.

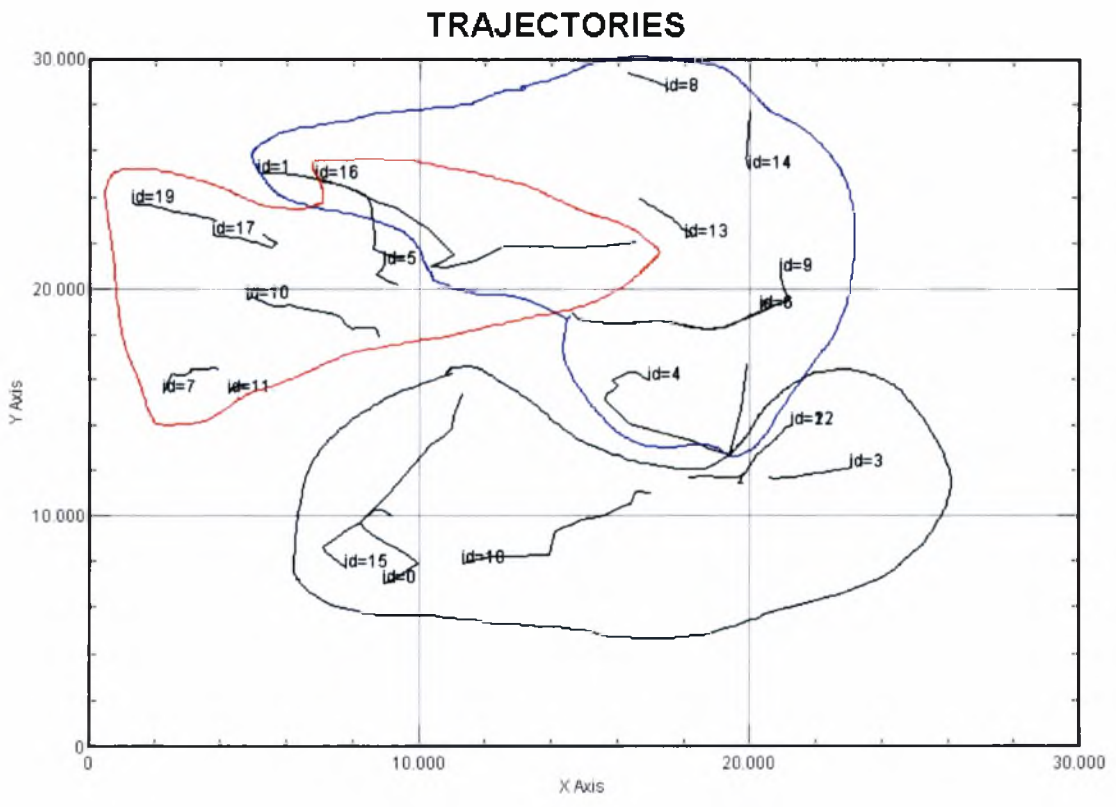
Στην εικόνα 43 φαίνεται η συσταδοποίηση με βάση τον αλγόριθμο δυναμικής χρονικής παραμόρφωσης. Ο ιεραρχικός αλγόριθμος δημιουργεί τους δύο πρώτους κόμβους N1(2,12) και N2(6,9). Στη συνέχεια δημιουργούνται οι κόμβοι N3 (1,8) και N4 (17,19). Στο επόμενο βήμα του αλγορίθμου ενώνονται τα αντικείμενα 7,11 στον κόμβο N5. Ο κόμβος N1 ενώνεται με το αντικείμενο 3 (κόμβος N6) και στη συνέχεια τα αντικείμενα 0,15 (κόμβος N7). Στον κόμβο N8 ενώνονται τα αντικείμενα 5,16 και έπειτα ο N8 με το αντικείμενο 10 (κόμβος N9). Στη συνέχεια ο αλγόριθμος δημιουργεί τον κόμβο N10 (N2,4) και τον κόμβο N11 (13,14). Στο επόμενο βήμα ενώνονται οι κόμβοι N12,N6 (κόμβος N17), N13, N11 (κόμβος N15) και N14, N5 (κόμβος N16). Έτσι δημιουργούνται οι συστάδες

