Πανεπιστήμιο Θεσσαλίας
Τμήμα Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων

Τίτλος Διπλωματικής Εργασίας:

«Μέθοδοι για τη διατήρηση της ιδιωτικότητας κατά την εξόρυξη χωρο-χρονικών δεδομένων»

Επιβλέπων καθηγητής
Βερύκιος Βασίλειος

Ζαχαρούλη Πολυξένη

Βόλος 2006

Περιεχόμενα

1. Εισαγωγή 3
2. Σχετική βιβλιογραφία 6
3. Διατύπωση του προβλήματος και ορισμοί 9
3.1 Προσδιοριστές Προστασίας Βασιζόμενοι στη Τοποθεσία (Location Based Quasi-Identifiers) 9
3.2 Προσωπικό Ιστορικό Τοποθεσιών (Personal History of Locations) 16
3.3 Διασύνδεση (linkability) μεταξύ αιτήσεων και χρηστών 18
3.4. Κ-Ανωνυμία 21
4. Το μοντέλο της k-Ανωνυμίας για Πολλαπλούς Προσδιοριστές Προστασίας 23
4.1 Μοντέλο για τη διατήρηση της ανωνυμίας 24
4.2 Προτεινόμενη λύση για την επίτευξη k-ανωνυμίας με τη χρήση γενίκευσης και μικτών ζωνών 26
4.2.1 Αλγόριθμος γενίκευσης 27
4.2.2 Αλγόριθμος αποσύνδεσης 37
4.3 Ψευδοκώδικας 42
5. Πειραματικά Δεδομένα 46
Επίλογος 56
Βιβλιογραφία 59

1. Εισαγωγή

Οι διαρκείς εξελίξεις στον τομέα της ασύρματης τεχνολογίας είχαν ως αποτέλεσμα την αύξηση των υπηρεσιών που μπορούν να προσφέρουν οι ασύρματες συσκευές (όπως για παράδειγμα τα κινητά τηλέφωνα ή τα PDAs). Παραδείγματα τέτοιων υπηρεσιών είναι οι υπηρεσίες πλοήγησης, υπηρεσίες πληροφόρησης για την κυκλοφορία στους δρόμους ή για τις καιρικές συνθήκες, υπηρεσίες αποστολής εικόνας και βίντεο κτλ. Απαραίτητη προϋπόθεση για να σταλούν επιτυχώς αιτήσεις σε αυτές τις υπηρεσίες, είναι να αποστέλλονται στον παροχέα υπηρεσιών, τα αναγκαία κάθε φορά δεδομένα (τα οποία διαφοροποιούνται με βάση το είδος της υπηρεσίας). Για παράδειγμα οι υπηρεσίες πλοήγησης χρειάζονται τη θέση του οχήματος για να μπορέσουν να πληροφορήσουν τον οδηγό σχετικά με τις πιθανές διαδρομές που μπορεί να ακολουθήσει ώστε να φτάσει στον προορισμό του. Από τη άλλη πλευρά, υπηρεσίες όπως η αποστολή εικόνας και βίντεο, δεν χρειάζονται αυτού του είδους την πληροφορία για να εκτελεστούν.

Προκειμένου να προστατευτεί η ιδιωτικότητα των ατόμων που χρησιμοποιούν υπηρεσίες όπως αυτές που προαναφέρθηκαν, η ταυτότητα τους αντικαθίστανται με κάποιο ψευδώνυμο κάθε φορά που αυτοί αποστέλλουν μία αίτηση σε έναν παροχέα υπηρεσιών. Παρόλο που στις αιτήσεις δεν εμφανίζεται η ακριβής ταυτότητα του εκάστοτε αιτούντα, ο κίνδυνος αποκάλυψης ευαίσθητων πληροφοριών εξακολουθεί να υπάρχει. Με τον όρο *ευαίσθητες πληροφορίες* εννοούμε δεδομένα που αφορούν την ταυτότητα του αιτούντα (το ονοματεπώνυμο του, τη διεύθυνση κατοικίας του, το τηλέφωνο του, το επάγγελμα του κτλ) αλλά και συνήθειες του (όπως π.χ. μέρη όπου συχνάζει, τις θρησκευτικές και πολιτικές πεποιθήσεις του κτλ).

Αυτού του είδους οι ευαίσθητες πληροφορίες μπορούν –μεταξύ άλλων- να αποκαλυφθούν από τη θέση όπου βρισκόταν ο χρήστης όταν έγινε η αίτηση, αλλά και από πιθανές διαδρομές που αυτός συνηθίζει να ακολουθεί. Αν για παράδειγμα ένας χρήστης στέλνει διαρκώς αιτήσεις από κάποιο πρακτορείο με αγώνες στοιχήματος, είναι πολύ πιθανό ο συγκεκριμένος χρήστης να είναι εθισμένος στον τζόγο. Αυτή η πληροφορία θα μπορούσε να χρησιμοποιηθεί από έναν τραπεζικό φορέα για την απόρριψη κάποιας αίτησης του χρήστη για χορήγηση δανείου.

Παραδείγματα όπως τα παραπάνω, τονίζουν πόσο σημαντικό είναι να μην αποκαλύπτεται η ταυτότητα του αιτούντα ή οποιεσδήποτε ευαίσθητες πληροφορίες τον αφορούν, οποτεδήποτε αυτός αποστέλλει μία αίτηση σε κάποιον παροχέα υπηρεσιών. Στην παρούσα εργασία, θα παρουσιαστεί μία στρατηγική, της οποίας στόχος είναι η προστασία της ιδιωτικότητας των χρηστών που αποστέλλουν αιτήσεις. Υπάρχουν πολλές διαφορετικές τεχνικές οι οποίες έχουν αναπτυχθεί ώστε να επιτυγχάνεται ο παραπάνω στόχος. Μία από αυτές είναι και η στρατηγική η οποία προτάθηκε για πρώτη φορά στο [4]. Σε αυτή την εργασία θεωρήθηκε η απλή περίπτωση όπου κάθε χρήστης μπορεί να έχει μόνο έναν προσδιοριστή προστασίας[1]. Στην πραγματικότητα όμως το πλήθος και το είδος των προσδιοριστών προστασίας μπορεί να διαφέρει από χρήστη σε χρήστη. Για παράδειγμα, ο αριθμός φοιτητικού μητρώου μπορεί να χρησιμοποιηθεί για να σκιαγραφήσει μοναδικά μόνο όσα άτομα φοιτούν σε κάποιο τμήμα. Αν ένας χρήστης δεν είναι φοιτητής, ο αριθμός φοιτητικού μητρώου δεν μπορεί να αποκαλύψει την ταυτότητα του ή άλλες ευαίσθητες πληροφορίες που είναι σχετικές με αυτόν. Επιπλέον στο [4], η λύση που προτάθηκε δεν υλοποιήθηκε ώστε να αξιολογηθεί η αποτελεσματικότητα της μέσω πειραματικών δεδομένων. Η επέκταση αυτής της στρατηγικής (θεωρούνται πολλαπλοί προσδιοριστές προστασίας για κάθε χρήστη), η υλοποίηση της και η χρήση της για τη διεξαγωγή πειραματικών δεδομένων, αποτελούν τις βασικές συνεισφορές αυτής της εργασίας.

Η διάρθρωση της εργασίας έχει ως εξής. Στο Κεφάλαιο 2 θα παρουσιαστεί η σχετική βιβλιογραφία. Στο Κεφάλαιο 3, θα δοθεί η ακριβής διατύπωση του προβλήματος, ορισμοί και παραδείγματα. Στο Κεφάλαιο 4, θα οριστεί μία στρατηγική για την επίλυση του προβλήματος, η οποία θα συνοδεύεται από παραδείγματα για την ευκολότερη κατανόηση της. Επιπλέον, η στρατηγική αυτή θα παρουσιαστεί με τη μορφή ψευδοκώδικα. Στο Κεφάλαιο 5, η προτεινόμενη λύση θα εφαρμοστεί σε πειραματικά δεδομένα και θα παρατεθούν στατιστικά αποτελέσματα. Τέλος, θα αξιολογηθούν τα αποτελέσματα από την εφαρμογή της στρατηγικής και θα

[1] Ένας προσδιοριστής προστασίας μπορεί να θεωρηθεί σαν τον Αριθμό Ταυτότητας, υπό την έννοια ότι η αποκάλυψη του θα μπορεί να σκιαγραφήσει μοναδικά ένα άτομο. Θα δοθεί ακριβής ορισμός αυτού στην ενότητα 3.1

παρουσιαστούν συμπεράσματα και κάποιες αναφορές για μελλοντική εργασία.

2. Σχετική βιβλιογραφία

Υπάρχει μία πληθώρα διαφορετικών τεχνικών, οι οποίες έχουν δημιουργηθεί με στόχο την προστασία της ιδιωτικότητας των χρηστών και την παραγωγή ανώνυμων δεδομένων. Οι τεχνικές αυτές ποικίλλουν ανάλογα με το είδος της βάσης δεδομένων στην οποία εφαρμόζονται. Στις στατιστικές βάσεις [2] για παράδειγμα, η ανωνυμία των δεδομένων επιτυγχάνεται μέσω της προσθήκης θορύβου. Αν και αυτή η τεχνική δεν επηρεάζει τα στατιστικά αποτελέσματα της βάσης, η εφαρμογή της έχει και κάποια μειονεκτήματα. Αφενός, επηρεάζεται η ακεραιότητα των δεδομένων της βάσης και αφετέρου τα δεδομένα δεν μπορούν να χρησιμοποιηθούν πλέον για τη διεξαγωγή χρήσιμων πληροφοριών (π.χ. κανόνες συσχέτισης) λόγω της προσθήκης θορύβου.

Ένας διαφορετικός αλγόριθμος, ο οποίος αφορά σχεσιακές βάσεις δεδομένων, είναι αυτός της k-ανωνυμίας (όπου k μία σταθερά), ο οποίος προτάθηκε για πρώτη φορά από τις Samarati και Sweeney στο [3]. Μία βάση λέγεται ότι είναι *k-ανώνυμη*, αν και μόνο αν κάθε ακολουθία γνωρισμάτων στον προσδιοριστή προστασίας εμφανίζεται στη βάση δεδομένων τουλάχιστον k φορές. Ο ορισμός της *k-ανωνυμίας* είναι αυτός που θα χρησιμοποιηθεί και στη παρούσα εργασία, αφού πρώτα πραγματοποιηθούν οι απαραίτητες αλλαγές, δεδομένου ότι δεν αναφερόμαστε σε σχεσιακές αλλά σε χωρο-χρονικές βάσεις δεδομένων.

Η *γενίκευση* και η *καταστολή* (suppression) των δεδομένων, υπήρξαν δύο από τις πιο διαδεδομένες τεχνικές που αναπτύχθηκαν με στόχο τα δεδομένα που εξάγονται από τη βάση όπου είναι αποθηκευμένοι οι χρήστες, να είναι ανώνυμα. Τις τεχνικές αυτές χρησιμοποίησαν και οι Samarati και Sweeney [3], όταν πρότειναν το μοντέλο της k-ανωνυμίας. Η Samarati μάλιστα στο [10], πρότεινε έναν αλγόριθμο δυαδικής αναζήτησης, για την εύρεση της ελάχιστης γενίκευσης που απαιτείται προκειμένου να επιτευχθεί k-ανωνυμία. Ένας θεωρητικός αλγόριθμος (MinGen) που επίσης επιτυγχάνει χρήση ελάχιστης γενίκευσης, προτάθηκε από τη Sweeney στο [11].

Βασιζόμενοι στην παρατήρηση ότι οι τεχνικές της γενίκευσης και της καταστολής δεν μπορούν να εφαρμοστούν πάντοτε επιτυχώς (για παράδειγμα

σε κατηγορικά δεδομένα), οι Domingo-Ferrer και Torra [12], πρότειναν μία διαφορετική προσέγγιση η οποία βασίζεται στην ετερογενή και στη συνεχή *μικρο-ολοκλήρωση* (microaggregation). Παρόλο που αυτή η προσέγγιση επιτυγχάνει βέλτιστα αποτελέσματα όταν εφαρμόζεται σε κατηγορικά και ονομαστικά δεδομένα, εντούτοις υπάρχει μία σαφής μείωση της απόδοσης όταν αναφερόμαστε σε άλλους τύπους δεδομένων. Επιπλέον, επειδή η προσέγγιση είναι άπληστη, απαιτείται πολύς χρόνος και μεγάλο κόστος για την υλοποίηση της.

Στο [13] οι Bayardo και Agrawal, πρότειναν μια τεχνική η οποία χρησιμοποιεί το μονοδιάστατο μοντέλο τεμαχισμού των γνωρισμάτων (τα οποία πρέπει να έχουν τοποθετηθεί σε μια διατεταγμένη σειρά). Η συγκεκριμένη τεχνική, αν και επιτυγχάνει βέλτιστη λύση ως προς το κόστος, έχει αποδειχθεί ότι παρέχει λιγότερο αποδοτικά αποτελέσματα από τις τεχνικές που χρησιμοποιούν το πολυδιάστατο μοντέλο. Την παρατήρηση αυτή εκμεταλλεύτηκαν οι LeFevre, DeWitt και Ramakrishnan [14], οι οποίοι χρησιμοποίησαν το πολυδιάστατο μοντέλο δεδομένων και ανέπτυξαν το σύστημα Incognito. Το Incognito χρησιμοποιεί δύο βασικές ιδέες του δυναμικού προγραμματισμού: την από κάτω-προς-τα-πάνω άθροιση κατά μήκος των ιεραρχιών κάθε διάστασης και τον a-priori υπολογισμό του αθροίσματος.

Οι παραπάνω τεχνικές (και πολλές άλλες οι οποίες δεν αναφέρονται σε αυτή τη σύντομη παρουσίαση) επιτυγχάνουν k-ανωνυμία, αλλά η δομή τους είναι τέτοια ώστε να μπορούν να εφαρμοστούν μόνο σε σχεσιακές βάσεις αντί σε βάσεις οι οποίες δύναται να μεταβάλλονται με τη πάροδο του χρόνου. Στη παρούσα εργασία θέλουμε το μοντέλο k-ανωνυμίας να εφαρμοστεί σε χωρο-χρονικά δεδομένα, τα οποία μεταβάλλονται δυναμικά στον χρόνο. Συνεπώς, εξαιτίας της φύσης του προβλήματος που καλούμαστε να επιλύσουμε, οι περισσότερες από τις παραπάνω τεχνικές είναι ακατάλληλες. Μία τεχνική η οποία μπορεί να εφαρμοστεί και σε χωρο-χρονικά δεδομένα (πραγματοποιώντας κάποιες αλλαγές) είναι αυτή της γενίκευσης, η οποία και στη γενική της μορφή θα χρησιμοποιηθεί στη παρούσα εργασία.

Μια προσέγγιση που χρησιμοποιεί χωρικά δεδομένα και η οποία εν μέρει μόνο χρησιμοποιεί χρονικά δεδομένα, είναι αυτή που παρουσιάζεται στο [15]. Οι αλγόριθμοι και η αρχιτεκτονική που προτείνονται, γενικεύουν άλλοτε

τα χωρικά και άλλοτε τα χρονικά δεδομένα, με στόχο την ικανοποίηση των περιορισμών ανωνυμίας που ορίζει ο ιδιοκτήτης του δικτύου. Για να επιτευχθεί ο παραπάνω στόχος χρησιμοποιούνται αισθητήρες κίνησης και υλικό χαρτογράφησης.

Η παραπάνω προσέγγιση επεκτάθηκε στο [16], ώστε η τιμή του k να μπορεί να μεταβάλλεται σε κάθε αίτηση. Επιπλέον προτάθηκε ένας νέος ορισμός για την k-ανωνυμία. Ο ορισμός αυτός είναι διαφορετικός από αυτόν που ορίστηκε στο [15] όπως και από αυτόν που θα χρησιμοποιηθεί στη παρούσα εργασία. Για την ακρίβεια στο [16] έχει θεωρηθεί ότι όλοι οι χρήστες (που αποτελούν μέλη του συνόλου ανωνυμίας) θα πρέπει να έχουν στείλει μία αίτηση την ίδια χρονική στιγμή και από την ίδια περιοχή όπου βρισκόταν ο αιτών. Η παραπάνω απαίτηση είναι ιδιαίτερα περιοριστική και γι' αυτόν τον λόγο δεν θα χρησιμοποιηθεί στην παρούσα εργασία. Αντίθετα, όπως και στο [3], αρκεί τα μέλη του συνόλου ανωνυμίας να είναι χρήστες οι οποίοι είχαν περάσει κάποια χρονική στιγμή από το σημείο όπου βρισκόταν ο αιτών όταν έκανε την αίτηση.

Οι δύο προηγούμενες προσεγγίσεις ([15], [16]) πραγματεύονται μία ειδική περίπτωση του προβλήματος που αναφέρεται στο [3] και το οποίο μας απασχολεί και στη παρούσα εργασία. Για την ακρίβεια οι δύο πρώτες προσεγγίσεις θεωρούν ότι κάθε τοποθεσία παριστάνει έναν προσδιοριστή προστασίας. Συνεπώς, θα πρέπει οπουδήποτε και αν βρίσκεται ο αιτών, να ελέγχεται αν υπάρχουν και άλλοι k χρήστες. Όμως αυτή η υπόθεση είναι ιδιαίτερα περιοριστική και είναι όμοια με το να θεωρούσαμε ότι σε κάθε τοποθεσία υπάρχει μία εξωτερική πηγή (όπως π.χ. μία κάμερα), η οποία μπορεί να αναγνωρίσει ανά πάσα στιγμή, όλους τους χρήστες που υπάρχουν σε αυτήν. Αντίθετα στο [3], μία τοποθεσία θεωρείται σαν προσδιοριστής προστασίας μόνο όταν ανήκει σε κάποιες από τις περιοχές που συνηθίζει να συχνάζει ο αιτών. Συνεπώς, κάθε χρήστης θα έχει τους δικούς του προσδιοριστές προστασίας, οι οποίοι θα προκύπτουν από τις δικές του συνήθειες μετακίνησης. Οι θεωρήσεις που έγιναν στο [3], όπως και η στρατηγική που προτάθηκε, θα αποτελέσουν το βασικό υπόβαθρο για την παρούσα εργασία.

3. Διατύπωση του προβλήματος και ορισμοί

Η απόκρυψη των ευαίσθητων πληροφοριών που αφορούν τον αιτούντα μπορεί να επιτευχθεί μέσω μιας κατάλληλης στρατηγικής, η οποία θα παρουσιαστεί στο Κεφάλαιο 4. Στο παρόν κεφάλαιο θα παρουσιαστούν οι βασικές δομές τις οποίες χρησιμοποιεί και στις οποίες βασίζεται η προτεινόμενη στρατηγική, και θα δοθούν κατάλληλοι ορισμοί ώστε να γίνει ευκολότερη η κατανόηση του προβλήματος.

3.1 Προσδιοριστές Προστασίας Βασιζόμενοι στη Τοποθεσία (Location Based Quasi-Identifiers)

Στις σχεσιακές βάσεις, όπου μεταξύ άλλων αποθηκεύονται και προσωπικά δεδομένα (όπως για παράδειγμα ο Αριθμός Ταυτότητας, το Ονοματεπώνυμο κτλ), είναι ιδιαίτερα σημαντικό να διασφαλιστεί ότι ακόμη και αν τα υπο-δημοσίευση δεδομένα συσχετιστούν με δεδομένα από άλλες πηγές (άλλες βάσεις δεδομένων, δημοτικά αρχεία, τηλεφωνικούς καταλόγους κτλ), δεν θα είναι δυνατό να προσδιοριστούν μεμονωμένα άτομα. Μειώνοντας την πιθανότητα ύπαρξης συσχετισμού, επιτυγχάνουμε προστασία της ιδιωτικότητας των ατόμων των οποίων τα στοιχεία είναι αποθηκευμένα στην υπο-μελέτη βάση. Το σύνολο των γνωρισμάτων της βάσης, τα οποία όταν συνδυαστούν με δεδομένα από άλλες πηγές μπορούν να προσδιορίσουν συγκεκριμένα άτομα, ονομάζεται *προσδιοριστής προστασίας* (quasi-identifiers) [3]. Ένας προσδιοριστής προστασίας μπορεί να αποτελείται από ένα και μόνο γνώρισμα όπως για παράδειγμα ο ΑΦΜ ή ο αριθμός ταυτότητας, αλλά και από περισσότερα του ενός γνωρίσματα, τα οποία όταν συνδυαστούν μπορούν να προσδιορίσουν συγκεκριμένα άτομα (π.χ. ο συνδυασμός της ημερομηνίας γεννήσεως, του ταχυδρομικού κώδικα και του φύλου).

Ένας διαφορετικού τύπου προσδιοριστής προστασίας, ο οποίος ονομάζεται *Προσδιοριστής Προστασίας βασιζόμενος στη Τοποθεσία* (Location Based Quasi-Identifier) παρουσιάστηκε στο [4]. Από εδώ και στο εξής οι Προσδιοριστές Προστασίας βασιζόμενοι στη Τοποθεσία, θα αναφέρονται σαν LBQID. Ένα LBQID, σε αντίθεση με τους παραδοσιακούς προσδιοριστές προστασίας, δεν αποτελείται από συγκεκριμένα γνωρίσματα μιας βάσης δεδομένων. Αντίθετα, χρησιμοποιείται για να αναπαραστήσει χωρο-χρονικά

πρότυπα (patterns) μετακίνησης που συνηθίζουν να ακολουθούν οι αιτούντες. Με τον όρο χωρο-χρονικά πρότυπα μετακίνησης, εννοούμε συγκεκριμένες (χωρικές) διαδρομές που συνηθίζει να ακολουθεί ο χρήστης σε συγκεκριμένες χρονικές στιγμές. Στόχος των LBQIDs είναι η προστασία της ταυτότητας των χρηστών οι οποίοι στέλνουν αιτήσεις για παροχή υπηρεσιών. Μέσω του παραδείγματος 3.1 θα προσπαθήσουμε να εξηγήσουμε τι ακριβώς είναι ένα LBQID.

Παράδειγμα 3.1: Παράδειγμα ενός LBQID

Ας θεωρήσουμε έναν φοιτητή, ο οποίος πραγματοποιεί τη διαδρομή από το σπίτι του στο πανεπιστήμιο όπου φοιτά, και το αντίστροφο (δηλαδή από το πανεπιστήμιο στο σπίτι του), όποτε αυτός έχει μάθημα. Η παραπάνω διαδρομή μπορεί να θεωρηθεί σαν ένα LBQID, αν πραγματοποιείται τουλάχιστον 4 φορές την εβδομάδα και για τουλάχιστον 3 εβδομάδες.

