



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

**Εξελιγμένες Τεχνικές Πρόβλεψης Θέσης στον Κινητό
Υπολογισμό**

Θεόδωρος Β. Αναγνωστόπουλος

ΑΘΗΝΑ

ΙΟΥΛΙΟΣ 2012

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Εξελιγμένες Τεχνικές Πρόβλεψης Θέσης στον Κινητό Υπολογισμό

Θεόδωρος Β. Αναγνωστόπουλος

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: Ευστάθιος Χατζηευθυμιάδης, Επίκουρος Καθηγητής
ΕΚΠΑ

ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ:

Λάζαρος Μεράκος, Καθηγητής ΕΚΠΑ

Μιχαήλ Βαζιργιάννης, Καθηγητής ΟΠΑ

Ευστάθιος Χατζηευθυμιάδης, Επίκουρος Καθηγητής ΕΚΠΑ

ΕΠΤΑΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

(Υπογραφή)

(Υπογραφή)

**Λάζαρος Μεράκος,
Καθηγητής ΕΚΠΑ**

**Μιχαήλ Βαζιργιάννης,
Καθηγητής ΟΠΑ**

(Υπογραφή)

(Υπογραφή)

**Ευστάθιος Χατζηευθυμιάδης,
Επίκουρος Καθηγητής ΕΚΠΑ**

**Δημήτριος Βαρουτάς,
Επίκουρος Καθηγητής ΕΚΠΑ**

(Υπογραφή)

(Υπογραφή)

**Ευάγγελος Ζέρβας,
Καθηγητής ΤΕΙ Αθήνας**

**Γεώργιος Ξυλωμένος,
Επίκουρος Καθηγητής ΟΠΑ**

(Υπογραφή)

**Αθανασία Αλωνιστιώτη,
Λέκτορας ΕΚΠΑ**

Ημερομηνία εξέτασης 10/07/2012

ΠΕΡΙΛΗΨΗ

Η επίγνωση-πλαisiού εμφανίζεται ως μία από τις πιο σημαντικές πτυχές στο αναδυόμενο περιβάλλον του διάχυτου υπολογισμού. Απαιτούνται κινητές εφαρμογές επίγνωσης πλαisiού για την αίσθηση και την αντίδραση σε συνθήκες μεταβαλλόμενου περιβάλλοντος. Τέτοιες εφαρμογές, συχνά, χρειάζεται να αναγνωρίζουν, να ταξινομούν και να προβλέπουν το πλαίσιο με σκοπό να δρουν αποδοτικά, εκ των προτέρων, προς όφελος του χρήστη. Πρώτον, προτείνουμε έναν αποδοτικό ταξινομητή χωρικού πλαisiού και έναν βραχείας-μνήμης προγνώστη για την μελλοντική θέση ενός κινητού χρήστη σε κυψελωτά δίκτυα. Εισάγουμε διαφορετικές παραλλαγές του προτεινόμενου προγνώστη θέσης που χρησιμοποιούν θέση (κυψέλη) και κατεύθυνση. Η ταξινόμηση συμβολικής θέσης αντιμετωπίζεται ως ένα πρόβλημα εποπτευόμενης μάθησης. Αξιολογούμε την απόδοση πρόβλεψης και την ακρίβεια των προτεινόμενων προγνωστών μέσω συνθετικών και πραγματικών ιχνών.

Δεύτερον, προτείνουμε έναν καινοτόμο προσαρμοστικό αλγόριθμο, ο οποίος χειρίζεται το πλαίσιο αναπαράστασης θέσης και την πρόβλεψη τροχιών των κινούμενων χρηστών. Ο αλγόριθμος μας υιοθετεί χωρική και χρονική on-line ομαδοποίηση, και βασίζεται στην Θεωρία Προσαρμοστικού Συντονισμού (ART) για την πρόβλεψη τροχιάς. Ο προτεινόμενος αλγόριθμος εφαρμόζει την Hausdorff απόσταση πάνω στις τροχιές για τον χειρισμό της πρόβλεψης θέσης. Δεδομένου ότι η προσέγγιση μας είναι χρονικά-ευαίσθητη, η απόσταση Hausdorff θεωρείται πιο κατάλληλη από ότι η Ευκλείδεια απόσταση. Δύο μέθοδοι εκμάθησης παρουσιάζονται και αξιολογούνται (η μη-ενισχυτική και η ενισχυτική μάθηση).

Τρίτον, προτείνουμε έναν βραχείας-μνήμης προσαρμοστικό προγνώστη θέσης που χειρίζεται την πρόβλεψη υπό την απουσία ιστορικής κινητής πληροφορίας. Ο προγνώστης μας βασίζεται σε ένα μοντέλο τοπικής γραμμικής παλινδρόμησης, ενώ η ικανότητα του να προσαρμόζεται επιτυγχάνεται μέσω ενός ασαφούς ελεγκτή. Ο ασαφής ελεγκτής αξιοποιεί το κατάλληλο μέγεθος της ιστορικής πληροφορίας κίνησης με σκοπό να ελαχιστοποιήσει το σφάλμα πρόβλεψης και να παρέχει γρήγορη προσαρμογή σε κάθε ανιχνευόμενη αλλαγή κίνησης. Τα πειράματά μας έγιναν με πραγματικά GPS ίχνη.

Τέταρτον, υποθέτουμε μία βάση προτύπων και προσπαθούμε να συγκρίνουμε το πρότυπο κίνησης ενός χρήστη με την αποθηκευμένη πληροφορία με σκοπό να προβλέψουμε μελλοντικές θέσεις. Μία συμβατική μέθοδος πρόβλεψης θα είχε μειωμένη απόδοση λόγω του θορύβου. Αυτός ο θόρυβος (α) επιδρά αρνητικά στην ικανότητα πρόβλεψης και (β) οδηγεί τη βάση σε μεγάλη αύξηση χωρητικότητας. Προσπαθούμε να ξεπεράσουμε αυτά τα προβλήματα μέσω της χρήσης της Θεωρίας Βέλτιστης Παύσης. Η συγκεκριμένη θεωρία χαλαρώνει την διαδικασία της ταξινόμησης έτσι ώστε ελαφρώς διαφορετικά πρότυπα να αντιμετωπίζονται σαν όμοια. Επιπρόσθετα, η βάση γνώσης διατηρείται αρκετά «καθαρή» διατηρώντας τα πρότυπα με την ελάχιστη χωρική διακύμανση.

Τα συμπεράσματά μας, συγκρινόμενα με άλλα σχήματα, είναι πολύ ελπιδοφόρα για το πρόβλημα της πρόβλεψης θέσης.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Πρόβλεψη Θέσης σε Περιβάλλοντα Διάχυτου Υπολογισμού

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Αλγόριθμοι Μηχανικής Μάθησης, Θεωρία Προσαρμοστικού Συντονισμού, Ασαφής Έλεγχος, Θεωρία Βέλτιστης Παύσης

ABSTRACT

Context-awareness is viewed as one of the most important aspects in the emerging pervasive computing paradigm. Mobile context-aware applications are required to sense and react to changing environment conditions. Such applications, usually, need to recognize, classify and predict context in order to act efficiently, beforehand, for the benefit of the user. Firstly, we propose an efficient spatial context classifier and a short-term predictor for the future location of a mobile user in cellular networks. We introduce different variants of the considered location predictor dealing with location (cell) identifiers and directions. Symbolic location classification is treated as a supervised learning problem. We evaluate the prediction efficiency and accuracy of the proposed predictors through synthetic and real-world traces.

Secondly, we propose a novel adaptive mobility prediction algorithm, which deals with location context representation and trajectory prediction of moving users. Machine Learning (ML) is used for trajectory classification. Our algorithm adopts spatial and temporal on-line clustering, and relies on Adaptive Resonance Theory (ART) for trajectory prediction. The proposed algorithm applies a Hausdorff-like distance over the extracted trajectories handling location prediction. Since our approach is time-sensitive, the Hausdorff distance is considered more advantageous than a simple Euclidean norm. Two learning methods (non-reinforcement and reinforcement learning) are presented and evaluated.

Thirdly, we propose a short-memory adaptive location predictor that realizes mobility prediction in the absence of extensive historical mobility information. Our predictor is based on a local linear regression model, while its adaptation capability is achieved through a fuzzy controller. Such fuzzy controller capitalizes on an appropriate size of historical mobility information in order to minimize the location prediction error and provide fast adaptation to any detected movement change. Our prediction experiments performed with real GPS data.

Fourthly, we assume the existence of a pattern base and try to compare the movement pattern of a certain user with stored information in order to predict future locations. A conventional prediction scheme would suffer from potential noise in movement patterns. Such noise (possibly manifested as small - random deviations from previously seen patterns) (a) negatively impacts the prediction capability of the classification system and (b) drives the base to state space explosion. We try to alleviate such shortcomings through the use of optimal stopping theory and the introduction of a very specific movement prediction work-flow. Optimal stopping theory relaxes the classification task so that slightly different patterns can be treated as equivalent. Moreover, the underlying pattern base is kept as 'clean' and concise as possible by retaining those patterns with limited spatial variance.

Our findings, compared with other schemes, are very promising for the location prediction problem and the adoption of proactive context-aware applications and services.

SUBJECT AREA: Location Prediction in Pervasive Computing Environments

KEYWORDS: Machine Learning Algorithms, Adaptive Resonance Theory, Fuzzy Control, Optimal Stopping Theory

Στη σύζυγό μου Γεωργία που έκανε την προσωπικότητα μου να ανθίσει και...

...στον νεογέννητο υιό μου Βασίλειο που με έκανε ξανά παιδί .

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τους γονείς μου που με έφεραν σε αυτόν τον κόσμο και με βοήθησαν, κατά το ανθρώπινο δυνατό, να γίνω αυτός που είμαι. Επίσης, ευχαριστώ τον αδερφό μου Χρήστο που μου έδειξε τη θύρα προς την ακαδημία και με βοήθησε ουσιαστικά στην εκπόνηση αυτής της διατριβής. Τέλος, ευχαριστώ θερμά το δάσκαλο μου Στάθη που με δέχτηκε και με νουθέτησε, προς όφελος της επιστημονικής μου πληρότητας, παρόλο τον ιδιαίτερα πολύπλοκο χαρακτήρα μου.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ	14
1. ΕΙΣΑΓΩΓΗ	15
1.1 Ορισμός του Προβλήματος της Πρόβλεψης Θέσης	15
1.2 Αποτελεσματική Πρόβλεψη Θέσης σε Κυψελωτά Δίκτυα.....	15
1.3 Ένας Προσαρμοστικός Αλγόριθμος Μηχανικής Μάθησης για την Πρόβλεψη Θέσης	16
1.4 Ένα Προσαρμοστικό Μοντέλο Πρόβλεψης Πρόβλεψης Θέσης βασισμένο σε Ασαφή Έλεγχο	17
1.5 Ευφυής Ταξινόμηση Τροχιάς για Βελτιωμένη Πρόβλεψη Κίνησης.....	18
2. ΑΠΟΤΕΛΕΣΜΑΤΙΚΗ ΠΡΟΒΛΕΨΗ ΘΕΣΗΣ ΣΕ ΚΥΨΕΛΩΤΑ ΔΙΚΤΥΑ	19
2.1 Ταξινόμηση τροχιάς.....	19
2.1.1 Μέθοδοι ταξινόμησης	19
2.2 Ταξινομητής Τροχιάς	19
2.2.1 Ο Μη-Παραμετρικός Ταξινομητής Τροχιάς	20
2.2.2 Ο Μη-Μετρικός Ταξινομητής Τροχιάς	21
2.2.3 Ο Συνδυαστικός Ταξινομητής Τροχιάς.....	21
2.3 Πρόβλεψη Θέσης.....	22
2.3.1 Ο Μάκρο-Προγνώστης Θέσης.....	23
2.3.1.1 Χαρακτηριστικά των MaC και MaD LPs	24
2.3.2 Ο Μίκρο-Προγνώστης Θέσης.....	26
2.4 Εκτίμηση Απόδοσης.....	27
2.4.1 Παράμετροι και Μετρικές για την Πρόβλεψη Θέσης	27
2.4.1.1 Βαθμός της Τυχαιότητας της Κίνησης	27
2.4.1.2 Πιθανότητα της Σωστής Πρόβλεψης.....	28
2.4.1.3 Στατιστική Σημαντικότητα.....	29
2.4.1.4 Υπολογιστική Πολυπλοκότητα και Αποδοτικότητα Πρόβλεψης.....	29
2.4.2 Εκτίμηση της Πρόβλεψης Θέσης	30
2.4.3 Συγκριτική Αξιολόγηση	34
2.4.3.1 Υπολογιστική Πολυπλοκότητα στην Πρόβλεψη Θέσης	34
2.4.3.2 Αποδοτικότητα στην Πρόβλεψη Θέσης.....	35
2.5 Συζήτηση σχετικά με την Οδηγούμενη από Χρόνο Πρόβλεψη Θέσης.....	37
3. ΕΝΑΣ ΠΡΟΣΑΡΜΟΣΤΙΚΟΣ ΑΛΓΟΡΙΘΜΟΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ ΓΙΑ ΤΗΝ ΠΡΟΒΛΕΨΗ ΘΕΣΗΣ	39
3.1 Μοντέλα Μηχανικής Μάθησης	39
3.1.1 Offline kMeans.....	39
3.1.2 Online kMeans	39

3.1.3	Θεωρία Προσαρμοστικού Συντονισμού	40
3.2	Αναπαράσταση Θέσης και Τροχιάς.....	40
3.3	Αλγόριθμος Πρόβλεψης Κίνησης.....	41
3.4	Αξιολόγηση Προσαρμογής και Πρόβλεψης Θέσης.....	43
3.4.1	Σύγκλιση των C-RLT και C-RLnT.....	45
3.4.2	Ακρίβεια των C-RLT και C-RLnT	46
3.4.3	Σύγκλιση των C-nRLT και C-nRLnT.....	47
3.4.4	Ακρίβεια των C-nRLT και C-nRLnT.....	48
3.5	Σύγκριση με άλλα Μοντέλα	50
4.	ΕΝΑ ΠΡΟΣΑΡΜΟΣΤΙΚΟ ΜΟΝΤΕΛΟ ΠΡΟΒΛΕΨΗΣ ΒΑΣΙΣΜΕΝΟ ΣΕ ΑΣΑΦΗ	
ΕΛΕΓΧΟ...	53
4.1	Προσαρμοστική Βραχείας-μνήμης Πρόβλεψη Θέσης	53
4.1.1	Τοπικό Γραμμικό Μοντέλο Παλινδρόμησης.....	53
4.1.2	Μοντέλο Πληροφορίας Θέσης και Τροχιάς.....	54
4.1.3	Σχεδιασμός του Προγνώστη Θέσης	55
4.2	Ασαφής Έλεγχος για Προσαρμοστική Πρόβλεψη Θέσης.....	57
4.2.1	Ασαφής Ελεγκτής.....	59
4.2.1.1	Σύμπαν του Λόγου (Universe of Discorse)	59
4.2.1.2	Γλωσσικές Μεταβλητές και Τιμές	60
4.2.1.3	Ασαφή Σύνολα και Ασαφοποίηση	60
4.2.1.4	Βάση-Κανόνων Ασαφούς Απόφασης.....	60
4.2.1.5	Αποασαφοποίηση.....	61
4.2.2	Ζώνες Ασαφούς Ελέγχου για την Πρόβλεψη της Θέσης	62
4.3	Απόδοση του Ασαφώς Ελεγχόμενου Προγνώστη Θέσης.....	66
4.3.1	Πειραματικές Τροχιάς – Ίχνη Κίνησης.....	66
4.3.2	Παράμετροι για Εκτίμηση Απόδοσης	67
4.3.3	Εκτίμηση Απόδοσης του Ασαφώς Ελεγχόμενου Προγνώστη Θέσης.....	71
4.3.3.1	Πειραματισμός με την (m+5) πρόβλεψη	73
4.3.3.2	Πειραματισμός με την (m+15) πρόβλεψη	74
4.3.3.3	Πειραματισμός με την (m+30) πρόβλεψη	76
4.3.3.4	Rlm προσαρμοστικότητα	76
4.3.4	Σύγκριση με άλλα Μοντέλα.....	77
4.3.4.1	Συγκριτική Εκτίμηση Απόδοσης.....	79
4.3.4.2	Αρχικοποίηση και Πρόβλεψη Υπολογιστικού Κόστους	82
5.	ΕΞΥΠΝΗ ΤΑΞΙΝΟΜΗΣΗ ΤΡΟΧΙΑΣ ΓΙΑ ΒΕΛΤΙΩΜΕΝΗ ΠΡΟΒΛΕΨΗ ΚΙΝΗΣΗΣ.....	84
5.1	Μοντέλο Δεδομένων	84
5.1.1	Αναπαράσταση Τροχιάς.....	84

5.1.2	Απόσταση Τροχιάς	85
5.2	Ακολουθιακή Διαδικασία Ταξινόμησης	87
5.2.1	Θεωρία Βέλτιστης Παύσης	86
5.2.2	Ακολουθιακή Ταξινόμηση.....	89
5.2.3	Διατύπωση Δυναμικού Προγραμματισμού	90
5.3	Πρόβλεψη Ακολουθίας Κίνησης	92
5.3.1	Πρόβλεψη Κίνησης	92
5.3.2	Διαδικασία Μείωσης Χωρικής Διακύμανσης	93
5.3.3	Ο Αλγόριθμος του Συστήματος Βέλτιστης Παύσης	94
5.4	Αξιολόγηση Απόδοσης.....	94
5.4.1	Μετρικές Απόδοσης.....	95
5.4.2	Παράμετροι Προσομοίωσης.....	96
5.4.3	Αξιολόγηση Απόδοσης.....	97
5.4.4	Συγκριτική Αξιολόγηση	101
6.	ΣΧΕΤΙΚΗ ΕΡΕΥΝΑ.....	108
7.	ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ	110
7.1	Αποτελεσματική Πρόβλεψη Θέσης σε Κυψελωτά Δίκτυα.....	110
7.2	Ένας Προσαρμοστικός Αλγόριθμος Μηχανικής Μάθησης για την Πρόβλεψη Θέσης	110
7.3	Ένα Προσαρμοστικό Μοντέλο Πρόβλεψης Θέσης βασισμένο σε Ασαφή Έλεγχο.....	111
7.4	Ευφυής Ταξινόμηση Τροχιάς για Βελτιωμένη Πρόβλεψη Κίνησης.....	111
7.5	Μελλοντική Έρευνα	111
	ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ	113
	ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ	114
	ΠΑΡΑΡΤΗΜΑ Ι – ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ	115
	ΠΑΡΑΡΤΗΜΑ ΙΙ – ΘΕΩΡΙΑ ΒΕΛΤΙΣΤΗΣ ΠΑΥΣΗΣ.....	132
	ΑΝΑΦΟΡΕΣ.....	135

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 2.1: Μοντέλο σχεδιασμού του TC και του LP.....	22
Σχήμα 2.2 : Ο Μάκρο-LP για ένα σύνολο από X κυψέλες και οι Μίκρο-LPs για τις x_i και x_j κυψέλες, η x_k κυψέλη ανήκει στο $V(\rho, x_i)$ και στο $V(\rho, x_j)$	25
Σχήμα 2.3 : Η πιθανότητα των σωστών προβλέψεων σε σχέση με το m για διάφορες τιμές δ	31
Σχήμα 2.4 : Η β σε σχέση με την δ τιμή για τους προτεινόμενους Προγνώστες Θέσης.....	33
Σχήμα 2.5 : Η β σε σχέση με την δ τιμή για τον MiD και τους συγκρινόμενους προγνώστες θέσης.....	36
Σχήμα 3.1 : Ο προτεινόμενος προσαρμοστικός αλγόριθμος C για προσαρμογή γνώσης και πρόβλεψη θέσης.....	42
Σχήμα 3.2 : Τα δημιουργημένα GPS αρχεία ιχνών για τον πειραματισμό.....	44
Σχήμα 3.3 : Σύγκλιση των C-RLT/-RLnT βασισμένα σε ενισχυτική μέθοδο μάθησης....	45
Σχήμα 3.4 : Ακρίβεια των C-RLT/-RLnT βασιζόμενα σε ενισχυτική μέθοδο μάθησης...	47
Σχήμα 3.5 : Σύγκλιση των C-nRLT/-nRLnT βασιζόμενα σε μη-ενισχυτική μέθοδο μάθησης.....	48
Σχήμα 3.6 : Ακρίβεια των C (-nRLT/-nRLnT) βασιζόμενα σε μη-ενισχυτική μέθοδο μάθησης.....	49
Σχήμα 3.7 : Σύγκριση του C-RLnT με τα Offline/Online kMeans μοντέλα.....	50
Σχήμα 3.8 : Η συμπεριφορά της παραμέτρου γ σε σχέση με τους χρονικούς και χωρικούς συντελεστές του κατωφλίου ευαισθησίας.....	51
Σχήμα 4.1 : Ο προσαρμοστικός βραχείας-μνήμης Προγνώστης Θέσης (LP) μοντέλο και ο ασαφής ελεγκτής.....	53
Σχήμα 4.2 : Οι βασικές παράμετροι για την $m+1$ πρόβλεψη.....	55
Σχήμα 4.3 : Μέρος της τροχιάς για πραγματικό GPS ίχνος (Ιταλία). (α) x-διάσταση της τροχιάς σε σχέση τον χρόνο, (β) y- διάσταση της τροχιάς σε σχέση τον χρόνο, συχνότητα δειγματοληψίας is 1Hz.....	56
Σχήμα 4.4 : Η ροή ελέγχου και η ανατροφοδότηση του ασαφούς ελεγκτή Plm στον προγνώστη θέσης LP.....	58

Σχήμα 4.5 : Οι συναρτήσεις συμμετοχής των εισόδων (για ασαφοποίηση) και της εξόδου (για αποασαφοποίηση) του P_{lm} και τα βασικά συστατικά του συστήματος ασαφούς λογικής.....	62
Σχήμα 4.6 : (α) οι ζώνες ελέγχου του P_{lm} ελεγκτή, (β) η επιφάνεια ελέγχου για Δm και Δe , (γ) η επιφάνεια ελεγκτή Δm και e	64
Σχήμα 4.7 : Αστικές, περιαστικές και δι-αστικές τροχιές.....	69
Σχήμα 4.8 : Πιθανότητα μίας επιτυχούς $m+1$ πρόβλεψης $P(m, l)$ με την τιμή κατωφλίου θ ($m = 10$).....	70
Σχήμα 4.9 : Πιθανότητα μίας επιτυχούς $m+1$ πρόβλεψης $P(m, l)$ σε σχέση με την τιμή κατοφλίου θ ($m = 5$).....	70
Σχήμα 4.10 : Πιθανότητα επιτυχούς πρόβλεψης για $m=10$ s και διαφορετικές l τιμές....	71
Σχήμα 4.11 : Πιθανότητα επιτυχούς πρόβλεψης σε σχέση με το $r = l/m$ για $l=5$ s.....	72
Σχήμα 4.12 : Σύγκλιση, σήματα ελέγχου και σφάλμα πρόβλεψης για δι-αστικές, αστικές και περιαστικές τροχιές για ($m + 5$) πρόβλεψη.....	72
Σχήμα 4.13 : Πιθανότητα επιτυχούς πρόβλεψης σε σχέση με το $r = l/m$ για $l=15$ s.....	73
Σχήμα 4.14 : Σύγκλιση, σήματα ελέγχου και σφάλμα πρόβλεψης για δι-αστικές, αστικές και περιαστικές τροχιές για ($m + 15$) πρόβλεψη.....	74
Σχήμα 4.15 : Πιθανότητα επιτυχούς πρόβλεψης σε σχέση με το $r = l/m$ για $l=30$ s.....	75
Σχήμα 4.16 : Σύγκλιση, σήματα ελέγχου και σφάλμα πρόβλεψης για δι-αστικές, αστικές και περιαστικές τροχιές για ($m + 30$) πρόβλεψη.....	75
Σχήμα 4.17 : Ο χρήστης αλλάζει συμπεριφορά κίνησης: περιαστική τροχιά ($m^* = 20$), δι-αστική τροχιά ($m^* = 25$), περιαστική τροχιά ($m^* = 18$), αστική τροχιά ($m^* = 6$).....	76
Σχήμα 4.18 : Ο αριθμός των ομάδων $c(n)$ στο βασισμένο με ART μοντέλο για τις τρεις τροχιές.....	78
Σχήμα 4.19 : Το $P(m, l)$ σε σχέση με το μέγεθος του m για τους διαφορετικούς τύπους τροχιών.....	80
Σχήμα 4.20 : Η $\gamma(m, l, N)$ μετρική σε σχέση με το μέγεθος του m για διαφορετικούς τύπους τροχιών.....	81
Σχήμα 5.1 : Οι P, Q, R, Y και T τροχιές του παραδείγματος.....	84

Σχήμα 5.2 : Το σύστημα: η ακολουθιακή διαδικασία ταξινόμησης, η διαδικασία πρόβλεψης κινούμενης ακολουθίας, η διαδικασία μείωσης χωρικής διακύμανσης, και η βάση γνώσης K	84
Σχήμα 5.3 : Οι τιμές απόφασης α_k σε σχέση με τα στάδια (παρατηρήσεις) $k = 1, \dots, n$, με $n = 30$	89
Σχήμα 5.4 : Η συνάρτηση απόστασης $u(P, \cdot, m)$ για τις τροχιές Q, R, T, Y , τιμές απόφασης α_m , και κατώφλι θ	92
Σχήμα 5.5 : (πάνω) Η ακρίβεια πρόβλεψης ανά Q (α) και (κάτω) η ακρίβεια πρόβλεψης (β) σε σχέση με τις τροχιές Q για $n = \{10, 20, 30, 40\}$; $\pi = 0.5$	97
Σχήμα 5.6 : (πάνω) Ο αριθμός των προτύπων $ K $ και (κάτω) η μείωση της διακύμανσης u σε σχέση με τις τροχιές Q για $n = \{10, 20, 30, 40\}$; $\pi = 0.5$	98
Σχήμα 5.7 : (πάνω) Ο αριθμός των προτύπων $ K $ και (κάτω) η μείωση της διακύμανσης u σε σχέση με τις τροχιές Q για $\pi = \{0.1, 0.5, 0.8\}$; $n = 30$	98
Σχήμα 5.8 : (πάνω) Η ακρίβεια πρόβλεψης ανά Q (α) και (κάτω) η ακρίβεια πρόβλεψης (β) σε σχέση με τις τροχιές Q για $\pi = \{0.1, 0.5, 0.8\}$; $n = 30$	99
Σχήμα 5.9 : (αριστερά) Οι μέσοι χρόνοι βέλτιστου τερματισμού (m) και (δεξιά) ο αριθμός των αντικαταστάσεων $ S $ σε σχέση με το μήκος του προτύπου n για $\pi = \{0.1, 0.5, 0.8\}$	100
Σχήμα 5.10 : Η ακρίβεια πρόβλεψης (β), αποδοτικότητα (e), μείωση διακύμανσης (u), και αριθμός των προτύπων $ K $ σε σχέση με το μήκος του προτύπου n για $\pi = \{0.1, 0.5, 0.8\}$	101
Σχήμα 5.11 : Ακρίβεια πρόβλεψης (β) σε σχέση με τον μέγιστο αριθμό των παρατηρήσεων m για (αριστερή στήλη) $n = 30$ και (δεξιά στήλη) $n = 20$ με $\pi \in \{0.1, 0.5, 0.8\}$	102
Σχήμα 5.12 : Αποδοτικότητα (e) σε σχέση με τον μέγιστο αριθμό των παρατηρήσεων m για (αριστερή στήλη) $n = 30$ και (δεξιά στήλη) $n = 20$ με $\pi \in \{0.1, 0.5, 0.8\}$	105

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 2.1 : Παράμετροι για τον Προγνώστη Θέσης	28
Πίνακας 2.2 : Μετρικές για την αξιολόγηση του Προγνώστη Θέσης.....	29
Πίνακας 2.3 : Οι πιθανότητες σωστών προβλέψεων $\alpha(LP)$ σε σχέση με το βαθμό της τυχειότητας κίνησης δ για όλους τους LPs που έχουν εκπαιδευτεί με ένα σύνολο 40-ημερών.....	30
Πίνακας 2.4 : Η πιθανότητα των σωστών προβλέψεων $\alpha(LP)$ σε σχέση με το μέγεθος του συνόλου εκπαίδευσης (σε ημέρες).....	32
Πίνακας 2.5 : Η μετρική της στατιστικής σημαντικότητας για όλους τους προτεινόμενους LPs για την α τιμή.....	32
Πίνακας 2.6 : Οι πιθανότητες των σωστών προβλέψεων σε σχέση με τον βαθμό της τυχειότητας κίνησης για τους MiD, MMP, LeZi και TAS προγνώστες.....	35
Πίνακας 3.1 : Ο υιοθετούμενος ART για τον C αλγόριθμο.....	42
Πίνακας 3.2 : Πειραματικές Παράμετροι.....	44
Πίνακας 3.3 : Σύγκριση Μοντέλων.....	50
Πίνακας 4.1 : Ασαφείς κανόνες στην Ζώνη-1: $e = \text{small}$	64
Πίνακας 4.2 : Ασαφείς κανόνες στην Ζώνη-2: $e = \text{high}$	64
Πίνακας 4.3 : Ασαφείς κανόνες στην Ζώνη-3: $\Delta e = \text{μηδέν}$ και $\text{error} = \text{high}$	65
Πίνακας 4.4 : Σύγκριση των μοντέλων πρόβλεψης.....	83
Πίνακας 5.1 : Η συνάρτηση $SVRP(P^*, Q)$	93
Πίνακας 5.2 : Ο Αλγόριθμος του Συστήματος Βέλτιστης Παύσης.....	94
Πίνακας 5.3 : Παράμετροι Συστήματος.....	96
Πίνακας 5.4 : Παράμετροι Προσομοίωσης.....	97
Πίνακας 5.5 : Ποσοστό των αποθηκευμένων προτύπων $ K $ όσον αφορά το μοντέλο μας.....	104

ΠΡΟΛΟΓΟΣ

Το θέμα της πρόβλεψης με απασχολούσε από την παιδική μου ηλικία. Συγκεκριμένα θυμάμαι φράσεις που έλεγαν οι φίλοι μου όταν παίζαμε ποδόσφαιρο και το πώς τις απομνημόνευα έτσι ώστε όταν ξανασυνέβαιναν σχετικά όμοιες καταστάσεις θυμόμουν τα προγενέστερα και με διάφορους συνδυασμούς που έκανα γνώριζα εκ των προτέρων ποια θέση θα κρατούσα έτσι ώστε να διατηρήσω την ισορροπία στην ομάδα.

Όταν τελείωσα την πρώτη μου σχολή (ΑΤΕΙ Αθήνας, Τμήμα Πληροφορικής) και έδινα κατατακτήριες εξετάσεις διάβαζα γραμμική άλγεβρα από παλιές μου σημειώσεις. Δεν θα ξεχάσω ποτέ τη στιγμή που έκανα το ίδιο ακριβώς λάθος στην αναπαράσταση ενός πίνακα που ήταν όμοιο με το λάθος που είχαν οι πρωτότυπες σημειώσεις. Τότε μου ήρθε η ιδέα να συστηματοποιήσω τον τρόπο αναπαράστασης της γνώσης μου έτσι ώστε να μαθαίνω από τα λάθη μου και να βελτιώνομαι.

Στη δεύτερη μου σχολή (ΟΠΑ (ΑΣΣΟΕ), Τμήμα Πληροφορικής) παρακολούθησα δύο μαθήματα που με στιγμάτισαν, πρώτον τις χρονολογικές σειρές και προβλέψεις και δεύτερον την εξόρυξη γνώσης από μεγάλες βάσεις δεδομένων. Ήταν η περίοδος που επιτέλους βρήκα τα εργαλεία για να κάνω τα όνειρα μου πραγματικότητα. Το μόνο που μου έλειπε ήταν η τεχνολογική κατεύθυνση που θα χρησιμοποιούσα αυτή τη γνώση. Η απάντηση ήρθε στο μεταπτυχιακό της ίδιας σχολής όπου ασχολήθηκα με θέματα ασφάλειας ασύρματων δικτύων. Τότε ήταν που συνειδητοποίησα ότι θα μπορούσα να κάνω προβλέψεις σε διαδικτυακά ασύρματα περιβάλλοντα.

Ωστόσο, αυτό έγινε πραγματικότητα όταν φοίτησα στην τρίτη μου σχολή (ΕΚΠΑ, Τμήμα Πληροφορικής και Τηλεπικοινωνιών) όπου προσέγγισα τα δίκτυα από μια πιο ανθρώπινη πλευρά, αυτή του διάχυτου υπολογισμού. Εδώ με το διαφωτισμό εκ μέρους του δασκάλου μου εστίασα στην πρόβλεψη θέσης των κινητών χρηστών σε ασύρματα δίκτυα.

Είναι πραγματικά όμορφο να γνωρίζεις εκ των προτέρων τι πρόκειται να κάνεις στο μέλλον όσον αφορά την κίνηση σου πάνω σε μία προσωπική σου τροχιά. Θα μπορούσες να προλαμβάνεις δυσάρεστες καταστάσεις ενώ ταυτόχρονα θα μπορούσες να αξιοποιήσεις εποικοδομητικά τις βέλτιστες. Παρόλα αυτά, ανακύπτουν θέματα προστασίας της προσωπικής σου ζωής αν αυτά τα δεδομένα χρησιμοποιηθούν με αθέμιτο τρόπο. Προφανώς όμως, δεν μπορούμε να λύσουμε όλα τα ζητήματα σε μία διδακτορική διατριβή. Αυτό που επιτυγχάνουμε είναι να βάλουμε ένα λιθαράκι στην πρόοδο και την εξέλιξη της επιστήμης και της τεχνολογίας. Ωστόσο, σε κάποιο μελλοντικό χρόνο θα με ενδιέφερε να ασχοληθώ με θέματα προστασίας της ιδιωτικής ζωής. Όμως αυτό αναφέρεται σε ένα επόμενο στάδιο μεταδιδακτορικής έρευνας που με βάσει τις προβλέψεις μου έχω όρεξη να ασχοληθώ.

Τέλος η παρούσα διατριβή δεν θα μπορούσε να γίνει χωρίς την ουσιαστική βοήθεια του δασκάλου μου Στάθη και του αδερφού μου Χρήστου, οι οποίοι που παρείχαν θεωρητικά εργαλεία και τεχνολογικά μέσα για να κάνω το όνειρο μου πραγματικότητα. Τους ευχαριστώ θερμά.

1. ΕΙΣΑΓΩΓΗ

1.1. Ορισμός του Προβλήματος της Πρόβλεψης Θέσης

Με σκοπό να καταστήσουμε τις κινητές εφαρμογές επίγνωσης-πλαισίου (context-aware) αρκετά ευφείς για να υποστηρίξουν τους χρήστες οπουδήποτε και οποτεδήποτε, πρέπει να αξιοποιηθεί και να επεξεργαστεί κατάλληλα το παρόν πλαίσιο των χρηστών. Ένας κοινά αποδεκτός ορισμός του πλαισίου είναι ο ακόλουθος: «πλαίσιο είναι κάθε πληροφορία που μπορεί να χρησιμοποιηθεί για να χαρακτηριστεί η κατάσταση μίας οντότητας. Μία οντότητα είναι ένα άτομο, μέρος ή αντικείμενο το οποίο θεωρείται σχετικό για την ολοκλήρωση μεταξύ ενός χρήστη και μίας εφαρμογής, περιλαμβάνοντας το χρήστη και την εφαρμογή» [1]. Το πλαίσιο αναφέρεται στις τρέχουσες τιμές συγκεκριμένων χαρακτηριστικών που αναπαριστούν την δραστηριότητα της οντότητας, της κατάστασης και της περιβαλλοντικής κατάστασης, δηλαδή, θέση, χρόνος, περπάτημα, παρουσία σε μία συνάντηση, οδήγηση ενός αυτοκινήτου, ταξίδι.

Μία από τις πιο σημαντικές ικανότητες των κινητών εφαρμογών επίγνωσης-πλαισίου είναι η *προδραστικότητα* (*proactivity*). Η πρόβλεψη των δραστηριοτήτων του χρήστη και των συστατικών του πλαισίου, γενικά, μας δίνει τη δυνατότητα να δημιουργήσουμε μία νέα κλάση εφαρμογών με την βελτίωση των ήδη υπαρχόντων. Το πιο σημαντικό συστατικό για τις κινητές εφαρμογές επίγνωσης-πλαισίου είναι η θέση του κινητού χρήστη. Η εκτίμηση και η πρόβλεψη της μελλοντικής θέσης ενός κινητού χρήστη μας δίνει την δυνατότητα να αναπτύξουμε καινοτόμες, βασισμένες στη θέση υπηρεσίες και εφαρμογές [2], [3]. Για παράδειγμα, η πρόβλεψη θέσης μπορεί να γίνει μέσο εκμετάλλευσης για να βελτιωθεί η διαχείριση πόρων στα ασύρματα δίκτυα. Αυτό διευκολύνει την παροχή των βασισμένων στη θέση υπηρεσιών με το να τις προετοιμάζει και να τις τροφοδοτεί με την κατάλληλη πληροφορία εκ των προτέρων.

1.2. Αποτελεσματική Πρόβλεψη Θέσης σε Κυψελωτά Δίκτυα

Σε αυτήν την δημοσίευση [4] (βλέπε Ενότητα 2), υιοθετούμε την ταξινόμηση προτύπων με σκοπό να προβλέψουμε τη μελλοντική θέση ενός κινητού χρήστη βασιζόμενοι στο χωρικό πλαίσιο, δηλαδή,

- Την τρέχουσα θέση και κατεύθυνση του χρήστη,
- Την ιστορία των τροχιών που ακολουθούνται από έναν χρήστη, και
- Την πληροφορία των περιχώρων του χρήστη (γειτονικές κυψέλες δικτύου).

Κάθε παρατηρούμενη τροχιά (πρότυπο) ανατίθεται σε μία συγκεκριμένη κλάση (μία συμβολική θέση) από ένα ορισμένο σύνολο κλάσεων. Ο σκοπός μας είναι να προβλέψουμε την συμβολική θέση μίας τροχιάς βασιζόμενοι σε ένα σύνολο από δημοφιλείς τροχιές. Προτείνουμε έναν Προγνώστη Θέσης (LP), ο οποίος βασίζεται σε έναν Ταξινομητή Τροχιάς (TC), με σκοπό να δοθεί λύση στο πρόβλημα της πρόβλεψης θέσης. Έχουμε σχεδιάσει, υλοποιήσει και αξιολογήσει διαφορετικές εκδόσεις του προτεινόμενου LP. Κάθε παραλλαγή εκμεταλλεύεται διαφορετικά την προερχόμενη γνώση της κινητής συμπεριφοράς του χρήστη. Ο προτεινόμενος, βασισμένος σε ταξινομητή, LP βασίζεται σε μικρής διάρκειας ιστορικά δεδομένα της κίνησης του χρήστη. Η πρόβλεψη επιτυγχάνεται μέσω του ταιριάσματος καθορισμένων (ιστορικών) προτύπων κίνησης με μίας μικρής διάρκειας ιστορίας της κίνησης του χρήστη (χωροχρονική γνώση) που παρατηρείται μέχρι τη στιγμή της πρόβλεψης. Στην περίπτωση μας, ένα πρότυπο κινητικότητας χρήστη είναι μία ακολουθία από συμβολικές θέσεις. Το μέγεθος του προτύπου κίνησης έχει άμεση επίπτωση στην ακρίβεια πρόβλεψης. Ελαχιστοποιώντας το μέγεθος της βραχείας ιστορίας του χρήστη οδηγεί σε απώλεια πληροφορίας, η οποία έχει ως αποτέλεσμα εσφαλμένες προβλέψεις και, από την άλλη πλευρά, επαναλαμβανόμενη ή υπερβολική ιστορική γνώση έχει ως

αποτέλεσμα θορυβώδη πληροφορία και προκατειλημμένες προβλέψεις. Η πρόκληση είναι να εισάγουμε μία προσέγγιση βασιζόμενοι σε (α) βραχεία χωροχρονική γνώση της κινητής συμπεριφοράς του χρήστη, και, (β) χαμηλό υπολογιστικό κόστος του προγνώστη θέσης βασιζόμενοι σε βραχεία ιστορικά δεδομένα κίνησης. Το αποτέλεσμα του προτεινόμενου LP παρέχεται μέσω αποδοτικών και χαμηλής πολυπλοκότητας αλγορίθμων μηχανικής μάθησης. Τα αποτελέσματα μας δείχνουν ένα σαφές πλεονέκτημα στην υιοθέτηση αλγορίθμων μηχανικής μάθησης για την πρόβλεψη της θέσης καθώς παρατηρούμε μία αυξημένη ακρίβεια πρόβλεψης (πιθανότητα σωστής πρόβλεψης) σε ένα λογικό υπολογιστικό κόστος. Στη συνέχεια, παρουσιάζουμε μία συγκριτική ανάλυση του προτεινόμενου LP με διάφορους δημοφιλείς αλγορίθμους που αναφέρονται στη διεθνή λογοτεχνία.

1.3. Ένας Προσαρμοστικός Αλγόριθμος Μηχανικής Μάθησης για την Πρόβλεψη Θέσης

Σε αυτή τη δημοσίευση [5] (βλέπε Ενότητα 3), προτείνουμε έναν προσαρμοστικό αλγόριθμο μηχανικής μάθησης ο οποίος,

1. Ομαδοποιεί τις κινήσεις του χρήστη,
2. Προσδιορίζει τις αλλαγές στις κινήσεις του χρήστη,
3. Προσαρμόζει τη δομή της γνώσης του σε αυτές τις αλλαγές, και,
4. Προβλέπει τη μελλοντική θέση.

Συγκεκριμένα, εισάγουμε δύο μεθόδους εκπαίδευσης για να εκπαιδεύσουμε τον αλγόριθμο μας: (α) την «σχεδόν» μηδενική-γνώση και (β) την *εποπτευόμενη* μέθοδο. Στην πρώτη μέθοδο, ο αλγόριθμος μας εκπαιδεύεται σταδιακά ξεκινώντας με μία μικρή γνώση της κινητής συμπεριφοράς του χρήστη. Στην δεύτερη μέθοδο, σύνολα από γνωστές τροχίες τροφοδοτούν τον αλγόριθμο μας. Επιπρόσθετα, εισάγουμε δύο μεθόδους εκμάθησης για τον προτεινόμενο αλγόριθμο όσον αφορά την επιτυχία της πρόβλεψης θέσης: (α) τη *μη-ενισχυτική μάθηση* (nRL) και (β) την *ενισχυτική μάθηση* (RL). Στην nRL μέθοδο, μία εσφαλμένα ταξινομημένη τροχιά δεν χρησιμοποιείται περαιτέρω στην φάση εκπαίδευσης του μοντέλου. Ως εκ τούτου, ο αλγόριθμος δεν έχει πλέον επίγνωση των ανεπιτυχών προβλέψεων. Στην RL μέθοδο, μία εσφαλμένα ταξινομημένη τροχιά εισάγεται στην βάση γνώσης, έτσι, επιτυγχάνεται η κατάλληλη ενημέρωση του μοντέλου.

Αξιολογούμε την απόδοση του αλγορίθμου μας με τροχίες πραγματικής κίνησης χρήστη. Καθιερώνουμε σημαντικές μετρικές για την διαδικασία εκτίμησης απόδοσης σε σχέση με (α) χαμηλές απαιτήσεις-συστήματος, δηλαδή, χωρητικότητα αποθήκευσης, και (β) την απαιτούμενη προσπάθεια για την δημιουργία του συστήματος, δηλαδή, επεξεργαστική ισχύ. Εκτιμούμε την ακρίβεια πρόβλεψης του αλγορίθμου μας, δηλαδή, την ακρίβεια των προβλέψεων θέσης, σε σχέση με το απαιτούμενο μέγεθος της αντίστοιχης βάσης γνώσης. Αυτό το μέγεθος υποδηλώνει την αναλογία των παραγόμενων ομάδων σε σχέση με το σύνολο των προτύπων εκπαίδευσης. Βέβαια, πρέπει να κρατήσουμε τη χωρητικότητα αποθήκευσης όσο πιο χαμηλά γίνεται ενώ παράλληλα πρέπει να επιτύχουμε καλή ακρίβεια πρόβλεψης. Επίσης μελετάμε και την ικανότητα του αλγορίθμου μας να προσαρμόζει το μοντέλο σε νέα πρότυπα. Η *προσαρμοστικότητα* υποδηλώνει την ικανότητα του προτεινόμενου αλγορίθμου να ανιχνεύει και να ενημερώνει κατάλληλα ένα συγκεκριμένο μέρος του εκπαιδευόμενου μοντέλου. Ο αλγόριθμος πρέπει να ανιχνεύει γρήγορα αλλαγές στην συμπεριφορά του κινητού χρήστη και να προσαρμόζει αντίστοιχα τη βάση γνώσης μέσω διαδοχικών ενημερώσεων του μοντέλου. Αυτό, ωστόσο, συχνά επιτυγχάνεται σε βάρος της ακρίβειας πρόβλεψης.

1.4. Ένα Προσαρμοστικό Μοντέλο Πρόβλεψης Θέσης βασισμένο σε Ασαφή Έλεγχο

Σε αυτή τη δημοσίευση [6] (βλέπε Ενότητα 4), προτείνουμε μία προσέγγιση για πρόβλεψη κίνησης εκμεταλλευόμενοι μόνο την τρέχουσα χωροχρονική γνώση της κίνησης του χρήστη. Στόχος μας είναι η αντιμετώπιση του προβλήματος της πρόβλεψης κίνησης στην περίπτωση απουσίας μεγάλης ιστορικής πληροφορίας κίνησης για τον εν λόγω χρήστη. Η προτεινόμενη προσέγγιση πρόβλεψης κίνησης μπορεί να εφαρμοστεί οποτεδήποτε η συμπεριφορά κίνησης ενός χρήστη δεν είναι γνωστή εκ των προτέρων. Για παράδειγμα, θεωρείστε έναν τουρίστα που έχει επισκεφθεί μία πόλη ή έναν πρωτοετή φοιτητή σε μία πανεπιστημιούπολη. Ωστόσο, λόγω της απουσίας προηγούμενων προτύπων κίνησης, η πρόβλεψη κίνησης πρέπει να προσαρμόζεται τακτικά σε νέες κινήσεις του χρήστη. Το προτεινόμενο μοντέλο μπορεί, με αρκετή ακρίβεια, να προβλέπει την τροχιά της κίνησης. Αυτό επιτυγχάνεται με τη χρήση και τη δυναμική προσαρμογή μόνο τοπικής τρέχουσας χωροχρονικής γνώσης της συμπεριφοράς κίνησης του χρήστη έχοντας πολύ μικρή σχετική ιστορική πληροφορία. Ο κύριος σκοπός της προσέγγισης μας είναι να παρακάμψουμε τις δυσκολίες που εγείρονται κατά την πρόβλεψη της μελλοντικής θέσης του χρήστη όταν δεν είναι διαθέσιμη εκτεταμένη γνώση της ιστορίας των προτύπων κίνησης του χρήστη ή όταν ο χρήστης συμπεριφέρεται αρκετά τυχαία. Μία άλλη πρόκληση είναι ο καθορισμός της χωρητικότητας της σχετικής ιστορικής πληροφορίας που απαιτείται για την πρόβλεψη κίνησης. Η χωρητικότητα της σχετικής ιστορικής πληροφορίας ποικίλει μεταξύ των κινητών χρηστών, δεδομένου ότι αυτοί παρουσιάζουν διαφορετικές συμπεριφορές κίνησης. Επιπρόσθετα, είναι ενδιαφέρον ότι η χωρητικότητα της σχετικής ιστορικής πληροφορίας είναι χρονικά μεταβαλλόμενη για έναν χρήστη. Αυτό ισχύει διότι ο κινητός χρήστης μπορεί να μεταβάλει απότομα τη συμπεριφορά κίνησης του, γεγονός το οποίο πρέπει να αντανakλάται στη σχετική ιστορική πληροφορία με σκοπό το σύστημα πρόβλεψης κίνησης να προσαρμοστεί σε νέες αλλαγές.

Συγκεκριμένα, εισάγουμε τον σχεδιασμό και την υλοποίηση ενός *προσαρμοστικού, βραχείας-μνήμης* Προγνώστη Θέσης (LP). Ο LP δεν απαιτεί μία εκτεταμένη βάση γνώσης από προηγούμενες κινήσεις του χρήστη. Αντίθετα ο LP εκτιμά την μελλοντική θέση του κινητού χρήστη βασιζόμενος σε βραχεία πληροφορία κίνησης. Το οποίο σημαίνει ότι ο LP εκμεταλλεύεται μόνο τοπική χωροχρονική γνώση της κίνησης του χρήστη προσπαθώντας να προβλέψει τις μελλοντικές θέσεις στο κοντινό μέλλον. Προφανώς, αυτό προϋποθέτει ότι ο LP προσαρμόζει δυναμικά το μοντέλο του για την επίτευξη μελλοντικών αποφάσεων στην εκτίμηση θέσης. Δεδομένου ότι δεν υπάρχουν αποθηκευμένα ιστορικά πρότυπα ή άλλη γνώση αναπαράστασης της συμπεριφοράς κίνησης του χρήστη, ο LP πρέπει επίσης να είναι αποδοτικός ακόμα και στην περίπτωση τυχαίων (αυθόρμητων) αλλαγών κίνησης, δηλαδή, ξαφνικές στροφές. Από την άλλη πλευρά, ο LP πρέπει να φθάσει σε μία σταθερή κατάσταση (μοντέλο) οποτεδήποτε ο χρήστης δεν αλλάζει την συμπεριφορά κίνησης του για μεγάλο χρονικό διάστημα. Ως εκ τούτου, ο LP πρέπει να είναι ικανός να ανιχνεύει ταχύτατα αλλαγές στην συμπεριφορά κίνησης και να προσαρμόζει το μοντέλο του στην *τρέχουσα* συμπεριφορά, έτσι ώστε, να παρέχει γρήγορη προσαρμογή του μοντέλου.

Τα κριτήρια που καθιερώνουμε για την εκτίμηση απόδοσης του προσαρμοστικού LP λαμβάνουν υπόψη τις απαιτήσεις του συστήματος (χωρητικότητα αποθήκευσης) και το υπολογιστικό κόστος για πρόβλεψη και γρήγορη προσαρμογή. Εκτός από την ακρίβεια πρόβλεψης, δηλαδή, την ακρίβεια των προβλέψεων θέσης, ενδιαφερόμαστε επίσης για το μέγεθος / χωρητικότητα της πληροφορίας που ο LP πρέπει να επεξεργαστεί για να κάνει προβλέψεις. Πρέπει να αναφέρουμε ότι στην περίπτωση μας, η καταναλωθείσα

μνήμη είναι εξαιρετικά μικρή αν την συγκρίνουμε με άλλες προσαρμοστικές τεχνικές για πρόβλεψη θέσης. Εν τέλει, ο στόχος μας είναι να εκτιμήσουμε την προσαρμοστική συμπεριφορά του LP που είναι η ικανότητα του να ανιχνεύει τάχιστα αλλαγές στην κίνηση του κινητού χρήστη και να αντιδρά αντίστοιχα.

1.5. Ευφυής Ταξινόμηση Τροχιάς για Βελτιωμένη Πρόβλεψη Κίνησης

Σε αυτήν την δημοσίευση [7] (βλέπε Ενότητα 5), έχουμε ως στόχο να αυξήσουμε την απόδοση πρόβλεψης με το να (α) βελτιώσουμε την ακρίβεια καθώς και να (β) απλοποιήσουμε (καθαρίσουμε) την προϋπάρχουσα γνώση πάνω στην οποία βασίζεται το μοντέλο πρόβλεψης. Τα ίχνη κίνησης είναι πάντα αρκετά θορυβώδη και, ως εκ τούτου, υπονομεύουν την απόδοση των μηχανισμών κινητού υπολογισμού. Το γεγονός αυτό μας παρακίνησε να εισάγουμε ένα ολιστικό πλαίσιο που μετριάξει την αρνητική επίπτωση της αυθόρμητης τυχαιότητας το οποίο μπορεί να χρησιμοποιηθεί στην (ως επί των πλείστον τακτική) συμπεριφορά κίνησης των ανθρώπων ή των οχημάτων.

Το πλαίσιο μας διαχειρίζεται όλον τον κύκλο-ζωής μίας ακολουθιακής ταξινόμησης τροχιάς και ενός μοντέλου πρόβλεψης της κίνησης, λειτουργεί σε έναν σχετικά αμνήμον (stateless) τρόπο και προσπαθεί να διατηρήσει την βάση γνώσης K όσο «καθαρή» και μειωμένη γίνεται. Ο όρος «καθαρή» αναφέρεται στις διαστάσεις του χρόνου και του χώρου, δηλαδή, τα περιεχόμενα της K έχουν ελάχιστη χωροχρονική διακύμανση. Ο προτεινόμενος ταξινομητής τροχιάς πρέπει να παρουσιάζει κάποια χρονική ανεκτικότητα (δηλαδή, υπομονή – καθυστέρηση) προκειμένου να ταξινομήσει μία τροχιά σε ένα υπάρχον (αποθηκευμένο) πρότυπο ή να εγκαταλείψει την διαδικασία ταιριάσματος ομοιότητας και να διαχειριστεί την εκτιμώμενη τροχιά ως εντελώς νέα. Εισάγουμε μία μετρική ομοιότητας τροχιάς, η οποία παρέχει μία καθολική άποψη του πόσο όμοια είναι ένα αποθηκευμένο πρότυπο και μία παρατηρούμενη τροχιά. Στη συνέχεια ο ταξινομητής συμπεραίνει, μέσω μίας διαδικασίας μείωσης χωρικής διακύμανσης (βασισμένη σε χωρικές στατιστικές μετρικές) αν το τρέχον πρότυπο πρέπει να διατηρηθεί στην K . Αν η νέα (αξιολογηθείσα) τροχιά παρουσιάζει χαμηλότερη χωρική διακύμανση από ότι το βέλτιστο πρότυπο τότε το τελευταίο αντικαθίσταται από την πρώτη. Αλλιώς η K παραμένει χωρίς τροποποιήσεις. Η συνεισφορά της δημοσίευσης είναι:

- Ένα κλειστού-βρόχου, ανεξάρτητο από την τοπολογία της περιοχής κίνησης, σύστημα πρόβλεψης σφάλματος-προσαρμογής για ακολουθιακή ταξινόμηση τροχιάς και k – βημάτων μελλοντική πρόβλεψη θέσης,
- Μία μετρική ομοιότητας τροχιάς η οποία υποστηρίζει την ακολουθιακή ταξινόμηση,
- Μία προσέγγιση βασισμένη σε δυναμικό προγραμματισμό (κανόνες βέλτιστου τερματισμού) για ακολουθιακή ταξινόμηση τροχιάς,
- Ένα αυξητικό και προσαρμοστικό μοντέλο εκμάθησης κινούμενων προτύπων. Το σύστημα δεν απαιτεί φάση εκπαίδευσης και αυξητικά τελειοποιεί την βάση γνώσης για να βελτιώσει τα αποτελέσματα πρόβλεψης,
- Μία διαδικασία μείωσης χωρικής διακύμανσης / τελειοποίησης για την βάση γνώσης. Η K περιέχει όσο πιο καθαρά πρότυπα είναι δυνατόν, ως εκ τούτου, αποφεύγει αποφάσεις ταξινόμησης που βασίζονται σε θορυβώδη πρότυπα.

2. ΑΠΟΤΕΛΕΣΜΑΤΙΚΗ ΠΡΟΒΛΕΨΗ ΘΕΣΗΣ ΣΕ ΚΥΨΕΛΩΤΑ ΔΙΚΤΥΑ

2.1. Ταξινόμηση τροχιάς

2.1.1. Μέθοδοι ταξινόμησης

Η διαδικασία της ταξινόμησης αποτελείται από δύο φάσεις: τη φάση της μάθησης και της δοκιμαστική φάση. Κατά τη φάση της μάθησης, ο ταξινομητής δημιουργείται, δηλαδή, ταιριάζει ιστορικά δεδομένα σε παραμέτρους και δομές δεδομένων. Στη δοκιμαστική φάση, ο ταξινομητής χρησιμοποιείται για την πρόβλεψη της κλάσης ενός άγνωστου προτύπου. Υπάρχουν δύο βασικές μέθοδοι για την ανάπτυξη ενός ταξινομητή: οι *παραμετρικές* και οι *μη-παραμετρικές* μέθοδοι [8], [9]. Η παραμετρική μέθοδος δημιουργεί ένα ταξινομητή από παραδείγματα εκπαίδευσης που ακολουθούν την κανονική κατανομή. Ωστόσο, στην περίπτωση μας δεν υπάρχει καμία υπόθεση σχετικά με την κατανομή των δεδομένων εκπαίδευσης. Στη μη-παραμετρική μέθοδο, η διαδικασία της ταξινόμησης ασχολείται με αυθαίρετες κατανομές. Ένα χαρακτηριστικό παράδειγμα αυτής της μεθόδου είναι ο ταξινομητής *k*-πλησιέστερου γείτονα (*k*NN). Μπορεί να γίνει διάκριση μεταξύ των *μετρικών* και των *μη-μετρικών* μεθόδων για την δημιουργία ενός ταξινομητή ανάλογα με τη φύση των ταξινομημένων δεδομένων. Η μετρική μέθοδος εφαρμόζεται σε δεδομένα πραγματικών-τιμών, ενώ η μη-μετρική μέθοδος που εφαρμόζεται σε ονομαστικά δεδομένα. Τέλος, η μέθοδος *συνδυασμού πολλαπλών-εμπειρογνωμόνων* περιλαμβάνει ταξινομητές βάσης που εκτελούνται παράλληλα (όπου κάθε ένας από αυτούς κάνει διαφορετικές υποθέσεις για τα δεδομένα). Η τελική απόφαση, (δηλαδή, πρόβλεψη) διαμορφώνεται από την συγχώνευση των αποφάσεων του κάθε ταξινομητή μέσω ενός συστήματος ψηφοφορίας.

2.2. Ο Ταξινομητής Τροχιάς

Θεωρούμε ένα κινητό χρήστη, ο οποίος διασχίζει μια σειρά κυψελών σε ένα κυψελωτό δίκτυο. Κάθε κυψέλη έχει ένα μοναδικό αναγνωριστικό x_i (αναγνωριστικό κυψέλης, CID). Ένα κυψελωτό δίκτυο αποτελείται από ένα σύνολο $X = \{x_i: i = 1, \dots, |X|\}$ κυψελών (όπου $|X|$ είναι η πληθικότητα του συνόλου X). Υποθέτουμε ότι σε κάθε δεδομένη στιγμή ο χρήστης βρίσκεται εντός μίας κυψέλης $x_i \in X$. Έχουμε διατυπώσει το πρόβλημα εκμάθησης της δημιουργίας ενός ταξινομητή τροχιάς (TC) και ενός προγνώστη θέσης (LP), ο οποίος μετεγκαθίσταται στον TC, ως εξής:

Λαμβάνοντας υπόψη τη συμπεριφορά της κινητικότητας ενός κινητού χρήστη, κατασκευάζουμε έναν TC που ταξινομεί μια μικρής διάρκειας τροχιά ενός χρήστη και έναν LP που προβλέπει την επόμενη κυψέλη στην οποία αυτός ο χρήστης αναμένεται να κινηθεί στην επόμενη χρονοθυρίδα.

Στην προσέγγισή μας, οι TC και LP καλύπτουν έναν μόνο κινητό χρήστη. Κάθε ζευγάρι εκπαίδευσης (X_i, g_i) αποτελείται από μια τροχιά $X_i = [x_{i1}, \dots, x_{im}]^T$ όπου x_{ij} είναι ένα CID και g_i είναι το CID της προβλεπόμενης κυψέλης, το οποίο καλείται η *ετικέτα κλάσης* για το συγκεκριμένο ζευγάρι. Υιοθετούμε τη σημειογραφία $t(x_i)$ για να δηλώσουμε τη στιγμή της εισόδου του χρήστη στην κυψέλη x_i . Για κάθε κυψέλη στην τροχιά X_i , ισχύει αυστηρά ότι $t(x_{i1}) < \dots < t(x_{id})$ και η κυψέλη g_i είναι η προγενέστερη της x_{id} κυψέλης, δηλαδή, $t(x_{id}) < t(g_i)$ - η επόμενη κίνηση του χρήστη. Στη συνέχεια, $T = \{(X_1, g_1), \dots, (X_N, g_N)\}$ είναι το σύνολο μάθησης, που αποτελείται από N ζεύγη μάθησης. Η ταξινόμηση μιας άγνωστης m -διάστασης X τροχιάς είναι η διαδικασία της

ανάθεσης της X σε μια γνωστή κλάσης ετικέτα-κυψέλης g . Έχουμε υιοθετήσει τις μεθόδους ταξινόμησης, k NN (μη-παραμετρική μέθοδος) και δέντρα απόφασης (μη-μετρική μέθοδος) προκειμένου να κατασκευαστεί το TC.

2.2.1. Ο Μη-Παραμετρικός Ταξινομητής Τροχιάς

Ο ταξινομητής k NN [10] δεν απαιτεί καμία δομή δεδομένων που θα προσαρμοστεί πάνω στα δεδομένα μάθησης. Λαμβάνοντας υπόψη μία τροχιά ερώτησης X_0 , βρίσκουμε τις k πιο κοντινές τροχιές $X_{(r)}$, $r = 1, \dots, k$ σε απόσταση από την X_0 . Στη συνέχεια, ο TC ταξινομεί χρησιμοποιώντας ψηφοφορία πλειοψηφίας μεταξύ των k γειτόνων, έτσι ώστε,

$$g_0 = \arg \max_{(r)} P(g_{(r)}|X_0) \quad (2.1)$$

$P(g_{(r)}|X_0)$ είναι η πιθανότητα να ταξινομηθεί η X_0 με την κλάση $g_{(r)}$. Ως εκ τούτου, ο k NN ταξινομεί την X_0 αποδίδοντας στην X_0 την πιο συχνή ετικέτα μεταξύ των k πλησιέστερων τροχιών, δηλαδή, την προβλεπόμενη κυψέλη g_0 . Επιπλέον, μια τροχιά αναλαμβάνει ονομαστικές τιμές (CIDs). Θεωρείστε τον $\gamma(\cdot, \cdot)$ δείκτη, όπως $\gamma_j(x_{ij}, x_{0j}) = 0$ αν τα δύο CIDs x_{ij}, x_{0j} είναι τα ίδια και 1 σε αντίθετη περίπτωση, όπου x_{ij} είναι η j η κυψέλη της i η τροχιάς, δηλαδή,

$$\gamma_j(x_{ij}, x_{0j}) = \begin{cases} 0 & \text{αν } x_{ij} = x_{0j} \\ 1 & \text{αντίθετα} \end{cases}$$

Η απόσταση ανάμεσα σε δύο τροχιές $X_{(i)}$, X_0 είναι τότε

$$\Gamma_i = \|X_{(i)} - X_0\| = \sum_{j=1}^m \gamma_j(x_{(i)j}, x_{0j})$$

Για τον k NN, το σφάλμα των δεδομένων μάθησης είναι μια αύξουσα συνάρτηση του k και είναι πάντα μηδέν για $k = 1$. Στην πραγματικότητα, ο 1-κοντινότερος-γείτονας (1NN) ταξινομητής υιοθετείται τυπικά σε χαμηλής-διάστασης προβλήματα [10], δηλαδή, για χαμηλές τιμές του m . Στην περίπτωσή μας η διάσταση της X_0 είναι $m = 4$ ή $m = 3$, ανάλογα με το χωρικό πλαίσιο αναπαράστασης, όπως περιγράφεται στην Ενότητα 2.4.2. Αν το m είναι μικρό, δηλαδή, 1 ή 2, δεν υπάρχει καμία ανάγκη να πραγματοποιηθεί πρόβλεψη διότι πρόκειται για μία απλή μετάβαση κυψέλης. Ο 1NN χωρίζει το χώρο κίνησης σε κυρτά κελύφη που αποτελούνται από όλες τις τροχιές που βρίσκονται πιο κοντά σε μία X_i από ό, τι σε οποιαδήποτε άλλη τροχιά εκπαίδευσης. Όλες οι τροχιές ενός κυρτού κελύφους έχουν επισημανθεί από το αντίστοιχο g_i . Ως εκ τούτου, ο χώρος κίνησης αποσπάται σαν μία ψηφίδωση Voronoi. Επιπρόσθετα, η προκατάληψη εκτίμησης για τον 1NN είναι χαμηλή εφόσον χρησιμοποιεί μόνο τροχιές εκπαίδευσης πλησιέστερα στην τροχιά ερώτησης.

2.2.2. Ο Μη-Μετρικός Ταξινομητής Τροχιάς

Τα δέντρα απόφασης είναι μία κοινή προσέγγιση για την ταξινόμηση ονομαστικών δεδομένων, όπου δεν μπορεί να καθιερωθεί καμία ομοιότητα και διάταξη. Η διαδικασία της ταξινόμησης γίνεται από πάνω προς τα κάτω, όπου κάθε κόμβος υποδηλώνει μία συγκεκριμένη κυψέλη της τροχιάς. Ένα μονοπάτι από την ρίζα σε ένα φύλλο αντιστοιχεί σε μία συγκεκριμένη τροχιά που οδηγεί σε ένα πιθανό g_0 , το οποίο εμφανίζεται στο δέντρο σαν ένας κόμβος φύλλο. Κάθε εξαγόμενη απόφαση σε έναν κόμβο r καλείται *διαίρεση* και αντιστοιχεί στη διαίρεση ενός υποσυνόλου T_r από το σύνολο εκπαίδευσης T . Μία τέτοια απόφαση επηρεάζεται από συγκεκριμένα μέτρα εντροπίας από τα υποδέντρα του τρέχοντος κόμβου. Ωστόσο, αντί να διαιρείται κάθε κόμβος μόνο σε δύο υποσύνολα του T σε κάθε στάδιο, μπορούν να χρησιμοποιηθούν διαιρέσεις πολλαπλών-διαδρομών σε περισσότερα από δύο υποσύνολα. Το πρόβλημα είναι ότι οι διαιρέσεις πολλαπλών-διαδρομών τεμαχίζουν το σύνολο T γρήγορα, με αποτέλεσμα να μη μένουν επαρκή δεδομένα για το επόμενο κατώτερο επίπεδο [11], δηλαδή, παραλείπεται η χρονική αλληλουχία των κυψελών του προτύπου κίνησης του χρήστη. Ωστόσο, η διαίρεση πολλαπλών-διαδρομών μπορεί να επιτευχθεί από μία σειρά (αναδρομικών) δυαδικών διαιρέσεων, η τελευταία μέθοδος είναι προτιμότερη (όπως θα δούμε παρακάτω). Ως εκ τούτου, ο TC ταξινομεί ένα πρότυπο τροχιάς στον κόμβο r στην κλάση πλειοψηφίας $g_{(r)}$, η οποία είναι,

$$g_{(r)} = \arg \max_g P_{(r)g} \quad (2.2)$$

Η κλάση $g_{(r)}$ μεγιστοποιεί την ποσότητα $P_{(r)g} = N_{(r)}^{-1} \sum_i (g_i = g)$, όπου $N_{(r)}$ είναι ο αριθμός των παρατηρήσεων στο T_r , με $X_i \in T_r$, $(X_i, g_i) \in T$, όπου το I είναι ενότητα αν ισχύει ότι $g_i = g$, αλλιώς το I είναι μηδέν.