1^H (0,15,18,2,12,3)
2^H (1,8,6,9,4,13,14)
3^H (5,16,10,17,19,7,11)



Εικόνα 43 Ιεραρχικός Αλγόριθμος-DTW

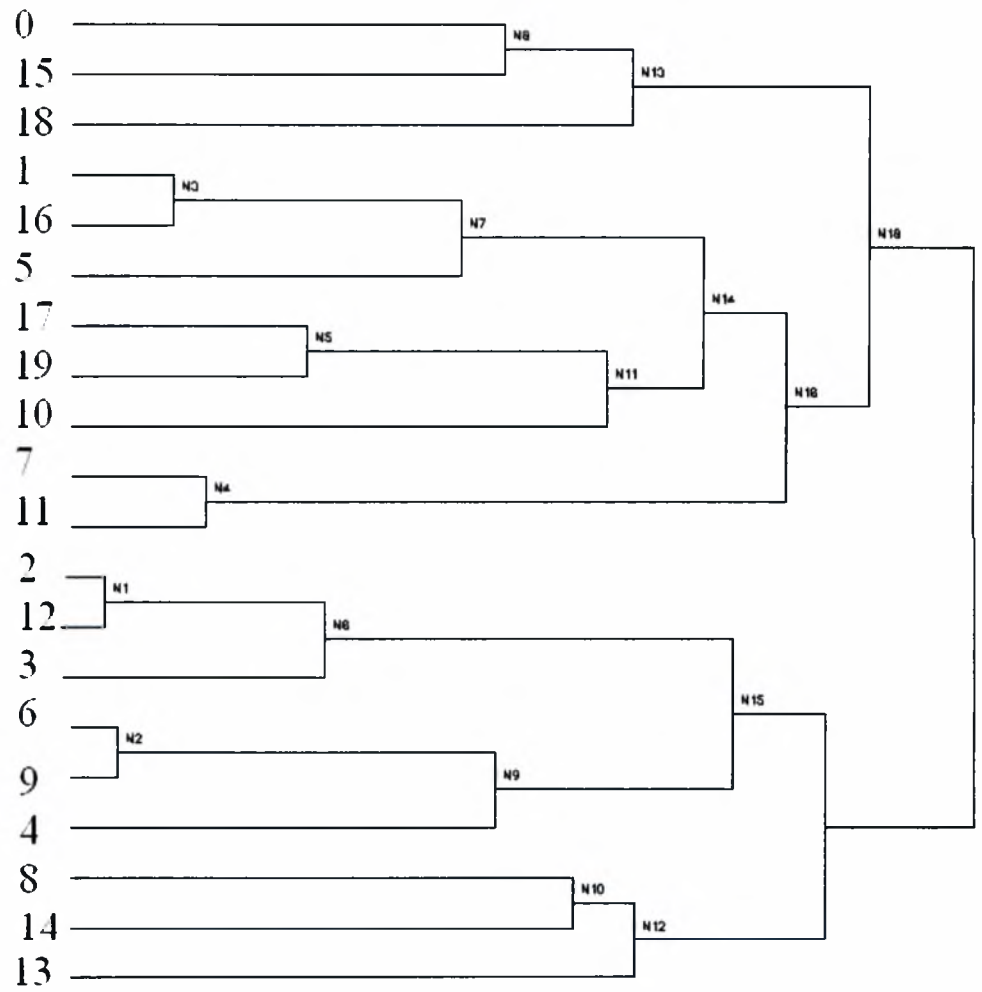
Στην εικόνα 44 βλέπουμε τις παραγόμενες συστάδες



Εικόνα 44 DTW -3 συστάδες

Ο αλγόριθμος DTW όπως αναφέραμε είναι ανεξάρτητος από το μήκος των τροχιών, για αυτόν τον λόγο ενώθηκε το αντικείμενο 1 με την 2^H συστάδα, σε αντίθεση με την ευκλείδεια απόσταση. Ωστόσο έχει μεγάλο υπολογιστικό κόστος όπως μας έδειξαν τα πειράματα. Στο τελευταίο βήμα του αλγορίθμου ενώνονται οι κόμβοι N15,N17 (κόμβος N19) και στη συνέχεια οι κόμβοι N19,N16.