Στο παραπάνω παράδειγμα, παρατηρούμε ότι είναι απαραίτητο να τηρούνται κάποιοι χρονικοί περιορισμοί προκειμένου να θεωρηθεί μία διαδρομή που ακολουθεί ένας χρήστης, ως LBQID. Αυτοί οι περιορισμοί εγγυώνται ότι οι τοποθεσίες που αποτελούν τμήμα ενός LBQID (τέτοιες είναι το πανεπιστήμιο και το σπίτι στο Παράδειγμα 3.1), όντως ανήκουν σε κάποιο από τα πρότυπα μετακίνησης του χρήστη και δεν είναι μέρη από τα οποία απλά έτυχε να περάσει κάποια στιγμή ο συγκεκριμένος χρήστης. Στο προηγούμενο παράδειγμα, ορίζοντας ότι η διαδρομή Σπίτι-Πανεπιστήμιο (και το αντίστροφο) πρέπει να εκτελεστεί από το χρήστη τουλάχιστον 4 φορές σε μία εβδομάδα και για τουλάχιστον 3 εβδομάδες, διασφαλίζουμε ότι ο χρήστης είναι φοιτητής (εναλλακτικά θα μπορούσε να εργάζεται στο Πανεπιστήμιο) και ότι δεν είναι απλά κάποιος που έτυχε να εκτελέσει αυτή τη διαδρομή. Χωρίς την ύπαρξη χρονικών περιορισμών, αυτόματα όλες οι διαδρομές που εκτελούσαν οι χρήστες θα θεωρούνταν σαν LBQIDs. Άλλωστε, δεν μας ενδιαφέρουν μετακινήσεις που εκτελούνται μόνο μία φορά (ή σπάνια), γιατί αυτού του είδους οι διαδρομές δεν μπορούν να αποκαλύψουν ευαίσθητες πληροφορίες για τον αιτούντα. Αν για παράδειγμα αποκαλυφθεί για κάποιον χρήστη (μέσω αιτήσεων που αυτός αποστέλλει) ότι συνηθίζει να επισκέπτεται κάθε μήνα ένα νοσοκομείο, τότε μπορούμε να συμπεράνουμε ότι έχει κάποιο πρόβλημα

υγείας. Σε αυτή τη περίπτωση κρίνεται αναγκαίο να αποκρύψουμε αυτή τη πληροφορία, γιατί κινδυνεύει η ιδιωτικότητα αυτού του χρήστη. Αντίθετα, αν αυτού του είδους οι επισκέψεις πραγματοποιούνται μία φορά τον χρόνο, δεν είναι αναγκαίο να τις αποκρύψουμε. Αυτό οφείλεται στο γεγονός ότι οι συγκεκριμένες επισκέψεις δεν αποκαλύπτουν κάποια ευαίσθητη πληροφορία σχετική με τον χρήστη, αλλά ούτε μπορούν να διασυνδεθούν με προηγούμενες επισκέψεις που αυτός μπορεί να έχει πραγματοποιήσει (το πιο πιθανό είναι αυτές οι επισκέψεις να είναι επισκέψεις ρουτίνας).

Δεδομένης της φύσης των LBQID (παριστάνουν διαδρομές που συνηθίζει να ακολουθεί ο αιτών σε συγκεκριμένες χρονικές στιγμές), ένας χρήστης είναι δυνατό να έχει ένα ή περισσότερα LBQIDs. Ο αριθμός αυτών εξαρτάται από τα χωρο-χρονικά πρότυπα μετακίνησης που συνηθίζει να ακολουθεί ο χρήστης. Ένας πιθανός τρόπος εξαγωγής των LBQIDs μπορεί να είναι μέσω της στατιστικής ανάλυσης. Ένας εναλλακτικός τρόπος θα ήταν να θεωρήσουμε ότι οι ίδιοι οι χρήστες ορίζουν τα LBQIDs τους, βάσει των μετακινήσεων τους. Ωστόσο, επειδή στα πλαίσια αυτής της εργασίας θεωρήσαμε ότι τα LBQIDs αποθηκεύονται σε έναν έμπιστο εξυπηρετητή (ακριβής ορισμός της προτεινόμενης αρχιτεκτονικής παρουσιάζεται στην Ενότητα 4.1), ο οποίος διατηρεί και ιστορικό με τις μετακινήσεις των χρηστών, είναι πιο λογικό να ορίσουμε ότι αυτός είναι και υπεύθυνος για την εξαγωγή των LBQID κάθε χρήστη. Στη συνέχεια παρουσιάζεται ένας ακριβής ορισμός για το τι είναι ένα LBQID.

Ορισμός 3.1. Ένα LBQID είναι ένα χωρο-χρονικό πρότυπο μετακίνησης το οποίο αποτελείται από μία ακολουθία χωρο-χρονικών στοιχείων και από έναν τύπο επανάληψης (recurrence formula). Κάθε χωρο-χρονικό στοιχείο είναι ένα ζεύγος της μορφής: <*Περιοχή, Χρονόσημο*>.

Στον προηγούμενο ορισμό, καθένα από τα χωρο-χρονικά στοιχεία αναπαριστά μία τοποθεσία την οποία (δεδομένων και κάποιων χρονικών περιορισμών) συνηθίζει να επισκέπτεται ο αιτών. Για την ακρίβεια, στο ζεύγος <*Περιοχή, Χρονόσημο*>, η *Περιοχή* παριστάνει μία συγκεκριμένη τοποθεσία η οποία ορίζεται από ένα ζεύγος σημείων και αναπαρίστανται σαν: $[(x_1,y_1), (x_2,y_2)]$. Επειδή θεωρούμε ότι κάθε χωρική περιοχή είναι τετράγωνη, τα δύο

αυτά σημεία είναι επαρκή για να την περιγράψουν. Το σημείο (x_1,y_1) παριστάνει το κάτω αριστερό άκρο της περιοχής και το σημείο (x_2,y_2) το άνω δεξί άκρο. Το *Χρονόσημο* παριστάνει εκείνο το χρονικό διάστημα στο εύρος μίας μέρας εντός του οποίου ο χρήστης μπορεί να βρεθεί στην *Περιοχή*. Αναπαρίσταται σαν ένα ζεύγος $[t_1,t_2]$, όπου τα t_1,t_2 είναι χρονικές στιγμές που μπορούν να ορίζονται σε ώρες, λεπτά ή ακόμη και δευτερόλεπτα. Ένα παράδειγμα ενός *Χρονόσημου* είναι το *[15:00 , 17:00]*, όπου τα 15:00, 17:00 παριστάνουν ώρες μίας ημέρας. Ο λόγος που δεν προσδιορίζονται σαφείς ημερομηνίες στα *Χρονόσημα* είναι επειδή με αυτόν τον τρόπο επιτυγχάνουμε να απεικονίσουμε οποιαδήποτε ημέρα του χρόνου, για οποιονδήποτε χρόνο. Ας θεωρήσουμε για παράδειγμα το χωρο-χρονικό στοιχείο *<Σπίτι, [09:00, 12:00]>.* Το *Χρονόσημο* [09:00, 12:00] δηλώνει ότι ο χρήστης (τον οποίο αφορά το συγκεκριμένο στοιχείο) μπορεί να βρεθεί στο Σπίτι του οποιαδήποτε ώρα μεταξύ (συμπεριλαμβανομένου) των 09:00 και 12:00. Μη δηλώνοντας ρητά την ημερομηνία, μας επιτρέπεται να πούμε ότι ο χρήστης μπορεί οποιαδήποτε ημέρα να βρεθεί σπίτι του τις συγκεκριμένες ώρες.

Ο *τύπος επανάληψης* ενός LBQID σχετίζεται με όλα τα χωρο-χρονικά στοιχεία αυτού του LBQID. Διαισθητικά, ο τύπος αυτός αποτελεί ένα είδος χρονικού περιορισμού. Αν ένας χρήστης εκτελεί μία συγκεκριμένη διαδρομή ή παραμένει στάσιμος σε κάποιο μέρος τόσες φορές όσες ορίζει ο τύπος επανάληψης, τότε αυτό σημαίνει ότι η συγκεκριμένη διαδρομή (ή τοποθεσία) αποτελεί ένα LBQID. Η σύνταξη που έχει ο *τύπος επανάληψης* είναι η ακόλουθη:

$$rec_1.G_1 * rec_2.G_2 * ... * rec_n.G_n$$

όπου το rec_i είναι ένας θετικός ακέραιος για κάθε i = 1,2,…n, και το G_i παριστάνει μία διαβάθμιση του χρόνου (granularity) [5]. Ουσιαστικά το G_i παριστάνει κάποιο χρονικό διάστημα το οποίο μπορεί να είναι ορισμένο σε ημέρες, εβδομάδες, μήνες κοκ. Η ερμηνεία που έχει ο παραπάνω τύπος είναι η εξής: η ακολουθία χωρο-χρονικών στοιχείων του LBQID πρέπει να εμφανίζεται τουλάχιστον rec_1 φορές εντός του χρονικού διαστήματος G_1. Όλες οι παρατηρήσεις (τουλάχιστον rec_1 σε πλήθος) που έγιναν εντός του διαστήματος G_1, πρέπει να εμφανιστούν τουλάχιστον rec_2 φορές εντός ενός στιγμιοτύπου του G_2. Συνεχίζοντας με αυτόν τον τρόπο, επεκτείνουμε το

παραπάνω σκεπτικό μέχρι να φτάσουμε στην τιμή *n*. Από τα παραπάνω είναι προφανές ότι αν η ακολουθία χωρο-χρονικών στοιχείων εμφανίζεται rec_{i-1} φορές στο διάστημα $G_{i-1,}$ τότε θα πρέπει σε κάθε στιγμιότυπο του G_i να εμφανίζεται τουλάχιστον rec_i φορές.

Δεδομένου του τρόπου ορισμού του *τύπου επανάληψης,* οποιαδήποτε υπο-έκφραση της μορφής $1.G_i$, η οποία δηλώνει ότι η ακολουθία χωρο-χρονικών στοιχείων πρέπει να εμφανιστεί εντός του υπο-διαστήματος G_i τουλάχιστον *1* φορά, είναι αυτονόητη και μπορεί να παραληφθεί. Εξάλλου, αν η ακολουθία δεν εμφανιστεί καθόλου τότε δεν έχει νόημα να αποτελεί τμήμα κάποιου LBQID. Τέλος, αν ο *τύπος επανάληψης* είναι κενός, θα υπονοείται ότι η ακολουθία με τα χωρο-χρονικά στοιχεία του LBQID πρέπει να εμφανιστεί μία φορά οποιαδήποτε χρονική στιγμή.

Θα προσπαθήσουμε μέσω ενός παραδείγματος να κάνουμε περισσότερο κατανοητή την έννοια του τύπου επανάληψης.

Παράδειγμα 3.2: Το LBQID του Παραδείγματος 3.1, μπορεί να οριστεί (βάση όσων ειπώθηκαν παραπάνω) ως εξής:
*<Σπίτι [08:00 , 09:00], Πανεπιστήμιο [09:00 , 10:00]>, Πανεπιστήμιο [13:00 , 14:00], Σπίτι [14:00 , 15:00]> ,4.Ημέρες * 3.Εβδομάδες*

Στην προηγούμενη έκφραση ο *τύπος επανάληψης* είναι ο: *4.Ημέρα * 3.Εβδομάδα*. Οι υπο-εκφράσεις «*Ημέρα*» και «*Εβδομάδα*», παριστάνουν τις διαβαθμίσεις του χρόνου G_i, ενώ οι αριθμοί 4 και 3 είναι αντίστοιχα τα rec_1 και rec_2. Ο συγκεκριμένος τύπος ορίζει ότι εάν η διαδρομή σπίτι-πανεπιστήμιο και το αντίστροφο εκτελεστεί κάποια ημέρα από τον χρήστη, τότε θα πρέπει να εκτελεστεί τουλάχιστον άλλες 3 φορές εντός της εβδομάδας, και για τουλάχιστον άλλες 2 εβδομάδες.

Στο παραπάνω παράδειγμα, ο *τύπος επανάληψης* ορίστηκε με τη μορφή «τουλάχιστον 4 φορές σε μία εβδομάδα και για τουλάχιστον 3 εβδομάδες». Ωστόσο αυτός ο τρόπος ορισμού δεν είναι μοναδικός. Ένας διαφορετικός τρόπος ορισμού του *τύπου επανάληψης* θα μπορούσε να είναι «την ίδια ημέρα για τουλάχιστον 2 εβδομάδες». Σε αυτή τη περίπτωση, τα στιγμιότυπα για τον χρόνο θα είναι μέρες της εβδομάδας και θα

αναπαρίστανται με ονόματα για τις συγκεκριμένες ημέρες όπως για παράδειγμα Δευτέρα ή Τρίτη. Ανεξαρτήτως του τρόπου που θα οριστεί ο *τύπος επανάληψης,* αυτό που έχει σημασία είναι να μπορεί να γίνει κατάλληλος χειρισμός αυτού. Για παράδειγμα, ένας *τύπος επανάληψης* της μορφής: «για τουλάχιστον 3 ζυγές μέρες σε τουλάχιστον 4 μονούς μήνες», αφενός δεν θα είχε νόημα, αφετέρου θα ήταν πολύ δύσκολο να υπολογιστεί.

Έχοντας πλέον ορίσει τι είναι ένα LBQID, μπορούμε να θεωρήσουμε τον ακόλουθο γενικό τύπο για την αναπαράσταση του:

$$< E_1, E_2, ...E_n >, rec_1.G_1 * rec_2.G_2 * ... * rec_n.G_n$$

Η ακολουθία $< E_1, E_2, ...E_n >$, παριστάνει την τα χωρο-χρονικά στοιχεία του LBQID, ενώ η ακολουθία $rec_1.G_1 * rec_2.G_2 * ... * rec_n.G_n$ χρησιμοποιείται για να αναπαραστήσει τον *τύπο επανάληψης* αυτού. Όπως προαναφέρθηκε, κάθε στοιχείο E_i είναι ένα ζεύγος της μορφής: <*Περιοχή, Χρονόσημο*>.

Οι ακόλουθοι ορισμοί επεξηγούν πότε μία αίτηση ή ένα σύνολο αιτήσεων «ταιριάζουν» με κάποια από τα στοιχεία ενός LBQID.

Ορισμός 3.2. Αν ένας χρήστης αποστέλλει την αίτηση r_i τη χρονική στιγμή t_i από το σημείο (x,y), λέμε ότι αυτή η αίτηση *ταιριάζει* με το στοιχείο E_j κάποιου από τα LBQID του, αν και μόνο αν (x,y)$\in$*Περιοχή* και $t_i \in$*Χρονόσημο*, όπου <*Περιοχή, Χρονόσημο*>$\in E_j$.

Παράδειγμα 3.3: Παράδειγμα ταιριάσματος αίτησης με το στοιχείο ενός LBQID.

Ας θεωρήσουμε έναν χρήστη ο οποίος στις 08/07/2006, στις 08:30 αποστέλλει μία αίτηση από το σπίτι του. Έστω ότι ένα από τα LBQIDs αυτού του χρήστη, ορίστηκε στο Παράδειγμα 3.2. Η συγκεκριμένη αίτηση ταιριάζει με το πρώτο στοιχείο αυτού του LBQID, δηλαδή με το <*Σπίτι [08:00,09:00]*>, καθώς η *Περιοχή* αυτού του στοιχείου είναι η ίδια με την περιοχή από όπου πραγματοποιήθηκε η αίτηση και 08:30 $\in$ *Χρονόσημο* αυτού του στοιχείου, δηλαδή στο *[08:00,09:00].*

Ορισμός 3.3. Ένα σύνολο αιτήσεων R ταιριάζει με ένα LBQID Q, αν και μόνο αν ισχύουν όλες οι παρακάτω συνθήκες: (1) κάθε αίτηση $r_i \in$ R, ταιριάζει με ένα στοιχείο E_j του LBQID Q και το αντίστροφο και (2) αν t_i είναι η χρονική στιγμή όπου στάλθηκε η αίτηση r_i, τότε θα πρέπει το σύνολο των t_i για όλες τις αιτήσεις $r_i \in$ R να ικανοποιεί τον *τύπο επανάληψης* του Q.

Παράδειγμα 3.4: Παράδειγμα ταιριάσματος ενός συνόλου αιτήσεων με ένα LBQID

Ας θεωρήσουμε ξανά τον χρήστη του Παραδείγματος 3.3, του οποίου ένα LBQID είναι αυτό που ορίστηκε στο Παράδειγμα 3.2. Θεωρούμε ότι αυτός ο χρήστης έστειλε σε διάφορους παροχείς υπηρεσιών ένα σύνολο από αιτήσεις R. Ακολούθως καταγράφονται η ημερομηνία, η ώρα και το σημείο από όπου εστάλη κάθε μία αίτηση που ανήκει στο σύνολο R:

Αίτηση	Ημερομηνία	Ώρα	Σημείο
r_1	08/08/2006	08:30	Σπίτι
r_2	08/08/2006	09:30	Πανεπιστήμιο
r_3	08/08/2006	14:00	Σπίτι
r_4	08/08/2006	14:30	Πανεπιστήμιο
r_5	09/08/2006	08:00	Σπίτι
r_6	09/08/2006	09:20	Πανεπιστήμιο
r_7	09/08/2006	13:15	Σπίτι
r_8	09/08/2006	14:45	Πανεπιστήμιο

Παρατηρούμε ότι η πρώτη συνθήκη του ορισμού 3.3 είναι αληθής καθώς καθεμία από τις παραπάνω αιτήσεις ταιριάζει και με ένα στοιχείο του LBQID. Για την ακρίβεια οι αιτήσεις r_1, r_5 ταιριάζουν με το στοιχείο <Σπίτι [08:00, 09:00]>, οι r_2, r_6 με το στοιχείο <Πανεπιστήμιο [09:00, 10:00]>, οι r_3, r_7 με το <Πανεπιστήμιο [13:00, 14:00]> και οι αιτήσεις r_4, r_8 με το στοιχείο <Σπίτι [14:00, 15:00]>. Ωστόσο, η δεύτερη συνθήκη δεν είναι αληθής, καθώς ο χρήστης1 έστειλε αιτήσεις για 2 ημέρες μέσα σε μία εβδομάδα (έστειλε τις

r_1, r_2, r_3, r_4 στις 08/08/2006 και τις r_5, r_6, r_7, r_8 στις 09/09/2006). Για να ήταν αληθής η δεύτερη συνθήκη θα έπρεπε ο *τύπος επανάληψης* να είναι της μορφής: *2.Ημέρες * 1.Εβδομάδα*. Αφού δεν είναι και οι δύο συνθήκες αληθείς, το σύνολο R δεν ταιριάζει με το συγκεκριμένο LBQID.

3.2 Προσωπικό Ιστορικό Τοποθεσιών (Personal History of Locations)

Στον έμπιστο εξυπηρετητή (ακριβής ορισμός αυτού θα δοθεί στην ενότητα 4.1), εκτός από τα LBQIDs όλων των χρηστών, αποθηκεύονται και τα Προσωπικά Ιστορικά Τοποθεσιών αυτών (από εδώ και στο εξής θα αναφέρονται σαν PHLs). Στα πλαίσια αυτής της εργασίας, η δομή που θα χρησιμοποιηθεί για να αναπαρασταθούν τα PHLs θα είναι ίδια με αυτή που παρουσιάστηκε στο [4], όπου και αυτά ορίστηκαν για πρώτη φορά.

Κάθε χρήστης διαθέτει ένα PHL το οποίο είναι μία ακολουθία χωρο-χρονικών στοιχείων. Κάθε στοιχείο περιέχει ένα συγκεκριμένο σημείο, όπως και τη χρονική στιγμή όπου ο χρήστης (στον οποίο ανήκει το PHL) επισκέφτηκε το συγκεκριμένο στοιχείο. Τα PHLs όλων των χρηστών αποθηκεύονται σε μία κατάλληλα διαμορφωμένη βάση δεδομένων, η οποία υπάρχει στον έμπιστο εξυπηρετητή. Δεδομένου ότι η συγκεκριμένη βάση έχει περιορισμένη χωρητικότητα, θα πρέπει στον έμπιστο εξυπηρετητή να υπάρχει και ένας κατάλληλος μηχανισμός ο οποίος θα ελέγχει τη χωρητικότητα της βάσης ανά τακτά χρονικά διαστήματα. Όποτε το μέγεθος της βάσης ξεπεράσει κάποιο άνω όριο, θα πρέπει τα PHLs όλων των χρηστών να διαγράφονται και να ξεκινάει εξαρχής η καταγραφή τους.

Για να είναι δυνατή η καταγραφή του PHL κάθε χρήστη, θα πρέπει ο έμπιστος εξυπηρετητής να «παρακολουθεί» τις κινήσεις των χρηστών (οι οποίοι αποστέλλουν αιτήσεις σε διάφορους παροχείς υπηρεσιών) και να τις καταγράφει ανά τακτά χρονικά διαστήματα στη βάση του. Αυτή η «παρακολούθηση» μπορεί να γίνει εντοπίζοντας το σήμα που εκπέμπεται από τις διάφορες ασύρματες συσκευές που χρησιμοποιεί ο εκάστοτε χρήστης.

Ορισμός 3.4. Το PHL κάθε χρήστη περιέχει μία ακολουθία τρισδιάστατων στοιχείων (δύο διαστάσεις για το χώρο και μία για τον χρόνο), όπου κάθε στοιχείο είναι μία τριάδα της μορφής: (x,y,t). Το ζεύγος (x,y) είναι οι

συντεταγμένες του διδιάστατου σημείου όπου βρισκόταν ο χρήστης τη χρονική στιγμή t.

Στον παραπάνω ορισμό, η χρονική στιγμή t θα είναι της μορφής: <Ημερομηνία, Ώρα: Λεπτά: Δευτερόλεπτα>, δηλαδή θα είναι μία λεπτομερής περιγραφή της χρονικής στιγμής όπου έγινε η καταγραφή. Θα πρέπει να τονίσουμε ότι η καταγραφή των στοιχείων ενός PHL γίνεται ανά τακτά χρονικά διαστήματα και ανεξαρτήτως του αν ο χρήστης (στον οποίο ανήκει το PHL) έχει στείλει τη δεδομένη χρονική στιγμή κάποια αίτηση. Είναι συνεπώς κατανοητό, ότι για όλες τις αιτήσεις που αποστέλλει ένας χρήστης, θα υπάρχει και μία εγγραφή στο PHL του. Το αντίστροφο όμως δεν ισχύει, δηλαδή δεν θα πρέπει για κάθε στοιχείο του PHL ενός χρήστη να υπάρχει και κάποια αντίστοιχη αίτηση. Ο λόγος που ορίστηκε η παραπάνω σχέση μεταξύ αιτήσεων και PHL, είναι γιατί το σύνολο ανωνυμίας κάθε αιτούντα δεν χρειάζεται να αποτελείται μόνο από χρήστες οι οποίοι θα πρέπει να είχαν στείλει την ίδια χρονική στιγμή με τον αιτούντα, μία αίτηση από το ίδιο σημείο (η πιθανότητα να βρεθούν k-1 τέτοιοι χρήστες είναι πολύ μικρή). Αρκεί, τα μέλη του συνόλου ανωνυμίας να είχαν περάσει κάποια χρονική στιγμή από το συγκεκριμένο σημείο, από όπου και θα μπορούσαν να είχαν αποστείλει κάποια αίτηση.

Ορισμός 3.5. Το PHL ενός χρήστη είναι *χωρο-χρονικά συνεπές* με ένα σύνολο αιτήσεων $R = \{r_1, r_2, ..., r_n\}$, αν και μόνο αν ισχύει μία από τις παρακάτω συνθήκες:

(1) για κάθε αίτηση $r_j \in R$ υπάρχει και ένα στοιχείο (x_i, y_i, t_i) στο PHL αυτού του χρήστη, τέτοιο ώστε η αίτηση r_j να εστάλη τη χρονική στιγμή t_i από το σημείο (x_i, y_i).

(2) για κάθε αίτηση $r_j \in R$ υπάρχουν δύο στοιχεία (x_i, y_i, t_i) και (x_k, y_k, t_k) στο PHL αυτού του χρήστη, τέτοια ώστε η αίτηση r_j να εστάλη τη χρονική στιγμή $t_i \le t \le t_k$ από ένα σημείο (x, y) τέτοιο ώστε $x_i \le x \le x_k$ και $y_i \le y \le y_k$.

Παράδειγμα 3.5: Έλεγχος αν ένα PHL είναι χωρο-χρονικά συνεπές με ένα σύνολο αιτήσεων.