Υιοθετούμε τον C4.5 αλγόριθμο [11], ο οποίος είναι ένας δημοφιλής ταξινομητής για δέντρα απόφασης. Στον C4.5 κάθε κόμβος φύλλο αντιστοιχεί σε έναν συνδυαστικό κανόνα σχέσης των αποφάσεων που οδηγούν από την ρίζα στο συγκεκριμένο φύλλο – ο οποίος είναι η χρονική σειρά μιας καταγεγραμμένης τροχιάς. Ο C4.5 ταξινομεί επίσης τροχιές με ελλiptικές τιμές, δηλαδή, μη καταγεγραμμένες κυψέλες και, ακόμα, τροχιές με χωρικά κενά. Εν τέλει, ο υιοθετούμενος C4.5 με δυαδικές διαιρέσεις έχει την ικανότητα να κάνει κλάδεμα / διαγράφοντας επαναλαμβανόμενους προγόνους, δηλαδή, μεταβάσεις μεταξύ των κυψελών, στο σχηματισμένο δέντρο.

2.2.3. Ο Συνδυαστικός Ταξινομητής Τροχιάς

Ένας κοινός τρόπος για να συνδυαστεί ένας μη-μετρικός TC και ένας μη-παραμετρικός TC είναι η προσέγγιση ψηφοφορίας, η οποία είναι ένας γραμμικός συνδυασμός αυτών [12]. Συγκεκριμένα, ας θεωρήσουμε $w_j \in [0,1]$ να είναι το βάρος του j η ταξινομητή $TC(j)$ (οι 1NN και C4.5 στην περίπτωση μας) και d_{ji} να είναι οι ύστερες πιθανότητες κλάσης $d_{ji} = P(g_i | X_0, TC(j))$, $i = 1, \dots, N$. Το υιοθετούμενο μοντέλο ψηφοφορίας υποθέτει ότι οι δύο ταξινομητές είναι ισότιμοι, το οποίο σημαίνει ότι $w_j = .5$. Ως εκ τούτου, η κλάση g_i που μεγιστοποιεί την $P(g_i | X_0)$ επιλέγεται, η οποία είναι,

$$P(g_i|X_0) = \sum_{j=1,2} P(g_i|X_0, TC(j)) \cdot P(TC(j)) \quad (2.3)$$

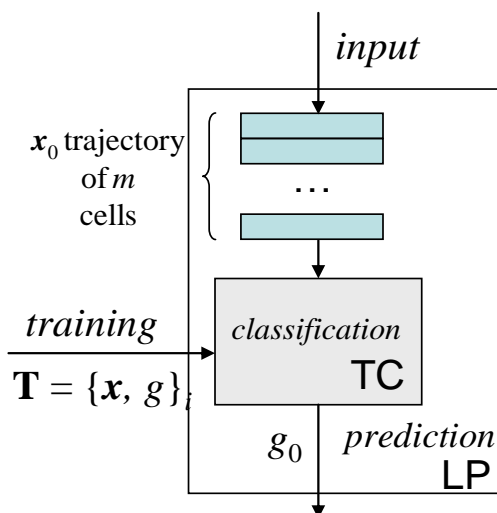
Και $g_0 = \arg \max_{i=1, \dots, N} P(g_i|X_0)$. Στην [13] παρουσιάζουμε τα ευρήματά μας για την επιλογή του κατάλληλου ταξινομητή. Έχουμε πειραματιστεί με τους 1NN, C4.5 και ψηφοφορία (η οποία περιλαμβάνει και τους δύο 1NN και C4.5).

2.3. Πρόβλεψη θέσης

Στην περίπτωση μας το χωρικό πλαίσιο αποτελείται από:

- το τρέχον αναγνωριστικό κυψέλης x_i του κινητού χρήστη,
- την κατεύθυνση της κίνησης του χρήστη, η οποία συνεπάγεται από το ζεύγος (x_i, x_j) με $t(x_j) > t(x_i)$ δηλώνοντας την μετάβαση από την κυψέλη x_i στη γειτονική κυψέλη x_j ,
- το σύνολο των τροχιών του χρήστη, στο οποίο κάθε τροχιά περιλαμβάνει m μεταβάσεις κυψέλης, και
- την περιβάλλουσα κυψελωτή πληροφορία του x_i , δηλαδή, τις γειτονικές κυψέλες του x_i .

Το μοντέλο σχεδιασμού για το LP και το αντίστοιχο TC απεικονίζεται στο Σχήμα 2.1. Συγκεκριμένα, το σύστημα αξιολογεί την βραχυπρόθεσμη κίνηση του χρήστη (m επισκεπτόμενες κυψέλες) και το TC προσπαθεί να ταξινομήσει τη συγκεκριμένη τροχιά σε μία γνωστή συμβολική ετικέτα κλάσης. Ο προγνώστης θέσης, ο οποίος βασίζεται στα αποτελέσματα ταξινόμησης του TC (ο οποίος έχει εκπαιδευτεί με ένα σύνολο \mathbf{T}), προβλέπει τη μέλλουσα συμβολική τοποθεσία, δηλαδή, τη μελλοντική κυψέλη.



Σχήμα 2.1. Μοντέλο σχεδιασμού του TC και του LP.

Προτείνουμε δύο εναλλακτικούς σχεδιασμούς για το LP. Εισάγουμε την έννοια του Μάκρο και Μίκρο LP. Η κύρια διαφορά είναι ο τρόπος που ο αντίστοιχος TC δημιουργεί τη βάση γνώσης. Στον Μάκρο-LP ο TC ταξινομητής διατηρεί γνώση για όλες τις επισκεπτόμενες κυψέλες στο χώρο κίνησης. Στον Μίκρο-LP, κάθε TC (ένας για κάθε

κυψέλη) διατηρεί γνώση μόνο για τις γειτονικές κυψέλες (βλέπε Σχήμα 2.1). Επιπροσθέτως, οι LPs είναι επεκτάσιμοι, το οποίο σημαίνει ότι, αν ένα πρότυπο κίνησης αλλάξει τότε το μόνο που έχουμε να κάνουμε είναι να ενημερώσουμε σε τοπικό επίπεδο τις νεώτερες τροχιές και όχι να ξαναδημιουργήσουμε τον TC του Μάκρο-LP ή όλους τους TCs των αντίστοιχων Μίκρο-LPs.

2.3.1. Ο Μάκρο-Προγνώστης Θέσης

Ο Μάκρο-LP βασίζεται σε έναν TC ο οποίος τροφοδοτείται με τις τροχιές του χρήστη από ολόκληρο το κυψελωτό δίκτυο. Εισάγουμε τον Μάκρο Cell-based LP (MaC) και τον Μάκρο Direction-based LP (MaD). Το μάκρο χαρακτηριστικό των MaC και MaD LPs δηλώνει ότι το θεωρούμενο πλαίσιο χωρικής πληροφορίας αναφέρεται στο σύνολο του χώρου κίνησης του χρήστη, δηλαδή, όλα τα CIDs του δικτύου. Στην περίπτωση του MaC, υπάρχουν $|X|$ διαφορετικές κυψέλες ενώ, στην περίπτωση του MaD, υπάρχουν $|U|$ διαφορετικές κατευθύνσεις (όπως θα δούμε παρακάτω).

Στο MaC, η τροχιά είναι η χρονοσημασμένη σειρά των τελευταίων m κυψελών που έχουν επισκεφθεί από τον χρήστη. Για παράδειγμα, ας θεωρήσουμε ένα δίκτυο από κυψέλες $X = \{x_1, x_2, x_3, x_4, x_5\}$ με $m = 3$. Το ζεύγος εκπαίδευσης $(x, g) = ([x_1, x_2, x_3], x_1)$ στο MaC αποτελείται από την τροχιά $x = [x_1, x_2, x_3]$, η οποία ταξινομείται ως $g = x_1$, το οποίο σημαίνει ότι ο χρήστης μεταγεται από το x_1 στο x_2 , από το x_2 στο x_3 και επιστρέφει στο από το x_1 . Θεωρούμε τώρα την επόμενη κίνηση του χρήστη: το ζεύγος εκπαίδευσης τώρα αναπαριστά τις μεταγωγές από το x_2 στο x_3 , από το x_3 στο x_1 και την επόμενη επισκεπτόμενη κυψέλη, ας πούμε, x_4 . Ως εκ τούτου, το χρονικά μετατοπιζόμενο ζεύγος εκπαίδευσης είναι τώρα το $([x_2, x_3, x_1], x_4)$. Η MaC τροχιά αναπαρίσταται μέσω του διανύσματος

$$x = [x_1, \dots, x_m] \quad (2.4)$$

Όπου x_m είναι η κυψέλη, στην οποία ο χρήστης βρίσκεται επί του παρόντος. Το αντίστοιχο σύνολο εκπαίδευσης είναι

$$T = \{(x_i, g_i), i = 1, \dots, N\}, g_i \in X \quad (2.5)$$

Στο MaD, η τροχιά είναι η χρονοσημασμένη σειρά των $m - 1$ απεικονίσεων κατεύθυνσης (δηλαδή, NW, W) που ο χρήστης έχει ακολουθήσει όταν κινήθηκε διαμέσου των m κυψελών. Στο μοντέλο των εξάγωνων κυψελών, μία κατεύθυνση u από μία κυψέλη x_i σε μία άλλη κυψέλη x_j , παίρνει έξι πιθανές απεικονίσεις κατεύθυνσης από το σύνολο,

$$U = \{E(\text{east}), W(\text{west}), SE(\text{southeast}), SW(\text{southwest}), NE(\text{northeast}), NW(\text{northwest})\}$$

Υποθέτοντας ένα άπειρο, τέλεια τμηματοποιημένο και «προσανατολισμένο» χώρο με εξάγωνες κυψέλες. Αυτό το χωρικό μοντέλο είναι πολύ δημοφιλές στην λογοτεχνία των ασύρματων δικτύων. Σε μία τροχιά x από m κυψέλες, ένας χρήστης έχει πραγματοποιήσει $m - 1$ μεταβάσεις κυψελών. Κατά τη διάρκεια της j th μετάβασης, ο χρήστης κινείται από την κυψέλη x_j σε μία άλλη κυψέλη x_i , με $x_j \neq x_i$. Η συμβολική

κατεύθυνση της μετάβασης (x_j, x_i) δίνεται από το $u_j \in U$. Για παράδειγμα, ας θεωρήσουμε το δίκτυο από κυψέλες $X = \{x_1, x_2, x_3, x_4, x_5\}$ με $m = 3$ από το προηγούμενο παράδειγμα. Ένα πιθανό ζεύγος εκπαίδευσης (\mathbf{u}, h) αποτελείται από ένα $m - 1$ διάστασης διάνυσμα με κατευθύνσεις $\mathbf{u} = [NW, E]$ και $h = SW$, το οποίο σημαίνει ότι ο χρήστης μεταγεται βορειοδυτικά από το x_1 στο x_2 , ανατολικά από το x_2 στο x_3 και επιστρέφει νοτιοδυτικά από το x_3 στο x_1 . Η MaD τροχιά δίνεται από μία σειρά συμβολικών κατευθύνσεων,

$$\mathbf{u} = [u_1, \dots, u_{m-1}] \quad (2.6)$$

Στην οποία η $(m - 1)th$ μετάβαση υποδεικνύει την τελευταία κατεύθυνση του χρήστη και το σύνολο εκπαίδευσης είναι,

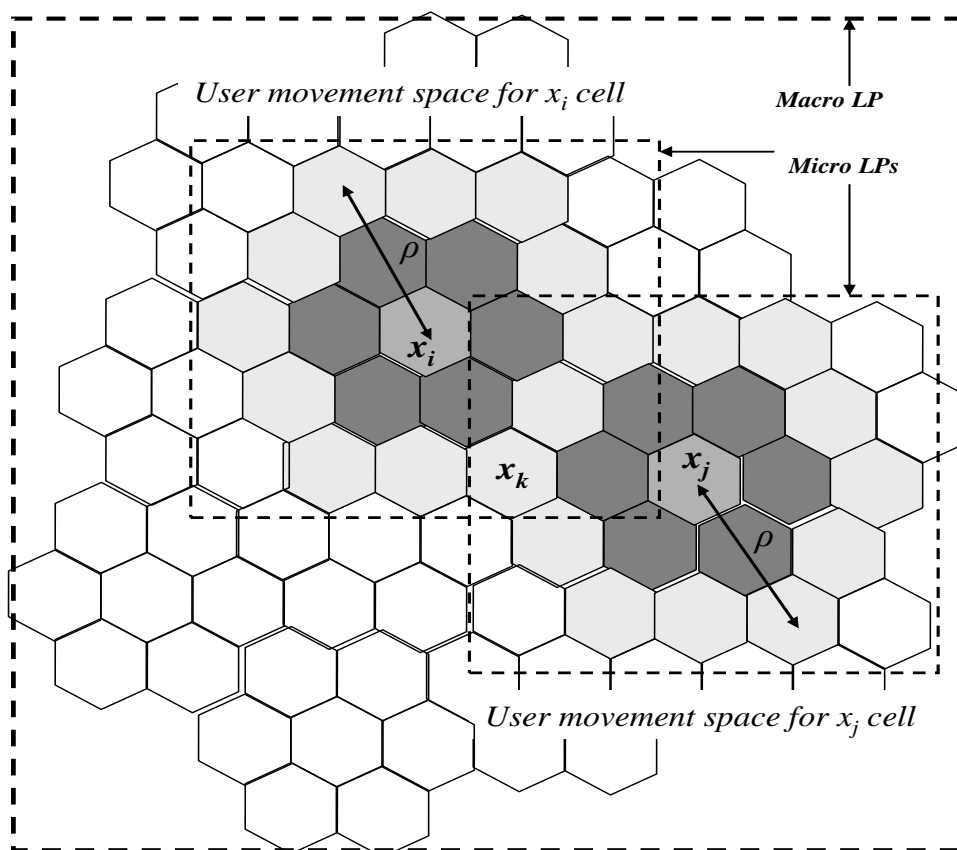
$$\mathbf{T} = \{(\mathbf{u}_i, h_i), i = 1, \dots, N\}, h \in U \quad (2.7)$$

2.3.1.1. Χαρακτηριστικά των MaC και MaD LPs

Κανένα από τα MaC και MaD LPs δεν λαμβάνουν υπόψη τους πόσο χρόνο παρέμεινε ο χρήστης μέσα σε μία συγκεκριμένη κυψέλη, δηλαδή, το χρόνο παραμονής μέσα στην κυψέλη, ή πότε, κατά το διάστημα της ημέρας, πραγματοποιήθηκε μια μετάβαση κυψέλης. Με άλλα λόγια, οι MaC και MaD προγνώστες θέσης βασίζονται μόνο σε χωρικό πλαίσιο (αναγνωριστικό κυψέλης και κατεύθυνση της κίνησης). Στο [14] οι συγγραφείς έχουν αναφερθεί στο θέμα του χρόνου επεκτείνοντας ML αλγόριθμους μέσω χρονοσημάνσεων. Τα σχετικά αποτελέσματα δεν δικαιολογούν το επιπρόσθετο πλήθος της πληροφορίας που χρειάζεται για να ενσωματωθεί η χρονική πληροφορία στην υποκείμενη βάση γνώσης. Το παρατηρούμενο όφελος είναι ασήμαντο. Μία συζήτηση σχετικά με την ένταξη του χρόνου/χρονοσημάνσεων στη διαδικασία πρόγνωσης θέσης παρέχεται στην Ενότητα 2.5.

Όταν υιοθετούμε το MaD, δεν προβλέπουμε την επόμενη κυψέλη $g \in X$ αλλά την μέλλουσα κατεύθυνση $u \in U$, στην οποία ο χρήστης κατευθύνεται. Ως εκ τούτου, προβλέποντας τη u κατεύθυνση και γνωρίζοντας, εκ των προτέρων, την τελευταία x_m κυψέλη μας παρέχεται επαρκής πληροφορία να προσδιορίσουμε την επόμενη κυψέλη, δηλαδή, την κλάση g_0 για μία τροχιά ερώτησης x_0 ή σειρά ερώτησης από κατευθύνσεις \mathbf{u}_0 . Ωστόσο, και για τα δύο MaD και MaC, αρκετά χαρακτηριστικά που σχετίζονται με τους φυσικούς περιορισμούς της κίνησης του χρήστη πρέπει να ληφθούν υπόψη. Συγκεκριμένα, στο MaC, υποτίθεται ότι ο χρήστης κινείται από την κυψέλη x_i σε μια άλλη κυψέλη x_j άσχετα από την φυσική απόσταση μεταξύ x_i και x_j και τα μεταξύ τους εμπόδια. Ωστόσο, στην πραγματικότητα, ο χρήστης κινείται μόνο μεταξύ γειτονικών κυψελών. Επιπρόσθετα, η κίνηση μεταξύ οποιονδήποτε γειτονικών κυψελών μπορεί να μην είναι δυνατή λόγω της ύπαρξης φυσικών εμποδίων, δηλαδή, μπορεί να μην υπάρχει καμία φυσική διαδρομή που να ενώνει δύο γειτονικές κυψέλες λόγω της ύπαρξης ενός λόφου. Στο MaD, υποτίθεται ότι, οι σειρές των κατευθύνσεων ακολουθούν όμοια πρότυπα μέσα στο χώρο των κυψελών. Για παράδειγμα, αν παρατηρήσουμε, σε δύο διαφορετικές κυψελωτές «περιοχές» του δικτύου, τη σειρά των κατευθύνσεων $\mathbf{u} = [NW, NW, NW, NW]$ τότε η προβλεπόμενη κατεύθυνση είναι πάντοτε η ίδια. Ωστόσο, ούτε το MaD δεν λαμβάνει υπόψη του τους διάφορους φυσικούς περιορισμούς στον χώρο κίνησης του χρήστη. Ως εκ τούτου, στο προηγούμενο παράδειγμα αν είμαστε μέσα σε μία κυψέλη και δεν μπορούμε να μετακινηθούμε στην

NW κατεύθυνση στο εγγύς μέλλον τότε πρέπει να ακολουθήσουμε μία διαφορετική κατεύθυνση από την NW. Αυτό κάνει το TC του MaD να ταξινομεί τη σειρά \mathbf{u} δύο φορές κάθε φορά με διαφορετική τιμή κλάσης, γεγονός που οδηγεί σε άστοχη-ταξινόμηση των σειρών δοκιμής. Αυτός είναι και ο λόγος που θεωρούμε τον MaD λιγότερο ακριβή από τον MAC. Από την άλλη πλευρά, αν λάβουμε υπόψη μας την τοπική πληροφορία όπως θα δούμε στην Ενότητα 2.3.2, παρατηρούμε πιο δυνατά μοντέλα όπως ο Μίκρο Direction-based LP (MiD). Στον MiD για το προηγούμενο παράδειγμα η σειρά \mathbf{u} μπορεί να εμφανιστεί σε έναν αριθμό από διαφορετικές μεταβάσεις κυψελών, αλλά η κάθε μία θα διαχειρίζονταν ξεχωριστά πάνω στις περιοχές που καλύπτονται από τα αντίστοιχα Μίκρο TCs.



Σχήμα 2.2. Ο Μάκρο-LP για ένα σύνολο από X κυψέλες και οι Μίκρο-LPs για τις x_i και x_j κυψέλες, η x_k κυψέλη ανήκει στο $V(\rho, x_i)$ και στο $V(\rho, x_j)$.

Ο MaC προγνώστης θέσης είναι ικανός να συλλαμβάνει χωρικούς περιορισμούς δια μέσου της τελευταίας θέσης του χρήστη x_m . Συγκεκριμένα, ο C4.5 TC υιοθετεί, ως το πρώτο χαρακτηριστικό στο δέντρο απόφασης την x_m τιμή της τροχιάς x . Από την άλλη πλευρά, ο k NN TC δεν είναι ικανός να μοντελοποιήσει τους χωρικούς περιορισμούς δεδομένου ότι αποδίδει όμοια σημειολογία σε όλα τα χαρακτηριστικά. Ο MaD προγνώστης θέσης είναι επίσης ανίκανος να χειριστεί περιορισμούς, δεδομένου ότι δεν έχει καθόλου γνώση σχετικά με το που παρατηρείται η συγκεκριμένη σειρά από κατευθύνσεις μέσα στην τοπολογία του δικτύου. Παρ' όλα αυτά, μπορούμε να εισάγουμε τέτοια πληροφορία με το να ενσωματώσουμε το χαρακτηριστικό x_m στην \mathbf{u} , έτσι ώστε,

$$\mathbf{u}' = [u_1, \dots, u_{m-1}, x_m] \quad (2.8)$$

Το \mathbf{u}' αναπαριστά μία τροχιά από $m - 1$ συμβολικές κατευθύνσεις και το αναγνωριστικό κυψέλης x_m . Το πλεονέκτημα αυτού του πλαισίου αναπαράστασης είναι ότι εκμεταλλεύεται την πληροφορία της κατεύθυνσης και όχι την πραγματική τοποθεσία. Για παράδειγμα, για συγκεκριμένες τοποθεσίες, μία σειρά κατεύθυνσης $\mathbf{u} = [\text{NW}, \text{NW}, \text{NW}, \text{NW}]$ θα μπορούσε να σημαίνει ότι η επόμενη κατεύθυνση θα μπορούσε επίσης να είναι NW, δηλαδή, ο χρήστης θα συνέχιζε να κινείται κατά την ίδια κατεύθυνση (κίνηση στην εθνική οδό). Επιπρόσθετα, αυτή η αναπαράσταση επιτρέπει την εξεύρεση ομοιοτήτων μεταξύ προτύπων κίνησης ακόμα και αν αυτά συμβαίνουν σε διαφορετικές περιοχές μέσα στο χώρο κίνησης του χρήστη. Για να μπορέσει ο C4.5 TC να ανακαλύψει τέτοια πρότυπα, υιοθετούμε δυαδικό διαχωρισμό (βλέπε Ενότητα 2.2.2).

2.3.2. Ο Μίκρο-Προγνώστης Θέσης

Ο Μίκρο-LP αναφέρεται ρητά στις χωρικές ιδιαιτερότητες της κυψέλης στην οποία βρίσκεται επί του παρόντος ο χρήστης, δηλαδή, τα φυσικά εμπόδια σε μία συγκεκριμένη περιοχή. Η ιδέα στον μίκρο-προγνώστη θέσης είναι να δημιουργηθεί ένας $\mathbf{T}(x)$ για κάθε κυψέλη $x \in X$ για έναν συγκεκριμένο κινητό χρήστη, όπως απεικονίζεται στο Σχήμα 2.2. Αυτό σημαίνει ότι, αντί να ανακαλύψουμε και να μάθουμε τις χωρικές ιδιαιτερότητες όλου του χώρου κίνησης του χρήστη, (για όλες τις κυψέλες του δικτύου) ένας μίκρο ταξινομητής $\mathbf{T}(x)$ για μία συγκεκριμένη κυψέλη x εκπαιδεύεται ρητά για να ανακαλύψει τις χωρικές ιδιαιτερότητες της γειτονιάς (γειτονικών κυψελών) της x . Ο $\mathbf{T}(x)$ μαθαίνει τοπικά την κινητή συμπεριφορά του χρήστη στην περιοχή της κυψέλης x και, κατά συνέπεια, προβλέπει την επόμενη κυψέλη που θα επισκεφθεί ο χρήστης που επί του παρόντος βρίσκεται στην x .

Υποθέτοντας ότι έχουμε ένα δίκτυο με $|X|$ κυψέλες, έχουμε $|X|$ μίκρο ταξινομητές / προγνώστες θέσεις. Αυτό σημαίνει ότι, υπάρχουν $|X|$ αντίστοιχα σύνολα εκπαίδευσης $\mathbf{T}(x)$ τα οποία απαιτούνται για αυτούς τους προγνώστες. Συγκεκριμένα, το $\mathbf{T}(x)$ περιγράφει τα πρότυπα κίνησης του χρήστη στην περιοχή της κυψέλης x . Βασιζόμενοι σε αυτό το χωρικό πλαίσιο αναπαράστασης έχουμε τον Μίκρο Cell-based LP, $\text{MiC}(x)$ Μίκρο Direction-based LP $\text{MiD}(x)$, $x \in X$ (βλέπε Σχήμα 2.2). Η γειτονιά της κυψέλης $V(\rho, x)$ του $\text{MiC}(x)$, καθορίζεται ως το σύνολο των κυψελών που απέχουν απόσταση ρ κυψελών από την x , $|X| \geq \rho \geq 1$. Ως εκ τούτου, το ζεύγος εκπαίδευσης (x_i, g_i) για τον $\text{MiC}(x)$ είναι το ίδιο με αυτό του MiC με τη μόνη διαφορά ότι αφαιρούμε την κυψέλη x_m δεδομένου ότι αυτή είναι πάντα η ίδια για τον μίκρο ταξινομητή. Αυτό σημαίνει ότι, ένας μίκρο προγνώστης $\text{MiC}(x)$ χρησιμοποιείται για ταξινόμηση και πρόβλεψη δεδομένου ότι η τελευταία θέση της τροχιάς του χρήστη είναι η κυψέλη $x_m = x$. Έτσι η τροχιά για τον $\text{MiC}(x)$ είναι,

$$\mathbf{x} = [x_1, \dots, x_{m-1}] \text{ with } x_i \in V(\rho, x), i = 1, \dots, m - 1 \quad (2.9)$$

Και το σύνολο εκπαίδευσης για τον MiC για μία συγκεκριμένη κυψέλη x είναι,

$$\mathbf{T}(x) = \{(x_i, g_i) : g_i \in V(1, x) \setminus \{x\}, i = 1, \dots, N(x)\} \quad (2.10)$$

$N(x) \leq N$ είναι ο αριθμός των τροχιών που αντιστοιχούν στην κυψέλη x . Κατά συνέπεια, το μήκος της x είναι τώρα $m - 1$ αντί για m . Το σύνολο εκπαίδευσης $T(x)$ για τον $MiD(x)$ ορίζεται όμοια με τα $h_i \in U$.

Η αδυναμία της δημιουργίας τοπικών $TC(x)$ ταξινομητών και προγνωστών για κάθε κυψέλη του δικτύου x είναι ότι δεν είναι δυνατό να εντοπίσουν πρότυπα κίνησης που είναι όμοια μεταξύ διαφορετικών κυψελών. Κατά συνέπεια, η γνώση που απορρέει από τα πρότυπα κίνησης που σχετίζονται με έναν $TC(x_i)$ δεν μπορούν να αξιοποιηθούν σε μία άλλη κυψέλη x_j . Πρέπει να σημειώσουμε ότι, η x_j μπορεί επίσης να είναι μία γειτονική κυψέλη της x_i . Για παράδειγμα, το πρότυπο κίνησης των κατευθύνσεων $\mathbf{u} = [NW, NW, NW, NW]$ μπορεί να εμφανιστεί σε έναν αριθμό από διαφορετικές μεταβάσεις κυψελών, αλλά κάθε μία θα εξεταστεί στις περιοχές που καλύπτονται από τα αντίστοιχα μικρο TCs .

2.4. Εκτίμηση Απόδοσης

Σε αυτήν την ενότητα αξιολογούμε τους τέσσερις προτεινόμενους προγνώστες θέσης, ονομαστικά, τους Μάκρο-LPs, MaC και MaD, και τους Μίκρο-LPs, MiC και MiD. Κάθε TC του κάθε LP συνδυάζει τους βασικούς ταξινομητές 1NN και C4.5 μέσω τις μεθόδου ψηφοφορίας όπως είδαμε στην Ενότητα 2.2.3. Τα αποτελέσματα της πρόβλεψης για τους Μίκρο-LPs είναι η μέση απόδοση κάθε Μίκρο-LP(x) για όλες τις κυψέλες $x \in X$. Στην αρχή ορίζουμε τη συγκεκριμένη παράμετρο *βαθμός της τυχαιότητας της κίνησης* για τα προσομοιωμένα δεδομένα κίνησης και στη συνέχεια εισάγουμε συγκεκριμένες μετρικές για την εκτίμηση της απόδοσης των προτεινόμενων LPs. Συγκεκριμένα, ένας LP εκτιμάται σε σχέση με,

1. την πιθανότητα των σωστών προβλέψεων (ακρίβεια),
2. τη στατιστική σημαντικότητα μεταξύ των προτεινόμενων LPs,
3. τον υπολογιστικό φόρτο που απαιτείται για έναν LP να πετύχει μία πρόβλεψη και
4. την μετρική απόδοσης πρόβλεψης η οποία δεικνύει την ικανότητα ενός LP να προβλέπει τη μελλοντική θέση του χρήστη σε σχέση με μία δοσμένη ακρίβεια πρόβλεψης και συμπεριφοράς κίνησης.

Τελικά, παρέχουμε μία ποιοτική και ποσοτική συγκριτική εκτίμηση των προτεινόμενων LPs με άλλους αλγορίθμους πρόβλεψης και διάφορα μοντέλα που υπάρχουν στην λογοτεχνία.

2.4.1. Παράμετροι και Μετρικές για την Πρόβλεψη Θέσης

2.4.1.1. Βαθμός της Τυχαιότητας της Κίνησης

Εισάγουμε την παράμετρο *βαθμός της τυχαιότητας της κίνησης*, $\delta \in [0,1]$ με σκοπό να εκφράσουμε τον τρόπο που ένας χρήστης μεταγεται μεταξύ κυψελών και αλλάζει κατευθύνσεις. Η υιοθεσία της παραμέτρου δ παρέχει ένα αντικειμενικό κριτήριο για την εκτίμηση των προτεινόμενων LPs δεδομένων ποικίλων επιπέδων αβεβαιότητας και μη-προβλεψιμότητας. Ο βαθμός δ υποδηλώνει τη συμπεριφορά της μετάβασης μεταξύ των κυψελών. Μία τροχιά μπορεί είτε να χαρακτηριστεί σαν μία ντετερμινιστική κίνηση όπως υποδηλώνεται από μία χαμηλή τιμή του δ ή σαν μία τυχαία κίνηση όπως υποδηλώνεται από μία υψηλή τιμή του δ . Η χρήση του δ μπορεί να εγγυηθεί μία ορθή ερμηνεία της εκτίμησης απόδοσης ενός προγνώστη εφόσον μία υψηλή ακρίβεια πρόβλεψης μπορεί όχι απαραίτητα να υποδηλώνει έναν αποδοτικό προγνώστη αν τα πρότυπα ήταν μάλλον ντετερμινιστικά. Οι ντετερμινιστικές τροχιές αναπαριστούν τακτικές κινήσεις, για παράδειγμα, η διαδρομή από το σπίτι στη δουλειά. Αντίθετα, οι τυχαιές τροχιές

αναπαριστούν καθαρά τυχαίες κινήσεις μεταξύ προκαθορισμένων θέσεων (ένα μικρό διάλειμμα για καφέ αφότου έχουμε φύγει από το σπίτι και πριν φθάσουμε στη δουλειά). Ως εκ τούτου, μία τιμή του δ κοντά στη μονάδα δεν σημαίνει μία ρητά μη-ντετερμινιστική συμπεριφορά κίνησης. Αντίθετα, περιορίζεται από εμπόδια που υπάρχουν στον συγκεκριμένο χώρο.

Χρησιμοποιήσαμε συνθετικές τροχιές από τον Realistic Mobility Pattern Generator (RMPG) [64]. Ο RMPG παράγει μεταβάσεις και ίχνη ενός κινητού χρήστη σε ένα κυψελωτό δίκτυο. Οι συνθετικές τροχιές εμπλέκουν δ τιμές στο σύνολο $\{0.0, 0.25, 0.5, 0.75, 1.0\}$. Οι πέντε διακριτές τιμές του δ έχουν μία εμβέλεια από το τακτικό πρότυπο ($\delta = 0.0$ με 500 παραδείγματα τροχιών) στο εντελώς τυχαίο πρότυπο ($\delta = 1.0$ με 1000 παραδείγματα τροχιών). Αξίζει να σημειωθεί ότι η τιμή του δ επηρεάζει επίσης το μέγεθος των τροχιών εκπαίδευσης \mathbf{T} αφού όσο πιο τυχαία είναι μία κίνηση, τόσο περισσότερες μεταβάσεις γενικά απαιτούνται για μία συγκεκριμένη διαδρομή, δηλαδή, μετακίνηση από μία συγκεκριμένη αφετηρία σε έναν συγκεκριμένο προορισμό.

2.4.1.2. Πιθανότητα της Σωστής Πρόβλεψης

Η πιο κοινή στατιστική μετρική για να εκτιμήσουμε την ικανότητα ενός LP να προβλέψει την μελλοντική θέση του χρήστη είναι η πιθανότητα των σωστών προβλέψεων. Ο TC εκπαιδεύεται με ένα σύνολο μάθησης \mathbf{T} . Ο παραγόμενος ταξινομητής $TC(\mathbf{T})$ εφαρμόζεται τότε σε ένα δοκιμαστικό σύνολο \mathbf{Y} , τέτοιο ώστε $\mathbf{Y} \cap \mathbf{T} = \emptyset$, και ο αντίστοιχος προγνώστης LP αξιολογείται. Συγκεκριμένα, η ακρίβεια πρόβλεψης για έναν προγνώστη LP ποσοτικοποιείται μέσω της *πιθανότητας της σωστής πρόβλεψης*, $\alpha(LP) \in [0,1]$. Η τιμή του $\alpha(LP)$ ποσοτικοποιείται μέσω των σωστών προβλέψεων που επιτυγχάνονται από τον LP πάνω στο \mathbf{Y} μετά την εκμάθηση του $TC(\mathbf{T})$. Δηλαδή, ως υποθέσουμε ότι g_P και g_Y είναι το προβλεπόμενο και το πραγματικό αναγνωριστικό κυψέλης ενός κινητού χρήστη στην m th μετάβαση κυψέλης υποθέτοντας τις τροχιές $x_P \in \mathbf{Y}$ και $x_Y \in \mathbf{Y}$ των επισκεπτόμενων $m - 1$ κυψελών, αντίστοιχα. Τότε η $\alpha(LP)$ είναι ο λόγος,

$$\alpha(LP) = \frac{|\mathbf{P}|}{|\mathbf{Y}|} \quad (2.11)$$

Πίνακας 2.1. Παράμετροι για τον Προγνώστη Θέσης.

Παράμετρος	Εμβέλεια	Σημειογραφία
m	> 2	Μήκος τροχιάς
N	$> m$	Αριθμός ζευγών εκπαίδευσης
$ \mathbf{X} $	> 6	Αριθμός κυψελών δικτύου
δ	$[0, 1]$	Βαθμός τυχειότητας κίνησης
n	> 1	10-φορές διασταύρωση επικύρωσης (επαναλήψεις της φάσης εκπαίδευσης και δοκιμής)

των ορθά προβλεπόμενων τροχιών $P = (x_P | g_P = g_Y, x_Y \in \mathbf{Y}$ από το σύνολο \mathbf{Y} . Μία δημοφιλής μέθοδος για να εκτιμηθεί η τιμή της $\alpha(LP)$ είναι η μέθοδος της εκ νέου δειγματοληψίας διασταυρωμένης επικύρωσης. Στην n -φορές διασταυρωμένη επικύρωση, το σύνολο εκπαίδευσης \mathbf{T} χωρίζεται σε n υποσύνολα $\mathbf{T}_i, i = 1, \dots, n$, ίσου μεγέθους. Ο ταξινομητής $TC(\mathbf{T})$ εκπαιδεύεται n φορές, κάθε φορά θέτοντας κατά μέρος ένα $\mathbf{Y}=\mathbf{T}_i$, το οποίο θα χρησιμοποιηθεί ως το σύνολο δοκιμής για τον καθορισμό της

τιμής α . Η αναφερόμενη τιμή είναι η μέση τιμή για τις n επαναλήψεις των φάσεων εκμάθησης και δοκιμής.

Πίνακας 2.2. Μετρικές για την αξιολόγηση του Προγνώστη Θέσης.

Μετρική	Εμβέλεια	Σημειογραφία
$\alpha(LP)$	$[0, 1]$	Πιθανότητα σωστής πρόβλεψης για τον LP
$e(p, q)$	$\{0, 1\}$	Στατιστική σημαντικότητα μεταξύ των p και q LP
$\beta(LP)$	\mathfrak{R}	Αποδοτικότητα πρόβλεψης για τον LP

2.4.1.3. Στατιστική Σημαντικότητα

Ένας LP εξαρτάται σε μεγάλο βαθμό από το σύνολο εκπαίδευσης T του αντίστοιχου TC, έτσι, υπάρχει μία ανάγκη για στατιστικό έλεγχο με σκοπό να (i) εκτιμηθεί η τιμή του αναμενόμενου σφάλματος πρόβλεψης του LP, και (ii) να συγκριθούν οι τιμές των αναμενόμενων σφαλμάτων πρόβλεψης μεταξύ δύο LPs, έτσι ώστε να αναδειχτεί η σχετική τους αποδοτικότητα. Είναι σημαντικό να καθορίσουμε πότε η διαφορά $|\alpha(p) - \alpha(q)|$ για τους p και q LPs είναι *στατιστικά σημαντική*, δηλαδή, αν η διαφορά μεταξύ των τιμών α για τους p και q δεν αποδίδεται σε στατιστικά σφάλματα.

Η στατιστική σημαντικότητα $e(p, q) \in \{0, 1\}$ καθορίζεται μέσω του McNemar ελέγχου. Βασιζόμενοι σε αυτόν τον έλεγχο, ενδιαφερόμαστε είτε για την αναλογία των ορθά προβλεπόμενων τροχιών από τον p και για την αναλογία των εσφαλμένα προβλεπόμενων τροχιών από τον q για το σύνολο εκπαίδευσης Y ή το αντίθετο. Και οι δύο p και q προγνώστες έχουν εκπαιδευτεί και δοκιμαστεί με τα ίδια T και Y σύνολα, αντίστοιχα. Ως εκ τούτου, ο McNemar έλεγχος υποδεικνύει πότε μία σωστή πρόβλεψη από τον p και μία εσφαλμένη πρόβλεψη από τον q είναι περισσότερο ή λιγότερο πιθανή από ότι το αντίστροφο. Στην περίπτωση που οι p και q είναι στατιστικά μη-σημαντικοί, δηλαδή, $e(p, q) = 0$, η διαφορά των πιθανοτήτων σωστής πρόβλεψης τους είναι άνευ σημασίας και, έτσι, και οι δύο προγνώστες μπορεί, με ασφάλεια, να θεωρηθούν ως ισοδύναμοι, αλλιώς $e(p, q) = 1$. Το κατώτατο όριο για την τιμή e έχει τυπικά τεθεί ως η 0.05. Αυτή η τιμή απορρέει από την chi-square κατανομή με έναν βαθμό ελευθερίας.

2.4.1.4. Υπολογιστική Πολυπλοκότητα και Αποδοτικότητα Πρόβλεψης

Η πιθανότητα της σωστής πρόβλεψης και η στατιστική σημαντικότητα δεν είναι οι μόνες μετρικές για την αξιολόγηση της αποδοτικότητας πρόβλεψης ενός LP. Για να φθάσουμε σε ασφαλή συμπεράσματα, έχουμε επίσης να λάβουμε υπόψη μας τον χρόνο που απαιτείται από έναν LP για να πάρει μία απόφαση (πρόβλεψη). Αυτός ο χρόνος είναι μείζονος σημασίας σε συγκεκριμένες κινητές εφαρμογές.

Ας υποθέσουμε N να είναι ο αριθμός των ταξινομημένων τροχιών εκπαίδευσης για τις m κυψέλες. Η πολυπλοκότητα του υιοθετούμενου 1NN είναι $O(m)$ για τον υπολογισμό της απόστασης και $O(mN)$ για την ταξινόμηση. Ωστόσο, μια υλοποίηση Voronoi ψηφίδωσης για τον 1NN καταφέρνει πολυπλοκότητα $O(1)$ σε χρόνο και $O(N)$ σε χώρο με το να υιοθετήσει τους αλγόριθμους μερικής απόστασης, δέντρου αναζήτησης και σύνταξης. Επιπρόσθετα, ο υιοθετούμενος C4.5 (με δυαδικό διαχωρισμό) επιτυγχάνει $O(N) + (N - 1)O(m)$ υπολογισμό χρόνου για να διαχωρίσει ένα εκπαιδευτικό παράδειγμα. Η γενική μέση πολυπλοκότητα του χρόνου για την δημιουργία του δέντρου είναι $O(mN(\log N)^2)$ ενώ ο διαχωρισμός της ρίζας έχει πολυπλοκότητα χρόνου $O(mN \log N)$, ο διαχωρισμός δύο κόμβων στο πρώτο επίπεδο χρειάζεται $O(mN \log(\frac{N}{2}))$

πολυπλοκότητα χρόνου, στο δεύτερο επίπεδο $O(mN \log(\frac{N}{4}))$, και ούτω κάθε εξής. Ο χρόνος ταξινόμησης είναι ανάλογος του βάθους του δέντρου, δηλαδή, $O(\log N)$ και η χωρική πολυπλοκότητα είναι ανάλογη με τον αριθμό των κόμβων, που είναι, $O(N)$. Ως εκ τούτου, η συνολική πολυπλοκότητα της ταξινόμησης για τον Μάκρο-LP κυμαίνεται μεταξύ $O(1) + O(\log N)$ και $O(dN) + O(\log N)$. Για έναν Μίκρο-LP η πολυπλοκότητα της ταξινόμησης και της πρόβλεψης είναι μικρότερη. Αυτό οφείλεται στο γεγονός ότι, ο αριθμός των τροχιών που απαιτούνται για την εκπαίδευση ενός Μίκρο-LP είναι σημαντικά μικρότερος από ότι ο αριθμός όλων των τροχιών στον χώρο κίνησης του χρήστη. Για έναν LP, εισάγουμε την μετρική *αποδοτικότητα πρόβλεψης*, $\beta(LP)$, της ακρίβειας πρόβλεψης $\alpha(LP)$ έναντι του απαιτούμενου χρόνου $\tau(\delta)$ δεδομένης μίας ιστορίας κίνησης T και μίας συμπεριφοράς κίνησης δ , τέτοια ώστε,

$$\beta(LP) = \frac{\alpha(LP)}{\tau(\delta)} \quad (2.12)$$

Για δύο LPs, των οποίων οι TCs έχουν εκπαιδευτεί με το ίδιο T , με όμοια $\tau(\delta)$ ο πιο αποδοτικός προγνώστης είναι αυτός που επιτυγχάνει υψηλότερες τιμές α . Όμοια, για δύο LPs, που έχουν πανομοιότυπες τιμές α , ο πιο αποδοτικός προγνώστης είναι αυτός που επιτυγχάνει τον πιο μικρό χρόνο πρόβλεψης $\tau(\delta)$. Και στις δύο περιπτώσεις, ο αποδοτικότερος LP πρέπει να επιτύχει μία υψηλή τιμή β . Οι πίνακες, Πίνακας 2.1 και Πίνακας 2.2 συνοψίζουν τις παραμέτρους και τις εισαγόμενες μετρικές για την εκτίμηση της απόδοσης των προγνωστών θέσης.

Πίνακας 2.3. Οι πιθανότητες σωστών προβλέψεων $\alpha(LP)$ σε σχέση με το βαθμό της τυχαιότητας κίνησης δ για όλους τους LPs που έχουν εκπαιδευτεί με ένα σύνολο 40-ημερών.

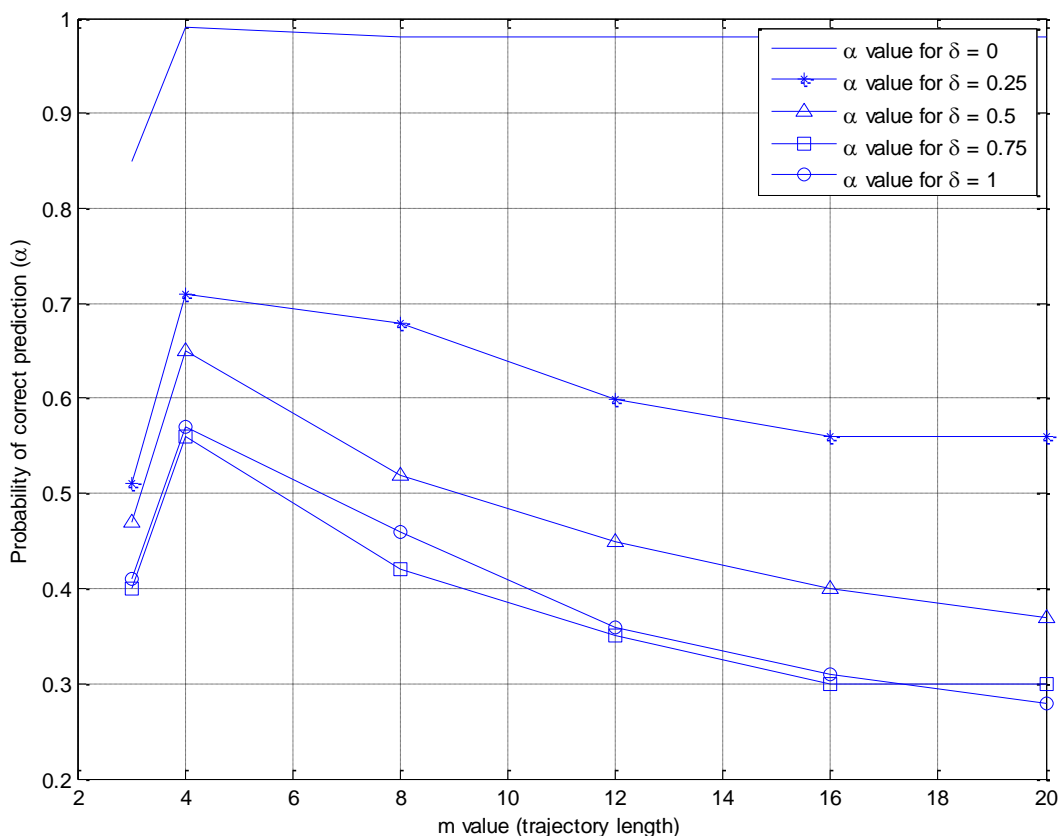
δ	MaC	MaD	MiC	MiD
0.00	0.9558	0.9635	0.9626	0.9584
0.25	0.7452	0.7460	0.7543	0.7551
0.50	0.7156	0.7186	0.7228	0.7144
0.75	0.6648	0.6505	0.6693	0.6790
1.00	0.6690	0.6555	0.6775	0.6783

2.4.2. Εκτίμηση της Πρόβλεψης Θέσης

Οι MaC, MaD, MiC και MiD προγνώστες υλοποιήθηκαν στο Weka ML workbench και τροφοδοτήθηκαν με ίχνη που παράχθηκαν από τον RMPG. Τα RMPG ίχνη δημιουργήθηκαν για έναν απλό κινούμενο χρήστη με κατάλληλα ορισμένο προφίλ κίνησης. Ένα τέτοιο προφίλ ήταν συντονισμένο να είναι συμβατό με τις τιμές του βαθμού δ . Χρησιμοποιήσαμε ένα 40-ημερών σύνολο εκπαίδευσης T , το οποίο αντιστοιχεί σε N ζεύγη εκπαίδευσης. Συγκεκριμένα, εξασφαλίσαμε $N = (2200, 2495, 3173, 3790, 4582)$ για $\delta = (0.0, 0.25, 0.5, 0.75, 1.0)$, αντίστοιχα. Ο χώρος κίνησης του χρήστη ήταν ένα κυψελωτό δίκτυο των $|X| = 100$ κυψέλες.

Η βασική παράμετρος για τα προτεινόμενα LPs είναι το μήκος m (επισκεπτόμενες κυψέλες) τις τροχιάς που απορρέει από την κίνηση του χρήστη. Συγκεκριμένα, αν η συμπεριφορά κίνησης είναι υψηλά τυχαία (δηλαδή, ο χρήστης κινείται μέσα στο κέντρο της πόλης) τότε όσο πιο μακριά είναι η ιστορία που συλλέγουμε για την τροχιά τόσο μεγαλώνει και η πιθανότητα να εισάγουμε θόρυβο μέσα στο μοντέλο παλινδρόμησης.

Αξίζει να σημειώσουμε ότι μία μεγάλη τιμή του m υποδηλώνει ότι ο προγνώστης λαμβάνει υπόψη του ένα εκτεταμένο ίχνος κίνησης το οποίο πιθανά να συνεπάγεται έναν μεγάλο αριθμό από ξαφνικές αλλαγές στην κίνηση. Συγκεκριμένα, πειραματιστήκαμε με το *voting* σχήμα (βλέπε Ενότητα 2.2.3) με σκοπό να εξετάσουμε ποια τιμή του m από το ζεύγος εκπαίδευσης είναι κατάλληλο για να μεγιστοποιήσει την πιθανότητα της σωστής πρόβλεψης (συνδυάζοντας τους ταξινομητές 1NN και C4.5). Επιπρόσθετα, πρέπει να λάβουμε υπόψη μας όχι μόνο την τιμή α αλλά επίσης και την συμπεριφορά κίνησης δ του κινητού χρήστη για να καθορίσουμε την τιμή του m . Μία διαφορετική συμπεριφορά κίνησης οδηγεί σε διαφορετική ακρίβεια πρόβλεψης. Το Σχήμα 2.3 απεικονίζει την τιμή α που παρατηρείται για συγκεκριμένες τιμές του m ($3 \leq m \leq 20$) σε σχέση με το δ . Για κανονικά πρότυπα ($\delta = 0$) όποια τιμή $m \geq 4$ έχει ως αποτέλεσμα όμοια ακρίβεια πρόβλεψης, έτσι ώστε, να μπορούμε να χρησιμοποιήσουμε την ελάχιστη τιμή (δηλαδή, $m = 4$) για να ιχνογραφήσουμε μία ιστορία κίνησης. Για συμπεριφορές κίνησης οι οποίες κυμαίνονται από ελαφρά κανονικές έως πλήρως στοχαστικές μπορούμε να παρατηρήσουμε ότι για $m = 4$ παίρνουμε τις υψηλότερες τιμές α συγκρινόμενες με άλλες τιμές m . Ως εκ τούτου, το μήκος της τροχιάς του κάθε ζεύγους εκπαίδευσης και της τροχιάς ανίχνευσης για την πρόβλεψη είναι $m = 4$, μέσω του οποίου παίρνουμε επίσης έναν πιο αποδοτικό 1NN (βλέπε Ενότητα 2.2.1). Τελικά, εκτελούμε μία 10-φορές διασταύρωση επικύρωσης (δηλαδή, $n = 10$) για να μετρήσουμε την πιθανότητα των σωστών προβλέψεων για κάθε LP.



Σχήμα 2.3. Η πιθανότητα των σωστών προβλέψεων σε σχέση με το m για διάφορες τιμές δ .

Ο Πίνακας 2.3 δείχνει τις πιθανότητες σωστών προβλέψεων για όλους τους προτεινόμενους LPs σε σχέση με τον βαθμό της τυχαιότητας κίνησης. Οι $\alpha(MiC)$ και $\alpha(MiD)$ αντλούν τιμές από τη μέση $\alpha(LP(x))$ τιμή από όλους τους μικρο-προγνώστες για κάθε κυψέλη $x \in X$. Κάποιος θα μπορούσε να παρατηρήσει ότι για μία υψηλή τιμή του δ ($\delta > .75$) όλοι οι προγνώστες επιδεικνύουν μία σχετικά ικανοποιητική ακρίβεια

πρόβλεψης. Επιπρόσθετα, για μία μετρίως στοχαστική συμπεριφορά κίνησης η πιθανότητα της σωστής πρόβλεψης για όλους τους προγνώστες είναι κοντά στο 0.7. Μπορούμε να δούμε ότι και οι δύο μάκρο και μικρο προγνώστες επιδεικνύουν πολύ όμοιες τιμές με βάση τη μετρική α . Αυτό ήταν σχετικά αναμενόμενο δεδομένου ότι υιοθετήσαμε τους ίδιους TCs για όλους τους προτεινόμενους LPs.

Για να επιλέξουμε τον πιο αποδοτικό LP σε σχέση με τη μετρική α , πραγματοποιήσαμε επίσης τον έλεγχο στατιστικής σημαντικότητας (βλέπε Ενότητα 2.4.1.3) για κάθε ζεύγος των προτεινόμενων LPs για διαφορετικό σύνολο εκπαίδευσης T με $\delta = 0.25$ (μία μάλλον κανονική συμπεριφορά κίνησης). Στον Πίνακα 4 παρατηρούμε ότι, για σύνολα εκπαίδευσης που αποτελούνται από y -ημερών ($10 \leq y \leq 40$) κινήσεις χρήστη παίρνουμε ικανοποιητικά αποτελέσματα πρόβλεψης (με συγκλίνουσα ακρίβεια) ακόμα και για περιορισμένα δεδομένα εκπαίδευσης ($y = 10$ ημέρες). Βασιζόμενοι σε αυτό το πείραμα οι στατιστικές συσχετίσεις μεταξύ των LPs αναπαριστώνται από τις τιμές του ελέγχου της στατιστικής σημαντικότητας (βλέπε Ενότητα 2.4.1.3) όπως φαίνονται στον Πίνακα 2.5.

Πίνακας 2.4. Η πιθανότητα των σωστών προβλέψεων $\alpha(LP)$ σε σχέση με το μέγεθος του συνόλου εκπαίδευσης (σε ημέρες).

Ημέρες (y)	MaC	MaD	MiC	MiD
10	0.6785	0.6590	0.7077	0.7012
20	0.7260	0.7092	0.7290	0.7245
30	0.7416	0.7234	0.7516	0.7305
40	0.7452	0.7460	0.7543	0.7551

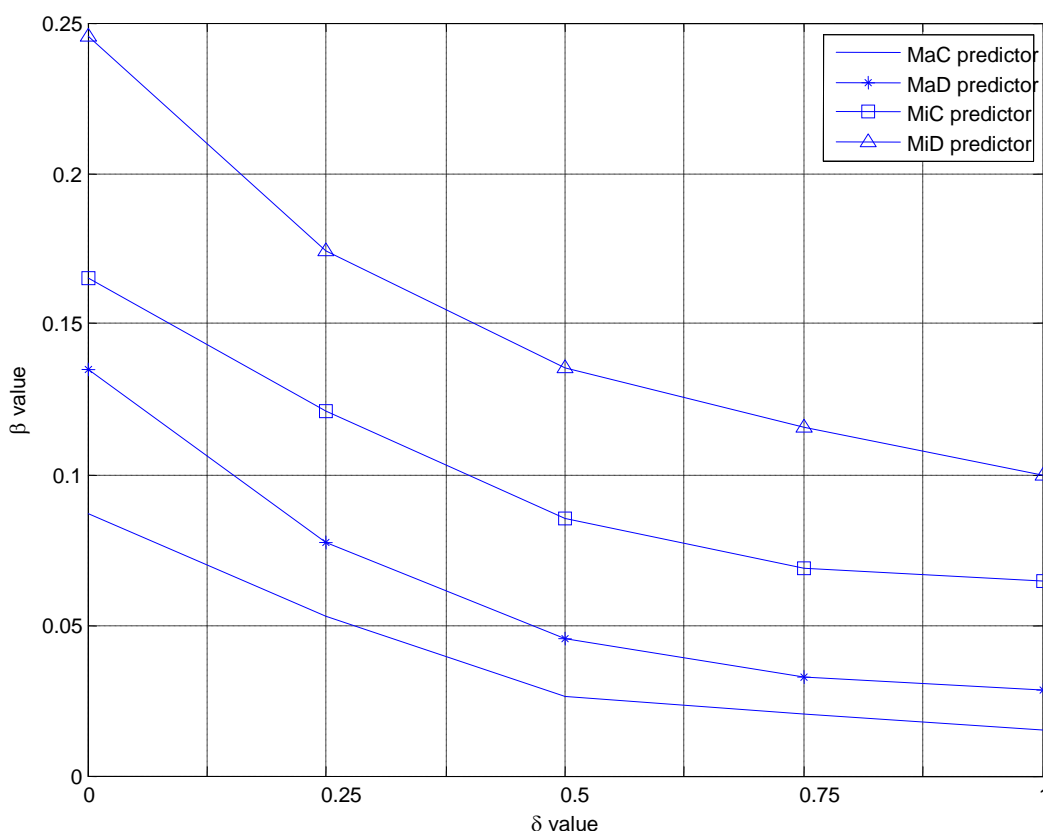
Πίνακας 2.5. Η μετρική της στατιστικής σημαντικότητας για όλους τους προτεινόμενους LPs για την α τιμή.

		$q(LP)$			
$p(LP)$	$e(p, q)$	MaC	MaD	MiC	MiD
$p(LP)$	MaC	-	0	1	1
$p(LP)$	MaD	0	-	1	1
$p(LP)$	MiC	0	0	-	0
$p(LP)$	MiD	0	0	0	-

Υιοθετώντας έναν Μικρο-LP (MiC ή MiD), συλλαμβάνουμε την ειδική γνώση των προτύπων κίνησης που σχετίζονται με μία κυψέλη και τις γειτονικές της κυψέλες. Ως εκ τούτου, είμαστε πιο σίγουροι με έναν Μικρο-LP από ότι με έναν Μάκρο-LP. Στον Μάκρο-LP (MaC ή MaD), αντίθετα, δεν μπορούμε να αποκτήσουμε γνώση που να σχετίζεται με μία συγκεκριμένη περιοχή (δηλαδή, την γειτονιά της κυψέλης) εφόσον ο LP μαθαίνει τις κινήσεις του χρήστη για όλον τον χώρο κίνησης του χρήστη (κυψελωτό δίκτυο). Ως εκ τούτου, βασιζόμενοι μόνο στη στατιστική σημαντικότητα πάνω στη μετρική α στον Πίνακα 2.5 και τις τιμές της μετρικής α από τον Πίνακα 2.4, οι πιο αποδοτικοί προγνώστες είναι οι MiC και MiD. Ωστόσο, έχουμε να προσδιορίσουμε τον πιο αποδοτικό LP βασιζόμενοι όχι μόνο στη μετρική α και στην αντίστοιχη μετρική e αλλά επίσης και στον χρόνο και το χώρο ταξινόμησης και πρόβλεψης που απαιτούνται για να επιτύχουμε μία απόφαση πρόβλεψης δεδομένης μίας τιμής α .

Εξετάζουμε τη συμπεριφορά των τρεχόντων αποδοτικών προγνωστών MiD και MiC σε σχέση με τη μετρική απόδοσης πρόβλεψης β για το ίδιο ακριβώς σύνολο εκπαίδευσης

$T(\delta)$ που χρησιμοποιείται για την παραγωγή της τιμής α . Το Σχήμα 2.4 απεικονίζει την τιμή β έναντι του δείκτη δ της συμπεριφοράς κίνησης. Αξίζει να παρατηρήσουμε ότι το μέγεθος του $T(\delta)$ αυξάνεται γραμμικά σε σχέση με την τιμή του δ . Στο ίδιο σχήμα έχουμε σχεδιάσει την τιμή β για όλους τους προτεινόμενους προγνώστες για λόγους σύγκρισης. Όπως αναφέρθηκε στην Ενότητα 2.4.1.3, για τους δύο Μίκρο-LPs των οποίων οι τιμές α είναι στατιστικά μη σημαντικές, ο πιο αποδοτικός είναι ο προγνώστης με τον μικρότερο χρόνο πρόβλεψης $\tau(\delta)$. Κάποιος θα μπορούσε να παρατηρήσει στο Σχήμα 2.4 ότι ο MiD προγνώστης επιδεικνύει την υψηλότερη β δηλώνοντας ότι επιτυγχάνει την υψηλότερη πιθανότητα σωστής πρόβλεψης με τον μικρότερο χρόνο ταξινόμησης και πρόβλεψης. Ο αντίστοιχος MiC προγνώστης έρχεται δεύτερος κατά σειρά αλλά με σημαντική απόσταση από τον MiD. Αυτό είναι πολύ σημαντικό επειδή, αν και δεν υπάρχει στατιστική σημαντικότητα μεταξύ των MiC και MiD (βλέπε Πίνακα 2.5), ο MiD επιτυγχάνει μία πολύ καλύτερη τιμή β , (παρατηρούμε μία αύξηση της β της τάξης του 58.57% για $\delta = 0.5$). Ο MaD επιτυγχάνει μία χαμηλή τιμή β (70% μείωση της β κατά μέσο όρο για όλες τις δ σε σχέση με τον αντίστοιχο MiD) και επίσης ο MaD αποδίδει καλύτερα από ότι ο MaC (56.66% αύξηση κατά μέσο όρο για όλες τις δ). Συνοψίζοντας, ο πιο αποδοτικός LP από όλους τους προτεινόμενους είναι ο MiD προγνώστης. Στην συγκριτική αξιολόγηση συγκρίνουμε τον MiD προγνώστη με άλλους αλγόριθμους πρόβλεψης και μοντέλα στη λογοτεχνία σε σχέση με τις μετρικές α και β και παρέχουμε την αντίστοιχη ανάλυση υπολογιστικής πολυπλοκότητας.



Σχήμα 2.4. Η β σε σχέση με την δ τιμή για τους προτεινόμενους Προγνώστες Θέσης

Τελικά, αξίζει να σημειωθεί ότι αξιολογήσαμε τους προτεινόμενους LPs προγνώστες όχι μόνο με συνθετικά ίχνη από το RMPG αλλά επίσης και με πραγματικά GSM κυψελωτά δεδομένα από το MIT Media Lab Reality πρόγραμμα. Τα πραγματικά δεδομένα αναφέρονται στις κινήσεις ενός φοιτητή του MIT σε ένα διάστημα 2.5 ετών (από το 2004 έως και τα μέσα του 2006). Συγκεκριμένα ο φοιτητής κινείται από το σπίτι του προς τη δουλειά του καθώς επίσης και από το σπίτι του προς το κέντρο της πόλης.

Επιπρόσθετα, τα πρότυπα κίνησης κυμαίνονται από πολύ απλά (καθημερινό δρομολόγιο, εξορμήσεις σαββατοκύριακων και ταξίδια διακοπών) έως πολύ πολύπλοκα. Έχουμε κάνει χρήση της ανάλυσης φάσματος με σκοπό να εκτιμήσουμε την ισοδυναμία μεταξύ των συνθετικών δεδομένων του RMPG με τα πραγματικά δεδομένα του MIT. Κατεργαστήκαμε 22 σύνολα των 40-ημερών επισκεπτόμενων αναγνωριστικών κυψελών από τον φοιτητή του MIT. Τα μισά από τα ίχνη (έντεκα) αποδείχτηκαν ισοδύναμα με τα ίχνη του RMPG με βαθμό τυχαιότητας $\delta = 0.5$. Για $\delta = 0.5$ η πιθανότητα της σωστής πρόβλεψης του RMPG ίχνους για τον MaC προγνώστη είναι $\alpha(MaC) = 71.56\%$ και η πιθανότητα της σωστής πρόβλεψης για το ίχνος του φοιτητή του MIT είναι $\alpha_{mit}(MaC) = 70.97\%$. Δεδομένου ότι και τα δύο, α και α_{mit} είναι συγκρίσιμα και στατιστικά μη-σημαντικά μεταξύ τους μπορούμε με ασφάλεια να υπεράνουμε ότι $\delta = \delta_{mit}$. Ομοιες παρατηρήσεις έγιναν και για τα MiC, MaD και MiD μοντέλα.

2.4.3. Συγκριτική Αξιολόγηση

Σε αυτήν την ενότητα παρέχουμε μία συγκριτική αξιολόγηση των προτεινόμενων LPs με αλγόριθμους πρόβλεψης θέσης που έχουν εκτενώς αναφερθεί στη σχετική λογοτεχνία δεδομένου ότι τα χαρακτηριστικά τους, η πολυπλοκότητα και οι υποθέσεις τους, τους καταστούν βιώσιμες λύσεις στο πρόβλημα της πρόβλεψης θέσης. Συγκεκριμένα, μας ενδιαφέρει περισσότερο η αποδοτικότητα και η πολυπλοκότητα που έχουν αυτοί οι αλγόριθμοι κατά την πρόβλεψη θέσης.

2.4.3.1. Υπολογιστική Πολυπλοκότητα στην Πρόβλεψη Θέσης

Ο προσεγγιστικός αλγόριθμος αντιστοίχισης προτύπων, Hierarchical Location Prediction (HLP), στο [12] χρησιμοποιεί προ-κατεγραμμένες κανονικές τροχιές για να εκτιμήσει την κατεύθυνση μέσα στην κυψέλη. Αυτός ο αλγόριθμος χρησιμοποιεί ένα επεκτεινόμενο αυτό-μαθησιακό KF για μη ταξινομημένες τυχαίες τροχιές. Ο HLP, που είναι μία διαδικασία semi-Markov, υπολογίζει πιθανότητες μετάβασης k καταστάσεων. Κάθε κατάσταση αναφέρεται σε ένα διακριτό επίπεδο επιτάχυνσης ταχύτητας, έτσι, προσθέτει $O(k^2)$ πολυπλοκότητα χώρου και χρόνου. Ο HLP χρησιμοποιεί την edit απόσταση για να αντιστοιχίσει μία παρατηρούμενη τροχιά μήκους cm (c είναι σταθερά) σε μία προκαθορισμένη τροχιά (μήκους m), έτσι προσθέτει $O(cm^2)$ χρονική και $O(m)$ χωρική πολυπλοκότητα. Στους δικούς μας προτεινόμενους LPs, ο υπολογισμός της απόστασης μεταξύ των παρατηρούμενων και προκαθορισμένων τροχιών είναι $O(m)$ χρονική και $O(1)$ χωρική πολυπλοκότητα. Επιπρόσθετα, ο HLP εξαρτάται σε μεγάλο βαθμό από σε πολύ συγκεκριμένη χωρική πληροφορία, η οποία είναι, ακριβής θέση του χρήστη, ταχύτητα, κατεύθυνση, γεωμετρία των κυψελών, υπολειπόμενος χρόνος μέσα στην κυψέλη και επικαλυπτόμενες περιοχές μεταξύ των κελιών. Από την άλλη πλευρά, οι δικοί μας LPs απαιτούν μόνο τα αναγνωριστικά των παρόντων και των παρελθόντων κυψελών που έχουν επισκεφθεί από τον χρήστη. Ο HLP πρέπει να εντοπίζει συνεχώς τη θέση του χρήστη, έτσι, οδηγείται σε ένα μεγάλο ποσό ιστορικών δεδομένων. Επιπρόσθετα, λόγω αυτής της λεπτομερούς πληροφορίας, ο HLP κάνει υποθέσεις που δεν ισχύουν κατά κανόνα στα κινητά περιβάλλοντα. Ο HLP εξαρτάται απόλυτα από τα υποκείμενα δεδομένα δεδομένου ότι υποθέτει ομαλές κινήσεις. Στα μοντέλα μας, οι υιοθετούμενοι TCs εκπαιδεύονται για κάθε κατανομή της υποκείμενης πληροφορίας κίνησης. Τελικά, ο HLP παρουσιάζει μία ικανοποιητική πιθανότητα σωστής πρόβλεψης όταν το KF σταθεροποιείται, το οποίο σημαίνει ότι το $\alpha(KF)$ φθάνει 0.75 για κάθε $\delta > 0.3$. Αυτό μπορεί να επιτευχθεί από τον MiD προγνώστη χωρίς την χρήση των επιπρόσθετων υποθέσεων κατανομής των ιστορικών πληροφοριών και των δεδομένων.