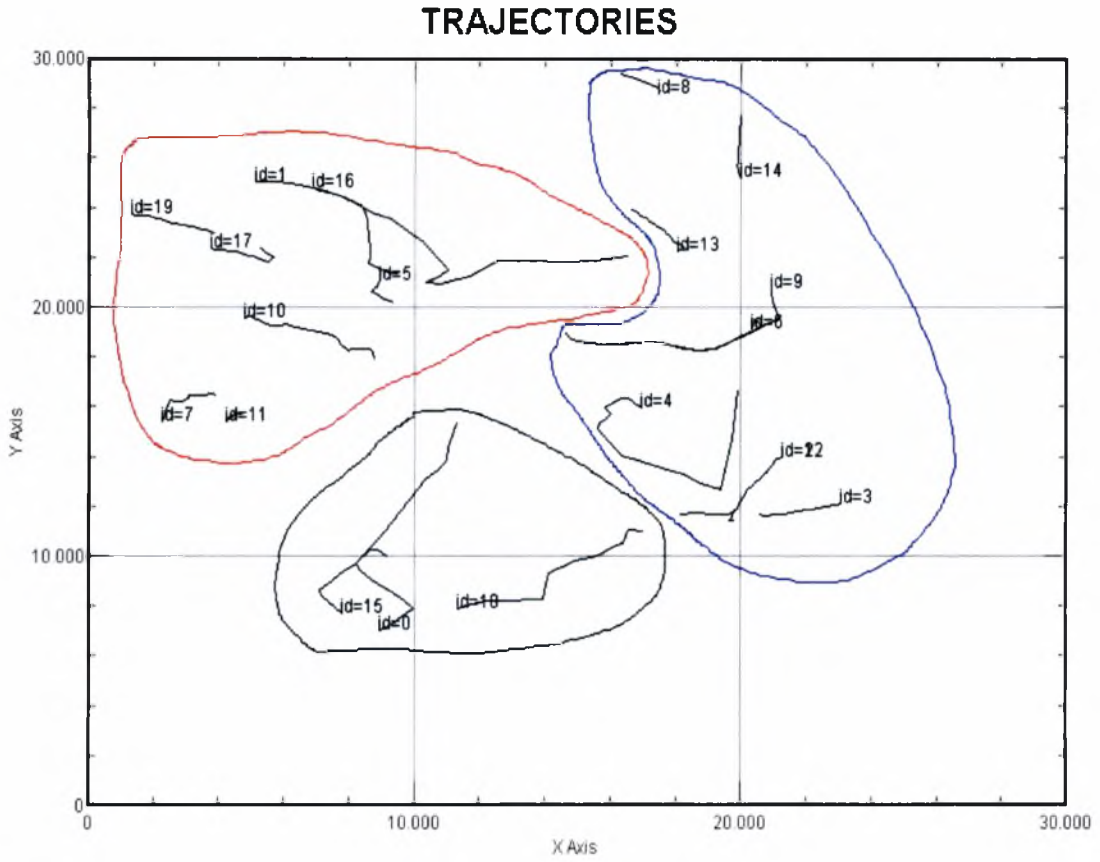
Τέλος στην εικόνα 45 φαίνεται η έξοδος του αλγορίθμου για την περίπτωση του επαναληπτικού αλγορίθμου σύγκρισης.