Ας θεωρήσουμε το ακόλουθο στιγμιότυπο του PHL ενός χρήστη:

< [(100,200), 01/09/2006, 08:30:15],
[(120,200), 01/09/2006, 08:35:15],
[(120,250), 01/09/2006, 08:40:15],
[(200,250), 01/09/2006, 08:45:15] >

Όπως διαφαίνεται και από τις εγγραφές του PHL, θεωρήσαμε ότι οι καταγραφές γίνονται κάθε πέντε λεπτά. Ακολούθως παρατίθενται τρεις αιτήσεις που πραγματοποίησε ο χρήστης την 01/09/2006:

Αίτηση	Ημερομηνία	Ώρα	Σημείο
r_1	01/09/2006	08:35:50	(120,250)
r_2	01/09/2006	08:40:15	(120,230)
r_3	01/09/2006	08:55:00	(200,250)

Παρατηρούμε ότι η αίτηση r_2 ταιριάζει με το τρίτο στοιχείο του PHL, δηλαδή με το [(120,250), 01/09/2006, 08:40:15], καθώς ικανοποιείται η πρώτη συνθήκη του ορισμού 2.5. Η αίτηση r_1 ικανοποιεί τη δεύτερη συνθήκη για τα στοιχεία [(120,200), 01/09/2006, 08:35:15] και [(120,250), 01/09/2006, 08:40:15], καθώς 08:35:15 $\leq$ 08:35:50 $\leq$ 08:40:15 και 200 $\leq$ 230 $\leq$ 250 (η x συντεταγμένη από όπου πραγματοποιήθηκε η αίτηση είναι η ίδια με αυτή των δύο στοιχείων του PHL). Ωστόσο, για την αίτηση r_3 δεν ικανοποιείται καμία από τις δύο συνθήκες που υπάρχουν στον ορισμό 2.5. Συνεπώς το συγκεκριμένο στιγμιότυπο του PHL του συγκεκριμένου χρήστη, δεν είναι χωρο-χρονικά συνεπές με το σύνολο των αιτήσεων R = $\{r_1, r_2, r_3\}$.

3.3 Διασύνδεση (linkability) μεταξύ αιτήσεων και χρηστών

Όπως προαναφέρθηκε και στην Εισαγωγή, οι αιτήσεις που αποστέλλονται στους διάφορους παροχείς υπηρεσιών δεν περιέχουν σαφώς την ταυτότητα του αιτούντα, αλλά αντίθετα περιέχουν κάποιο ψευδώνυμο για αυτόν. Προκειμένου τα ψευδώνυμα να παρέχουν ασφάλεια υψηλού επιπέδου,

έχουμε θεωρήσει ότι το ψευδώνυμο κάθε χρήστη είναι μοναδικό και ότι δεν είναι δυνατόν δύο ή περισσότεροι χρήστες να έχουν το ίδιο ψευδώνυμο.

Ο όρος *διασύνδεση* χρησιμοποιείται για να περιγράψει την συσχέτιση που υπάρχει μεταξύ ενός χρήστη και των αιτήσεων που αυτός μελλοντικά μπορεί να στείλει. Στην πραγματικότητα, παρόλο που τα ψευδώνυμα παρέχουν κάποιο επίπεδο ασφάλειας, η πιθανότητα διασύνδεσης ενός χρήστη με τις μελλοντικές του αιτήσεις, εξακολουθεί να υπάρχει. Μία απλή τεχνική για την εξάλειψη αυτού του προβλήματος θα ήταν η αλλαγή του ψευδωνύμου κάθε χρήστη ανά τακτά χρονικά διαστήματα. Ωστόσο, αυτή η τεχνική αφενός επιλύει μονομερώς το πρόβλημα και αφετέρου απαιτεί πολύ επεξεργαστική ισχύ προκειμένου να πραγματοποιήσει όλες τις αναγκαίες αλλαγές (θα πρέπει να αλλάξει το ψευδώνυμο του χρήστη στη βάση όπου αποθηκεύονται τα PHLs όπως και στα LBQIDs του). Γενικότερα, επειδή υπάρχει μία πληθώρα τεχνικών που μπορεί να χρησιμοποιήσει ένας κακόβουλος χρήστης για να συσχετίσει δύο ή περισσότερες αιτήσεις με έναν συγκεκριμένο αιτούντα, η διαδικασία εύρεσης μίας λύσης στο παραπάνω πρόβλημα είναι μία εξαιρετικά δύσκολη διαδικασία.

Μία σειρά από διαφορετικές τεχνικές, οι οποίες μπορούν να χρησιμοποιηθούν για να ελεγχθεί η ύπαρξη διασύνδεσης μεταξύ ενός χρήστη και διαφορετικών αιτήσεων, έχει παρουσιαστεί στο [6]. Αυτές οι τεχνικές ελέγχουν ανά τακτά χρονικά διαστήματα τη θέση του κάθε χρήστη και χρησιμοποιούν τύπους πιθανοτήτων και μέτρα συσχέτισης για να προβλέψουν αν θα μπορεί να υπάρξει κάποια διασύνδεση μεταξύ μελλοντικών αιτήσεων του ίδιου χρήστη. Σε περίπτωση όπου η πιθανότητα ύπαρξης διασύνδεσης είναι μεγάλη, θα πρέπει να ληφθούν κατάλληλα μέτρα προκειμένου να προστατευτεί η ταυτότητα του αιτούντα (ο οποίος είναι σε κίνδυνο) και να μην κινδυνεύσει η ασφάλεια αυτού, όσον αφορά την ιδιωτικότητα του.

Στη παρούσα εργασία δεν μας ενδιαφέρει να ορίσουμε νέες τεχνικές οι οποίες θα ελέγχουν για την ύπαρξη διασύνδεσης μεταξύ αιτήσεων και χρηστών. Στόχος μας είναι να ορίσουμε μία κατάλληλη στρατηγική, η οποία θα προστατεύει οποιεσδήποτε ευαίσθητες πληροφορίες αφορούν χρήστες υπηρεσιών, ακόμη και όταν υπάρχει πιθανότητα διασύνδεσης. Για τον παραπάνω λόγο, θα θεωρήσουμε ότι ο έμπιστος εξυπηρετητής (στον οποίο

αποστέλλονται οι αιτήσεις προτού σταλούν στους παροχείς υπηρεσιών)[2] διαθέτει μία σειρά από συναρτήσεις (τις οποίες θα συμβολίζουμε με Link) οι οποίες ελέγχουν για την ύπαρξη διασύνδεσης μεταξύ διαφορετικών αιτήσεων του ίδιου χρήστη. Οι συναρτήσεις αυτές έχουν τις ακόλουθες ιδιότητες:

(1) Είναι συμμετρικές, δηλαδή $Link(r_i, r_j) = Link(r_j, r_i)$, όπου r_i και r_j είναι δύο διαφορετικές αιτήσεις

(2) Είναι ανακλαστικές δηλαδή $Link(r_i, r_i) = 1$

Ορισμός 3.6. [4] Ας θεωρήσουμε το σύνολο $R = \{r_1, r_2, .., r_n\}$ το οποίο περιέχει όλες τις αιτήσεις που έχουν σταλεί σε έναν συγκεκριμένο παροχέα υπηρεσιών. Το αποτέλεσμα που επιστρέφει η συνάρτηση $Link : R \times R \rightarrow [0,1]$ παριστάνει τη πιθανότητα δύο διαφορετικές αιτήσεις r_i και r_j να έχουν σταλεί από τον ίδιο χρήστη.

Όσο μεγαλύτερη είναι η τιμή που επιστρέφει η συνάρτηση Link(), τόσο μεγαλύτερη είναι και η πιθανότητα δύο διαφορετικές αιτήσεις να έχουν σταλεί από τον ίδιο χρήστη. Ας θεωρήσουμε δύο διαφορετικές αιτήσεις, τις r_i και r_j οι οποίες έχουν σταλεί από τον ίδιο χρήστη. Αν η τιμή που επιστρέφει η Link() είναι 1, σημαίνει ότι ένας οποιοσδήποτε παρατηρητής θα μπορεί να καταλάβει ότι αυτές οι δύο αιτήσεις έχουν σταλεί από τον ίδιο χρήστη. Αν η τιμή που επιστρέφεται είναι 0 σημαίνει ότι κανείς δεν μπορεί να συσχετίσει τις αιτήσεις r_i και r_j με τον ίδιο χρήστη. Στην δεύτερη περίπτωση, δεν χρειάζεται να ληφθούν κάποια πρόσθετα μέτρα καθώς δεν «διαρρέουν» οποιεσδήποτε πληροφορίες αφορούν τον αιτούντα.

Ορισμός 3.7. [4] Έστω ότι R είναι το σύνολο των αιτήσεων που έχουν σταλεί σε έναν παροχέα υπηρεσιών και ότι $R' \subseteq R$. Λέμε ότι το R' *διασυνδέεται* με πιθανότητα Θ, αν για κάθε ζεύγος αιτήσεων $r_i, r_j \in R'$, υπάρχει μία ακολουθία αιτήσεων $r_{i1}, r_{i2}, ..., r_{ik} \in R'$, με $r_{i1} = r_i$ και $r_{ik} = r_j$, τέτοια ώστε $Link(r_{il}, r_{il+1}) \geq \Theta$ για όλα τα $l = 1, ..., k-1$

[2] Ακριβής ορισμός της προτεινόμενης αρχιτεκτονικής θα δοθεί στην Ενότητα 4.1

Με βάση τον προηγούμενο ορισμό, θα λέμε ότι όλες οι αιτήσεις ενός συνόλου $R' \subseteq R$ (όπου R όλες οι αιτήσεις που έχουν σταλεί σε κάποιο παροχέα υπηρεσιών), έχουν σταλεί από τον ίδιο χρήστη, αν και μόνο αν το R' *διασυνδέεται* με πιθανότητα Θ=1.

3.4. K-Ανωνυμία

Όπως έχουμε αναφέρει και νωρίτερα, στα πλαίσια αυτής της εργασίας, μας ενδιαφέρουν εκείνες οι υπηρεσίες οι οποίες για να παράγουν αποτελέσματα χρησιμοποιούν την πληροφορία για τη θέση και τη χρονική στιγμή όπου βρισκόταν ο χρήστης όταν έκανε την αίτηση. Στόχος μας είναι να διασφαλίσουμε ότι ο αιτών δεν θα μπορεί να ξεχωρίσει από τους k-1 χρήστες που υπάρχουν στο σύνολο ανωνυμίας του[3]. Το σύνολο ανωνυμίας ορίζεται με τρόπο όπως αυτόν του [3]. Το μέγεθος του k προσδιορίζει και το μέγεθος της ανωνυμίας για τον αιτών: όσο μεγαλύτερη είναι η τιμή του k, τόσο μεγαλύτερη είναι και η ανωνυμία που παρέχεται.

Στο [7] προτάθηκε για πρώτη φορά ένα μοντέλο το οποίο χρησιμοποιεί τη θέση του αιτούντα για να επιτύχει ανωνυμία. Στη συνέχεια θα εξηγήσουμε τον τρόπο που λειτουργεί αυτό το μοντέλο. Αν θεωρήσουμε ότι ο αιτών είχε ως χωρο-χρονικές συντεταγμένες την τριάδα (x,y,t) όταν πραγματοποίησε την αίτηση, το μοντέλο αρχικά ορίζει μία χωρική περιοχή Π τέτοια ώστε $(x,y) \in \Pi$ και ένα χρονικό διάστημα Δ, τέτοιο ώστε $t \in \Delta$. Για να μπορέσει να σταλεί η αίτηση στον κατάλληλο παροχέα υπηρεσιών, θα πρέπει στην περιοχή Π να έχουν βρεθεί τουλάχιστον k-1 χρήστες κατά τη διάρκεια του χρονικού διαστήματος Δ. Επιβάλλοντας αυτή τη συνθήκη επιτυγχάνεται ανωνυμία, καθώς ακόμη και στην περίπτωση όπου ο παροχέας υπηρεσιών μάθει ποιοι χρήστες βρίσκονταν στην Π, το χρονικό διάστημα Δ, δεν θα μπορεί να γνωρίζει ποιος από αυτούς έστειλε την αίτηση.

Μία θεώρηση που έχει γίνει στο παραπάνω μοντέλο (η οποία θα χρησιμοποιηθεί και στη παρούσα εργασία), είναι ότι οι χρήστες που ανήκουν στο σύνολο ανωνυμίας δεν είναι ανάγκη να έχουν πραγματοποιήσει όλοι

[3] Ακριβής ορισμός για το τι είναι το σύνολο ανωνυμίας θα δοθεί στην εισαγωγή του Κεφαλαίου 4

κάποια αίτηση προς τον ίδιο παροχέα υπηρεσιών. Αν έπρεπε όλοι οι χρήστες (που ανήκουν στο σύνολο ανωνυμίας) να έχουν στείλει τον ίδιο τύπο αίτησης με τον αιτούντα, δεν θα μας ενδιέφερε καθόλου το μέγεθος που θα πρέπει να έχει το σύνολο ανωνυμίας. Αυτό οφείλεται στο γεγονός ότι αν αποκαλύπτονταν τα μέλη του συνόλου ανωνυμίας καθώς και κάποια αίτηση που είχε αποστείλει ένας από αυτούς, αυτόματα θα γινόταν γνωστό το είδος της αίτησης που είχαν αποστείλει και οι υπόλοιποι. Αντίθετα, θεωρώντας ότι τα μέλη του συνόλου ανωνυμίας μπορούν να έχουν στείλει διαφορετικές αιτήσεις (ή να μην έχουν στείλει καμία αίτηση) διασφαλίζεται ότι ακόμη και αν αποκαλυφθούν τα μέλη του συνόλου ανωνυμίας, δεν θα μπορούν να συσχετιστούν οι αιτήσεις τους.

Ακολούθως ορίζουμε πότε επιτυγχάνεται ανωνυμία όταν ένας χρήστης αποστέλλει μία αίτηση (χρησιμοποιούμε τον ίδιο ορισμό με αυτόν του [4]) :

Ορισμός 3.8. Αν R είναι το σύνολο όλων των αιτήσεων που έχουν σταλεί σε κάποιον παροχέα υπηρεσιών και R' είναι το σύνολο των αιτήσεων που έχουν σταλεί από τον ίδιο χρήστη U, λέμε ότι επιτυγχάνεται *k-ανωνυμία*, αν και μόνο αν υπάρχουν k-1 PHLs: $P_1, P_2, ..., P_{k-1}$ για k-1 χρήστες διαφορετικών του U, τέτοια ώστε κάθε $P_j, \quad j = 1,2,...k-1$ να είναι *χωρο-χρονικά συνεπές* με το R'.

Αυτό που θέλουμε να επιτύχουμε με τη στρατηγική που θα προταθεί στο Κεφάλαιο 4, είναι να διασφαλίσουμε ότι όταν ένα σύνολο από αιτήσεις R' *ταιριάζει* με ένα LBQID και *διασυνδέεται* με πιθανότητα Θ, τότε επιτυγχάνεται k ανωνυμία. Αυτό σημαίνει ότι ακόμη και αν ένας παροχέας υπηρεσιών ταιριάξει όλες τις αιτήσεις ενός χρήστη με τα στοιχεία κάποιων LBQIDs, θα υπάρχουν k-1 άλλοι χρήστες των οποίων το PHL είναι χωρο-χρονικά συνεπές με αυτές τις αιτήσεις. Δηλαδή, δεν θα μπορούμε να πούμε με απόλυτη βεβαιότητα ποιος ήταν εκείνος ο χρήστης που έκανε τις συγκεκριμένες αιτήσεις.

4. Το μοντέλο της k-Ανωνυμίας για Πολλαπλούς Προσδιοριστές Προστασίας

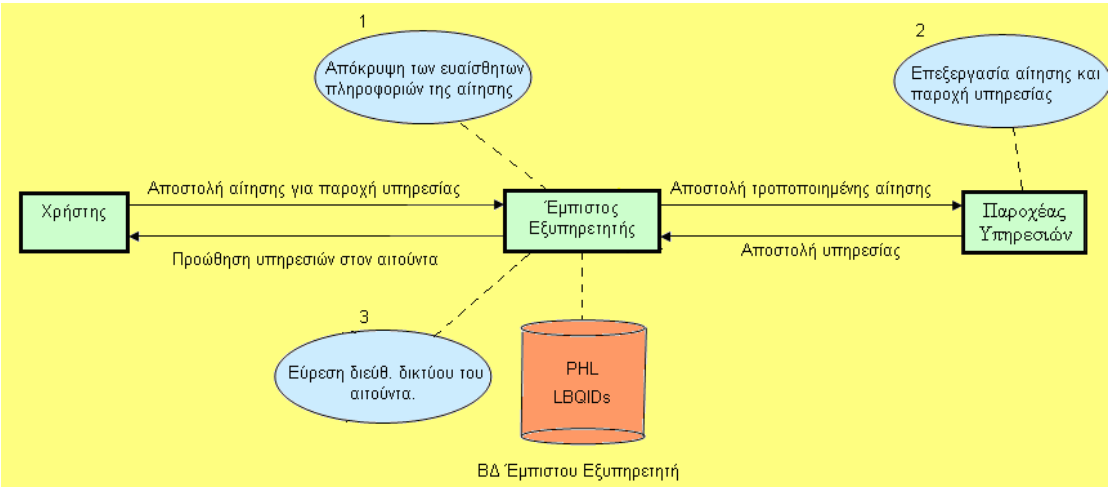
Στο παρόν κεφάλαιο θα παρουσιάσουμε μια στρατηγική, η οποία έχει ως στόχο την προστασία της ταυτότητας εκείνων των χρηστών που στέλνουν αιτήσεις σε παροχείς υπηρεσιών, οι οποίοι χρειάζονται τις χωρο-χρονικές συντεταγμένες των αιτούντων. Όπως αναφέρθηκε και στην Εισαγωγή, τέτοιου είδους υπηρεσίες είναι υπηρεσίες πλοήγησης (navigation services) όπως αυτές που παρέχονται από ένα GPS σύστημα, υπηρεσίες που παρέχουν πληροφορίες σχετικά με την κατάσταση του οδοστρώματος ή τον καιρό, αλλά και υπηρεσίες που επιστρέφουν την ακριβή θέση κτιρίων ή καταστημάτων (π.χ. φαρμακείων, ξενοδοχείων κτλ). Αυτές τις υπηρεσίες μπορεί να τις ζητήσει ο χρήστης μέσω ενός κινητού τηλεφώνου, ενός συστήματος πλοήγησης (GPS), ενός PDA ή γενικότερα μέσω οποιασδήποτε ασύρματης συσκευής η οποία παρέχει τέτοιου είδους δυνατότητες.

Η παραπάνω απαίτηση για προστασία της ταυτότητας των χρηστών, μπορεί να ικανοποιηθεί μέσω της αποστολής *ανώνυμων* αιτήσεων. Σύμφωνα με το [1] *ανωνυμία είναι η κατάσταση όπου ένα άτομο δεν μπορεί να προσδιοριστεί μέσα από ένα σύνολο ατόμων, το σύνολο ανωνυμίας*. Στα πλαίσια αυτής της εργασίας, όπου εξετάζονται υπηρεσίες οι οποίες χρησιμοποιούν τις χωρο-χρονικές συντεταγμένες του αιτούντα (δύο διαστάσεις για τη θέση και μία για τον χρόνο), το σύνολο ανωνυμίας θα αποτελείται από χρήστες τέτοιου είδους υπηρεσιών.

Η στρατηγική που προτείνεται, απαρτίζεται από δύο βασικούς αλγορίθμους: έναν αλγόριθμο γενίκευσης (generalization) και έναν αλγόριθμο αποσύνδεσης (unlinking). Καθένας από αυτούς πραγματοποιεί κάποιες αλλαγές στα δεδομένα που αφορούν τον χρήστη και τις αιτήσεις που αυτός αποστέλλει, ώστε να παραμείνουν μυστικές η ταυτότητα του χρήστη αλλά και άλλες ευαίσθητες πληροφορίες. Αυτοί οι αλγόριθμοι εφαρμόζονται στον έμπιστο εξυπηρετητή, προτού η αίτηση του χρήστη σταλεί στον παροχέα υπηρεσιών. Στην ακόλουθη ενότητα θα παρουσιαστεί η αρχιτεκτονική του μοντέλου που θεωρήσαμε.

4.1 Μοντέλο για τη διατήρηση της ανωνυμίας

Στο Σχήμα 4.1, παρουσιάζεται η αρχιτεκτονική του μοντέλου που θεωρήσαμε. Το μοντέλο αυτό, το οποίο έχει επίσης χρησιμοποιηθεί στα [4],[7],[8], αντιπροσωπεύει σε μεγάλο βαθμό τον τρόπο που λειτουργούν τα σύγχρονα συστήματα παροχής υπηρεσιών πραγματικού χρόνου. Κεντρικό ρόλο στο μοντέλο διαδραματίζει ένας έμπιστος κεντρικός εξυπηρετητής (Trusted Server), στον οποίο αποστέλλονται οι αιτήσεις των χρηστών. Η ύπαρξη ενός και μόνο κεντρικού εξυπηρετητή, μπορεί να θεωρηθεί απειλή για την ασφάλεια (σε σχέση με την ιδιωτικότητα των χρηστών). Αυτό οφείλεται στο γεγονός ότι η επεξεργασία των αιτήσεων αλλά και η προώθηση αυτών στους παροχείς υπηρεσιών, γίνεται στον κεντρικό εξυπηρετητή. Αν λοιπόν πραγματοποιηθεί επίθεση στον κεντρικό εξυπηρετητή, η αδυναμία λειτουργίας αυτού θα συνεπάγεται και αδυναμία επεξεργασίας των αιτήσεων των χρηστών, άρα και παροχή υπηρεσιών σε αυτούς. Ωστόσο, επειδή οι δυνατότητες των ασύρματων συσκευών είναι περιορισμένες (όσον αφορά τους πόρους που αυτές προσφέρουν για να διεξαχθεί σοβαρή επίθεση), μπορούμε να θεωρήσουμε ότι το μοντέλο είναι ασφαλές (όσον αφορά τη δυνατότητα λειτουργίας του).



Σχήμα 4.1: Μοντέλο για τη διατήρηση της ανωνυμίας

Οι χρήστες συνδέονται μέσω των ασύρματων συσκευών τους (π.χ. κινητών τηλεφώνων) με τον έμπιστο εξυπηρετητή οποτεδήποτε θέλουν να πραγματοποιήσουν κάποια αίτηση για παροχή υπηρεσιών. Οι χρήστες

μπορούν ανά πάσα στιγμή να ενεργοποιήσουν ή να απενεργοποιήσουν ένα σύστημα για την προστασία της ιδιωτικότητας τους μέσω μιας εύχρηστης και απλής γραφικής διεπαφής, όπως και να καθορίσουν το επίπεδο της ασφάλειας που τους παρέχεται (χαμηλού, μετρίου ή υψηλού επιπέδου). Οι επιλογές τους σχετικά με το είδος της ασφάλειας μπορούν να εφαρμοστούν είτε οποτεδήποτε αυτοί πραγματοποιούν αιτήσεις, είτε σε μεμονωμένες περιπτώσεις.

Ο έμπιστος εξυπηρετητής λαμβάνει τις αιτήσεις που αποστέλλουν οι χρήστες, και αφού τις τροποποιήσει κατάλληλα, λαμβάνοντας υπόψη και τις εκάστοτε επιλογές του αιτούντα, τις προωθεί στον παροχέα υπηρεσιών. Ο έμπιστος εξυπηρετητής εκτελεί τις συνήθεις αρμοδιότητες ενός τοπικού εξυπηρετητή. Διαθέτει μία κατάλληλη βάση δεδομένων όπου αποθηκεύονται όλοι οι χρήστες, διαχειρίζεται τις αιτήσεις που αυτοί αποστέλλουν και έχει την δυνατότητα να πραγματοποιεί χωρο-χρονικά ερωτήματα. Επιπλέον, έχει πρόσβαση στη λίστα με τα LBQIDs του κάθε χρήστη και σε αυτόν είναι αποθηκευμένα τα PHLs όλων των χρηστών.