Συγκρίνουμε επίσης την αποδοτικότητα των δικών μας LPs με μία HMM προσέγγιση για πρόβλεψη θέσης. Η υιοθέτηση ενός HMM ως έναν πιθανό TC για τα δικά μας LPs, δηλαδή αντί για 1NN και C4.5 TC, παρουσιάζει τα ακόλουθα μειονεκτήματα: (1) την χρονική και χωρική πολυπλοκότητα [20], (2) την ανικανότητα του HMM να ταξινομήσει εντελώς άγνωστες τροχιές (το ίδιο πρόβλημα όπως στο [15]) και, (3) την περιορισμένη επεκτασιμότητα του HMM όταν προστίθενται νέες, μη επισκεπτόμενες κυψέλες (όλες οι πιθανότητες εκπομπών και οι πιθανότητες των κρυμμένων καταστάσεων πρέπει να επαναυπολογιστούν). Συγκεκριμένα, θεωρώντας την πρώτη, u_1 , και την τελευταία, u_m , κατεύθυνση ενός διανύσματος \mathbf{u} τότε ο HMM υπολογίζει όλες τις πιθανές κρυμμένες μεταβάσεις κυψελών μεταξύ των u_1 , και u_m για τις αντίστοιχες κυψέλες x_1 , και x_m . Αυτό εννοεί ότι, ο HMM υπολογίζει την πιθανότητα $P(x)$ που παράγει ο χρήστης όταν κάνει μία τροχιά x με m κρυμμένες μεταβάσεις κυψελών. Στους MaC (MaD) με $|X|$ κρυμμένες καταστάσεις και μία σειρά από m παρατηρήσεις παίρνουμε $|X^m|$ πιθανούς όρους για τον υπολογισμό της $P(x)$, έτσι, οδηγούμαστε σε $O(m|X|^m)$ πολυπλοκότητα, που είναι απαγορευτική. Ωστόσο, για τους MiD(x) (MiC(x)) μπορεί να επιτευχθεί μία βελτίωση του υπολογισμού της $P(x)$ της τάξης του $O(m|V(\rho, x)|^m)$. Η ποσότητα $|V(\rho, x)|$ αυξάνει αναλογικά με το τετράγωνο του ρ , το οποίο είναι, $|V(\rho, x)| = c\rho^2$ σε ένα εξαγωνικό κυψελωτό δίκτυο, στο οποίο $c = 6$. Σημειώνουμε επίσης ότι ο HMM forward αλγόριθμος μειώνει την υπολογιστική πολυπλοκότητα σε $O(m|X|^2)$ - πολύ πιο αποδοτική από την χρονική πολυπλοκότητα που σχετίζεται με την εξαντλητική απαρίθμηση των διαδρομών – αλλά ακόμα αφήνει τη χωρική πολυπλοκότητα αδικαιολόγητα υψηλή. Ο HMM που υιοθετήθηκε στο [20] επιτυγχάνει απόδοση συγκρινόμενη με τους δικούς μας LPs δεδομένης της αυξανόμενης χρονικής πολυπλοκότητας (πρόβλεψης θέσης).

2.4.3.2. Αποδοτικότητα στην Πρόβλεψη Θέσης

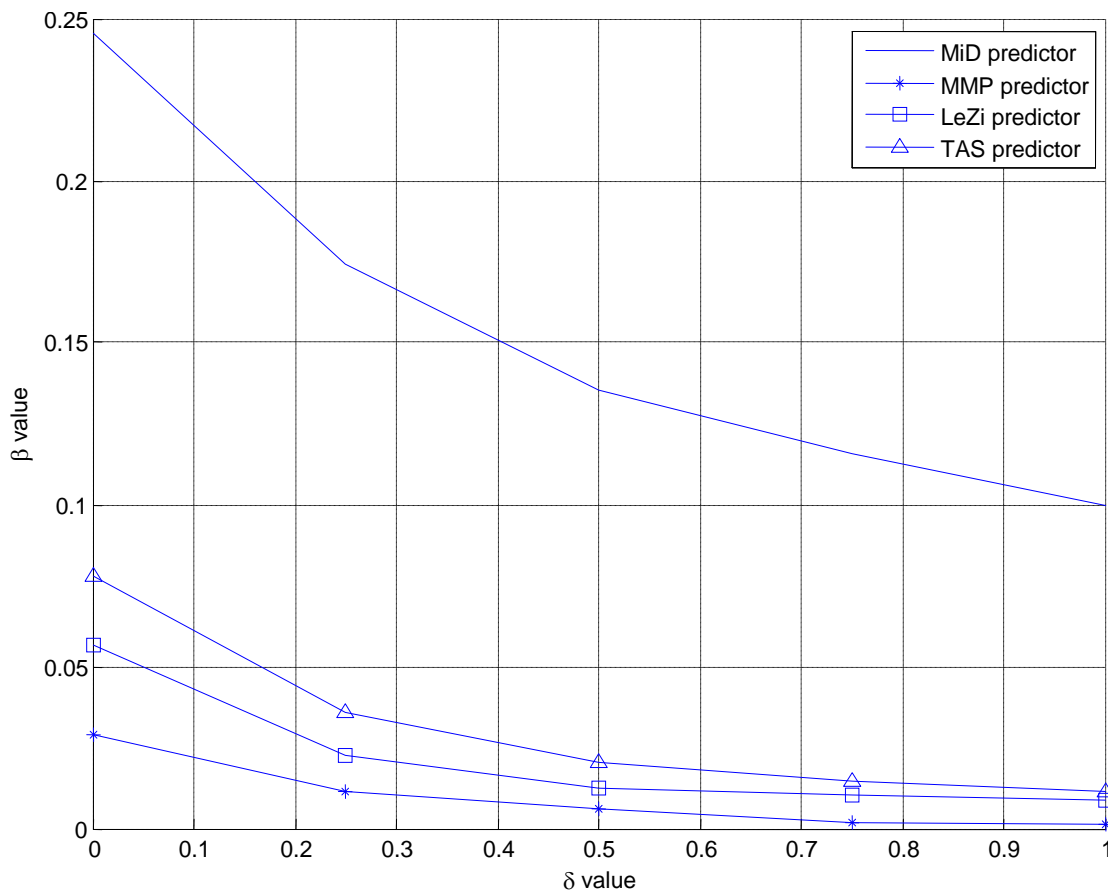
Στο [17], οι συγγραφείς εισάγουν τον Temporally Annotated Sequence (TAS) αλγόριθμο μηχανικής μάθησης. Εμείς υλοποιήσαμε τον TAS με σκοπό να συγκρίνουμε την απόδοση πρόβλεψης του με αυτήν των δικών μας LPs. Συγκεκριμένα, βασιζόμενοι στο μοντέλο του [29], στην περίπτωση μας μία temporally annotated sequence καθορίζεται ως το ζεύγος (x, A) , όπου η τροχιά $x = [x_1, \dots, x_m]$ είναι μεγέθους m και $A = [(t(x_2) - t(x_1)), \dots, (t(x_m) - t(x_{m-1}))]$ είναι ο χρόνος μετάβασης μεταξύ των κυψελών (το A αναφέρεται ως το temporal annotation του TAS στο [17]). Η απόδοση πρόβλεψης του μοντέλου TAS είναι της τάξης του $\alpha(TAS) = 0.69$ με $\delta = 0.25$, ενώ αντίθετα τα δικά μας LPs επιτυγχάνουν μία απόδοση πρόβλεψης της τάξης του $\alpha(LP) = 0.75$ με αντίστοιχο δ .

Πίνακας 2.6. Οι πιθανότητες των σωστών προβλέψεων σε σχέση με τον βαθμό της τυχαιότητας κίνησης για τους MiD, MMP, LeZi και TAS προγνώστες.

$\alpha(LP)$				
MiD	MMP	LeZi	TAS	δ
0.9584	0.5785	0.8332	0.9525	0.00
0.7551	0.3543	0.4567	0.6908	0.25
0.7144	0.2822	0.3841	0.6444	0.50
0.6790	0.1513	0.3123	0.5908	0.75
0.6783	0.1118	0.2719	0.5642	1.00

Επιπρόσθετα, ασχολούμαστε με τα χαρακτηριστικά δύο γνωστών αλγορίθμων [15], [16]. Αυτοί οι αλγόριθμοι προσομοιώθηκαν εξίσου και συγκρίθηκαν με τα δικά μας LPs.

Ο προγνώστης στο [10] μοντελοποιεί τα πρότυπα κίνησης του χρήστη ως μία ακολουθία από κάποια στοιχειώδη πρότυπα κίνησης. Ένας αλγόριθμος αντιστοιχίας προτύπων, Mobile Motion Prediction (MMP), προτείνεται με σκοπό να εκτιμηθεί η μέλλουσα θέση του χρήστη. Το κυριότερο μειονέκτημα αυτού του αλγόριθμου είναι η υψηλή ευαισθησία του στην τυχαία κίνηση. Κάθε τροχιά που δεν μπορεί να ταξινομηθεί μέσω κάποιου προκαθορισμένου στοιχειώδους προτύπου κίνησης αντιμετωπίζεται ως τυχαία. Όπως αναφέρεται στο [15], η ακρίβεια πρόβλεψης του MMP μειώνεται γραμμικά με την αύξηση του παράγοντα δ .



Σχήμα 2.5. Η β σε σχέση με την δ τιμή για τον MiD και τους συγκρινόμενους προγνώστες θέσης.

Ο προγνώστης στο [16] υιοθετεί έννοιες από την lossless compression (δηλαδή, τον αλγόριθμο Lempel-Ziv) για την πρόβλεψη θέσης. Αυτός ο προγνώστης (LeZi-Update) δημιουργεί δυναμικά ένα δέντρο αναζήτησης (trie), το οποίο είναι ένα λεξικό από ενημερώσεις διαδρομών. Ένα τέτοιο λεξικό υποστηρίζει μία προσαρμοστική on-line μέθοδο η οποία μαθαίνει την συμπεριφορά της κίνησης του χρήστη. Αφότου επεξεργαστεί την τροχιά εισόδου, χρησιμοποιεί μία στρατηγική ανάμειξης με σκοπό να προβλέψει την επόμενη θέση του χρήστη.

Στον Πίνακα 6 και το Σχήμα 2.5, παρέχουμε μία ποσοτική σύγκριση από τους αλγορίθμους που συζητήθηκαν στο [15] (MMP), [16] (LeZi-update) και [17] (TAS) σε σχέση με τις μετρικές α και β , αντίστοιχα. Αυτοί οι αλγόριθμοι υλοποιήθηκαν από τους συγγραφείς σαν αυτόνομες εφαρμογές σε Java έτσι ώστε να επιτευχθεί η διαδικασία της συγκριτικής αξιολόγησης. Ο προτεινόμενος MiD και οι MMP, LeZi και TAS προγνώστες τροφοδοτήθηκαν με το ίδιο σύνολο εκπαίδευσης (ένα σύνολο T 40-ημερών). Στον Πίνακα 6, μπορούμε να παρατηρήσουμε ότι οι TAS, LeZi-Update και MMP προγνώστες επιτυγχάνουν χαμηλότερες τιμές $\alpha(LP)$ σε σχέση με τον MiD προγνώστη όσο η τιμή του

δ αυξάνει. Επιπρόσθετα, οι TAS και LeZi-Update είναι πιο σταθεροί για μεγάλες τιμές του δ από ότι ο MMP. Εκτός αυτού, έχουμε να εξετάσουμε την αποδοτικότητα όλων των προγνωστών. Το Σχήμα 2.5 αναπαριστά την τιμή β για όλους τους θεωρούμενους προγνώστες. Μπορούμε να παρατηρήσουμε την αποδοτικότητα του MiD προγνώστη, που είναι 522% υψηλότερη από ότι αυτή του TAS προγνώστη κατά μέσο όρο για όλες τις τιμές του δ . Αυτό είναι πολύ ενδιαφέρον δεδομένου ότι οι προγνώστες MiD και TAS έχουν πολύ όμοιες τιμές α , επίσης παρατηρούμε μία 11% διαφορά στην τιμή του α μεταξύ των MiD και TAS, κατά μέσο όρο, για κάθε δ . Από την άλλη πλευρά, οι MMP και LeZi προγνώστες έχουν παρόμοιες και χαμηλές τιμές β . Σημειώστε ότι ο LeZi προγνώστης έχει 59.75% καλύτερη ακρίβεια πρόβλεψης από τον MMP (μέση τιμή του α για όλες τις τιμές δ). Βασιζόμενοι σε αυτά τα αποτελέσματα σύγκρισης μπορούμε να συμπεράνουμε στο γεγονός ότι η μετρική β είναι πολύ πιο σημαντική από ότι η μετρική α για τα μοντέλα των LPs που εξετάστηκαν από συγκεκριμένες εφαρμογές κίνησης.

2.5. Συζήτηση σχετικά με την Οδηγούμενη από Χρόνο Πρόβλεψη Θέσης

Σύμφωνα με την οπτική γωνία των συγγραφέων, πολλές διαφορετικές προσεγγίσεις για την ένταξη του χρόνου στην πρόβλεψη θέσης μπορούν να υποτεθούν. Θεωρούμε δύο περιπτώσεις όπως και θα συζητήσουμε παρακάτω.

Περίπτωση 1: Ο «απλούστερος» προγνώστης είναι εντελώς ανεξάρτητος από τον χρόνο, δηλαδή το χωρικό πλαίσιο είναι χρόνο-αγνωστικιστικό. Αυτός ο προγνώστης θεωρεί ότι πολύ όμοια πρότυπα κίνησης παρατηρούνται σε διαφορετικές χρονικές στιγμές (δηλαδή, διαφορετικές περιόδους κατά τη διάρκεια της ημέρας), ωστόσο, είναι άνευ σημασίας να εντάξουμε χρονική πληροφορία στον προγνώστη. Η ένταξη της χρονικής πληροφορίας συνεπάγεται την εισαγωγή επιπλέον πληροφορίας στην υποκείμενη βάση γνώσης και πιο σύνθετα σχήματα απόφασης. Τέτοιο φορτίο πληροφορίας θα έπρεπε ωστόσο, να ληφθεί υπόψη μόνο αν οδηγήσει σε σημαντική αύξηση της απόδοσης. Η απουσία της χρονικής πληροφορίας στο υποκείμενο ML μοντέλο υιοθετείται σε αυτήν την εργασία και στο [13].

Περίπτωση 2: Το υποκείμενο μοντέλο χωρικού πλαισίου έχει εμπλουτιστεί με χρονική πληροφορία. Η χρονική πληροφορία μπορεί να κβαντιστεί (δηλαδή, να υιοθετηθεί ο βαθμός ανάλυσης του τετάρτου της ώρας ή της μισής ώρας) για να αποφευχθεί η κατάσταση έκρηξης του χώρου στην υποκείμενη βάση γνώσης. Εκτός από αυτό, είναι λογικό να θεωρήσουμε μικρές αποκλίσεις στην συμπεριφορά κίνησης που έχουν οι νομαδικοί χρήστες. Πρότυπα με όμοια χωρικά χαρακτηριστικά πρέπει να αντιμετωπίζονται ομοίως ανεξάρτητα από μικρές χρονικές μετατοπίσεις. Η χρονική πληροφορία μπορεί να εισαχθεί στην βάση γνώσης με δύο διαφορετικούς τρόπους, εξαρτώμενα με το ιστορικό παράθυρο που το σχήμα πρόβλεψης υιοθετεί. Συγκεκριμένα,

- Αν το ιστορικό παράθυρο, περιλαμβάνει περισσότερες από μία μεταβάσεις, η χρονική πληροφορία μπορεί να επισυναφθεί σε κάθε μία από αυτές τις μεταβάσεις (αυτή είναι η περίπτωση που μελετήθηκε στο [17])
- Μία εναλλακτική προσέγγιση περιλαμβάνει την επισύναψη της χρονικής πληροφορίας μόνο στην τελευταία μετάβαση (αυτή είναι η περίπτωση που θεωρήθηκε στο [14]). Προφανώς, αυτή η επιλογή είναι το μόνο εφικτό σενάριο όταν το ιστορικό παράθυρο εκτείνεται σε μία απλή μετάβαση κυψέλης.

Στην Περίπτωση 2, η χρονική πληροφορία χρησιμοποιείται για να διαχωρίσουμε μεταξύ διαφορετικών χωροχρονικών προτύπων και, επιπρόσθετα, να εξειδικεύσουμε τις αποφάσεις που λαμβάνονται από το ML σχήμα. Ένας LP θα μπορούσε να ενισχυθεί με ένα σχήμα υπολειμματικής εκτίμησης του χρόνου. Αυτό το σχήμα, υπολογίζει, βασιζόμενο στην πληροφορία που συλλέγεται από διαφορετικές πηγές (δηλαδή, το δίκτυο, ιστορικά δεδομένα), τον προσεγγιστικό χρόνο μετάβασης. Μία τέτοια εκτίμηση μπορεί να αξιοποιηθεί από ευφυή σχήματα διαχείρισης πόρων και προηγμένες υπηρεσίες [18], [19]. Η εκτίμηση του χρόνου μετάβασης συμπληρώνει τις αποφάσεις που έχουν ληφθεί από τους προηγούμενους αναφερθέντες αλγόριθμους (δηλαδή, πρόβλεψη θέσης) με σκοπό να παρέχει ένα ολοκληρωμένο μοντέλο κινητής πρόβλεψης.

3. ΕΝΑΣ ΠΡΟΣΑΡΜΟΣΤΙΚΟΣ ΑΛΓΟΡΙΘΜΟΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ ΓΙΑ ΤΗΝ ΠΡΟΒΛΕΨΗ ΘΕΣΗΣ

3.1. Μοντέλα Μηχανικής Μάθησης

Θα συζητήσουμε εκτενώς τους αλγόριθμους ομαδοποίησης που υιοθετήθηκαν για τον δικό μας αλγόριθμο. Θα επισημάνουμε τη διαφορά μεταξύ offline και online ομαδοποίησης και θα εξηγήσουμε τους αλγόριθμους Offline k Means, Online k Means και ART.

3.1.1 Offline k Means

Στον Offline k Means [8] υποθέτουμε ότι υπάρχουν $k > 1$ αρχικές ομάδες στα δεδομένα. Ο αντικειμενικός σκοπός αυτού του αλγόριθμου είναι να ελαχιστοποιήσει το σφάλμα ανοικοδόμησης, το οποίο είναι η συνολική Ευκλείδεια απόσταση μεταξύ των προτύπων \mathbf{u}_i και της αναπαράστασής τους, δηλαδή, τα κέντρα των ομάδων (ομάδες), \mathbf{c}_j . Το σφάλμα ανοικοδόμησης ορίζεται ως εξής:

$$E(\{\mathbf{c}_i\}_{i=1}^k | U) = \frac{1}{2} \sum_t \sum_i b_{i,t} \|\mathbf{u}_t - \mathbf{c}_i\|^2 \quad (3.1)$$

Όπου

$$b_{i,t} = \begin{cases} 1, & \text{if } \|\mathbf{u}_t - \mathbf{c}_i\| = \min_l \|\mathbf{u}_t - \mathbf{c}_l\| \\ 0, & \text{αντίθετα} \end{cases}$$

$U = \{\mathbf{u}_t\}$ είναι το καθολικό σύνολο των προτύπων και $C = \{\mathbf{c}_i\}, i = 1, \dots, k$ είναι το σύνολο των ομάδων. $b_{i,t}$ είναι 1 αν \mathbf{c}_i είναι το πλησιέστερο κέντρο στο \mathbf{u}_t με βάση την Ευκλείδεια απόσταση. Για κάθε εισερχόμενο \mathbf{u}_t κάθε \mathbf{c}_i ενημερώνεται ως ακολούθως:

$$\mathbf{c}_i = \frac{\sum_t b_{i,t} \mathbf{u}_t}{\sum_t b_{i,t}} \quad (3.2)$$

Δεδομένου ότι ο αλγόριθμος λειτουργεί σε offline τρόπο, οι αρχικές ομάδες μπορούν να τεθούν κατά τη διάρκεια της φάσης εκπαίδευσης και δεν μπορούν να αλλάξουν, για παράδειγμα να αυξηθούν ή να επανατοποθετηθούν, κατά τη φάση δοκιμής.

3.1.2. Online k Means

Στον Online k Means [8] θεωρούμε ότι υπάρχουν $k > 1$ αρχικές ομάδες που διαχωρίζουν τα δεδομένα. Αυτός ο αλγόριθμος επεξεργάζεται άγνωστα πρότυπα ένα προς ένα και κάνει *μικρές* ενημερώσεις στην θέση της κατάλληλης ομάδας (\mathbf{c}_i) σε κάθε βήμα. Ο αλγόριθμος δεν απαιτεί μία φάση εκπαίδευσης. Η ενημέρωση για κάθε καινούργιο (άγνωστο) πρότυπο \mathbf{u}_t είναι η επόμενη:

$$\mathbf{c}_i = \mathbf{c}_i + \eta \cdot b_{i,t} \cdot (\mathbf{u}_t - \mathbf{c}_i)$$

Αυτή η ενημέρωση μετακινεί την πλησιέστερη ομάδα (για την οποία $b_{i,t} = 1$) προς την κατεύθυνση του εισερχόμενου προτύπου \mathbf{u}_t κατά έναν παράγοντα η . Οι ομάδες που βρίσκονται σε μεγαλύτερες αποστάσεις από το θεωρούμενο πρότυπο δεν ενημερώνονται. Η σημειολογία των $b_{i,t}$, η και $\mathbf{u}_t - \mathbf{c}_i$ είναι:

- $b_{i,t} \in \{0, 1\}$ υποδηλώνει ποια ομάδα τροποποιείται,
- $\eta \in [0, 1]$ υποδηλώνει πόσο πολύ μετακινείται η ομάδα προς το νέο πρότυπο, και
- $(\mathbf{u}_t - \mathbf{c}_i)$ υποδηλώνει την απόσταση που πρέπει να μαθευτεί

Δεδομένου ότι ο αλγόριθμος είναι online, οι αρχικές ομάδες πρέπει να είναι γνωστές εκ των προτέρων και μπορούν μόνο να επανατοποθετηθούν κατά την φάση της δοκιμής. Ο αριθμός των ομάδων παραμένει σταθερός. Ως εκ τούτου, ο αλγόριθμος έχει μειωμένη ευκαμψία.

3.1.3. Θεωρία Προσαρμοστικού Συντονισμού

Ο αλγόριθμος ART (Adaptive Resonance Theory – Θεωρία Προσαρμοστικού Συντονισμού) [9] είναι ένα online μαθησιακό σχήμα στο οποίο το σύνολο των προτύπων U λαμβάνονται ένα προς ένα και το μοντέλο ενημερώνεται προοδευτικά. Ο όρος *ανταγωνιστική μάθηση* χρησιμοποιείται στον ART υποδηλώνοντας ότι οι τοπικές ομάδες *ανταγωνίζονται* ανάμεσα τους έτσι ώστε να θεωρηθούν «υπεύθυνες» για να αναπαραστήσουν ένα άγνωστο πρότυπο. Το μοντέλο ονομάζεται επίσης και *ο-νικητής-τα-παίρνει-όλα* επειδή μία ομάδα «κερδίζει στο ανταγωνισμό» και ενημερώνεται, και οι άλλες δεν ενημερώνονται καθόλου.

Η προσέγγιση του ART είναι *αυξητική*. Αυτό εννοεί ότι ένας ξεκινά με μία ομάδα και προσθέτει μία καινούργια, αν χρειαστεί. Δοσμένης μίας εισόδου \mathbf{u}_t , η απόσταση b_t υπολογίζεται για όλες τις ομάδες $\mathbf{c}_i, i = 1, \dots, k$ και η πλησιέστερη (δηλαδή, με την ελάχιστη Ευκλείδεια απόσταση) στην \mathbf{u}_t ενημερώνεται. Συγκεκριμένα, αν η ελάχιστη απόσταση b_t είναι μικρότερη από μία συγκεκριμένη τιμή, την ονομαζόμενη *vigilance*, ρ , η ενημέρωση γίνεται όπως και στον Online k Means (βλέπε Εξίσωση (3)). Αλλιώς ένα καινούργιο κέντρο \mathbf{c}_{k+1} που αναπαριστάνει την αντίστοιχη είσοδο \mathbf{u}_t προστίθεται στο μοντέλο (βλέπε Εξίσωση (3)). Αξίζει να σημειωθεί ότι το όριο της ευαισθησίας (*vigilance*) αναφέρεται στο κριτήριο του να θεωρήσουμε δύο πρότυπα ισοδύναμα ή όχι κατά την φάση εκμάθησης του αλγορίθμου. Όπως θα δούμε, η τιμή της ευαισθησίας θεωρείται βασική δεδομένου να παρατηρήσουμε υψηλές τιμές από σωστά ταξινομημένα πρότυπα. Οι ακόλουθες εξισώσεις υιοθετούνται σε κάθε βήμα ενημέρωσης του ART:

$$b_t = \|\mathbf{c}_i - \mathbf{u}_t\| = \min_{l=1}^k \|\mathbf{c}_l - \mathbf{u}_t\|$$

$$\begin{cases} \mathbf{c}_{k+1} \leftarrow \mathbf{u}_t & \text{if } b_t > \rho \\ \mathbf{c}_i = \mathbf{c}_i + \eta(\mathbf{u}_t - \mathbf{c}_i) & \text{αντίθετα} \end{cases} \quad (3.3)$$

3.2. Αναπαράσταση Θέσης και Τροχιάς

Πολλές προσεγγίσεις έχουν προταθεί με σκοπό να αναπαραστήσουν την ιστορία κίνησης, ή αλλιώς τροχιά, ενός κινητού χρήστη [21]. Υιοθετούμε ένα χωροχρονικό ιστορικό μοντέλο στο οποίο η τροχιά αναπαρίσταται ως η ακολουθία από 3-διαστάσεων σημεία (3DPs) τα οποία έχουν επισκεφθεί από τον κινητό χρήστη, δηλαδή, χρονοσημασμένα σημεία τροχιάς σε μία επιφάνεια 2-διαστάσεων (2D surface). Τα χωρικά χαρακτηριστικά σε αυτό το μοντέλο δηλώνουν γεωγραφικό πλάτος και μήκος.

Ας υποθέσουμε ότι $\mathbf{e} = (x, y, t)$ είναι ένα 3DP. Η *τροχιά του χρήστη* \mathbf{u} αποτελείται από αρκετά χρονοδιατεταγμένα 3DPs, $\mathbf{u} = [\mathbf{e}_i] = [\mathbf{e}_1, \dots, \mathbf{e}_N], i = 1, \dots, N$ και αποθηκεύεται σε μία βάση δεδομένων του συστήματος. Ισχύει ότι $t(\mathbf{e}_1) < t(\mathbf{e}_2) < \dots < t(\mathbf{e}_N)$, δηλαδή, χρονοσημασμένες συντεταγμένες. Οι x και y διαστάσεις υποδηλώνουν γεωγραφικό πλάτος και μήκος αντίστοιχα ενώ η t διάσταση δηλώνει τον χρόνο ($t(\cdot)$ επιστρέφει τη χρονική συντεταγμένη του \mathbf{e}). Ο χρόνος παίρνει τιμές μεταξύ 00:00 και 23:59. Για να

αποφύγουμε μία κατάσταση έκρηξης πληροφοριών, οι τροχιές περιλαμβάνουν χρονοσημασμένα σημεία τα οποία έχουν προκύψει από δειγματοληψία σε συγκεκριμένες χρονικές στιγμές. Συγκεκριμένα, δειγματοληπτούμε την κίνηση του κάθε χρήστη στα $1.66 \cdot 10^{-3}$ Hertz (δηλαδή, κάθε 10 λεπτά). Η δειγματοληψία σε πολύ υψηλά ποσοστά (δηλαδή στην τάξη του Hertz) είναι άνευ νοήματος, καθότι τα παραγόμενα σημεία θα έχουν υψηλή επικάλυψη.

Στον αλγόριθμο μας, το \mathbf{u} είναι μία ορισμένη σειρά από N 3DPs, δηλαδή, το \mathbf{u} είναι ένα $3 \cdot N$ διαστάσεων διάνυσμα. Έχουμε υιοθετήσει την τιμή $N = 6$ για τα πειράματά μας. Αυτό υποδηλώνει ότι εκτιμούμε την μελλοντική θέση ενός κινητού τερματικού από μία τροχιά 50 λεπτών (δηλαδή, 5 δείγματα). Συγκεκριμένα, σκοπεύουμε να ρωτήσουμε το σύστημα με μία $N-1$ 3DP σειρά έτσι ώστε ο αλγόριθμος μας να επιστρέψει ένα 3DP, το οποίο είναι η προβλεπόμενη θέση του κινητού τερματικού.

Μία ομάδα τροχιών \mathbf{c} αποτελείται από έναν ορισμένο αριθμό από 3DPs, $\mathbf{c} = [\mathbf{e}_i]$, $i = 1, \dots, N$ αποθηκευμένα στην βάση γνώσης. Σημειώστε ότι μία ομάδα τροχιών \mathbf{c} και μία τροχιά χρήστη \mathbf{u} είναι διανύσματα το ίδιου μήκους N . Αυτό είναι επειδή η \mathbf{c} , η οποία δημιουργείται από τον ART και βασίζεται πάνω σε άγνωστες τροχιές χρηστών, είναι ένα αντιπροσωπευτικό δρομολόγιο των κινήσεων του χρήστη. Επιπρόσθετα, η τροχιά ερώτησης \mathbf{q} αποτελείται από έναν αριθμό από 3DPs, $\mathbf{q} = [\mathbf{e}_j]$, $j = 1, \dots, N - 1$. Αξίζει να σημειωθεί ότι η \mathbf{q} είναι μία σειρά από $N-1$ 3DPs. Δοσμένης της \mathbf{q} με μία ιστορία από $N-1$ 3DPs προβλέπουμε το \mathbf{e}_N της πλησιέστερης \mathbf{c} ως την επόμενη κίνηση του χρήστη.

3.3. Αλγόριθμος Πρόβλεψης Κίνησης

Από την οπτική γωνία της μηχανικής μάθησης το υπό συζήτηση πρόβλημα της πρόβλεψης θέσης ανάγεται σε ένα $m + l$ μοντέλο [22]. Στα $m + l$ μοντέλα έχουμε m βήματα της τροχιών του χρήστη και θέλουμε να προβλέψουμε τη μελλοντική κίνηση του χρήστη μετά από l βήματα (τα βήματα έχουν χρονοσημασμένες συντεταγμένες). Στην περίπτωση μας, $m = N - 1$, δηλαδή, την τροχιά ερώτησης \mathbf{q} , ενώ $l = 1$, δηλαδή, το προβλεπόμενο \mathbf{e}_N . Αναπτύσσουμε έναν νέο χωροχρονικό αλγόριθμο (C) όπου δοσμένης της \mathbf{q} μπορούμε να προβλέψουμε το \mathbf{e}_N . Συγκεκριμένα, \mathbf{q} και \mathbf{c} είναι τροχιές διαφορετικού μήκους – διάστασης- έτσι χρησιμοποιούμε την απόσταση Hausdorff σαν μέτρο για να υπολογίσουμε την απόσταση $\|\mathbf{q} - \mathbf{c}\|$. Δοσμένης μίας τροχιών ερώτησης \mathbf{q} , ο προτεινόμενος αλγόριθμος C προσπαθεί να βρει την πλησιέστερη ομάδα \mathbf{c} στη βάση γνώσης και, τότε, να λάβει το \mathbf{e}_N ως το προβλεπόμενο 3DP. Για να αξιολογήσουμε το C, υπολογίζουμε την Ευκλείδεια απόσταση μεταξύ του προβλεπόμενου 3DP και του πραγματικού 3DP (δηλαδή, την πραγματική κίνηση του χρήστη). Αν αυτή η απόσταση είναι μεγαλύτερη από ένα προκαθορισμένο όριο σφάλματος θ τότε η πρόβλεψη δεν θα είναι επιτυχής. Αφού προβλεφθεί η μελλοντική θέση ενός κινητού τερματικού, ο αλγόριθμος C μπορεί να λάβει ανατροφοδότηση από το περιβάλλον εξετάζοντας το ενδεχόμενο η πρόβλεψη να ήταν επιτυχής ή όχι, και στη συνέχεια να αναδιοργανώσει ανάλογα τη βάση γνώσης [23].

Στην περίπτωση μας, η ανατροφοδότηση είναι το πραγματικό 3DP που παρατηρήθηκε κατά την κίνηση του τερματικού. Μπορούμε να έχουμε δύο βασικές εκδόσεις του αλγορίθμου C:

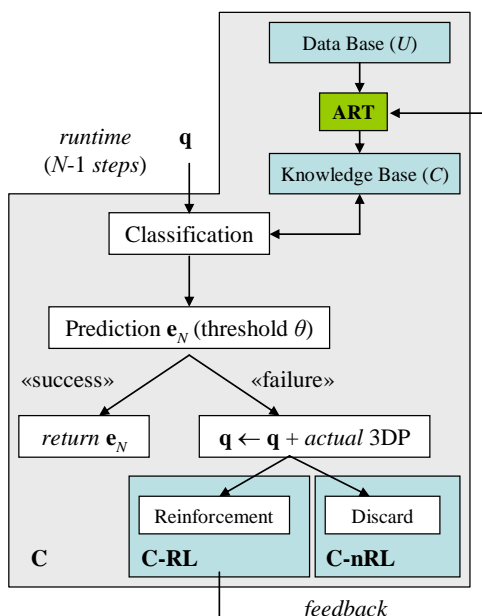
1. Τον C-RL αλγόριθμο, ο οποίος αλληλεπιδρά με το περιβάλλον και μαθαίνει νέα πρότυπα μέσω της ενισχυτικής μάθησης εφόσον συμβεί μία ανεπιτυχής πρόβλεψη, και,
2. Τον C-nRT αλγόριθμο, ο οποίος αγνοεί τις μη επιτυχείς προβλέψεις.

Συγκεκριμένα,

- Στην περίπτωση μίας εσφαλμένης πρόβλεψης, ο C-RL προσαρτά το πραγματικό 3DP στην \mathbf{q} και ενισχύει αυτήν την εκτεταμένη ακολουθία στο μοντέλο θεωρώντας την ως νέα γνώση, δηλαδή, ως μία άγνωστη συμπεριφορά της κίνησης του χρήστη.
- Στην περίπτωση μίας επιτυχούς πρόβλεψης, δεν ενισχύουμε τον C να μάθει. Μία επιτυχής πρόβλεψη αναφέρεται σε ένα καλά καθιερωμένο μοντέλο ικανό να χειρίζεται άγνωστες τροχιές του χρήστη.

Πίνακας 3.1. Ο υιοθετούμενος ART για τον C αλγόριθμο

1.	$j \leftarrow 1$
2.	$\mathbf{c}_j \leftarrow \mathbf{u}_j$
3.	For ($\mathbf{u}_t \in U$) Do
4.	$b_t = \ \mathbf{c}_j - \mathbf{u}_t\ = \min_{l=1, \dots, j} \ \mathbf{c}_l - \mathbf{u}_t\ $
5.	If ($b_t > \rho$) Then /*επέκτεινε τη γνώση*/ $j \leftarrow j + 1$
6.	$\mathbf{c}_j \leftarrow \mathbf{u}_t$
7.	Else
8.	$\mathbf{c}_j \leftarrow \mathbf{c}_j + \eta(\mathbf{u}_t - \mathbf{c}_j)$ /*ενημέρωσε το μοντέλο τοπικά*/
9.	End If
10.	End for



Σχήμα 3.1. Ο προτεινόμενος προσαρμοστικός αλγόριθμος C για προσαρμογή γνώσης και πρόβλεψη θέσης

Ο πυρήνας του προτεινόμενου αλγόριθμου είναι ο ART αλγόριθμος. Ο ART συγκεντρώνει άγνωστες τροχιές του χρήστη σε ήδη υπάρχουσες ομάδες τροχιών ή δημιουργεί νέες ομάδες τροχιών ανάλογα με την τιμή της ευαισθησίας. Ο ART παίρνει το \mathbf{u}_1 πρότυπο από το εισερχόμενο σύνολο U των προτύπων και το αποθηκεύει ως την \mathbf{c}_1 ομάδα στην βάση γνώσης. Για την t -ιοστή άγνωστη τροχιά του χρήστη η ακόλουθη διαδικασία ακολουθείται (βλέπε Πίνακα 1): Ο αλγόριθμος υπολογίζει την Ευκλείδεια

απόσταση b_t μεταξύ του \mathbf{u}_t και της πλησιέστερης c_i . Αν η b_t είναι μικρότερη από την ευαισθησία ρ τότε η c_i ενημερώνεται από το \mathbf{u}_t κατά έναν παράγοντα η . Στην αντίθετη περίπτωση, μία νέα ομάδα $c_i \equiv \mathbf{u}_t$ εισάγεται στην βάση γνώσης. Ο ART αλγόριθμος παρουσιάζεται στον Πίνακα 3.1.

Ας υποθέσουμε ότι τα T, P είναι υποσύνολα του U για τα οποία ισχύει ότι $T \subseteq P \subseteq U$. Το σύνολο των προτύπων T χρησιμοποιείται για την εκπαίδευση του C , το οποίο σημαίνει ότι, ο C αναπτύσσει μία βάση γνώσης σε αντιστοιχία με την εποπτευόμενη μέθοδο εκπαίδευσης. Το σύνολο P χρησιμοποιείται για την εκτέλεση on-line προβλέψεων. Εισάγουμε τις εκδόσεις C-RTL και C-nRTL, που είναι οι αλγόριθμοι C-RL και C-nRL που έχουν εκπαιδευτεί με το σύνολο T , αντίστοιχα. Επιπρόσθετα, άπαξ και το σύνολο T είναι κενό τότε ο C δεν εκπαιδεύεται εκ των προτέρων, σύμφωνα με το μοντέλο εκπαίδευσης μηδενικής γνώσης, και εκτελεί on-line πρόβλεψη με το σύνολο P . Σε αυτήν την περίπτωση, έχουμε τους C-RLnT και C-nRLnT που αντιστοιχούν στους C-RL και C-nRL, αντίστοιχα, όταν προβλέπεται κατά την φάση της εκπαίδευσης.

Επιπρόσθετα, προκειμένου για τον αλγόριθμο C να επιτευχθεί μία πρόβλεψη, υιοθετείται μία προσεγγιστική Hausdorff μετρική [24] για να εκτιμηθεί η απόσταση μεταξύ των \mathbf{q} και \mathbf{c} . Συγκεκριμένα, ο υιοθετούμενος τύπος υπολογίζει την απόσταση από σημείο-σε-διάνυσμα μεταξύ των $\mathbf{e}_j \in \mathbf{q}$ και \mathbf{c} , $\delta'(\mathbf{e}_j, \mathbf{c})$, ως ακολούθως:

$$\delta'(\mathbf{e}_j, \mathbf{c}) = \|\mathbf{f}_i - \mathbf{e}_j\|_{\min_{\mathbf{f}_i} |t(\mathbf{f}_i) - t(\mathbf{e}_j)|}$$

Όπου $\|\cdot\|$ είναι η Ευκλείδεια νόρμα για το $\mathbf{f}_i \in \mathbf{c}$ και \mathbf{e}_j . Η τιμή $\delta'(\mathbf{e}_j, \mathbf{c})$ υποδηλώνει την ελάχιστη απόσταση μεταξύ των \mathbf{e}_j και \mathbf{f}_i σε σχέση με τη χρονοσημασμένη πληροφορία της διαδρομής του χρήστη, η οποία είναι η Ευκλείδεια απόσταση των πλησιέστερων 3DPs στο χρόνο. Ως εκ τούτου, η συνολική απόσταση μεταξύ του $N - 1$ σε μήκος \mathbf{q} και του N σε μήκος \mathbf{c} υπολογίζεται ως

$$\delta_{N-1}(\mathbf{q}, \mathbf{c}) = \frac{1}{N-1} \sum_{\mathbf{e}_j \in \mathbf{q}} \delta'(\mathbf{e}_j, \mathbf{c}) \quad (3.4)$$

Το Σχήμα 3.1 απεικονίζει τη διαδικασία της πρόβλεψης της επόμενης κίνησης του χρήστη λαμβάνοντας υπόψη τις ποικίλες εκδόσεις του C αλγορίθμου. Συγκεκριμένα, άπαξ και έρθει μία τροχιά ερώτησης \mathbf{q} , τότε ο C προσπαθεί να ταξινομήσει την \mathbf{q} σε μία γνωστή c_i στην βάση γνώσης σε σχέση με την μετρική Hausdorff. Ο C αλγόριθμος επιστρέφει το προβλεπόμενο $\mathbf{e}_N \in c_i$ του πλησιέστερου c_i στο \mathbf{q} . Στην περίπτωση που ένα τέτοιο αποτέλεσμα αναφέρεται σε μία μη επιτυχημένη πρόβλεψη σε σχέση με το προκαθορισμένο όριο σφάλματος θ τότε ο C-RLT (ή ο C-RLnT) επεκτείνουν το διάνυσμα \mathbf{q} με το πραγματικό 3DP και εισάγουν το \mathbf{q} μέσα στη βάση γνώσης για περαιτέρω μάθηση σύμφωνα με τον αλγόριθμο στον Πίνακα I (ανατροφοδότηση). Υιοθετώντας την μη-ενισχυτική μέθοδο μάθησης, ο C (οι εκδόσεις C-nRLT/nRLnT) δεν δίνουν ανατροφοδότηση στη βάση γνώσης (καμία προσαρμογή).

3.4. Αξιολόγηση Προσαρμογής και Πρόβλεψης Θέσης

Αξιολογήσαμε το προσαρμοστικό μας μοντέλο με σκοπό να εκτιμήσουμε την απόδοση του. Στα πειράματά μας, ο συνολικός χώρος κίνησης του χρήστη καλύπτει επιφάνεια 540 km^2 . Ο συγκεκριμένος χώρος παράγεται από πραγματικά GPS ίχνη τα οποία έχουν συλληφθεί στην Δανία [25]. Τα GPS ίχνη τροφοδοτήθηκαν στο μοντέλο μας και παρατηρήθηκε η απόδοση του C συστήματος σε σχέση με τις προκαθορισμένες μετρικές. Ο Πίνακας 3.2 υποδεικνύει τις παραμέτρους που χρησιμοποιήθηκαν στα

πειράματα μας. Οι τιμές των παραμέτρων αναφέρονται σε πραγματικό σύνολο GPS δεδομένων.

Πίνακας 3.2. Πειραματικές Παράμετροι

Παράμετρος	Τιμή	Σχόλιο
Ρυθμός μάθησης (η)	0.5	Στην περίπτωση ενός νέου προτύπου \mathbf{u}_b , η πλησιέστερη ομάδα \mathbf{c}_i κινείται προς το \mathbf{u}_i κατά τη μισή χωρική και χρονική απόσταση.
Χωρικός συντελεστής της ευαισθησίας (ρ_s)	100 m	Δύο 2D σημεία θεωρούνται διαφορετικά αν η χωρική διάσταση υπερβαίνει τα 100 μέτρα.
Χρονικός συντελεστής της ευαισθησίας (ρ_t)	10 min	Δύο χρόνο-σημάνσεις θεωρούνται διαφορετικές αν η χρονική απόσταση υπερβαίνει τα 10 λεπτά.
Κατώφλι ακρίβειας / ακρίβεια θέσης (θ)	10 m	Η προβλεπόμενη θέση βρίσκεται μέσα σε έναν κύκλο ακτίνας 10 μέτρων από την πραγματική θέση.

Τα GPS ίχνη μαζί με 1200 πρότυπα που περιέχονταν σε αυτά προεπεξεργάστηκαν. Παράξαμε δύο αρχεία εκπαίδευσης όπως απεικονίζονται στο Σχήμα 3.2. Το πρώτο αρχείο εκπαίδευσης, *TrainA*, παράχθηκε από το πρώτο μισό των εγγραφών των GPS ιχνών. Το δεύτερο αρχείο εκπαίδευσης, *TrainB*, αποτελείται από μία και μόνο εγγραφή ίχνους. Το πρώτο αρχείο δοκιμής, *TestA*, παράγεται από ολόκληρο το σύνολο των εγγραφών του ίχνους, συμπεριλαμβανομένου – σε αύξουσα σειρά – του πρώτου μισού των GPS ιχνών και του άλλου μισού των άγνωστων ιχνών. Τελικά, το δεύτερο αρχείο δοκιμής, *TestB*, παράγεται από ολόκληρο το σύνολο εγγραφών των GPS ιχνών, συμπεριλαμβανομένου - σε αύξουσα σειρά – του δεύτερου μισού από τα άγνωστα ίχνη και από το πρώτο μισό από τα GPS ίχνη. Κατά τη δημιουργία των αρχείων εκπαίδευσης και δοκιμής, εισαγάγαμε τεχνητά λευκό θόρυβο στις εγγραφές των ιχνών.

Αναλύσαμε την κίνηση του χρήστη και βρήκαμε ότι ο χρήστης κινείται από την *TownA* στην *TownB* είτε απευθείας (δηλαδή, ένα πρότυπο) ή έμμεσα μέσω της *TownC* (δηλαδή, νέο πρότυπο). Αυτά τα πρότυπα είναι αποθηκευμένα στα πρώτα 600 πρότυπα του *TestA*. Κατά την ώρα που ο χρήστης αλλάζει συμπεριφορά και ανακαλύπτει έναν τρίτο έμμεσο τρόπο για να πάει από την *TownA* στην *TownB* μέσω της *TownD* (δηλαδή, νεότερο πρότυπο), αυτό το πρότυπο αποθηκεύεται στα δεύτερα 600 πρότυπα του *TestA*. Αποδεικνύουμε ότι ο προτεινόμενος αλγόριθμος ανακαλύπτει τέτοιες αλλαγές της συμπεριφοράς του χρήστη και προσαρμόζει ανάλογα τη βάση γνώσης του.

GPS Pattern Instances (i.e., the U set)

$$\begin{array}{ccc}
 \mathbf{u}_1 & & \mathbf{u}_{600} & & \mathbf{u}_{1200} \\
 \hline
 \underline{TrainA} = \{\mathbf{u}_1, \dots, \mathbf{u}_{600}\} & & \underline{TestA} = \{\mathbf{u}_1, \dots, \mathbf{u}_{600}, \mathbf{u}_{601}, \dots, \mathbf{u}_{1200}\} \\
 \underline{TrainB} = \{\mathbf{u}_1\} & & \underline{TestB} = \{\mathbf{u}_{601}, \dots, \mathbf{u}_{1200}, \mathbf{u}_1, \dots, \mathbf{u}_{600}\}
 \end{array}$$

Σχήμα 3.2. Τα δημιουργημένα GPS αρχεία ιχνών για τον πειραματισμό

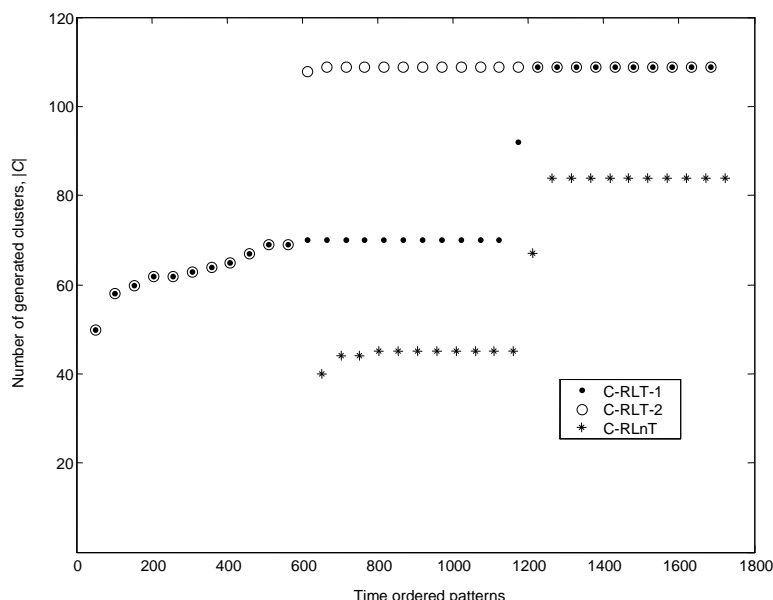
Πρέπει να αξιολογήσουμε τον προτεινόμενο αλγόριθμο ποσοτικά και ποιοτικά. Εισάγουμε τις επόμενες ποσοτικές και ποιοτικές παραμέτρους:

1. Την ακρίβεια που επιτυγχάνεται από το μοντέλο πρόβλεψης. Συγκεκριμένα όσο πιο υψηλή είναι η ακρίβεια τόσο πιο ακριβείς είναι οι αποφάσεις για την μελλοντική θέση του χρήστη.
2. Το μέγεθος της βάσης γνώσης. Πρέπει να υιοθετήσουμε λύσεις με το χαμηλότερο δυνατό μέγεθος βάσης γνώσης. Τέτοιες λύσεις είναι πολύ πιο αποδοτικές και εφικτές από πλευράς υλοποίησης.
3. Την ικανότητα του αλγορίθμου να αντιδρά τάχιστα σε αλλαγές στο πρότυπο κίνησης του χρήστη/κινητού τερματικού και να αναπροσαρμόζεται.

Ορίζουμε την ακρίβεια, p , ως το κλάσμα των σωστά προβλεπόμενων θέσεων, p_+ , εναντίον του συνολικού αριθμού των προβλέψεων που έχουν γίνει από τον αλγόριθμο C , p_{total} , η οποία είναι,

$$p = \frac{P_+}{P_{total}}$$

Στις επόμενες υποενότητες, αξιολογούμε τις διάφορες εκδόσεις του C σε σχέση με τις μεθόδους εκπαίδευσης και μάθησης εξετάζοντας την σύγκλιση του αλγορίθμου (ταχύτητα μάθησης και προσαρμογής) και την παραγόμενη ακρίβεια στην πρόβλεψη των μελλοντικών θέσεων.



Σχήμα 3.3. Σύγκλιση των C-RLT/RLnT βασισμένα σε ενισχυτική μέθοδο μάθησης.

3.4.1. Σύγκλιση των C-RLT και C-RLnT

Ο αλγόριθμος C συγκλίνει όταν η βάση γνώσης δεν επεκτείνεται με άγνωστα πρότυπα, δηλαδή το σύνολο U δεν εξελίσσεται. Στο Σχήμα 3.3, σχεδιάζουμε τον αριθμό των ομάδων $|U|$ που δημιουργούνται από τα μοντέλα C-RLT/RLnT με ενισχυτική μάθηση κατά τη διάρκεια της φάσης δοκιμής. Ο οριζόντιος άξονας δεικνύει τα εισερχόμενα (χρονοσημασμένα) GPS πρότυπα. Η σημασμένη με κουκίδα (\cdot) γραμμή απεικονίζει την συμπεριφορά του C-RLT-1 μοντέλου που έχει εκπαιδευτεί με το *TrainA* και έχει δοκιμαστεί με το *TestA*. Στην φάση της εκπαίδευσης, Τα πρώτα 600 πρότυπα του *TrainA* έχουν σταδιακά δημιουργήσει 70 ομάδες στο U . Στην φάση της δοκιμής, τα πρώτα 600 πρότυπα είναι γνωστά στον αλγόριθμο και συνεπώς δεν υπάρχει καμία δημιουργία νέας ομάδας. Από την άλλη πλευρά, στα υπόλοιπα 600 άγνωστα πρότυπα,

ο αριθμός των ομάδων κλιμακώνεται μέχρι τις 110 υποδηλώνοντας ότι ο ART αλγόριθμος έχει «ενισχυθεί» να μάθει αυτά τα νέα πρότυπα.

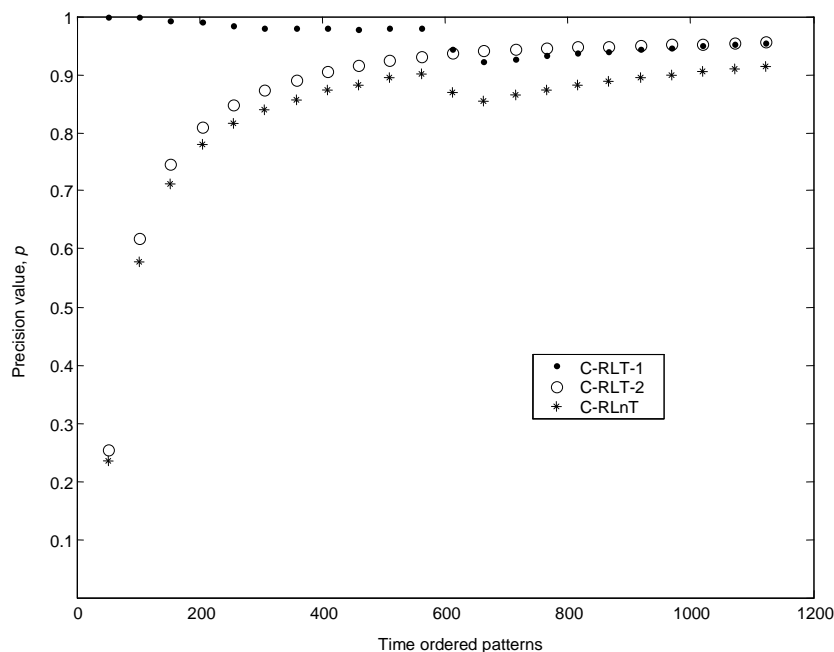
Η σημασμένη με κύκλο (ο) γραμμή απεικονίζει το C-RLT-2 μοντέλο, το οποίο έχει εκπαιδευτεί με το *TrainA* και έχει δοκιμαστεί με το *TestB*. Δεδομένου ότι το αρχείο εκπαίδευσης είναι το ίδιο με αυτό του C-RLT-1 μοντέλου, οι πρώτες δημιουργημένες ομάδες είναι ίδιες στον αριθμό ($|U| = 70$). Στην φάση της δοκιμής, παρατηρούμε μία σημαντική διαφορά. Ο ART δεν γνωρίζει τα δεύτερα 600 άγνωστα πρότυπα, έτσι, σταδιακά «ενισχύεται» να μάθει νέα πρότυπα μέχρι τις 110 ομάδες. Στα επόμενα 600 γνωστά πρότυπα, ο C-RLT-2 δεν χρειάζεται να μάθει επιπλέον ομάδες έτσι κατασταλάζει στις 110 ομάδες.

Τώρα θα εξετάσουμε τη συμπεριφορά του C-RLnT μοντέλου αντίστοιχα με την μηδενικής-γνώσης μέθοδο εκπαίδευσης. Η σημασμένη με αστερίσκο (*) γραμμή απεικονίζει την φάση εκπαίδευσης (με το *TrainB*) ακολουθούμενη από τη φάση δοκιμής (με το *TestA*) του C-RLnT. Σε αυτή την περίπτωση, έχουμε έναν αυξητικό ART που δεν έχει την ανάγκη να εκπαιδευτεί. Για τεχνικούς λόγους συνέπειας, απαιτεί μόνο ένα απλό πρότυπο, το οποίο είναι η μοναδική ομάδα στη βάση γνώσης στην αρχή. Στην φάση δοκιμής, για τα πρώτα 600 πρότυπα του *TestA* παρατηρούμε μία προοδευτική δημιουργία ομάδων (έως 45 ομάδες). Για τα επόμενα 600 άγνωστα πρότυπα, παρατηρούμε επίσης μία κλιμακωτή δημιουργία ομάδων (έως 85 ομάδες) ακολουθούμενη από σύγκλιση. Συγκρίνοντας τα μοντέλα C-RLT-1/-2 και C-RLnT, το τελευταίο επιτυγχάνει τον ελάχιστο αριθμό ομάδων (22.72% λιγότερο κόστος αποθήκευσης). Αυτό είναι λόγω του γεγονότος ότι ο C-RLnT ξεκινά τη μάθηση μόνο από ανεπιτυχείς προβλέψεις με έναν αυξητικό τρόπο προσαρμόζοντας την προ-υπάρχουσα βάση γνώσης με νέες περιπτώσεις. Ωστόσο, πρέπει να λάβουμε υπόψη μας την ακρίβεια πρόβλεψης έτσι ώστε να φθάσουμε σε ασφαλή συμπεράσματα για την αποδοτικότητα και την αποτελεσματικότητα των προτεινόμενων μοντέλων.

3.4.2. Ακρίβεια των C-RLT και C-RLnT

Στο Σχήμα 3.4, εξετάζουμε την ακρίβεια που επιτυγχάνεται από τους αλγόριθμους που υιοθετούν ενισχυτική μάθηση. Ο κατακόρυφος άξονας απεικονίζει την τιμή της ακρίβειας p που επιτυγχάνεται κατά τη φάση της δοκιμής. Η σημασμένη με κουκίδα (·) γραμμή απεικονίζει την ακρίβεια του C-RLT-1 μοντέλου που έχει εκπαιδευτεί με το *TrainA* και έχει δοκιμαστεί με το *TestA*. Κατά τη διάρκεια της φάσης δοκιμής, για τα 600 γνωστά πρότυπα ο C-RLT-1 επιτυγχάνει τιμή ακρίβειας που κυμαίνεται από 97% έως και 100%. Στα επόμενα 600 άγνωστα πρότυπα, παρατηρούμε ότι για τα πρώτα παραδείγματα η ακρίβεια πέφτει ομαλά στο 95% και όπως ο C-RLT-1 ενισχύεται να μάθει, δηλαδή, να μάθει νέες ομάδες και να βελτιστοποιήσει τις παλιές, η ακρίβεια συγκλίνει στο 96%.

Η σημασμένη με κύκλο (ο) γραμμή απεικονίζει την συμπεριφορά της ακρίβειας για το C-RLT-2 μοντέλο δοκιμασμένο με το *TestB* και εκπαιδευμένο με το *TrainA*. Με τα πρώτα 600 συνολικά άγνωστα πρότυπα κατά τη φάση της δοκιμής, ο C-RLT-2 επιτυγχάνει ακρίβεια από 26% έως και 96%. Αυτό υποδηλώνει ότι το μοντέλο ακόμα μαθαίνει κατά τη φάση της δοκιμής υιοθετώντας την μέθοδο ενισχυτικής μάθησης αυξάνοντας την τιμή της ακρίβειας. Στα επόμενα 600 γνωστά πρότυπα, το μοντέλο δεν έχει τίποτα να μάθει και η τιμή της ακρίβειας συγκλίνει στο 96%.



Σχήμα 3.4. Ακρίβεια των C-RLT/RLnT βασιζόμενα σε ενισχυτική μέθοδο μάθησης.

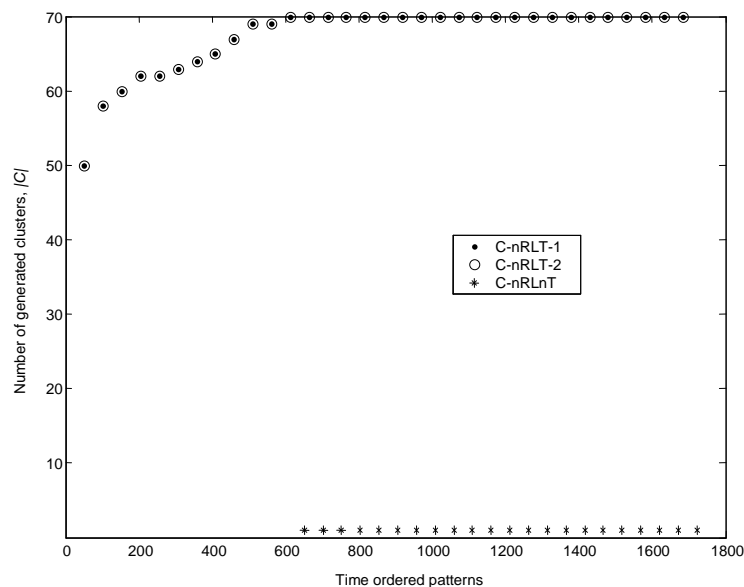
Η σημασμένη με αστερίσκο (*) γραμμή απεικονίζει τη συμπεριφορά της ακρίβειας για το C-RLnT μοντέλο δοκιμασμένο με το *TestA* και εκπαιδευμένο με το *TestB*. Σε αυτήν την περίπτωση, ο C-RLnT εκπαιδεύεται με ένα μόνο παράδειγμα προτύπου, δηλαδή, ο αλγόριθμος είναι πλήρως αυξητικός, έτσι, όλα τα παραδείγματα αντιμετωπίζονται ως άγνωστα. Στην φάση της δοκιμής, για τα πρώτα 600 πρότυπα, το μοντέλο επιτυγχάνει ακρίβεια, η οποία κυμαίνεται από 25% έως 91% (λόγω του μηχανισμού της ενίσχυσης). Στα επόμενα 600 πρότυπα, μπορούμε να παρατηρήσουμε ότι για τα πρώτα παραδείγματα η ακρίβεια πέφτει ομαλά στο 88% και όπως το μοντέλο «ενισχύεται» να μάθει, η ακρίβεια προοδευτικά συγκλίνει στο 93%. Από τα πρώτα 200 πρότυπα ξεκινώντας από το 601-ιοστο πρότυπο, ο αλγόριθμος ενημερώνει τοπικά τη δομή της γνώσης του, όπως απεικονίζεται στο Σχήμα 3.4. Όταν ο αλγόριθμος έχει εκπαιδευτεί με αυτές τις τοπικές ενημερώσεις, η ακρίβεια του σταδιακά συγκλίνει έως ότου παρατηρηθεί μία νέα αλλαγή στη συμπεριφορά της κίνησης του χρήστη.

Προφανώς, η υιοθεσία της μεθόδου εκπαίδευσης, δηλαδή, το C-RLT-1 και C-RLT-2 μοντέλο, επιτυγχάνει καλύτερη ακρίβεια. Ωστόσο, αν συσχετίσουμε τα αποτελέσματά μας με τα αποτελέσματα που φαίνονται στο Σχήμα 3.3, αποφαινόμαστε ότι μία μικρή βελτίωση στη ακρίβεια έχει ένα προφανές κόστος αποθήκευσης. Συγκεκριμένα, πρέπει να αποθηκεύσουμε 110 ομάδες, στην περίπτωση του C-RLT, συγκρίνοντας με 85 ομάδες στη περίπτωση του C-RLnT (22.72% λιγότερο κόστος αποθήκευσης). Επιπρόσθετα, τα πρότυπα κίνησης του χρήστη μπορεί να αλλάξουν επανειλημμένως προϊόντος του χρόνου. Έτσι, υιοθετώντας την μέθοδο εκπαίδευσης, θα πρέπει σε τακτά χρονικά διαστήματα να εκπαιδεύουμε και να ανοικοδομούμε το μοντέλο. Ωστόσο, πρέπει να εξετάσουμε τη συμπεριφορά του C-nRLT και C-nRLnT μοντέλων έτσι ώστε να είμαστε σε θέση να αποφασίσουμε ποιο είναι το κατάλληλο μοντέλο για τον συζητούμενο τομέα.

3.4.3. Σύγκλιση των C-nRLT και C-nRLnT

Στο Σχήμα 3.5, βλέπουμε τον αριθμό των ομάδων που δημιουργούνται από τους C-nRLT/nRLnT υποθέτοντας τη μη-ενισχυτική μέθοδο εκμάθησης. Ο κατακόρυφος

άξονας υποδεικνύει τις δημιουργημένες ομάδες κατά τη διάρκεια των φάσεων εκπαίδευσης και δοκιμής. Η σημασμένη με κουκίδα (·) γραμμή απεικονίζει την φάση εκπαίδευσης με το *TrainA* και την φάση δοκιμής με το *TestA* για το C-nRLT-1 μοντέλο. Όπως μπορούμε να δούμε στην φάση εκπαίδευσης, τα πρώτα 600 πρότυπα (*TrainA*) έχουν δημιουργήσει 70 ομάδες. Στην φάση της δοκιμής, το μοντέλο γνωρίζει τα πρώτα 600 πρότυπα έτσι δεν υπάρχει κάποια δημιουργία νέας ομάδας. Σε αντιδιαστολή με το μοντέλο C-RLT-1 (Ενότητα 3.4.1), τα εναπομείναντα 600 άγνωστα πρότυπα δεν μπορούν να μαθευτούν περαιτέρω από τον C-nRLT-1, έτσι, ο συνολικός αριθμός των ομάδων παραμένει σταθερός στις 70 ομάδες.



Σχήμα 3.5. Σύγκλιση των C-nRLT/-nRLnT βασιζόμενα σε μη-ενισχυτική μέθοδο μάθησης.

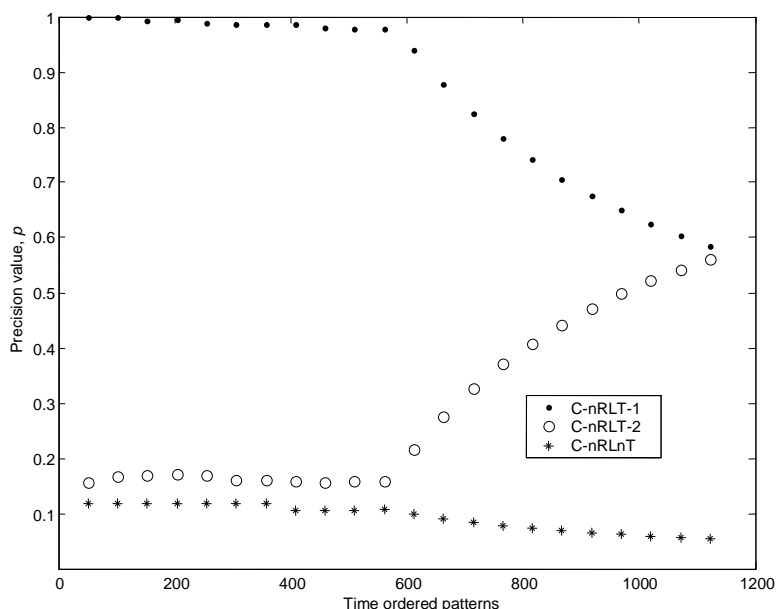
Η σημασμένη με κύκλο (o) γραμμή απεικονίζει την φάση εκπαίδευσης με το *TrainA* και τη φάση δοκιμής με το *TestB* για το C-nRLT-2 μοντέλο. Εφόσον το αρχείο εκπαίδευσης είναι το ίδιο, (δηλαδή, *TrainA*), ο αριθμός των δημιουργημένων ομάδων είναι ο ίδιος (70 ομάδες) όπως και πριν. Σε αντίθεση με τον C-RLT-2, στην φάση της δοκιμής δεν παρατηρούμε καμία αλλαγή στον αριθμό των ομάδων. Το μοντέλο δεν γνωρίζει τα δεύτερα 600 άγνωστα πρότυπα και δεν ενισχύεται να μάθει νέα, έτσι, ο συνολικός αριθμός των ομάδων παραμένει σταθερός 70. Στα επόμενα 600 γνωστά πρότυπα, το μοντέλο διατηρεί τον ίδιο αριθμό ομάδων (70 ομάδες).

Τελικά, η σημασμένη με αστερίσκο (*) γραμμή απεικονίζει τον αριθμό των δημιουργημένων ομάδων για το C-nRLnT μοντέλο το οποίο έχει δοκιμαστεί με το *TestA*. Σε αυτήν την περίπτωση, έχουμε ένα αυξητικό μη-ενισχυτικό μοντέλο το οποίο δεν εκπαιδεύεται. Στην φάση της δοκιμής, για τα 600 άγνωστα πρότυπα παρατηρούμε ότι δεν υπάρχει καμία επιπλέον δημιουργία ομάδας εκτός από τη μοναδική ομάδα (του αρχείου *TrainB*). Επιπρόσθετα, για τα επιπλέον 600 άγνωστα πρότυπα, δεν παρατηρείται καμία επιπρόσθετη δημιουργία ομάδας. Παρατηρήσαμε μόνο ότι τα μοντέλα C-nRLT/-nRLnT επιτυγχάνουν ένα μικρότερο αριθμό ομάδων από ότι τα C-RLT/-RLnT μοντέλα.