Εικόνα 45 Ιεραρχικός Αλγόριθμος-Επαναληπτικός Αλγόριθμος

Ο ιεραρχικός αλγόριθμος δημιουργεί τους δύο πρώτους κόμβους N1(2,12) και N2(6,9). Στη συνέχεια δημιουργούνται οι κόμβοι N3 (1,16) και N4 (7,11). Στο επόμενο βήμα του αλγορίθμου ενώνονται τα αντικείμενα 17,19 στον κόμβο N5. Ο κόμβος N1 ενώνεται με το αντικείμενο 3 (κόμβος N6) και ο κόμβος N3 και το αντικείμενο 5 (κόμβος N7). Στον κόμβο N8 ενώνονται τα αντικείμενα 0,15 και έπειτα ο N8 με το αντικείμενο 18 (κόμβος N13). Στη συνέχεια ο αλγόριθμος δημιουργεί τον κόμβο N9 (N2,4) και τον κόμβο N10 (8,14).Ο κόμβος N10 ενώνεται με το αντικείμενο 13 (κόμβος N12) Στο επόμενο βήμα ενώνονται οι κόμβοι N6,N9 (κόμβος N15), N4, N14 (κόμβος N16) και N15 ,N1.2 Έτσι δημιουργούνται οι συστάδες

1^H (0,15,18)
2^H (1,16,5,17,19,10,7,11)
3^H (2,12,3,6,9,4,8,13)
Στο τελευταίο βήμα του αλγορίθμου ενώνονται οι κόμβοι N16,N13 (κόμβος N19) και στη συνέχεια οι κόμβοι N19,N17.
Στην εικόνα 46 βλέπουμε τις παραγόμενες συστάδες με τον επαναληπτικό αλγόριθμο.



Εικόνα 46 Επαναληπτικός Αλγόριθμος -3 συστάδες

Ο επαναληπτικός αλγόριθμος είναι ανεξάρτητος από το μήκος των τροχιών και δείχνει να έχει την πιο ορθολογική συμπεριφορά στην ανάθεση των συστάδων.

Κεφάλαιο 7

Επίλογος

Στην παρούσα εργασία μελετήσαμε ζητήματα ιδιωτικότητας χώρο-χρονικών δεδομένων από την σκοπιά της μοντελοποίησης δεδομένων και πιο συγκεκριμένα κατά την συσταδοποίηση δεδομένων. Επικεντρωθήκαμε στην διατήρηση της ιδιωτικότητας και υλοποιήσαμε δύο μεθόδους για την επίτευξη της.
Βασιζόμενοι σε μια γνωστή μέθοδο προστασίας της ιδιωτικότητας, η οποία έχει σαν στόχο την κατασκευή ενός καθολικού πίνακα ανομοιότητας μέσω ενός πρωτοκόλλου ασφαλούς ανταλλαγής δεδομένων μεταξύ των κατόχων δεδομένων και ενός τρίτου συμμετέχοντα, υλοποιήσαμε μια δεύτερη μέθοδο η οποία βελτιώνει την πρώτη μέθοδο σε θέματα ασφάλειας και κυρίως περιορίζει τον ρόλο του τρίτου συμμετέχοντα. Ο στόχος και των δύο μεθόδων δηλαδή ο καθολικός πίνακας ανομοιότητας μπορεί να χρησιμοποιηθεί από οποιονδήποτε αλγόριθμο συσταδοποίησης. Ωστόσο η μέτρηση της ανομοιότητας χώρο-χρονικών δεδομένων δεν είναι μια εύκολη υπόθεση για αυτόν τον λόγο υλοποιήσαμε τρεις διαφορετικές συναρτήσεις μέτρησης της ανομοιότητας, την ευκλείδεια απόσταση, έναν δυναμικό αλγόριθμο χρονικής παραμόρφωσης και έναν επαναληπτικό αλγόριθμο, όπως επίσης και δύο διαφορετικούς αλγόριθμους συσταδοποίησης για την εξαγωγή των αποτελεσμάτων.
Στα πλαίσια της εργασίας κατασκευάσαμε μια εύχρηστη εργαλειοθήκη η οποία περιέχει τα όσα αναφέραμε παραπάνω και μπορεί να χρησιμοποιηθεί για μετρήσεις για διαφορετικά αρχεία δεδομένων.
Για την μέτρηση του υπολογιστικού κόστους των δύο μεθόδων χρησιμοποιήσαμε δύο τύπους αρχείων με πραγματικά και συνθετικά δεδομένα. Τα αποτελέσματα των πειραμάτων έδειξαν ότι οι δύο μέθοδοι συμπεριφέρονται περίπου το ίδιο, έχουν δηλαδή το ίδιο υπολογιστικό κόστος, το οποίο είναι ιδιαίτερα σημαντικό αν αναλογιστούμε το κέρδος της δεύτερης μεθόδου ως προς την ασφάλεια. Επίσης οι μετρήσεις έβγαλαν χρήσιμα συμπεράσματα και για τις συναρτήσεις σύγκρισης όπου η ευκλείδεια απόσταση έχει μικρότερο υπολογιστικό κόστος σε σχέση με τον επαναληπτικό αλγόριθμο και τον δυναμικό αλγόριθμο, ο οποίος έχει το μεγαλύτερο κόστος. Ωστόσο η ευκλείδεια απόσταση με βάση τη αξιολόγηση της συσταδοποίησης δεν έχει τόσο καλή συμπεριφορά όσο οι άλλοι δύο αλγόριθμοι σύγκρισης.
Οι μέθοδοι που προτάθηκαν στην παρούσα εργασία μπορούν να επεκταθούν και σε άλλες τεχνικές της μοντελοποίησης δεδομένων, εκτός της συσταδοποίησης, όπως σε αλγορίθμους εύρεσης πλησιέστερου γείτονα ή προβλήματα ανίχνευσης έκτοπων.
Το γενικότερο συμπέρασμα της εργασίας είναι ότι σε ένα τόσο αναπτυσσόμενο τομέα, όπως είναι τα χώρο-χρονικά δεδομένα μπορούμε να επιτύχουμε την καλύτερη δυνατή προστασία των δεδομένων με το μικρότερο δυνατό κόστος υπολογισμών.