Οι παροχείς υπηρεσιών λαμβάνουν αιτήσεις που αποστέλλει ο έμπιστος εξυπηρετητής, οι οποίες είναι της μορφής:

< msgID, userID, 3DArea, Data >

Το αναγνωριστικό *msgID*, χρησιμοποιείται για να αποκρύψουμε την διεύθυνση δικτύου (network address) του κάθε χρήστη. Το αναγνωριστικό αυτό χρησιμοποιείται από τον έμπιστο εξυπηρετητή, για να προωθήσει την απάντηση που έστειλε ο παροχέας υπηρεσιών, πίσω στον χρήστη. Το αναγνωριστικό *userID*, χρησιμοποιείται για να αποκρύψουμε την ταυτότητα του αιτούντα και είναι μοναδικό για κάθε χρήστη. Βάση αυτού του αναγνωριστικού μπορούν οι παροχείς υπηρεσιών να αναγνωρίσουν τον εκάστοτε αιτούντα, να τον χρεώσουν για τη χρήση υπηρεσιών, να του αποστείλουν μηνύματα ή να πραγματοποιήσουν άλλου είδους λειτουργίες. Η *3DArea*, χρησιμοποιείται για να απεικονίσει σε τρεις διαστάσεις (δύο διαστάσεις για τον χώρο και μία διάσταση για τον χρόνο) την πληροφορία σχετικά με την θέση και την χρονική στιγμή όπου βρισκόταν ο χρήστης όταν έστειλε την αίτηση. Η περιοχή αυτή αποτελεί γενίκευση της ακριβής θέσης του χρήστη και της χρονικής στιγμής όπου εστάλη η αίτηση. Η γενίκευση (ακριβής περιγραφή της γίνεται στην επόμενη ενότητα) αυτή γίνεται στον

έμπιστο εξυπηρετητή, στον οποίον είναι γνωστές και οι ακριβείς χωρο-χρονικές συντεταγμένες. Τέλος, *Data* είναι μία συλλογή δεδομένων, που σχετίζονται με την εκάστοτε αίτηση αλλά και το είδος της υπηρεσίας, τα οποία επεξεργάζεται ο παροχέας υπηρεσιών προτού αποστείλει την απάντηση. Μόλις ολοκληρωθεί η επεξεργασία του μηνύματος, ο παροχέας αποστέλλει την απάντηση στον έμπιστο εξυπηρετητή ο οποίος εντοπίζει την πραγματική διεύθυνση δικτύου του αιτούντα, και αφού κάνει κατάλληλες τροποποιήσεις και αφαιρέσει μη αναγκαία δεδομένα, του την αποστέλλει.

4.2 Προτεινόμενη λύση για την επίτευξη k-ανωνυμίας με τη χρήση γενίκευσης και μικτών ζωνών

Στη παρούσα ενότητα θα παρουσιαστούν οι δύο αλγόριθμοι που απαρτίζουν την προτεινόμενη στρατηγική, και θα αναλυθεί πλήρως η λειτουργία τους. Στόχος της στρατηγικής είναι να επιτευχθεί k-ανωνυμία, να διασφαλιστεί δηλαδή ότι ο παροχέας υπηρεσιών δεν θα μπορέσει να ξεχωρίσει τον αιτούντα από οποιουσδήποτε άλλους *k-1* διαφορετικούς χρήστες. Σε περίπτωση όπου η στρατηγική αποτύχει, δεν ήταν δηλαδή δυνατό να εφαρμόσουμε το σχήμα της k-ανωνυμίας, αποστέλλεται κατάλληλο μήνυμα στον αιτούντα. Το μήνυμα αυτό τον ενημερώνει ότι δεν θα πρέπει για κάποιο χρονικό διάστημα να στείλει αιτήσεις, -ή αν στείλει θα πρέπει να είναι προσεκτικός-, καθώς υπάρχει μεγάλος κίνδυνος να αποκαλυφθούν ιδιωτικές πληροφορίες που τον αφορούν. Μπορεί μάλιστα να ενεργοποιηθεί κατάλληλη επιλογή, βάση της οποίας ο έμπιστος εξυπηρετητής θα απορρίπτει τυχόν αιτήσεις αυτού του χρήστη, για όσο καιρό κρίνεται ότι είναι σε κίνδυνο.

Οι δύο αλγόριθμοι που θα παρουσιαστούν, εφαρμόζονται στον έμπιστο εξυπηρετητή οποτεδήποτε ένας χρήστης αποστέλλει αίτηση για παροχή κάποιας υπηρεσίας. Στόχος των αλγορίθμων είναι να διασφαλίσουν ότι οι χωρο-χρονικές πληροφορίες που περιλαμβάνονται στην εκάστοτε αίτηση, δεν θα μπορούν να αποκαλύψουν οποιαδήποτε ευαίσθητη πληροφορία αφορά τον αιτούντα. Αυτές οι χωρο-χρονικές πληροφορίες είναι συνήθως η χρονική στιγμή και η θέση όπου βρισκόταν ο χρήστης όταν έστειλε την αίτηση.

Για να διαπιστωθεί αν αυτές οι χωρο-χρονικές πληροφορίες μπορούν να βλάψουν την ιδιωτικότητα του χρήστη, αρχικά ελέγχεται αν η τρέχουσα αίτηση ταιριάζει με κάποιο στοιχείο από τα LBQIDs του αιτούντα. Αν βρεθεί ένα τέτοιο στοιχείο, αυτό σημαίνει ότι η αίτηση αποστέλλεται από κάποιο σημείο που ανήκει στα μοτίβα μετακίνησης του αιτούντα. Συνεπώς η αποστολή της αίτησης χωρίς την τροποποίηση των χωρο-χρονικών στοιχείων θα έχει ως αποτέλεσμα την παραβίαση της ιδιωτικότητας του αιτούντα. Προκειμένου να αποφευχθεί αυτός ο κίνδυνος, η αίτηση τροποποιείται και έπειτα προωθείται προς τον κατάλληλο παροχέα υπηρεσιών. Αν η αίτηση δεν ταιριάξει με κάποιο στοιχείο, οι χωρο-χρονικές συντεταγμένες δεν χρειάζεται να τροποποιηθούν και μένουν ως έχουν.

Στην περίπτωση όπου η τροποποίηση των χωρο-χρονικών συντεταγμένων του αιτούντα δεν είναι επιτυχής (όταν η αίτηση ταίριαξε με κάποιο στοιχείοι από τα LBQIDs του αιτούντα), η στρατηγική προσπαθεί μέσω μιας σειράς βημάτων να διασφαλίσει ότι οι παλαιότερες αιτήσεις του αιτούντα δεν θα μπορούν να διασυνδεθούν με μελλοντικές του αιτήσεις. Για να επιτευχθεί αυτός ο στόχος, η στρατηγική αλλάζει το αναγνωριστικό του αιτούντα, και αν στον έμπιστο εξυπηρετητή υπάρχουν κάποιες παλαιότερες αιτήσεις του αιτούντα, αυτές διαγράφονται και δεν προωθούνται στους παροχείς υπηρεσιών. Έπειτα από αυτή τη διαδικασία, ο αιτών φαίνεται να είναι ένας εντελώς νέος χρήστης, συνεπώς επιτυγχάνεται η απόκρυψη των ευαίσθητων πληροφοριών που τον αφορούν.

4.2.1 Αλγόριθμος γενίκευσης

Όπως διαφαίνεται και από την ονομασία του, ο αλγόριθμος γενίκευσης έχει σαν στόχο να γενικεύσει την πληροφορία αναφορικά με τη θέση και τη χρονική στιγμή όπου βρισκόταν ο χρήστης όταν πραγματοποίησε την αίτηση, προτού αυτή σταλεί στον παροχέα υπηρεσιών. Θα εξηγήσουμε την έννοια της γενίκευσης μέσω ενός απλού παραδείγματος. Ας θεωρήσουμε έναν χρήστη, ο οποίος την χρονική στιγμή t αποστέλλει μία αίτηση από την θέση (x,y). Η γενίκευση του τρισδιάστατου χώρου (x,y,t), θα είναι πάλι ένας τρισδιάστατος χώρος, όπου η χρονική στιγμή t θα έχει αντικατασταθεί από ένα χρονικό

διάστημα $[t_1,t_2]$ (στο οποίο περιέχεται το t), και το σημείο (x,y) από μία περιοχή $\Pi=[(x_1,y_1), (x_2,y_1), (x_2,y_2), (x_1,y_2)]$, με $(x,y)\in\Pi$.

Είσοδο σε αυτόν τον αλγόριθμο αποτελεί μια σταθερά k, η οποία παριστάνει τον αριθμό των χρηστών (συμπεριλαμβανομένου του αιτούντα) που πρέπει είτε να έχουν βρεθεί είτε να έχουν περάσει κάποια χρονική στιγμή, από το σημείο όπου βρισκόταν ο αιτών. Μία προφανής λύση θα ήταν ο χώρος και ο χρόνος, όπου βρισκόταν ο αιτών, να γενικεύεται διαρκώς ωσότου βρεθούν k-1 διαφορετικοί «γείτονες». Ωστόσο, για να είναι χρήσιμα στον αιτούντα τα αποτελέσματα που του επεστράφησαν, είναι απαραίτητο να υπάρχουν κάποιοι περιορισμοί σχετικά με το βαθμό της γενίκευσης που μπορεί να εφαρμοστεί. Για παράδειγμα, όταν κάποιος χρήστης που βρίσκεται στο κέντρο της πόλης, πραγματοποιεί αίτηση για να του επιστραφεί η θέση του πλησιέστερου φαρμακείου, δεν θα ήταν επιθυμητό να του επιστραφούν οι συντεταγμένες ενός φαρμακείου που βρίσκεται εκτός των ορίων της πόλης (ως αποτέλεσμα της εφαρμογής της γενίκευσης προκειμένου να βρεθούν k-1 γείτονες). Συνεπώς, σαν είσοδο στον αλγόριθμο πρέπει να δίνονται και οι περιορισμοί που αφορούν το βαθμό της γενίκευσης που μπορεί να εφαρμοστεί στον χώρο και στον χρόνο.

Για την υλοποίηση του αλγορίθμου γενίκευσης θεωρήσαμε ότι κάθε χρήστης μπορεί να έχει περισσότερα από ένα LBQIDs, τα οποία αποθηκεύονται σε μία λίστα. Οι Bettini, Wang και Jajodia είχαν θεωρήσει στο [4] την απλή περίπτωση όπου ένας χρήστης μπορεί να έχει ένα μόνο LBQID. Η παραπάνω θεώρηση (ότι ένας χρήστης μπορεί να έχει πολλαπλά LBQIDs) αποτελεί μία από τις βασικές συνεισφορές αυτής της εργασίας και μέσω αυτής επεκτείνεται ο αλγόριθμος γενίκευσης που είχε προταθεί στο [4]. Ένα παράδειγμα μιας λίστας με τα LBQIDs ενός χρήστη παριστάνεται στο Σχήμα 4.2. Σε αυτό το Σχήμα ο χρήστης θεωρείται ότι έχει 4 LBQIDs. Σε κάθε γραμμή είναι αποθηκευμένα τα στοιχεία κάθε LBQID, τα οποία αναπαριστούν περιοχές όπου συχνάζει ο χρήστης. Δίπλα σε κάθε στοιχείο υπάρχει και το αντίστοιχο χρονόσημο, το οποίο αναπαριστά ώρες μιας μέρας, χωρίς όμως να προσδιορίζεται μια συγκεκριμένη ημερομηνία. Καθένα από αυτά τα LBQIDs αποτελείται από ένα ή περισσότερα στοιχεία και από ένα τύπο επανάληψης, ο οποίος για ευκολία δεν απεικονίζεται στο σχήμα.

Παρατηρούμε ότι τα LBQIDs μπορεί να έχουν κοινά στοιχεία. Για παράδειγμα τα δύο πρώτα LBQIDs, έχουν ίδιο το πρώτο στοιχείο ενώ τα υπόλοιπα στοιχεία τους είναι διαφορετικά. Ομοίως το δεύτερο στοιχείο του $2^{ου}$ και του $3^{ου}$ LBQID είναι ίδια. Για να θεωρήσουμε ότι δύο στοιχεία είναι ίδια, θα πρέπει και τα χρονόσημα αυτών να είναι ίδια. Για παράδειγμα το πρώτο στοιχείο του $1^{ου}$ και του $4^{ου}$ LBQID δεν είναι ίδια γιατί παρόλο που παριστάνουν την ίδια περιοχή, τα χρονόσημα τους είναι διαφορετικά, δηλαδή [07-08] ≠ [18-20]. Ενώ γενικότερα, δύο διαφορετικά χρονόσημα ενός LBQID μπορούν να έχουν κάποια τομή (αφού παριστάνουν διαστήματα χρόνου) , για ευκολία θεωρήσαμε ότι δεν υπάρχει καμία τέτοια τομή. Για παράδειγμα, ένα LBQID της μορφής:

< Σπίτι [07-08], Γραφείο [07.30-09], Γραφείο [15-16], Σπίτι [16-17] >

δεν θα ήταν αποδεκτό, γιατί τα χρονόσημα των δύο πρώτων στοιχείων έχουν τομή διαφορετική από το κενό σύνολο, δηλαδή: [07-08] ∩ [07.30-09] ≠ ∅.

Αύξων αριθμός LBQID	**$1^{ο}$ Στοιχείο**	**Χρονόσημο $1^{ου}$ Στοιχείου**	**$2^{ο}$ Στοιχείο**	**Χρονόσημο $2^{ου}$ Στοιχείου**	**$3^{ο}$ Στοιχείο**	**Χρονόσημο $3^{ου}$ Στοιχείου**	**$4^{ο}$ Στοιχείο**	**Χρονόσημο $4^{ου}$ Στοιχείου**
1	Σπίτι	[07-08]	Γραφείο	[08-09]	Γραφείο	[16-17]	Σπίτι	[18-20]
2	Σπίτι	[07-08]	Καφετέρια	[16-17]	Σπίτι	[18-19]		
3	Δουλειά	[14-15]	Καφετέρια	[16-17]	Καφετέρια	[17-18]		
4	Σπίτι	[18-20]	Σινεμά	[23-24]				

Σχήμα 4.2: Παράδειγμα της λίστας με τα LBQID ενός χρήστη

Στη συνέχεια θα παρουσιαστούν τα δύο βασικά τμήματα του αλγορίθμου γενίκευσης, και θα αναλυθούν τα βήματα καθενός εξ αυτών. Το *τμήμα* του *ταιριάσματος με το πρώτο στοιχείο ενός LBQID* του αλγορίθμου γενίκευσης εκτελείται στις δύο ακόλουθες περιπτώσεις: όταν ο αλγόριθμος γενίκευσης κληθεί για πρώτη φορά και όταν η τρέχουσα αίτηση πρέπει να ταιριάξει με το πρώτο στοιχείο κάποιου από τα LBQIDs του χρήστη. Για καθεμία από τις δύο παραπάνω περιπτώσεις, αρχικά ελέγχεται αν η τρέχουσα αίτηση ταιριάζει με το πρώτο στοιχείο κάποιου από τα LBQIDs του αιτούντα. Αν υπάρχει τέτοιο

στοιχείο ο αλγόριθμος γενίκευσης προσπαθεί να βρει τους k-1 πλησιέστερους γείτονες (όπου η τιμή της σταθεράς k έχει δοθεί στην είσοδο του). Αν βρεθούν k-1 διαφορετικοί χρήστες, τότε ο αλγόριθμος γενίκευσης επιτυγχάνει k-ανωνυμία και στην έξοδο του παράγονται τα αναγνωριστικά των γειτόνων.

Το *τμήμα ταιριάσματος με ενδιάμεσο στοιχείο ενός LBQID* του αλγορίθμου γενίκευσης εκτελείται όταν θέλουμε η τρέχουσα αίτηση να ταιριάξει με ένα ενδιάμεσο στοιχείο κάποιου από τα LBQIDs του αιτούντα. Σε αυτή τη περίπτωση, στην είσοδο του αλγορίθμου γενίκευσης θα πρέπει να δίνονται τα αναγνωριστικά των k-1 πλησιέστερων γειτόνων (τα οποία είχαν βρεθεί σε προηγούμενη εκτέλεση του). Αν η τρέχουσα αίτηση δεν ταιριάξει με το «κατάλληλο» ενδιάμεσο στοιχείο, τότε εκτελείται το *τμήμα ταιριάσματος με το πρώτο στοιχείο ενός LBQID* του αλγορίθμου γενίκευσης. Διαφορετικά, θα πρέπει σε καθένα από τα PHLs των k-1 γειτόνων να βρεθεί ένα σημείο που να ταιριάζει με την τρέχουσα αίτηση. Ο λόγος που πρέπει τα αναγνωριστικά των k-1 γειτόνων να δοθούν στην είσοδο του αλγορίθμου, είναι επειδή θέλουμε να ελέγξουμε αν το PHL κάθε «γείτονα» είναι χωρο-χρονικά συνεπές με την τρέχουσα αίτηση. Επιπλέον, μέσω των αναγνωριστικών, είναι δυνατόν να αποκτήσουμε πρόσβαση στα στοιχεία των PHLs που μας ενδιαφέρουν, δηλαδή αυτά των γειτόνων. Αν και τα k-1 PHLs είναι χωρο-χρονικά συνεπή με την τρέχουσα αίτηση (λαμβάνοντας υπόψη και τους σχετικούς περιορισμούς), ο αλγόριθμος γενίκευσης επιτυγχάνει k-ανωνυμία.

Τμήμα ταιριάσματος αίτησης με το πρώτο στοιχείο ενός LBQID. Αν σαν είσοδο στον αλγόριθμο γενίκευσης δοθούν μια σταθερά k και οι περιορισμοί για το βαθμό της γενίκευσης που μπορεί να εφαρμοστεί στις χωρο-χρονικές συντεταγμένες όπου βρισκόταν ο αιτών, τότε εκτελούνται τα ακόλουθα βήματα:

a) Αρχικά θεωρείται μία λίστα στην οποία αποθηκεύονται εκείνα τα LBQIDs του αιτούντα, των οποίων ορισμένα στοιχεία έχουν ταιριάξει με κάποια προηγούμενη αίτηση του. Έστω ότι αυτή η λίστα συμβολίζεται με L. Στην L δεν θα προστεθούν εκείνα τα LBQID των οποίων *όλα* τα στοιχεία έχουν ταιριάξει με προηγούμενες αιτήσεις, επειδή αυτά τα LBQIDs μπορούν να ταιριάξουν ξανά με μελλοντικές αιτήσεις, ειδικά αν αυτός συνηθίζει να

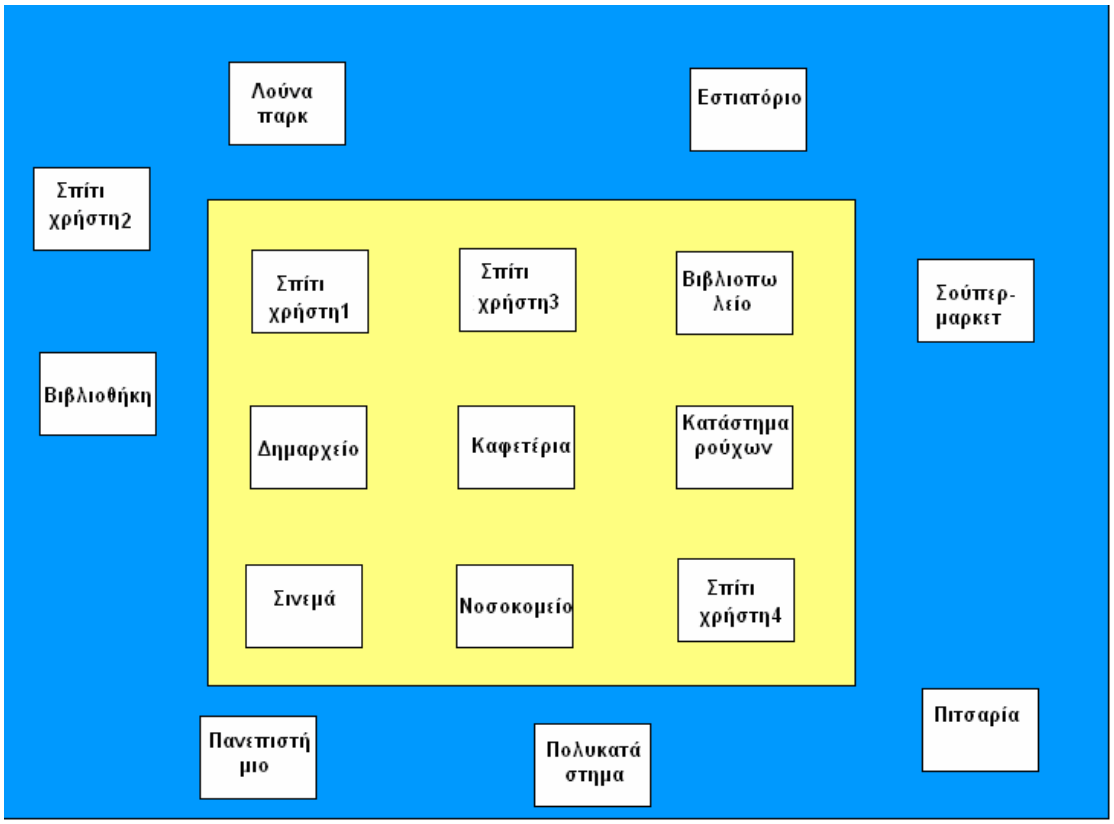
μετακινείται στις ίδιες περιοχές. Αν για παράδειγμα, κάποια αίτηση ταίριαξε με το τελευταίο στοιχείου του 1ου LBQID στο Σχήμα 4.2, δηλαδή με το <Σπίτι [18-20]>, υπάρχει μεγάλη πιθανότητα κάποια επόμενη αίτηση να ταιριάξει με το πρώτο στοιχείο αυτού του LBQID, δηλαδή με το <Σπίτι [07-08]>. Όταν ο αλγόριθμος γενίκευσης κληθεί για πρώτη φορά η L θα είναι η κενή λίστα. Στη συνέχεια, από την λίστα με όλα τα LBQID του αιτούντα, αφαιρούνται αυτά που υπάρχουν στην L. Ας συμβολίσουμε με F την τελική λίστα που προέκυψε. Την πρώτη φορά που θα κληθεί ο αλγόριθμος γενίκευσης, επειδή η L θα είναι κενή, η F θα περιέχει όλα τα LBQID του αιτούντα. Αν η F δεν είναι κενή λίστα, ο αλγόριθμος γενίκευσης προχωρά με το επόμενο βήμα. Διαφορετικά ο αλγόριθμος τερματίζει, και οι συντεταγμένες (x,y,t) όπου βρισκόταν ο αιτών, δεν χρειάζεται να γενικευτούν. Επειδή οι συντεταγμένες όπου βρισκόταν ο αιτών δεν ταίριαξαν με κανένα στοιχείο του LBQID, αυτό σημαίνει ότι αυτό το σημείο δεν ανήκει σε κανένα από τα πρότυπα μετακίνησης του. Άρα αποκάλυψη αυτού του σημείου δεν συνεπάγεται και αποκάλυψη πληροφοριών που αφορούν τον αιτούντα.

b) Ο αλγόριθμος γενίκευσης ελέγχει αν η τρέχουσα αίτηση ταιριάζει με το πρώτο στοιχείο κάποιου LBQID της F. Αν δεν βρεθεί κανένα τέτοιο στοιχείο, τότε δεν χρειάζεται να γενικευτεί η θέση και η χρονική στιγμή όπου βρισκόταν ο αιτών. Αυτό συμβαίνει επειδή το συγκεκριμένο τρισδιάστατο σημείο δεν ανήκει σε κάποια από τα μοτίβα μετακίνησης που συνηθίζει να ακολουθεί ο αιτών, και συνεπώς η αποκάλυψη της θέσης του όταν έκανε την αίτηση δεν αποτελεί απειλή για την ασφάλεια. Για παράδειγμα αν ένας χρήστης δεν συνηθίζει να συχνάζει σε θέατρα, και ξαφνικά εμφανιστεί ότι έστειλε μία αίτηση από κάποιο θέατρο, η αποκάλυψη της τριάδας (x,y,t) (από όπου έγινε η αίτηση) δεν θα μπορέσει να αποκαλύψει την ταυτότητα του αιτούντα. Αυτό οφείλεται στο γεγονός ότι η συγκεκριμένη αίτηση δεν θα είναι δυνατόν να συσχετιστεί με άλλες προηγούμενες αιτήσεις που έγιναν σε μέρη όπου συχνάζει ο αιτών. Στην περίπτωση όπου η αίτηση ταιριάξει με κάποιο στοιχείο ενός LBQID της F, ο αλγόριθμος συνεχίζει με το επόμενο βήμα.

c) Ο αλγόριθμος γενίκευσης προσπαθεί να βρει k-1 διαφορετικούς χρήστες (χωρίς να περιλαμβάνεται ο αιτών), οι οποίοι κάποια χρονική στιγμή είχαν

διασχίσει το σημείο (x,y) όπου βρισκόταν ο αιτών όταν έστειλε την αίτηση. Για να το επιτύχει αυτό γενικεύεται σταδιακά ο χρόνος και ο χώρος όπου πραγματοποιήθηκε η αίτηση, όσο το επιτρέπουν βέβαια οι περιορισμοί που εισήγαγε ο αιτών. Σε κάθε βήμα της γενίκευσης, αποθηκεύονται εκείνοι οι χρήστες που ικανοποιούν τους περιορισμούς και οι οποίοι είχαν περάσει από το σημείο (x,y), στην λίστα με τους k-1 πλησιέστερους γείτονες. Μόλις βρεθούν k-1 χρήστες, ο αλγόριθμος τερματίζει και στην έξοδο δίνονται τα αναγνωριστικά των k-1 γειτόνων καθώς και τα όρια της τρισδιάστατης περιοχής (χώρος + χρόνος) όπου αυτοί βρέθηκαν. Αν δεν μπορεί να εφαρμοστεί άλλο γενίκευση και δεν έχουν ήδη βρεθεί k διαφορετικοί χρήστες (συμπεριλαμβανομένου του αιτούντα), αυτό σημαίνει ότι ο αλγόριθμος γενίκευσης έχει αποτύχει. Πρέπει να εφαρμοστεί ο αλγόριθμος αποσύνδεσης ο οποίος και αποτελεί το δεύτερο μέλος της στρατηγικής.