3.4.4. Ακρίβεια των C-nRLT και C-nRLnT

Στο Σχήμα 3.6, βλέπουμε την ακρίβεια των μοντέλων που υιοθετούν τη μη-ενισχυτική μάθηση. Ο κατακόρυφος άξονας δείχνει την τιμή της ακρίβειας που επιτυγχάνεται κατά τη δημιουργία της φάσης δοκιμής. Η σημασμένη με κουκίδα (·) γραμμή αναφέρεται στο

μοντέλο C-nRLT-1 (το οποίο έχει εκπαιδευτεί με το *TrainA*, και έχει δοκιμαστεί με το *TestA*). Στην φάση δοκιμής, τα πρώτα 600 γνωστά πρότυπα επιτυγχάνουν επίπεδα ακρίβειας που κυμαίνονται από 97% έως 100%. Στα επόμενα 600 άγνωστα πρότυπα, παρατηρούμε ότι η τιμή της ακρίβειας πέφτει σταδιακά στο 59%. Αυτό αποδίδεται στο γεγονός ότι δεν υπάρχουν νέες ομάδες για να βελτιστοποιήσουν τις παλιές λόγω της έλλειψης του μηχανισμού ενισχυτικής μάθησης. Αυτό υποδηλώνει ότι το C-nRLT-1 μοντέλο δεν προσαρμόζεται σε νέα γνώση.



Σχήμα 3.6. Ακρίβεια των C (-nRLT/-nRLnT) βασιζόμενα σε μη-ενισχυτική μέθοδο μάθησης.

Η σημασμένη με κύκλο (ο) γραμμή αναφέρεται στο μοντέλο C-nRLT-2 (το οποίο έχει εκπαιδευτεί με το *TrainA*, και έχει δοκιμαστεί με το *TestB*). Στην φάση της δοκιμής, τα πρώτα 600 άγνωστα πρότυπα επιτυγχάνουν επίπεδα ακρίβειας που κυμαίνονται από 16% έως 19% επειδή ο μηχανισμός ενίσχυσης δεν χρησιμοποιείται και, έτσι, δεν μπορούν να παραχθούν νέες ομάδες. Στα επόμενα 600 γνωστά πρότυπα, παρατηρούμε ότι η τιμή της ακρίβειας συγκλίνει στο 59%

Τελικά, η σημασμένη με αστερίσκο (*) γραμμή απεικονίζει τη συμπεριφορά της ακρίβειας του C-nRLnT μοντέλου (το οποίο εκπαιδεύεται με το *TrainB* και δοκιμάζεται με το *TestA*). Σε αυτό το σενάριο, ο αλγόριθμος είναι πλήρως αυξητικός και όλα τα παραδείγματα αντιμετωπίζονται ως άγνωστα. Όπως μπορούμε να δούμε στην φάση της δοκιμής τα πρώτα 600 πρότυπα επιτυγχάνουν επίπεδα ακρίβειας που κυμαίνονται από 10% έως 14% και, στα επόμενα 600 πρότυπα, παρατηρούμε ότι η τιμή της ακρίβειας πέφτει ομαλά στο 7%. Προφανώς, τα C-nRLT /-nRLnT μοντέλα επιτυγχάνουν πολύ μικρότερη ακρίβεια από ότι τα αντίστοιχα C-RLT /-RLnT αφού τα πρώτα μοντέλα δεν υποστηρίζουν προσαρμοστικότητα.

Επομένως, η υιοθεσία της μεθόδου ενισχυτικής μάθησης για την πρόβλεψη της θέσης παράγει καλύτερα αποτελέσματα από ότι ένας μη-ενισχυτικός αλγόριθμος. Κάθε προσέγγιση έχει κάποια πλεονεκτήματα αλλά και περιορισμούς. Αν το κύριο μέλημα μας είναι να κρατήσουμε το κόστος αποθήκευσης όσο πιο μικρό γίνεται, τότε πρέπει να επιλεγούν τα μοντέλα C-RLnT και C-nRLnT. Αν η κινητή εφαρμογή επίγνωσης-πλαισίου στοχεύει να μεγιστοποιήσει την ποιότητα της υπηρεσίας σε σχέση με την ακρίβεια, ενώ παράλληλα να διατηρήσει το κόστος αποθήκευσης σταθερό, τότε πρέπει να υιοθετηθεί το μοντέλο C-RLnT. Ο Πίνακας 3.3 συνοψίζει τα συμπεράσματα μας σε σχέση με τη

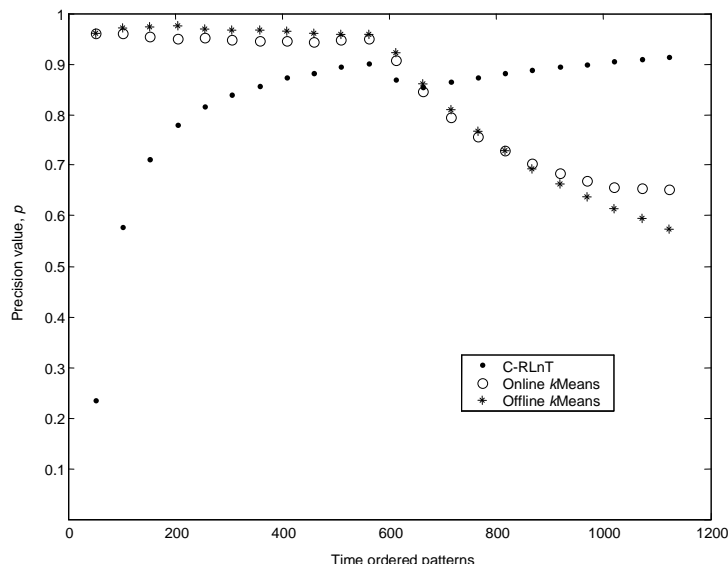
σύγκριση των τεσσάρων εναλλακτικών μοντέλων τα οποία συζητήθηκαν σε αυτές τις ενότητες. Συγκρίνοντας τις τέσσερις εκδόσεις του C αλγορίθμου, μπορούμε καθαρά να ξεχωρίσουμε το C-RLnT μοντέλο ως τον πιο αποδοτικό αλγόριθμο για την πρόβλεψη θέσης.

Πίνακας 3.3. Σύγκριση Μοντέλων

Μετρική/Μοντέλο	C-RLT	C-RLnT	C-nRLT	C-nRLnT
Ακρίβεια	++	+	-	--
Αποθήκευση	--	+	--	++
++: Άριστη απόδοση, --: Πολύ χαμηλή απόδοση				

3.5. Σύγκριση με άλλα Μοντέλα

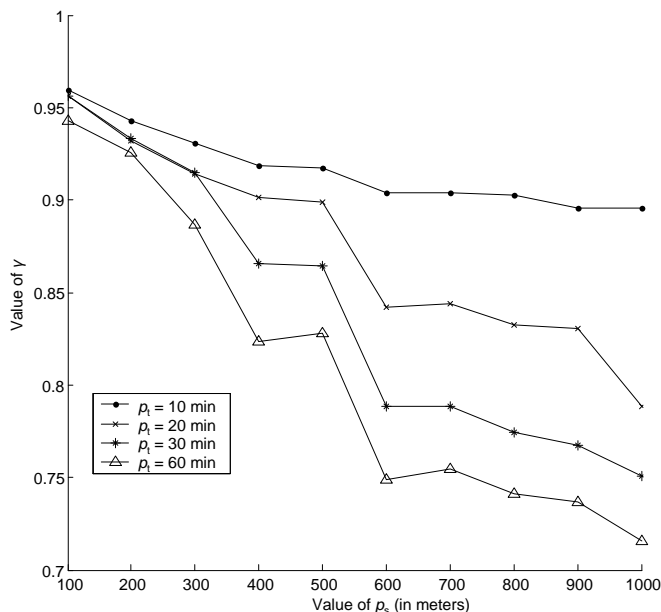
Συγκρίνουμε το C-RLnT μοντέλο με άλλα γνωστά μοντέλα που μπορούν να χρησιμοποιηθούν για την πρόβλεψη θέσης. Τέτοια μοντέλα υλοποιούν τους Offline *k*Means και Online *k*Means αλγορίθμους. Αυτά τα μοντέλα απαιτούν έναν προκαθορισμένο αριθμό από $k > 1$ αρχικές ομάδες για την δημιουργία της αντίστοιχης βάσης γνώσης. Πρέπει να επισημάνουμε εδώ ότι, όσο μεγαλύτερο είναι το k τόσο μεγαλύτερη είναι και η τιμή της ακρίβειας που επιτυγχάνεται από τους Offline και Online *k*Means. Στην περίπτωση μας, θα μπορούσαμε να θέσουμε το $k = 110$, που είναι ο αριθμός σύγκλισης των ομάδων για τα C-RL μοντέλα (Ενότητα 3.4). Για το C-RLnT, χρησιμοποιούμε το *TrainB* για την φάση εκπαίδευσης και το *TestA* για την φάση δοκιμής (αυτό το μοντέλο υιοθετεί τη μέθοδο εκπαίδευσης μηδενικής-γνώσης). Επιπρόσθετα, για τα Offline και Online *k*Means μοντέλα χρησιμοποιούμε το *TrainA* για την φάση εκπαίδευσης και το *TestA* για τη φάση δοκιμής επειδή και τα δύο μοντέλα απαιτούν $k > 1$ αρχικές ομάδες.



Σχήμα 3.7. Σύγκριση του C-RLnT με τα Offline/Online *k*Means μοντέλα.

Το Σχήμα 3.7 απεικονίζει την ακρίβεια που επιτυγχάνεται από τα μοντέλα C-RLnT (η σημασμένη με κουκίδα (·) γραμμή), Offline *k*Means (η σημασμένη με αστερίσκο (*) γραμμή) και Online *k*Means (η σημασμένη με κύκλο (o) γραμμή). Ο οριζόντιος άξονας

αναπαριστά τα διατεταγμένα παραδείγματα και ο κάθετος άξονας αναπαριστά την επιτευχθείσα ακρίβεια. Μπορούμε να παρατηρήσουμε στα πρώτα 600 πρότυπα ότι ο C-RLnT επιτυγχάνει επίπεδα ακρίβειας που κυμαίνονται από 25% έως 91% υποδεικνύοντας προσαρμογή σε νέα γνώση. Αυτό οφείλεται στον μηχανισμό ενίσχυσης (ο C-RLnT αναγνωρίζει και μαθαίνει νέες κινήσεις του χρήστη). Στα επόμενα 600 πρότυπα παρατηρούμε ότι για τα πρώτα παραδείγματα, η ακρίβεια πέφτει ομαλά στο 88% και καθώς η βάση γνώσης προσαρμόζεται σε νέες κινήσεις και βελτιστοποιεί τις ήδη υπάρχουσες, η ακρίβεια συγκλίνει στο 93%.



Σχήμα 3.8. Η συμπεριφορά της παραμέτρου γ σε σχέση με τους χρονικούς και χωρικούς συντελεστές του κατωφλίου ευαισθησίας.

Στην περίπτωση του Offline k Means, παρατηρούμε ότι για τα πρώτα 600 πρότυπα, επιτυγχάνει επίπεδα ακρίβειας που κυμαίνονται από 96% έως 98% δεδομένου ότι οι αρχικές ομάδες έχουν καθοριστεί να είναι $k = 110$. Στα επόμενα 600 πρότυπα παρατηρούμε ότι η ακρίβεια πέφτει απότομα και συγκλίνει στο 57% καθώς η βάση γνώσης δεν ενημερώνεται από άγνωστες κινήσεις του χρήστη. Υιοθετώντας τον Online k Means παρατηρούμε ότι για την φάση της δοκιμής (τα πρώτα 600 πρότυπα) επιτυγχάνει επίπεδα ακρίβειας που κυμαίνονται από 94% έως 97% δεδομένου του αρχείου εκπαίδευσης *TrainA*. Στα επόμενα 600 πρότυπα παρατηρούμε ότι για τα πρώτα παραδείγματα η ακρίβεια πέφτει μάλλον ομαλά στο 86% και, όσο η βάση γνώσης προσαρμόζεται αυξητικά σε νέα πρότυπα, η τιμή της ακρίβειας συγκλίνει στο 65%. Προφανώς, συγκρίνοντας αυτά τα τρία μοντέλα, το πιο κατάλληλο μοντέλο για πρόβλεψη θέσης είναι το C-RLnT αφού:

- (i) Επιτυγχάνει μεγαλύτερη ακρίβεια μέσω του μοντέλου προσαρμογής και,
- (ii) Απαιτεί βάση γνώσης μικρότερου μεγέθους (δηλαδή, λιγότερες ομάδες) από ότι τα μοντέλα Offline/Online k Means.

Ως αυτό το σημείο έχουμε καταλήξει ότι το μοντέλο C-RLnT επιτυγχάνει καλή ακρίβεια με περιορισμένες απαιτήσεις μνήμης, που είναι πολύ σημαντικές παράμετροι για τα κινητά συστήματα επίγνωσης-πλαισίου. Ωστόσο, πρέπει να κάνουμε κάποιες δοκιμές με τον C-RLnT με σκοπό να καθορίσουμε την καλύτερη τιμή για την χωροχρονική παράμετρο ευαισθησίας ρ . Με άλλα λόγια, σκοπεύουμε να καθορίσουμε τις καλύτερες τιμές και για τους δύο συντελεστές της ευαισθησίας, δηλαδή, τον χωρικό συντελεστή ρ_s και τον χρονικό συντελεστή ρ_t έτσι ώστε να επιτύχουμε την υψηλότερη ακρίβεια με τις

χαμηλότερες απαιτήσεις σε μνήμη. Εισάγουμε το σταθμισμένο άθροισμα γ ως ακολούθως:

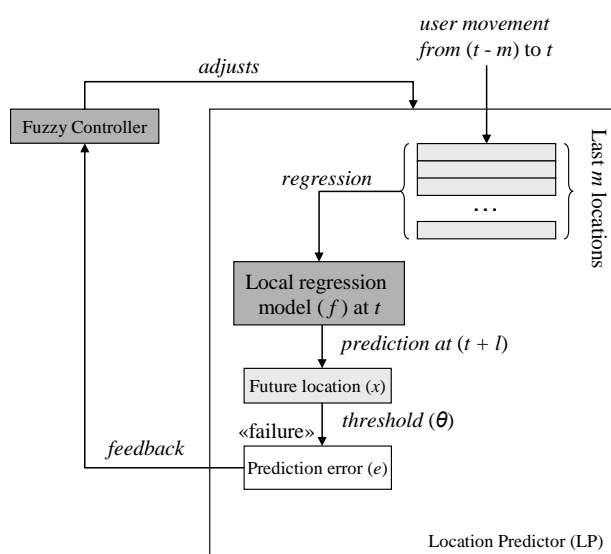
$$\gamma = w \cdot p + (1 - w) \cdot (1 - a)$$

Όπου a είναι το ποσοστό των δημιουργημένων ομάδων από τον αλγόριθμο (δηλαδή, το μέγεθος της βάσης γνώσης σε ομάδες) από το σύνολο των κινητών προτύπων (δηλαδή, το σύνολο της βάσης δεδομένων σε πρότυπα), το οποίο είναι: $a = |C|/|U|$, και $|C|$ είναι η πληθικότητα του συνόλου C . Η σταθμισμένη τιμή $w \in [0, 1]$ υποδεικνύει την σημαντικότητα των απαιτήσεων ακρίβειας και μνήμης, μία τιμή του $w = 0.5$ αναθέτει ίση σημαντικότητα στα a και p . Στην δική μας ανάθεση, θέτουμε $w = 0.7$. Απαιτούμε ότι το a υποθέτει χαμηλές τιμές ελαχιστοποιώντας το κόστος αποθήκευσης του αλγορίθμου. Μία χαμηλή τιμή του a υποδεικνύει ότι ο συγκεκριμένος αλγόριθμος υιοθετεί και μαθαίνει κατάλληλα τις κινήσεις των χρηστών χωρίς να διατηρεί περιττές πληροφορίες. Η τιμή του γ υποδεικνύει ποιες τιμές των ρ_s και ρ_t μεγιστοποιούν την ακρίβεια ενώ, παράλληλα, ελαχιστοποιούν τις απαιτήσεις σε μνήμη. Έτσι, η προσπάθεια να επιτύχουμε μία υψηλή τιμή για το γ υποδεικνύει έναν προσαρμοστικό αλγόριθμο με υψηλή τιμή ακρίβειας μαζί με χαμηλό κόστος αποθήκευσης. Όπως παρουσιάζεται στο Σχήμα 3.8, παρατηρούμε μία καθολική μέγιστη τιμή για το γ όταν το $\rho_s = 100 m$ και το $\rho_t = 10 min$ (τα οποία είναι και οι καθορισμένες τιμές κατά τη διάρκεια των πειραμάτων – βλέπε Πίνακα 3.3).

4. ΕΝΑ ΠΡΟΣΑΡΜΟΣΤΙΚΟ ΜΟΝΤΕΛΟ ΠΡΟΒΛΕΨΗΣ ΒΑΣΙΣΜΕΝΟ ΣΕ ΑΣΑΦΗ ΕΛΕΓΧΟ

4.1. Προσαρμοστική Βραχείας-μνήμης Πρόβλεψη Θέσης

Υιοθετούμε ένα τοπικό μοντέλο παλινδρόμησης το οποίο βασίζεται σε σταθμισμένες συναρτήσεις πυρήνα με σκοπό να καθορίσουμε τη μελλοντική θέση του χρήστη μέσω παρέκτασης (extrapolation). Ο LP εκμεταλλεύεται έναν ασαφή ελεγκτή με σκοπό να αποφασίσει ποιο είναι το κατάλληλο μέγεθος της μνήμης του τοπικού μοντέλου παλινδρόμησης που να ελαχιστοποιεί το σφάλμα πρόβλεψης. Αυτό σημαίνει ότι, ο ασαφής ελεγκτής προσαρμόζει το μέγεθος της μνήμης του μοντέλου παλινδρόμησης βασιζόμενος στην πρόσφατη συμπεριφορά της κίνησης του χρήστη. Οποιαδήποτε αλλαγή της κινητής συμπεριφοράς ανιχνευθεί διαχειρίζεται μέσω ενός σήματος ασαφούς ελέγχου το οποίο προσαρμόζει το τρέχον μέγεθος της μνήμης (ιστορικό παράθυρο) της διαδικασίας παλινδρόμησης. Το συνολικό μοντέλο του προτεινόμενου LP παρουσιάζεται στο Σχήμα 4.1. Συγκεκριμένα, το μοντέλο παλινδρόμησης συσσωρεύει τις τελευταίες $m \geq 1$ θέσεις και κατασκευάζει μία στατιστική συνάρτηση παλινδρόμησης f τη χρονική στιγμή t . Τότε, η μελλοντική θέση x , η οποία προβλέπεται για τις επόμενες $l > 1$ χρονικές στιγμές (δηλαδή, στον χρονικό δείκτη $t + l$), είναι ένα σημείο παρέκτασης το οποίο βασίζεται στη μέση τιμή της ταχύτητας και της κατεύθυνσης του χρήστη για τις τελευταίες m θέσεις. Το σφάλμα πρόβλεψης e τη χρονική στιγμή $t + l$ υπολογίζεται οποτεδήποτε η προβλεπόμενη θέση x δεν είναι η πραγματική θέση τη χρονική στιγμή $t + l$. Συγκεκριμένα το σφάλμα πρόβλεψης e είναι η χωρική απόσταση μεταξύ της προβλεπόμενης θέσης και της πραγματικής θέσης τη χρονική στιγμή $t + l$. Ο ασαφής ελεγκτής τροφοδοτείται με το σφάλμα πρόβλεψης e και προσαρμόζει το μήκος των m τελευταίων θέσεων στον LP.



Σχήμα 4.1. Ο προσαρμοστικός βραχείας-μνήμης Προγνώστης Θέσης (LP) μοντέλο και ο ασαφής ελεγκτής

4.1.1. Τοπικό Γραμμικό Μοντέλο Παλινδρόμησης

Στην παρούσα υποενότητα εστιάζουμε στη συνιστώσα της τοπικής παλινδρόμησης του LP όπως παρουσιάζεται στο Σχήμα 4.1. Η εκτιμώμενη συνάρτηση (μοντέλο) τοπικής παλινδρόμησης $f(X)$ σε ένα χώρο $X \subset \mathbb{R}^p$ ταιριάζει μία συνάρτηση f ξεχωριστά σε κάθε σημείο στόχο x_0 . Αυτό επιτυγχάνεται χρησιμοποιώντας μόνο εκείνες τις παρατηρήσεις (σημεία) κοντά στο x_0 για να ταιριάξουν στο μοντέλο με τέτοιο τρόπο ώστε το αποτέλεσμα να είναι μία εκτιμώμενη συνάρτηση \hat{f} η οποία να είναι ομαλή (smooth),

δηλαδή, να έχει παραγώγους όλων των τάξεων, στον χώρο R^p . Η τοπική γραμμική παλινδρόμηση γίνεται λιγότερο χρήσιμη για $p > 2$. Στην περίπτωση μας $p = 1$ όπως θα συζητήσουμε παρακάτω. Αυτός ο εντοπισμός βασίζεται σε μία σταθμισμένη συνάρτηση ή πυρήνας $K_\lambda(x_0, x)$. Ο πυρήνας αναθέτει βάρη στο x_i ανάλογα από την απόσταση του από το x_0 . Η τυπική παράμετρος ομαλοποίησης λ υποδεικνύει το πλάτος της γειτονιάς. Ένα μεγαλύτερο λ υποδηλώνει μικρότερη διακύμανση. Σε αυτές τις μεθόδους το μοντέλο δομείται από ένα σύνολο παρατηρήσεων και δεν απαιτεί εκπαίδευση, δηλαδή, όλη η εργασία πραγματοποιείται στο χρόνο της εκτίμησης θέσης. Η τοπικά σταθμισμένη παλινδρόμηση λύνει ένα ξεχωριστό σταθμισμένο πρόβλημα ελαχίστων τετραγώνων σε κάθε σημείο στόχο x_0 για ένα σύνολο από m παρατηρήσεις $(x_i, f(x_i) = y_i)$. Πρέπει να σημειωθεί ότι αν και ταιριάζουμε ένα ολόκληρο γραμμικό μοντέλο στα δεδομένα της περιοχής των m παρατηρήσεων, το χρησιμοποιούμε για να αξιολογήσουμε το ταίριασμα στο x_0 . Ας ορίσουμε τη συνάρτηση διανύσματος $b(x)^T = [1, x]$, τον πίνακα παλινδρόμησης \mathbf{B} (διαστάσεων $m \times 2$) με την i -th γραμμή να έχει τιμή $b(x_i)^T$ και τον διαγώνιο πίνακα $\mathbf{W}(x_0)$ $m \times m$ με το i -th διαγώνιο στοιχείο $K_\lambda(x_0, x_i)$, το οποίο είναι, $\mathbf{W}(x_0) = \text{diag}(K_\lambda(x_0, x_i))_{m \times m}$. Τότε, ισχύει ότι

$$\hat{f}(x_0) = b(x_0)^T (\mathbf{B}^T \mathbf{W}(x_0) \mathbf{B})^{-1} \mathbf{B}^T \mathbf{W}(x_0) \mathbf{y} \quad (4.1)$$

Όπου $\mathbf{y} = (f(x_1), f(x_2), \dots, f(x_m))^T$,

$$\mathbf{B}^T = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_m \end{bmatrix} \text{ και}$$

$$\mathbf{W}(x_0) = \begin{bmatrix} K_\lambda(x_0, x_1) & 0 & \dots & 0 \\ 0 & K_\lambda(x_0, x_2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & K_\lambda(x_0, x_m) \end{bmatrix}$$

Στην Εξίσωση 4.1 παρατηρούμε ότι η εκτίμηση της f είναι γραμμική στα y_i στοιχεία του \mathbf{y} και εξαρτάται από τις τιμές του πυρήνα $K_\lambda(x_0, x_i)$ του πίνακα $\mathbf{W}(x_0)$. Η παρέκταση $\hat{f}(x_0)$ δημιουργείται από τις m προηγούμενες παρατηρήσεις $f(x_i), i = 1, \dots, m$. Η συνάρτηση πυρήνα K_λ μπορεί να είναι για παράδειγμα, ο Epanechnikov τετραγωνικός πυρήνας, ο οποίος ορίζεται ως:

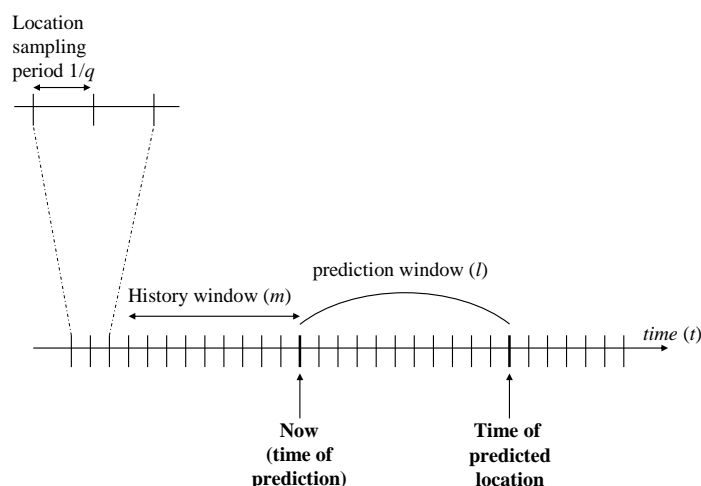
$$K_\lambda(x_0, x) = D\left(\frac{|x - x_0|}{\lambda}\right), D(u) = \begin{cases} \frac{3}{4}(1 - u^2), & |u| \leq 1 \\ 0, & \text{elsewhere} \end{cases}$$

Σύμφωνα με τη συνάρτηση πυρήνα $K_\lambda(x_0, x_i)$, όσο «μετακινούμε» το σημείο x_0 από τα αριστερά στα δεξιά, γειτονικά σημεία x_i μπαίνουν στο παράθυρο ιστορίας με μηδενικό βάρος, και τότε η συνεισφορά τους σιγά-σιγά αυξάνεται. Ένας άλλος διάσημος πυρήνας βασίζεται στην συνάρτηση tri-cube ή στην Gaussian συνάρτηση [10]. Το μοντέλο παλινδρόμησης χρησιμοποιείται για παρεμβολή μεταξύ της πληροφορίας θέσης της τροχιάς του χρήστη.

4.1.2. Μοντέλο Πληροφορίας Θέσης και Τροχιάς

Υιοθετούμε ένα μοντέλο πληροφορίας θέσης και τροχιάς, στο οποίο η θέση του χρήστη τη χρονική στιγμή t αναπαριστάται από ένα 2D διάνυσμα $(x(t), y(t))$ του γεωγραφικού μήκους $x(t)$ και του γεωγραφικού πλάτους $y(t)$. Η ακριβής πληροφορία της θέσης μπορεί εύκολα να εκτιμηθεί. Στην περίπτωση μας χρησιμοποιούμε πληροφορία θέσης

από GPS δέκτες ενσωματωμένους στο φορητό τερματικό του κινητού χρήστη. Βασισμένη στην παρούσα αναπαράσταση, η ιστορία της κίνησης του χρήστη έως το χρονικό δείκτη n παρουσιάζεται ως μία σειρά – τροχιά – από n σημεία $k(t) = (x(t), y(t)), t = 1, \dots, n$, το οποίο είναι $\mathbf{k} = [k(t)] = [(x(1), y(1)), \dots, (x(n), y(n))]$. Η ακολουθία \mathbf{k} για $t = 1, \dots, n$ είναι η τροχιά του χρήστη για τα τελευταία n σημεία ξεκινώντας από το σημείο στο χρονικό δείκτη $t = 1$. Κάθε σημείο $k(t)$ δειγματοληπτείται με συχνότητα q , δηλαδή, στα ίχνη μας, οι θέσεις GPS είναι δείγματα με $q = 1\text{Hz}$. Ένα παράθυρο \mathbf{w} είναι μία υπό-ακολουθία των πρώτων m σημείων από το \mathbf{k} , το οποίο είναι $\mathbf{w} = [k(t)], t = 1, \dots, m, m < n$. Η εκτίμηση-πρόβλεψη της μελλοντικής θέσης ενός κινητού χρήστη βασίζεται στο παράθυρο \mathbf{w} που έχει m σημεία. Με άλλα λόγια, ο προγνώστης τροφοδοτείται με μία τροχιά από m σημεία με σκοπό να προβλέψουμε τα μελλοντικά σημεία. Ο υπό συζήτηση προγνώστης αναφέρεται ως ένας $m + l$ μοντέλο πρόβλεψης [22][26]. Το οποίο σημαίνει ότι ο προγνώστης εκτιμά το μελλοντικό σημείο που έχει χρονικό δείκτη $m + l$, δηλαδή, προβλέπει το σημείο $k(m + l) = (x(m + l), y(m + l))$, με $l > 0$ από μία τροχιά (ιστορικό παράθυρο) με m σημεία (βλέπε επίσης Σχήμα 4.2). Για παράδειγμα, θέτουμε $l = 1$ για την πρόβλεψη του μελλοντικού σημείου στην επόμενη χρονική στιγμή. Το παράθυρο μεγέθους m , ο αριθμός των l χρονικών στιγμών για την πρόβλεψη του $k(l)$ σημείου και η συχνότητα της δειγματοληψίας q είναι οι βασικές παράμετροι στο δικό μας LP, όπως παρουσιάζεται στο Σχήμα 4.2.



Σχήμα 4.2. Οι βασικές παράμετροι για την $m+l$ πρόβλεψη.

4.1.3. Σχεδιασμός του Προγνώστη Θέσης

Σε αυτήν την ενότητα συζητάμε την υιοθέτηση της τοπικής γραμμικής παλινδρόμησης στην πρόβλεψη θέσης. Μπορούμε να διαχωρίσουμε τις ακόλουθες περιπτώσεις για την κίνηση του χρήστη για μία πολύ μικρή χρονική διάρκεια $\Delta t > 0$:

- Περίπτωση-1: ο χρήστης κινείται πιο σταθερά στην κατεύθυνση $x(t)$ του γεωγραφικού μήκους από ότι στην κατεύθυνση του γεωγραφικού πλάτους. Αυτό σημαίνει ότι, ισχύει ότι η διαφορά $|x(t + \Delta t) - x(t)| \ll |y(t + \Delta t) - y(t)|$ σε μία Δt .
- Περίπτωση-2: ο χρήστης κινείται πιο σταθερά στην κατεύθυνση $y(t)$ του γεωγραφικού πλάτους από ότι στην κατεύθυνση του γεωγραφικού μήκους, το οποίο σημαίνει ότι, $|y(t + \Delta t) - y(t)| \ll |x(t + \Delta t) - x(t)|$, και,
- Περίπτωση-3: ο χρήστης κινείται κατά τέτοιο τρόπο έτσι ώστε $|x(t + \Delta t) - x(t)| = |y(t + \Delta t) - y(t)|$ σε μία Δt .

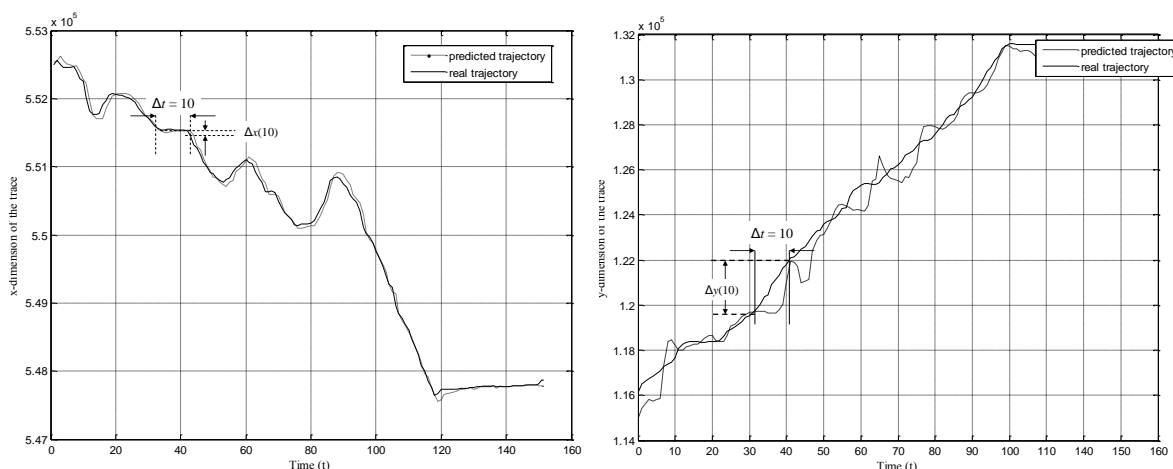
Θεωρείστε ότι για το χρονικό πλαίσιο m , υπολογίζουμε τις μέσες διαφορές και τις δύο κατευθύνσεις γεωγραφικού μήκους και γεωγραφικού πλάτους. Το οποίο σημαίνει, ότι θεωρούμε τα διανύσματα:

$$\begin{aligned}\Delta x(m) &= \frac{1}{m-1} \sum_{i=2}^m (x(i) - x(i-1)) = \frac{1}{m-1} (x(m) - x(1)) \\ \Delta y(m) &= \frac{1}{m-1} \sum_{i=2}^m (y(i) - y(i-1)) = \frac{1}{m-1} (y(m) - y(1))\end{aligned}\quad (4.2)$$

Ο προγνώστης μπορεί να εκτιμήσει το σημείο μετά από l χρονικές στιγμές όταν οι επόμενες περιπτώσεις ισχύουν:

- Αν $|\Delta x(m)| < |\Delta y(m)|$ τότε $k(m+l) = (x(m+l), y(m+l))$ με την παρέκταση $x(m+l) = x(m) + l \cdot \Delta x(m)$ και την εκτίμηση της τοπικής παλινδρόμησης $y(m+l) = \hat{f}(x(m+l))$. Σε αυτήν την περίπτωση, το $y(m+l)$ δημιουργείται από το παράθυρο w και από την τοπική γραμμική εκτίμηση $\hat{f}(x(m+l))$ στο $x(m+l)$ έχοντας τοπική παλινδρόμηση στον μονοδιάστατο (1-dimension) χώρο ($p = 1$),
- Αν $|\Delta x(m)| > |\Delta y(m)|$ τότε $k(m+l) = (x(m+l), y(m+l))$ με την παρέκταση $y(m+l) = y(m) + l \cdot \Delta y(m)$ και την εκτίμηση της τοπικής παλινδρόμησης $x(m+l) = \hat{f}(y(m+l))$. Σε αυτήν την περίπτωση, το $x(m+l)$ δημιουργείται από το παράθυρο w και από την τοπική γραμμική εκτίμηση $\hat{f}(y(m+l))$ στο $y(m+l)$.

Αξίζει να μελετήσουμε την περίπτωση όπου $|\Delta x(m)| = |\Delta y(m)|$. Συγκεκριμένα, το χρονικό παράθυρο m που χρησιμοποιείται για τον υπολογισμό του $|\Delta x(m)|$ ή του $|\Delta y(m)|$ μπορεί να περιέχει μικρότερες χωρικές μεταβάσεις που δεν μπορούν να συλληφθούν από τις απόλυτες τιμές της μετατόπισης στους x ή y άξονες. Ένα μικρό χρονικό παράθυρο m καθιστά το μοντέλο πιο ευαίσθητο σε αυτές τις μικρές μετατοπίσεις ενώ ένα μεγάλο χρονικό παράθυρο ίσως «αφομοιώσει» τις κινήσεις και παράγει προβλέψεις που δεν είναι αντιπροσωπευτικές στην πραγματική κίνηση (όχι απαραίτητα εσφαλμένες προβλέψεις). Βεβαίως, λόγω των παραπάνω ελλείψεων, αν $|\Delta x(m)| = |\Delta y(m)| = 0$, ο προγνώστης αποτυγχάνει να διαχωρίσει μεταξύ τις εντελώς στατικής περίπτωσης και μίας επαναληπτικής (κυκλικής) διαδρομής. Έτσι, σε αυτήν την περίπτωση, το μελλοντικό σημείο $k(m+l)$ είναι $x(m+l), y(m+l)$ με την παρέκταση είτε $y(m+l) = y(m) + l \cdot \Delta y(m)$ (και $x(m+l) = \hat{f}(y(m+l))$) ή $x(m+l) = x(m) + l \cdot \Delta x(m)$ (και $y(m+l) = \hat{f}(x(m+l))$).



Σχήμα 4.3. Μέρος της τροχιάς για πραγματικό GPS ίχνος (Ιταλία). (α) x -διάσταση της τροχιάς σε σχέση τον χρόνο, (β) y - διάσταση της τροχιάς σε σχέση τον χρόνο; συχνότητα δειγματοληψίας is 1Hz.

Με σκοπό να κρίνουμε εάν ο προγνώστης είναι επιτυχής στην πρόβλεψη του μελλοντικού σημείου $k(m+l) = (x(m+l), y(m+l))$, εισάγουμε ένα κατώφλι σφάλματος $\theta > 0$. Ένα τέτοιο κατώφλι είναι η ελάχιστη απόσταση από το πραγματικό σημείο $z(m+l)$ τη χρονική στιγμή $m+l$ και του προβλεπόμενου σημείου $k(m+l)$. Το $z(m+l)$ σημείο είναι η πραγματική (παρατηρούμενη) θέση του κινητού χρήστη τη χρονική στιγμή $m+l$. Έτσι, η πρόβλεψη θεωρείται επιτυχής αν $\|z(m+l) - k(m+l)\| \leq \theta$. Αλλιώς, η πρόβλεψη αποτυγχάνει.

Το Σχήμα 4.3 απεικονίζει τις x και y διαστάσεις μίας τροχιάς πάνω στο χρόνο. Στο Σχήμα 4.3 για $31 < t < 41$ μπορούμε να παρατηρήσουμε ότι η αλλαγή στην x διάσταση είναι μικρότερη από την αντίστοιχη αλλαγή στην y διάσταση $|\Delta x(m)| < |\Delta y(m)|$ στο Σχήμα 4.3(α) και στο Σχήμα 4.3(β) για την χρονική περίοδο $\Delta t = 31 - 41 = 10$ δευτερόλεπτα. Μπορούμε να προβλέψουμε τη θέση του μελλοντικού σημείου την χρονική στιγμή $m+l, l > 0$ με $x(m+l) = x(m) + l \cdot \Delta x(m)$ και $y(m+l)$ να εκτιμείται από το $\hat{f}(x(m+l))$. Οι Εικόνες 3(a) και 3(b) απεικονίζουν επίσης την προβλεπόμενη τροχιά για $l = 5$ δευτερόλεπτα.

Αν χρησιμοποιήσουμε τον $m+l$ προγνώστη για N φορές (για μεγάλο N) τότε μπορούμε να εκτιμήσουμε την *πιθανότητα μίας επιτυχούς $m+l$ πρόβλεψης* $P(m,l)$. Το $P(m,l)$ είναι το κλάσμα των ορθά προβλεπόμενων θέσεων, δηλαδή, αυτά τα προβλεπόμενα σημεία $k(m+l)$ που βρίσκονται εντός του κύκλου με ακτίνα θ , προς το σύνολο των N επικλήσεων του προγνώστη, δηλαδή,

$$P(m,l) = \frac{|\{k(m+l) : \|z(m+l) - k(m+l)\| \leq \theta\}|}{N}$$

Όπου $z(m+l)$ είναι η πραγματική (παρατηρούμενη) θέση του κινητού χρήστη κατά τη χρονική στιγμή $m+l$ και $|U|$ είναι η πληθικότητα του συνόλου U . Το Σχήμα 4.3 παρουσιάζει την πραγματική και την προβλεπόμενη τροχιά ενός κινητού χρήστη με $m = 10$ και $l = 5$ (ένα «10+5» μοντέλο πρόβλεψης).

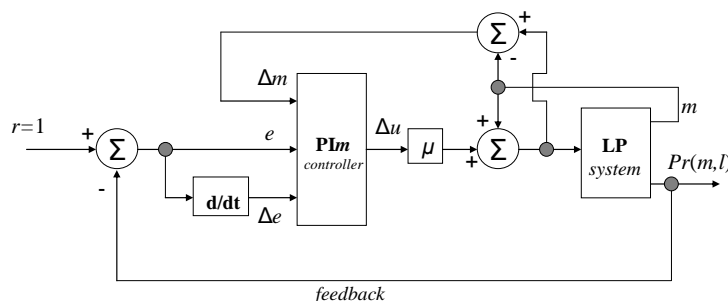
4.2. Ασαφής Έλεγχος για Προσαρμοστική Πρόβλεψη Θέσης

Ο ασαφής έλεγχος είναι μία άμεση μη γραμμική απεικόνιση μεταξύ των εισόδων (δηλαδή, σφάλμα πρόβλεψης θέσης, αλλαγή του σφάλματος πρόβλεψης) και των εξόδων (δηλαδή, αλλαγή στο μέγεθος της μνήμης m)[13]. Ο ασαφής έλεγχος παρέχει τυπικές τεχνικές για την αναπαράσταση και την υλοποίηση ευρεστικών λύσεων των ανθρώπων ειδικών για τον έλεγχο ενός συστήματος. Τέτοιες τεχνικές βασίζονται σε ορισμένους *if – then* κανόνες αντί να βασίζονται σε μία μαθηματική μοντελοποίηση του συστήματος. Οι ελεγκτές ασαφούς λογικής χρησιμοποιούνται ευρέως, συγκεκριμένα, στη μορφή των αναλογικών, ολοκληρωματικών και παραγώγων (PID), PD, και PI ελεγκτών [27]. Σε αυτήν την ενότητα θεωρούμε έναν ασαφή, αναλογικό-ολοκληρωματικό ελεγκτή PI_m για να ρυθμίσουμε το μέγεθος του παραθύρου m στην πρόβλεψη θέσης. Το σήμα ελέγχου βασίζεται σε ένα σύνολο από ασαφείς κανόνες οι οποίοι δημιουργούν την αναμενόμενη έξοδο [28].

Το Σχήμα 4.4 απεικονίζει την υψηλού επιπέδου δομή του προτεινόμενου κλειστού-βρόγχου, πραγματικού-χρόνου συστήματος για να ρυθμίσει το ιστορικό παράθυρο μήκους m . Για να επιτύχουμε την μέγιστη τιμή της πιθανότητας των επιτυχών προβλέψεων $P(m(t), l)$ τη χρονική στιγμή t , η οποία συμβολίζεται από το μοντέλο αναφοράς $\alpha(t) = 1$ τη χρονική στιγμή t (το πάνω όριο της πιθανότητας των επιτυχών

προβλέψεων), ο PI_m ελεγκτής υπολογίζει την απαιτούμενη τιμή του m , που συμβολίζεται ως $u(t)$, βασιζόμενο στο σφάλμα $e(t)$, αλλαγή του σφάλματος $\Delta e(t)$ και την τελευταία απόφαση του ελεγκτή κατά την αλλαγή του m , $\Delta m(t)$. Συγκεκριμένα, το σφάλμα $e(t)$ την χρονική στιγμή t είναι:

$$e(t) = a(t) - P(m(t), l) \quad (4.3)$$



Σχήμα 4.4. Η ροή ελέγχου και η ανατροφοδότηση του ασαφούς ελεγκτή PI_m στον προγνώστη θέσης LP

Στην πρόβλεψη θέσης, στοχεύουμε στο να μεγιστοποιήσουμε την $P(m(t), l)$ σε κάθε δοσμένο t το οποίο είναι, $\alpha(t) = 1$ σε κάθε πρόβλεψη. Η αλλαγή του σφάλματος τη χρονική στιγμή t είναι:

$$\Delta e(t) = e(t) - e(t-1) \quad (4.4)$$

Και η αλλαγή της προηγούμενης απόφασης του ελεγκτή στο m είναι:

$$\Delta m(t) = m(t) - m(t-1) \quad (4.5)$$

Η ακριβής τιμή του u η οποία είναι το μήκος του παραθύρου που ελαχιστοποιεί το σφάλμα δεν μπορεί να εκτιμηθεί κατευθείαν. Αντιθέτως, ο PI_m προσαρμόζει το $m(t)$ με σκοπό να ελαχιστοποιήσει το $e(t)$ για κάθε χρονική στιγμή t . Επιπρόσθετα, πρέπει να σημειωθεί ότι η τιμή m που ελαχιστοποιεί το $e(t)$ (συμβολίζεται με m^*) δεν είναι σταθερή αφού ο κινητός χρήστης μπορεί αυθόρμητα να αλλάξει την συμπεριφορά της κίνησης του. Σε αυτή την περίπτωση, ο ελεγκτής έχει να προσαρμοστεί σε αυτή την αλλαγή και να προσαρμόσει το m έτσι ώστε να ελαχιστοποιήσει το $e(t)$ μέχρι την επόμενη κίνηση της συμπεριφοράς του χρήστη.

Ορίζουμε την απαιτούμενη αλλαγή σε μία τιμή του m , $\Delta u(t) \in [-1, 1]$, η οποία δηλώνει την σημανόμενη ποσότητα της αλλαγής της τιμής του $m(t)$ - αύξηση, μείωση ή να παραμείνει σταθερή - και έχει ως αποτέλεσμα την τιμή $m(t+1)$ στην επόμενη χρονική στιγμή, $t+1$. Με άλλα λόγια παρατηρούμε τον ακόλουθο κανόνα προσαρμογής:

$$m(t+1) = m(t) + \mu(\Delta u(t)m(t)) \quad (4.6)$$

με $\mu > 0$ έναν βαθμό προσαρμογής (εκπαίδευσης). Ο προτεινόμενος ασαφής ελεγκτής υπολογίζει την τιμή του σήματος ελέγχου $\Delta u(t) = u(t) - u(t-1)$. Η οποία είναι, η απόφαση για επόμενη τιμή του m (που ελαχιστοποιεί το σφάλμα) που βασίζεται στην προηγούμενη απόφαση για το m μαζί με το παρατηρούμενο σφάλμα που προκαλείται από τον LP. Η αλλαγή στο m , Δm , και η αλλαγή στο σφάλμα, Δe , είναι τα βασικά συστατικά του ελεγκτή μας τα οποία, μαζί με την τιμή του σφάλματος, καθορίζουν την κατεύθυνση (θετική, μηδενική, αρνητική) του Δu κατά την απόφαση της επόμενης τιμής του m , όπως θα περιγράψουμε στη συνέχεια.

Η αναδρομική εξίσωση ενός διακριτού PID ελεγκτή είναι:

$$u(t) = G_p e(t) + G_I \sum_{\tau=0}^t e(\tau) + G_D \Delta e(t)$$

Όπου G_P, G_I, G_D είναι τα αντίστοιχα σταθερά αναλογικά, ολοκληρωτικά και παραγωγικά κέρδη. Ο ορισμός της διαφοράς μεταξύ δύο διαδοχικών εξόδων ελέγχου $\Delta u(t)$ είναι:

$$\Delta u(t) = u(t) - u(t-1) = G_I e(t) + G_P \Delta e(t) + G_D \Delta^2 e(t) \quad (4.7)$$

Όπου $\Delta^2 e(t) = \Delta e(t) - \Delta e(t-1)$. Στην περίπτωση μας, ο PI_m έχει τρεις εισόδους - $e(t), \Delta e(t)$, και $\Delta m(t)$ - και μία έξοδο - την τρέχουσα αλλαγή στο m , $\Delta u(t)$. Ο δικός μας PI_m ελεγκτής έχει την ακόλουθη φόρμα:

$$u(t) = G_P e(t) + G_I \sum_{\tau=0}^t e(\tau) + G_D (\Delta e(t) + \Delta m(t)) \quad (4.8)$$

Η επαναληπτική φόρμα του $u(t)$ παρατηρείται παίρνοντας την παράγωγο και των δύο πλευρών της Εξίσωσης (4.8) έχοντας επίσης $\Delta m(t) = m(t) - m(t-1)$, έτσι, στην περίπτωση μας,

$$\Delta u(t) = G_P \Delta e(t) + G_I e(t) + G_D (\Delta^2 e(t) + \Delta^2 m(t)) \quad (4.9)$$

Και τότε, $u(t) = u(t-1) + G_I e(t) + G_P \Delta e(t) + G_D (\Delta^2 e(t) + \Delta^2 m(t))$ με $\Delta^2 m(t) = \Delta m(t) - \Delta m(t-1)$. Βασιζόμενοι στην Εξίσωση (4.9), οι εισοδοί του δικού μας ελεγκτή ($e, \Delta e$ και Δm) και η έξοδος του ($\Delta u(t)$) κάνουν τον ελεγκτή μας έναν PI ελεγκτή. Το Σχήμα 4.4 απεικονίζει την αρχιτεκτονική του προτεινόμενου ελεγκτή.

4.2.1. Ασαφής Ελεγκτής

Σε αυτήν την υποενότητα, παρουσιάζουμε το βασικό σύστημα ασαφούς ελέγχου για να αποφανθούμε σχετικά με το απαιτούμενο σήμα που βασίζεται σε ασαφείς κανόνες απόφασης. Ένας ασαφής ελεγκτής είναι ένα σύστημα ασαφούς λογικής με n εισόδους και k εξόδους. Στην περίπτωση μας, ο PI_m ελεγκτής είναι ένα Πολλαπλής Εισόδου Απλής Εξόδου (MISO) σύστημα ασαφούς λογικής με $n = 3$ και $k = 1$, τέτοιο ώστε η είσοδος κατά τη χρονική στιγμή t είναι $\mathbf{p}(t) = [e(t), \Delta e(t), \Delta m(t)]$ και η έξοδος είναι $\mathbf{q}(t) = [\Delta u(t)]$. Το ασαφές σύστημα αποτελείται από τις διεργασίες που φαίνονται στο Σχήμα 4.5:

- Ασαφοποίηση (fuzzification),
- Διεργασία ασαφούς απόφασης, και
- Αποασαφοποίηση (defuzzification).

Η διαδικασία ασαφοποίησης μετατρέπει τα σαφή συστατικά του $\mathbf{p}(t)$ σε ασαφή σύνολα, η διαδικασία απόφασης χρησιμοποιεί ασαφείς κανόνες απόφασης από μία ασαφή βάση-κανόνων με σκοπό να παράγει ασαφή συμπεράσματα, και η διαδικασία αποασαφοποίησης μετατρέπει τα ασαφή συμπεράσματα σε σαφή έξοδο $\mathbf{q}(t)$.

4.2.1.1. Σύμπαν του Λόγου (Universe of Discourse)

Το πεδίο ορισμού του κάθε συστατικού εισόδου στο \mathbf{p} και του συστατικού εξόδου στο \mathbf{q} του ασαφούς ελεγκτή καλείται *σύμπαν του λόγου*. Το σύμπαν του λόγου για το σφάλμα e είναι $[0, 1]$, αφού $e(t) = 0$ αν $P(m(t), l) = 1$ και το μοντέλο αναφοράς είναι $\alpha(t) = 1$ (ο προγνώστης έχει να επιτύχει την μέγιστη πιθανότητα). Από την άλλη πλευρά, $e(t) = 1$ αν $P(m(t), l) = 0$. Προφανώς, το σύμπαν του λόγου για την αλλαγή στο σφάλμα $\Delta e(t)$ είναι $[-1, 1]$. Το σύμπαν του λόγου για το $\Delta u(t)$ είναι $[-1, 1]$ και δηλώνει την ποσοστιαία αναλογία της αύξησης ή της μείωσης της τιμής του m . Τελικά, η προηγούμενη απόφαση

της αλλαγής των m , Δm , είναι θετική, μηδέν ή αρνητική. Αν ορίσουμε μία επί τοις εκατό αναλογία στο Δm , τότε το αντίστοιχο σύμπταν του λόγου είναι [-1, 1].

4.2.1.2. Γλωσσικές Μεταβλητές και Τιμές

Ένα ασαφές σύστημα υιοθετεί *γλωσσικές μεταβλητές* από τη Θεωρία Ασαφών Συνόλων [18] με σκοπό να περιγράψει τις εισόδους του και την έξοδο του. Οι τρεις εισοδοί στον PI_m κατά τη χρονική στιγμή t περιγράφονται ως η μεταβλητή *σφάλμα* (ασαφές $e(t)$), *μεταβολή του σφάλματος* (ασαφές $\Delta e(t)$), και *μεταβολή του m* (ασαφές $\Delta m(t)$). Η έξοδος από τον ελεγκτή ορίζεται ως η ασαφής μεταβλητή *σήμα ελέγχου* $\Delta u(t)$, δηλαδή, η επί τοις εκατό μεταβολή στο m που απαιτείται για να επιτευχθεί η τιμή αναφοράς του 1 (ελαχιστοποιώντας το $e(t)$) εξεφρασμένη γλωσσικά. Η διαδικασία ασαφοποίησης μετατρέπει τις σαφείς εισόδους $e(t)$, $\Delta e(t)$ και $\Delta m(t)$ στις αντίστοιχες γλωσσικές τιμές.

Οι γλωσσικές μεταβλητές σχετίζονται με *γλωσσικές τιμές* οι οποίες περιγράφουν τα χαρακτηριστικά των μεταβλητών. Η γλωσσική μεταβλητή *σφάλμα* (e) σχετίζεται με τις γλωσσικές τιμές *μικρό* (small) και *μεγάλο* (high). Η γλωσσική μεταβλητή *μεταβολή του σφάλματος* (Δe), *μεταβολή του m* (Δm) και *σήμα ελέγχου* (Δu) έχουν γλωσσικές τιμές *αρνητικά μεγάλο* (nlarge), *αρνητικά μικρό* (nsmall), *μηδέν* (zero), *θετικά μικρό* (psmall), *θετικά μεγάλο* (plarge). Μία γλωσσική τιμή αναπαριστάται από ένα ασαφές σύνολο.

4.2.1.3. Ασαφή Σύνολα και Ασαφοποίηση

Ένα ασαφές σύνολο A είναι το σύνολο $\{(a, \mu_A(a)), a \in \mathfrak{R}, \mu_A(a) \in [0, 1]\}$ όπου $\mu_A(a)$ υποδηλώνει τον *βαθμό συμμετοχής* του a στο A . Για παράδειγμα, η γλωσσική μεταβλητή *σφάλμα* έχει δύο γλωσσικές τιμές: *small* και *high*. Η γλωσσική τιμή *small* είναι το ασαφές σύνολο $small = \{(e(t), \mu_{small}(e(t))), e(t) \in [0, 1], \mu_{small}(e(t)) \in [0, 1]\}$ με συνάρτηση συμμετοχής $\mu_{small}(e(t))$. Αυτή η συνάρτηση συμμετοχής ποσοτικοποιεί τη συμμετοχή της τιμής $e(t)$ στο ασαφές σύνολο *small*, δηλαδή, περιγράφει σε ποιο βαθμό μία σαφής τιμή του $e(t)$ εκφράζεται ως μικρό (*small*) σφάλμα. Η συνάρτηση $\mu_A(a)$ μπορεί να είναι μία συμμετρική τριγωνική ή τραπεζοειδής συνάρτηση μίας ίσης βάσης και 50% επικάλυψης με τις γειτονικές συναρτήσεις (επίσης παρουσιάζεται στο Σχήμα 4.5) η οποία είναι,

$$\mu_A(a) = \begin{cases} 0, & a < l \\ \frac{a-l}{b-l}, & l \leq a < b \\ 1, & b \leq a < c \\ \frac{h-a}{h-c}, & c \leq a < h \end{cases}$$

Με $l \leq b \leq c \leq h$. Για παράδειγμα, αν $e(t) = 0.4$ τότε $\mu_{small}(0.4) = 0.65$ και $\mu_{high}(0.4) = 0.35$. Έτσι, το σαφές σφάλμα $e(t) = 0.4$ αναπαριστάται επίσης από το ασαφές σύνολο $\{(0.4, \mu_{small}(0.4)), (0.4, \mu_{high}(0.4))\} = \{(0.4, 0.65), (0.4, 0.35)\}$. Αυτό είναι το αποτέλεσμα ασαφοποίησης της σαφούς εισόδου $e(t) = 0.4$. Όλες οι σαφείς εισοδοί ασαφοποιούνται με όμοιο τρόπο. Η διαδικασία ασαφοποίησης ασαφοποιεί το διάνυσμα εισόδου \mathbf{p} , το οποίο περνάει μέσα στη μηχανή ασαφούς απόφασης.

4.2.1.4 Βάση-Κανόνων Ασαφούς Απόφασης

Η μηχανή ασαφούς απόφασης αποτελείται από ένα σύνολο από κανόνες IF *υπόθεση* THEN *αποτέλεσμα* που απεικονίζουν την είσοδο \mathbf{p} στην έξοδο \mathbf{q} του PI_m . Οι κανόνες

IF-THEN (κανόνες ελέγχου) καθορίζουν την τιμή του Δu σύμφωνα με την ασαφοποιημένη είσοδο \mathbf{p} των $e(t)$, $\Delta e(t)$, και $\Delta m(t)$. Ένας ασαφής κανόνας R_i είναι της εξής μορφής:

$$R_i: \text{IF (σφάλμα} = A_i^j) \text{ AND (μεταβολή στο σφάλμα} = A_i^k) \text{ AND (μεταβολή στο } m = A_i^l) \text{ THEN (σήμα ελέγχου} = A_i^n)$$

Όπου A_i^j είναι η j – ιοστή γλωσσική τιμή για το *σφάλμα*, A_i^k είναι η k – ιοστή γλωσσική τιμή για τη *μεταβολή στο σφάλμα*, A_i^l είναι η l – ιοστή γλωσσική τιμή για τη *μεταβολή στο m* και A_i^n είναι η n – ιοστή γλωσσική τιμή για το *σήμα ελέγχου*. Ο μηχανισμός απόφασης αξιολογεί όλους τους R_i κανόνες και αποφασίζει ποιο θα έπρεπε να είναι το ασαφές σήμα ελέγχου κοιτάζοντας τον πίνακα της βάσης-κανόνων (επόμενη ενότητα) χρησιμοποιώντας τις ασαφοποιημένες τιμές εισόδου. Για παράδειγμα, αν ισχύει ότι το *σφάλμα* = μικρό και η *μεταβολή στο σφάλμα* = αρνητικά μεγάλη και η *μεταβολή στο m* = θετικά μικρή κατά τη χρονική στιγμή t , τότε ο ελεγκτής αποφαινεται ότι η προηγούμενη απόφαση για την μεταβολή της τιμής του m (που είναι μία μικρή αύξηση στο m - θετικά μικρή) έχει ως αποτέλεσμα την βελτίωση της πιθανότητας πρόβλεψης του LP (Δe = αρνητικά μεγάλη). Το οποίο σημαίνει ότι, η απόφαση πάνω στο $\Delta m(t) > 0$ έχει σαν αποτέλεσμα $P(m(t), l) > P(m(t-1), l)$. Έτσι, ο ελεγκτής επιβραβεύει μία τέτοια απόφαση με το να αυξήσει ελαφρά την επόμενη τιμή του m . Αυτό επιτυγχάνεται με το να δημιουργηθεί ένα θετικά μικρό σήμα ελέγχου $\Delta u(t)$. Ο αντίστοιχος ασαφής κανόνας αυτού του παραδείγματος είναι τότε:

$$\text{IF (σφάλμα} = \text{small}) \text{ AND (μεταβολή στο σφάλμα} = \text{nlarge}) \text{ AND (μεταβολή στο } m = \text{psmall}) \text{ THEN (σήμα ελέγχου} = \text{psmall})$$

Για να αποφύγουμε πιθανές ταλαντώσεις κοντά στο m^* , για υψηλά (ή ακόμα και χαμηλά) σφάλματα πρόβλεψης και υψηλές μεταβολές στο Δm , ο ελεγκτής δημιουργεί σχετικά μικρά σήματα ελέγχου αντί για επιθετικά (μεγάλα).

4.2.1.5. Αποασαφοποίηση

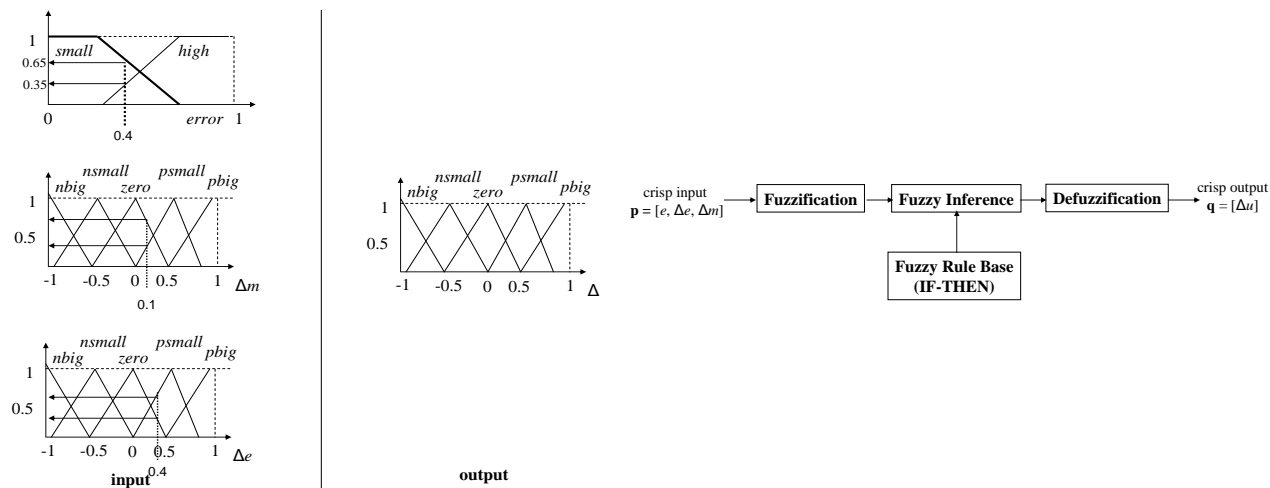
Ο μηχανισμός αποασαφοποίησης έχει ως αποτέλεσμα την έξοδο \mathbf{q} , δηλαδή το σήμα ελέγχου $\Delta u(t)$ κατά την χρονική στιγμή t καθορίζεται από τον μηχανισμό απόφασης σαν ένας πραγματικός αριθμός στο $[-1, 1]$. Ως εκ τούτου, το ασαφές σύστημα ελέγχου με M ασαφείς κανόνες $R_i, i = 1, \dots, M$, δίνεται από

$$\mathbf{q} = \frac{\sum_{i=1}^M c_i \mu_i(\mathbf{p})}{\sum_{i=1}^M \mu_i(\mathbf{p})}$$

Όπου c_i είναι τα κέντρα των γλωσσικών τιμών της τιμής εξόδου Δu (οι τιμές $\{-1, -0.5, 0, 0.5, 1\}$ – βλέπε τις συναρτήσεις συμμετοχής του Δu στο Σχήμα 4.5). Και $\mu_i(\mathbf{p}) = \mu_i([e, \Delta e, \Delta m])$ είναι η *βεβαιότητα* της προϋπόθεσης του R_i κανόνα που καθορίζεται από τις συναρτήσεις συμμετοχής στην είσοδο \mathbf{p} το οποίο είναι, παίρνοντας το γινόμενο

$$\mu_i(\mathbf{p}) = \mu_i([e, \Delta e, \Delta m]) = \mu_{A_i^j}(e) \cdot \mu_{A_i^k}(\Delta e) \cdot \mu_{A_i^l}(\Delta m) \quad (4.10)$$

Όπου $\mu_{A_i^x}(x)$ είναι ο βαθμός συμμετοχής της γλωσσικής τιμής για την x είσοδο στον i – οστό ασαφή κανόνα απόφασης. Για να συνοψίσουμε, ο *Plm* ελεγκτής χρησιμοποιεί μονήρης ασαφοποίηση, γινόμενο για την προϋπόθεση $\mu_i(\mathbf{p})$ και τον συμπερασμό του κάθε κανόνα R_i , και αποασαφοποίηση κεντρικού μέσου όρου. Το αποασαφοποιημένο σήμα ελέγχου προσαρμόζει την τιμή του m για την επόμενη διαδικασία πρόβλεψης.



Σχήμα 4.5. Οι συναρτήσεις συμμετοχής των εισόδων (για ασαφοποίηση) και της εξόδου (για αποασαφοποίηση) του PIm και τα βασικά συστατικά του συστήματος ασαφούς λογικής

4.2.2. Ζώνες Ασαφούς Ελέγχου για την Πρόβλεψη της Θέσης

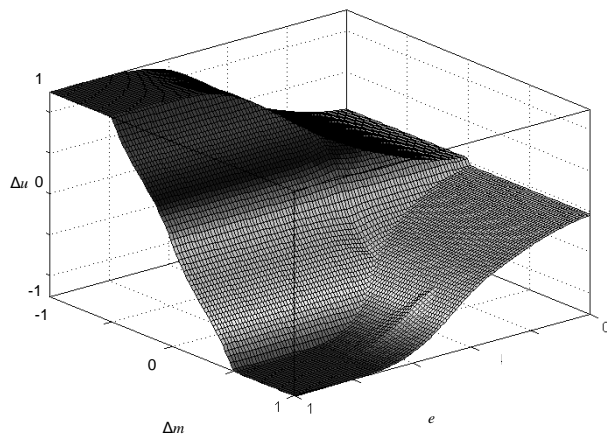
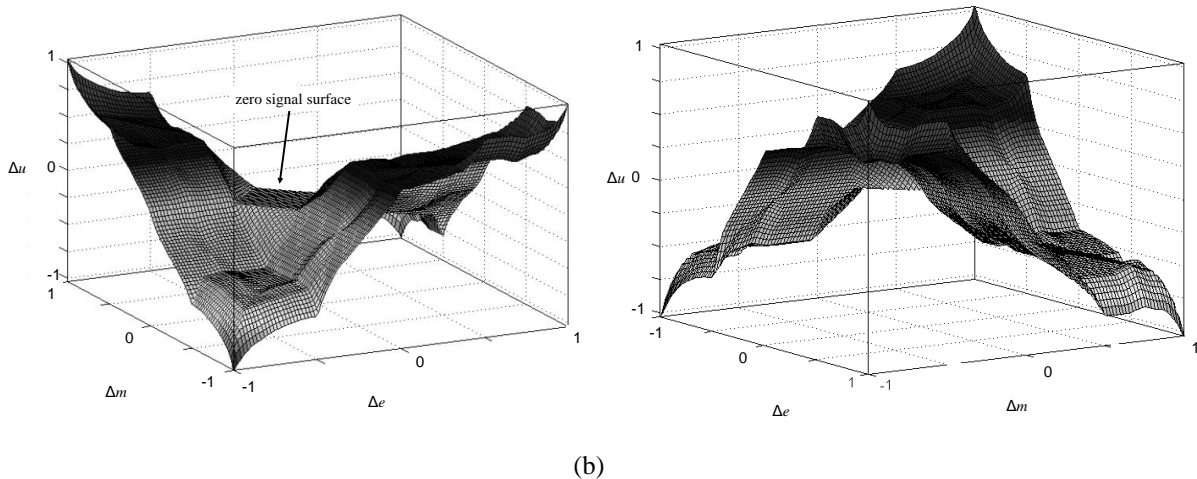
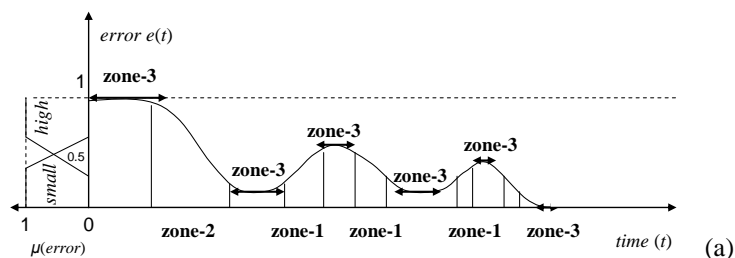
Σε αυτήν την ενότητα περιγράφουμε την δημιουργία των ασαφών κανόνων απόφασης για το σύστημα ασαφούς ελέγχου. Για να ελέγξουμε τον LP θέτοντας ένα παράθυρο μήκους m το οποίο ελαχιστοποιεί το σφάλμα πρόβλεψης, ξεχωρίζουμε τρεις βασικές ζώνες που χαρακτηρίζουν το σήμα ελέγχου. Το Σχήμα 4.6(α) απεικονίζει τις υπό συζήτηση ζώνες που χαρακτηρίζουν τις ενέργειες του ελεγκτή. Συγκεκριμένα, η βασική είσοδος για τον PI m ελεγκτή είναι το σφάλμα e . Η τιμή του σφάλματος μαζί με την απόφαση για την τιμή του m καθορίζει το σήμα ελέγχου για τον ελεγκτή. Από αυτές τις ζώνες παράγουμε την ασαφή βάση-κανόνων. Αναφερόμαστε στο $|\Delta u|$ σαν την σαφή τιμή του σήματος ελέγχου και στο πρόσημο του σήματος ελέγχου σαν το πρόσημο του Δu , δηλαδή, θετικό ή αρνητικό. Βασιζόμενοι στην τιμή του σφάλματος, το σήμα ελέγχου εκφράζεται είτε ως μία *επιβράβευση* ή ως μία *ποινή* για την προηγούμενη απόφαση για το m . Η κατάσταση σταθερότητας του ελεγκτή σηματοδοτείται όταν δεν υπάρχει καμία αλλαγή στο σφάλμα και το σφάλμα έχει πολύ μικρή τιμή. Το Σχήμα 4.6(β) και το Σχήμα 4.6(γ) δείχνουν το αποτέλεσμα της *επιφάνειας ελέγχου* που παρατηρείται κατά τη σχεδίαση του αποφαινόμενου σήματος ελέγχου Δu για διακριτές τιμές των $(\Delta m, \Delta e)$ και $(\Delta m, e)$, αντίστοιχα, τα οποία θα εξηγηθούν παρακάτω σε αυτήν την ενότητα.