Αναφορές:

[1] S.R.M. Oliveira, O.R. Zaïane, Achieving privacy preservation when sharing data for clustering, in: Proceedings of the International Workshop on Secure Data Management in a Connected World, 2004

[2] S.R.M. Oliveira, O.R. Zaïane, Privacy preserving clustering by data transformation, in: Proceedings of the 18th Brazilian Symposium on Databases, 2003

[3] S.R.M. Oliveira, O.R. Zaïane, Privacy preserving clustering by object similarity-based representation and dimensionality reduction transformation, in: Proceedings of the 2004 ICDM Workshop on Privacy and Security Aspects of Data Mining, 2004

[4] P. Kusuma Kumari KVSVN Raju S. Srinivasa Rao "Privacy Preserving in Clustering using Fuzzy Sets,in: Proceedings of the 2006 International Conference on Data Mining, DMIN 2006, Las Vegas, Nevada, USA

[5] S S. Jha, L. Kruger, P. McDaniel, Privacy preserving clustering, in: Proceedings of the 10th European Symposium on Research in Computer Security, 2005

[6] G. Jagannathan, K. Pillaipakkamnatt, R. Wright, A new privacy-preserving distributed k-clustering algorithm, in: Proceedings of the 6th SIAM International Conference on Data Mining, 2006

[7] Jaideep Vaidya , Chris Clifton, Privacy-preserving *k*-means clustering over vertically partitioned data, in: Proceedings of the 9th ACM SIGKDD international conference on Knowledge discovery and data mining, August 24-27, 2003, Washington, D.C.

[8] G. Jagannathan and R.N. Wright, Privacy-Preserving Distributed Clustering over Arbitrarily Partitioned Data, in: Proceedings of the 11th ACM SIGKDD Int'l Conf. Knowledge *Discovery and Data Mining,* 2005

[9] D.K. Tasoulis1, E.C. Laskari, G.K. Meletiou, M.N. Vrahatis, Privacy Preserving Unsupervised Clustering over Vertically Partitioned Data, in: Proceedings of International Conference on Computational Science and its Applications, Glasgow , (2006)

[10] Waseem Ahmad, Ashfaq Khokhar, Phoenix: Privacy Preserving Biclustering on Horizontally Partitioned Data amid Malicious Adversaries, Accepted for ACM SIGKDD International Workshop of Privacy, Security and Trust in KDD, San Jose, 2007

[11] Merugu, S. and J. Ghosh, A privacy-sensitive approach to distributed clustering, Pattern Recognition Letters, vol.26, no.4.