Παράδειγμα 4.1: Παράδειγμα εκτέλεσης του αλγορίθμου γενίκευσης όταν αυτός κληθεί για πρώτη φορά



Σχήμα 4.3: Όρια για τη γενίκευση της περιοχής στο Παράδειγμα 1

Θεωρούμε τον χρήστη1, ο οποίος έστειλε μία αίτηση από το σπίτι του στις 08-08-2006, στις 18.00, για να μάθει ποιες ταινίες προβάλλονται στο πλησιέστερο σινεμά. Η λίστα με τα LBQID του χρήστη1, είναι αυτή που εικονίζεται στο Σχήμα 4.2. Προτού αυτή η αίτηση σταλεί στον παροχέα υπηρεσιών, θα πρέπει να κληθεί ο αλγόριθμος γενίκευσης. Θεωρούμε ότι η τιμή του k που δίνεται στην είσοδο είναι 3. Όσον αφορά τους περιορισμούς για τη γενίκευση του χρόνου και του χώρου, θεωρούμε ότι ο χρόνος μπορεί να γενικευτεί μέχρι και 2 ώρες (δηλαδή το επιτρεπόμενο διάστημα για γενίκευση είναι το [16.00, 20.00]), ενώ τα όρια για τη γενίκευση του χώρου παριστάνονται στο Σχήμα 4.3. Με κίτρινο συμβολίζεται η περιοχή που βρίσκεται μέσα στα επιτρεπόμενα όρια για τη γενίκευση, ενώ με μπλε η περιοχή που είναι εκτός ορίων.

PHL για τον χρήστη2	
06.00	Σπίτι χρήστη2
07.00	Σπίτι χρήστη2
08.00	Νοσοκομείο
09.00	Νοσοκομείο
10.00	Νοσοκομείο
11.00	Νοσοκομείο
12.00	Νοσοκομείο
13.00	Νοσοκομείο
14.00	Νοσοκομείο
15.00	Νοσοκομείο
16.00	Σπίτι χρήστη2
17.00	Σπίτι χρήστη2
18.00	Βιβλιοθήκη
19.00	Βιβλιοθήκη
20.00	Σπίτι χρήστη2
21.00	Σπίτι χρήστη2
22.00	Σινεμά
23.00	Σινεμά
24.00	Σινεμά
01.00	Σπίτι χρήστη2
02.00	Σπίτι χρήστη2
03.00	Σπίτι χρήστη2
04.00	Σπίτι χρήστη2
05.00	Σπίτι χρήστη2

PHL για τον χρήστη3	
06.00	Σπίτι χρήστη3
07.00	Σπίτι χρήστη3
08.00	Σπίτι χρήστη3
09.00	Σπίτι χρήστη3
10.00	Σούπερ-μάρκετ
11.00	Σούπερ-μάρκετ
12.00	Πανεπιστήμιο
13.00	Πανεπιστήμιο
14.00	Πανεπιστήμιο
15.00	Βιβλιοθήκη
16.00	Σπίτι χρήστη1
17.00	Σπίτι χρήστη1
18.00	Σπίτι χρήστη1
19.00	Κατάστημα ρούχων
20.00	Καφετέρια
21.00	Καφετέρια
22.00	Σινεμά
23.00	Σινεμά
24.00	Σπίτι χρήστη3
01.00	Σπίτι χρήστη3
02.00	Σπίτι χρήστη3
03.00	Σπίτι χρήστη3
04.00	Σπίτι χρήστη3
05.00	Σπίτι χρήστη3

PHL για τον χρήστη4	
06.00	Σπίτι χρήστη4
07.00	Σπίτι χρήστη4
08.00	Σπίτι χρήστη4
09.00	Σπίτι χρήστη4
10.00	Σπίτι χρήστη4
11.00	Σπίτι χρήστη4
12.00	Δημαρχείο
13.00	Βιβλιοπωλείο
14.00	Σπίτι χρήστη4
15.00	Σπίτι χρήστη4
16.00	Κατάστημα ρούχων
17.00	Κατάστημα ρούχων
18.00	Λούνα Παρκ
19.00	Λούνα Παρκ
20.00	Πολυκατάστημα
21.00	Πολυκατάστημα
22.00	Πιτσαρία
23.00	Πιτσαρία
24.00	Σπίτι χρήστη2
01.00	Σπίτι χρήστη2
02.00	Σπίτι χρήστη2
03.00	Σπίτι χρήστη2
04.00	Σπίτι χρήστη2
05.00	Σπίτι χρήστη2

Σχήμα 4.4: PHLs για τους χρήστη2, χρήστη3,χρήστη4 στο Παράδειγμα 4.1

Θεωρούμε ότι στον κεντρικό εξυπηρετητή, εκτός από τον χρήστη1, αποθηκεύονται τα PHLs και άλλων τριών χρηστών, των χρήστη2, χρήστη3 και χρήστη4. Το PHL κάθε χρήστη περιέχει τριάδες της μορφής <x, y, t> που δηλώνουν ότι ο χρήστης τη χρονική στιγμή t βρισκόταν στο σημείο (x,y). Ενώ γενικότερα εγγραφές στα PHLs μπορούν να γίνονται ανά τακτά χρονικά διαστήματα, για το παράδειγμα μας θεωρήσαμε ότι εγγραφές γίνονται κάθε μία ώρα. Οι πίνακες στο Σχήμα 4.4, περιέχουν τα PHLs των χρήστη2, χρήστη3 και χρήστη4 για την ημέρα 08-08-2006.

Επειδή ο χρήστης1 πραγματοποιεί αίτηση για πρώτη φορά, δεν θα υπάρχουν LBQIDs των οποίων κάποια στοιχεία να έχουν ταιριάξει με προηγούμενη αίτηση, άρα η L θα είναι η κενή λίστα και η F θα είναι αυτή που εικονίζεται στο Σχήμα 4.2. Σύμφωνα με το Βήμα b του *τμήματος ταιριάσματος με το πρώτο στοιχείο ενός LBQID*, θα πρέπει πρώτα να ελέγξουμε αν η αίτηση του χρήστη1, ταιριάζει με το πρώτο στοιχείο κάποιου από τα LBQID του. Παρατηρούμε ότι ταιριάζει με το πρώτο στοιχείο του τέταρτου LBQID, δηλαδή με το <Σπίτι [18-20]>, αφού η αίτηση έγινε στο σπίτι του χρήστη1 και επιπλέον έγινε στις 18.00 που ανήκει στο διάστημα [18-20]. Αφού βρέθηκε κάποιο LBQID, θα πρέπει να προχωρήσουμε με το Βήμα c. Γενικεύουμε σταδιακά τον χρόνο και τον χώρο ώσπου να βρούμε 2 γείτονες. Στις 18.00, εντός των επιτρεπόμενων χωρικών ορίων βρισκόταν μόνο ο χρήστης3, που ήταν στο σπίτι του χρήστη1. Θα πρέπει να γενικεύσουμε συνεπώς τον χρόνο. Στο διάστημα [17-19], βρίσκονταν δύο χρήστες εντός της επιτρεπόμενης περιοχής, αφού ο χρήστης3 στις 17.00 βρισκόταν στο σπίτι του χρήστη1 και ο χρήστης4 την ίδια ώρα βρισκόταν στο κατάστημα ρούχων. Αφού βρέθηκαν k-1 = 2 γείτονες (που είναι οι χρήστης3 και χρήστης4), ο αλγόριθμος γενίκευσης πέτυχε και η γενικευμένη χωρική περιοχή είναι αυτή που σημειώνεται με κίτρινο στο Σχήμα 4.3, ενώ η γενικευμένη χρονική περιοχή είναι η [17.00 - 19.00]. Αν δεν είχαμε κατορθώσει εντός των επιτρεπόμενων ορίων (που προαναφέρθηκαν) να βρούμε 2 γείτονες, θα έπρεπε να εκτελέσουμε τον αλγόριθμο γενίκευσης.

Τμήμα ταιριάσματος αίτησης με ενδιάμεσο στοιχείο ενός LBQID. Αν σαν είσοδο στον αλγόριθμο γενίκευσης δοθούν τα αναγνωριστικά των k-1 γειτόνων και οι περιορισμοί για τον βαθμό της γενίκευσης που μπορεί να

εφαρμοστεί στις χωρο-χρονικές συντεταγμένες όπου βρισκόταν ο αιτών, τότε εκτελούνται τα ακόλουθα βήματα:

a) Αρχικά ο αλγόριθμος γενίκευσης ελέγχει αν η τρέχουσα αίτηση ταιριάζει με το «κατάλληλο» στοιχείο κάποιου από τα LBQIDs του αιτούντα, των οποίων ορισμένα στοιχεία είχαν ταιριάξει με προηγούμενες αιτήσεις του. Με την έννοια «κατάλληλο» στοιχείο εννοούμε το εξής: αν κάποια προηγούμενη αίτηση r_i του αιτούντα είχε ταιριάξει με το στοιχείο E_i κάποιων από τα LBQIDs του, η τωρινή αίτηση r_{i+1} θα πρέπει να ταιριάξει με το στοιχείο E_{i+1} αυτών των LBQIDs. Στο Παράδειγμα 4.1, επειδή η προηγούμενη αίτηση του χρήστη1 ταίριαξε με το πρώτο στοιχείο του τέταρτου LBQID, δηλαδή με το <Σπίτι [18-20]>, η τρέχουσα αίτηση του χρήστη1, θα πρέπει να ταιριάξει με το δεύτερο στοιχείο αυτού του LBQID, δηλαδή με το <Σινεμά [23-24]>. Αν η τρέχουσα αίτηση δεν ταιριάξει με το «κατάλληλο» ενδιάμεσο στοιχείο κανενός LBQID από αυτά που προηγουμένως είχαν ταιριάξει με κάποια αίτηση, δεν σημαίνει ότι η στρατηγική απέτυχε, ότι δηλαδή δεν είναι δυνατόν να επιτευχθεί k-ανωνυμία. Αντίθετα, επειδή έχουμε θεωρήσει ότι δύο ή περισσότερα LBQIDs του αιτούντα μπορούν να έχουν κοινά στοιχεία, ο αλγόριθμος γενίκευσης θα πρέπει να ελέγξει αν η τρέχουσα αίτηση ταιριάζει με το πρώτο στοιχείο κάποιου άλλου LBQID (του οποίου κανένα στοιχείο δεν είχε ταιριάξει με προηγούμενη αίτηση). Θεωρώντας ξανά το Παράδειγμα 4.1, αν η τρέχουσα αίτηση δεν ταιριάξει με το «κατάλληλο» στοιχείο που είναι το: <Σπίτι [18-20]>, ο αλγόριθμος γενίκευσης θα πρέπει να ελέγξει αν ταιριάζει με ένα από τα: <Σπίτι [07-08]> και <Δουλειά [14-15]> που αποτελούν τα πρώτα στοιχεία των υπολοίπων LBQIDs του χρήστη1. Για τον παραπάνω λόγο καλούμε ξανά τον αλγόριθμο γενίκευσης με τους ίδιους περιορισμούς για την γενίκευση που μπορεί να εφαρμοστεί στο χώρο και στον χρόνο και αντί να δώσουμε τα αναγνωριστικά των k-1 γειτόνων, δίνουμε σαν είσοδο μια σταθερά k. Αν κάποιο «κατάλληλο» στοιχείο, ταιριάξει με την τρέχουσα αίτηση, προχωράμε με το επόμενο βήμα.
b) Ο αλγόριθμος προσπαθεί να βρει στα PHLs των γειτόνων, των οποίων τα αναγνωριστικά έχουν δοθεί στην είσοδο, τρισδιάστατα σημεία τα οποία

είναι χωρο-χρονικά συνεπή με την τρέχουσα αίτηση. Υπενθυμίζουμε ότι το PHL κάθε χρήστη περιέχει τριάδες της μορφής $<x_1,y_1,t_1>$, που δείχνουν ότι ο χρήστης τη χρονική στιγμή t_1 βρισκόταν στο σημείο (x_1,y_1). Για να είναι η εφαρμογή του αλγορίθμου γενίκευσης επιτυχής θα πρέπει να βρεθούν k-1 τέτοια διαφορετικά σημεία, ένα για κάθε χρήστη που έχει σημειωθεί ως γείτονας. Για να επιτευχθεί αυτός ο στόχος γενικεύεται σταδιακά, πρώτα ο χρόνος και έπειτα ο χώρος, όσο βέβαια το επιτρέπουν και οι περιορισμοί που δόθηκαν στην είσοδο. Μόλις βρεθούν k-1 διαφορετικά σημεία, ο αλγόριθμος τερματίζει και σαν έξοδο παράγεται ο τρισδιάστατος χώρος στον οποίο βρέθηκαν αυτά τα σημεία. Αν δεν μπορεί να γενικευτεί άλλο ο χρόνος και ο χώρος και δεν βρέθηκαν k-1 σημεία, αυτό σημαίνει ότι ο αλγόριθμος γενίκευσης απέτυχε. Θα πρέπει να κληθεί ο αλγόριθμος αποσύνδεσης ο οποίος και αποτελεί το δεύτερο τμήμα της στρατηγικής μας και αναλύεται ακολούθως.

Παράδειγμα 4.2: Παράδειγμα εκτέλεσης του αλγορίθμου γενίκευσης

Ας θεωρήσουμε ξανά τον χρήστη1, του οποίου η λίστα με τα LBQIDs παρουσιάζεται στο Σχήμα 4.2. Θεωρούμε ότι αυτός στις 08-08-2006, στις 23.30, αποστέλλει από το σινεμά αίτηση για να μάθει που βρίσκεται το πλησιέστερο εστιατόριο. Επειδή, ο χρήστης1 έχει αποστείλει ξανά αίτηση στο παρελθόν, αυτή τη φορά σαν είσοδο στον αλγόριθμο γενίκευσης θα πρέπει να δοθούν τα αναγνωριστικά των χρήστη3 και χρήστη4 (οι οποίοι βάση του Παραδείγματος 4.1 είχαν βρεθεί σαν οι k-1 πλησιέστεροι γείτονες, όταν k=3), και οι περιορισμοί για τη γενίκευση. Θεωρούμε ότι ο χρόνος μπορεί να γενικευτεί μέχρι και δύο ώρες, δηλαδή το επιτρεπόμενο διάστημα για τη γενίκευση είναι το [21.30, 01.30], ενώ η επιτρεπόμενη περιοχή για τη γενίκευση του χώρου απεικονίζεται στο Σχήμα 4.3.

Σύμφωνα με το Βήμα a του *τμήμα ταιριάσματος με ενδιάμεσο στοιχείο ενός LBQID* του αλγορίθμου γενίκευσης, αρχικά ελέγχεται αν η τρέχουσα αίτηση ταιριάζει με το «κατάλληλο» στοιχείο του τέταρτου LBQID που είναι το: <Σινεμά [23-24]>. Ο έλεγχος αυτός είναι αληθής, συνεπώς προχωράμε με το Βήμα b του αλγορίθμου. Θα πρέπει να δούμε αν οι χρήστης3 και χρήστης4, είχαν περάσει κάποια χρονική στιγμή από το Σινεμά (λαμβάνοντας υπόψη και τα επιτρεπόμενα όρια για τη γενίκευση). Παρατηρούμε ότι στο διάστημα

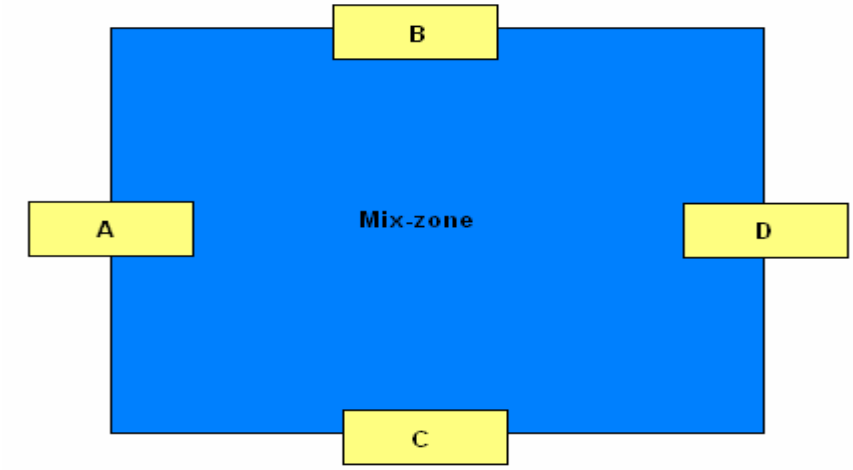
[21.30, 01.30] ο χρήστης3 είχε βρεθεί στο Σινεμά, ενώ σε αυτό το χρονικό διάστημα ο χρήστης4, δεν είχε περάσει καθόλου από αυτό το σημείο. Συνεπώς αφού δεν βρέθηκαν 2 διαφορετικά σημεία στα PHL των χρήστη3 και χρήστη4, ο αλγόριθμος γενίκευσης αποτυγχάνει και θα πρέπει να εκτελεστεί ο αλγόριθμος αποσύνδεσης.

4.2.2 Αλγόριθμος αποσύνδεσης

Ο αλγόριθμος αυτός εφαρμόζεται στα δεδομένα μας, όταν -για τους λόγους που προαναφέρθηκαν- αποτύχει ο αλγόριθμος γενίκευσης. Στόχος του αλγορίθμου αποσύνδεσης είναι η αλλαγή του αναγνωριστικού του αιτούντα, ώστε να μην υπάρχει καμία διασύνδεση μεταξύ των αιτήσεων που αυτός πραγματοποίησε στο παρελθόν και αυτών που πρόκειται να πραγματοποιήσει μελλοντικά. Η αλλαγή του αναγνωριστικού του αιτούντα δεν μπορεί να γίνει ανά πάσα στιγμή, αλλά μόνο όποτε αυτός διασχίσει μια μικτή ζώνη (mix-zone) [9]. Οι μικτές ζώνες, είναι φυσικές τοποθεσίες συγκεκριμένης έκτασης, όπου καμία υπηρεσία δεν είναι διαθέσιμη σε κανέναν χρήστη. Για διαφορετικούς χρήστες ή ομάδες χρηστών, μπορούν να οριστούν διαφορετικές μικτές ζώνες, διαφορετικής εκτάσεως η καθεμία. Η βασική ιδιότητα των περιοχών αυτών είναι ότι όποτε ένας χρήστης τις διασχίσει, στην έξοδο του από αυτές θα έχει ένα τελείως νέο αναγνωριστικό το οποίο δεν ανήκε προηγουμένως σε κανέναν άλλο χρήστη. Συνεπώς, θα είναι πολύ δύσκολο για έναν παροχέα υπηρεσιών να ξεχωρίσει αυτόν τον χρήστη από οποιονδήποτε άλλον βρισκόταν στην μικτή ζώνη στο ίδιο διάστημα του χρόνου, όπως είναι και πολύ δύσκολο να υπάρξει κάποια διασύνδεση μεταξύ των χρηστών που εισήλθαν και αυτών που εξήλθαν από την μικτή ζώνη. Βασική προϋπόθεση για να είναι επιτυχής αυτή η «αποσύνδεση» είναι ότι στην μικτή ζώνη πρέπει να υπάρχει ένας ικανοποιητικός αριθμός χρηστών.

Στο Σχήμα 4.5 υπάρχει ένα παράδειγμα μιας μικτής ζώνης. Οι περιοχές A,B,C και D παριστάνουν πιθανές περιοχές από όπου οι χρήστες μπορεί να είχαν στείλει αιτήσεις προτού εισέλθουν στην μικτή ζώνη. Οι περιοχές αυτές έχουν τοποθετηθεί τυχαία και μπορούν να εκτείνονται σε οποιοδήποτε σημείο των ορίων της μικτής ζώνης. Από την στιγμή που οι χρήστες εξέλθουν από την μικτή ζώνη, οποιεσδήποτε νέες αιτήσεις σταλούν

δεν θα μπορέσουν να συνδεθούν με αιτήσεις που είχαν σταλεί προηγουμένως. Ας θεωρήσουμε για παράδειγμα ότι υπήρχαν δύο χρήστες οι Χρήστης1 και Χρήστης2, οι οποίοι προτού εισέλθουν στην μικτή ζώνη είχαν αντίστοιχα στείλει αιτήσεις από τις περιοχές A και B. Αν στην έξοδο τους από την μικτή ζώνη σταλούν ξανά αιτήσεις από τις περιοχές A και B, ο παροχέας υπηρεσιών δεν θα μπορεί να γνωρίζει ποιος χρήστης στέλνει ποια αίτηση ακόμη και αν οι χρήστες Χρήστης1 και Χρήστης2 δεν είχαν μετακινηθεί αρκετά από την προηγούμενη τους θέση (η οποία ήταν γνωστή στον παροχέα). Φυσικά απαραίτητη προϋπόθεση για να ισχύσουν τα προηγούμενα είναι ότι στην μικτή ζώνη υπάρχει ένας ικανοποιητικός αριθμός χρηστών ικανός για να «μπερδέψει» τον παροχέα υπηρεσιών. Αν οι μοναδικοί χρήστες ήταν οι Χρήστης1 και Χρήστης2, δεν θα ήταν δύσκολο για τον παροχέα υπηρεσιών να παρακολουθήσει τις διαδρομές που θα ακολουθήσει καθένας από τους φαινομενικά «νέους» χρήστες και να μπορέσει να συνδέσει καθέναν από αυτούς με τους ήδη γνωστούς Χρήστης1 και Χρήστης2.



Σχήμα 4.5: Παράδειγμα μιας μικτής ζώνης.