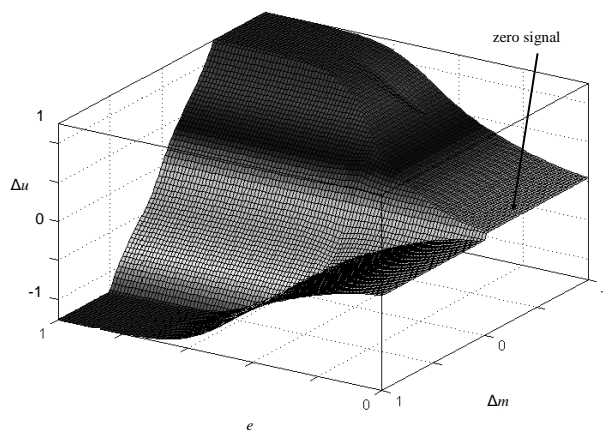
Ζώνη Ελέγχου – 1: Η ζώνη-1 ασχολείται με τον προσδιορισμό του Δu όταν το σφάλμα είναι *μικρό*, το οποίο συμβαίνει, όταν ο LP έχει ως αποτέλεσμα επιτυχείς προβλέψεις τις περισσότερες φορές. Σε αυτήν την ζώνη μπορούμε να διαχωρίσουμε τις εξής περιπτώσεις:

Περίπτωση 1.1: Το $e(t)$ είναι *μικρό* και το $\Delta e(t)$ είναι αρνητικά μικρό ή αρνητικά μεγάλο κατά τη χρονική στιγμή t .

Σε αυτήν την περίπτωση, ο ελεγκτής θα πρέπει να επιβραβεύσει την προηγούμενη απόφαση για το m δεδομένου ότι υπάρχει μία βελτίωση η οποία φαίνεται από την αρνητική μεταβολή στο σφάλμα. Το $\Delta e(t) < 0$, το οποίο εκφράζεται σε κάποιο βαθμό ως αρνητικά μικρό και σε κάποιο βαθμό ως αρνητικά μεγάλο, δηλώνει ότι το τρέχον σφάλμα πρόβλεψης $e(t)$ είναι μικρότερο από ότι το προηγούμενο σφάλμα πρόβλεψης $e(t - 1)$. Το σήμα ελέγχου, το οποίο σε αυτή την περίπτωση αντιπροσωπεύει μία

επιβράβευση, βασίζεται στην προηγούμενη απόφαση Δm . Το οποίο σημαίνει ότι, αν $\Delta m(t) > 0$ (δηλαδή, θετικά μικρό ή θετικά μεγάλο) τότε ο ελεγκτής πρέπει να συνεχίσει να αυξάνει το m , αλλιώς ο ελεγκτής πρέπει να συνεχίσει να μειώνει το m . Το ποσό της αύξησης (μείωσης) υποδεικνύεται από τα $|\Delta e(t)|$ και $|\Delta m(t)|$. Μία μικρή βελτίωση συμβολίζεται από το $\Delta e(t) = \text{nsmall}$, μία μικρή θετική αλλαγή στο m συμβολίζεται από το $\Delta m(t) = \text{psmall}$, και μία μικρή επιβράβευση στην προηγούμενη απόφαση συμβολίζεται από το $\Delta u(t) = \text{psmall}$. Έτσι, ο αντίστοιχος ασαφής κανόνας είναι: IF $e(t) = \text{small}$ AND $\Delta e(t) = \text{nsmall}$ AND $\Delta m(t) = \text{psmall}$ THEN $\Delta u(t) = \text{psmall}$. Ανάλογα, αν υπάρχει μία μεγάλη βελτίωση, δηλαδή, $\Delta e(t) = \text{nlarge}$, και $\Delta m(t) = \text{nsmall}$ τότε ο ελεγκτής πρέπει να εφαρμόσει μία επιβράβευση στην προηγούμενη απόφαση που συμβολίζεται από $\Delta u(t) = \text{nsmall}$ (IF $e(t) = \text{small}$ AND $\Delta e(t) = \text{nlarge}$ AND $\Delta m(t) = \text{nsmall}$ THEN $\Delta u(t) = \text{nsmall}$).





(c)

Σχήμα 4.6: (α) οι ζώνες ελέγχου του PIm ελεγκτή, (β) η επιφάνεια ελέγχου για Δm και Δe , (γ) η επιφάνεια ελεγκτή Δm και e .

Περίπτωση 1.2: Το $e(t)$ είναι μικρό και το $\Delta e(t)$ είναι θετικά μικρό ή θετικά μεγάλο κατά τη χρονική στιγμή t .

Σε αυτήν την περίπτωση, ο ελεγκτής δεν πρέπει να επιβραβεύσει την προηγούμενη απόφαση για το m δεδομένου ότι υπάρχει μία αύξηση στο σφάλμα. Επιπρόσθετα, το σφάλμα είναι μικρό σε αυτή την περίπτωση, έτσι, ο ελεγκτής πρέπει να εφαρμόσει μία μικρή ποινή στην προηγούμενη απόφαση για να αντιστρέψει την τάση. Το μικρό σφάλμα, δηλαδή, $|\Delta u(t)|$, είναι αναλογικό με την θετική αύξηση στο $\Delta e(t)$ και, το πρόσημο του $\Delta u(t)$ είναι το αντίθετο πρόσημο από το $\Delta m(t)$. Ο Πίνακας 4.1 απεικονίζει τους ασαφείς κανόνες για τη Ζώνη-1. Συγκεκριμένα, για την i – οστή τιμή του Δe και την j – οστή τιμή του Δm , το $[i, j]$ στοιχείο του πίνακα είναι μία τιμή του Δu .

Πίνακας 4.1. Ασαφείς κανόνες στην Ζώνη-1: $e = small$.

$\Delta m/\Delta e$	<i>nlarge</i>	<i>Nsmall</i>	<i>psmall</i>	<i>plarge</i>
<i>nlarge</i>	nlarge	Nsmall	psmall	Plarge
<i>nsmall</i>	nsmall	Nsmall	psmall	Psmall
<i>psmall</i>	psmall	Psmall	nsmall	nsmall
<i>plarge</i>	plarge	Psmall	nsmall	nlarge

Ζώνη – 2: Η δεύτερη ζώνη ασχολείται με τον προσδιορισμό του Δu όταν το σφάλμα είναι μεγάλο, το οποίο συμβαίνει, όταν ο προγνώστης αποτυγχάνει συχνά να εκτιμήσει τη μελλοντική θέση του χρήστη. Σε αυτήν την ζώνη μπορούμε να διαχωρίσουμε τις εξής περιπτώσεις:

Πίνακας 4.2. Ασαφείς κανόνες στην Ζώνη-2: $e = high$.

$\Delta m/\Delta e$	<i>nlarge</i>	<i>nsmall</i>	<i>psmall</i>	<i>plarge</i>
<i>nlarge</i>	nlarge	nsmall	psmall	plarge
<i>nsmall</i>	nsmall	nsmall	psmall	psmall
<i>psmall</i>	plarge	psmall	nsmall	nlarge
<i>plarge</i>	plarge	plarge	nlarge	nlarge

Περίπτωση 2.1: Το $e(t)$ είναι μεγάλο και το $\Delta e(t)$ είναι αρνητικά μικρό ή αρνητικά μεγάλο κατά τη χρονική στιγμή t .

Σε αυτήν την περίπτωση, υπάρχει μία αύξηση στην πρόβλεψη αλλά το σφάλμα είναι μεγάλο. Έτσι, ο ελεγκτής πρέπει να επιβραβεύσει την απόφαση για την αλλαγή του m το οποίο έχει αποτέλεσμα μία τέτοια βελτίωση. Πραγματικά, σε περίπτωση μεγάλου σφάλματος, ο ελεγκτής εφαρμόζει μεγαλύτερες επιβραβεύσεις ή ποινές οι οποίες υποδηλώνονται από υψηλές τιμές του $|\Delta u|$.

Περίπτωση 2.2: Το $e(t)$ είναι μεγάλο και το $\Delta e(t)$ είναι θετικά μικρό ή θετικά μεγάλο κατά τη χρονική στιγμή t .

Σε αυτήν την περίπτωση, υπάρχει μία επιδείνωση στην πρόβλεψη και, επίσης, το σφάλμα είναι μεγάλο. Ο ελεγκτής εφαρμόζει υψηλή ποινή στην προηγούμενη απόφαση της αλλαγής του m , η οποία έχει ως αποτέλεσμα αυτήν την επιδείνωση. Η ποινή είναι ανάλογη με το $\Delta e(t)$ και, προφανώς, το πρόσημο του σήματος ελέγχου Δu είναι το αντίθετο από αυτό του Δm . Ο Πίνακας 4.2 απεικονίζει τους ασαφείς κανόνες για τη Ζώνη-2.

Ζώνη-3: Η τρίτη ζώνη ασχολείται με τη μηδενική βελτίωση ή επιδείνωση της διαδικασίας πρόβλεψης, δηλαδή, $\Delta e = \text{μηδέν}$. Στην περίπτωση μικρού σφάλματος, η ελεγχόμενη τιμή του m συγκλίνει σε μία τιμή m^* όπου το σφάλμα ελαχιστοποιείται. Αυτό σημαίνει ότι ο προγνώστης έχει φθάσει σε σταθερή κατάσταση έως ότου ο κινητός χρήστης αλλάξει απότομα τη συμπεριφορά της κίνησης του. Ωστόσο, ο ελεγκτής εφαρμόζει ένα μηδενικό σήμα ελέγχου άπαξ και φθάσει σε σταθερή κατάσταση. Ο αντίστοιχος κανόνας είναι:

IF ($e(t) = \text{small}$) AND ($\Delta e(t) = \text{zero}$) THEN ($\Delta u(t) = \text{zero}$)

Αν το σφάλμα είναι υψηλό και δεν υπάρχει καμία επιδείνωση ή βελτίωση, ο ελεγκτής δεν εφαρμόζει ούτε μία επιβράβευση ούτε μία ποινή. Αντίθετα, εφαρμόζει ένα σήμα ελέγχου του οποίου το πρόσημο είναι το αντίθετο πρόσημο του Δm , το οποίο είχε ως αποτέλεσμα αυτήν την κατάσταση. Επιπρόσθετα, το Δu είναι ανάλογο του Δm . Ο Πίνακας 4.3 δείχνει τους ασαφείς κανόνες για αυτήν την περίπτωση.

Αξίζει να σημειωθεί ότι, όταν το όλο σύστημα είναι σε κατάσταση όπου $\Delta e = \text{μηδέν}$ και $e = \text{μεγάλο}$ τότε ο ελεγκτής εφαρμόζει ένα σήμα ελέγχου για να οδηγήσει το σύστημα στην κατάσταση όπου $\Delta e = \text{μηδέν}$ και $e = \text{μικρό}$. Όλες οι ζώνες καλύπτουν τη συνολική περιοχή ελέγχου του ασαφούς ελεγκτή (Ζώνη-1 για μικρό σφάλμα, Ζώνη-2 για μεγάλο σφάλμα και Ζώνη-3 για καμία μεταβολή στο σφάλμα).

Πίνακας 4.3. Ασαφείς κανόνες στην Ζώνη-3: $\Delta e = \text{μηδέν}$ και $\text{error} = \text{high}$.

$e/\Delta m$	<i>nlarge</i>	<i>nsmall</i>	<i>psmall</i>	<i>plarge</i>
<i>high</i>	psmall	psmall	nsmall	nsmall

Η συνολική περιοχή ελέγχου αναπαριστάται από τις επιφάνειες ελέγχου στο Σχήμα 4.6(β) και στο Σχήμα 4.6(γ). Συγκεκριμένα το Σχήμα 4.6(β) απεικονίζει το σήμα ελέγχου Δu ως μία συνάρτηση του Δm και του Δe που προβάλλεται σε όλες τις τιμές του

σφάλματος e . Μπορούμε να παρατηρήσουμε την επιφάνεια σταθερής κατάστασης στην οποία το σήμα ελέγχου είναι μηδέν όταν τα $\Delta m = 0$ και $\Delta e = 0$. Επιπρόσθετα, όταν το Δe είναι αρνητικά μεγάλο (δηλαδή, $\Delta e \rightarrow -1$) και το Δm κινείται από το -1 στο 1, το αντίστοιχο σήμα Δu κινείται από το -1 στο 1 (βλέπε Πίνακα 1). Με όμοιο τρόπο, όταν το Δm είναι θετικά μεγάλο (δηλαδή, $\Delta m \rightarrow 1$) και το Δe , κινείται από το 1 στο -1, το αντίστοιχο σήμα Δu κινείται από το -1 στο 1 (βλέπε Πίνακα 2). Το Σχήμα 4.6(γ) απεικονίζει το σήμα ελέγχου Δu ως μία συνάρτηση του Δm και του e που προβάλλεται σε όλες τις τιμές του Δe . Συγκεκριμένα, όταν το e είναι υψηλό ($e \rightarrow 1$) και το Δm κινείται από το -1 στο 1 το σήμα ελέγχου κινείται από το 1 στο -1 (βλέπε Πίνακα 4.3). Πιο ενδιαφέρουσα είναι η περίπτωση όπου παρατηρούμε ότι το σήμα ελέγχου είναι μηδέν όταν το σφάλμα είναι μικρό ($e \rightarrow 0$) ανεξάρτητα από το Δm (βλέπε τον ασαφή κανόνα στη Ζώνη-3).

4.3. Απόδοση του Ασαφώς Ελεγχόμενου Προγνώστη Θέσης

Σε αυτήν την ενότητα, θα αξιολογήσουμε την απόδοση του προτεινόμενου LP και την ικανότητα του προτεινόμενου PIm ελεγκτή να ελέγχει τον LP με σκοπό να ελαχιστοποιεί προσαρμοστικά το σφάλμα πρόβλεψης. Καταρχάς, εξετάζουμε την απόδοση του LP πειραματιζόμενοι με πραγματικά GPS ίχνη. Εξετάζουμε την πιθανότητα των ορθών προβλέψεων $P(m, l)$ και καθορίζουμε την καλύτερη τιμή του μεγέθους του παραθύρου m^* που ελαχιστοποιεί το σφάλμα πρόβλεψης. Η τιμή m^* δεν μπορεί να καθοριστεί εκ των προτέρων (δηλαδή, αναλυτικά) λόγω της εγγενούς τυχαιότητας στην συμπεριφορά κίνησης των ανθρώπων. Γι αυτόν τον λόγο, απαιτούμε ότι ο προτεινόμενος ελεγκτής προσαρμόζει την τιμή του m με σκοπό να συγκλίνει στο m^* . Όταν ο χρήστης αλλάζει τη συμπεριφορά κίνησης του, ο PIm πρέπει να αναπροσαρμόσει το m . Με στόχο την αξιολόγηση του PIm ελεγκτή, έχουμε καθορίσει τις καλύτερες τιμές για το m για τις οποίες ο LP υποθέτει το ελάχιστο σφάλμα πρόβλεψης. Έτσι, εξετάζουμε εάν ο PIm ελεγκτής συγκλίνει σε αυτές τις τιμές. Επιπρόσθετα, εξετάζουμε την προσαρμοστική συμπεριφορά του ελεγκτή. Η οποία είναι, η ικανότητα του να ανιχνεύει αλλαγές στην συμπεριφορά της κίνησης του χρήστη και να αναπροσαρμόζει τις αποφάσεις του σε νέες τιμές m^* .

4.3.1. Πειραματικές Τροχιές - Ίχνη Κίνησης

Εξετάζουμε τη συμπεριφορά του προτεινόμενου LP με ίχνη πραγματικής κίνησης από κινητούς χρήστες από τη Γερμανία, Ιταλία, Γαλλία, και Δανία. Σε αυτά τα ίχνη ο κινητός χρήστης κινείται

- Ανάμεσα σε διαφορετικές τοποθεσίες σε μία πόλη (η εμπονομαζόμενη αστική τροχιά),
- Ανάμεσα σε διαφορετικά προάστια μίας πόλης (η εμπονομαζόμενη περιαστική τροχιά), και,
- Ανάμεσα σε διαφορετικές πόλεις μέσω εθνικών οδών (η εμπονομαζόμενη δι-αστική τροχιά).

Συγκεκριμένα, οι αστικές τροχιές προέρχονται από πραγματικά GPS ίχνη ενός κινητού χρήστη που κινείται σε διάφορες τοποθεσίες στο Groningen, στη Γερμανία ((2.84 × 2.14)Km², διάρκειας: 28 λεπτών). Οι περιαστικές τροχιές είναι πραγματικά GPS ίχνη ενός κινητού χρήστη που κινείται μεταξύ προαστιακών περιοχών στην Parma, στην Ιταλία ((12.99 × 9.94)Km², διάρκειας: 31 λεπτών). Οι δι-αστικές τροχιές αναφέρονται σε πραγματικά GPS ίχνη ενός κινητού χρήστη που κινείται μεταξύ Γαλλικών πόλεων ((15.55 × 44.40)Km², διάρκειας: 51 λεπτών). Τελικά, εξετάζουμε την περίπτωση στην οποία ο κινητός χρήστης ταξιδεύει μέσα σε μία χώρα. Το οποίο σημαίνει ότι, ο κινητός

χρήστης αλλάζει από αστικές τροχιές σε περιαστικές τροχιές και/ή σε δι-αστικές τροχιές και οποιοδήποτε συνδυασμό μεταξύ τους. Στα πειράματα μας, χρησιμοποιήσαμε πραγματικά GPS ίχνη ενός κινητού χρήστη που ταξιδεύει στη Δανία και καλύπτει επιφάνεια 540 Km^2 . Ο χρήστης κινείται μεταξύ διαφορετικών τοποθεσιών στις πόλεις, μεταξύ διαφορετικών προαστίων μίας πόλης, και μεταξύ διαφορετικών πόλεων. Ο στόχος μας είναι να εξετάσουμε εάν ο προτεινόμενος PIm είναι ικανός να ανιχνεύει αλλαγές στην συμπεριφορά του χρήστη και να προσαρμόζει το μοντέλο του ανάλογα. Οι GPS εγγραφές εγγράφονται κάθε δευτερόλεπτο, το οποίο σημαίνει ότι $q = 1\text{Hz}$. Το Σχήμα 4.7 απεικονίζει την αστική τροχιά ενός κινητού χρήστη στο Groningen, την δι-αστική τροχιά ενός κινητού χρήστη που ταξιδεύει σε εθνικές οδούς μεταξύ πόλεων και την περιαστική τροχιά στην Ιταλία. Περισσότερα GPS ίχνη μπορούν επίσης να βρεθούν στο [25].

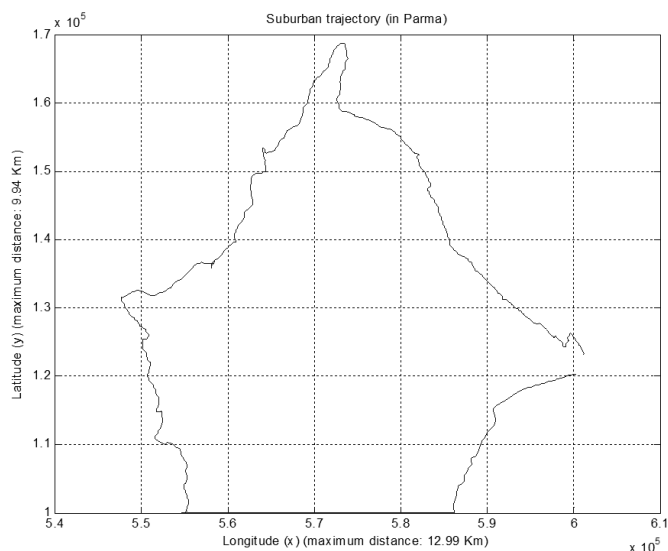
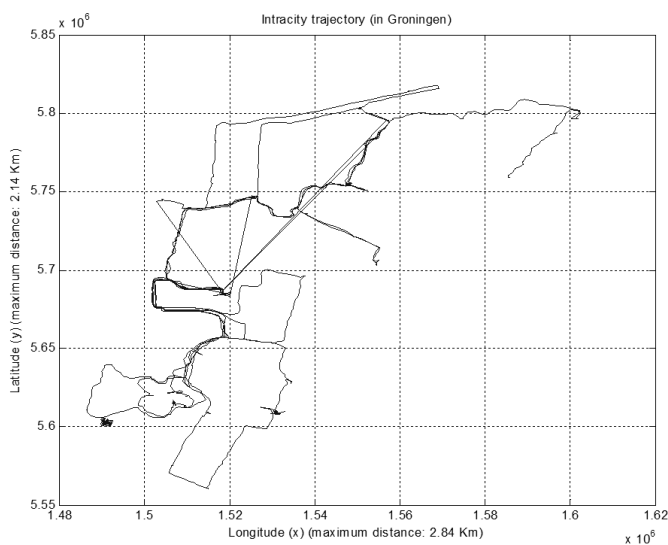
4.3.2. Παράμετροι για Εκτίμηση Απόδοσης

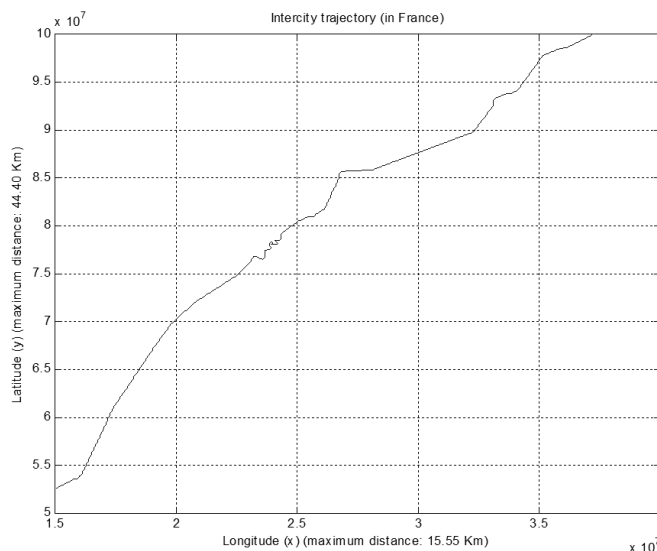
Σε αυτήν την ενότητα εισάγουμε τις παραμέτρους του συστήματος της προσαρμοστικής βραχυπρόθεσμης πρόβλεψης θέσης. Η μετρική απόδοσης για τον προτεινόμενο LP είναι η πιθανότητα μίας επιτυχούς $m + l$ πρόβλεψης $P(m, l)$. Μία απόφαση πρόβλεψης στον χρονικό ορίζοντα l σε σχέση με το μέγεθος του παραθύρου m θεωρείται επιτυχής εφόσον η πραγματική θέση (x, y) και η προβλεπόμενη θέση (x', y') του κινητού χρήστη βρίσκονται μέσα σε έναν κύκλο ακτίνας (κατώφλι) θ , δηλαδή, $\|(x, y) - (x', y')\| \leq \theta$. Η τιμή του κατωφλίου θ εξαρτάται από την εφαρμογή, και υποδηλώνει την ευαισθησία της κινητής εφαρμογής στην εκτίμηση θέσης και πρόβλεψη θέσης του κινητού χρήστη. Η τιμή του κατωφλίου δεν μπορεί να είναι μικρότερη από ότι η ακρίβεια του στίγματος του υιοθετούμενου συστήματος εντοπισμού (δηλαδή, GPS), που υποδηλώνεται ως θ_{min} . Συγκεκριμένα, η απαίτηση για ακριβή εκτίμηση θέσης, και με τη σειρά της, μίας επιτυχούς πρόβλεψης, εξαρτάται από την ακρίβεια του συστήματος εντοπισμού. Εάν το σύστημα εντοπισμού δεν μπορεί να διαχωρίσει μεταξύ θέσεων που βρίσκονται μέσα σε έναν κύκλο ακτίνας μικρότερο ή ίσης με θ_{min} τότε η πραγματική και η προβλεπόμενη θέσεις δεν είναι διακριτές. Εκτός των άλλων, κάποιος θα μπορούσε να ισχυριστεί ότι το εξαρτώμενο από την εφαρμογή κατώφλι θ σε μία ντετερμινιστική κίνηση χρήστη (δηλαδή, δι-αστική τροχιά) μπορεί να υποθέσει μία μικρή τιμή κοντά στο θ_{min} . Από την άλλη πλευρά, μία τροχιά με υψηλό βαθμό τυχαιότητας έχει ως αποτέλεσμα χαμηλές τιμές στην $P(m, l)$ όσο η τιμή του θ μειώνεται.

Το Σχήμα 4.8 και το Σχήμα 4.9 απεικονίζουν την συμπεριφορά της τιμής της $P(m, l)$ σε σχέση με την τιμή του θ για διαφορετικές $m + l$ προβλέψεις. Συγκεκριμένα, η τιμή θ κυμαίνεται από 5 έως 30 μέτρα δεδομένου ότι ένας απλός GPS δέκτης έχει ακρίβεια περίπου 10 μέτρων. Ένας θα μπορούσε να παρατηρήσει, στο Σχήμα 4.8, ότι η $P(10, 5)$ τιμή (δηλαδή, $10 + 5$ μοντέλο πρόβλεψης) για την δι-αστική τροχιά με θ κοντά στα 5 μέτρα είναι σχεδόν ίση με την $P(10, 5)$ τιμή για την περιαστική τροχιά με θ κοντά στα 10 μέτρα, το οποίο είναι 100% αύξηση του θ . Αρκετά όμοιες παρατηρήσεις ισχύουν και στο Σχήμα 4.9 ($5 + 5$ μοντέλο πρόβλεψης). Επιπρόσθετα, για μία τιμή θ πάνω από 20 (και 15) μέτρα, η αντίστοιχη $P(10, l)$ (και $P(5, l)$). Πιθανότητα υποθέτει τιμή πάνω από 0.95 (βλέπε Σχήμα 4.8 και Σχήμα 4.9). Ως εκ τούτου, στα πειράματα μας θέτουμε το θ στα 10 μέτρα με σκοπό να εξετάσουμε την πιθανότητα των επιτυχών προβλέψεων που επιτυγχάνονται από το μοντέλο μας. Ως εκ τούτου, μία κινητή εφαρμογή μπορεί να προσαρμόσει την τιμή θ στη συμπεριφορά κίνησης του χρήστη.

Ας ορίσουμε τον παράγοντα $r = l/m$ ως το ποσό της πληροφορίας θέσης του μεγέθους m , το οποίο απαιτείται για να προβλέψουμε τη μελλοντική θέση l χρονικές στιγμές εκ των προτέρων. Η συχνότητα δειγματοληψίας της θέσης είναι $q = 1\text{Hz}$. Οι προηγούμενες

m θέσεις του χρήστη αναφέρονται στα προηγούμενα m δευτερόλεπτα της κίνησης του χρήστη. Η διαδικασία πρόβλεψης κατά τη χρονική στιγμή t για την εκτίμηση της θέσης του χρήστη κατά το χρονικό δείκτη $t + l$, εκμεταλλεύεται την ιστορία κίνησης στο παράθυρο που κυμαίνεται από $t - m$ έως και t . Ως εκ τούτου, ο παράγοντας r υποδηλώνει την αναλογία της ιστορίας που χρησιμοποιείται με σκοπό να προβλέψουμε τις μελλοντικές θέσεις. Για παράδειγμα, $r = 15/10 = 1.5$, με $m = 10$ προηγούμενες θέσεις και $l = 15$ s, υποδηλώνει ότι απαιτούμε εγγραφές θέσης για τα τελευταία 10 δευτερόλεπτα με σκοπό να προβλέψουμε τη μελλοντική θέση μετά από 15 s. Αξίζει να σημειώσουμε ότι η διάρκεια των l χρονικών μονάδων εξαρτώνται από την εφαρμογή.

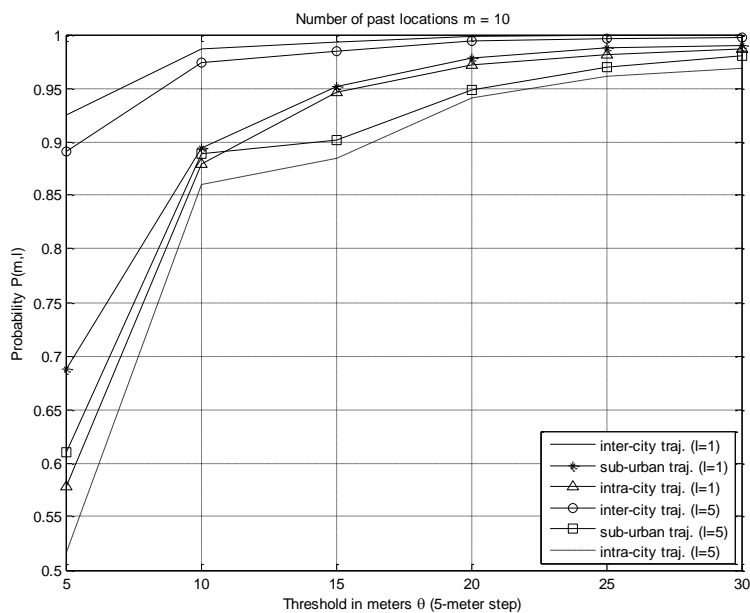




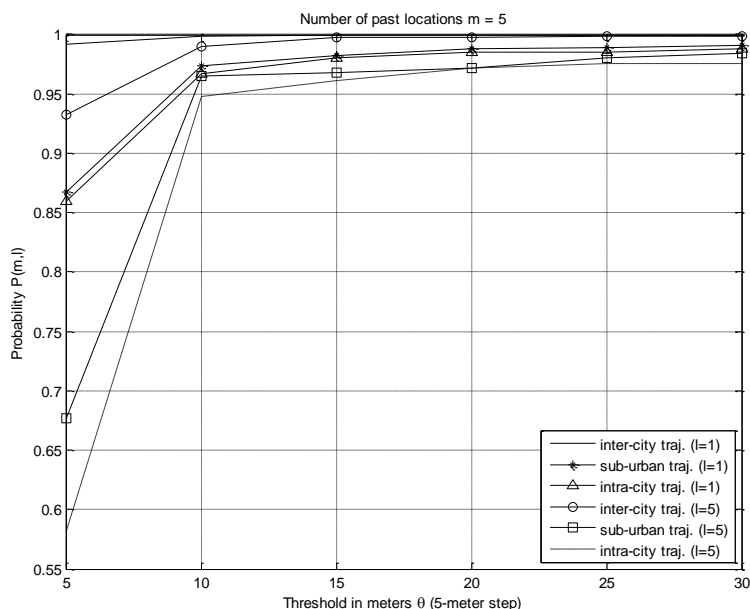
Σχήμα 4.7. Αστικές, περιαστικές και δι-αστικές τροχιές

Όπως ήδη συζητήσαμε, το m^* δεν είναι σταθερό για ορισμένο χρονικό διάστημα. Όταν ο χρήστης αλλάξει τη συμπεριφορά κίνησης του, ο ελεγκτής επαναπροσδιορίζει το m με σκοπό να προσαρμόσει αυτήν την αλλαγή έτσι ώστε να προκύψει ένα νέο m^* . Αυτό σημαίνει ότι το αντίστοιχο σήμα ελέγχου Δu ποτέ δεν παίρνει τιμή μηδέν, αντίθετα, κυμαίνεται γύρο από το μηδέν έως ότου μία αλλαγή στην κίνηση του χρήστη ανιχνευτεί. Έτσι, ο παράγοντας r είναι μικρότερος, ίσος ή μεγαλύτερος από την μονάδα εξαρτώμενος από την συμπεριφορά κίνησης του χρήστη.

- Μία τιμή του r μικρότερη από τη μονάδα ($r < 1$) υποδηλώνει ότι η απαιτούμενη μνήμη (m εγγραφές θέσης) είναι μεγαλύτερη από ότι ο ορίζοντας πρόβλεψης (l παραδείγματα μπροστά στο χρόνο). Θεωρείστε την περίπτωση στην οποία $l = 1$ s και $m > 1$: η πρόβλεψη γίνεται για κάθε επόμενο δευτερόλεπτο.
- Η περίπτωση όπου το $r = 1$, δηλαδή, $m = l$, είναι πολύ σημαντική. Θεωρείστε ότι $l = m = 30$, το οποίο δηλώνει ότι αποθηκεύουμε ιστορική κίνηση μισού λεπτού για να προβλέψουμε την θέση του χρήστη στο επόμενο μισό λεπτό. Αυτό δεν θα έχει ως αποτέλεσμα το ίδιο σφάλμα πρόβλεψης όπως είχαμε όταν $l = m = 15$, αν και το $r = 1$ και για τις δύο περιπτώσεις. Για παράδειγμα, αν η συμπεριφορά κίνησης είναι υψηλά τυχαία (δηλαδή, ο χρήστης κινείται μέσα στο κέντρο της πόλης) τότε όσο πιο μεγάλη είναι η ιστορία που αποθηκεύουμε τόσο πιο μεγάλη είναι η πιθανότητα να έχουμε επιπρόσθετο θόρυβο στο μοντέλο παλινδρόμησης μας. Αξίζει να σημειωθεί ότι μία μεγάλη τιμή του m υποδηλώνει ότι ο προγνώστης λαμβάνει υπόψη του ένα εκτεταμένο ίχνος κίνησης το οποίο πιθανά περιλαμβάνει έναν μεγάλο αριθμό από ξαφνικές αλλαγές στην κίνηση.
- Μία τιμή του r υψηλότερη από τη μονάδα ($r > 1$) υποδηλώνει ότι ο ορίζοντας πρόβλεψης (l) είναι μεγαλύτερος από ότι η αποθηκευμένη μνήμη (m). Για παράδειγμα, αποθηκεύουμε τα τελευταία 20 s από την κίνηση του χρήστη και προβλέπουμε την κίνηση του χρήστη μετά από 30 δευτερόλεπτα, $r = 3/2$.



Σχήμα 4.8. Πιθανότητα μίας επιτυχούς $m+l$ πρόβλεψης $P(m, l)$ με την τιμή κατωφλίου θ ($m = 10$)

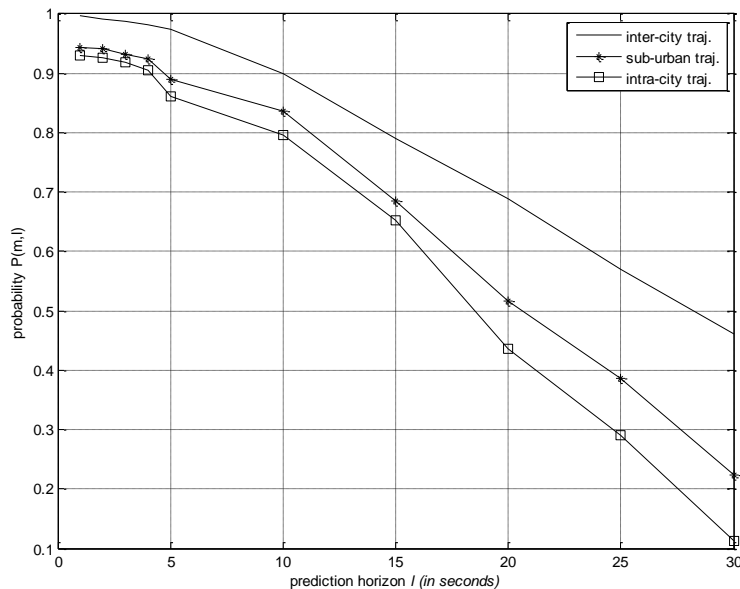


Σχήμα 4.9. Πιθανότητα μίας επιτυχούς $m+l$ πρόβλεψης $P(m, l)$ σε σχέση με την τιμή κατοφλίου θ ($m = 5$)

Μία άλλη βασική παράμετρος του PIm είναι το ποσοστό υποστήριξης μ της αλλαγής στο σήμα ελέγχου Δu . Αυτό το ποσοστό υποδηλώνει πόσο γρήγορα το ελεγχόμενο σύστημα LP αλλάζει την έξοδο του, λόγω του σήματος ελέγχου του PIm . Η τιμή του μ υποδηλώνει έναν πολλαπλασιαστή επί του ποσοστού αλλαγής του m λόγω του Δu . Για παράδειγμα, μία 10% μείωση ($\Delta u = -0.1$) με $\mu = 4$, έχει ως αποτέλεσμα 40% μείωση στο m . Μία μικρή τιμή του μ υποδηλώνει ότι ο LP κάνει μικρές αλλαγές στο m ενώ μία μεγάλη τιμή έχει ως αποτέλεσμα πιο δραστικές αλλαγές στο m . Η τιμή του μ επηρεάζει την ταχύτητα σύγκλισης του ελεγκτή όπως θα δούμε παρακάτω.

4.3.3. Εκτίμηση Απόδοσης του Ασαφώς Ελεγχόμενου Προγνώστη Θέσης

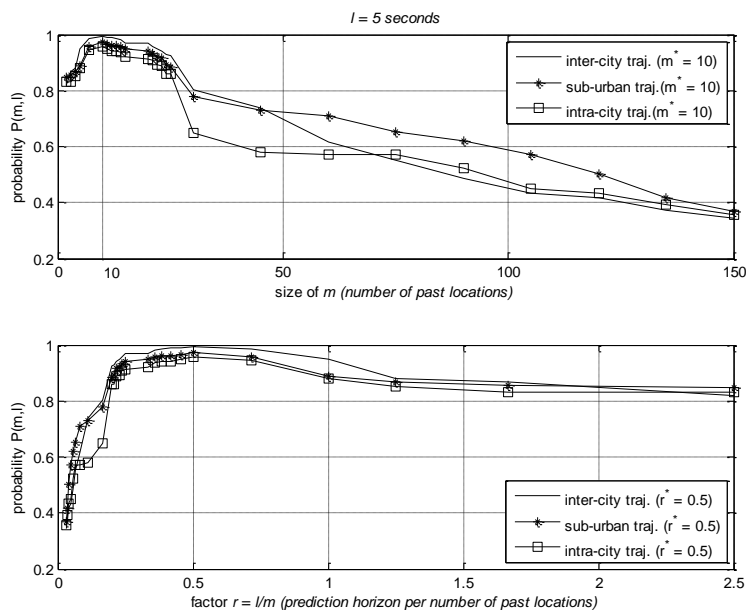
Καταρχάς, εξετάζουμε την πιθανότητα επιτυχών προβλέψεων που προέρχονται από τον LP με το να πειραματιστούμε με διάφορες τιμές του m και του l χωρίς να εισάγουμε τον PIm ελεγκτή.



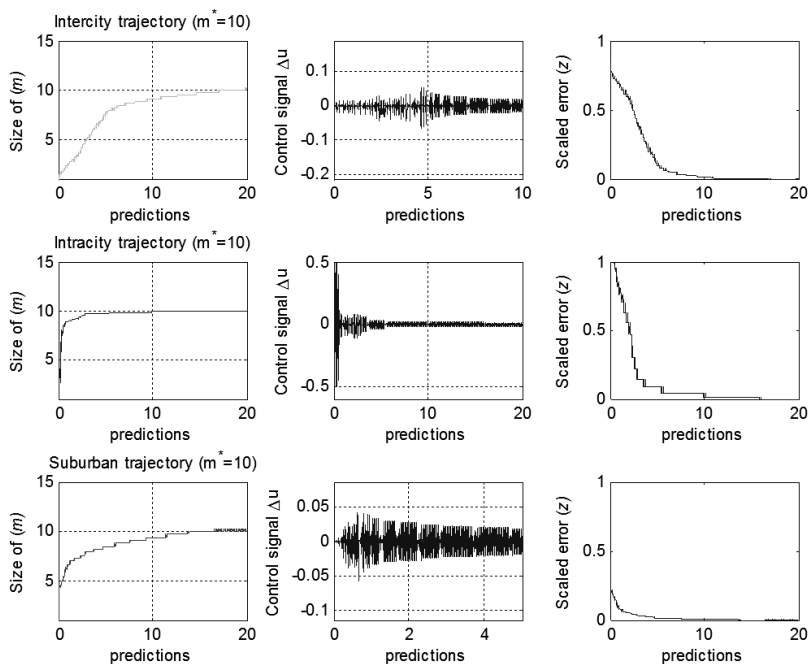
Σχήμα 4.10. Πιθανότητα επιτυχούς πρόβλεψης για $m=10$ s και διαφορετικές l τιμές

Το Σχήμα 4.10 απεικονίζει την πιθανότητα της επιτυχούς πρόβλεψης $P(10,l)$ για διάφορες τιμές του l για $N = 500$ προβλέψεις. Αυτό το μέγεθος του m δεν είναι το πιο κατάλληλο για να κάνουμε προβλέψεις με ελάχιστο σφάλμα πρόβλεψης. Όπως παρουσιάζεται στο Σχήμα 4.10, για χαμηλή τιμή του l (βραχυπρόθεσμη πρόβλεψη), η $P(10,l)$ έχει υψηλές τιμές. Για παράδειγμα, για $l = 5 < m$, η $P(10,l)$ έχει υψηλές τιμές για κάθε τύπο τροχιάς ($r = 0.5$). Επιπρόσθετα, για $r = 1$, το οποίο σημαίνει ότι $m = l = 10$, η πιθανότητα της πρόβλεψης έχει τιμές 0.8 και 0.9 για τις αστικές και δι-αστικές τροχιές, αντίστοιχα. Όταν η τιμή του l αυξάνεται το σφάλμα πρόβλεψης επίσης αυξάνεται, για σταθερό m . Για $r = 2$, η $P(10,20)$ έχει τιμές μεταξύ 0.45 και 0.7 για όλους τους τύπους των τροχιών.

Όπως δείξαμε, με χαμηλές απαιτήσεις μνήμης (m) μπορούμε να προβλέψουμε με μεγάλη ακρίβεια τοποθεσίες στο κοντινό μέλλον (μικρό l). Επιπρόσθετα, η τιμή του l είναι πολύ κρίσιμη για την εφαρμοσιμότητα του LP. Συγκεκριμένα, για $r = 1$ ($m = l$) η $P(m,l)$ έχει διαφορετικές τιμές για διαφορετικές τιμές του l και τύπων τροχιών. Ας θεωρήσουμε την δι-αστική τροχιά, στην περίπτωση μίας πολύ βραχυπρόθεσμης πρόβλεψης με $l = 5$ s και $r = 1$, ο LP επιτυγχάνει $P(5,5) = 0.98$ (βλέπε Σχήμα 4.11). Στην περίπτωση όπου το $l = 15$ και $l = 30$, ο LP επιτυγχάνει $P(15,15) = 0.7611$ (βλέπε Σχήμα 4.10) και $P(30,30) = 0.4627$ (βλέπε Σχήμα 4.12), αντίστοιχα για τον ίδιο τύπο τροχιάς. Ωστόσο, η τιμή του m η οποία ελαχιστοποιεί το σφάλμα πρόβλεψης θα καθοριστεί από τον PIm ελεγκτή. Όπως θα δούμε παρακάτω, η καλύτερη r τιμή για βραχυπρόθεσμες προβλέψεις είναι μικρότερη της μονάδας, και για μακροπρόθεσμες προβλέψεις είναι μεγαλύτερη της μονάδας.



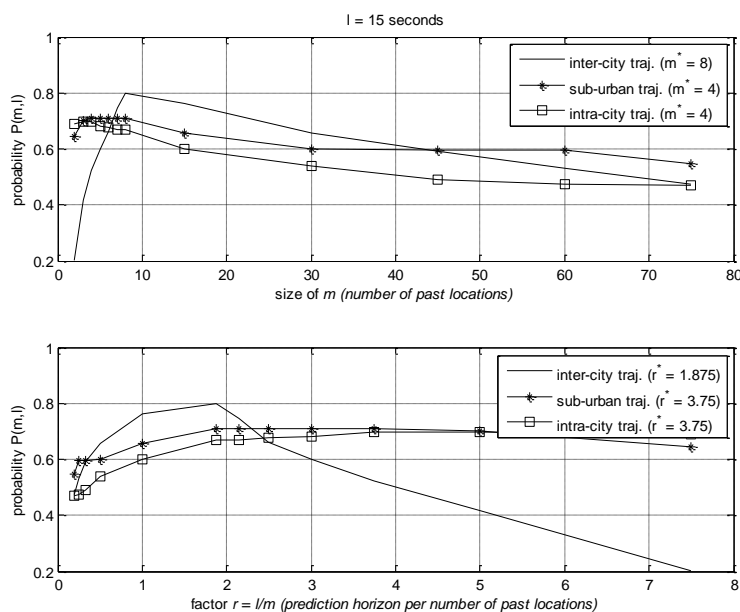
Σχήμα 4.11. Πιθανότητα επιτυχούς πρόβλεψης σε σχέση με το $r = l/m$ για $l = 5$ s.



Σχήμα 4.12. Σύγκλιση, σήματα ελέγχου και σφάλμα πρόβλεψης για δι-αστικές, αστικές και περιαστικές τροχιές για $(m + 5)$ πρόβλεψη.

Τώρα εκτιμούμε την απόδοση του LP με τη χρήση του PI_m . Η τιμή του m δεν μπορεί να καθοριστεί εκ των προτέρων και, το πιο σημαντικό είναι ότι, δεν μπορεί να είναι η ίδια για όλα τα είδη των τροχιών. Ο PI_m ελεγκτής προσαρμόζει την τιμή του m με σκοπό να καθορίσει την τιμή του r που ελαχιστοποιεί το σφάλμα πρόβλεψης για έναν συγκεκριμένο τύπο τροχιάς σε σχέση με το l . Με σκοπό να εκτιμήσουμε την ικανότητα ελέγχου του ελεγκτή, βρήκαμε πειραματικά την τιμή του m που ελαχιστοποιεί το σφάλμα πρόβλεψης, m^* , για μία δεδομένη τροχιά. Έτσι, απαιτούμε ότι ο PI_m συγκλίνει στο m^* . Αρχικά θέτουμε την ελάχιστη πιθανή τιμή για το m , δηλαδή, $m = 1$. Αναφέρουμε τα αποτελέσματά μας για $l = 5, 15$ και 30 s (βράχυ-, μέσο- και μακρο-πρόθεσμες

προβλέψεις). Σε κάθε περίπτωση εκτιμούμε την συμπεριφορά του ελεγχόμενου LP και την συμπεριφορά σύγκλισης του PIm σε σχέση με το l .

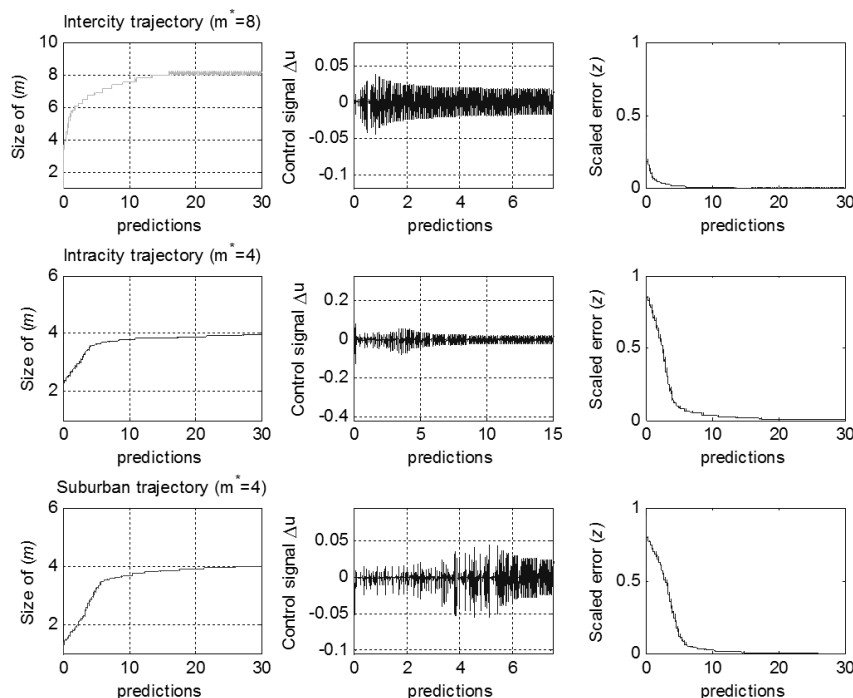


Σχήμα 4.13. Πιθανότητα επιτυχούς πρόβλεψης σε σχέση με το $r = l/m$ για $l=15$ s.

4.3.3.1. Πειραματισμός με την $(m + 5)$ πρόβλεψη

Εξετάζουμε την περίπτωση των βραχυπρόθεσμων προβλέψεων, δηλαδή, $l = 5$. Το Σχήμα 4.11 απεικονίζει την συμπεριφορά της $P(m, 5)$ για m να κυμαίνεται από 1 έως 150 με σταθερό $l = 5$ s. Προφανώς, ο LP επιτυγχάνει υψηλές τιμές ορθών πιθανοτήτων πρόβλεψης ($P(m, 5) > 0.9$) ειδικά όταν το r έχει τιμές υψηλότερες από 0.5 και μικρότερες από την μονάδα. Αυτό υποδηλώνει την ικανότητα του LP να προβλέπει με ακρίβεια κοντινές μελλοντικές θέσεις με χαμηλή απαίτηση σε μνήμη. Επιπρόσθετα, αξίζει να σημειωθεί ότι, στην περίπτωση των βραχυπρόθεσμων προβλέψεων η αύξηση του m δεν είναι σημαντική. Επιπλέον, από την σχεδίαση της $P(m, l)$ σε σχέση με το r καθορίζουμε το εύρος των τιμών για το r όπου μία αύξηση του m δεν σηματοδοτεί περαιτέρω αύξηση στην πιθανότητα της επιτυχούς πρόβλεψης. Όπως δείξαμε στο Σχήμα 4.11, για τιμές από $10 < m < 50$ ($0.1 < r < 0.5$), η $P(m, 5)$ κυμαίνεται μεταξύ 0.6 και 0.99. Για $m > 50$ η $P(m, 5)$ έχει χαμηλές τιμές, έτσι, αποθηκεύοντας ένα μεγάλο αριθμό από προηγούμενες θέσεις δεν είναι μία ορθή προσέγγιση. Επιπρόσθετα, για $r \geq 0.5$ ($m \leq 10$) η τιμή της $P(m, 5)$ κυμαίνεται από 0.82 έως 1. Το m^* για όλες τις τροχιές για $l = 5$ είναι $m^* = 10$, δηλαδή, ο παράγοντας $r^* = l/m^*$ είναι 0.5. Η αντίστοιχη προσαρμογή του m μέσω του σήματος ελέγχου του PIm για όλες τις τροχιές παρουσιάζεται στο Σχήμα 4.12. Αρχικοποιούμε τον ελεγκτή με $m = 1$. Ο ελεγκτής επιτυγχάνει την m^* τιμή για τις αστικές, περιαστικές και δι-αστικές τροχιές με ποσοστό υποστήριξης $\mu = 4$. Στην περίπτωση της αστικής τροχιάς, το σήμα ελέγχου του ελεγκτή είναι πιο επιθετικό από ότι στις υπόλοιπες τροχιές, όπως φαίνεται στο Σχήμα 4.12. Επιπρόσθετα, ο PIm συγκλίνει γρηγορότερα στην περίπτωση της αστικής τροχιάς από ότι στην περίπτωση της δι-αστικής και της περιαστικής τροχιάς. Ο ελεγκτής έχει φθάσει σε σταθερή κατάσταση στις πρώτες 10 (για την αστική τροχιά) και 18 (για την δι-αστική τροχιά) προβλέψεις. Όταν ο PIm συγκλίνει στο m^* , αυτό αντιστοιχεί στο $e^* = 1 - P(m^*, l)$. Το οποίο σημαίνει ότι το μικρότερο εφικτό σφάλμα είναι e^* όταν $m = m^*$. Η τιμή του e^* καθορίζει την βέλτιστη ποιότητα πρόβλεψης του προτεινόμενου LP η οποία μπορεί να επιτευχθεί για δοσμένο l ($e^* = 0.003, 0.0101$ και 0.03 για την δι-αστική,

περιαστική και αστική τροχιά, αντίστοιχα). Το Σχήμα 4.12 απεικονίζει το κλιμακωτό σφάλμα πρόβλεψης z , το οποίο έχει μηδενική τιμή για το m^* . Το κλιμακωτό z είναι η τιμή για το $z = (e - e^*) / (1 - P(m^*, l))$, το οποίο σημαίνει ότι, $z = 0$ υποδηλώνει το ελάχιστο σφάλμα πρόβλεψης z που μπορεί να επιτευχθεί από τον LP.



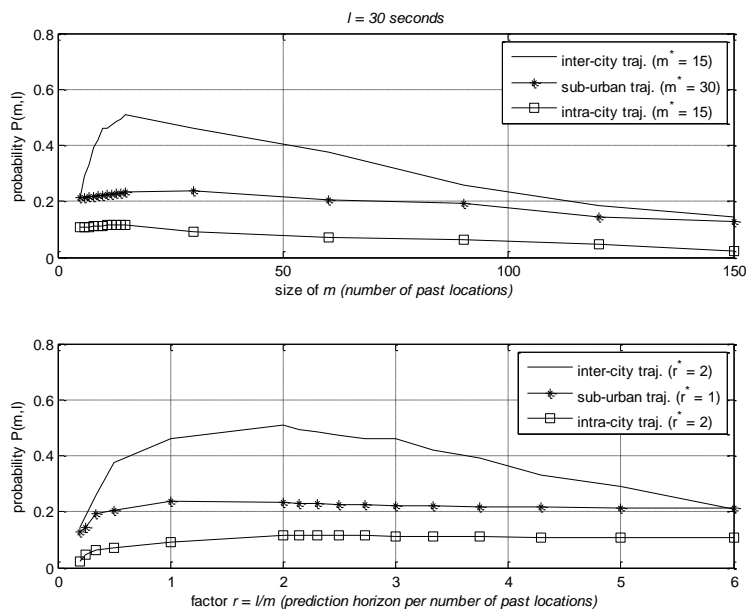
Σχήμα 4.14. Σύγκλιση, σήματα ελέγχου και σφάλμα πρόβλεψης για δι-αστικές, αστικές και περιαστικές τροχιές για $(m + 15)$ πρόβλεψη

4.3.3.2. Πειραματισμός με την $(m + 15)$ πρόβλεψη

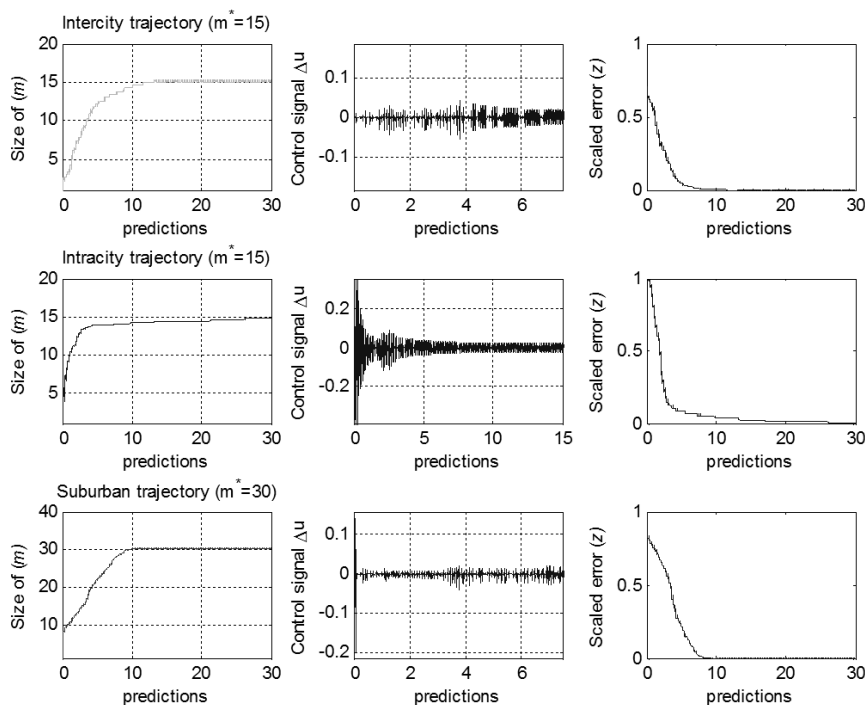
Το Σχήμα 4.13 απεικονίζει τη συμπεριφορά της $P(m, 15)$ με r για m να κυμαίνεται από 1 έως 75 έχοντας σταθερό $l = 15$ s. Για την δι-αστική τροχιά το $r^* = l/m^*$ είναι 1.875. Αυτό δηλώνει ότι, χρησιμοποιούμε τις προηγούμενες $m^* = 8$ θέσεις του χρήστη με σκοπό να προβλέψουμε τη θέση του χρήστη μετά από 15 s. Στην περίπτωση της περιαστικής τροχιάς, το r^* είναι 3.75 ($m^* = 4$), και για την αστική τροχιά το r^* είναι 3.75 ($m^* = 4$). Στην τελευταία περίπτωση, στην οποία ο χρήστης ταξιδεύει μέσα στην πόλη, το μήκος του παραθύρου αναφέρεται στα τελευταία 4 s για μία πρόβλεψη 15 s στο μέλλον. Για μία αστική τροχιά στην οποία ο χρήστης αλλάζει πορεία ξαφνικά, ο LP πρέπει να αποθηκεύει ένα μικρό παράθυρο κίνησης έτσι ώστε να ελαχιστοποιείται ο θόρυβος στο μοντέλο παλινδρόμησης, σε σχέση με το l . Στην περίπτωση της δι-αστικής τροχιάς, ο LP μπορεί να αποθηκεύσει περισσότερες προηγούμενες θέσεις από ότι στην αστική τροχιά εφόσον οι προηγούμενες θέσεις απαιτούνται για να καθορίσουν μία ομαλότερη κίνηση του χρήστη.

Τώρα εκμεταλλευόμαστε τον PIm έτσι ώστε να συγκλίνει στο m^* το οποίο ελαχιστοποιεί την $P(m^*, l)$ για ένα δοσμένο l . Συγκεκριμένα, ο ελεγκτής προσαρμόζει δυναμικά το m για μία συγκεκριμένη κίνηση του χρήστη έως ότου το σφάλμα $e = 1 - P(m, l)$ είναι όσο μικρότερο γίνεται. Το ελάχιστο επιτευχθέν σφάλμα e^* όταν $m = m^*$ είναι $1 - P(m^*, l) = 0.2$ για μία δι-αστική τροχιά με $l = 15$ δευτερόλεπτα και $m^* = 8$. Επιπρόσθετα $e^* = 0.28$ και $e^* = 0.3$ για την περιαστική (με $m^* = 4$) και την αστική (με $m^* = 4$) τροχιά, αντίστοιχα. Το Σχήμα 4.14 απεικονίζει την σύγκλιση του PIm στο m^* για όλες τις τροχιές με $l = 15$ μαζί με τα αντίστοιχα σήματα ελέγχου και τα κλιμακωτά σφάλματα.

Αρχικοποιούμε τον ελεγκτή με $m = 1$ και ποσοστό υποστήριξης $\mu = 4$. Για κάθε πρόβλεψη ο ελεγκτής προσαρμόζει την απόφαση για το m βασιζόμενος στο σήμα ελέγχου Δu όπως παρουσιάζεται στο Σχήμα 4.14. Ο ελεγκτής έχει συγκλίνει στις πρώτες 16 (για την δι-αστική τροχιά) και 27 (για την αστική τροχιά) προβλέψεις νωρίτερα από ότι στην περίπτωση της $m + 5$ πρόβλεψης.



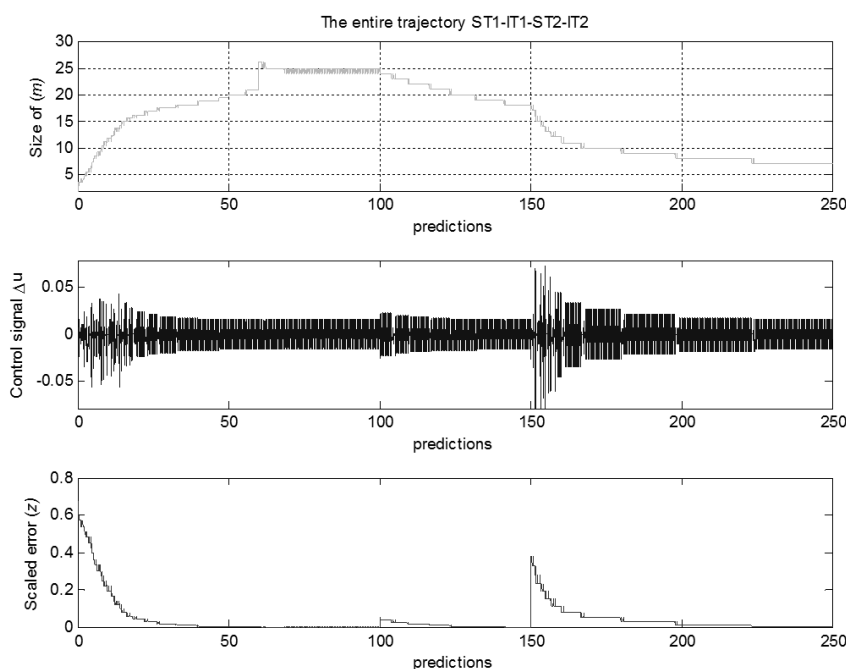
Σχήμα 4.15. Πιθανότητα επιτυχούς πρόβλεψης σε σχέση με το $r = l/m$ για $l=30$ s.



Σχήμα 4.16. Σύγκλιση, σήματα ελέγχου και σφάλμα πρόβλεψης για δι-αστικές, αστικές και περιαστικές τροχιές για $(m + 30)$ πρόβλεψη.

4.3.3.3. Πειραματισμός με την $(m + 30)$ πρόβλεψη

Μελετούμε τη συμπεριφορά του LP και του PIm για τις μακροπρόθεσμες προβλέψεις, δηλαδή, $l = 30$. Το Σχήμα 4.15 απεικονίζει τη συμπεριφορά της $P(m, l)$ με m να κυμαίνεται από 1 έως 100 χρονικά παραδείγματα. Όπως παρουσιάζεται στο Σχήμα 4.15, για την δι-αστική τροχιά το $r^* = l/m^* = \text{argmax}\{P(m, l) | r = l/m\}$ είναι 2, το οποίο σημαίνει ότι $m^* = 15$. Ως εκ τούτου, εκμεταλλευόμαστε τις προηγούμενες $m^* = 15$ θέσεις της κίνησης του χρήστη με σκοπό να προβλέψουμε την θέση του χρήστη μετά από 30 s σε μία δι-αστική τροχιά. Το ίδιο ισχύει και για την αστική τροχιά στην οποία $r^* = 2$, έτσι, $m^* = 15$. Στην περίπτωση μίας περιαστικής τροχιάς, το $r^* = 1$, δηλαδή, $m^* = 30$. Στο Σχήμα 4.16 παρουσιάζουμε την προσαρμογή του m , το αντίστοιχο σήμα ελέγχου και το κλιμακωτό σφάλμα του PIm με σκοπό τη σύγκλιση στο m^* για κάθε τύπο τροχιάς έχοντας ορίζοντα πρόβλεψης $l = 30$ s. Αρχικοποιούμε τον ελεγκτή με $m = 1$ έχοντας ποσοστό υποστήριξης $\mu = 4$ και παρατηρούμε ότι το σήμα ελέγχου Δu είναι πιο επιθετικό στην περίπτωση της αστικής τροχιάς από ότι στην δι-αστική τροχιά αφού η πρώτη κίνηση είναι πιο τυχαία από ότι η τελευταία. Ο ελεγκτής έχει συγκλίνει στις πρώτες 10 (για την περιαστική τροχιά) και 27 (για την αστική τροχιά) προβλέψεις αντίστοιχα.



Σχήμα 4.17. Ο χρήστης αλλάζει συμπεριφορά κίνησης: περιαστική τροχιά ($m^* = 20$), δι-αστική τροχιά ($m^* = 25$), περιαστική τροχιά ($m^* = 18$), αστική τροχιά ($m^* = 6$)

4.3.3.4. PIm προσαρμοστικότητα

Η τιμή m^* που απορρέει από τον ελεγκτή είναι δυναμική όταν ο χρήστης ακολουθεί ένα δυναμικό πρότυπο κίνησης (το οποίο αποδίδεται είτε σε χωρικούς περιορισμούς ή σε στοχαστική κίνηση του χρήστη). Ένα τέτοιο πρότυπο υποδηλώνει ότι ο ελεγκτής πρέπει να αναπροσαρμόσει την τιμή του m . Το m^* αλλάζει μετά από μία αλλαγή στην κίνηση του χρήστη, δηλαδή, η νέα τροχιά είναι *διαφορετική* από τις προηγούμενες. Η διαφορά μεταξύ δύο τροχιών σηματοδοτείται από μία αλλαγή στο ελάχιστο σφάλμα e^* . Έτσι, βασιζόμενοι στους ασαφείς κανόνες ελέγχου στους Πίνακες 1, 2, και 3, μία αλλαγή στο σφάλμα $\Delta e(t) = e(t) - e^*$ ανιχνεύεται και, έτσι, όλοι οι ασαφείς κανόνες «ενεργοποιούνται» βασιζόμενοι στους βαθμούς συμμετοχής του $\Delta e(t)$, $e(t)$ και στην προηγούμενη απόφαση για το m , $\Delta m(t)$. Το οποίο σημαίνει ότι όλοι οι ασαφείς κανόνες

παράγουν εξόδους, και μέσω της μεθόδου αποασαφοποίησης, ο PI_m δημιουργεί τα σήματα ελέγχου $\Delta u(t)$. Ακολούθως, ο PI_m βασιζόμενος στον προσαρμοστικό κανόνα στην (6) προσαρμόζεται στο $m(t) = m^* + \mu \cdot \Delta u(t) \cdot m^*$. Όταν μία νέα αλλαγή στο ελάχιστο σφάλμα ανιχνευτεί, ο PI_m αρχικοποιεί ξανά την ίδια διαδικασία. Ένα νέο m^* επιτυγχάνεται όταν το αντίστοιχο σήμα ελέγχου Δu διαταράσσεται γύρω από το μηδέν (έως ότου μία αλλαγή στην συμπεριφορά κίνησης ανιχνευτεί).