[12] A. Inan Y. Saygyn E. Savas A.A. Hintoglu A. Levi ,Privacy Preserving Clustering on Horizontally Partitioned Data, in: Proceedings of the 22nd International Conference on Data Engineering Workshops, 2006.

[13] Ali Ulvi Kasapoğlu, Mahir Can Doğanay "Privacy-Preserving Distributed Data Mining On Trajectories Using Multiplicative Data Perturbation"

[14] The Spatio-Temporal Resources site

http://dke.cti.gr/people/pfoser/data.html

[15] Rakesh Agrawal, Christos Faloutsos, Arun Swami. Efficient similarity search in sequence databases. in: Proceedings of the 4th International ConferenceonFoundations of Data Organization and Algorithms, Chicago, October 1993

[16] Kin-pong Chan ,Ada Wai-chee Fu ,Department of Computer Science and Engineering,Efficient time series matching by Wavelets in: Proceedings of the 15th International Conference on Data Engineering, 1999

[17] Berndt, D.J. and Clifford, J. Finding patterns in time series:A dynamic programming approach. In Advances in Knolwedge Discovery and Data Mining, Fayyad, U.M., Piatesky-Shapiro, G., Smyth, P., and Uthursamy, R., Eds., Menlo Park, CA: AAAI Press,1996

[18] Yasushi Sakurai, Masatoshi Yoshikawa, Christos Faloutsos. FTW: fast similarity search under the time warping distance. In Proceedings of PODS, 2005

[19] Michail Vlachos,George Kollios,Dimitrios Gunopulos, Discovering similar multidimensional Trajectories, in: Proceedings of the 18th International Conference on Data Engineering 2002

[20] Lei Chen, M. Tamer Ozsu, Vincent Oria, Robust and fast similarity search for moving object Trajectories, in: Proceedings of the 2005 ACM SIGMOD international conference on Management of data, 2005

[21] Byoung-Kee Yi, H.V. Jagadish, Christos Faloutsos, Efficient retrieval of time sequences under Time Warping, in: Proceedings of Proceedings of the Fourteenth International Conference on Data Engineering,1998

[22] Perttu Laurinen, Pekka Siirtola, Juha Röning, Intelligent Systems Group, Computer Engineering Laboratory University of Oulu,Efficient algorithm for calculating similarity between trajectories containing an increasing dimension, in: Proceedings of of the 24th IASTED international conference on Artificial intelligence and applications,2006

[23] Chris Clifton,Wenliang (Kevin) Du , Mikhail Atallah,Collaborative Research :ITR : Distributed Data Mining to Protect Information Privacy, (NSF-ITR, 8/03-7/06, PI)

[24] A. Inan, Y. Saygin, Privacy-preserving spatio-temporal clustering on horizontally partitioned data, in: Proceedings of DAWAK06, Eighth International Conference on Data Warehousing and Knowledge Discovery, 2006.

[25] Ester M., Kriegel H.-P., Xu X.: "A Database Interface for Clustering in Large Spatial Databases", in: Proceedings of the 1st Int. Conf. on Knowledge Discovery & Data Mining (KDD '95), Montreal, Canada, 1995

[26] The R-tree portal site

http://www.rtreeportal.org/

[27] The Visad Home Page

http://www.ssec.wisc.edu/~billh/visad.html

[28] Ddraw-A simple Application to draw Dendrograms using the Minimum Spanning Tree Algorithm

http://lis.snv.jussieu.fr/~chalubert/ddraw/ddraw.html

[29] The Java Plot Package site

http://homepage.mac.com/jhuwaldt/java/Packages/Plot/PlotPackage.html