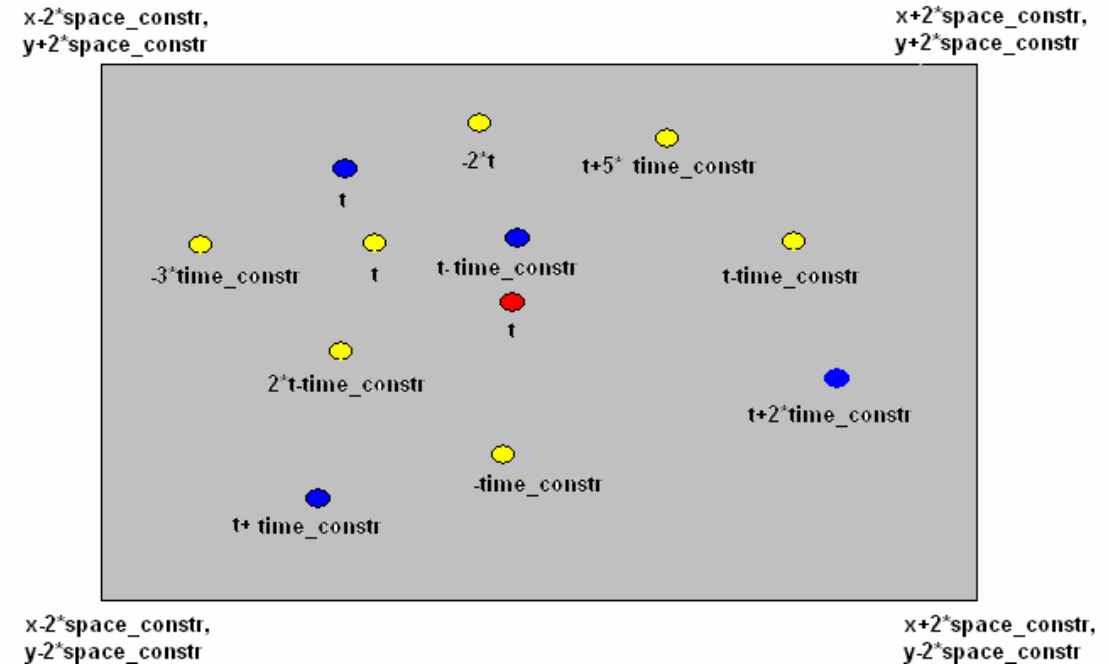
Στην περίπτωση της στρατηγικής μας δεν ορίζουμε εξαρχής περιοχές ως μικτές ζώνες, αλλά μας ενδιαφέρει να ορίσουμε τέτοιες περιοχές όποτε είναι αναγκαίο (όποτε δηλαδή κληθεί ο αλγόριθμος αποσύνδεσης). Ας θεωρήσουμε ότι με (x, y) και t, ορίζονται αντίστοιχα οι χωρικές και χρονικές συντεταγμένες που βρισκόταν ο αιτών. Με *space_constr* και *time_constr* ορίζουμε αντίστοιχα τους περιορισμούς που εισήγαγε ο αιτών σχετικά με τον βαθμό της γενίκευσης που μπορεί να εφαρμοστεί στις χωρικές και στις

χρονικές συντεταγμένες. Οι τιμές αυτές είχαν δοθεί στην είσοδο του αλγορίθμου γενίκευσης. Ως μικτή ζώνη, θεωρούμε στον αλγόριθμο αποσύνδεσης μία τετράγωνη περιοχή, της οποίας κέντρο αποτελεί το σημείο (x,y) και της οποίας άκρα είναι τα σημεία *(x + 2 * space_constr, y + 2 * space_constr), (x + 2 * space_constr, y – 2 * space_constr)*, *(x – 2 * space_constr, y - 2 * space_constr)* και *(x - 2 * space_constr, y + 2 * space_constr)*. Σε αυτή την περιοχή θα πρέπει να έχουν εμφανιστεί k διαφορετικοί χρήστες (θεωρώντας και τον αιτούντα) μέσα σε ένα συγκεκριμένο χρονικό διάστημα, προκειμένου να έχει νόημα η αλλαγή του αναγνωριστικού του αιτούντα.

Υπάρχουν δύο περιπτώσεις όπου ο αλγόριθμος αποσύνδεσης μπορεί να αποτύχει: είτε δεν μπορούν να βρεθούν k-1 διαφορετικοί χρήστες (μη συμπεριλαμβανομένου αυτού που έκανε η αίτηση), οι οποίοι έχουν περάσει από τη μικτή ζώνη οποιαδήποτε χρονική στιγμή ανήκει στο διάστημα *[t – 2 * time_constr, t + 2 * time_constr]*, είτε επειδή ο αλγόριθμος αποσύνδεσης κλήθηκε όταν η αίτηση ταίριαξε με το τελευταίο στοιχείο κάποιου LBQID. Στην τελευταία περίπτωση έχουμε αποτυχία, καθώς όλα τα προηγούμενα στοιχεία αυτού του LBQID είχαν ταιριάξει επιτυχώς με αιτήσεις, βάση του παλιού αναγνωριστικού του αιτούντα. Ακόμη και αν γίνει η αλλαγή του αναγνωριστικού, θα είναι πολύ εύκολο για τον παροχέα υπηρεσιών, να συσχετίσει τις προηγούμενες αιτήσεις του αιτούντα με αυτές που θα γίνουν μετά την αλλαγή του αναγνωριστικού του. Αν καμία από τις παραπάνω συνθήκες δεν είναι αληθής, τότε ο αλγόριθμος αποσύνδεσης επιτυγχάνει. Δίνεται στον αιτούντα ένα νέο αναγνωριστικό, το οποίο δεν έχει κανένας άλλος χρήστης, και όλα τα «μερικώς» ταιριασμένα LBQIDs του αιτούντα από προηγούμενες αιτήσεις του, διαγράφονται. Διαφορετικά, αν έστω και μία από τις παραπάνω συνθήκες είναι αληθής, ο αλγόριθμος αποσύνδεσης αποτυγχάνει και στέλνεται μήνυμα στον αιτούντα που τον ενημερώνει ότι πρέπει να είναι προσεκτικός με τις αιτήσεις που θα στείλει μελλοντικά, καθώς είναι σε ευάλωτη θέση και υπάρχει μεγάλη πιθανότητα να αποκαλυφθεί η ταυτότητα του ή άλλου είδους ευαίσθητες πληροφορίες (π.χ. η διεύθυνση της κατοικίας του, μέρη όπου συχνάζει κ.α.).

Για ευκολία στο Σχήμα 4.6, εικονίζεται ένα παράδειγμα εφαρμογής του αλγορίθμου αποσύνδεσης. Η τετράγωνη περιοχή παριστάνει μια μικτή ζώνη

με άκρα τις συντεταγμένες που ορίζονται στο σχήμα. Θεωρούμε ότι η τιμή του k είναι ίση με 5. Συνεπώς αρκεί από την μικτή ζώνη να έχουν περάσει στο επιτρεπόμενο χρονικό διάστημα (ορίστηκε παραπάνω ως το *[t–2*time_constr, t+2*time_constr]*) 4 διαφορετικοί χρήστες. Με κόκκινο συμβολίζεται ο αιτών ενώ με μπλε οι γείτονες. Με κίτρινο συμβολίζονται οι χρήστες που βρέθηκαν στην μικτή ζώνη αλλά είτε δεν ήταν οι πλησιέστεροι γείτονες, είτε βρέθηκαν εκεί σε χρονικές στιγμές εκτός των επιτρεπτών ορίων, είτε είναι χρήστες που σε κάποια προηγούμενη επανάληψη είχαν σημειωθεί ως γείτονες. Η μεταβλητή κάτω από κάθε χρήστη παριστάνει τη χρονική στιγμή όπου αυτός βρέθηκε στη μικτή ζώνη.



Σχήμα 4.6: Παράδειγμα εφαρμογής του αλγορίθμου αποσύνδεσης.

Θα πρέπει να τονίσουμε ότι για να εφαρμοστεί επιτυχώς ο αλγόριθμος γενίκευσης, δεν αρκεί μόνο να εκτελεστούν με επιτυχία τα βήματα του, αλλά πρέπει να δοθεί και ιδιαίτερη προσοχή στην επιλογή των τιμών που αποτελούν την είσοδο αυτού. Για παράδειγμα, αν οι περιορισμοί που εισάγει ο αιτών είναι ιδιαίτερα «ελαστικοί», δηλαδή ο τρισδιάστατος χώρος όπου πραγματοποιήθηκε η αίτηση μπορεί να γενικευτεί σε μεγάλο βαθμό, υπάρχει μεγάλη πιθανότητα οι k-1 γείτονες που εντοπίζονται κάθε φορά να είναι ακριβώς οι ίδιοι, ειδικά αν ο χρήστης αποστέλλει αιτήσεις σε τακτά χρονικά διαστήματα. Αυτό συμβαίνει, επειδή η περιοχή όπου ο αλγόριθμος αναζητεί

γείτονες είναι πολύ μεγάλη, και το πιο πιθανό είναι οι χρήστες που βρίσκονται εντός αυτής να μην έχουν προλάβει να μετακινηθούν εκτός των ορίων της ως την επόμενη αίτηση. Αν από την άλλη πλευρά, οι περιορισμοί επιτρέπουν μικρή γενίκευση του χρόνου και του χώρου, υπάρχει μεγάλη πιθανότητα να μην βρεθούν k χρήστες, οπότε και θα πρέπει να θεωρηθεί ότι ο αιτών έχει βρεθεί εντός μιας μικτής ζώνης. Όμως κάτι τέτοιο θα σήμαινε ότι ο αιτών δεν μπορεί να στείλει ή να λάβει αιτήσεις όσο βρίσκεται εντός αυτής της περιοχής. Εξίσου σημαντική είναι και η επιλογή του k. Μικρή τιμή του k μπορεί να σημαίνει διαρκής επιλογή ίδιων χρηστών, ενώ μεγάλη τιμή αυτού μπορεί να σημαίνει αναγκαία εφαρμογή του αλγορίθμου αποσύνδεσης και άρα παύση παροχής υπηρεσιών. Μια τεχνική για την επίλυση αυτού του προβλήματος, θα ήταν να ξεκινάμε με μια τιμή k' μεγαλύτερη από αυτή που έδωσε ο χρήστης, την οποία και θα μειώνουμε σταδιακά ωσότου φτάσουμε στην τιμή k που εισήγαγε ο αιτών. Με αυτόν τον τρόπο μειώνουμε την πιθανότητα να επιλέγονται οι ίδιοι k χρήστες για διαφορετικές αιτήσεις. Από τα παραπάνω διαφαίνεται, ότι η σωστή επιλογή του k και των περιορισμών έχει ιδιαίτερη σημασία, και το καλύτερο θα ήταν να υπάρχει κάποιου είδους πολιτική ασφαλείας, βάση της οποίας θα γίνεται και η επιλογή.

Παράδειγμα 4.3: Παράδειγμα εκτέλεσης του αλγορίθμου αποσύνδεσης

Επειδή στο Παράδειγμα 4.2 ο αλγόριθμος γενίκευσης απέτυχε, θα πρέπει να εκτελεστεί ο αλγόριθμος αποσύνδεσης. Στην τελευταία του αίτηση, ο αιτών χρήστης1, είχε θεωρήσει επιτρεπόμενα όρια για τη γενίκευση του χρόνου τις ώρες 21.30 και 01.30 για την ημέρα 08-08-2006. Συνεπώς για τον αλγόριθμο αποσύνδεσης θα θεωρήσουμε το διπλάσιο διάστημα χρόνου, δηλαδή το [19.30, 03.30]. Τα όρια για τη γενίκευση των χωρικών συντεταγμένων θα είναι τα διπλάσια από αυτά που εικονίζονται στο Σχήμα 4.3. Γνωρίζουμε ότι για να είναι επιτυχής η εφαρμογή του αλγορίθμου αποσύνδεσης, πρέπει να ισχύουν δύο συνθήκες. Η μια συνθήκη που θα πρέπει να αληθεύει είναι ότι από αυτή τη περιοχή πρέπει να έχουν περάσει 2 χρήστες οποιαδήποτε χρονική στιγμή ανήκει στο διάστημα [19.30, 03.30]. Παρατηρούμε ότι στις 22.00 ο χρήστης2, βρισκόταν στο σινεμά, ενώ ο χρήστης3 στις 19.30 βρισκόταν στο κατάστημα ρούχων, δηλαδή και οι δύο βρέθηκαν στην μικτή ζώνη στο επιτρεπόμενο χρονικό διάστημα. Άρα η πρώτη

συνθήκη είναι αληθής. Η δεύτερη συνθήκη που πρέπει να αληθεύει, είναι ότι το στοιχείο <Σινεμά [23-24]> όπου έγινε η κλήση του αλγορίθμου αποσύνδεσης δεν είναι το τελευταίο στοιχείο του 4^{ου} LBQID του χρήστη1. Ωστόσο, η συνθήκη αυτή δεν είναι αληθής και ο αλγόριθμος αποσύνδεσης αποτυγχάνει. Επειδή δεν κατορθώσαμε να βρούμε k-1 σημεία στον αλγόριθμο γενίκευσης, αλλά ούτε κατορθώσαμε να αλλάξουμε το αναγνωριστικό του χρήστη1 μέσω του αλγορίθμου αποσύνδεσης, θα πρέπει να σταλεί ενημερωτικό μήνυμα στον χρήστη1 ότι πρέπει να είναι προσεκτικός με μελλοντικές αιτήσεις, καθώς υπάρχει κίνδυνος να αποκαλυφθεί η ταυτότητά του.

4.3 Ψευδοκώδικας

Ο Αλγόριθμος 1 είναι ο αλγόριθμος γενίκευσης για χωρο-χρονικά δεδομένα με την μορφή ψευδοκώδικα. Τα δύο βασικά βήματα του αλγορίθμου βρίσκονται στις γραμμές 1 και 8. Την πρώτη φορά που θα καλέσουμε τον αλγόριθμο γενίκευσης αλλά και κάθε φορά που θέλουμε να ελέγξουμε αν η αίτηση ταιριάζει με το πρώτο στοιχείο κάποιου LBQID θα εκτελεστούν οι γραμμές 8-15. Στη γραμμή 9, αφαιρούμε από τη λίστα με τα LBQID του αιτούντα, όσων LBQID τα στοιχεία (όχι όλα) έχουν ταιριάξει με κάποια προηγούμενη αίτηση. Ονομάζουμε την λίστα που προκύπτει new_LBQID. Στη γραμμή 10 ελέγχουμε αν η τρέχουσα αίτηση ταιριάζει με το πρώτο στοιχείο, κάποιου LBQID από τη λίστα new_LBQID. Αν δεν βρεθεί κανένα τότε δεν χρειάζεται να γενικευτούν οι χωρο-χρονικές συντεταγμένες του αιτούντα (όπου βρισκόταν όταν έκανε την αίτηση) και η αίτηση αποστέλλεται ως έχει. Διαφορετικά, αποθηκεύουμε τα LBQID που βρέθηκαν στη λίστα matched_LBQID και προχωράμε με τα επόμενα βήματα. Στις γραμμές 11-13, γενικεύουμε σταδιακά τον χώρο και τον χρόνο όσο μας το επιτρέπουν οι περιορισμοί για τη γενίκευση και μέχρι να βρεθούν k-1 γείτονες. Κάθε φορά που βρίσκουμε κάποιον χρήστη, αποθηκεύουμε το αναγνωριστικό του στη λίστα k_neighbors. Στη γραμμή 14, ελέγχουμε αν μετά το πέρας του αλγορίθμου βρέθηκαν k-1 χρήστες Αν έχουν βρεθεί θέτουμε k_Anonymity = true, που σημαίνει ότι ο αλγόριθμος γενίκευσης πέτυχε, και επιστρέφουμε τη λίστα k_neighbors με τα αναγνωριστικά των k-1 γειτόνων, όπως και τη γενικευμένη περιοχή 3D_Area.

Διαφορετικά στη γραμμή 15 θέτουμε k_Anonymity = false, και διαγράφουμε όσους χρήστες είχαν τυχόν αποθηκευτεί στη λίστα k_neighbors.

Αλγόριθμος 1: Αλγόριθμος γενίκευσης χωρο-χρονικών δεδομένων

Είσοδος: - Το ακριβές σημείο και η χρονική στιγμή (x,y,t) όπου ο χρήστης έστειλε την αίτηση

- Η επιθυμητή τιμή k

- Οι περιορισμοί για την γενίκευση του χώρου και του χρόνου, space_constr και time_constr αντίστοιχα

- Η λίστα k_neighbors που περιέχει τα Id των k-1 πλησιέστερων γειτόνων (όταν η αίτηση πρέπει να ταιριάξει με ένα ενδιάμεσο στοιχείο κάποιου LBQID)

Έξοδος: - Η τρισδιάστατη γενικευμένη περιοχή 3D_Area,

- Η τιμή αληθείας για τη μεταβλητή k_Anonymity, που αντιπροσωπεύει την επιτυχία του αλγορίθμου γενίκευσης

- Η λίστα k_neighbors με τα id των k γειτόνων (όταν στην είσοδο δοθεί η παράμετρος k)

Σώμα αλγορίθμου:

1. Αν η λίστα k_neighbors δεν είναι κενή:
2. Ψάξε να βρεις αν η αίτηση ταιριάζει με το κατάλληλο στοιχείο κάποιου LBQID, από αυτά που βρίσκονται στη λίστα matched_LBQIDs. Αν βρεις προχώρα με το επόμενο βήμα. Διαφορετικά πήγαινε στο βήμα 8.
3. Όσο επιτρέπουν οι περιορισμοί space_constr, time_constr και δεν έχουν βρεθεί k-1 σημεία:
4. Γενίκευσε πρώτα τον χρόνο και έπειτα τον χώρο. Αποθήκευσε τις νέες γενικευμένες συντεταγμένες στον πίνακα 3D_Area.
5. Για κάθε χρήστη από την λίστα k_neighbors, ψάξε να βρεις αν στο PHL του, υπάρχει καταχώρηση για κάποιο τρισδιάστατο σημείο εντός της περιοχής 3D_Area. Μην λάβεις υπόψη χρήστες, για τους οποίους έχει ήδη βρεθεί σημείο σε κάποια προηγούμενη επανάληψη
6. Αν μετά το πέρας του βρόχου στο βήμα 3, έχουν βρεθεί k-1 σημεία, θέσε k_Anonymity = true, και επέστρεψε την τιμή αυτής όπως και τον πίνακα 3D_Area.
7. Διαφορετικά θέσε k_Anonymity = false. Επέστρεψε την τιμή της μεταβλητής.
8. Διαφορετικά:
9. Όρισε ως new_LBQID την λίστα με τα LBQID του αιτούντα αφαιρώντας τα στοιχεία της matched_LBQIDs.
10. Ψάξε να βρεις αν η τρέχουσα αίτηση ταιριάζει με το πρώτο στοιχείο κάποιου LBQID, από τη λίστα new_LBQID. Αν δεν βρεις τερμάτισε τον αλγόριθμο. Διαφορετικά προχώρα με το επόμενο βήμα.
11. Όσο επιτρέπουν οι περιορισμοί space_constr, time_constr και δεν έχουν βρεθεί k-1 γείτονες:
12. Γενίκευσε πρώτα τον χρόνο και έπειτα τον χώρο. Αποθήκευσε τις νέες γενικευμένες συντεταγμένες στον πίνακα 3D_Area.
13. Για κάθε υπάρχον χρήστη, ψάξε να βρεις αν στο PHL του, υπάρχει καταχώρηση για κάποιο τρισδιάστατο σημείο εντός της περιοχής 3D_Area.Αν υπάρχει αποθήκευσε το id του στη λίστα k_neighbors. Μην λάβεις υπόψη χρήστες, οι οποίοι έχουν ήδη σημειωθεί ως γείτονες σε κάποια προηγούμενη επανάληψη
14. Αν μετά το πέρας του βρόχου στο βήμα 11, έχουν βρεθεί k-1 γείτονες, θέσε k_Anonymity = true, και επέστρεψε την τιμή αυτής, όπως και τον πίνακα 3D_Area.
15. Διαφορετικά θέσε κενή τη λίστα k_neighbors, θέσε k_Anonymity = false. Επέστρεψε την τιμή της μεταβλητής.

Όταν στην είσοδο έχει δοθεί η λίστα k_neighbors με τα αναγνωριστικά των k-1 γειτόνων, θα πρέπει να ελέγξουμε αν υπάρχει κάποιο LBQID στη λίστα matched_LBQID, του οποίου το «κατάλληλο» ενδιάμεσο στοιχείο, ταιριάζει με τη τρέχουσα αίτηση. Ο έλεγχος αυτός γίνεται στη γραμμή 2. Αν βρεθεί ένα στοιχείο της λίστα matched_LBQID που να ταιριάζει ο βρόχος τερματίζει και προχωράμε στη γραμμή 3. Αν δεν βρεθεί κανένα, τότε θα πρέπει να ελέγξουμε αν η αίτηση ταιριάζει με το πρώτο στοιχείο κάποιου LBQID του αιτούντα το οποίο δεν αποτελεί μέλος της matched_LBQID και προχωράμε στη γραμμή 8. Στις γραμμές 3-5 γενικεύουμε σταδιακά τον χρόνο και τον χώρο, ώσπου να βρεθούν k-1 διαφορετικά σημεία, ένα στο PHL κάθε γείτονα. Στη γραμμή 6, μετά το πέρας του βρόχου, ελέγχουμε αν έχουν βρεθεί k-1 σημεία. Αν ο έλεγχος είναι αληθής, θέτουμε k_Anonymity = true, που σημαίνει ότι ο αλγόριθμος γενίκευσης πέτυχε και επιστρέφουμε τη γενικευμένη περιοχή 3D_Area. Διαφορετικά στη γραμμή 7 θέτουμε k_Anonymity = false.

Αν η τιμή για τη μεταβλητή k_Anonymity που επιστρέφεται είναι false, σημαίνει ότι ο αλγόριθμος γενίκευσης απέτυχε. Θα προσπαθήσουμε να επιτύχουμε k-ανωνυμία, εκτελώντας τον Αλγόριθμο 2, του οποίου τα βήματα αναλύονται με τη μορφή ψευδοκώδικα. Στόχος αυτού του αλγορίθμου είναι να αλλάξει το αναγνωριστικό του αιτούντα ώστε να μην υπάρχει καμία διασύνδεση μεταξύ των παλιών και των νέων αιτήσεων του. Στις γραμμές 1-4 γενικεύουμε σταδιακά την περιοχή όπου βρισκόταν ο αιτών όσο μας το επιτρέπουν οι τιμές 2 * space_constr και 2 * time_constr. Η περιοχή αυτή θεωρείται ότι είναι μια μικτή ζώνη. Θεωρήσαμε ότι για να μπορέσουμε να αλλάξουμε το αναγνωριστικό του αιτούντα, αρκεί στη μικτή ζώνη να υπάρχουν και άλλοι k-1 χρήστες. Η υπόθεση αυτή δεν είναι ελλιπής, δεδομένου ότι στον αλγόριθμο γενίκευσης για μια περιοχή μισού εμβαδού δεν κατορθώσαμε να βρούμε k-1 χρήστες. Η τιμή αυτή μπορεί φυσικά να αλλάξει ανάλογα με τις απαιτήσεις της κάθε υπηρεσίας. Στη γραμμή 4, ελέγχουμε αν τελικά μετά το πέρας του βρόχου κατορθώσαμε να βρούμε k-1 χρήστες (μη περιλαμβανομένου αυτού που έκανε την αίτηση). Αν ο έλεγχος αυτός είναι αληθής και αν δεν έχουν ταιριάξει όλα τα στοιχεία κάποιου LBQID που ανήκε στη λίστα matched_LBQID (η ερμηνεία της δόθηκε προηγουμένως), τότε αυτό σημαίνει ότι ο αλγόριθμος αποσύνδεσης πέτυχε. Θέτουμε k_Unlinkability =

true, δίνουμε στον αιτούντα ένα νέο, μη υπάρχον αναγνωριστικό και διαγράφουμε όλα τα στοιχεία της λίστας matched_LBQID. Διαφορετικά θέτουμε k_Unlinkability = false και ενημερώνουμε τον αιτούντα ότι πρέπει να είναι προσεκτικός με τυχόν μελλοντικές του αιτήσεις, καθώς κινδυνεύει η ασφάλεια του και μπορεί να αποκαλυφθεί η ταυτότητα του.