Εκτιμούμε την ικανότητα προσαρμοστικότητας του προτεινόμενου ελεγκτή με το να πειραματιστούμε με πραγματικά GPS ίχνη από τη Δανία. Ο χρήστης οδηγεί σε προάστια (περιαστική τροχιά 1 – ST1), μετά ο χρήστης ταξιδεύει μέσω της εθνικής οδού (δι-αστική τροχιά – IT1), ο χρήστης μπαίνει ξανά σε προάστιο (περιαστική τροχιά 2 – ST2) και, τελικά, ο χρήστης συνεχίζει την διαδρομή του/της στο κέντρο της πόλης (αστική τροχιά – IT2). Έχουμε καθορίσει την τιμή του m^* για κάθε τροχιά για ορίζοντα πρόβλεψης $l = 20$ s, δηλαδή, $m^*(ST1) = 20$, $m^*(IT1) = 25$, $m^*(ST2) = 18$, και $m^*(IT2) = 6$. Το Σχήμα 4.17 απεικονίζει την εξέλιξη του m , το κλιμακωτό σφάλμα πρόβλεψης και το σήμα ελέγχου για τη συνολική τροχιά ST1-IT1-ST2-IT2 που απορρέουν από τον PI_m ελεγκτή έχοντας ποσοστό υποστήριξης $\mu = 4$ και αρχική τιμή $m = 1$. Μόλις η κινητή συμπεριφορά αλλάξει, δηλαδή, μετάβαση από έναν τύπο τροχιάς σε έναν άλλο, ο ελεγκτής ανιχνεύει άμεσα τέτοιες αλλαγές και προσαρμόζει την τιμή του m ανάλογα. Επιπρόσθετα, ο ελεγκτής συγκλίνει στο m^* για κάθε μετάβαση της κάθε τροχιάς. Το Σχήμα 4.16 επίσης απεικονίζει τις αλλαγές στο σήμα ελέγχου. Πιο ξαφνικές αλλαγές δημιουργούνται όταν ο χρήστης αλλάζει συμπεριφορά κίνησης το οποίο επίσης φαίνεται στο αντίστοιχο κλιμακωτό σφάλμα.

4.3.4. Σύγκριση με άλλα Μοντέλα

Σε αυτήν την ενότητα συγκρίνουμε την συμπεριφορά του αμνήμον (stateless) μοντέλου μας με άλλα με γνώση κατάστασης (stateful) και αμνήμονα μοντέλα σε σχέση με την πιθανότητα της σωστής πρόβλεψης, απαιτήσεων χώρου και χρονικής πολυπλοκότητας. Στην αρχή, συγκρίνουμε το μοντέλο μας με την αμνήμον πολυωνυμική Lagrange μέθοδο για παρέκταση πληροφορίας θέσης (πρόβλεψη). Επιπρόσθετα, πειραματιζόμαστε με δύο με γνώση κατάστασης μοντέλα, το Adaptive Resonance Theory (ART) μοντέλο ενδυναμωμένο με ενισχυτική μάθηση [29] και τον on – line k – means αλγόριθμο [8]. Τέτοια μοντέλα βασίζονται σε πρότυπα κίνησης και χρησιμοποιούν ταξινόμηση προτύπων για μία δοσμένη τροχιά με σκοπό να προβλέψουμε την μελλοντική θέση του χρήστη. Αξίζει να σημειωθεί ότι όλα τα θεωρούμενα μοντέλα προσαρμόζουν την χωροχρονική τους βάση γνώσης στην συμπεριφορά κίνησης του χρήστη. Το κύριο κριτήριο για σύγκριση είναι τότε πόσο ακριβές και αποδοτικό, σε σχέση με τον χώρο και τον χρόνο, είναι κάθε προσαρμοστικό μοντέλο κατά την πρόβλεψη της μελλοντικής θέσης του χρήστη.

A. Πολυωνυμική Lagrange μέθοδος

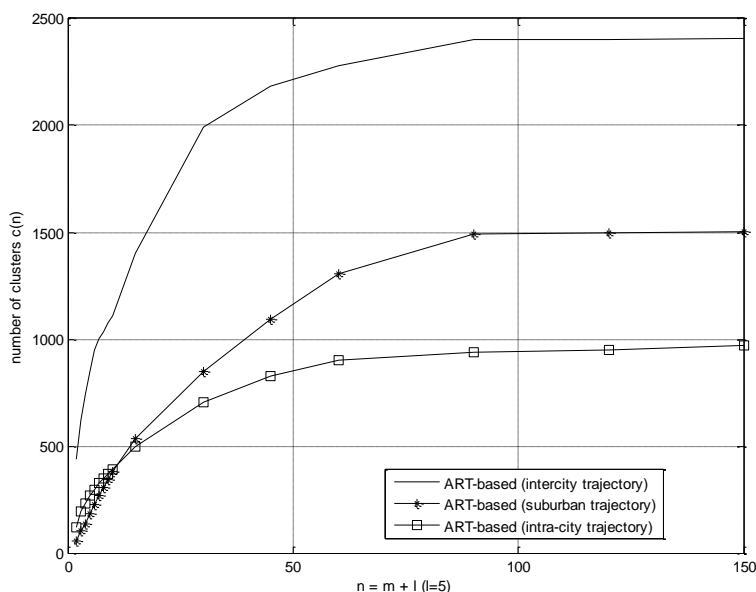
Η αμνήμον πολυωνυμική Lagrange μέθοδος είναι ένα πολυώνυμο βαθμού μικρότερου ή ίσου με $m - 1$ με τη μορφή: $L(x) = l_1(x)y_1 + \dots + l_m(x)y_m$, με $l_j(x) = \prod_{i=1, \dots, m} ((x - x_i x_j - x_i - 1))$, $i \neq j$ και παρατηρήσεις $y_{1,x_1}, \dots, y_{m,x_m}$. Η παρέκταση $f_{x_0} = L(x_0)$ δημιουργείται από τις m προηγούμενες παρατηρήσεις, δηλαδή, $L(x_0) = l_1(x_0)y_1 + \dots + l_m(x_0)y_m$. Ως εκ τούτου, βασιζόμενη στη μεθοδολογία στην Ενότητα 4.1.3 έχουμε ότι $x(m+l) = x(m) + l \cdot \Delta x(m)$ και εκτιμούμε ότι $y(m+l) = \hat{f}(x(m+l)) = L(x(m+l))$ (βλέπε επίσης Ενότητα 4.1.3).

B. On-line k -means αλγόριθμος

Θυμηθείτε ότι μία τροχιά $\mathbf{k}(n) = [(x(i), y(i))], i = 1, \dots, n$, αποτελείται από n ζεύγη από σημεία θέσης (x, y) . Στον με γνώση κατάσταση $on - line$ $k - means$ αλγόριθμο [20] υπάρχουν $k > 1$ ομάδες (δηλαδή, τροχιές) που τελικά αναπαριστούν την συμπεριφορά της κίνησης του χρήστη. Ο αλγόριθμος διαχειρίζεται νέες τροχιές μεγέθους n μία προς μία και εκτελεί προσαρμογές στην πλησιέστερη ομάδα $\mathbf{k}_i(n), i = 1, \dots, k$. Ο κανόνας προσαρμογής για κάθε νέα τροχιά $\mathbf{k}(n)$ κοντά στην $\mathbf{k}_i(n)$ είναι

$$\mathbf{k}_i(n) = \mathbf{k}_i(n) + \eta \cdot (\mathbf{k}(n) - \mathbf{k}_i(n)) \quad (4.11)$$

Το οποίο σημαίνει ότι, η πλησιέστερη ομάδα κινείται προς την τροχιά $\mathbf{k}(n)$ κατά έναν παράγοντα $\eta \in [0, 1]$. Οι υπόλοιπες ομάδες δεν ενημερώνονται. Αν και ο αλγόριθμος είναι online, ο αριθμός των ομάδων k πρέπει να είναι γνωστός εκ των προτέρων, να είναι σταθερές και ανεξάρτητες από το μέγεθος του n . Ωστόσο, ο αλγόριθμος είναι μάλλον δύσκαμπτος. Στην περίπτωση μας, έχουμε να φτιάξουμε μία $m + l$ πρόβλεψη εφόσον έχουμε καταγράψει μία τροχιά $\mathbf{k}(m)$ μεγέθους m . Ως εκ τούτου, κατασκευάζουμε έναν $k - means$ αλγόριθμο με k ομάδες με $n = m + l$. Καταρχάς, ταξινομούμε την $\mathbf{k}(m)$ τροχιά στην πλησιέστερη ομάδα $\mathbf{k}^*(m + l)$. Αυτό επιτυγχάνεται με το να υπολογίσουμε την Ευκλείδεια απόσταση $d(\cdot, \cdot)$ μεταξύ $\mathbf{k}(m)$ και $\mathbf{k}^*(m)$ (διανύσματα ίσων διαστάσεων), δηλαδή, $\mathbf{k}^*(m + l) = \min\{d(\mathbf{k}(m), \mathbf{k}_i(m)), i = 1, \dots, k\}$. Η προβλεπόμενη θέση του χρήστη $(x(m + l), y(m + l))$ είναι το $(m + l)$ ζεύγος της $\mathbf{k}^*(m + l)$ ομάδας.



Σχήμα 4.18. Ο αριθμός των ομάδων $c(n)$ στο βασισμένο με ART μοντέλο για τις τρεις τροχιές

C. Το βασισμένο στον ART μοντέλο

Το με γνώση κατάσταση προσαρμοστικό ενισχυτικής μάθησης μοντέλο πρόβλεψης θέσης στο [29] υιοθετεί τον ART online αλγόριθμο. Σε αυτόν τον αλγόριθμο νέες τροχιές μεγέθους n (αποτελούμενες από n ζεύγη από (x, y) θέσεις) λαμβάνονται μία προς μία και το μοντέλο ενημερώνεται προοδευτικά. Αυτός ο αλγόριθμος είναι αυξητικός, που σημαίνει ότι αρχικοποιείται με μία ομάδα – τροχιά μεγέθους n – και προσθέτει μία νέα, αν χρειάζεται, σύμφωνα με την κινητή συμπεριφορά του χρήστη. Αρχικά, ο αριθμός των

ομάδων είναι $c(n) = 1$. Αξίζει να σημειωθεί ότι το $c(n)$ εξαρτάται σε μεγάλο βαθμό από το μέγεθος της τροχιάς n , όπως θα δούμε παρακάτω. Δοσμένης μίας τροχιάς $\mathbf{k}(n)$, η Ευκλείδεια απόσταση $d(\mathbf{k}(n), \mathbf{k}_i(n))$ υπολογίζεται για όλες τις τρέχουσες ομάδες $\mathbf{k}_i(n), i = 1, \dots, c(n)$ και η πλησιέστερη στην $\mathbf{k}(n)$ ενημερώνεται. Αν η ελάχιστη απόσταση $d(\mathbf{k}(n), \mathbf{k}^*(n))$ μεταξύ $\mathbf{k}(n)$ και της πλησιέστερης ομάδας $\mathbf{k}^*(n)$ είναι μικρότερη από ότι μία συγκεκριμένη τιμή κατωφλίου, η επονομαζόμενη *vigilance* h , ο κανόνας προσαρμογής είναι ο ίδιος όπως αυτός του *on – line k – means* αλγόριθμου στην Εξίσωση (11). Αλλιώς ένα νέο κέντρο $\mathbf{k}_{c(n)+1}(n)$ που αναπαριστά το αντίστοιχο $\mathbf{k}(n)$ προστίθεται στο μοντέλο και το $c(n)$ αυξάνεται κατά ένα. Είναι ενδιαφέρον να αναφέρουμε ότι το κατώφλι της *vigilance* αναφέρεται στο κριτήριο του να θεωρήσουμε δύο κινητά πρότυπα ισοδύναμα ή όχι κατά τη φάση προσαρμογής του μοντέλου. Στην περίπτωση μας, η παράμετρος *vigilance* h είναι ίση με το κατώφλι του σφάλματος θ . Ο κανόνας προσαρμογής για την πρόβλεψη θέσης στο βασισμένο στον ART μοντέλο είναι

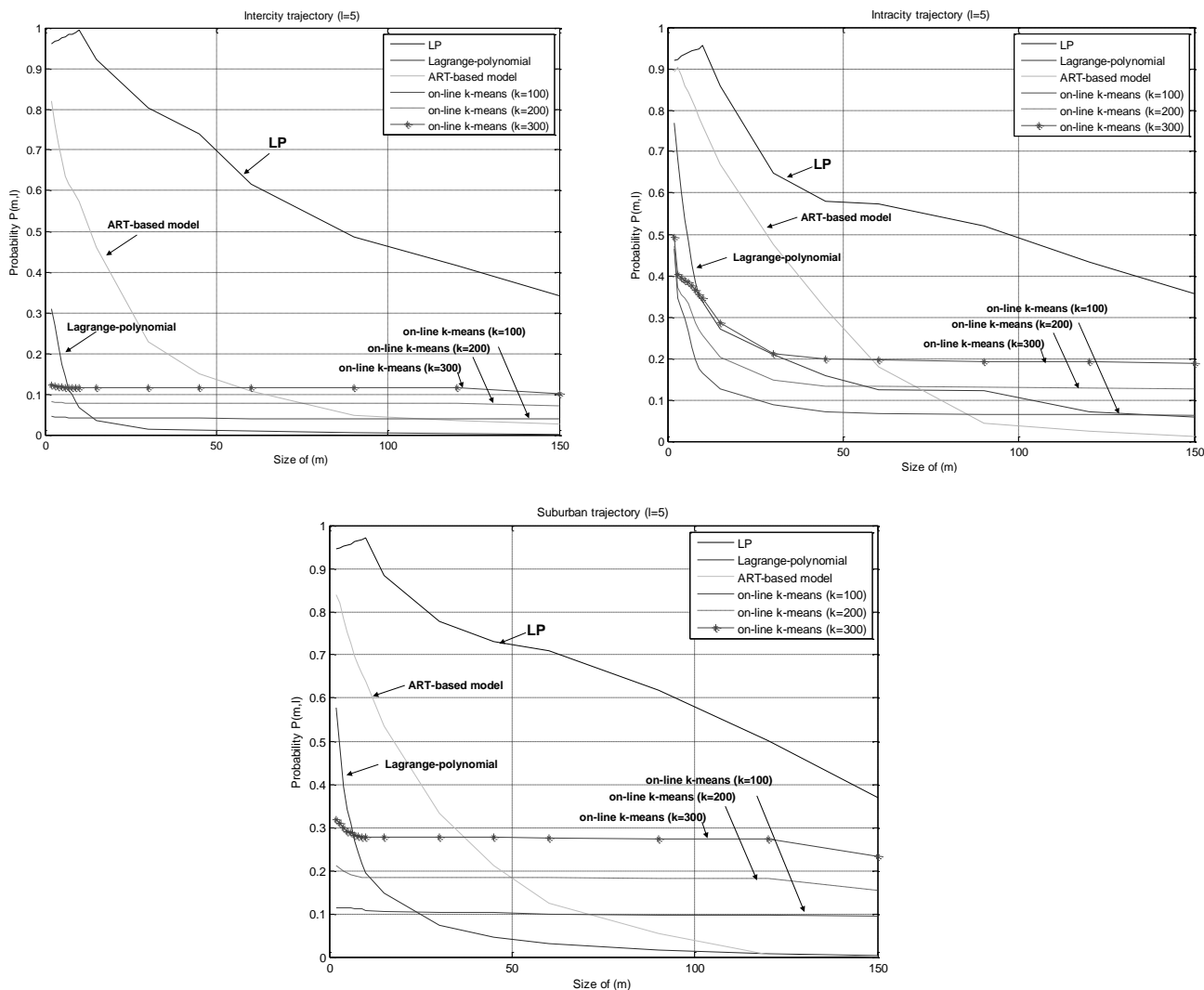
$$\begin{cases} \mathbf{k}_{c(n)+1}(n) \leftarrow \mathbf{k}(n) & \text{if } d(\mathbf{k}^*(n), \mathbf{k}(n)) > h \\ \mathbf{k}^*(n) = \mathbf{k}^*(n) + \eta(\mathbf{k}^*(n) - \mathbf{k}(n)) & \text{otherwise} \end{cases} \quad (4.12)$$

Όμοια με τον *on – line k – means* αλγόριθμο, η προβλεπόμενη θέση του χρήστη $(x(m+l), y(m+l))$ σε μία " $m+l$ " πρόβλεψη είναι το $(m+l)$ – ισοτό ζεύγος της $\mathbf{k}^*(m+l)$ ομάδας. Ωστόσο, σε μία ανεπιτυχή πρόβλεψη, το μοντέλο ενισχύεται να μάθει και προσαρμόζει τα πρότυπα του σε σχέση με την πραγματική τροχιά $\mathbf{k}(m+l)$ του κινητού χρήστη. Το Σχήμα 4.18 απεικονίζει πως ο αριθμός των ομάδων $c(n)$ εξαρτάται από το n στο βασισμένο στον ART μοντέλο για τους τρεις τύπους τροχιών. Παρατηρούμε ότι για μία χαμηλή τιμή του $n = m+l$, με το l σταθερό ($l = 5$), το μοντέλο δημιουργεί νέες ομάδες, με σκοπό να συλλάβει (μάθει) και να προσαρμοστεί στην κίνηση του χρήστη. Ωστόσο, για ένα μεγάλο n (δηλαδή, $n > 100$) το μοντέλο δεν δημιουργεί περισσότερες ομάδες σηματοδοτώντας ότι, το μοντέλο προσαρμόζει τις τρέχουσες ομάδες στην κινητή συμπεριφορά του χρήστη. Σίγουρα, ο αριθμός των ομάδων πρέπει να ληφθεί υπόψη για τη χωρική πολυπλοκότητα αυτής της μεθόδου. Περισσότερες λεπτομέρειες σε αυτό το μοντέλο μπορεί να βρεθούν στο [29].

4.3.4.1. Συγκριτική Εκτίμηση Απόδοσης

Καταρχάς, εξετάζουμε την πιθανότητα μίας επιτυχούς πρόβλεψης $P(m, l)$ που παρατηρείται από τα αμνήμονα και με γνώση κατάστασης μοντέλα για τους τρεις τύπους των τροχιών. Πειραματιστήκαμε με την " $m+l$ " πρόβλεψη με $l = 5$ δευτερόλεπτα, η *vigilance* h και η τιμή του κατωφλίου θ έχουν τεθεί στα 10 μέτρα, και $k = 100, 200$ και 300 στον *on – line k – means* αλγόριθμο. Το Σχήμα 4.19 απεικονίζει την $P(m, l)$ (έναντι του μεγέθους m) που επιτυγχάνεται από τους αλγορίθμους για τις δι-αστικές, αστικές και περιαστικές τροχιές. Παρατηρούμε ότι ο LP αλγόριθμος επιτυγχάνει καλύτερες $P(m, l)$ τιμές από ότι τα άλλα μοντέλα σε όλους τους τύπους των τροχιών, συγκεκριμένα, το Σχήμα 4.19 απεικονίζει την m^* τιμή για τον LP, για την οποία η $P(m^*, l)$ είναι μέγιστη ($m^* = 10$ για $l = 5$). Από την άλλη πλευρά το βασισμένο στον ART μοντέλο επιτυγχάνει καλή ακρίβεια πρόβλεψης, ειδικά, για τιμές του $m < 25$ και για τους τρεις τύπους τροχιών. Συγκεκριμένα, το βασισμένο στον ART μοντέλο επιδεικνύει καλή ακρίβεια πρόβλεψης για την αστική τροχιά (μόνο 10% χαμηλότερη από ότι αυτή του LP) λόγω του γεγονότος ότι ο κινητός χρήστης «επαναλαμβάνει» κάποια πρότυπα κίνησης μέσα στην πόλη. Ωστόσο, για $m > 50$ το βασισμένο στον ART μοντέλο έχει πολύ μικρή $P(m, l)$ και για $m > 100$ δείχνει την χειρότερη απόδοση. Επιπρόσθετα το πολυωνυμικό Lagrange μοντέλο έχει αρκετά καλή ακρίβεια πρόβλεψης όταν το m είναι μικρό. Συγκεκριμένα, έχει $P(m, l) = 0.78$ με $m = 5$ για την δι-αστική τροχιά. Για $m > 10$ το πολυώνυμο Lagrange δεν είναι κατάλληλο για προβλέψεις. Τελικά όλα τα *on – line k –*

means μοντέλα ($k = 100, 200$ και 300), για κάθε m και για όλα τα είδη των τροχιών, επιτυγχάνουν χαμηλή απόδοση σε σχέση με την ακρίβεια πρόβλεψης. Αυτό οφείλεται στο γεγονός ότι ένα τέτοιο μοντέλο δεν είναι ικανό να αυξήσει τον προκαθορισμένο αριθμό των ομάδων. Αντίθετα μπορεί μόνο να τα αναπροσαρμόσει. Αξίζει να σημειωθεί ότι το *on-line k-means* μοντέλο είναι ανεξάρτητο από την τιμή του m , εκτός από την περίπτωση της αστικής τροχιάς, όπου $m < 5$. Αλλά, ακόμα και σε αυτήν την περίπτωση η απόδοση στην ακρίβεια πρόβλεψης είναι χαμηλή.



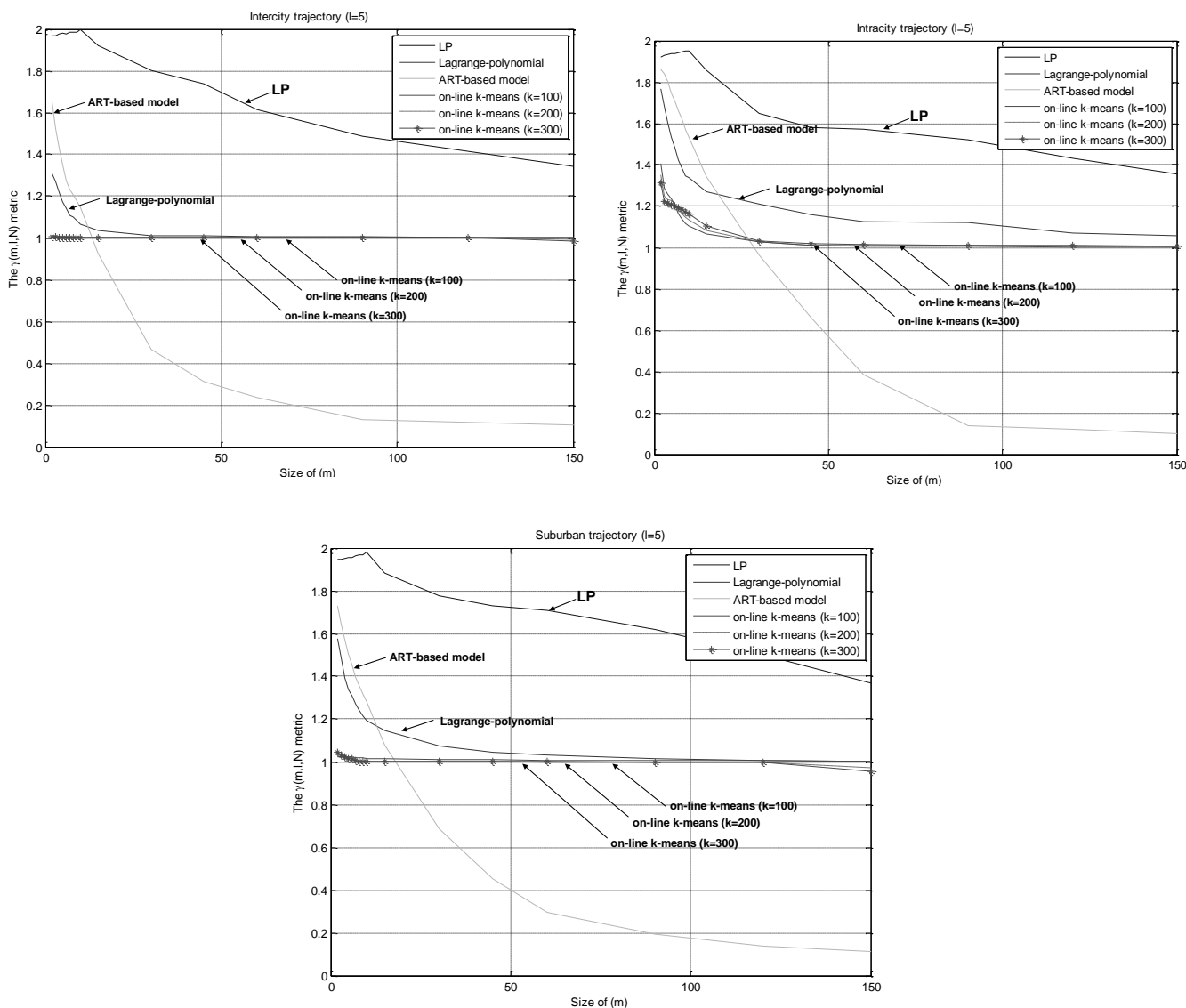
Σχήμα 4.19. Το $P(m, l)$ σε σχέση με το μέγεθος του m για τους διαφορετικούς τύπους τροχιών

Με σκοπό να συγκρίνουμε δίκαια το μοντέλο μας με άλλα αμνήμονα και με γνώση κατάστασης μοντέλα πρέπει να λάβουμε υπόψη μας τις χωρικές απαιτήσεις του κάθε μοντέλου. Γι αυτόν τον λόγο εισάγουμε μία καθολική μετρική γ με σκοπό να συγκρίνουμε τα μοντέλα μέσω της ακρίβειας πρόβλεψης ($P(m, l)$) και τις χωρικής απόδοσης. Για ένα συγκεκριμένο μοντέλο πρόβλεψης, η μετρική γ καθορίζεται ως ακολούθως:

$$\gamma(m, l, N) = P(m, l) + (1 - space/N)$$

Όπου N είναι ο συνολικός όγκος των ιστορικών δεδομένων κίνησης (δηλαδή, το σύνολο των (x, y) ιχνών) και $space$ είναι τα αποθηκευμένα δεδομένα κίνησης που απαιτούνται για την πρόβλεψη από ένα συγκεκριμένο μοντέλο. Στην περίπτωση ενός αμνήμον

μοντέλου, το *space* αναφέρεται στο μέγεθος του m του κυλιόμενου παραθύρου. Στην περίπτωση ενός με γνώση κατάστασης μοντέλου, το *space* είναι ο αριθμός των ομάδων που περιγράφουν την συμπεριφορά κίνησης του χρήστη. Η μετρική $\gamma(m, l, N)$ παίρνει τιμές στο διάστημα $[0, 2]$. Μία τιμή του $\gamma(m, l, N)$ κοντά στο 0 υποδηλώνει ότι το μοντέλο πρόβλεψης δεν είναι κατάλληλο για πρόβλεψη θέσης και οι απαιτήσεις σε χώρο είναι απαγορευτικές. Μία τιμή του $\gamma(m, l, N)$ κοντά στο 2 δηλώνει ότι το μοντέλο έχει υψηλή απόδοση στην ακρίβεια πρόβλεψης και παράλληλα απαιτεί λίγο χώρο. Συγκεκριμένα, στην περίπτωση του LP και του πολυωνύμου Lagrange, ο παράγοντας *space* στο $\gamma(m, l, N)$ είναι m/N . Για το βασισμένο στον ART μοντέλο και τα *on-line k-means* μοντέλα, το *space* είναι $c(m + l) \cdot (m + l)$ και $k \cdot (m + l)$, αντίστοιχα. Το Σχήμα 4.20 απεικονίζει τη μετρική $\gamma(m, l, N)$ έναντι του μεγέθους του m για την δι-αστική, αστική και περιαστική τροχιά, αντίστοιχα.



Σχήμα 4.20. Η $\gamma(m, l, N)$ μετρική σε σχέση με το μέγεθος του m για διαφορετικούς τύπους τροχιών

Στο Σχήμα 4.20 παρατηρούμε κάποια ενδιαφέροντα χαρακτηριστικά για κάθε μοντέλο. Καταρχάς σε όλους τους τύπους των τροχιών μπορούμε να παρατηρήσουμε τα ακόλουθα: (a) το LP μοντέλο έχει την υψηλότερη τιμή του γ (πολύ κοντά στο 2 για $m^* = 10$) και το γ προσεγγίζει το 1.38 για μεγάλο m , (b) το βασισμένο στον ART

μοντέλο έχει την ελάχιστη τιμή του γ για $m > 15$ και παρουσιάζει παρόμοια συμπεριφορά με το πολυώνυμο Lagrange για $m < 10$, (c) η τιμή γ για το πολυώνυμο Lagrange προσεγγίζει το 1 για $m > 10$, και, (d) η τιμή γ για όλα τα *on-line k-means* μοντέλα είναι κοντά στο 1 ανεξάρτητα από το m .

Επιπρόσθετα, στην περίπτωση της αστικής τροχιάς, ο LP, το βασισμένο στον ART και το πολυωνυμικό Lagrange μοντέλο παρουσιάζουν καλή απόδοση σε σχέση με το γ έχοντας $m \leq 5$. Στην περίπτωση των δι-αστικών τροχιών, ο LP (για $m = m^*$) και το βασισμένο στον ART μοντέλο (για $m = 2$) θεωρούνται τα πιο κατάλληλα για πρόβλεψη θέσης. Όμοια συμπεριφορά για τον LP, το βασισμένο στον ART και το πολυωνυμικό Lagrange μοντέλο παρατηρείται και στην περιαστική τροχιά. Συνολικά, το LP μοντέλο είναι εφαρμόσιμο σε κάθε είδους τροχιά σε σχέση με τη μετρική γ και $m^* = 10$ (για $l = 5$). Ως εκ τούτου, για $l = 5$, η τιμή του m είτε για το μέγεθος της ομάδας (στα με γνώση κατάστασης μοντέλα) ή για το μήκος του κυλιόμενου παραθύρου (m^* στα αμνήμονα μοντέλα) δεν είναι υψηλή. Έτσι, το βασισμένο στον ART μοντέλο επίσης θεωρείται κατάλληλο για πρόβλεψη θέσης. Ωστόσο, πρέπει επίσης να λάβουμε υπόψη μας το υπολογιστικό κόστος για την πρόβλεψη, όπως θα αναφερθεί στην επόμενη ενότητα.

4.3.4.2. Αρχικοποίηση και Πρόβλεψη Υπολογιστικού Κόστους

Σε αυτήν την ενότητα αναλύουμε το υπολογιστικό κόστος για κάθε μοντέλο πρόβλεψης. Η τοπική (σταθμισμένη με πυρήνα) γραμμική παλινδρόμηση του προτεινόμενου LP είναι μία m βασισμένη σε κυλιόμενο παράθυρο μέθοδος, στην οποία το μοντέλο προσαρμογής $f(X)$ δημιουργείται κατά τον χρόνο της πρόβλεψης (βλέπε Ενότητα 4.1.1). Η τοπικά σταθμισμένη παλινδρόμηση λύνει ένα ξεχωριστά σταθμισμένο πρόβλημα ελαχίστων τετραγώνων στο σημείο στόχος $x_0 \in X$. Η προσαρμογή των ελαχίστων τετραγώνων συνήθως εκτελείται υιοθετώντας την Cholesky αποσύνθεση. Συγκεκριμένα, με m παρατηρήσεις και $p = 1$ (Ενότητα 4.1.1), η κατασκευή του f μοντέλου χρησιμοποιώντας Cholesky αποσύνθεση απαιτεί $O(m)$. Επιπρόσθετα, το κόστος για τον υπολογισμό του $f(x_0)$ είναι $O(m)$. Ως εκ τούτου, για το προτεινόμενο μοντέλο το συνολικό υπολογιστικό κόστος για την πρόβλεψη είναι $O(m)$. Ωστόσο, δημοφιλείς υλοποιήσεις της τοπικής παλινδρόμησης, όπως η συνάρτηση *loess* στο S-PLUS, χρησιμοποιεί συστήματα τριγωνισμού για να μειώσουν τους υπολογισμούς [30]. Τελικά, η παράμετρος λ για την συνάρτηση πυρήνα τυπικά καθορίζεται off-line, για παράδειγμα χρησιμοποιώντας διασταυρωμένη επικύρωση (cross-validation), σε $O(m^2)$.

Το υπολογιστικό κόστος για το πολυώνυμο Lagrange $L(x_0)$ δεδομένου ότι $L(x)$ είναι βαθμού λιγότερου ή ίσου με $m - 1$ είναι $O(m)$. Επιπρόσθετα, οι συντελεστές $l_j(x)$ του $L(x)$ πολυώνυμου κατασκευάζονται σε $O(m^2)$. Έτσι, το υπολογιστικό κόστος για την πρόβλεψη χρησιμοποιώντας πολυωνυμική Lagrange παρέκταση στο x_0 είναι $O(m^2)$. Επιπρόσθετα, το υπολογιστικό κόστος για το μοντέλο ταξινόμησης στον *on-line k-means* αλγόριθμο βασίζεται στον αριθμό (k) των ομάδων και στο μέγεθος ($m + l$) των ομάδων. Το υπολογιστικό κόστος για να αρχικοποιηθεί ο *on-line k-means* αλγόριθμος με k ομάδες και μεγέθους $m + l$ και με αρχικό σύνολο S των τροχιών (κάθε μία από m σημεία θέσης) είναι $O(k|S|)$. Για ένα παρατηρούμενο πρότυπο m θέσεων, το κόστος πολυπλοκότητας ταξινόμησης είναι $O(mk)$ (συμπεριλαμβανομένης της προσαρμογής της πλησιέστερης ομάδας στο παρατηρούμενο πρότυπο). Επιπρόσθετα, το υπολογιστικό κόστος για το βασισμένο στον ART μοντέλο πρόβλεψης στην αρχή της διαδικασίας είναι $O(1)$ δεδομένου ότι το μοντέλο ξεκινάει με μία ομάδα μεγέθους $m + l$. Το κόστος για την ενημέρωση του μοντέλου με μία νέα ομάδα και για την προσαρμογή ενός νέου προτύπου στο μοντέλο είναι επίσης $O(1)$. Το υπολογιστικό κόστος για την ταξινόμηση και/ή προσαρμογή ενός παρατηρούμενου προτύπου με μέγεθος m στο

μοντέλο εξαρτάται από το μέγεθος $m + l$ των ομάδων και τον τρέχον αριθμό των ομάδων $c(m + l)$. Έτσι, το υπολογιστικό κόστος για την πρόβλεψη είναι $O(m \cdot c(m + l))$. Θα έπρεπε να σημειωθεί ότι ο αριθμός των ομάδων $c(m + l)$ εξαρτάται όχι μόνο από το m αλλά επίσης και από την πολιτική προσαρμογής του μοντέλου λόγω της παραμέτρου *vigilance* h .

Πίνακας 4.4. Σύγκριση των μοντέλων πρόβλεψης

Μετρική/Μοντέλο	LP	Lagrange-polynomial model	ART-based model	on-line k -means model ($k=100, 200, 300$)
χωρική πολυπλοκότητα	++	++	--	-
πιθανότητα επιτυχούς πρόβλεψης $P(m, l)$	++	-	++	+
Καθολική μετρική $\gamma(m, l, N)$	++	-	+ (για χαμηλό m) -- (για υψηλό m)	-
υπολογιστικό κόστος	++	+	--	-

Ο Πίνακας 4.4 συνοψίζει τα συμπεράσματα μας σε σχέση με την συγκριτική αξιολόγηση των μοντέλων που συζητήθηκαν στην παρούσα ενότητα. Συγκρίνοντας τα μοντέλα, μπορούμε καθαρά να ξεχωρίσουμε το LP μοντέλο ως τον πιο αποδοτικό προγνώστη για πρόβλεψη θέσης για όλους τους τύπους των τροχιών.

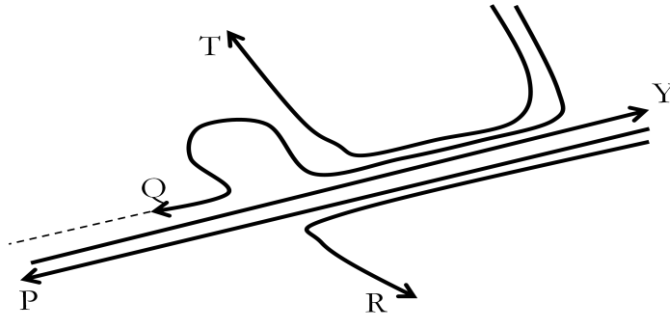
5. ΕΞΥΠΝΗ ΤΑΞΙΝΟΜΗΣΗ ΤΡΟΧΙΑΣ ΓΙΑ ΒΕΛΤΙΩΜΕΝΗ ΠΡΟΒΛΕΨΗ ΚΙΝΗΣΗΣ

5.1. Μοντέλο Δεδομένων

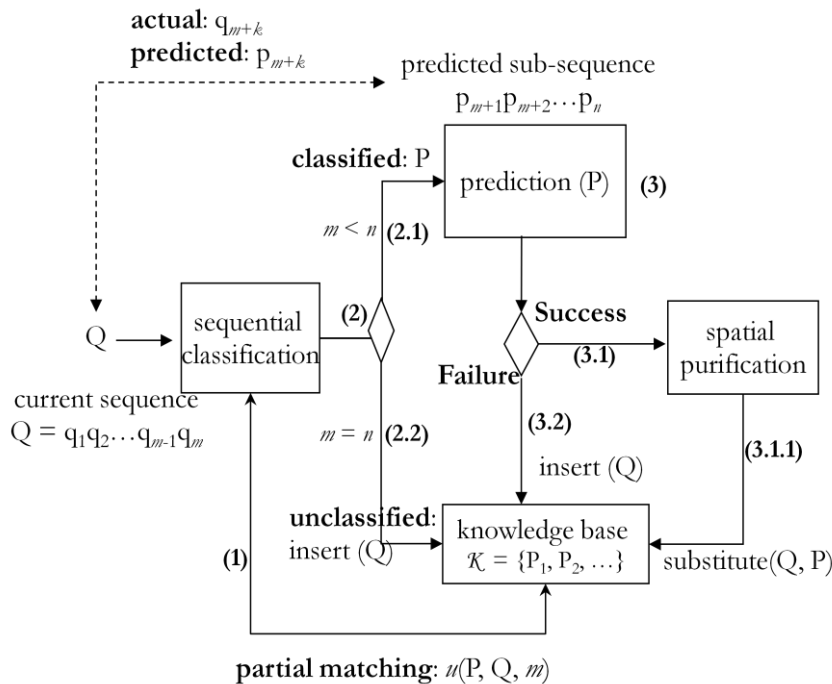
Σε αυτήν την ενότητα παρέχουμε έναν ορισμό για την τροχιά ενός κινούμενου αντικειμένου και εισάγουμε μία συνάρτηση απόστασης τροχιάς για το σύστημα μας.

5.1.1. Αναπαράσταση Τροχιάς

Μία τροχιά είναι μία σειρά από δύο διαστάσεων (δηλαδή, 2D) σημεία. Υποδηλώνεται ως $P = p_1 p_2 \dots p_n$ και έχει μήκος $n > 0$. Εδώ, το p_i ($1 \leq i \leq n$) είναι ένα 2D σημείο., το οποίο αναπαριστά την θέση (γεωγραφικό μήκος, πλάτος) στο επίπεδο, δηλαδή, $p_i = (x_i, y_i)$. Μία τροχιά $p_{k_1} p_{k_2} \dots p_{k_m}$ ($1 \leq k_1 < k_2 < \dots < k_m \leq n$) καλείται μία *υπό-τροχιά* της P .



Σχήμα 5.1. Οι P, Q, R, Y και T τροχιές του παραδείγματος



Σχήμα 5.2. Το σύστημα: η ακολουθιακή διαδικασία ταξινόμησης, η διαδικασία πρόβλεψης κινούμενης ακολουθίας, η διαδικασία μείωσης χωρικής διακύμανσης, και η βάση γνώσης K.

Μία τροχιά P αναπαριστά μία σειρά από n GPS εγγραφές ενός GPS δέκτη που είναι τοποθετημένος σε ένα όχημα ή σε κάποιο άλλο κινούμενο αντικείμενο. Το κινούμενο αντικείμενο είναι στη θέση p_i στο χρόνο t_i . Υποθέτουμε ότι το αντικείμενο κινείται κατά μήκος μίας ευθείας γραμμής, με σταθερή ταχύτητα, από το p_i στο p_{i+1} κατά τη χρονική διάρκεια $[t_i, t_{i+1}]$.

Μία αντιπροσωπευτική τροχιά, ή *πρότυπο*, είναι μία τροχιά, η οποία υποδηλώνει την κύρια συμπεριφορά του κινητού αντικείμενου. Ένα πρότυπο (α) αποθηκεύεται στη βάση γνώσης (K) του συστήματος, (β) χρησιμοποιείται για την επίτευξη προβλέψεων κίνησης, (γ) μπορεί να αντικατασταθεί από μία άλλη τροχιά. Όλα τα πρότυπα στην K έχουν το ίδιο μήκος.

Στο Σχήμα 5.2 αναφέρουμε τα βασικά βήματα του συστήματος της ταξινόμησης τροχιάς και της πρόβλεψης θέσης. Μέσω αυτών των βημάτων θα διευκρινιστεί η εισαγωγή μίας συνάρτησης απόστασης τροχιάς που χρησιμοποιείται για σειριακή ταξινόμηση. Υποθέτουμε μία K η οποία είναι ένα σύνολο από έναν καθορισμένο αριθμό προτύπων. Κάθε πρότυπο στην K έχει μήκος n . Ενώ το κινητό αντικείμενο ταξιδεύει, η προηγούμενη σειρά από $m - 1$ θέσεις ($q_1 q_2 \dots q_{m-1}$) μαζί με την τρέχουσα θέση q_m ($m \leq n$), δηλαδή, η $Q = q_1 q_2 \dots q_m$, είναι εν μέρει αντιστοιχισμένη με όλα τα πρότυπα στην K . Αν ένα πρότυπο $P = p_1 p_2 \dots p_n$ είναι σημαντικά όμοιο με την τρέχουσα σειρά Q , το σύστημα προβλέπει ότι το κινητό αντικείμενο θα ταξιδέψει κατά μήκος αυτού του προτύπου. Αυτό σημαίνει ότι η προβλεπόμενη σειρά των θέσεων που θα βρεθεί το κινούμενο αντικείμενο στο κοντινό μέλλον είναι η σειρά $p_{m+1} p_{m+2} \dots p_n$.

5.1.2. Απόσταση Τροχιάς

Σημαντική έρευνα έχει γίνει πάνω στις συναρτήσεις απόστασης τροχιών. Μέθοδοι αντιστοίχισης που βασίζονται σε αντιστοίχιση συμβολοσειρών για «ακατέργαστες» συμβολοσειρές (δηλαδή, χωρίς αλλαγή στην αναπαράσταση όπως η εκπροσώπηση γωνίας περιστροφής [31] και υπογραφές καμπύλης [32]) έχει αποδειχτεί ότι είναι επιτυχείς. Αντιπροσωπευτικές μέθοδοι είναι η απόσταση Δυναμικής Αναδίπλωσης Χρόνου (DTW) [33], η Ευκλείδεια απόσταση για τις τροχιές [34], η απόσταση Μεγαλύτερης Κοινής Ακολουθίας (LCS) [35], οι αποστάσεις Levenshtein και Hausdorff για τις τροχιές [36], [37]. Οι DTW και η Ευκλείδεια μετρικές αποστάσεων είναι σχετικά ευαίσθητες στο θόρυβο. Τέτοιες μετρικές δεν λαμβάνουν υπόψη τους τα πιο ανόμοια μέρη των τροχιών. Ωστόσο, είναι χρήσιμες επειδή αναπαριστούν ένα ακριβές μοντέλο της ανθρώπινης αντίληψης, δεδομένου ότι όταν συγκρίνουμε οποιοδήποτε είδος δεδομένων (δηλαδή, τροχιές), εστιάζουν κυρίως στα μέρη που είναι όμοια και παραβλέπουν περιοχές μεγάλης ομοιότητας. Η LCS μετρική προήλθε από το πεδίο των αλγορίθμων αντιστοίχισης συμβολοσειρών και επιστρέφει το μήκος της μεγαλύτερης κοινής ακολουθίας που αντιστοιχείται από δύο συμβολοσειρές [38]. Η βασική ιδέα της LCS είναι να αντιστοιχίσει δύο συμβολοσειρές επιτρέποντας τους να «εκταθούν», χωρίς αναδιάταξη της ακολουθίας των σημείων αλλά επιτρέποντας κάποια σημεία να είναι μη-αντιστοιχισμένα.

Ορίζουμε μία συνάρτηση απόστασης η οποία αξιολογεί την ομοιότητα μεταξύ ενός προτύπου P και του τρέχοντος Q . Η συνάρτηση απόστασης είναι ο βασικός τελεστής για την αντιμετώπιση (i) της (μερικής) αντιστοίχισης των Q και P , (ii) την ταξινόμηση του Q ως P , και (iii) την πρόβλεψη της μελλοντικής κίνησης της ακολουθίας βασιζόμενη σε μία υπο-τροχιά του P . Η προτεινόμενη συνάρτηση απόστασης της τροχιάς μεταξύ του Q και κάποιου προτύπου P στην K είναι, ωστόσο, πολύ συγκεκριμένη με την σειριακή ταξινόμηση και την πρόβλεψη της ακολουθίας κίνησης. Μας ενδιαφέρει να αξιολογήσουμε την ομοιότητα μεταξύ των Q και P βασιζόμενοι στην μερική αντιστοίχιση του $Q = q_1 \dots q_m$ με την αντίστοιχη υπο-τροχιά του $P = p_1 \dots p_m$. Αυτό γίνεται διότι η συνάρτηση απόστασης περιορίζεται στο να βρει ένα πρότυπο P από την K το οποίο να περιλαμβάνεται σε μία αρκετά κοντινή περιοχή με αυτή του Q . Έτσι, στην περίπτωση μας, όταν το Q ταξινομείται ως P τότε ένα μέρος του P θα χρησιμοποιηθεί για να κάνουμε προβλέψεις. Αυτό περιορίζει την προτεινόμενη μετρική να είναι αναλλοίωτη από μετάφραση (translation invariant), το οποίο υποστηρίζεται από την LCS μετρική

απόστασης και την Ευκλείδεια μετρική χρονικής κατανομής. Επιπρόσθετα, αφού ασχολούμαστε με πραγματικού χρόνου αποφάσεις πρόβλεψης, μία συνεχής μερική αντιστοίχιση του τρέχοντος Q με ένα πρότυπο από την K μπορεί να είναι υπολογιστικά αποδοτική. Η προτεινόμενη συνάρτηση απόστασης είναι απλή, με γραμμική πολυπλοκότητα (βλέπε παρακάτω), συγκρινόμενη με άλλες μετρικές υψηλής υπολογιστικής πολυπλοκότητας (δηλαδή, την LCS μετρική απόστασης και την Ευκλείδεια μετρική χρονικής κατανομής). Επίσης, η σειρά των σημείων στο Q είναι υψίστης σημασίας στην δική μας διαδικασία σειριακής ταξινόμησης. Οι τροχιές με αντίστροφη σειρά από σημεία (δηλαδή, κινήσεις με αντίθετες κατευθύνσεις), δεν μπορούν να θεωρηθούν όμοιες ακόμα και αν επιδεικνύουν όμοια πρότυπα κίνησης. Αυτό πρέπει να ληφθεί υπόψη στην προτεινόμενη συνάρτηση απόστασης δεδομένου ότι η πρόβλεψη ακολουθίας βασίζεται στην υπο-ακολουθία του κοντινότερου προτύπου P . Τελικά, ασχολούμαστε με τροχιές πάνω στο επίπεδο, έτσι, δεν λαμβάνουμε υπόψη μας οποιαδήποτε χρονική μετάφραση (time translation) στις τροχιές για τον υπολογισμό της απόστασης τους.

Αναφέρουμε μία λίστα από τα χαρακτηριστικά που η προτεινόμενη συνάρτηση υποστηρίζει. Αξίζει να σημειωθεί ότι ένα μέρος αυτών των χαρακτηριστικών υποστηρίζονται από τις προαναφερόμενες μετρικές. Ωστόσο, μερικές μετρικές παρέχουν επιπρόσθετα χαρακτηριστικά που περιορίζονται στην περίπτωση μας όπως αναφέρθηκε παραπάνω (δηλαδή, αναλλοίωτη μετάφραση, ευαισθησία στο θόρυβο, υπολογιστική αναποτελεσματικότητα). Η προτεινόμενη συνάρτηση απόστασης:

- Παίρνει πραγματικές τιμές στο $[0, 1]$. Μία μηδενική τιμή υποδηλώνει όμοιες τροχιές, ενώ η μονάδα υποδηλώνει ανόμοιες τροχιές,
- Λαμβάνει υπόψη της τροχιές διαφορετικού μήκους,
- Εξαρτάται μόνο από τις m πιο πρόσφατες θέσεις, δηλαδή, λαμβάνει υπόψη την σχέση της ακολουθίας $q_1q_2 \dots q_m$ και $p_1p_2 \dots p_m$, $p_i \in P$ και $q_i \in Q$,
- Θεωρεί πολύ σημαντικές τις τιμές απόστασης που αναφέρονται στις πιο πρόσφατες κινήσεις από ότι στις πιο παλιές κινήσεις. Το οποίο σημαίνει ότι η τιμή της απόστασης μεταξύ της ακολουθίας $q_1q_2 \dots q_m$ και της ακολουθίας $p_1p_2 \dots p_m$ είναι πιο σημαντική από την τιμή της απόστασης μεταξύ των ακολουθιών $q_1q_2 \dots q_l$ και $p_1p_2 \dots p_l$ με $l < m$,
- Είναι αθροιστική, δηλαδή, όσο πιο όμοιες είναι οι ακολουθίες $q_1q_2 \dots q_l$ και $p_1p_2 \dots p_l$, τόσο πιο όμοιες είναι και οι ακολουθίες $q_1q_2 \dots q_m$ και $p_1p_2 \dots p_m$ με $l < m$,
- Ασχολείται με χωρικά κοντινά πρότυπα κίνησης. Αυτό σημαίνει ότι οι Q και P πρέπει να αναφέρονται σε τροχιές που περιλαμβάνονται σε σχετικά κοντινές περιοχές (στο επίπεδο), αφού η προβλεπόμενη σειρά εξαρτάται υψηλά από το κοντινότερο πρότυπο P ,
- Χειρίζεται θορυβώδη πληροφορία θεωρώντας όμοιες κινήσεις κάτω από συγκεκριμένο χωρικό κατώφλι,
- Είναι υπολογιστικά αποδοτική.

Η προτεινόμενη συνάρτηση απόστασης έχει το ακόλουθο μειονέκτημα: Δεν μπορεί να χειριστεί διαφορετικής συχνότητας δειγματοληψίας σημεία. Για παράδειγμα δύο τροχιές που κινούνται στην ίδια ακριβώς κατεύθυνση, αλλά η μία κινείται με διπλάσια ταχύτητα από ότι η άλλη, θα οδηγήσει σε σημαντική διαφορετική τιμή απόστασης. Σε αυτήν την εργασία, υποθέτουμε ότι η ταχύτητα του κινούμενου αντικειμένου είναι σχετικά όμοια πάνω στην ίδια περιοχή (δηλαδή, καθημερινές κινήσεις με όχημα σε κάποιους δρόμους). Μια τέτοια λειτουργικότητα υποστηρίζεται από την LCS μετρική απόστασης.

Θεωρείστε τις τροχιές $P = p_1p_2 \dots p_n$ και $Q = q_1q_2 \dots q_m$ με $m \leq n$. Ορίζουμε την συνάρτηση απόστασης σημείου $f(P, Q, i)$ μεταξύ του i -οστού σημείου του P , p_i , και του i -οστού σημείου του Q , q_i , ($1 \leq i \leq m$), ως ακολούθως

$$f(P, Q, i) = \begin{cases} 1 & \text{αν } \|p_i - q_i\| \leq \epsilon \\ 0 & \text{αλλιώς} \end{cases} \quad (5.1)$$

Για κάποιο χωρικό κατώφλι $\epsilon > 0$. $\|p_i - q_i\|$ είναι η Ευκλείδεια απόσταση μεταξύ του p_i και του q_i .

Η προτεινόμενη συνάρτηση απόστασης παρέχει έναν μηχανισμό στάθμισης όταν αξιολογείται η ομοιότητα μεταξύ των P και Q . Οι δύο τροχιές είναι πολύ όμοιες όταν έχουν ένα μεγάλο μερίδιο από κοντινά σημεία σε σχέση με ένα καθορισμένο ϵ . Συγκεκριμένα, οι $f(P, Q, \cdot)$ τιμές σε κάποια σημεία στο μέσο των τροχιών είναι πιο σημαντικές για την αξιολόγηση της ομοιότητας μεταξύ των P και Q από ότι οι $f(P, Q, \cdot)$ τιμές σε κάποια σημεία στην αρχή των τροχιών. Επιπρόσθετα, η $f(P, Q, m)$ τιμή είναι περισσότερο σημαντική από ότι οποιαδήποτε άλλη $f(P, Q, k)$ τιμή με $k < m$. Αυτό εκφράζεται μέσω μίας αυξανόμενης ακολουθίας από βάρη $w_i > 0$ ($1 \leq i \leq m$). Το w_i βάρος, που αντιστοιχίζεται στην $f(P, Q, i)$ τιμή, είναι μεγαλύτερο από ότι το w_j βάρος, που αντιστοιχίζεται στην $f(P, Q, j)$ τιμή, με $j < i$. Θέτουμε $w(i) = i$. Ως εκ τούτου, η μερίδα των κοντινών σημείων, που σχετίζονται με μία ακολουθία κίνησης Q (ενός κινούμενου αντικειμένου) από το σημείο q_1 μέχρι το σημείο q_k , είναι τότε το κλάσμα $\frac{\sum_{i=1}^k f(P, Q, i)w(i)}{\sum_{i=1}^k w(i)}$.

Η ομοιότητα μεταξύ των P και Q σχετίζεται επίσης με έναν συντηρητικό παράγοντα στην μερίδα των κοντινών σημείων. Μία μεγάλη μερίδα από κοντινά σημεία σε ένα πρώιμο στάδιο του υπολογισμού της απόστασης ίσως οδηγήσει σε μία πρόωρη απόφαση σχετικά με την ομοιότητα μεταξύ των τροχιών. Εισάγουμε μία αυξανόμενη ακολουθία από συντηρητικούς παράγοντες $g(i) \in (0, 1]$ η οποία υποτιμά την τιμή της μερίδας των κοντινών σημείων μέχρι και την i -οστή κίνηση. Θέτουμε το $g(i) = \frac{i}{i+1}$, ($1 \leq i \leq m$). Ως εκ τούτου, η συνάρτηση απόστασης μεταξύ των P και Q , μεγέθους n και m , αντίστοιχα, είναι η ποσότητα

$$u(P, Q, m) = 1 - g(m) \frac{\sum_{i=1}^m f(P, Q, i)w(i)}{\sum_{i=1}^m w(i)} \quad (5.2)$$

Η $u(P, Q, m)$ τιμή υποδηλώνει πόσο όμοια είναι η ακολουθία $Q = q_1 q_2 \dots q_m$ με την $P = p_1 p_2 \dots p_m$ βασιζόμενη πάνω στις προηγούμενες κινήσεις μέχρι και την m -οστή κίνηση. Αν αυτή η τιμή δεν θεωρηθεί σημαντική για την ταξινόμηση του τρέχοντος Q ως P (και να κάνει προβλέψεις βασιζόμενη στην P) τότε το σύστημα συνεχίζει με την επόμενη κίνηση του κινούμενου αντικειμένου και επαναξιολογεί την $u(P, Q', m+1)$ τιμή με $Q' = q_1 q_2 \dots q_m q_{m+1}$.

Ο υπολογισμός της $u(P, Q, m)$ τιμής βασίζεται στον υπολογισμό της $u(P, Q, m-1)$ της προηγούμενης φάσης ξεκινώντας από $u(P, Q, 1) = 1 - g(1)f(P, Q, 1)$. Η υπολογιστική πολυπλοκότητα της προτεινόμενης συνάρτησης απόστασης $u(P, Q, m)$ είναι $O(m)$ σε χρόνο.

5.2. Ακολουθιακή Διαδικασία Ταξινόμησης

Για να ταξινομήσουμε μία ακολουθία Q σε ένα πρότυπο $P \in K$, χρειάζεται να γίνουν παρατηρήσεις πάνω στην κίνηση του κινούμενου αντικειμένου. Το ερώτημα που ανακύπτει είναι πόσες πολλές παρατηρήσεις m της ακολουθίας $Q = q_1 \dots q_m$ απαιτούνται για να ταξινομηθεί η Q ως P . Θεωρούμε ότι η $K = \{P_1, \dots, P_N\}$ με N πρότυπα

μεγέθους n . Ας θεωρήσουμε επίσης και την m -ιστή παρατήρηση της Q που αντιστοιχεί σε ένα κινούμενο αντικείμενο. Μετά την m -ιστή παρατήρηση, ο ταξινομητής υπολογίζει την απόσταση $u(P, Q, m)$ της τρέχουσας Q με κάθε πρότυπο $P \in K$ και παίρνει την ελάχιστη απόσταση

$$u_m^* = \min_{P \in K} u(P, Q, m)$$

Μία απόφαση με βάση το u_m^* περιλαμβάνει:

- Την επιλογή να τερματιστεί η ακολουθία των παρατηρήσεων και να ταξινομηθεί η Q στο πλησιέστερο P^* στο m -ιστό στάδιο, δηλαδή,

$$P^* = \arg \min_{P \in K} u(P, Q, m)$$
- Την επιλογή να ληφθεί η $(m + 1)$ -ιστή παρατήρηση της κίνησης του κινητού αντικειμένου πριν να επιχειρηθεί μία ταξινόμηση.

Αξίζει να σημειωθεί ότι το P^* δεν είναι απαραίτητα το ίδιο σε όλες τις παρατηρήσεις. Αυτό εξαρτάται σε μεγάλο βαθμό από την συμπεριφορά κίνησης του κινούμενου αντικειμένου.

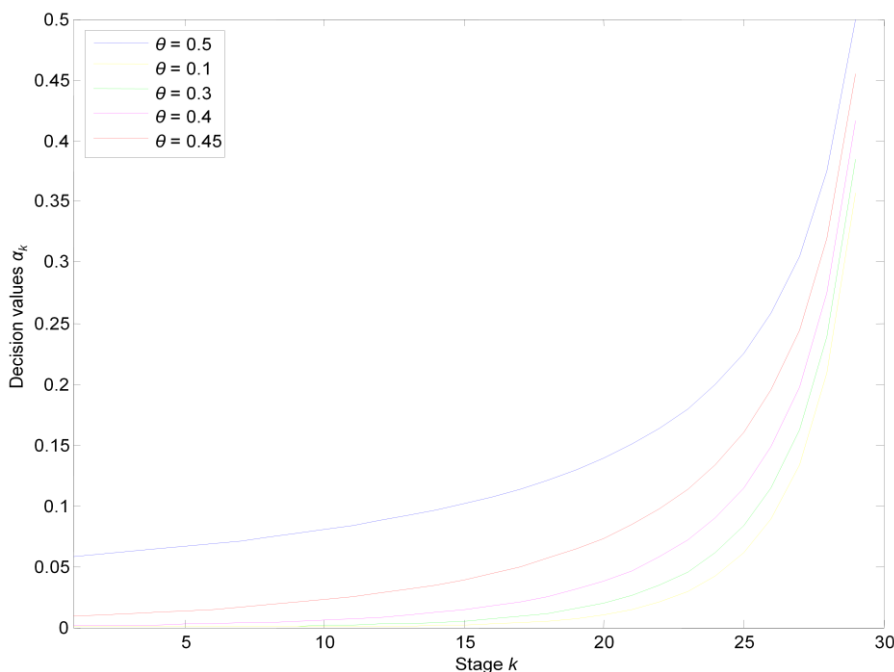
5.2.1. Θεωρία Βέλτιστης Παύσης

Η OST (Optimal Stopping Theory – Θεωρία Βέλτιστης Παύσης) σχετίζεται με το πρόβλημα της επιλογής της καλύτερης χρονικής στιγμής για να πάρουμε την απόφαση να εκτελέσουμε μία συγκεκριμένη ενέργεια. Αυτή η απόφαση βασίζεται στις ακολουθιακές παρατηρούμενες τυχαίες μεταβλητές με σκοπό να ελαχιστοποιήσει τον αναμενόμενο κίνδυνο ή κόστος (ή να μεγιστοποιήσει την αναμενόμενη εξόφληση) [39]. Τα προβλήματα βέλτιστου τερματισμού χαρακτηρίζονται από την διαθεσιμότητα, σε κάθε στάδιο της διαδικασίας, μίας οντότητας ελέγχου που τερματίζει την εξέλιξη της διαδικασίας. Σε κάθε στάδιο η οντότητα ελέγχου παρατηρεί την παρούσα κατάσταση του συστήματος και αποφασίζει εάν θα συνεχίσει (πιθανώς με κάποιο κόστος) ή εάν θα τερματίσει τη διαδικασία και θα προβεί σε μία ορισμένη απώλεια ή έκπτωση. Ο καλύτερος χρόνος, κατά τον οποίο η οντότητα τερματίζει τη διαδικασία, είναι ο χρόνος βέλτιστου τερματισμού.

Υπάρχουν πολλά προβλήματα με βέλτιστες λύσεις που παρέχονται μέσω της OST [40]. Σε ένα πρόβλημα κανόνα τερματισμού, η οντότητα παρατηρεί μία ακολουθία X_1, X_2, \dots από ανεξάρτητες και ομοιόμορφα κατανομημένες (i.i.d.) τυχαίες μεταβλητές, κάθε μία με αθροιστική συνάρτηση κατανομής $f(x) = Prob(X \leq x)$ και $E\{X\} < \infty$. Σε κάθε στάδιο $m = 1, 2, \dots$ αφού παρατηρηθούν οι x_1, x_2, \dots, x_m τιμές, η οντότητα μπορεί να συνεχίσει και να παρατηρήσει την x_{m+1} . Ο κανόνας βέλτιστου τερματισμού είναι να τερματιστεί η διαδικασία σε κάποιο στάδιο m με σκοπό να ελαχιστοποιηθεί ο αναμενόμενος κίνδυνος [34]. Ένα πρόβλημα βέλτιστου τερματισμού έχει έναν πεπερασμένο ορίζοντα εάν υπάρχει ένα γνωστό άνω όριο στον αριθμό των σταδίων στα οποία η οντότητα τερματίζει. Τέτοια προβλήματα μπορούν να λυθούν με την μέθοδο της «backward induction».

- 1) *Βέλτιστο Πρόβλημα Τερματισμού αγοράς-αγαθού*: Θεωρείστε ότι μία συγκεκριμένη ποσότητα από πρώτη ύλη απαιτείται σε έναν συγκεκριμένο χρόνο. Αν η τιμή αυτού του υλικού αυξομειώνεται, τότε εγείρεται το πρόβλημα της απόφασης εάν θα αγοράσουμε στην παρούσα τιμή ή εάν θα περιμένουμε μία περαιτέρω περίοδο, κατά τη διάρκεια της οποίας η τιμή μπορεί να αυξηθεί ή να μειωθεί. Ένας αγοραστής (οντότητα ελέγχου), ως εκ τούτου, θέλει να ελαχιστοποιήσει την αναμενόμενη τιμή της αγοράς. Η προσφορά της τιμής έρχεται κάθε μέρα, x_m δηλώνοντας την τιμή που λαμβάνεται την ημέρα m , και η αγορά πρέπει να γίνει μέσα σε n ημέρες. Ο

αγοραστής μπορεί να τερματίσει την ημέρα m και να λάβει x_m . Για την αποδοχή της κάθε προσφοράς τιμής, ο αγοραστής έχει να αποφασίσει άμεσα εάν θα δεχθεί την τιμή ή εάν την απορρίψει. Άπαξ και την δεχθεί, ο αγοραστής τερματίζει τη διαδικασία. Άπαξ και την απορρίψει, μία προσφορά τιμής δεν μπορεί να επανεξεταστεί (οι παρελθούσες τιμές δεν ανακαλούνται). Ο αγοραστής πρέπει να βρει έναν κανόνα τερματισμού που θα ελαχιστοποιεί την αναμενόμενη τιμή του υλικού [39].



Σχήμα 5.3. Οι τιμές απόφασης α_k σε σχέση με τα στάδια (παρατηρήσεις) $k = 1, \dots, n$, με $n = 30$.

5.2.2. Ακολουθιακή Ταξινόμηση

Τα αποτελέσματα των αποφάσεων βέλτιστου τερματισμού σχετικά με τον αναμενόμενο αριθμό των παρατηρήσεων έχουν εξεταστεί στα [41] και [42], από τα οποία έχουμε εμπνευστεί για την βασισμένη σε OST ακολουθιακή ταξινόμηση τροχιάς. Χρησιμοποιούμε το πρόβλημα βέλτιστου τερματισμού της αγοράς-αγαθού με πεπερασμένο ορίζοντα n [36] για ακολουθιακή ταξινόμηση τροχιάς. Το υπό εξέταση πρόβλημα είναι με πεπερασμένο ορίζοντα επειδή πρέπει να πάρουμε μία απόφαση ταξινόμησης μέσα σε ένα διάστημα n ακολουθιακών παρατηρήσεων (κινήσεων) του κινούμενου αντικειμένου. Ο ταξινομητής υιοθετεί μία προς τα πίσω (backward) δυναμική διαδικασία προγραμματισμού μέσω της οποίας αποκτά ένα σύνολο από τιμές απόφασης για την ταξινόμηση της τροχιάς, όπως θα εξηγηθεί αργότερα. Η προσφερόμενη τιμή στο στάδιο m ($1 \leq m \leq n$) αντιστοιχεί στην τιμή u_m^* .

Θεωρούμε μία (χρόνο-ομοιογενή) Markov αλυσίδα $(u_m^*)_{1 \leq m \leq n}$ με πεπερασμένο χρονικό ορίζοντα n ορισμένη πάνω σε έναν (φιλτραρισμένο) χώρο πιθανότητας $(\Omega, \mathcal{F}, (\mathcal{F}_m)_{m \geq 1}, \mathcal{P}_{u^*})$ και παίρνει τιμές στο διάστημα $[0, 1]$. Θεωρείται ότι η αλυσίδα ξεκινά από u^* υπό \mathcal{P}_{u^*} για $u^* \in [0, 1]$.

Τότε ορίζουμε το ακόλουθο πρόβλημα βέλτιστου τερματισμού

$$J(u^*) = \inf_{1 \leq m \leq n} E\{u_m^*\} \quad (5.3)$$

$u^* \in [0, 1]$ και το infimum αναλαμβάνεται πάνω σε όλους τους χρόνους τερματισμού m . Ο ταξινομητής μπορεί να τερματίσει την διαδικασία παρατήρησης με μία u_m^* τιμή στην m -ιστή παρατήρηση και να ταξινομήσει την Q στο P^* . Οι τιμές u_l^* , $l = 1, \dots, m$, που δεν γίνονται δεκτές από τον ταξινομητή δεν ανακαλούνται. Η τελευταία τιμή u_n^* πρέπει να γίνει δεκτή εάν κάθε προηγούμενη τιμή έχει απορριφθεί. Η αποδοχή της τελευταίας τιμής u_n^* έχει την ακόλουθη ερμηνεία: ο ταξινομητής τερματίζει την διαδικασία και είτε ταξινομεί την Q στο P^* ή αποφασίζει να μην ταξινομήσει την Q σε κανένα πρότυπο από την K , και εισάγει την Q στην K ως νέο πρότυπο. Αυτή η ειδική περίπτωση θα εξηγηθεί αργότερα.

Ο αντικειμενικός σκοπός του ταξινομητή είναι να βρει ένα χρόνο βέλτιστου τερματισμού m για να τερματίσει με την ελάχιστη τιμή u_m^* με σκοπό να εξασφαλίσει ότι η τρέχουσα τροχιά Q ταιριάζει επαρκώς με το P^* , με αποτέλεσμα, να επιτευχθεί υψηλό αποτέλεσμα πρόβλεψης. Το προτεινόμενο σύστημα, αφού ταξινομήσει την Q ως P^* , θεωρεί την υποτροχιά του P^* , $p_{m+1} \dots p_n$ ως την πιο πιθανή μελλοντική ακολουθία κίνησης για το κινούμενο αντικείμενο (βλέπε Ενότητα 5.3).

5.2.3. Διατύπωση Δυναμικού Προγραμματισμού

Έστω ότι η z_m αντιπροσωπεύει την τρέχουσα κατάσταση του ταξινομητή στην m -ιστή παρατήρηση της κίνησης του κινούμενου αντικειμένου και ας ορίσουμε την κατάσταση τερματισμού z_T για την διαδικασία παρατήρησης. Η κατάσταση $z_m = z_T$ στο $m \leq n$ υποδηλώνει ότι ο ταξινομητής ταξινομεί την τρέχουσα Q ως P^* . Από την άλλη πλευρά, εάν ο ταξινομητής είναι στην κατάσταση $z_m \neq z_T$ τότε δεν έχει ακόμη ταξινομήσει την Q . Σε αυτήν την περίπτωση $z_m = u_{m-1}^*$. Επίσης θεωρούμε την πλασματική κατάσταση $z_1 = 0$. Επιπρόσθετα, η απόφαση του ταξινομητή στο m , d_m , παίρνει δύο τιμές: «τερμάτισε και ταξινόμησε» και «συνέχισε και μην ταξινομήσεις». Ως εκ τούτου, μπορούμε να γράψουμε την εξίσωση του συστήματος του ταξινομητή ως ακολούθως

$$z_{m+1} = \begin{cases} z_T & \text{αν } z_m = z_T \\ z_T & \text{αν } z_m \neq z_T \text{ και } d_m = \text{«τερμάτισε και ταξινόμησε»} \\ u_m^* & \text{αλλιώς, «συνέχισε και μην ταξινομήσεις»} \end{cases} \quad (5.4)$$

Τελικά, έστω ότι $J_m(z_m)$ δηλώνει τις συναρτήσεις βέλτιστης τιμής cost-to-go. Τότε, η εξίσωση του Bellman για αυτό το σύστημα είναι:

$$J_n(z_n) = z_n \quad (5.5)$$

Και για την $m = 1, \dots, n - 1$ παρατήρηση

$$J_m(z_m) = \min [(1+r)^{n-m} z_m, E\{J_{m+1}(u_m^*)\}] \quad (5.6)$$

Η παράμετρος $r \geq 0$ στις εξισώσεις (3), (6) είναι ένας παράγοντας επιβράβευσης, όσο μεγαλύτερος είναι ο r τόσο μικρότερη είναι η τιμή της επιβράβευσης, και, όσο $m \rightarrow n - 1$, η τιμή της επιβράβευσης $(1+r)^{n-m}$ τείνει στο $(1+r)$ (δηλαδή, μέγιστη επιβράβευση). Μέσω αυτού του παράγοντα, η διαδικασία της ταξινομήσης γίνεται (προσωρινά) συντηρητική υποδηλώνοντας ότι καθυστερεί την απόφαση ταξινομήσης στα πρώιμα στάδια. Συγκεκριμένα, μία χαμηλή τιμή απόστασης u_m^* σε ένα πρώιμο στάδιο, μπορεί να οδηγήσει σε ανεπαρκή (ανώριμη γνώση) πίστη για την ομοιότητα της Q με το P^* . Όσο ο χρόνος περνάει, παρατηρούμε περισσότερα σημεία και, ως εκ

τούτου, ο ταξινομητής επιβραβεύει την απόφαση του να τερματίσει. Έτσι, μία διαρκώς χαμηλή τιμή της u_m^* όσο $m \rightarrow n$ υποδηλώνει ότι είμαστε πιο βέβαιοι ότι η Q ομοιάζει με κάποιο πρότυπο P^* . Ωστόσο, ένας τέτοιος παράγοντας επιβράβευσης παρακινεί τον ταξινομητή να «καθυστερήσει» την βέλτιστη απόφαση του. Από την άλλη πλευρά, ο ταξινομητής πρέπει να λάβει μία απόφαση μέσα σε n ακολουθιακές παρατηρήσεις.

Στην Εξίσωση (5.6), ο όρος $(1+r)^{n-m}z_m$ δηλώνει τον κίνδυνο που απορρέει εάν ο ταξινομητής τερματίσει και ταξινομήσει την Q ως P^* . Ο όρος $E\{J_{m+1}(u_m^*)\}$ δηλώνει τον αναμενόμενο κίνδυνο που απορρέει αν ο ταξινομητής συνεχίσει να παρατηρεί κινήσεις. Έτσι, είναι βέλτιστο να τερματίσουμε στην κατάσταση z_m (και να ταξινομήσουμε το Q) αν:

$$z_m \leq \frac{E\{J_{m+1}(u_m^*)\}}{(1+r)^{n-m}}$$

Αλλιώς, είναι βέλτιστο να συνεχίσουμε. Η τιμή $\alpha_m = \frac{E\{J_{m+1}(u_m^*)\}}{(1+r)^{n-m}}$ είναι ο αναμενόμενος κίνδυνος έως και την κατάσταση m . Η βέλτιστη πολιτική για τον ταξινομητή στην m -ιστή παρατήρηση είναι:

- Τερμάτισε την παρατήρηση με z_m και ταξινόμησε την Q αν $z_m < \alpha_m$
- Συνέχισε την παρατήρηση και μην ταξινομείς αν $z_m > \alpha_m$

Αν $z_m = \alpha_m$ τότε και οι δύο αποφάσεις είναι βέλτιστες. Η βέλτιστη πολιτική καθορίζεται από τις βαθμωτές «τιμές απόφασης» $\alpha_1, \alpha_2, \dots, \alpha_n$ μέσω των οποίων ο ταξινομητής αποφασίζει είτε να τερματίσει (και να ταξινομήσει) ή να συνεχίσει την παρατήρηση της πορείας του κινούμενου αντικειμένου.

Λήμμα 1: Οι τιμές απόφασης $\alpha_1, \alpha_2, \dots, \alpha_n$ για την διαδικασία ακολουθιακής ταξινόμησης μπορούν να ληφθούν επαγωγικά για $m = n$ έως το 1 σύμφωνα με την εξίσωση

$$\alpha_m = \frac{1}{1+r} \left(\alpha_{m+1}(1 - \wp(\alpha_{m+1})) + \int_0^{\alpha_{m+1}} u^* dP(u^*) \right) \quad (5.7)$$

Με τερματική συνθήκη

$$\alpha_n = \frac{1}{1+r} \int_0^1 u^* dP(u^*) = \frac{1}{1+r} E\{u^*\} \quad (5.8)$$

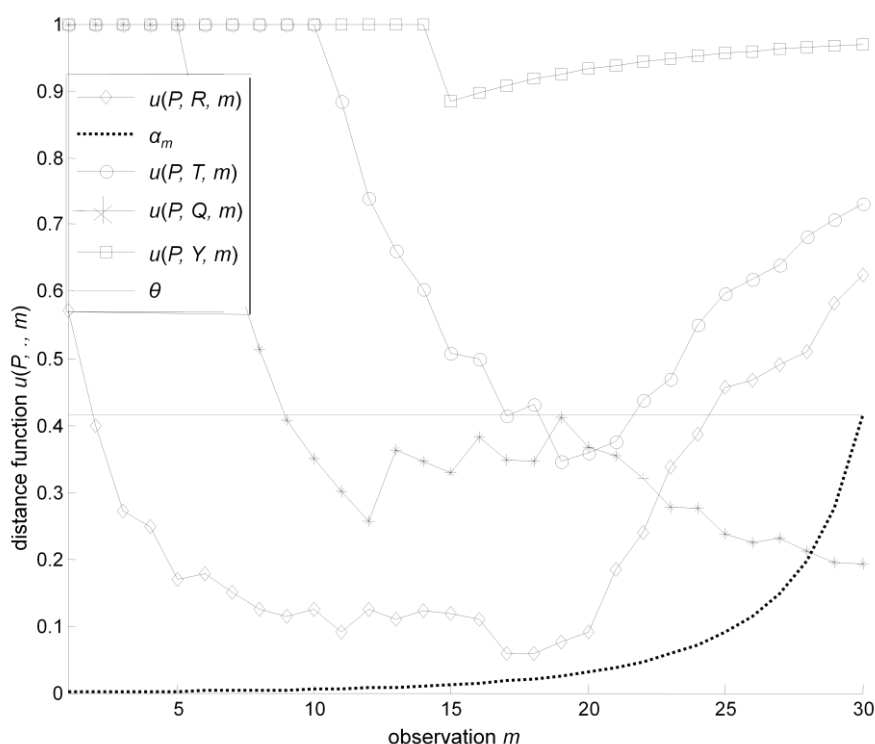
Όπου η συνάρτηση \wp ορίζεται για όλα τα βαθμωτά u τέτοια ώστε $\wp(u) = Prob\{u^* < u\}$ (βλέπε Παράρτημα Α για απόδειξη).

Σημειώστε ότι οι τιμές απόφασης δεν εξαρτώνται από το μέγιστο αριθμό των παρατηρήσεων n . Επίσης, ισχύει ότι $\alpha_m \leq \alpha_{m+1}$ (βλέπε Παράρτημα Α για απόδειξη, επίσης Σχήμα 5.3), δηλαδή, ο ταξινομητής χαλαρώνει την προσωρινά συντηρητική πολιτική για την ταξινόμηση της Q .

Η τιμή α_n υποδηλώνει την τελική απόφαση της ταξινόμησης της Q ή όχι, άπαξ και καμία από τις προηγούμενες αποφάσεις δεν έκανε ταξινόμηση. Σε αυτό το σημείο, θα επεκτείνουμε την απόφαση του ταξινομητή. Θεωρήστε ένα κατώφλι απόστασης $\theta \in (0, 1]$ τέτοιο ώστε η Q να ταξινομείται ως P^* άπαξ και $u_m^* \leq \theta$. Η διαδικασία «καθυστερεί» την απόφαση ταξινόμησης έτσι ώστε η u^* να είναι το πολύ θ . Η τελική τιμή $u_n^* > \theta$ υποδηλώνει ότι η Q δεν μπορεί να ταξινομηθεί (αν δεν είχε ήδη γίνει σε ένα προηγούμενο στάδιο) σε κανένα πρότυπο. Σε αυτήν την περίπτωση, η Q εισάγεται στην K ως ένα νέο πρότυπο. Ως εκ τούτου, για μία ορισμένη τιμή θ , έχουμε ότι $\theta = \alpha_n = \frac{1}{1+r} E\{u^*\}$. Θεωρούμε ότι οι u^* τιμές είναι ομοιόμορφα κατανομημένες στο $[0, 1]$ και, έτσι, $\wp(u^*) = u^*$ με $E\{u^*\} = 0.5$. Η γνώση των βέλτιστων πολιτικών σε αυτήν την περίπτωση

είναι επαρκής για να αποκτήσουμε έναν κανόνα βέλτιστης απόφασης για κάθε συνεχόμενη $\varphi(u^*)$ λαμβάνοντας αποτελέσματα που παρατηρούνται για την ομοιόμορφη κατανομή στο $[0, 1]$ στην ακολουθία $\varphi(u_1^*), \varphi(u_1^*), \dots, \varphi(u_n^*)$ (46). Τότε για ένα ορισμένο θ , παρατηρούμε τον παράγοντα έκπτωσης $r = \max\{0, \frac{1}{2\theta} - 1\}$. Σημειώστε ότι τα u_m^* είναι στην πραγματικότητα επικαλυπτόμενα. Η $J_m(z_m)$ αξιολόγηση απαιτεί γνώση στην $\varphi(u_m^*)$ για κάθε στάδιο m με πολύπλοκο αναλυτικό τύπο για κάθε $J_m(z_m) = \min[(1+r)^{n-m} z_m, E\{J_{m+1}(H(u_m^*), \xi_m)\}]$, $H(u_m^*, \xi_m)$ είναι η συνάρτηση επικάλυψης που οδηγείται από ανεξάρτητες διαταραχές ξ_m . Ο αναγνώστης μπορεί να ανατρέξει στο [53] (Κεφ. 4).

Το Σχήμα 5.4 δείχνει τις $u(P, \cdot, m)$ τιμές για τις Q, R, T, Y τροχιές από το Σχήμα 5.1 με $\theta = 0.4167$ ($r = 0.2$), $n = 30$. Η Y δεν ταξινομείται ποτέ ως P δεδομένου ότι $u(P, Y, m)$ ποτέ δεν διασχίζει την a_m καμπύλη, η Y είναι στην αντίθετη κατεύθυνση της P . Η R και η T επίσης δεν ταξινομούνται ως P . Αν δεν καθυστερούσαμε την απόφαση ταξινόμησης για τις R και T τότε οι R και T θα είχαν ταξινομηθεί ως P κατά την $2^{\text{η}}$ και $19^{\text{η}}$ παρατήρηση, αντίστοιχα. Ωστόσο, σε μία μεταγενέστερη παρατήρηση (δηλαδή, $24^{\text{η}}$), οι R και T δεν ταξινομούνται ως P . Η Q ταξινομείται ως P στην $28^{\text{η}}$ παρατήρηση. Αφού παρατηρούμε ότι η Q συνεχώς ομοιάζει με την P τότε είμαστε πιο σίγουροι για μία απόφαση ταξινόμησης. Θα υπήρχε μεγάλος κίνδυνος εσφαλμένης ταξινόμησης αν είχαμε ταξινομήσει την Q μετά την $9^{\text{η}}$ παρατήρηση.



Σχήμα 5.4. Η συνάρτηση απόστασης $u(P, \cdot, m)$ για τις τροχιές Q, R, T, Y , τιμές απόφασης α_m , και κατώφλι θ .

5.3. Πρόβλεψη Ακολουθίας Κίνησης

5.3.1. Πρόβλεψη Κίνησης

Η προβλεπόμενη ακολουθία κίνησης για ένα κινούμενο αντικείμενο είναι η σειρά των σημείων της υπο-τροχιάς του P^* . Αν ο ταξινομητής σταματήσει στην m -ιστή παρατήρηση της Q ($m < n$), τότε η προβλεπόμενη ακολουθία είναι η

$$p_{m+1}^* p_{m+2}^* \cdots p_n^*$$

Θεωρείστε τον δείκτη $\delta_k \in \{0, 1\}$ για $k = m + 1, \dots, n$ τέτοιον ώστε: $\delta_k = 1$ αν η πραγματική θέση του κινούμενου αντικειμένου q_k είναι κοντά στην προβλεπόμενη θέση p_k^* σε σχέση με το κατώφλι ακρίβειας $\varphi > 0$. Αυτό υποδηλώνει μία επιτυχή πρόβλεψη στο k . Αλλιώς, $\delta_k = 0$. Έτσι

$$\delta_k = \begin{cases} 1 & \text{αν } \|q_k - p_k^*\| \leq \varphi \\ 0 & \text{αλλιώς} \end{cases} \quad (5.9)$$

Αν η πλειοψηφία των προσπαθειών πρόβλεψης ήταν ανεπιτυχής, τότε η Q εισάγεται στην K . Αλλιώς, μία Διαδικασία Μείωσης Χωρικής Διακύμανσης (SVRP) ενεργοποιείται για την πιθανή υποκατάσταση του P^* με την Q (βλέπε Ενότητα 5.3.2). Το οποίο είναι

- Αν $\sum_{k=m+1}^n \delta_k \geq \frac{1}{2}(n - m)$ τότε ενεργοποίησε την $SPP(P^*, Q)$,
- Αλλιώς $K \leftarrow K \cup \{Q\}$

5.3.2. Διαδικασία Μείωσης Χωρικής Διακύμανσης

Η Διαδικασία Μείωσης Χωρικής Διακύμανσης (SVRP) είναι ένας μηχανισμός ο οποίος ενεργοποιείται υπό όρους από την πρόβλεψη κίνησης. Η ερώτηση είναι ποια τροχιά μεταξύ των P^* και Q είναι η πιο αντιπροσωπευτική για την συμπεριφορά του κινούμενου αντικειμένου. Με τον όρο «πιο αντιπροσωπευτική» εννοούμε ότι η τροχιά που θα διατηρηθεί στην K παρουσιάζει τη μικρότερη χωρική διακύμανση (στο επίπεδο), έτσι ελαχιστοποιεί τον πιθανό χωρικό θόρυβο.

Πίνακας 5.1. Η συνάρτηση $SVRP(P^*, Q)$

Input: P^*, Q **Output:** T

$T \leftarrow P^*$

If $(\sigma_{Q_x}^2 \sigma_{P_x^*}^2 > \sigma_{Q_y}^2 \sigma_{P_y^*}^2)$ **Then**

/*μέγιστη διακύμανση στην X διάσταση*/

If $(\sigma_{Q_x}^2 < \sigma_{P_x^*}^2)$ **Then** $T \leftarrow Q$ **End If**

Else /*μέγιστη διακύμανση στην Y διάσταση*/

If $(\sigma_{Q_y}^2 < \sigma_{P_y^*}^2)$ **Then** $T \leftarrow Q$ **End If**

End If

Return T

Η SVRP πιθανότατα ενημερώνει την K με το να υποκαταστήσει το P^* με την Q αν η Q παρουσιάζει λιγότερη διακύμανση (με το να είναι λιγότερο θορυβώδης, και έτσι, πιο «καθαρή») στην διαδρομή του κινούμενου αντικειμένου από ότι το P^* . Έστω ότι $\sigma_{Q_x}^2, \sigma_{Q_y}^2$ και $\sigma_{P_x^*}^2, \sigma_{P_y^*}^2$ είναι οι διακυμάνσεις στις X (γεωγραφικό μήκος) και Y (γεωγραφικό πλάτος) διαστάσεις των Q και P^* , αντίστοιχα. Η αντικατάσταση του P^* με την Q επιτυγχάνεται εξετάζοντας εάν η διακύμανση στην X (ή Y) διάσταση του P^* είναι μεγαλύτερη από αυτή της Q . Ο Πίνακας 5.1 δείχνει την SVRP συνάρτηση. Η έξοδος της συνάρτησης είναι μία πιο απλοποιημένη τροχιά. Ως εκ τούτου, η K πάντα διατηρεί πρότυπα με τις μικρότερες διακυμάνσεις.

Πίνακας 5.2. Ο Αλγόριθμος του Συστήματος Βέλτιστης Παύσης

Input: $\theta; \varphi; n$

Output: K

$K \leftarrow \{P\}$ /*αρχικοποίηση με ένα πρότυπο*/

$\{\alpha_i\}_{i=1}^n$ /*τιμές απόφασης*/

While (true) **Do**

$m \leftarrow 1; k \leftarrow 0; stopped = false$ /*χρόνος βέλτιστου τερματισμού*/

$Q = ()$ /*άδεια ακολουθία*/

While $m \leq n$ **Do**

$Q \leftarrow q_1 \cdots q_{m-1}q_m$ /* m - ιοστή παρατήρηση*/

$u_m^* \leftarrow \min_{P \in K} u(P, Q, m)$

If ($u_m^* \leq \alpha_m$) **Then** /*ταξινόμηση*/

$k \leftarrow m; P^* \leftarrow \arg \min u_m^*; stopped \leftarrow true; \mathbf{break}$

Else $m \leftarrow m + 1$ /*επόμενη παρατήρηση*/

End If

End While

If ($stopped$) **Then** /*ταξινόμηση*/

$\{\delta_i\}_{i=m+1}^n = prediction(P^*, Q, m)$

If $\sum_{i=m+1}^n \delta_i \geq \frac{1}{2}(n - m)$ **Then** $SVRP(P^*, Q)$

Else /*εισαγωγή νέου προτύπου*/ $K \leftarrow K \cup \{Q\}$

End If

Else /*μη-ταξινόμηση*/

$K \leftarrow K \cup \{Q\}$ /*εισαγωγή νέου προτύπου*/

End If

End While

5.3.3. Ο Αλγόριθμος του Συστήματος Βέλτιστης Παύσης

Ο αλγόριθμος για την σειριακή ταξινόμηση, πρόβλεψη κίνησης και SVRP δίνεται στη λίστα του Πίνακα 5.2. Αξίζει να σημειωθεί ότι η SVRP μπορεί να υιοθετήσει αλγορίθμους απλοποίησης τροχιάς, δηλαδή, [38], για να διατηρήσει την τροχιά με την ελάχιστη χωρική διακύμανση. Το οποίο σημαίνει ότι, η SVRP θα μπορούσε να επιλέξει την «απλούστερη» τροχιά στην K . Τέτοιοι αλγόριθμοι «απλοποιούν» μία τροχιά P με μέγεθος n με το να διατηρήσουν το συνολικό πλαίσιο της πληροφορίας του P σε μία τροχιά P' μεγέθους $l < n$, αφαιρώντας επαναλαμβανόμενη χωρική πληροφορία, και διατηρώντας τη σημασιολογία της τροχιάς. Ωστόσο, μία τέτοια διαδικασία έχει ως αποτέλεσμα μία απλοποιημένη τροχιά P' η οποία κρατάει μόνο το σκελετό της αρχικής τροχιάς P . Αυτό σημαίνει ότι δεν περιλαμβάνει όλη τη χωρική πληροφορία για να κάνει προβλέψεις (δηλαδή, GPS σημεία), και έτσι, υποβαθμίζει το αποτέλεσμα της πρόβλεψης.

5.4. Αξιολόγηση Απόδοσης

Πρώτα, εισάγουμε τις μετρικές απόδοσης. Μετά, εξετάζουμε την απόδοση του προτεινόμενου συστήματος με οδηγούμενες από ίχνη προσομοιώσεις από πραγματικά

δεδομένα. Τελικά, παρέχουμε μία συγκριτική αξιολόγηση με άλλα μοντέλα και συζητάμε θέματα υπολογιστικής πολυπλοκότητας.

5.4.1. Μετρικές Απόδοσης

Η είσοδος στο σύστημα είναι ένα σύνολο από τροχιές $I = \{Q_1, Q_2, \dots\}$. Οι τροχιές επεξεργάζονται ακολουθιακά από το σύστημα. Αρχικά, η K περιέχει ένα πρότυπο, το οποίο είναι η πρώτη τροχιά από το σύνολο. Τότε, η K ενημερώνεται ακολουθιακά μέσω του αλγορίθμου στον Πίνακα 2. Μία τροχιά μπορεί είτε να εισαχθεί στην K ή να ταξινομηθεί άπαξ και ο ταξινομητής σταματήσει σε κάποια παρατήρηση $m_i < n$. Έστω ότι $C = \{Q_i: m_i < n\} \subseteq I$ είναι το σύνολο των τροχιών που έχουν ταξινομηθεί και, έτσι, το σύστημα μπορεί να τις χρησιμοποιήσει για να κάνει προβλέψεις. Άπαξ το σύστημα ταξινομήσει μία $Q \in C$ ως P^* στην $m - \text{ιστή}$ παρατήρηση ($m < n$) τότε:

$$\alpha_Q = \frac{1}{n - m} \sum_{k=m+1}^n \delta_k$$

Υποδηλώνει την *ακρίβεια πρόβλεψης ανά Q* . $\alpha_Q \in [0, 1]$, όσο πιο υψηλή είναι η τιμή του α_Q τόσο πιο επιτυχώς το σύστημα προβλέπει την κίνηση του κινούμενου αντικειμένου. Επιπρόσθετα, η μετρική

$$\alpha = \frac{1}{|C|} \sum_{Q \in C} \alpha_Q$$

Με $\alpha \in [0, 1]$ να υποδηλώνει την ικανότητα του συστήματος να κάνει επιτυχείς προβλέψεις δεδομένου ότι ταξινομεί την ακολουθία του κινούμενου αντικειμένου Q , $|C|$ είναι η πληθικότητα του C .

Επιπρόσθετα, ορίζουμε την *ακρίβεια πρόβλεψης* ως ακολούθως

$$\beta = \frac{\sum_{Q \in C} (\sum_{k=m+1}^n \delta_k)}{\sum_{Q \in C} (n - m)}$$

Όπου ο ταξινομητής τερματίζει στην $m - \text{ιστή}$ παρατήρηση της Q ακολουθίας ($m < n$) και τότε κάνει $n - m$ απόπειρες πρόβλεψης. Η $\beta \in [0, 1]$ μετρική υποδηλώνει μία συνολική ικανότητα του συστήματος να κάνει επιτυχείς προβλέψεις από το σύνολο των προσπαθειών πρόβλεψης.

Επίσης εξετάζουμε πόσες αποθηκευμένες τροχιές στην K απαιτούνται από το σύστημα για να αποκτήσει υψηλές τιμές α και β . Αυτό είναι πολύ σημαντικό ειδικά όταν αντιμετωπίζουμε περιορισμούς περιορισμένης χωρητικότητας και υπολογιστικής πολυπλοκότητας. Η τιμή $\frac{|K|}{|I|}$ είναι ένας ποσοτικός δείκτης της ικανότητας του ταξινομητή να μαθαίνει και να αναπαριστά την συμπεριφορά κίνησης του κινούμενου αντικειμένου χωρίς να αποθηκεύει πλεονάζουσα πληροφορία. Η τιμή $\frac{|K|}{|I|}$ πρέπει να είναι χαμηλή, έτσι ώστε, να ελαχιστοποιεί το κόστος αποθήκευσης του ταξινομητή. Ορίζουμε την αποδοτικότητα $e \in [0, 1]$ ως το γινόμενο της ακρίβειας πρόβλεψης με την απαιτούμενη πληροφορία αποθήκευσης στην K έως και την ακολουθιακή είσοδο συστήματος I :

$$e = \beta \left(1 - \frac{|K|}{|I|} \right)$$

Αξίζει να επισημανθεί ότι στην αρχή το σύστημα έχει $|K| = 1$, και η K προοδευτικά αυξάνεται και/ή ενημερώνεται (μέσω της αντικατάστασης προτύπου). Μία υψηλή τιμή e υποδηλώνει ένα σύστημα με υψηλή απόδοση πρόβλεψης και χαμηλές απαιτήσεις

μνήμης. Η ελάχιστη τιμή της e είναι μηδέν δεδομένου ότι η ελάχιστη $\beta = 0$ και/ή $|K| \simeq |I|$. Η τιμή e εκφράζει από μόνη της την απόδοση του συστήματος. Η σχέση μεταξύ ακρίβειας πρόβλεψης και αποθήκευσης εξαρτάται από την εφαρμογή. Μία κινητή εφαρμογή μπορεί να απαιτεί υψηλά αποτελέσματα πρόβλεψης με σκοπό να εκμεταλλευτεί περαιτέρω τις προβλεπόμενες θέσεις ενός κινούμενου αντικειμένου, ωστόσο, με τίμημα το κόστος αποθήκευσης. Η μετρική e βοηθάει στην σύγκριση απόδοσης μεταξύ δύο συστημάτων, τα οποία επιτυγχάνουν είτε σχεδόν ίση ακρίβεια πρόβλεψης αλλά διαφορετικό κόστος αποθήκευσης ή, όμοιες $|K|$ τιμές αλλά διαφορετικά αποτελέσματα πρόβλεψης.

Πίνακας 5.3. Παράμετροι Συστήματος

Παράμετρος	Σημειογραφία
Μήκος προτύπου	n
Παράγοντας επιβράβευσης	r
Κατώφλι απόστασης	θ
Κατώφλι ακρίβειας	φ
Χωρικό κατώφλι	ϵ
Τιμές απόφασης	$\{a_m\}_{m=1}^n$
Κατάσταση ταξινομητή	$z_m, m = 1 \dots n$

Τελικά, μέσω της SVRP συνάρτησης το σύστημα προσπαθεί να διατηρήσει «καθαρά» πρότυπα έτσι ώστε να επιτύχει υψηλά αποτελέσματα πρόβλεψης. Έστω ότι $S \subseteq C$ είναι το σύνολο των ταξινομημένων τροχιών Q που αντικαθιστούν τα αντίστοιχα πρότυπα P^* μετά την ενεργοποίηση της SVRP. Το μέσο ποσοστό της μείωσης της χωρικής διακύμανσης ορίζεται ως

$$v_X = \frac{1}{|S|} \sum_{Q \in S} \frac{\|\sigma_{Q_X}^2 - \sigma_{P_X}^2\|}{\|\sigma_{P_X}^2\|}$$

Η ποσότητα v_X υποδηλώνει το ποσοστό μείωσης της διακύμανσης στην X διάσταση των τροχιών που αντικαθιστούν πρότυπα από την K από το συνολικό αριθμό των αντικαταστάσεων $|S|$. Όμοια ορίζεται και το v_Y . Η ποσότητα $v = \frac{1}{2}(v_X + v_Y) \in [0, 1]$ υποδηλώνει την ποσοστιαία μείωση των διακυμάνσεων και στις δύο διαστάσεις ανά αριθμό των αντικαταστάσεων.

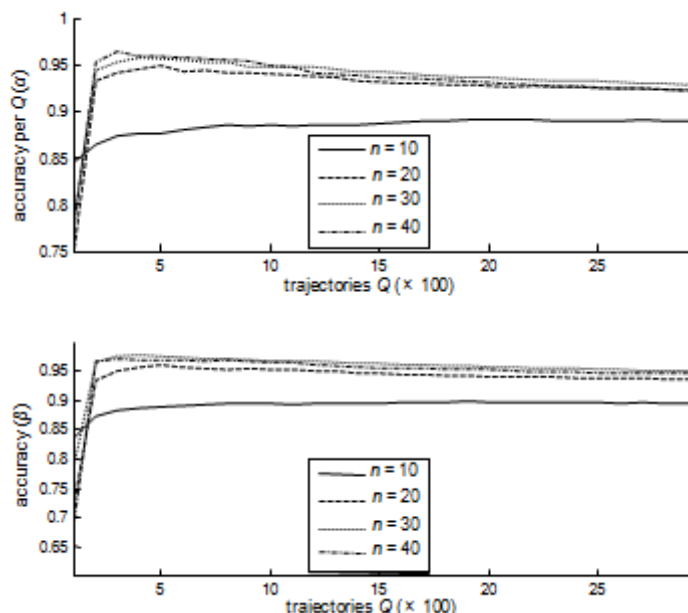
5.4.2. Παράμετροι Προσομοίωσης

Για τις προσομοιώσεις μας, χρησιμοποιήσαμε 10 σύνολα δεδομένων που περιέχουν κινητά ίχνη GPS. Τα σύνολα δεδομένων I_i περιέχουν 3000, κατά μέσο όρο, χρονοσημασμένες (x, y) GPS συντεταγμένες. Ο μέσος χρόνος απόστασης μεταξύ δύο συνεχόμενων θέσεων είναι 5s. Για να εξετάσουμε την ανθεκτικότητα του συστήματος μας (σε σχέση με τα θορυβώδη χωροχρονικά δεδομένα), δημιουργούμε (ήμι) συνθετικά σύνολα δεδομένων $I_i(\pi)$ από κάθε σύνολο δεδομένων I_i με το να εισάγουμε θόρυβο ως ακολούθως: (i) τεμαχίζουμε το I σε τροχιές με n σημεία, (ii) για κάθε τροχιά $Q \in I$, και για κάθε σημείο $q \in Q$, δημιουργούμε μία παράκαμψη (τυχαίου μήκους) με πιθανότητα $\pi \in \{0.1, 0.5, 0.8\}$. Οι α, β και e τιμές στην αξιολόγηση της απόδοσης αναφέρονται στις μέσες τιμές όλων των συνόλων δεδομένων I_i και $I_i(\pi)$.

Οι παράμετροι προσομοίωσης (σημειογραφία και το εύρος/τιμών) δίνονται στον Πίνακα 5.4. Το χωρικό κατώφλι ϵ για δύο σημεία στην συνάρτηση απόστασης σημείων είναι 5m. Το κατώφλι απόστασης $\theta = 0.3$ ($r = 0.67$) υποδηλώνει μία μάλλον συντηρητική ακολουθιακή διαδικασία ταξινόμησης. Το κατώφλι ακρίβειας φ εξαρτάται από την εφαρμογή, δηλώνει την ευαισθησία της κινητής εφαρμογής στην εκτίμηση θέσης και πρόβλεψης του κινούμενου αντικειμένου. Η τιμή φ δεν μπορεί να είναι μικρότερη από την ακρίβεια θέσης του υιοθετούμενου συστήματος εντοπισμού, φ_{min} . Ένας απλός GPS δέκτης παρουσιάζει ακρίβεια περίπου $\varphi_{min} = 7m$. Η απαίτηση για ακριβή εκτίμηση θέσης, και στη συνέχεια, για μία επιτυχή πρόβλεψη, εξαρτάται από την ακρίβεια του συστήματος εντοπισμού. Αν το σύστημα εντοπισμού δεν μπορεί να ξεχωρίσει μεταξύ θέσεων που βρίσκονται μέσα σε έναν κύκλο με ακτίνα μικρότερη ή ίση από το φ_{min} τότε η πραγματική και η προβλεπόμενη θέση δεν είναι διακριτές. Ορίζουμε το $\varphi = 7.5m$. Τελικά, πειραματιζόμαστε με $10 \leq n \leq 40$, έτσι, ένα πρότυπο αναπαριστά κινήσεις διάρκειας 50s έως 200s.

Πίνακας 5.4. Παράμετροι Προσομοίωσης

Παράμετρος	Τιμή/Εύρος
Μήκος προτύπου n	[10, 40]
Παράγοντας επιβράβευσης r	0.67
Χωρικό κατώφλι ϵ	5m
Κατώφλι απόστασης θ	0.3
Κατώφλι ακρίβειας φ	7.5m

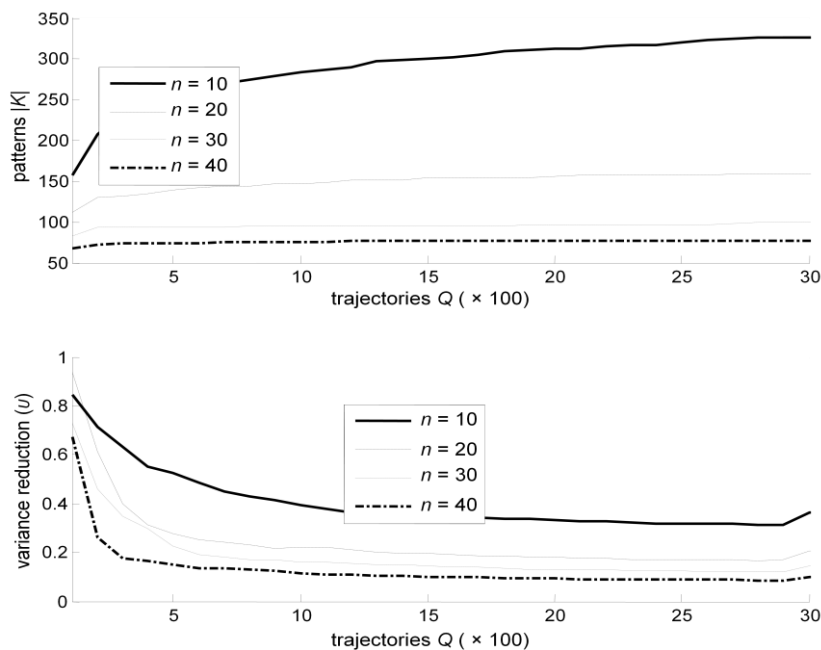


Σχήμα 5.5. (πάνω) Η ακρίβεια πρόβλεψης ανά Q (α) και (κάτω) η ακρίβεια πρόβλεψης (β) σε σχέση με τις τροχιές Q για $n = \{10, 20, 30, 40\}$; $\pi = 0.5$.

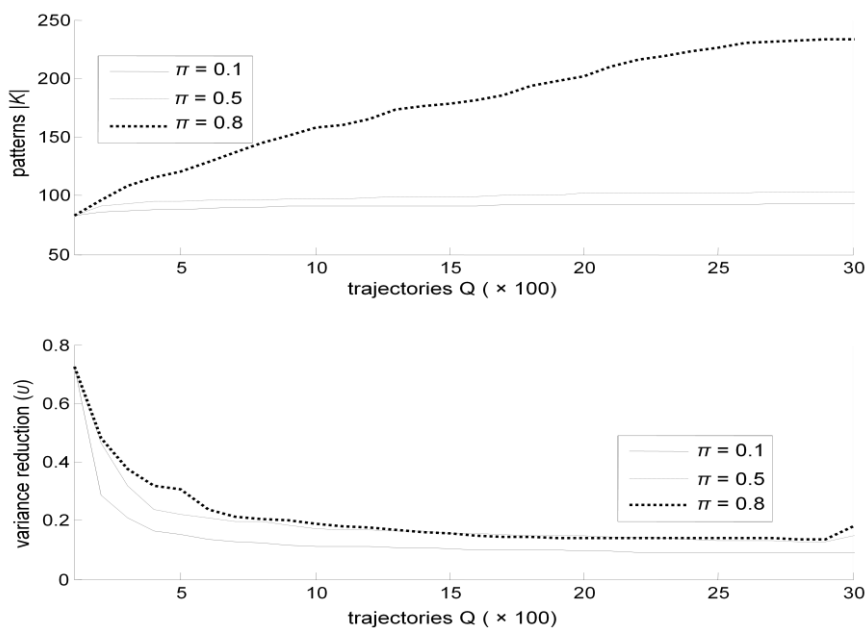
5.4.3. Αξιολόγηση Απόδοσης

Σε αυτήν την ενότητα εξετάζουμε τη συμπεριφορά του προτεινόμενου συστήματος με τις υπό συζήτηση μετρικές απόδοσης. Πειραματιστήκαμε με τον αριθμό των αποθηκευμένων προτύπων που χρησιμοποιούνται για να κάνουμε προβλέψεις, την ικανότητα του συστήματος να διατηρεί όσο πιο καθαρή γίνεται την K , και την

ανθεκτικότητα του συστήματος σε θορυβώδη πρότυπα. Τελικά, συνοψίζουμε την επίπτωση της παραμέτρου του μήκους του προτύπου στην συμπεριφορά του συστήματος.



Σχήμα 5.6. (πάνω) Ο αριθμός των προτύπων $|K|$ και (κάτω) η μείωση της διακύμανσης v σε σχέση με τις τροχιές Q για $n = \{10, 20, 30, 40\}$; $\pi = 0.5$.



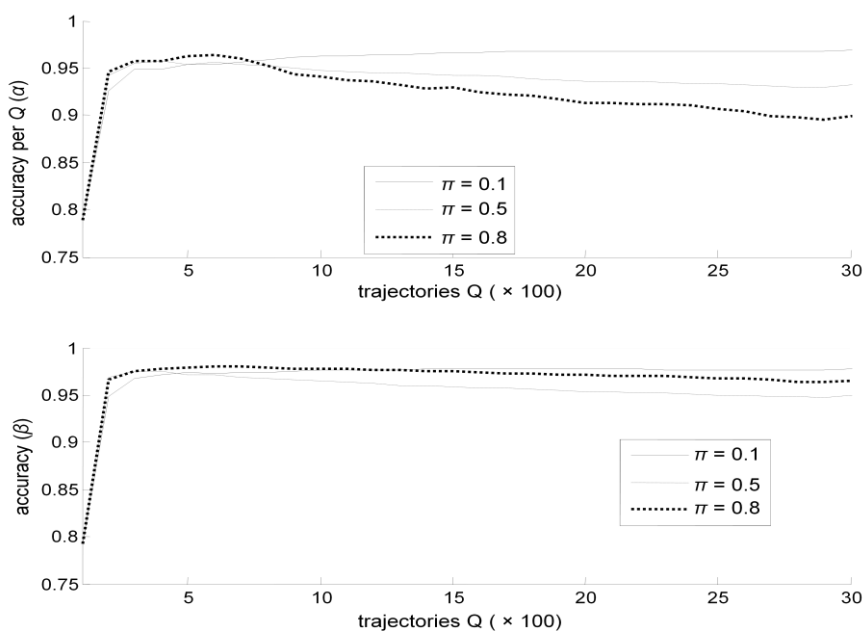
Σχήμα 5.7. (πάνω) Ο αριθμός των προτύπων $|K|$ και (κάτω) η μείωση της διακύμανσης v σε σχέση με τις τροχιές Q για $\pi = \{0.1, 0.5, 0.8\}$; $n = 30$.

Το Σχήμα 5.5 δείχνει την εξέλιξη της ακρίβειας πρόβλεψης ανά τροχιά, α , και την ακρίβεια πρόβλεψης, β , του προτεινόμενου συστήματος για διαφορετικό μήκος προτύπου n σε σχέση με τον αριθμό των (εισερχόμενων) τροχιών δείγματος $Q \in I$ και $\pi = 0.5$.

Το σύστημα επιτυγχάνει υψηλές τιμές α και β ακόμα και για αρκετά θορυβώδη πρότυπα ($\pi = 0.5$). Παρατηρούμε αποτελέσματα πρόβλεψης της τάξης του $\alpha = 0.92$ και $\beta = 0.95$ για $n = \{20, 30, 40\}$. Για $n = 10$ παρατηρούμε μία μείωση της τάξης του 3% και 6% για

τις τιμές α και β , αντίστοιχα. Το μήκος των προτύπων επηρεάζει την ικανότητα του συστήματος να κάνει επιτυχείς προβλέψεις όπως αναφέρεται αργότερα σε αυτήν την ενότητα.

Στο Σχήμα 5.6 (πάνω) σχεδιάζουμε την εξέλιξη του αριθμού των αποθηκευμένων προτύπων $|K|$ σε σχέση με τον αριθμό του δείγματος των τροχιών για διαφορετικές τιμές του n . Ο ρυθμός της αύξησης του $|K|$ μειώνεται περισσότερο στην περίπτωση όπου $n > 10$ από ότι στην περίπτωση όπου $n = 10$. Επιπρόσθετα, το σύστημα για $\pi = 0.5$ αποθηκεύει 52%, 69%, 77% λιγότερα πρότυπα για $n = (20, 30, 40)$, αντίστοιχα, από ότι για $n = 10$. Ανάλογα ποσοστά αναφέρονται στο πραγματικό μέγεθος αποθήκευσης (δηλαδή, $|K| \cdot n$). Αυτό υποδηλώνει ότι η απόφαση για το n επηρεάζει το κόστος αποθήκευσης του συστήματος όπως παρατηρείται ομοίως για την ακρίβεια πρόβλεψης (όπως αναφέρεται παραπάνω). Ωστόσο, το μήκος n επηρεάζει την «καθαρότητα» της K όπως θα δούμε παρακάτω.

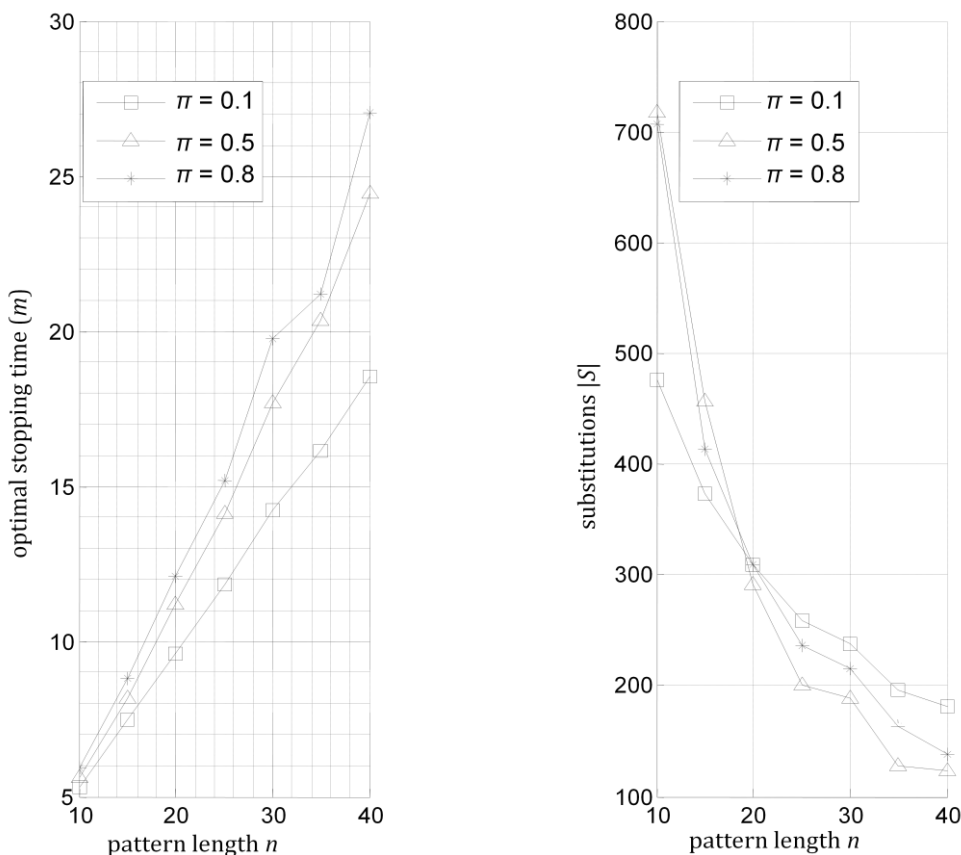


Σχήμα 5.8. (πάνω) Η ακρίβεια πρόβλεψης ανά Q (α) και (κάτω) η ακρίβεια πρόβλεψης (β) σε σχέση με τις τροχιές Q για $\pi = \{0.1, 0.5, 0.8\}$; $n = 30$.

Στο Σχήμα 5.6 (κάτω) δείχνουμε την εξέλιξη του ποσοστού της μείωσης διακύμανσης ν για διαφορετικά μήκη προτύπων n με $\pi = 0.5$. Η K διαδοχικά καθαρίζεται δεδομένου ότι η τιμή της ν μειώνεται με τον αριθμό των αντικαταστάσεων. Αυτό σημαίνει ότι οι αντικαταστάσεις των P^* με τις Q σταδιακά αναφέρονται σε ακόμα πιο καθαρά πρότυπα. Μία τιμή ν δηλώνει το μέσο ποσοστό της μείωσης της διακύμανσης δεδομένου ότι μία αντικατάσταση προτύπου λαμβάνει χώρα. Ως εκ τούτου, η τιμή της ν παίρνει ακόμα μικρότερες τιμές συγκρινόμενη με αυτές τις τιμές στην αρχή του συστήματος. Επιπλέον, παρατηρούμε 73% λιγότερη τιμή της ν για $n = 40$ συγκρινόμενη με $n = 10$, έτσι, παρουσιάζεται η σημαντικότητα του μεγέθους του προτύπου για τον καθαρισμό της K .

Έχουμε να εξετάσουμε την αντοχή του συστήματος σε σχέση με το π . Αυτό παρατηρείται με το να εξετάσουμε την επίπτωση των θορυβωδών προτύπων στις θεωρούμενες μετρικές απόδοσης. Το Σχήμα 5.7 δείχνει την εξέλιξη της K και της ν σε σχέση με το δείγμα των τροχιών Q για διαφορετικές τιμές π , με μέγεθος προτύπου $n = 30$. Η επίπτωση των θορυβωδών προτύπων στην $|K|$ είναι σημαντική για $\pi = 0.8$ ($|K| = 235$). Από την άλλη πλευρά, για $\pi = (0.1, 0.5)$ η αντίστοιχες $|K|$ τιμές είναι σχετικά χαμηλές $|K| = (93, 103)$. Ο μηχανισμός SVRP για τον καθαρισμό της K έχει ως

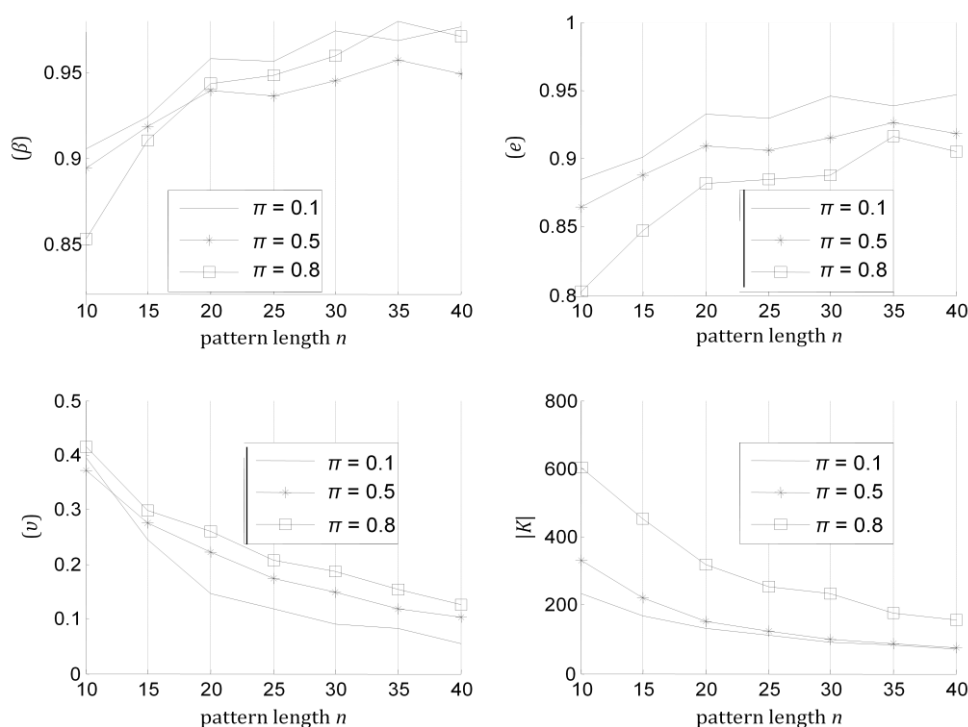
αποτέλεσμα χαμηλές τιμές v ακόμα και για υψηλές τιμές π , όπως φαίνεται στο Σχήμα 5.7 (κάτω). Το σύστημα επιτυγχάνει όμοιες τιμές v για όλες τις τιμές π . Έτσι, τα πρότυπα που είναι αποθηκευμένα στην K καθαρίζονται ανεξάρτητα από τις τιμές π .



Σχήμα 5.9. (αριστερά) Οι μέσοι χρόνοι βέλτιστου τερματισμού (m) και (δεξιά) ο αριθμός των αντικαταστάσεων $|S|$ σε σχέση με το μήκος του προτύπου n για $\pi = \{0.1, 0.5, 0.8\}$.

Το Σχήμα 5.8 δείχνει την ανθεκτικότητα του συστήματος στην πρόβλεψη θέσεων εκφρασμένη μέσω των μετρικών α και β , για διαφορετικές τιμές π , με $n = 30$. Οι μέσες τιμές β για όλα τα π είναι 0.97 ενώ παρατηρούμε 3% μείωση στην α τιμή για $\pi = 0.8$ συγκρινόμενη με αυτή για $\pi = (0.1, 0.5)$.

Στο Σχήμα 5.9 (αριστερά) πειραματιζόμαστε με τη μέση αναμονή της διαδικασίας ακολουθιακής ταξινόμησης λόγω του αλγορίθμου βέλτιστου χρόνου τερματισμού. Το Σχήμα 5.9 (αριστερά) δείχνει τις μέσες τιμές των βέλτιστων χρόνων τερματισμού m σε σχέση με τα μήκη των προτύπων n . Οι τιμές m υποδηλώνουν τις παρατηρήσεις που απαιτούνται από τον ταξινομητή για να φθάσει μία απόφαση. Όσο πιο μικρή είναι η τιμή m τόσο πιο μικρή γίνεται η καθυστέρηση για την ταξινόμηση. Όσο το n αυξάνει, παρατηρούμε μία γραμμική αύξηση στην καθυστέρηση για την ταξινόμηση με μία κλίση της τάξης του $\approx (0.5, 0.6, 0.7)$ για $\pi = (0.1, 0.5, 0.8)$, αντίστοιχα. Το οποίο συμβαίνει διότι, ο ταξινομητής παρατείνει την απόφαση ταξινόμησης όσο περισσότερες παρατηρήσεις είναι διαθέσιμες. Μία υψηλή τιμή n έχει ως αποτέλεσμα ένα υψηλότερο υπολογιστικό κόστος και καθυστέρηση ταξινόμησης. Ωστόσο, υψηλές τιμές n έχουν ως αποτέλεσμα μία υψηλή ακρίβεια πρόβλεψης και χαμηλές τιμές v , όπως φαίνεται στο Σχήμα 5.10 (πάνω – αριστερά), (κάτω – αριστερά). Επιπλέον, το Σχήμα 5.9 (δεξιά) δείχνει τον αριθμό των αντικαταστάσεων $|S|$ σε σχέση με το n για διάφορες τιμές του π . Όσο πιο μεγάλο είναι το n τόσο περισσότερες είναι οι αντικαταστάσεις που πραγματοποιούνται από τον SVRP για τον καθαρισμό του K με λιγότερο αριθμό από αποθηκευμένα πρότυπα $|K|$, βλέπε επίσης και Σχήμα 5.10 (κάτω – δεξιά).



Σχήμα 5.10. Η ακρίβεια πρόβλεψης (β), αποδοτικότητα (e), μείωση διακύμανσης (v), και αριθμός των προτύπων $|K|$ σε σχέση με το μήκος του προτύπου n για $\pi = \{0.1, 0.5, 0.8\}$.

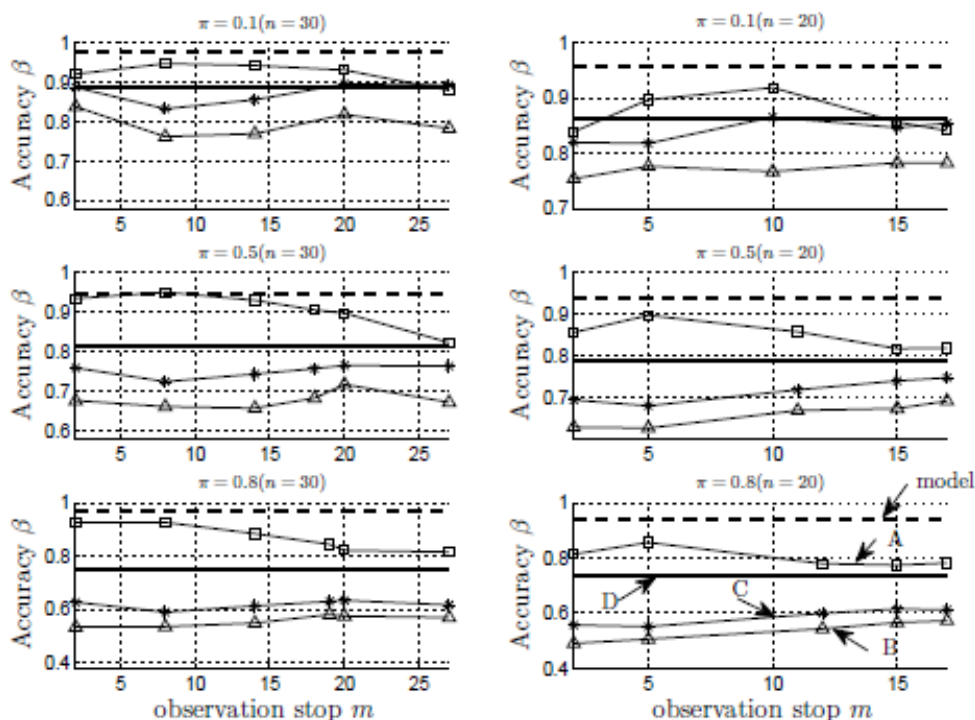
Τελικά, το Σχήμα 5.10 (πάνω – δεξιά) δείχνει την τιμή e για διάφορες τιμές n και $\pi = \{0.1, 0.5, 0.8\}$. Το σύστημα είναι μόνο 6% λιγότερο αποδοτικό για $\pi = 0.8$ από ότι για $\pi = 0.1$ έχοντας $n = 40$. Αυτό δηλώνει την ανθεκτικότητα του συστήματος στο να διατηρεί την αποδοτικότητα σχετικά σταθερή ακόμα και για θορυβώδη δεδομένα. Επιπρόσθετα, το σύστημα είναι 7%, 4% και 11% πιο αποδοτικό για $n = 40$ από ότι για $n = 10$ για $\pi = (0.1, 0.5, 0.8)$, αντίστοιχα. Όσο πιο υψηλό είναι το μήκος του προτύπου n τόσο πιο αποδοτικό είναι το σύστημα.

Για να συνοψίσουμε, η συνολική συμπεριφορά του συστήματος εξαρτάται από το μήκος του προτύπου n και την θορυβώδη φύση των παρατηρούμενων προτύπων κίνησης (που αντανακλάται από τις τιμές π). Όσο πιο υψηλή είναι η τιμή n , τόσο πιο ψηλή είναι η ακρίβεια πρόβλεψης, και τόσο πιο καθαρή είναι η K σε σχέση με τις τιμές v , και τόσο πιο μικρή είναι η τιμή $|K|$ (βλέπε επίσης Σχήμα 5.10 (κάτω – δεξιά)). Από την άλλη πλευρά, πρότυπα με μεγάλο μήκος έχουν ως αποτέλεσμα υψηλό υπολογιστικό κόστος για την ταξινόμηση και καθυστέρηση στην πρόβλεψη. Τελικά, το σύστημα είναι ανθεκτικό σε σχέση με την τιμή π για να κάνει προβλέψεις και να καθαρίζει την K .

5.4.4. Συγκριτική Αξιολόγηση

Ο προτεινόμενος ταξινομητής κατηγοριοποιείται ως ένας ταξινομητής ανταγωνιστικής μάθησης υιοθετώντας τον ορισμό της ταξινόμησης από το γνωστικό πεδίο της Μηχανικής Μάθησης [8]. Ο όρος ανταγωνιστική μάθηση υποδηλώνει ότι τα αποθηκευμένα πρότυπα ανταγωνίζονται μεταξύ τους να αναλάβουν την «ευθύνη» να αναπαραστήσουν μία νέα τροχιά Q . Το σύστημα μας αναφέρεται σε ένα on-line (αυξητικό) σχήμα μάθησης στο οποίο οι τροχιές λαμβάνονται μία προς μία και η K ενημερώνεται (υπό όρους) προοδευτικά. Έτσι, συγκρίνουμε την προσέγγισή μας με τον αυξητικό αλγόριθμο Adaptive Resonance Theory (ART) [43] [61], τον βασισμένο στον ART ταξινομητή τροχιάς και προγνώστη κίνησης στο [29], και τον αλγόριθμο Self-organizing Maps (SOM), ο οποίος υιοθετείται στο [44] ως ένας προγνώστης κίνησης του

χρήστη. Αναφέρουμε εν συντομία τον ART, SOM και τα βασιζόμενα στον ART συστήματα για σύγκριση ως ακολούθως.



Σχήμα 5.11. Ακρίβεια πρόβλεψης (β) σε σχέση με τον μέγιστο αριθμό των παρατηρήσεων m για (αριστερή στήλη) $n = 30$ και (δεξιά στήλη) $n = 20$ με $\pi \in \{0.1, 0.5, 0.8\}$.

1) Adaptive Resonance Theory και Self Organizing Maps:

Στον ART, το πλησιέστερο πρότυπο P_i^* στην Q ακολουθία ενημερώνεται, και τα άλλα πρότυπα στην K δεν ενημερώνονται. Στον SOM, ένα σύνολο από πρότυπα κοντά στο πρότυπο νικητή ενημερώνονται σε κάποιο βαθμό ανάλογα με την κατάταξη απόστασης, με το P_i^* να είναι το πλησιέστερο (νικητής) πρότυπο στην Q , και $P_{i-N}, P_{i-N+1}, \dots, P_{i-1}, P_{i+1}, \dots, P_{i+N}$ είναι η γειτονιά του P_i^* . Σε αυτή την περίπτωση, η γειτονιά είναι μεγέθους N . Ο ART και ο SOM αρχικοποιούνται με ένα πρότυπο και με N (τυχαία) πρότυπα, αντίστοιχα. Και οι δύο αλγόριθμοι προσθέτουν νέα πρότυπα στην K , αν χρειαστεί.

Η διαδικασία ταξινόμησης στον ART και τον SOM έχει ως ακολούθως: Δεδομένης μίας τροχιάς εισόδου Q , η απόσταση $u(P, Q, m)$ για κάποιο συγκεκριμένο m υπολογίζεται για όλα τα $P \in K$. Έστω P_i^* να είναι το πλησιέστερο πρότυπο στην Q με ελάχιστη απόσταση $u_i^* = u(P_i^*, Q, m)$. Αν $u_i^* \leq \theta$ τότε η ενημέρωση του P_i^* στον ART γίνεται από τον κανόνα προσαρμογής $P_i^* \leftarrow P_i^* + \eta(P_i^* - Q)$. Αλλιώς, η Q εισάγεται στην K . Ο όρος ενημέρωσης $\eta(P_i^* - Q)$ κινεί το P_i^* προς την Q με έναν παράγοντα $\eta \in [0, 1]$. Αν ο η μειώνεται προς το 0, χάνουμε την προσαρμογή σε νέα πρότυπα τα οποία μπορεί να προκύψουν προϊόντος του χρόνου επειδή οι ενημερώσεις γίνονται πολύ μικρές. Αν κρατήσουμε τον η υψηλά, τότε το P_i^* μπορεί να ταλαντώνεται (αυτό εξηγείται περεταίρω στο δίλλημα σταθερότητας – πλαστικότητας στο [48]). Στον SOM, αν $u_i^* \leq \theta$ τότε η ενημέρωση της γειτονιάς $\{P_{i-N}, \dots, P_{i+N}\}$ παριστάνεται με τον κανόνα προσαρμογής $P_l \leftarrow P_l + \eta \zeta(i, l)(P_i - Q)$. Αλλιώς, η Q εισάγεται στην K . Όπου $\zeta(i, l)$ είναι η συνάρτηση γειτονιάς τέτοια ώστε $\zeta(i, l) = 1$ όταν $l = i$ και μειώνεται όσο το $|i - l|$ αυξάνεται. Υιοθετούμε την αντίστοιχη Gaussian συνάρτηση $\zeta(i, l) = \exp(-\frac{(i-l)^2}{2s^2})$, όπου s^2 είναι η παράμετρος πλάτους που βαθμιαία μειώνεται όσο το $\frac{|K|}{|I|}$ συγκλίνει.

2) Περιγραφή Συστημάτων:

Επιλέγουμε να πειραματιστούμε με τον αλγόριθμο SOM επειδή έχει υιοθετηθεί από το μοντέλο στο [44]. Για λόγους σύγκρισης, πειραματιζόμαστε με συστήματα που υιοθετούν τους αλγορίθμους ART και SOM για ομαδοποίηση και πρόβλεψη ακολουθιακής κίνησης χωρίς καμία φάση εκπαίδευσης. Η υιοθέτηση αυτών των αλγορίθμων έχει ως ακολούθως. Πρώτα, τα πρότυπα στην K του συστήματος μας, ART, και SOM είναι μήκους n . Θέτουμε έναν συγκεκριμένο μέγιστο αριθμό από παρατηρήσεις $m < n$, το οποίο σημαίνει ότι τα ART και SOM μοντέλα τερματίζουν την παρατήρηση των κινούμενων αντικειμένων μετά τις πρώτες m παρατηρήσεις της Q και, τότε, προβαίνουν στην ταξινόμηση και πρόβλεψη. Θυμηθείτε ότι, αντίθετα με το προτεινόμενο μοντέλο, τα βασισμένα στον ART και SOM συστήματα απαιτούν έναν προκαθορισμένο μέγιστο αριθμό από παρατηρήσεις m με σκοπό να αποφασίσουν στην ταξινόμηση μίας κινούμενης ακολουθίας Q . Μία Q ταξινομείται ως P^* με $u^* = \min_{P \in K} u(P, Q, m)$. Τότε η πρόβλεψη κίνησης επιτυγχάνεται λαμβάνοντας υπόψη μας την $p_{m+1} \cdots p_n$ υπό-ακολουθία του P^* . Επιπρόσθετα, αν $u^* \leq \theta$ τότε το P^* και η γειτονιά του P^* ενημερώνονται βασιζόμενοι στους κανόνες προσαρμογής για τον ART και SOM, αντίστοιχα, δοσμένου του η . Στα πειράματα μας θέτουμε $\eta = 0.5$ και το μέγεθος της γειτονιάς $N = 2$. Αν $u^* > \theta$, η Q εισάγεται στην K . Στη συνέχεια, θα αναφερόμαστε στο σύστημα που υιοθετεί τον ART αλγόριθμο ως System B, και το σύστημα που υιοθετεί τον SOM αλγόριθμο ως System C.

Το σύστημα στο [29], το οποίο αναφέρεται ως System A, επεκτείνει τον ART δίνοντας του ανατροφοδότηση στην διαδικασία ενημέρωσης της K . Το System A αρχικοποιεί την K με ένα πρότυπο. Το System A είτε εισάγει ένα νέο πρότυπο άπαξ και η Q δεν μπορεί να ταξινομηθεί αλλιώς ενημερώνει την K βασιζόμενο στο αποτέλεσμα πρόβλεψης. Η πλειοψηφία των προσπαθειών πρόβλεψης όπως στην Ενότητα. Στην περίπτωση μίας επιτυχούς πρόβλεψης, το System A δεν ενημερώνει την K . Στην περίπτωση μίας ανεπιτυχούς πρόβλεψης, το System A ενεργοποιεί τον ART για την ενημέρωση της K . Το οποίο σημαίνει ότι, είτε ενημερώνει το P^* με τον παράγοντα η ή εισάγει την Q στην K . Το System A χρησιμοποιεί Hausdorff-like απόσταση για να υπολογίσει το u^* .

Έχουμε επίσης υλοποιήσει μία παραλλαγή του συστήματος μας η οποία υιοθετεί τον αλγόριθμο πιθανότητας βέλτιστης παύσης στο [62] για την ακολουθιακή ταξινόμηση τροχιάς. Θεωρήστε τις τ_1, \dots, τ_n ανεξάρτητες συναρτήσεις δείκτη $\tau_m \in \{0, 1\}$ με $p_m = E(\tau_m)$. Το $r_m = \frac{p_m}{1-p_m}$ είναι η πιθανότητα του γεγονότος $\{\tau_m = 1\}$. Παρατηρούμε τους δείκτες ακολουθιακά και μπορούμε να κάνουμε παύση το πολύ σε έναν, αλλά μόνο online, το οποίο σημαίνει, τη στιγμή της παρατήρησης. Κερδίζουμε αν κάνουμε παύση στην τελευταία $\tau_m = 1$ (αν υπάρχει κάποια) και χάνουμε αλλιώς (συμπεριλαμβανομένου να μην κάνουμε καμία παύση). Ο κανόνας βέλτιστης παύσης αυτού του προβλήματος είναι: κάνε παύση στον πρώτο δείκτη τ_m με $\tau_m = 1$ και $m \geq \kappa$ όπου $\kappa = \sup \{m : \sum_m^n r_m \geq 1\}$. Αν δεν υπάρχει κανένας, κάνε παύση στο n και χάσε. Στην δική μας ακολουθιακή ταξινόμηση, $\tau_m = f(P^*, Q, m)$ με $P^* = \arg \min_{P \in K} u(P, Q, m)$. Το οποίο σημαίνει ότι, ταξινομούμε την Q ως P^* άπαξ και η συνάρτηση απόστασης σημείου $f(P^*, Q, m) = 1$ για τις παρατηρήσεις $m \geq \kappa$. Θέτουμε $p_m = \frac{1}{m}$ όπως υιοθετείται στο [63], και αναφερόμαστε σε αυτήν την παραλλαγή ως System D.