Αλγόριθμος 2: Αλγόριθμος αποσύνδεσης

Είσοδος: - Οι περιορισμοί για την γενίκευση του χρόνου και του χώρου time_constr και space_constr αντίστοιχα

Έξοδος: - Η τιμή αλήθειας για τη μεταβλητή k_Unlinkability που αντιπροσωπεύει την επιτυχία του αλγορίθμου αποσύνδεσης.

Σώμα αλγορίθμου:

1. Όσο επιτρέπουν οι περιορισμοί 2 * space_constr, 2 * time_constr και δεν έχουν βρεθεί k-1 χρήστες:
2. Γενίκευσε πρώτα τον χρόνο και έπειτα τον χώρο. Αποθήκευσε τις νέες γενικευμένες συντεταγμένες στον πίνακα 3D_Area.
3. Ψάξε να βρεις πόσοι χρήστες είχαν βρεθεί στην περιοχή 3D_Area και αποθήκευσε το πλήθος αυτών.
4. Αν μετά το πέρας του βρόχου στο βήμα 1, έχουν βρεθεί k-1 χρήστες (χωρίς τον αιτούντα) και δεν έχουν ταιριάξει όλα τα στοιχεία των LBQID που βρίσκονται στη λίστα matched_LBQID, προχώρα με το επόμενο βήμα. Διαφορετικά θέσε k_Unlinkability = false και ενημέρωσε τον αιτούντα ότι είναι σε κίνδυνο. Επέστραψε την τιμή της μεταβλητής.
5. Θέσε k_Unlinkability = true, διέγραψε όλα τα στοιχεία της λίστας matched_LBQID και άλλαξε το αναγνωριστικό του αιτούντα. Επέστραψε την τιμή της μεταβλητής.

5. Πειραματικά Δεδομένα

Στο παρόν κεφάλαιο θα παρουσιαστούν και θα αξιολογηθούν τα πειραματικά αποτελέσματα που προέκυψαν από την εφαρμογή της προτεινόμενης στρατηγικής (παρουσιάστηκε στο Κεφάλαιο 4), σε μία σειρά δεδομένων. Τα συγκεκριμένα πειράματα αποτελούν μία από τις βασικές συνεισφορές αυτής της εργασίας, δεδομένου ότι στο [4], όπου προτάθηκε μία αντίστοιχη στρατηγική, δεν αξιολογήθηκε η αποτελεσματικότητα της προτεινόμενης λύσης μέσω πειραματικών δεδομένων. Μέσω των αποτελεσμάτων που θα προκύψουν από την διεξαγωγή των πειραμάτων, θα μπορέσουμε να κρίνουμε αν η εφαρμογή της στρατηγικής όντως επιτυγχάνει ανωνυμία των δεδομένων και προστασία της ιδιωτικότητας των χρηστών.

Στα πλαίσια των πειραμάτων, θεωρήσαμε τη σταδιακή ύπαρξη 10, 50, 100, 500 και 1000 χρηστών, οι οποίοι μετακινούνται με τυχαίο τρόπο σε μία τετράγωνη περιοχή συνολικού εμβαδού 1000 τετραγωνικών μέτρων. Η ταχύτητα μετακίνησης των χρηστών (η οποία ορίζεται σε $^{m}/_{\text{sec}}$ και της οποίας το πεδίο ορισμού είναι το [0,10]) όπως και η κατεύθυνση που αυτοί ακολουθούν (βόρεια, νότια, ανατολικά και δυτικά) ορίστηκε με τυχαίο τρόπο. Επιπλέον, θεωρήσαμε ότι τα PHLs όλων των χρηστών περιέχουν πληροφορίες για το διάστημα ενός μήνα και ότι οι εγγραφές σε αυτά πραγματοποιούνται κάθε 15 λεπτά.

Αύξων αριθμός LBQID	**Αύξων αριθμός στοιχείου**	**Περιεχόμενα**	**Τύπος επανάληψης**
1°	1°	<800.000,345.000,805.000,350.000>, [07:00, 09:00]	3.Ημέρες * 2.Εβδομάδες
	2°	<820.000,358.000,830.000,363.000>, [09:00, 10:00]	
	3°	<835.000,360.000,840.000,370.000>, [10:00, 12:30]	
	4°	<850.000,355.000,855.000,360.000>, [12:30, 14:30]	
2°	1°	<740.000,300.000,750.000,310.000>, [12:00, 14:00]	5.Ημέρες * 1.Εβδομάδα
	2°	<760.000,300.000,780.000,310.000>, [14:00, 18:00]	
	3°	<700.000,370.000,720.000,380.000>, [19:00, 21:00]	
3°	1°	<800.000,400.000,820.000,420.000>, [21:00, 23:45]	1.Ημέρα * 4.Εβδομάδες

Σχήμα 5.1: LBQIDs του αιτούντα που θεωρήσαμε στα πλαίσια των πειραμάτων

Θεωρήσαμε την ύπαρξη ενός και μόνο αιτούντα (στο πλήθος των συνολικών χρηστών), ο οποίος απέστειλε στα πλαίσια ενός μήνα 1000 συνολικά αιτήσεις. Θεωρήσαμε ότι ο αιτών διαθέτει 3 LBQIDs, με διαφορετικό αριθμό στοιχείων το καθένα, τα οποία αναπαρίστανται στο Σχήμα 5.1.

Όσον αφορά το επίπεδο ανωνυμίας (που παρίστανται από την τιμή του k), θεωρήσαμε ότι το σύνολο τιμών για αυτό είναι το {2, 3, 5, 10, 25, 50, 75, 100}, ενώ όσον αφορά τους χωρικούς και τους χρονικούς περιορισμούς, οι οποίοι πρέπει να δοθούν σαν είσοδο στον αλγόριθμο γενίκευσης, θεωρήσαμε ότι τα σύνολα τιμών για αυτούς είναι αντίστοιχα τα: {15, 30, 120, 360} λεπτά και {100, 200, 500} μέτρα. Για μεγαλύτερη ευκολία, οι τιμές όλων των παραπάνω παραμέτρων παρουσιάζονται στο Σχήμα 5.2.

Παράμετρος	**Τιμές**
Μέγεθος περιοχής	1000 m^2
Αριθμός χρηστών	{10, 50, 100, 500, 1000}
Ταχύτητα μετακίνησης	[0, 10] m/sec
Αριθμός αιτήσεων	1000
Χρονικό διάστημα που καλύπτουν τα PHLs	1 μήνας
Συχνότητα εγγραφών	Κάθε 15 λεπτά
Επίπεδο ανωνυμίας (k)	{2, 3, 5, 10, 25, 50, 75, 100}
Περιορισμός για τη γενίκευση του χρόνου	{15, 30, 120, 360} min
Περιορισμός για τη γενίκευση του χώρου	{100, 200, 500} m

Σχήμα 5.2: Παράμετροι για τη διεξαγωγή των πειραμάτων

Οι πίνακες στα σχήματα: Σχήμα 5.3, Σχήμα 5.4, Σχήμα 5.5, Σχήμα 5.6 και Σχήμα 5.7, περιέχουν τις τιμές που δόθηκαν σαν είσοδο στη προτεινόμενη στρατηγική καθώς και το ποσοστό επιτυχίας που προέκυπτε από την εκάστοτε εφαρμογή αυτής στα δεδομένα. Παρατηρούμε ότι για ένα συνολικό πλήθος 1000 αιτήσεων, στη χειρότερη περίπτωση ο αλγόριθμος αποτύγχανε για 28 αιτήσεις, ενώ στη βέλτιστη περίπτωση αποτύγχανε για 2 μόνο αιτήσεις (όταν θεωρήσαμε 1000 συνολικά χρήστες).

Αριθμός χρηστών	Περιορισμός για την γενίκευση του χρόνου (σε λεπτά)	Περιορισμός για την γενίκευση του χώρου (σε μέτρα)	Τιμή k	Αριθμός Αποτυχιών (1000 Αιτήσεις)
10	15	100	2	27
10	30	100	2	20
10	120 (2 ώρες)	200	2	18
10	360 (6 ώρες)	500	2	14
10	15	100	3	28
10	30	100	3	28
10	120	200	3	26
10	360	500	3	25

Σχήμα 5.3: Πειραματικά αποτελέσματα για 10 χρήστες

Αριθμός χρηστών	Περιορισμός για την γενίκευση του χρόνου (σε λεπτά)	Περιορισμός για την γενίκευση του χώρου (σε μέτρα)	Τιμή k	Αριθμός Αποτυχιών (1000 Αιτήσεις)
50	15	100	2	27
50	30	100	2	19
50	120 (2 ώρες)	200	2	13
50	360 (6 ώρες)	500	2	6
50	15	100	3	28
50	30	100	3	27
50	120 (2 ώρες)	200	3	26
50	360 (6 ώρες)	500	3	24
50	15	100	5	28
50	30	100	5	28
50	120 (2 ώρες)	200	5	26
50	360 (6 ώρες)	500	5	25

Σχήμα 5.4: Πειραματικά αποτελέσματα για 50 χρήστες

Αριθμός χρηστών	Περιορισμός για την γενίκευση του χρόνου (σε λεπτά)	Περιορισμός για την γενίκευση του χώρου (σε μέτρα)	Τιμή k	Αριθμός Αποτυχιών (1000 Αιτήσεις)
100	15	100	2	22
100	30	100	2	16
100	120 (2 ώρες)	200	2	10
100	360 (6 ώρες)	500	2	6
100	15	100	3	27
100	30	100	3	26
100	120 (2 ώρες)	200	3	23
100	360 (6 ώρες)	500	3	22
100	15	100	5	28
100	30	100	5	27
100	120 (2 ώρες)	200	5	25
100	360 (6 ώρες)	500	5	23
100	15	100	10	28
100	30	100	10	28
100	120 (2 ώρες)	200	10	27
100	360 (6 ώρες)	500	10	25

Σχήμα 5.5: Πειραματικά αποτελέσματα για 100 χρήστες

Αριθμός χρηστών	Περιορισμός για την γενίκευση του χρόνου (σε λεπτά)	Περιορισμός για την γενίκευση του χώρου (σε μέτρα)	Τιμή k	Αριθμός Αποτυχιών (1000 Αιτήσεις)
500	15	100	2	12
500	30	100	2	6
500	120 (2 ώρες)	200	2	6
500	360 (6 ώρες)	500	2	5
500	15	100	3	20
500	30	100	3	18
500	120 (2 ώρες)	200	3	14
500	360 (6 ώρες)	500	3	11
500	15	100	5	26
500	30	100	5	26
500	120 (2 ώρες)	200	5	24
500	360 (6 ώρες)	500	5	18
500	15	100	10	28
500	30	100	10	26
500	120 (2 ώρες)	200	10	23
500	360 (6 ώρες)	500	10	22
500	15	100	25	28
500	30	100	25	28
500	120 (2 ώρες)	200	25	27
500	360 (6 ώρες)	500	25	26
500	15	100	50	28
500	30	100	50	28
500	120 (2 ώρες)	200	50	27
500	360 (6 ώρες)	500	50	27

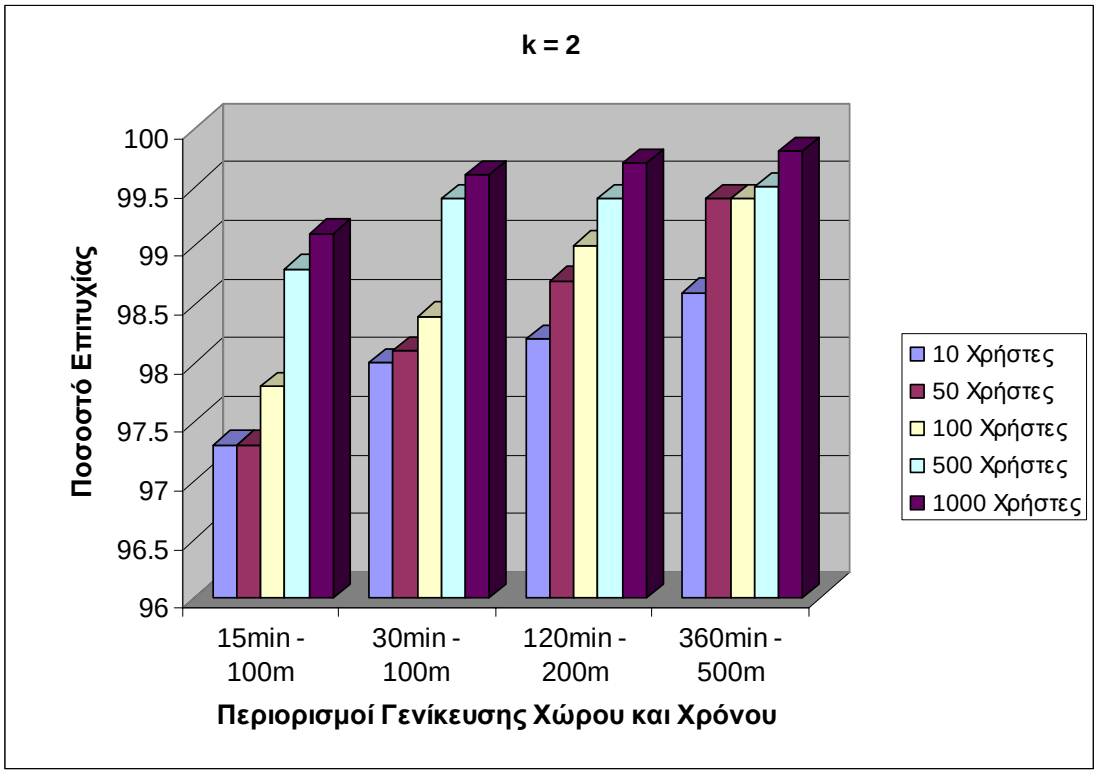
Σχήμα 5.6: Πειραματικά αποτελέσματα για 500 χρήστες

Αριθμός χρηστών	Περιορισμός για την γενίκευση του χρόνου (σε λεπτά)	Περιορισμός για την γενίκευση του χώρου (σε μέτρα)	Τιμή k	Αριθμός Αποτυχιών (1000 Αιτήσεις)
1000	15	100	2	9
1000	30	100	2	4
1000	120 (2 ώρες)	200	2	3
1000	360 (6 ώρες)	500	2	2
1000	15	100	3	14
1000	30	100	3	12
1000	120 (2 ώρες)	200	3	9
1000	360 (6 ώρες)	500	3	7
1000	15	100	5	18
1000	30	100	5	17
1000	120 (2 ώρες)	200	5	14
1000	360 (6 ώρες)	500	5	10
1000	15	100	10	24
1000	30	100	10	21
1000	120 (2 ώρες)	200	10	17
1000	360 (6 ώρες)	500	10	15
1000	15	100	25	26
1000	30	100	25	24
1000	120 (2 ώρες)	200	25	24
1000	360 (6 ώρες)	500	25	20
1000	15	100	50	28
1000	30	100	50	26
1000	120 (2 ώρες)	200	50	25
1000	360 (6 ώρες)	500	50	23
1000	15	100	75	28
1000	30	100	75	28
1000	120 (2 ώρες)	200	75	27
1000	360 (6 ώρες)	500	75	26
1000	15	100	100	28
1000	30	100	100	28
1000	120 (2 ώρες)	200	100	28
1000	360 (6 ώρες)	500	100	27

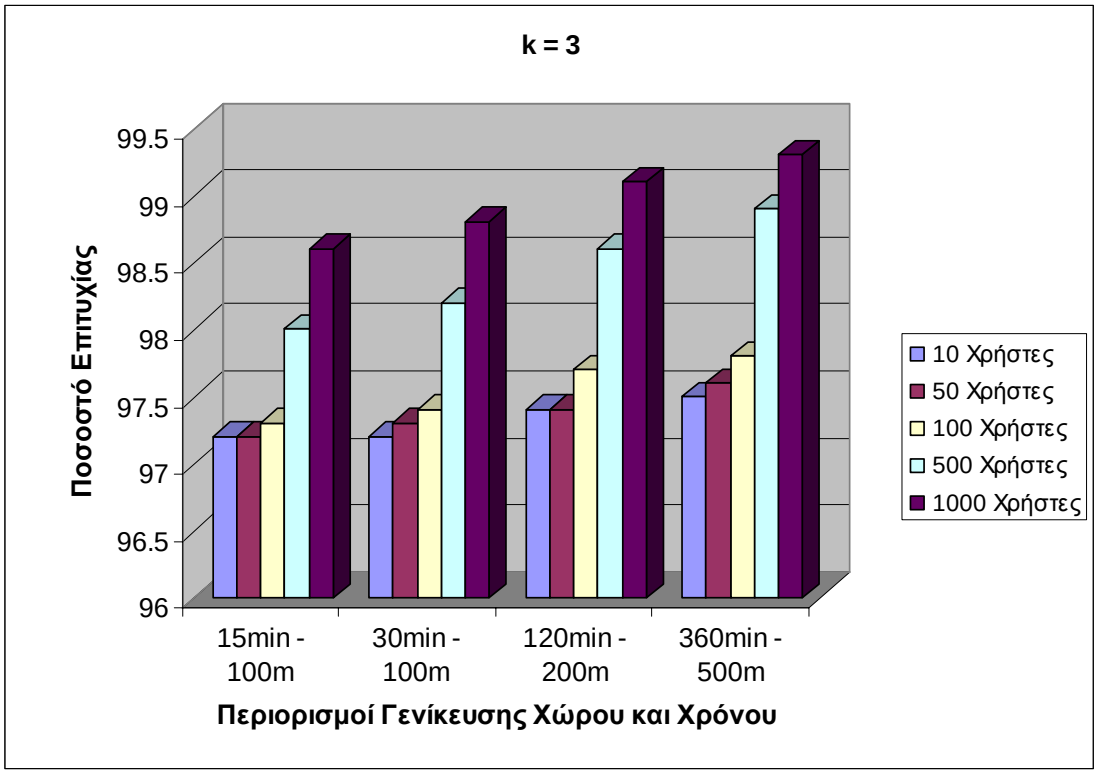
Σχήμα 5.7: Πειραματικά αποτελέσματα για 1000 χρήστες

Τα ακόλουθα σχήματα παριστάνουν τα ποσοστά επιτυχίας της στρατηγικής για έναν διαφορετικό αριθμό χρηστών και για διαφορετικές τιμές των περιορισμών, όταν θέλουμε να επιτύχουμε 2-ανωνυμία, 3-ανωνυμία, 5-ανωνυμία, 10-ανωνυμία, 25-ανωνυμία, 50-ανωνυμία, 75-ανωνυμία και 100-ανωνυμία. Τα μεγαλύτερα ποσοστά επιτυχίας προκύπτουν όταν θεωρούμε ότι ο περιορισμός για τη γενίκευση του χρόνου είναι 360 λεπτά (3 ώρες) και ότι ο περιορισμός για τη γενίκευση του χώρου είναι τα 500 μέτρα. Τα αποτελέσματα αυτά είναι αναμενόμενα, δεδομένου ότι θεωρώντας μεγαλύτερη τρισδιάστατη περιοχή (2 διαστάσεις για τον χώρο και μία για τον χρόνο),

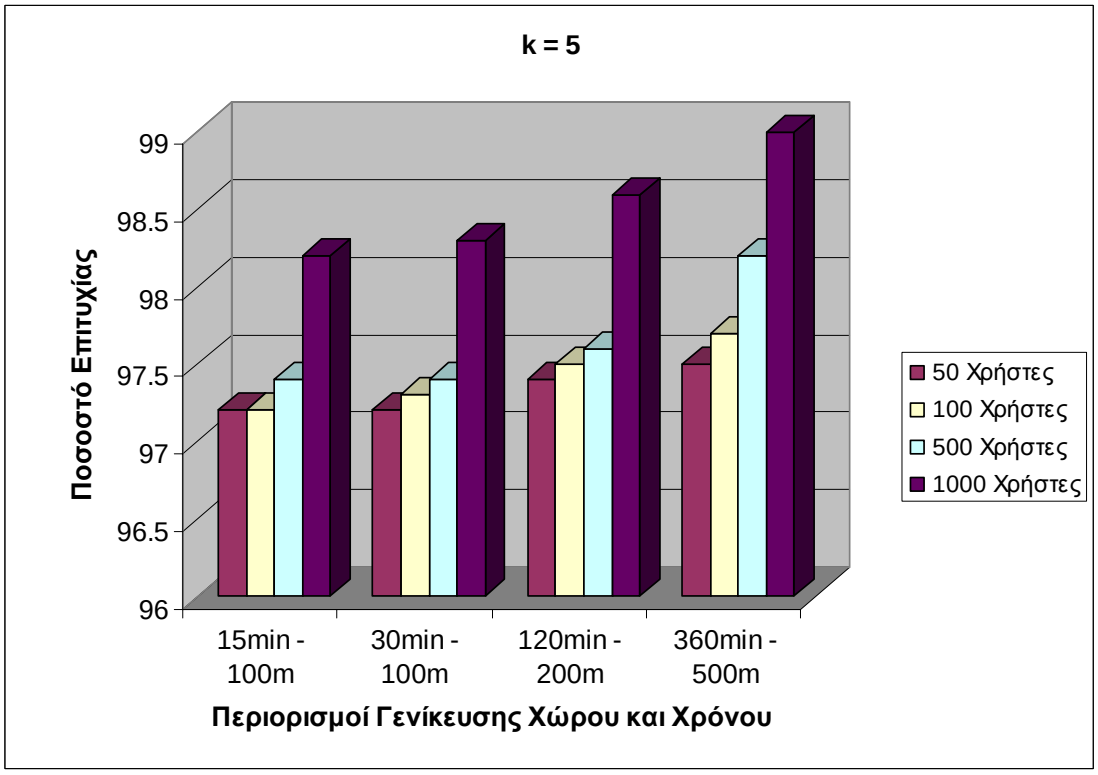
έχουμε μεγαλύτερη πιθανότητα να βρούμε τον επιθυμητό αριθμό γειτόνων απ' ότι όταν θεωρούμε μικρότερες τιμές για αυτούς τους περιορισμούς. Επιπλέον, παρατηρούμε ότι ο μέσος όρος των αποτυχιών είναι μεγαλύτερος όταν θεωρούμε ότι υπάρχουν 10 χρήστες, ενώ είναι μικρότερος όταν θεωρούμε ότι στην περιοχή υπάρχουν 1000 χρήστες. Είναι λογικό να σημειώνονται περισσότερες αποτυχίες όταν ο συνολικός αριθμός χρηστών που μετακινούνται στην περιοχή είναι μικρότερος, δεδομένου ότι σε αυτή τη περίπτωση η πιθανότητα να είχαν περάσει από το σημείο όπου έγινε η αίτηση είναι μικρότερη.



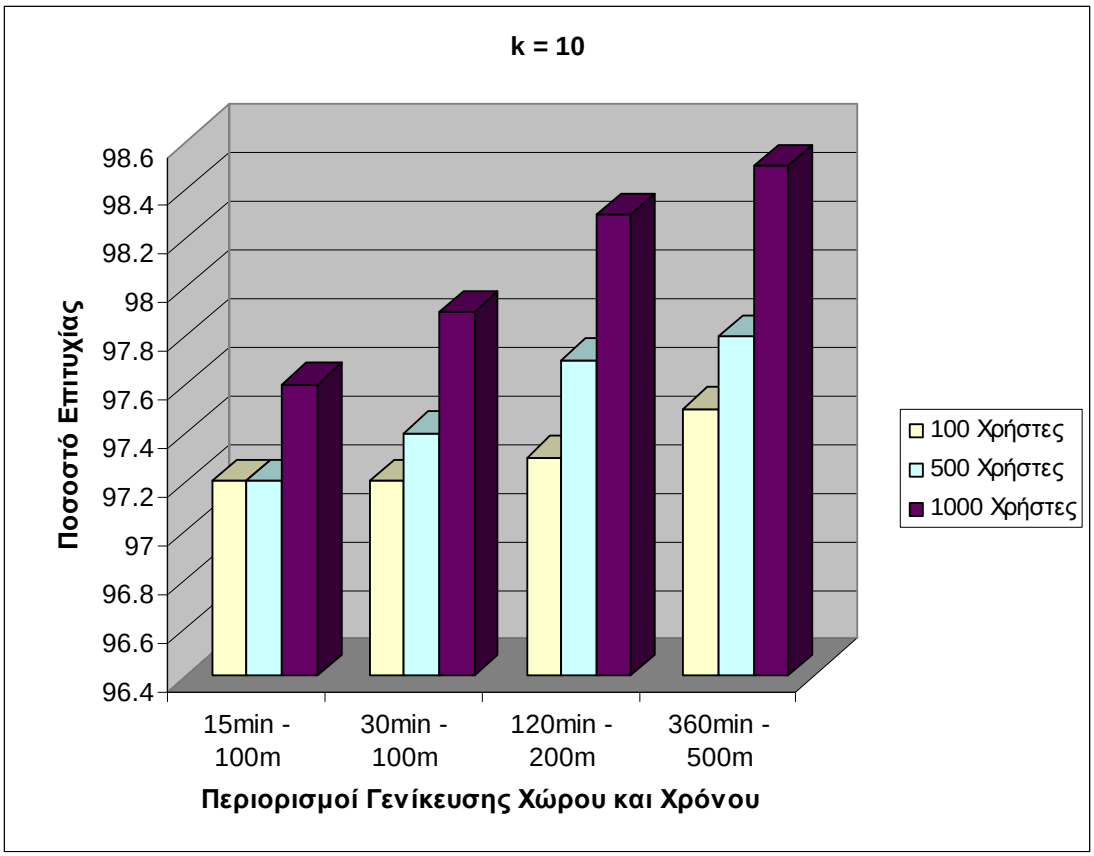
Σχήμα 5.8: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 2-ανωνυμία



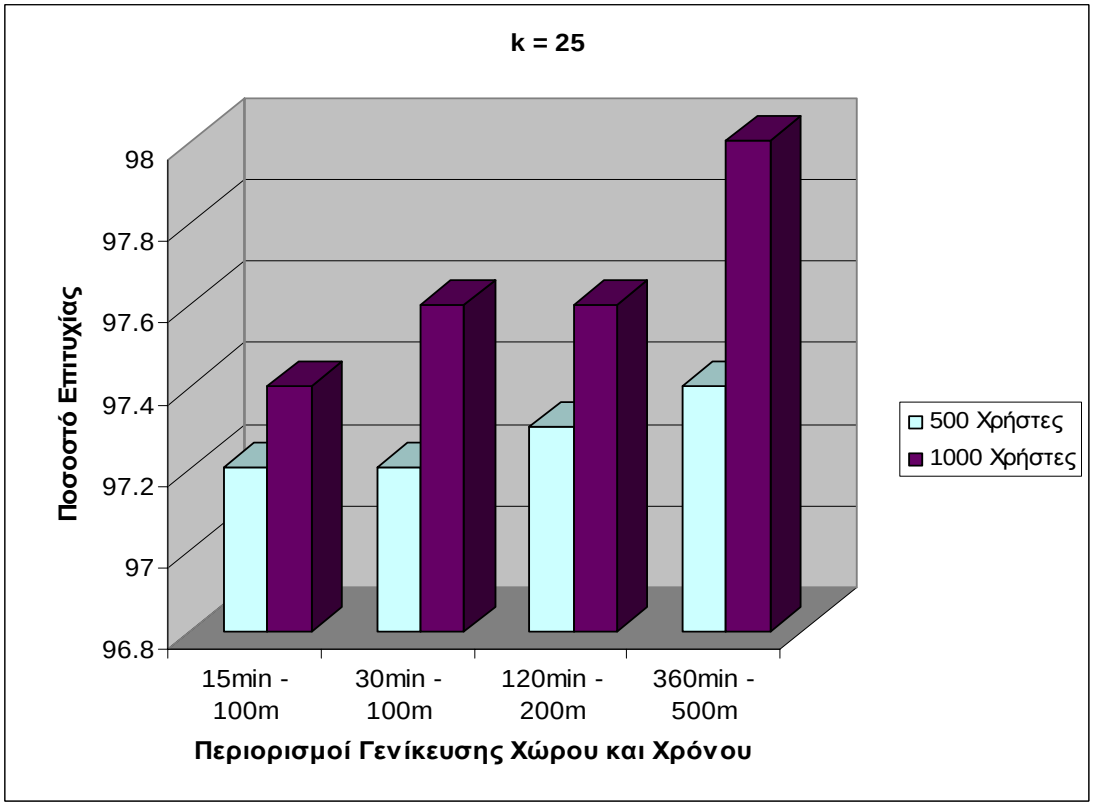
Σχήμα 5.9: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 3-ανωνυμία



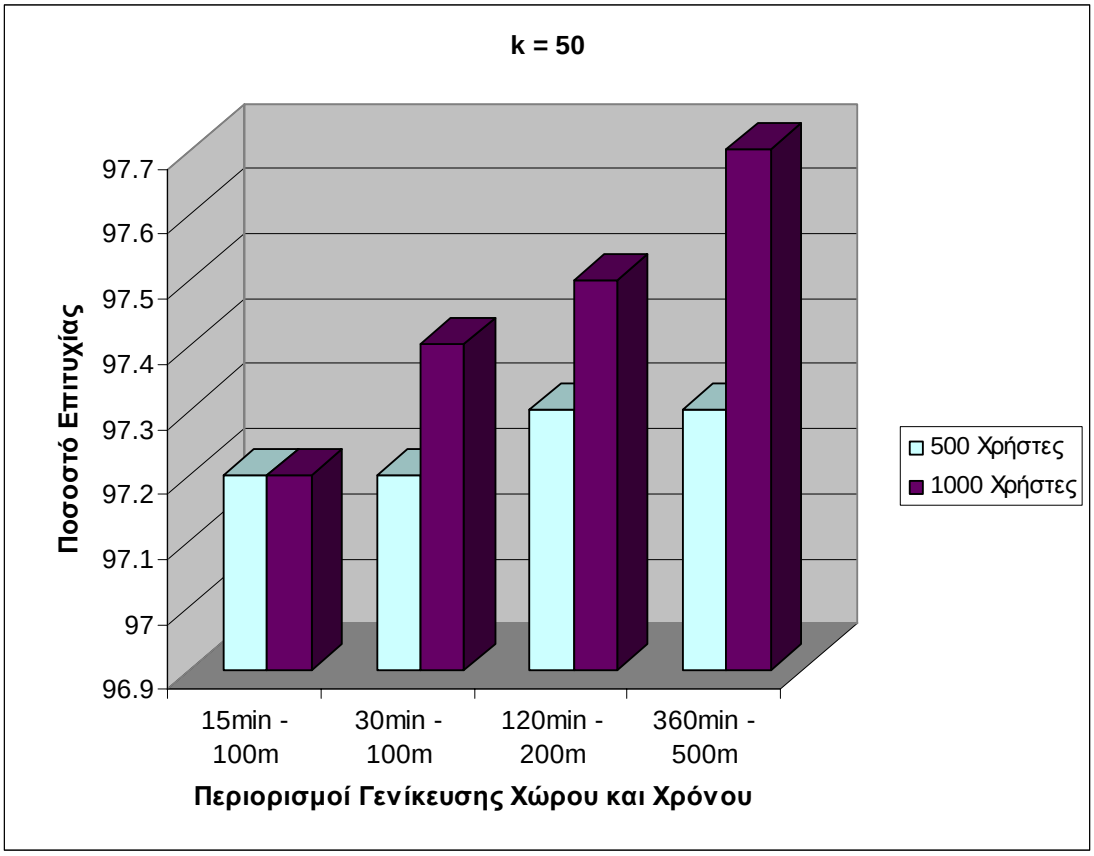
Σχήμα 5.10: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 5-ανωνυμία



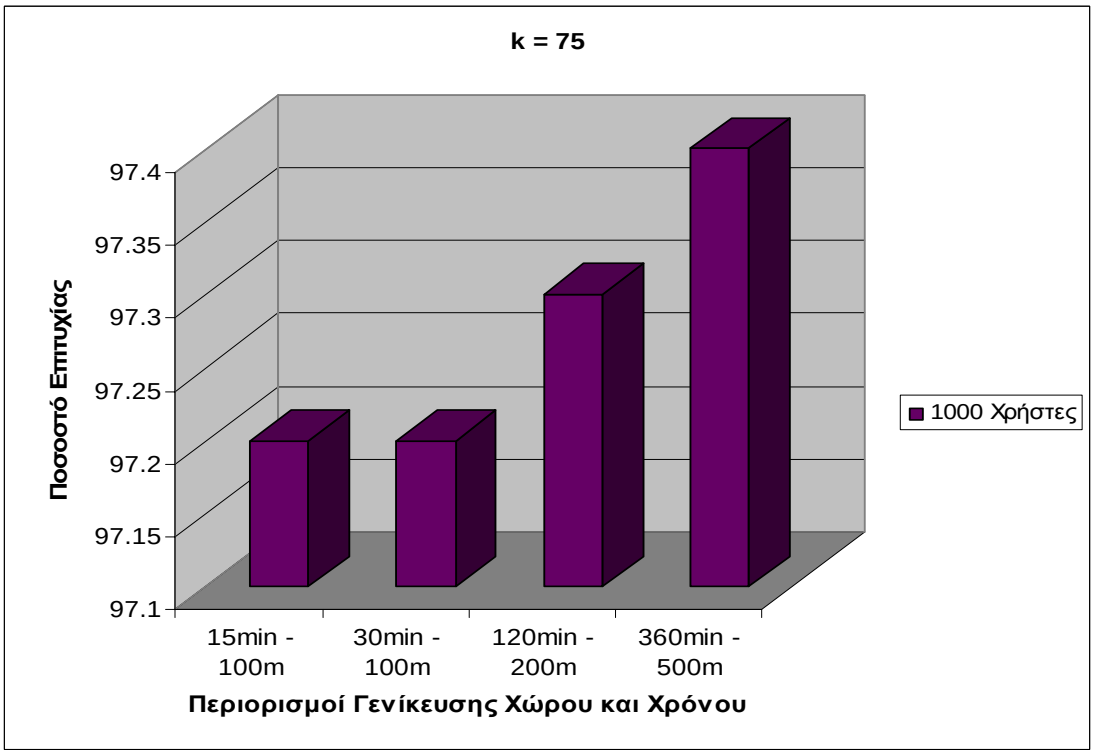
Σχήμα 5.11: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 10-ανωνυμία



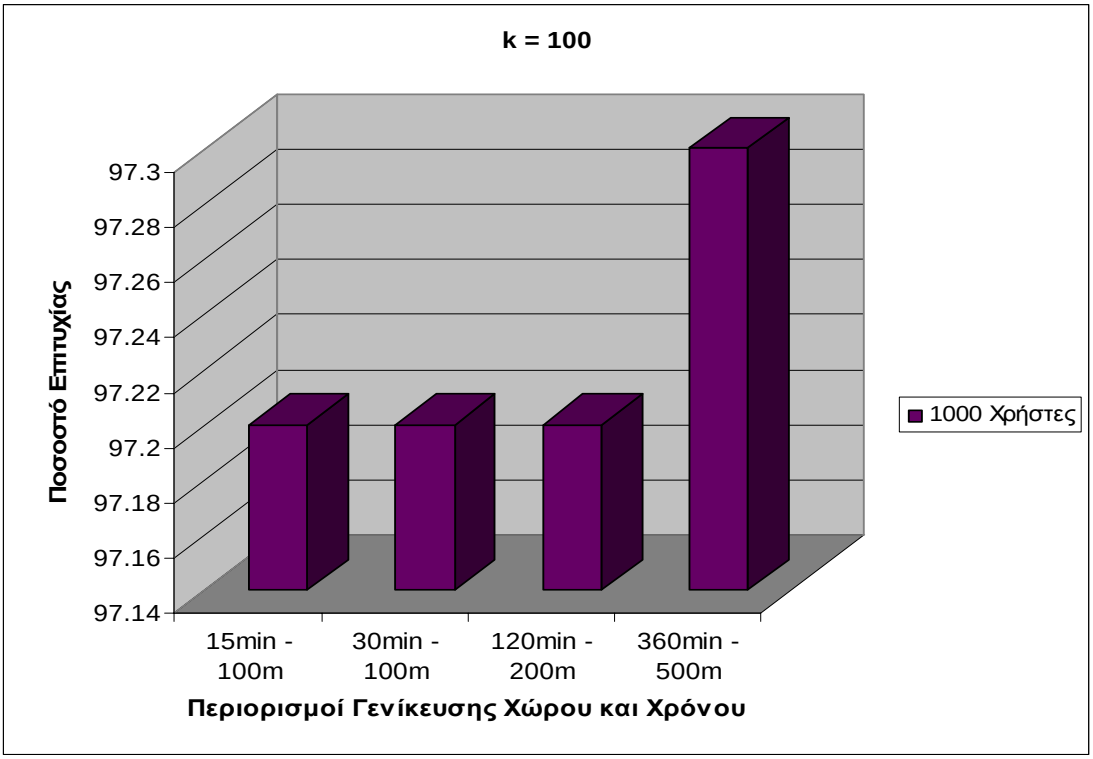
Σχήμα 5.12: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 25-ανωνυμία



Σχήμα 5.13: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 50-ανωνυμία



Σχήμα 5.14: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 75-ανωνυμία



Σχήμα 5.15: Ποσοστά επιτυχίας όταν θέλουμε να επιτευχθεί 100-ανωνυμία

Επίλογος

Στα πλαίσια αυτής της διπλωματικής εργασίας, προτάθηκε μία κατάλληλη στρατηγική, της οποίας στόχος είναι η επίτευξη της k-ανωνυμίας, δηλαδή της διασφάλισης ότι τουλάχιστον k-1 χρήστες θα έχουν βρεθεί κάποια χρονική στιγμή στην ίδια περιοχή από όπου εστάλη η αίτηση από κάποιον χρήστη. Η στρατηγική αυτή θα εφαρμόζεται στον έμπιστο εξυπηρετητή, οποτεδήποτε ένας χρήστης αποστέλλει μία αίτηση σε κάποιον παροχέα υπηρεσιών, ο οποίος χρησιμοποιεί τις χωρο-χρονικές συντεταγμένες του αιτούντα για να παράγει επιτυχώς αποτελέσματα.

Η προαναφερθείσα στρατηγική απαρτίζεται από δύο βασικούς αλγορίθμους, τον *αλγόριθμο γενίκευσης* και τον *αλγόριθμο αποσύνδεσης*. Αυτοί οι αλγόριθμοι, μέσω μιας σειράς βημάτων κατορθώνουν να προστατεύσουν όσες ευαίσθητες πληροφορίες αφορούν τον αιτούντα. Η αποκάλυψη αυτών των πληροφοριών θα είχε ως συνέπεια την παραβίαση της ιδιωτικότητας του αιτούντα και την πιθανή έκθεση του σε κίνδυνο.

Ο αλγόριθμος γενίκευσης γενικεύει σταδιακά τον τρισδιάστατο χώρο από όπου εστάλη η αίτηση, λαμβάνοντας υπόψη κάποιους περιορισμούς για τη γενίκευση, ωσότου βρεθούν άλλοι k-1 γείτονες. Σε περίπτωση αποτυχίας του αλγορίθμου γενίκευσης, εφαρμόζεται ο αλγόριθμος αποσύνδεσης. Στόχος αυτού είναι να «μπερδέψει» τον εκάστοτε παροχέα υπηρεσιών, ώστε αυτός να μην μπορεί να ξεχωρίσει τον αιτούντα από τους υπόλοιπους χρήστες, οι οποίοι είχαν βρεθεί στην ίδια μικτή ζώνη. Στα πλαίσια αυτής της εργασίας, θεωρήσαμε ότι ικανή συνθήκη για να επιτευχθεί ο παραπάνω στόχος, είναι να βρίσκονται στην ίδια μικτή ζώνη, άλλοι k-1 χρήστες. Φυσικά, ο αριθμός αυτός μπορεί να αλλάζει με βάση το είδος της υπηρεσίας, την κινητικότητα που παρατηρείται στην περιοχή όπου έγινε η αίτηση, καθώς και το «trade-off» μεταξύ της παροχής υπηρεσιών και της διατήρησης της ιδιωτικότητας. Αυτή η αναγκαία «εξισορρόπηση» προκύπτει από το γεγονός ότι όσο ο αιτών βρίσκεται σε μία μικτή ζώνη, δεν μπορεί να στείλει ή να λάβει αιτήσεις. Συνεπώς, όσο μεγαλύτερη είναι η έκταση της μικτής ζώνης, τόσο μεγαλύτερη είναι η πιθανότητα ο αλγόριθμος αποσύνδεσης να επιτύχει, αλλά ο χρήστης δεν μπορεί να έχει πρόσβαση σε οποιουδήποτε είδους υπηρεσία κατά τη διάρκεια παραμονής σε αυτή. Σε περίπτωση όπου και ο αλγόριθμος

αποσύνδεσης αποτύχει (δηλαδή δεν βρέθηκε ο απαραίτητος αριθμός χρηστών στην μικτή ζώνη), αποστέλλεται μήνυμα στον αιτούντα, το οποίο τον ενημερώνει ότι βρίσκεται σε ευάλωτη θέση και ότι η ταυτότητα του μπορεί να αποκαλυφθεί.

Μία εναλλακτική λύση θα ήταν ο έμπιστος εξυπηρετητής να «μπλοκάρει» αυτόματα τις αιτήσεις αυτού του χρήστη, ωσότου περάσει ένα ικανοποιητικό χρονικό διάστημα και αυτός δεν βρίσκεται πλέον σε κίνδυνο. Η εφαρμογή ή όχι αυτής της εναλλακτικής λύσης εξαρτάται σε μεγάλο βαθμό από το είδος των υπηρεσιών που θα ζητήσουν μελλοντικά οι χρήστες. Αν για παράδειγμα ένας χρήστης, του οποίου η ταυτότητα κινδυνεύει να αποκαλυφθεί επειδή σε κάποια προηγούμενη αίτηση του αποτύγχανε ο αλγόριθμος αποσύνδεσης, ζητήσει μία υπηρεσία η οποία δεν χρειάζεται τις χωρο-χρονικές συντεταγμένες του, τότε η αίτηση του μπορεί να σταλεί στον επιθυμητό παροχέα επειδή δεν τον θέτει σε κίνδυνο. Σε αντίθετη περίπτωση, η αποστολή αυτής της αίτησης θα επιδείνωνε την κατάσταση του αιτούντα, και για αυτό τον λόγο ο έμπιστος εξυπηρετητής θα έπρεπε να μην την προωθήσει στον αντίστοιχο παροχέα.

Μέσω μίας σειράς πειραμάτων που διεξήχθησαν για διαφορετικό αριθμό χρηστών και για διαφορετικές τιμές των παραμέτρων των αλγορίθμων, παρατηρήσαμε ότι –στη χειρότερη περίπτωση- η προτεινόμενη στρατηγική αποτύγχανε μόλις για ποσοστό 2,8%. Η τιμή αυτή προέκυπτε σε δύο περιπτώσεις: όταν θεωρούσαμε μικρό αριθμό χρηστών και όταν θεωρούσαμε μικρές τιμές για την γενίκευση που μπορεί να εφαρμοστεί στα χωρο-χρονικά δεδομένα (που αναπαριστούν τη θέση και τη χρονική στιγμή όπου βρισκόταν ο χρήστης όταν έστειλε την αίτηση). Ωστόσο, τα αποτελέσματα είναι ιδιαίτερα ενθαρρυντικά, δεδομένου ότι στα πραγματικά συστήματα ο αριθμός των χρηστών υπερβαίνει κατά πολύ τους 10, και συνήθως θεωρούμε μεγαλύτερες τιμές για τη γενίκευση του χρόνου και του χώρου.

Παρόλο που η προτεινόμενη στρατηγική κατόρθωσε να επιτύχει ανωνυμία σχεδόν σε όλες τις περιπτώσεις, εντούτοις επιδέχεται βελτιώσεις σε ορισμένα σημεία. Για παράδειγμα, οι αιτήσεις θα μπορούσαν να αποστέλλονται στους παροχείς υπηρεσιών με διαφορετική σειρά από αυτή που ελήφθησαν από τον έμπιστο εξυπηρετητή. Με αυτόν τον τρόπο, μειώνεται η πιθανότητα ύπαρξης συσχετισμού μεταξύ αιτήσεων και χρηστών.

Ακόμη, θα μπορούσε να θεωρηθεί ένας περισσότερο πολύπλοκος και αποτελεσματικός αλγόριθμος αποσύνδεσης, ο οποίος θα λάμβανε υπόψη του και άλλες παραμέτρους, όπως την τοποθεσία από όπου προήλθε η αίτηση, το είδος της αίτησης και πιθανόν κάποιο ιστορικό σχετικό με τον αιτούντα, που θα περιλάμβανε τις περιπτώσεις όπου η εφαρμογή της στρατηγικής αποτύγχανε για αυτόν στο παρελθόν.

Βιβλιογραφία

[1] A. Pfitzman, M. Kohntopp. Anonymity, Unobservability and Pseudonimity - A proposal for Terminology. 2001.

[2] J. Kim. A method for limiting disclosure of microdata based on random noise and transformation. Proceedings of the Section on Survey Research Methods of the American Statistical Association 382-387.1986.

[3] L. Sweeney, P. Samarati. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. IEEE Security and Privacy, 1998.

[4] C. Bettini, X.S. Wang, S. Jajodia. Protecting Privacy against Location-Based Personal Identification. In Proceedings of 2nd VLDB Workshop on Secure Data Management (SDM). 05.

[5] C. Bettini, S. Jajodia, X.S. Wang. Time Granularities in Databases, Data Mining and Temporal Reasoning. Springer. 2000.

[6] M. Gruteser, B. Hoh. On the anonymity of Periodic Location Samples. In Proc. Of 2nd International Conference on Security in Pervasive Computing. LNCS series, Springer. 2005

[7] B. Gedik, L. Liu. A Customizable k-Anonymity Model for Protecting Location Privacy. The 25th International Conference on Distributed Computing Systems (IEEE ICDCS), 2005.

[8] M. Gruteser, D. Grunwald, Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. In Proc.

[9] A. Beresford, F. Stajano. Location Privacy in Pervasive Computing. IEEE Pervasive Computing, 2(1): 46-55, 2003

[10] P. Samarati. Protecting respondents' identities in microdata release. IEEE Transactions on Knowledge and Data Engineering. 13(6), November/December 2001.

[11] L. Sweeney. Achieving k-Anonymity privacy protection using generalization and suppression. International Journal on Uncertainty, Fuzziness, and Knowledge-based Systems. pages 571-588.2002.

[12] J.Domingo-Ferrer and V.Torra. Ordinal, Continuous and Heterogeneous k-Anonymity through Microaggregation. Data Mining and Knowledge Discovery, pages 195–212. 2005

[13] R.Bayardo and R.Agrawal. Data privacy through optimal k-anonymity. In Proc. of the 21st Int'l Conference on Data Engineering. April 2005.

[14] K.LeFevre, D.J.DeWitt and R.Ramakrishnan. Incognito: Efficient Full Domain k-Anonymity. In ACM SIGMOD. 2005.

[15] M.Gruteser and D.Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. First International Conference on Mobile Systems, Applications, and Services (MobiSys'03). pages 31-42. May 2003.

[16] B.Gedik and L.Liu. A Customizable k-Anonymity Model for Protecting Location Privacy. The 25th International Conference on Distributed Computing Systems. IEEE ICDCS 2005.