Με στόχο να συγκρίνουμε αντικειμενικά τα Systems A, B και C με το σύστημα μας, τρέχουμε προσομοιώσεις για διαφορετικό μέγιστο αριθμό παρατηρήσεων m δοσμένου ενός συγκεκριμένου μήκους προτύπου n . Το οποίο σημαίνει ότι, συγκρίνουμε τις μετρικές: ακρίβεια πρόβλεψης, αριθμός των αποθηκευμένων προτύπων και

αποδοτικότητας για τα συστήματα σε σχέση με τον ίδιο ορίζοντα πρόβλεψης $n - m$ δεδομένου ότι η Q ταξινομείται στην $m - \text{ιστή}$ παρατήρηση. Επιπρόσθετα, τα σύνολα εισόδου I και $I(\pi)$ είναι ακριβώς τα ίδια για όλα τα συστήματα. Αρχικά η K του συστήματος μας, του System A, του System B και του System D περιέχει ένα πρότυπο, ενώ του System C περιέχει $2N + 1$ τυχαία πρότυπα.

3) Σύγκριση των Συστημάτων:

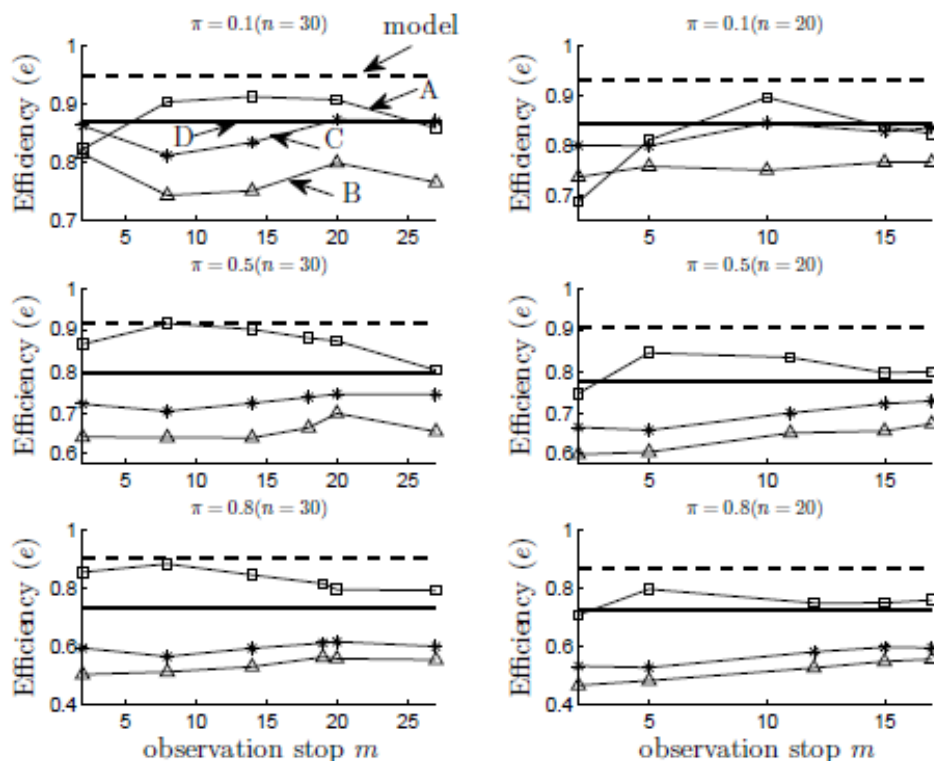
Το Σχήμα 5.11 δείχνει την ακρίβεια πρόβλεψης β για το σύστημα μας (μοντέλο) και για τα συστήματα A, B, C και D σε σχέση τον τερματισμό των παρατηρήσεων $m < n$ για $n \in \{20,30\}$ και $\pi \in \{0.1, 0.5, 0.8\}$. Η γραμμή δείκτη του κάθε συστήματος σε όλες τις καμπύλες του Σχήματος 5.11 είναι το ίδιο.

Πίνακας 5.5. Ποσοστό των αποθηκευμένων προτύπων $|K|$ όσον αφορά το μοντέλο μας

$\pi = 0.1, n = 30$				$\pi = 0.1, n = 20$			
m	A	B	C	m	A	B	C
2	272%	-1%	-3%	2	556%	-16%	-10%
8	62%	-12%	-1%	5	247%	-15%	-15%
14	11%	-14%	-1%	10	-10%	-17%	-15%
19	4%	-15%	-1%	12	-18%	-18%	-16%
20	-10%	-16%	-2%	14	-20%	-19%	-16%
27	-15%	-18%	-1%	17	-23%	-25%	-16%
$\pi = 0.5, n = 30$				$\pi = 0.5, n = 20$			
m	A	B	C	m	A	B	C
2	132%	57%	5%	2	503%	117%	98%
8	18%	0%	-10%	5	180%	60%	45%
14	0%	-10%	-11%	11	38%	20%	16%
19	-10%	-11%	-10%	12	23%	13%	16%
20	-10%	-10%	-12%	14	15%	13%	16%
27	-11%	-12%	-11%	17	14%	13%	15%
$\pi = 0.8, n = 30$				$\pi = 0.8, n = 20$			
m	A	B	C	m	A	B	C
2	17%	10%	-30%	2	82%	16%	10%
8	17%	7%	-30%	5	80%	15%	10%
14	10%	4%	-28%	10	45%	15%	20%
19	4%	4%	-25%	12	44%	10%	26%
20	-1%	1%	-22%	14	40%	9%	25%
27	-5%	-8%	-21%	17	33%	-2%	23%

Το σύστημα μας επιτυγχάνει μεγαλύτερη ακρίβεια πρόβλεψης μεταξύ όλων των συστημάτων για όλες τις τιμές π και n . Παρατηρούμε τη δεύτερη υψηλότερη β τιμή για το System A. Το System C επιτυγχάνει πάντοτε υψηλότερες β τιμές από ότι το System B, ενώ το System D επιτυγχάνει χαμηλότερο β από ότι το System A. Με σκοπό να πειραματιστούμε αντικειμενικά περισσότερο με την ακρίβεια πρόβλεψης για όλα τα συστήματα, και έτσι, να θεωρήσουμε αρκετά όμοιο ορίζοντα πρόβλεψης $n - m$, θέτουμε $m = m^*$ για τα συστήματα A, B και C, όπου m^* είναι ο μέσος χρόνος τερματισμού (μέγιστος αριθμός παρατηρήσεων) στον οποίο το σύστημα μας και το System D τερματίζει την παρατήρηση και συνεχίζει με την πρόβλεψη. Στην περίπτωση όπου $(\pi, n) = (0.1, 30)$ παρατηρούμε ότι $[m^*] = 14$, για το σύστημα μας και το System D, αντίστοιχα. Σε αυτήν την περίπτωση, τα συστήματα A, B και C επιτυγχάνουν $\beta = (0.94, 0.76, 0.85)$, αντίστοιχα, ενώ για το σύστημα μας $\beta = 0.98$, και για το System D,

$\beta = 0.88$. Στην περίπτωση όπου $(\pi, n) = (0.8, 30)$ έχουμε $[m^*] = 19$ και $[m^*] = 14$ για το σύστημα μας και για το System D, αντίστοιχα. Τα συστήματα A, B και C επιτυγχάνουν $\beta = (0.84, 0.58, 0.63)$, αντίστοιχα, ενώ για το σύστημα μας, $\beta = 0.968$ και για το System D, $\beta = 0.75$. Όμοια συμπεριφορά παρατηρείται και για τις υπόλοιπες τιμές (π, n) . Επιπρόσθετα, παρατηρούμε την ανθεκτική συμπεριφορά του δικού μας συστήματος συγκρινόμενου με τα άλλα συστήματα. Τα συστήματα B και C επιτυγχάνουν πολύ χαμηλές τιμές β για $\pi = 0.8$ και $n = 30$ ($\beta = 0.58$ και $\beta = 0.62$, αντίστοιχα), ενώ το σύστημα μας κάνει 20% πιο επιτυχείς προβλέψεις από ότι το System A όσο το $m \rightarrow n$. Το System D, επιτυγχάνει 15% χαμηλότερες τιμές β από ότι το σύστημα μας για όλες τις π, n τιμές, κατά μέσο όρο. Η β τιμή είναι ακόμα μικρότερη για τα συστήματα B και C για $\pi = 0.8$ και $n = 20$.



Σχήμα 5.12. Αποδοτικότητα (e) σε σχέση με τον μέγιστο αριθμό των παρατηρήσεων m για (αριστερή στήλη) $n = 30$ και (δεξιά στήλη) $n = 20$ με $\pi \in \{0.1, 0.5, 0.8\}$.

Ο Πίνακας 5.5 δείχνει το ποσοστό αύξησης/μείωσης των αποθηκευμένων προτύπων $|K|$ για τα συστήματα A, B, και C όσον αφορά το δικό μας μοντέλο σε σχέση με το m για διαφορετικές π τιμές και $n = \{20, 30\}$. Οι έντονες m τιμές αναφέρονται στην $[m^*]$ τιμή του δικού μας μοντέλου για κάθε περίπτωση (π, n) . Το System A έχει υψηλές απαιτήσεις μνήμης για όλες τις π τιμές με $m < [m^*]$. Για $m = [m^*]$, το System A αποθηκεύει 28% περισσότερα πρότυπα από ότι το μοντέλο μας, κατά μέσο όρο, για όλες τις (π, n) τιμές. Για $\pi = 0.1$ τα συστήματα B και C αποθηκεύουν λιγότερα πρότυπα από ότι το σύστημα μας, ωστόσο, σε βάρος των χαμηλών β τιμών. Συγκεκριμένα, τα συστήματα B και C αποθηκεύουν 18% λιγότερα πρότυπα, κατά μέσο όρο, από ότι το σύστημα μας αλλά επιτυγχάνουν 44% και 24%, αντίστοιχα χαμηλότερες β τιμές από ότι το σύστημα μας. Το System B αποθηκεύει περισσότερα πρότυπα για θορυβώδη πρότυπα κίνησης ($\pi = 0.8$) και $n = 20$, ενώ το System C αποθηκεύει 22% λιγότερα πρότυπα για $n = 30$. Σε αυτήν την περίπτωση τα συστήματα B και C δεν είναι για πρόβλεψη λόγω των πολύ χαμηλών β τιμών. Για το System D, παρατηρούμε (α) $[m^*] = 14$ για όλα τα π και $n = 30$ και (β) $[m^*] = 10$ για όλα τα π και $n = 20$. Το System D αποθηκεύει 48% και 56%

λιγότερα πρότυπα από ότι το μοντέλο μας, κατά μέσο όρο, για τις περιπτώσεις (α) και (β), αντίστοιχα (η σύγκριση έγινε για τον ίδιο ορίζοντα πρόβλεψης). Τώρα, εξετάζουμε την αποδοτικότητα του κάθε συστήματος βασιζόμενοι στην ήδη συμπληρωμένη K .

Το Σχήμα 5.12 δείχνει την αποδοτικότητα όλων των συστημάτων σε σχέση με το m για διάφορες τιμές π με $n = \{20, 30\}$. Το System B είναι το λιγότερο αποδοτικό μέσω του οποίου παρατηρούμε μία μείωση της τάξης του 34% στην e (Δe) από $\pi = 0.1$ έως $\pi = 0.8$. Το System C είναι σχετικά αποδοτικό, συγκεκριμένα για υψηλές τιμές m με λιγότερο θορυβώδη πρότυπα κίνησης. Παρατηρούμε μία μείωση της τάξης του 32% στην e από $\pi = 0.1$ έως $\pi = 0.8$ (αρκετά συγκρίσιμη με την $\Delta e_{0.1 \rightarrow 0.8}$ του System B). Το System A και το σύστημα μας παρουσιάζουν αρκετά όμοιες τιμές αποδοτικότητας για $\pi = 0.1$ και $10 \leq m \leq 20$. Το System A επιτυγχάνει μάλλον σταθερές e τιμές ($e = 0.8$) για κάθε m και $\pi = \{0.5, 0.8\}$. Αυτό δηλώνει μία σχεδόν ανθεκτική συμπεριφορά όσον αφορά τα θορυβώδη πρότυπα. Το προτεινόμενο σύστημα επιτυγχάνει τιμές αποδοτικότητας πάνω από 0.9 για όλα τις τιμές π και n . Αυτό υποδηλώνει ότι το σύστημα μας επιτυγχάνει όχι μόνο υψηλότερη ακρίβεια πρόβλεψης από τα συστήματα A, B και C, αλλά επίσης διατηρεί την ποσότητα $\frac{|K|}{|I|}$ σε σχετικά χαμηλά επίπεδα ενώ παράλληλα «καθαρίζει» την K . Αυτή η συμπεριφορά έρχεται σε αντίθεση με την εισαγωγή μίας νέας Q ή την ταλάντωση γύρω από μία νέα Q , που είναι στρατηγικές που ακολουθούνται από τα άλλα μοντέλα.

Τελικά, συγκρίνουμε τις e τιμές για όλα τα συστήματα πάνω στον ίδιο ορίζοντα πρόβλεψης $n - m^*$. Για το σύστημα μας έχουμε $[m^*] = (14, 19, 19)$ με $n = 30$ και $[m^*] = (10, 11, 12)$ με $n = 20$ για $\pi = (0.1, 0.5, 0.8)$, αντίστοιχα. Για $n = 30$ τα συστήματα A, B και C, είναι 7%, 29%, και 19% λιγότερο αποδοτικά συγκρινόμενα με το δικό μας σύστημα κατά μέσο όρο για όλες τις π τιμές. Για $n = 20$ παρατηρούμε τα αντίστοιχα ποσοστά 8%, 30% και 22% για τα συστήματα A, B και C, αντίστοιχα. Το System D είναι 18% λιγότερο αποδοτικό από ότι το σύστημα μας και 8% (11%) λιγότερο αποδοτικό από το System A για $n = 20$ ($n = 30$) με $[m^*] = 10$ ($[m^*] = 14$).

Για να ολοκληρώσουμε, το προτεινόμενο σύστημα επιτυγχάνει υψηλά επίπεδα πρόβλεψης ($\beta = 0.95$) ενώ διατηρεί την $|K|$ τιμή σε σχετικά χαμηλά επίπεδα για αρκετά θορυβώδη πρότυπα ($\pi \in \{0.1, 0.5\}$). Αυτό οφείλεται στο γεγονός ότι το σύστημα μας «καθαρίζει» την K βάση γνώσης με σκοπό να διατηρεί τα πρότυπα τα οποία είναι πιο αντιπροσωπευτικά για την συμπεριφορά του κινούμενου αντικειμένου από το να εισάγει απλά ένα νέο πρότυπο. Από την άλλη πλευρά, τα συστήματα B και C αποθηκεύουν λιγότερα πρότυπα αλλά, κατά μέσο όρο, επιτυγχάνουν χαμηλές τιμές ακρίβειας πρόβλεψης. Το System A αποθηκεύει λίγα πρότυπα για υψηλές m τιμές και παρουσιάζει σταθερή συμπεριφορά στην ακρίβεια πρόβλεψης για θορυβώδη πρότυπα κίνησης ($\beta = 0.8$). Το System D αποθηκεύει σχετικά λίγα πρότυπα αλλά η ακρίβεια παραμένει ως το 80%.

4) Υπολογιστική Πολυπλοκότητα

Το χρονικό κόστος για να ταξινομηθεί η Q στην $m - \text{ιστή}$ παρατήρηση ως P^* με μήκος n όσον αφορά την συνάρτηση απόστασης για το σύστημα μας και το System D είναι $O(m \cdot |K|)$. Αυτό γίνεται επειδή ο υπολογισμός της $u(P, Q, m)$ τιμής βασίζεται στον υπολογισμό της $u(P, Q, m - 1)$. Το χρονικό κόστος ταξινόμησης για τα A, B και C συστήματα είναι $O(m \cdot |K|)$. Το κόστος αποθήκευσης για όλα τα συστήματα είναι $O(n \cdot |K|)$. Το κόστος για να ενημερωθεί η K είναι $O(n)$ για το σύστημα μας και το System D αν η SVRP είναι ενεργοποιημένη ή $O(1)$ αν η Q εισάγεται στην K . Για το System A και B, το κόστος της ενημέρωσης της K είναι $O(n)$ λόγω του κανόνα προσαρμογής και/ή $O(1)$ λόγω της εισαγωγής της Q . Για το System C, το κόστος της

ενημέρωσης της K είναι $O(N \cdot n)$ λόγω της προσαρμογής των γειτονικών προτύπων. Το κόστος πρόβλεψης εξαρτάται από τον ορίζοντα πρόβλεψης $n - m$, έτσι είναι, $O(n - m)$ σε χρόνο, αυτό είναι όμοιο σε όλα τα συστήματα.

5) Περιορισμοί του μοντέλου

Το προτεινόμενο μοντέλο προσθέτει περισσότερη πολυπλοκότητα στην ταξινόμηση της τροχιάς αφού, λόγω της καθυστέρησης κατά τη διαδικασία λήψης απόφασης, το μέγεθος της υπό παρατήρηση τροχιάς αυξάνει, και, σε κάποιες περιπτώσεις, φθάνει το n . Ωστόσο, μία τέτοια καθυστέρηση κάνει το σύστημα πιο (α) σίγουρο για τις προβλέψεις συσσωρεύοντας περισσότερες παρατηρήσεις, έτσι, μειώνει το ρίσκο των λάθους προβλέψεων (το οποίο είναι ένα συχνό φαινόμενο όταν οι απαιτήσεις για πρόβλεψη είναι «εξαναγκασμένες» - πραγματοποιούνται ακριβώς όταν χρειάζονται), ή (β) ικανό να μαθαίνει μία νέα τροχιά όταν ο αριθμός των παρατηρήσεων φθάνει το n . Επιπρόσθετα, το μοντέλο λειτουργεί κατά αμνήμον τρόπο για την αντικατάσταση μίας τροχιάς μέσω της SVRP. Το οποίο σημαίνει, ότι μόνο το πλησιέστερο πρότυπο P^* χρησιμοποιείται από την πολιτική αντικατάστασης όσον αφορά τη νέα τροχιά Q . Αυτή η αμνήμον προσέγγιση προφανώς υποβαθμίζει την ικανότητα του μοντέλου να αποφασίσει πιο ευφυέστερα σχετικά με την αντικατάσταση του προτύπου. Για παράδειγμα, το μοντέλο μπορεί να διατηρήσει έναν (σταθμισμένο) γραμμικό συνδυασμό των k -πλησιέστερων προτύπων, όπου κάθε ένα μπορεί να επιδράσει στην πολιτική αντικατάστασης. Αυτή η προσέγγιση προσθέτει επιπλέον υπολογιστικές και αποθηκευτικές απαιτήσεις. Αυτή τη στιγμή, το μοντέλο μας είναι «τυφλό» σχετικά με την ιστορία των όμοιων προτύπων και υιοθετεί την υπόθεση ότι το πρότυπο με τη μικρότερη χωρική διακύμανση είναι το σωστό, έτσι, το εισάγει στην K . Τελικά, η προτεινόμενη μετρική απόστασης τροχιάς έχει έναν περιορισμό στην διαχείριση διαφορετικών συχνοτήτων χωρικής δειγματοληψίας, όπως έχει αναφερθεί στην Ενότητα 5.1.2. Από την άλλη πλευρά, μία τέτοια μετρική, με το να είναι απλή και αυξητικά ενημερώσιμη έχει γραμμική χρονική πολυπλοκότητα, και έτσι, είναι κατάλληλη για ακολουθιακή ταξινόμηση.

6. ΣΧΕΤΙΚΗ ΕΡΕΥΝΑ

Αναφερόμαστε σε μεθόδους που υιοθετούν ταξινόμηση τροχιάς για ένα βήμα μπροστά (βραχυπρόθεσμη) πρόβλεψη θέσης και k – βήματα μπροστά (μακροπρόθεσμη) πρόβλεψη θέσης, δηλαδή, πρόβλεψη ενός μελλοντικού μονοπατιού, ενός κινούμενου αντικειμένου. Οι μέθοδοι κατηγοριοποιούνται σε αυτές που (α) πρώτα μαθαίνουν την κινητή συμπεριφορά δημιουργώντας την βάση γνώσης K και μετά είναι ικανές για να κάνουν προβλέψεις (off-line μέθοδοι), (β) αυξητικά μαθαίνουν την κινητή συμπεριφορά και, ως εκ τούτου, είναι ικανές να επεκτείνουν υπό προϋποθέσεις την K (on-line μέθοδοι), (γ) ενημερώνουν την K (προσαρμοστικές μέθοδοι). Μία προσαρμοστική μέθοδος επεκτείνει την on-line μέθοδο με την έννοια ότι ενημερώνει την K με τις αλλαγές της κινητής συμπεριφοράς του κινούμενου αντικειμένου. Το κύριο μειονέκτημα των off-line μεθόδων είναι ότι άπαξ και η K δημιουργηθεί, δεν υφίσταται πλέον δυνατότητα αναβάθμισης. Ο περιορισμός αυτής της μεθόδου οδηγεί τον ταξινομητή να είναι ευάλωτος σε πιθανές αποκλίσεις (παρακάμψεις) στις τροχιές, οι οποίες μπορεί να προκαλέσουν σημαντική υποβάθμιση στην ακρίβεια πρόβλεψης.

Το off-line βραχυπρόθεσμο μοντέλο πρόβλεψης στο [45] χρησιμοποιεί Naïve Bayes ταξινόμηση πάνω σε μια βάση γνώσης K από τροχιές. Ένα τέτοιο μοντέλο υποθέτει ότι τα πρότυπα κίνησης ακολουθούν μία κανονική κατανομή. Μία τέτοια υπόθεση δεν είναι πάντοτε αληθής δεδομένου ότι η πληροφορία κινητικότητας αναφέρεται σε τροχιές άγνωστης κατανομής. Το off-line μαθησιακό αυτόματο στο [46] ακολουθεί μία γραμμική ενισχυτική μαθησιακή μέθοδο επιβράβευσης-ποινής. Αυτό το μοντέλο χρειάζεται να αποθηκεύει όλες τις πιθανές μεταβάσεις του προτύπου κινητικότητας του χρήστη, έτσι, οδηγείται σε αξιοσημείωτες απαιτήσεις μνήμης. Το βραχυπρόθεσμο μοντέλο πρόβλεψης στο [47] ασχολείται με πρόβλεψη κινητικότητας χωρίς πρότερη γνώση της συμπεριφοράς του χρήστη. Εφαρμόζει αποδεικτική λογική για την πρόβλεψη της θέσης βασιζόμενο σε κοινωνικές δραστηριότητες ενός κινητού χρήστη εκτός από την πληροφορία θέσης. Ένα τέτοιο μοντέλο συνδέεται με σημαντική υπολογιστική πολυπλοκότητα (λόγω του αλγορίθμου συλλογιστικής Dempster-Schafer) άπαξ και το ποσό των πιθανών θέσεων του χρήστη αυξάνεται και απαιτεί λεπτομερή πληροφορία χρήστη (δηλαδή, καθημερινό προφίλ, προτιμήσεις, αγαπημένα σημεία συνάντησης). Επιπρόσθετα, οι συγγραφείς στο [48] προτείνουν ένα off-line βραχυπρόθεσμο και μακροπρόθεσμο υβριδικό μοντέλο πρόβλεψης. Ένα τέτοιο μοντέλο χρησιμοποιεί σχεσιακό κανόνα εξόρυξης για συγκεκριμένα πρότυπα χρηστών. Επιπρόσθετα, το μοντέλο στο [49] υιοθετεί Self-Organizing feature Map (SOM) και πολλαπλών-επιπέδων δίκτυα perceptron για την εκμάθηση όλων των προτύπων κίνησης για ένα σύνολο από χρήστες σε κυβελωτά δίκτυα. Και τα δύο μοντέλα στο [48] και [49] υιοθετούν off-line εκπαίδευση βασιζόμενα στην ιστορία της κίνησης ενός μεγάλου αριθμού από κινητούς χρήστες. Τέτοιες προσεγγίσεις επίσης συγκεντρώνουν έναν μεγάλο όγκο πληροφορίας.

Ορισμένα μοντέλα απαιτούν πολύ συγκεκριμένη πληροφορία στο χώρο κίνησης των κινούμενων αντικειμένων. Το μακροπρόθεσμο μοντέλο πρόβλεψης στο [50] βασίζεται σε έναν Maximum Likelihood και Greedy αλγόριθμο για την πρόβλεψη τροχιάς. Ένα τέτοιο μοντέλο απαιτεί λεπτομερή πληροφορία της δομής του οδικού δικτύου για να συλλάβει τα πρότυπα των στροφών στις διασταυρώσεις και τις ταχύτητες κίνησης στους δρόμους. Ένα τέτοιο μοντέλο εφαρμόζεται βασικά σε μία συγκεκριμένη δομή οδικού δικτύου. Μία σχετικά όμοια προσέγγιση ενός off-line μακροπρόθεσμου μοντέλου πρόβλεψης προτείνεται στο [51]. Ένα τέτοιο μοντέλο προβλέπει την μελλοντική θέση των οχημάτων αρκετά δευτερόλεπτα στο μέλλον βασιζόμενο σε έναν radial basis function ταξινομητή τροχιάς.

Το off-line μοντέλο πρόβλεψης στο [52] χρησιμοποιεί Bayesian μάθηση για να εκπαιδεύσει ένα Νευρωνικό Δίκτυο και μία μέθοδο Markov chain Monte Carlo.

Επιπρόσθετα, οι συγγραφείς στο [53] πρότειναν ένα off-line βραχυπρόθεσμο μοντέλο πρόβλεψης με Partial Matching. Το βασικό συστατικό του μοντέλου στο [53] είναι ένα Markov Model μεταβλητής τάξης, το οποίο χρησιμοποιεί διαφορετικά μήκη προτύπων για να δημιουργήσει τη βάση γνώσης K . Ένα τέτοιο μοντέλο μεταχειρίζεται την μερική ταύτιση των τροχιών αλλά δίνει μικρή προσοχή στις χωρικές αποκλίσεις μεταξύ μιας πραγματικής τροχιάς και των προτύπων τροχιάς που είναι αποθηκευμένα στην K . Αφού οι χωρικές αποκλίσεις δεν αναγνωρίζονται από το μοντέλο τότε δεν μπορεί να επιτευχθεί πρόβλεψη. Αυτό οφείλεται στο γεγονός ότι δεν υπάρχει καμία μετρική για να υπολογιστεί η ομοιότητα μεταξύ μίας νέας τροχιάς και ενός αποθηκευμένου προτύπου. Και τα δύο μοντέλα στα [52] και [53] απαιτούν υψηλή υπολογιστική πολυπλοκότητα λόγω του υιοθετούμενου Markov μοντέλου. Επιπρόσθετα, το βραχυπρόθεσμο μοντέλο πρόβλεψης στο [54] υιοθετεί επίσης ένα μοντέλο Markov, το οποίο συνεπάγεται μία μεγάλη ιστορία από κινήσεις χρηστών για την εξόρυξη πληροφορίας από GPS ίχνη.

Οι συγγραφείς στο [55] προτείνουν ένα βραχυπρόθεσμο off-line μοντέλο πρόβλεψης, το οποίο βασίζεται στην θεωρία Grey (μία θεωρία λήψης αποφάσεων κάτω από συνθήκες ανακριβούς / ελλιπούς πληροφορίας). Ένα τέτοιο μοντέλο δημιουργεί μη-γραμμικές συναρτήσεις πάνω σε αβέβαια πρότυπα κίνησης. Άλλες μέθοδοι για την πρόβλεψη τροχιών έχουν επίσης προταθεί στην επιστημονική λογοτεχνία αλλά αυτές έχουν γενικά περιορισμένο εύρος αφού θεωρούν μόνο ευθύγραμμο πρότυπα κίνησης (δηλαδή, εθνικές οδούς) και όχι άγνωστα πρότυπα [56].

Οι τεχνικές μηχανικής μάθησης στο [14] έχουν επίσης εφαρμοστεί στο πρόβλημα της πρόβλεψης κίνησης σε οδικά δίκτυα βασιζόμενα σε κυψελωτές επικοινωνίες. Ο σκοπός αυτού του μοντέλου είναι να προβλέπει την επόμενη κυψέλη που θα επισκεφτεί ένα κινούμενο αντικείμενο. Ωστόσο, μικρή προσοχή δίνεται από αυτό το μοντέλο για την παροχή πρόβλεψης της διαδρομής του κινούμενου αντικειμένου. Επιπρόσθετα, το μοντέλο στο [57] επιτυγχάνει πρόβλεψη της επόμενης κυψέλης με το να διατηρεί την ιστορία μετάβασης όλων των κινούμενων αντικειμένων. Η πρόβλεψη βασίζεται στις συχνότητες των περιστατικών των μεταβάσεων και την κατεύθυνση των κινήσεων των κινούμενων αντικειμένων σχετικά με την τοπολογική τοποθέτηση της κυψέλης. Ωστόσο, το μοντέλο στο [57] υποθέτει ότι τα κινούμενα αντικείμενα έχουν όμοια πρότυπα κίνησης που υπαγορεύονται από τη δομή των κτιρίων ή των οικοδομικών τετραγώνων. Επιπρόσθετα, οι κυψέλες μπορεί να εκτείνονται πολλά τετραγωνικά μίλια, έτσι, καθιστώντας την πρόβλεψη πολύ χοντροκομμένη και άνευ αξίας για εφαρμογές που απαιτούν λεπτομερή πρόβλεψη σε χωρική πληροφορία (δηλαδή, GPS συντεταγμένες). Η προσαρμοστική μέθοδος στο [29] υιοθετεί την Adaptive Resonance Theory (ART) πάνω σε GPS ίχνη. Ο ART είναι ένας on-line μαθησιακός και προσαρμοστικός αλγόριθμος ικανός να προβλέψει αλλαγές και να προσαρμόσει / ενημερώσει μόνο συγκεκριμένα μέρη της K , έτσι παρέχει προσαρμοστικότητα του υφιστάμενου μοντέλου. Τελικά, το μοντέλο στο [44] υιοθετεί Recurrent SOM (RSOM) [58] για ομαδοποίηση και Markov μοντέλα για την αναπαράσταση κάθε προτύπου κίνησης. Ο RSOM εξειδικεύεται σε χρονική επεξεργασία ακολουθίας. Το μοντέλο στο [44] απαιτεί μία διαδικασία εκπαίδευσης για τα Markov μοντέλα πριν από την πρόβλεψη. Κατά την διαδικασία εκπαίδευσης, τα πρότυπα κίνησης κωδικοποιούνται στο σύστημα δημιουργώντας Markov μοντέλα. Άπαξ και η εκπαίδευση θεωρείται ότι είναι επαρκής, το σύστημα είναι έτοιμο για πρόβλεψη, και δεν επιτρέπεται καμία περεταίρω τροποποίηση. Αυτή η περίπτωση είναι αποδεκτή δεδομένου ότι τα πρότυπα κίνησης του κινούμενου αντικειμένου αντιστοιχούν σε καλώς-ορισμένα όρια. Ωστόσο, σε πολλές πραγματικές καταστάσεις, τα πρότυπα κίνησης δεν είναι γνωστά, ή τουλάχιστον, διαθέσιμα κατά τη φάση της εκπαίδευσης. Η διαδικασία εκπαίδευσης περιορίζει την ικανότητα του συστήματος να προσαρμόζει τη βάση γνώσης του σε νέα πρότυπα.

7. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ

7.1. Αποτελεσματική Πρόβλεψη Θέσης σε Κυψελωτά Δίκτυα

Στη δημοσίευση [4] (βλέπε Ενότητα 2) προτείναμε αποδοτικά μοντέλα LP που βασίζονται σε αλγορίθμους μηχανικής μάθησης για πρόβλεψη τροχιάς. Συγκεκριμένα, ο προτεινόμενος ταξινομητής χωρικού πλαισίου και ένας βραχυπρόθεσμος προγνώστης για την πρόβλεψη της θέσης ενός κινητού χρήστη σε κυψελωτά δίκτυα εκμεταλλεύονται (α) την τρέχουσα θέση και κατεύθυνση του χρήστη, (β) την ιστορία των τροχιών του χρήστη, και, (γ) τη περιβάλλουσα πληροφορία θέσης. Σχεδιάσαμε, υλοποιήσαμε και αξιολογήσαμε διαφορετικές παραλλαγές του προτεινόμενου LP. Κάθε παραλλαγή εκμεταλλεύεται διαφορετικά την απορρέουσα γνώση της κινητής συμπεριφοράς του χρήστη (δηλαδή, μάκρο και μικρο LP). Ορίζουμε τις παραμέτρους του βραχυπρόθεσμου LP και εισάγουμε συγκεκριμένες μετρικές για να αξιολογήσουμε την ικανότητα των σωστών προβλέψεων και την αποδοτικότητα στην διαδικασία πρόβλεψης. Επιπρόσθετα, συγκρίνουμε τον LP μας (δηλαδή, όλες τις αντίστοιχες παραλλαγές) με δημοφιλείς προγνώστες που αναφέρονται στη σχετική επιστημονική λογοτεχνία. Τέτοιοι προγνώστες στηρίζονται επίσης σε αλγορίθμους μηχανικής μάθησης. Προσομοιώσεις με συνθετικά και πραγματικά δεδομένα κίνησης δείχνουν ότι ο προτεινόμενος βραχυπρόθεσμος MiD προγνώστης επιτυγχάνει υψηλή απόδοση πρόβλεψης και ακρίβεια, έτσι, καθιστά τους LPs κατάλληλους για προηγμένες εφαρμογές επίγνωσης-πλαισίου.

7.2. Ένας Προσαρμοστικός Αλγόριθμος Μηχανικής Μάθησης για την Πρόβλεψη Θέσης

Στη δημοσίευση [5] (βλέπε Ενότητα 3) παρουσιάσαμε πώς η μηχανική μάθηση μπορεί να εφαρμοστεί στην πρόβλεψη θέσης των κινητών εφαρμογών επίγνωσης-πλαισίου. Συγκεκριμένα, προτείναμε έναν προσαρμοστικό αλγόριθμο μηχανικής μάθησης για πρόβλεψη θέσης χρησιμοποιώντας ART (ένα ειδικό τοπικό μοντέλο νευρωνικών δικτύων). Εισάγαμε δύο μεθόδους μάθησης: μία με μη-ενισχυτική μάθηση και μία με ενισχυτική μάθηση. Επιπρόσθετα, ασχοληθήκαμε με δύο μεθόδους εκπαίδευσης για κάθε μέθοδο μάθησης: στην εποπτευόμενη μέθοδο το μοντέλο χρησιμοποιεί δεδομένα εκπαίδευσης με σκοπό να κάνει ταξινόμηση και στη μηδενικής-γνώσης μέθοδο το μοντέλο μαθαίνει αυξητικά από μη-επιτυχείς προβλέψεις. Αξιολογήσαμε τα μοντέλα μας (εκδόσεις του προτεινόμενου αλγορίθμου) με διαφορετικές χωρικές και χρονικές παραμέτρους. Εξετάσαμε το κόστος αποθήκευσης των βάσεων γνώσης (δηλαδή, ομάδες που προέκυψαν) και τις μετρικές ακρίβειας (ακρίβεια πρόβλεψης). Τα συμπεράσματά μας δηλώνουν ότι το C-RLnT μοντέλο ταιριάζει περισσότερο στα συστήματα επίγνωσης-πλαισίου. Το πλεονέκτημα του C-RLnT είναι ότι: (1) δεν απαιτεί προ-υπάρχουσα γνώση της κινητής συμπεριφοράς του χρήστη με σκοπό να προβλέπει μελλοντικές κινήσεις, (2) προσαρμόζει την on-line βάση γνώσης σε νέα πρότυπα και (3) δεν καταναλώνει πολύ μνήμη για να αποθηκεύσει τις ομάδες που προκύπτουν. Για αυτό το λόγο, το C-RLnT είναι αρκετά χρήσιμο στις εφαρμογές επίγνωσης-πλαισίου όπου δεν είναι διαθέσιμη καμία προ-υπάρχουσα γνώση σχετικά με το πλαίσιο του χρήστη. Επιπρόσθετα, μέσω πειραμάτων, αποφασίσαμε σε ποια τιμή της ευαισθησίας επιτυγχάνεται η κατάλληλη ακρίβεια σε σχέση με τους περιορισμούς της μνήμης και το σφάλμα πρόβλεψης.

7.3. Ένα Προσαρμοστικό Μοντέλο Πρόβλεψης Θέσης βασισμένο σε Ασαφή Έλεγχο

Στη δημοσίευση [6] (βλέπε Ενότητα 4) μελετήσαμε το στοιχείο της προδραστηριότητας των κινητών εφαρμογών που εξαρτώνται από τη θέση και παρουσιάζουμε μία προσέγγιση για πρόβλεψη κίνησης εκμεταλλευόμενοι μόνο την πρόσφατη γνώση της κίνησης του χρήστη. Προτείναμε έναν βραχείας-μνήμης προσαρμοστικό LP να αντιμετωπίσει το πρόβλημα της πρόβλεψης κίνησης υπό την απουσία εκτενούς ιστορικής πληροφορίας. Η πρόβλεψη θέσης του προτεινόμενου LP επιτυγχάνεται από ένα τοπικό μοντέλο γραμμικής παλινδρόμησης, ενώ η ικανότητα προσαρμογής επιτυγχάνεται μέσω ενός ασαφούς PI_m ελεγκτή. Ένας τέτοιος ελεγκτής παράγει σήματα ελέγχου για την εκτίμηση του καλύτερου μεγέθους του παραθύρου της κινητής ιστορίας, προσπαθώντας να ελαχιστοποιήσει το σφάλμα πρόβλεψης θέσης. Πειραματιστήκαμε με πραγματικά GPS ίχνη και εξετάσαμε την προβλεψιμότητα και προσαρμοστικότητα της συμπεριφοράς του LP μας. Ο LP ελέγχει δυναμικά το μέγεθος της ιστορικής συμπεριφοράς κίνησης, μέσω των παραγόμενων σημάτων ελέγχου, για τις αστικές, περιαστικές και δι-αστικές τροχιές. Ο LP σταθεροποιείται σε μικρά m^* όταν έρχεται αντιμέτωπος με ξαφνικές αλλαγές στην κίνηση. Αυτό οδηγεί στην ελαχιστοποίηση της πιθανότητας του προκαλούμενου θορύβου στο τοπικό μοντέλο παλινδρόμησης. Επιπρόσθετα, μπορούμε να συμπεράνουμε ότι τα σήματα ελέγχου είναι πιο επιθετικά στην περίπτωση της αστικής τροχιάς από ότι στην δι-αστική τροχιά δεδομένου ότι η πρώτη κίνηση ενσωματώνει απότομες μεταβολές στην κίνηση ενώ η δεύτερη όχι. Τελικά δείχνουμε ότι ο LP επιτυγχάνει γρήγορη προσαρμογή σε απότομες αλλαγές ειδικά όταν ο χρήστης αλλάζει συμπεριφορά κίνησης (δηλαδή, μετάβαση μεταξύ διαφορετικών τύπων τροχιών).

7.4. Ευφυής Ταξινόμηση Τροχιάς για Βελτιωμένη Πρόβλεψη Κίνησης

Στη δημοσίευση [7] (βλέπε Ενότητα 5) προτείνουμε ένα μοντέλο, το οποίο βασίζεται σε ακολουθιακή ταξινόμηση τροχιάς και σε μία διαδικασία μείωσης της χωρικής διακύμανσης για την επίτευξη πρόβλεψης της κίνησης των κινούμενων αντικειμένων. Το προτεινόμενο μοντέλο αντιμετωπίζει θορυβώδη πρότυπα κίνησης λόγω των τυχαίων αποκλίσεων από τα προηγούμενα παρατηρούμενα πρότυπα, το οποίο επηρεάζει αρνητικά την ικανότητα πρόβλεψης (ακρίβεια) του συστήματος ταξινόμησης. Εισάγουμε μία διαδικασία μείωσης χωρικής διακύμανσης (SVRP), στην οποία η υποκείμενη βάση προτύπων διατηρείται τόσο «καθαρή» και συμπαγής όσο αυτό είναι δυνατόν με το να διατηρεί αυτά τα πρότυπα με περιορισμένη χωρική διακύμανση. Προσομοιώσεις έχουν γίνει για την αξιολόγηση της απόδοσης του μοντέλου μας με πραγματικά και (ήμι) συνθετικά ίχνη κινούμενων αντικειμένων μέσω της πιθανότητας των επιτυχών προβλέψεων και των μετρικών απόδοσης πρόβλεψης. Επιπρόσθετα, αξιολογούμε το προτεινόμενο μοντέλο με άλλα αυξητικά μοντέλα ταξινόμησης και πρόβλεψης τροχιάς. Συμπεραίνουμε ότι η εισαγωγή της διαδικασίας SVRP, επιτυγχάνει υψηλά επίπεδα πρόβλεψης καθώς και αποτελεσματικής αποθήκευσής των δεδομένων της χωροχρονικής πληροφορίας στην πρόβλεψη κίνησης.

7.5. Μελλοντική Έρευνα

Στα αμέσως επόμενα σχέδια μας αποβλέπουμε σε δύο κατευθύνσεις: (α) βελτίωση των ήδη υπάρχοντων μοντέλων με επιπλέον πληροφορία και καθορισμό νέων στόχων σχετικά με την πρόβλεψη κίνησης, και (β) στη δημιουργία ασφαλών αλγορίθμων και πολιτικών για την διασφάλιση της ιδιωτικής ζωής των χρηστών από κακόβουλες οντότητες. Συγκεκριμένα, σκοπεύουμε να εμπλουτίσουμε τον MiD προγνώστη θέσης με

χρονική πληροφορία, έτσι ώστε να έχουμε μία επιπλέον εκτίμηση του χρονικού πλαισίου μέσα στο οποίο κινείται ο χρήστης και κατά συνέπεια να εξάγουμε συμπεράσματα για συγκεκριμένες χρονικές περιόδους κίνησης. Επιπλέον, στην επιστημονική λογοτεχνία των τοπικών μοντέλων νευρωνικών δικτύων υπάρχουν άλλα μοντέλα (δηλαδή, Self-Organizing Maps) που θα μπορούσαν να υλοποιηθούν και να αξιολογηθούν με τον C-RLnT μέσω των απαιτήσεων της βάσης γνώσης, ακρίβειας της πρόβλεψης θέσης και πιθανής προσαρμοστικότητας στο μοντέλο που δημιουργείται. Επιπρόσθετα, στην περίπτωση του ασαφούς ελεγκτή PI_m θέλουμε να εστιάσουμε στην υιοθέτηση του από άλλους προγνώστες θέσης που εκμεταλλεύονται την ιστορική πληροφορία κίνησης. Επιπλέον, έχουμε σκοπό να εξετάσουμε έναν βελτιωμένο PI_m ελεγκτή ο οποίος θα εκτιμά δυναμικά την καλύτερη παράμετρο εξομάλυνσης λ για το θεωρούμενο τοπικό μοντέλο γραμμικής παλινδρόμησης. Επίσης, στην περίπτωση του ακολουθιακού ταξινομητή βέλτιστου τερματισμού στοχεύουμε στην ενίσχυση της SVRP με μετρικές βασιζόμενες στην εντροπία για να διατηρήσουμε μία «καθαρή» βάση γνώσης.

Όσον αφορά την διασφάλιση της ιδιωτικής ζωής [59], [60] ερχόμαστε αντιμέτωποι με το εξής πρόβλημα που πρέπει να λυθεί. Η πληροφορία θέσης, με τη μορφή μίας συγκεκριμένης θέσης ή μίας τροχιάς κίνησης, μπορεί πραγματικά να οδηγήσει σε προσωπική ταυτοποίηση, ως εκ τούτου αποκαλύπτοντας τη σχέση μεταξύ του ψευδώνυμου και του πραγματικού προσώπου. Για παράδειγμα, μία αίτηση υπηρεσίας η οποία περιέχει πληροφορία θέσης της μορφής ακριβών συντεταγμένων ενός ιδιωτικού σπιτιού παρέχει ικανή πληροφορία για να αναγνωριστεί προσωπικά ο ιδιοκτήτης του σπιτιού δεδομένου ότι μπορεί να γίνει διασταύρωση των συντεταγμένων με τις διευθύνσεις των σπιτιών από τον τηλεφωνικό κατάλογο (ή αντίστοιχες πηγές), με αποτέλεσμα να αποκαλυφθούν οι άνθρωποι που ζουν εκεί. Επίσης αν γίνουν πολλές αιτήσεις από την ίδια θέση με το ίδιο ψευδώνυμο, είναι πολύ πιθανό ότι ο χρήστης με αυτό το ψευδώνυμο να είναι ένα μέλος της οικογένειας. Με βάση αυτό το έναυσμα, αλλά και άλλων όμοιων στην επιστημονική λογοτεχνία, τίθεται ένα θέμα σχετικά με τους αλγόριθμους και τις πολιτικές που πρέπει να υλοποιηθούν και να εφαρμοστούν για να διασφαλιστούν τέτοιες καταστάσεις διείσδυσης στην προσωπική ζωή των χρηστών.

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

ART	Adaptive Resonance Theory – Θεωρία Προσαρμοστικού Συντονισμού
Classification	Ταξινόμηση
clustering	Ομαδοποίηση
Context	Πλαίσιο
Context Adaptation	Προσαρμοστικότητα Πλαισίου
Context Aware Computing	Υπολογιστική Ικανότητα με Επίγνωση Πλαισίου
Context Awareness	Επίγνωση του Πλαισίου
Context Classification	Ταξινόμηση Πλαισίου
Context Clustering	Ομαδοποίηση Πλαισίου
Context Model	Μοντέλο Αναπαράστασης Πλαισίου
Context Prediction	Πρόβλεψη Πλαισίου
Context-Aware Application	Εφαρμογή Επίγνωσης Πλαισίου
Contextual Information	Πληροφορία Πλαισίου
Defuzzification	Αποασαφοποίηση
Fuzzification	Ασαφοποίηση
Fuzzy Logic	Ασαφής Λογική
Fuzzy Controller	Ασαφής Ελεγκτής
Historical Context	Πλαίσιο Παρελθόντος
Location-Based Services	Υπηρεσίες Βασισμένες στην Θέση
Misclassification Error	Σφάλμα Ταξινόμησης
OST	Optimal Stopping Theory – Θεωρία Βέλτιστης Παύσης
Pervasive Computing	Διάχυτος Υπολογισμός
Self Organizing Maps	Αυτό-Οργανωμένη Δομή Προτύπων
Supervised Learning	Εποπτευόμενη Μάθηση
Unsupervised Learning	Μη Εποπτευόμενη Μάθηση

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΟΝΥΜΙΑ

TC	Trajectory Classifier – Ταξινομητής τροχιάς
LP	Location Predictor – Προγνώστης Θέσης
Μάκρο LP	Καθολικός LP
Μίκρο LP	Τοπικός LP
RL	Reinforcement Learning – Ενισχυτική Μάθηση
nRL	Not Reinforcement Learning – Μη Ενισχυτική Μάθηση
T	Training – Εκπαιδευόμενο
nT	Non Training – Μη Εκπαιδευόμενο

ΠΑΡΑΡΤΗΜΑ Ι - ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Ι.1. Ορισμός της Μηχανικής Μάθησης

Με την πρόοδο της τεχνολογίας των ηλεκτρονικών υπολογιστών έχουμε σήμερα τη δυνατότητα να αποθηκεύουμε και να επεξεργαζόμαστε μεγάλες ποσότητες δεδομένων, καθώς και να τις προσπελαύνουμε από απομεμακρυσμένες τοποθεσίες μέσω ενός δικτύου υπολογιστών. Οι σύγχρονες συσκευές λήψης δεδομένων είναι ψηφιακές, γεγονός που μας επιτρέπει να καταγράφουμε αξιόπιστα το πλήθος των δεδομένων. Σκεφτείτε, για παράδειγμα, μια αλυσίδα σουπέρ μάρκετ που έχει εκατοντάδες καταστήματα σε όλη την χώρα που πωλούν χιλιάδες προϊόντα σε εκατομμύρια πελάτες. Τα τερματικά στα σημεία πώλησης καταγράφουν τα στοιχεία κάθε συναλλαγής, όπως: την ημερομηνία, τον κωδικό αναγνώρισης του πελάτη, τα εμπορεύματα που αγόρασε και την αξία τους, το σύνολο των χρημάτων που δαπανώνται, και ούτω καθεξής. Η αποθήκευση αυτών των δεδομένων σε ημερήσια βάση απαιτεί κατά κανόνα πλήθος από gigabytes. Το σύνολο αυτών των δεδομένων είναι χρήσιμο μόνο όταν αναλύεται και μετατρέπεται σε πληροφορίες που μπορούν να χρησιμοποιηθούν για περαιτέρω έρευνα, για παράδειγμα, να κάνουμε προβλέψεις.

Δεν γνωρίζουμε ακριβώς ποιοι άνθρωποι είναι πιθανότερο να αγοράσουν ένα συγκεκριμένο προϊόν, ή ποιον συγγραφέα θα προτείναμε στα άτομα που απολαμβάνουν την ανάγνωση του Γιάννη Ρίτσου. Εάν ξέραμε, δεν θα χρειαζόμασταν καμία ανάλυση των δεδομένων, σε αυτήν την περίπτωση απλά θα προχωρούσαμε στη σύνταξη του αντίστοιχου κώδικα. Αλλά επειδή δεν το γνωρίζουμε, το μόνο που μπορούμε να κάνουμε είναι να συλλέξουμε τα κατάλληλα δεδομένα με την απώτερη ελπίδα να εξαγάγουμε τις καλύτερες δυνατές απαντήσεις γι αυτά, και για άλλα παρόμοια, ερωτήματα.

Δεν πιστεύουμε ότι υπάρχει μια διαδικασία που να εξηγεί επακριβώς τα δεδομένα που παρατηρούμε. Αν και δεν γνωρίζουμε τις λεπτομέρειες της διαδικασίας στην οποία στηρίζεται η δημιουργία των δεδομένων, για παράδειγμα η συμπεριφορά του καταναλωτή, ξέρουμε διαισθητικά ότι δεν είναι εντελώς τυχαία. Οι άνθρωποι δεν πηγαίνουν στο σουπέρ μάρκετ για να αγοράσουν πράγματα στην τύχη. Όταν αγοράζουν μπύρα, αγοράζουν και πατατάκια. Επίσης αγοράζουν παγωτά το καλοκαίρι και ζεστά ροφήματα το χειμώνα. Υπάρχουν ορισμένα πρότυπα όσον αφορά τα δεδομένα.

Είναι πολύ πιθανό να μην είμαστε σε θέση να προσδιορίσουμε τη διαδικασία επακριβώς, ωστόσο πιστεύουμε ότι μπορούμε να κατασκευάσουμε μια *ικανοποιητική και χρηστική προσέγγιση*. Η εν λόγω προσέγγιση δεν μπορεί να εξηγήσει τα πάντα, αλλά πιθανότατα να είναι εφαρμόσιμη για μία σημαντική ποσότητα δεδομένων. Πιστεύουμε ότι αν και ο εντοπισμός της πλήρους διαδικασίας μπορεί να μην είναι δυνατός, ωστόσο θα μπορούν να εντοπιστούν ορισμένα πρότυπα ή ομοιότητες. Αυτή είναι η θέση της Μηχανικής Μάθησης [8]. Αυτά τα πρότυπα μπορούν να μας βοηθήσουν να κατανοήσουμε τη διαδικασία, ή επίσης μπορούμε να τα χρησιμοποιήσουμε για να κάνουμε προβλέψεις. Προϋποθέτοντας ότι το μέλλον, τουλάχιστον το εγγύς μέλλον, δεν θα είναι πολύ διαφορετικό από το πρόσφατο παρελθόν, όταν δηλαδή συλλέχθηκε το δείγμα δεδομένων, μπορούμε να αναμένουμε ότι οι μελλοντικές προβλέψεις θα είναι σωστές.

Η εφαρμογή των μεθόδων της μηχανικής μάθησης σε μεγάλες βάσεις δεδομένων ονομάζεται data mining. Η αναλογία είναι η εξής: όπως ένας μεγάλος όγκος πρώτων υλών που εξάγεται από ένα ορυχείο και στη συνέχεια υποβάλλεται σε επεξεργασία οδηγεί σε μια μικρή ποσότητα από πολύτιμο υλικό, ομοίως στην εξόρυξη δεδομένων, η επεξεργασία ενός μεγάλου όγκου δεδομένων οδηγεί στην κατασκευή ενός απλού μοντέλου με πολύτιμη χρήση, για παράδειγμα, ένα μοντέλο που έχει υψηλή διαγνωστική

ακρίβεια. Υπάρχει πληθώρα εφαρμογών της μηχανικής μάθησης: εκτός από την λιανική πώληση, χρησιμοποιείται σε τράπεζες όπου αναλύονται δεδομένα του παρελθόντος με σκοπό τη δημιουργία μοντέλων για χρήση σε πιστοληπτικές εφαρμογές, στην ανίχνευση απάτης, και στο χρηματιστήριο. Στη βιομηχανία, τα μαθησιακά μοντέλα που δημιουργούνται από τη μηχανική μάθηση χρησιμοποιούνται για τη βελτιστοποίηση, τον έλεγχο και την αντιμετώπιση προβλημάτων στη διαδικασία της παραγωγής. Στην ιατρική, τα μαθησιακά μοντέλα χρησιμοποιούνται για την επίτευξη ιατρικής διάγνωσης. Στον τομέα των τηλεπικοινωνιών, τα πρότυπα των κλήσεων αναλύονται με σκοπό τη βελτιστοποίηση του δικτύου και τη μεγιστοποίηση της ποιότητας των παρεχόμενων υπηρεσιών. Στις επιστήμες γενικότερα, μεγάλες ποσότητες δεδομένων στη φυσική, την αστρονομία και τη βιολογία μπορούν να αναλυθούν αρκετά γρήγορα από τους υπολογιστές. Το World Wide Web είναι τεράστιο και συνεχώς μεγαλώνει, με αποτέλεσμα η αναζήτηση των σχετικών πληροφοριών να μην μπορεί να γίνει χειροκίνητα.

Όμως, η μηχανική μάθηση δεν είναι απλώς ένα πρόβλημα βάσεων δεδομένων, είναι επίσης ένα μέρος της τεχνητής νοημοσύνης. Για να είναι ευφυές, ένα σύστημα που βρίσκεται σε ένα διαρκώς μεταβαλλόμενο περιβάλλον θα πρέπει να έχει τη δυνατότητα να μαθαίνει. Εάν το σύστημα μπορεί να μαθαίνει και να προσαρμόζεται στις αλλαγές του περιβάλλοντος, ο σχεδιαστής του συστήματος δεν χρειάζεται πλέον να προβλέπει και να παρέχει λύσεις για όλες τις πιθανές καταστάσεις που μπορεί να προκύψουν.

Η μηχανική μάθηση, μπορεί επίσης, να μας βοηθήσει να βρούμε λύσεις σε πολλά προβλήματα όπως στην όραση, στην αναγνώριση ομιλίας, και στη ρομποτική. Ας πάρουμε για παράδειγμα την αναγνώριση προσώπων: αυτή είναι μία διαδικασία που κάνουμε αβίαστα, κάθε μέρα αναγνωρίζουμε τα μέλη της οικογένειας μας και τους φίλους μας κοιτάζοντας τα πρόσωπά τους ή τις φωτογραφίες τους, παρά τις διαφορές που δημιουργούνται από το φωτισμό, το στυλ των μαλλιών, και ούτω καθεξής. Το θέμα είναι ότι εμείς κάνουμε αυτή τη διαδικασία ασυναίσθητα και συνεπώς δεν είμαστε σε θέση να εξηγήσουμε πώς γίνεται. Επειδή δεν είμαστε σε θέση να εξηγήσουμε την τεχνογνωσία μας, δεν μπορούμε να γράψουμε αυτή τη διαδικασία σε ένα πρόγραμμα στον υπολογιστή. Ταυτόχρονα όμως, γνωρίζουμε ότι μια εικόνα προσώπου δεν είναι απλά μια τυχαία συλλογή από pixels, επειδή ένα πρόσωπο έχει δομή. Είναι συμμετρικό. Υπάρχουν τα μάτια, η μύτη, το στόμα, που βρίσκονται σε ορισμένα σημεία στο πρόσωπο. Το πρόσωπο του κάθε ανθρώπου είναι ένα σχέδιο που αποτελείται από ένα συγκεκριμένο συνδυασμό αυτών. Αναλύοντας δείγματα από εικόνες ενός προσώπου, ένα μαθησιακό μοντέλο συλλαμβάνει το πρότυπο του προσώπου και στη συνέχεια το αναγνωρίζει ελέγχοντας αυτό το πρότυπο με μια συγκεκριμένη εικόνα που του δίνεται. Αυτό είναι ένα παράδειγμα της *αναγνώρισης προτύπων*.

Θα μπορούσαμε να ορίσουμε τη μηχανική μάθηση ως τη διαδικασία προγραμματισμού των υπολογιστών με σκοπό τη βελτιστοποίηση ενός σαφώς καθορισμένου κριτηρίου απόδοσης με βάση τα εισερχόμενα δεδομένα ή την εμπειρία του παρελθόντος. Έχουμε δηλαδή ένα μοντέλο που ορίζεται μέσω καλά ορισμένων παραμέτρων, και η διαδικασία της μάθησης ανάγεται στην εκτέλεση ενός προγράμματος σε υπολογιστή με σκοπό τη βελτιστοποίηση των παραμέτρων του μοντέλου χρησιμοποιώντας τα δεδομένα εκπαίδευσης ή την εμπειρία του παρελθόντος. Το μοντέλο μπορεί να είναι *προγνωστικό* για να κάνει προβλέψεις για το μέλλον, ή *περιγραφικό* για να αποκτήσουμε γνώσεις από τα δεδομένα, ή και τα δύο.

Η μηχανική μάθηση χρησιμοποιεί τη θεωρία της στατιστικής για τη δημιουργία μαθηματικών μοντέλων, όπου το κυρίως ζητούμενο είναι η επίτευξη απόφασης από ένα δείγμα δεδομένων. Ο ρόλος της επιστήμης των υπολογιστών είναι διπτός: πρώτον, στην εκπαίδευση, όπου χρειαζόμαστε αποτελεσματικούς αλγόριθμους για την επίλυση

του προβλήματος της βελτιστοποίησης, καθώς και την αποθήκευση και επεξεργασία του μεγάλου όγκου πληροφορίας που ανακύπτει. Δεύτερον, εφόσον έχει δημιουργηθεί το μοντέλο, ενδιαφερόμαστε για την αναπαράσταση και την αλγοριθμική λύση που απαιτείται για την επίτευξη απόφασης. Σε ορισμένες εφαρμογές, η απόδοση του μαθησιακού μοντέλου, δηλαδή, η πολυπλοκότητα του σε χώρο και χρόνο, μπορεί να είναι εξίσου σημαντική με την προγνωστική ακρίβειά του.

Στη συνέχεια του παραρτήματος θα παρουσιάσουμε ορισμένες εφαρμογές της μηχανικής μάθησης, όπως για παράδειγμα, την εποπτευόμενη μάθηση, την μη εποπτευόμενη μάθηση και την ενισχυτική μάθηση που είναι και τα κύρια εργαλεία που χρησιμοποιήθηκαν στην παρούσα διδακτορική διατριβή.

1.2. Εποπτευόμενη μάθηση

Θα παρουσιάσουμε την έννοια της εποπτευόμενης μάθησης αρχίζοντας από την πιο απλή περίπτωση, η οποία είναι η εκμάθηση μιας κλάσης από θετικά και αρνητικά παραδείγματα της. Στη συνέχεια θα γενικευθούμε και θα συζητήσουμε την περίπτωση πολλαπλών κλάσεων. Ακολούθως θα παρουσιάσουμε την παλινδρόμηση, όπου οι έξοδοι είναι συνεχείς τιμές.

1.2.1. Ταξινόμηση

Ας υποθέσουμε ότι θέλουμε να μάθουμε την *κλάση*, C , ενός "οικογενειακού αυτοκινήτου". Έχουμε ένα σύνολο από παραδείγματα αυτοκινήτων, και έχουμε μια ομάδα ανθρώπων όπου κάνουμε μια έρευνα σε οποίους δείχνουμε αυτά τα αυτοκίνητα. Οι άνθρωποι κοιτούν τα αυτοκίνητα που θα τους δείξουμε και την ετικέτα τους, τα αυτοκίνητα τα οποία πιστεύουν ότι είναι οικογενειακά αυτοκίνητα αποτελούν τα *θετικά παραδείγματα* και τα άλλα αυτοκίνητα είναι τα *αρνητικά παραδείγματα*. Η εκμάθηση μιας κλάσης είναι η εξεύρεση μιας περιγραφής που είναι κοινή για όλα τα θετικά παραδείγματα και για κανένα από τα αρνητικά παραδείγματα. Με αυτόν τον τρόπο, μπορούμε να κάνουμε μια πρόβλεψη: αν μας δοθεί ένα αυτοκίνητο που δεν έχουμε ξαναδεί, τότε συγκρίνοντας το με την περιγραφή που έχουμε μάθει, θα είμαστε σε θέση να αποφανθούμε αν είναι ένα οικογενειακό αυτοκίνητο ή όχι. Επίσης θα μπορούσαμε να κάνουμε εξαγωγή γνώσης: για παράδειγμα η μελέτη αυτή μπορεί να χρηματοδοτηθεί από μία αυτοκινητοβιομηχανία, με απώτερο στόχο να είμαστε σε θέση να καταλάβουμε τι αναμένουν οι άνθρωποι από ένα οικογενειακό αυτοκίνητο.

Μετά από κάποιες συζητήσεις με εμπειρογνώμονες στον τομέα αυτό, ας πούμε ότι έχουμε καταλήξει στο συμπέρασμα ότι, μεταξύ όλων των χαρακτηριστικών ενός αυτοκινήτου, τα χαρακτηριστικά που διαχωρίζουν ένα οικογενειακό αυτοκίνητο από τα άλλα αυτοκίνητα είναι η τιμή του και η ισχύς του κινητήρα του. Τα δύο αυτά χαρακτηριστικά είναι οι *είσοδοι* στον ταξινομητή. Θα πρέπει να κάνουμε την παραδοχή ότι για την συγκεκριμένη *αναπαράσταση εισόδου*, θα αγνοήσουμε διάφορα άλλα χαρακτηριστικά ως άσχετα με την διαδικασία της ταξινόμησης. Αν και κάποιος θα μπορούσε να σκεφτεί και άλλα χαρακτηριστικά, όπως το πλήθος των θέσεων και το χρώμα, που θα μπορούσαν να είναι σημαντικά για τη διάκριση μεταξύ των τύπων των αυτοκινήτων, θα εξετάσουμε μόνο την τιμή και την ισχύ του κινητήρα για να δουλέψουμε πάνω σε αυτό το απλό παράδειγμα.

Ας δηλώσουμε την τιμή ως το πρώτο χαρακτηριστικό εισόδου x_1 (π.χ., στην ΕΕ, ευρώ) και την ισχύ του κινητήρα, ως το δεύτερο χαρακτηριστικό x_2 (π.χ., ο όγκος του κινητήρα σε κυβικά εκατοστά). Έτσι, κάθε αυτοκίνητο αναπαριστάται κάνοντας χρήση δύο αριθμητικών τιμών

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (1.1)$$

και η ετικέτα του υποδηλώνει τον τύπο του

$$r = \begin{cases} 1 & \text{αν } \mathbf{x} \text{ είναι ένα θετικό παράδειγμα} \\ 0 & \text{αν } \mathbf{x} \text{ είναι ένα αρνητικό παράδειγμα} \end{cases} \quad (1.2)$$

Κάθε αυτοκίνητο αναπαριστάται από αυτό το διατεταγμένο ζεύγος (\mathbf{x}, r) , και το σύνολο εκπαίδευσης περιέχει N τέτοια παραδείγματα

$$X = \{\mathbf{x}^t, r^t\}_{t=1}^N \quad (1.3)$$

Όπου, t , είναι ένας δείκτης που αναφέρεται στα διάφορα παραδείγματα του συνόλου, και δεν αντιπροσωπεύει χρόνο ή κάποια αντίστοιχη διάταξη.

Τα δεδομένα εκπαίδευσης μας μπορούν τώρα να καταγραφούν στο διδιάστατο (x_1, x_2) χώρο όπου κάθε παράδειγμα t είναι ένα σημείο δεδομένων με συντεταγμένες (x_1^t, x_2^t) και ο τύπος του, δηλαδή, θετικός ή αρνητικός, δίνεται από το r^t .

Μετά από περαιτέρω συζητήσεις με τον εμπειρογνώμονα και την ανάλυση των δεδομένων, μπορεί να έχουμε λόγο να πιστεύουμε ότι ένα αυτοκίνητο είναι οικογενειακό αυτοκίνητο, αν η τιμή του και η ισχύς του κινητήρα κινείται είναι σε μια ορισμένη περιοχή

$$(p_1 \leq \text{τιμή} \leq p_2) \text{ και } (e_1 \leq \text{ισχύς} \leq e_2) \quad (1.4)$$

Για κατάλληλες τιμές των p_1, p_2, e_1 και e_2 , η εξίσωση 1.4 υποθέτει ότι η κλάση C είναι ένα ορθογώνιο στο χώρο τιμή-ισχύς του κινητήρα.

Η εξίσωση 1.4 καθορίζει την κλάση υπόθεσης H , που στην ουσία είναι το σύνολο των ορθογωνίων, από την οποία πιστεύουμε ότι προέρχεται η κλάση C . Ο μαθησιακός αλγόριθμος θα πρέπει να βρει μια συγκεκριμένη υπόθεση, $h \in H$, που να προσεγγίσει την κλάση C όσο το δυνατόν περισσότερο.

Αν και ο εμπειρογνώμονας καθορίζει αυτή την κλάση υπόθεσης, δεν μπορεί να πει με βεβαιότητα ποιες είναι οι τιμές των παραμέτρων. Δηλαδή, αν και επιλέγουμε μία κλάση υπόθεσης H , δεν ξέρουμε ποια συγκεκριμένη $h \in H$ είναι ίση ή πλησιέστερη, στην κλάση C . Αλλά από τη στιγμή που θα εστιάσουμε την προσοχή μας σε αυτή την κλάση υπόθεσης, η μάθηση της κλάσης περιορίζεται στο ευκολότερο πρόβλημα της εύρεσης των τεσσάρων παραμέτρων που καθορίζουν την h .

Ο στόχος είναι να βρεθεί η $h \in H$ που είναι όσο το δυνατόν πιο όμοια με την C . Ας πούμε ότι η υπόθεση h κάνει μια πρόβλεψη για ένα παράδειγμα \mathbf{x} τέτοια ώστε

$$h(\mathbf{x}) = \begin{cases} 1 & \text{αν η } h \text{ ταξινομεί το } \mathbf{x} \text{ σαν ένα θετικό παράδειγμα} \\ 0 & \text{αν η } h \text{ ταξινομεί το } \mathbf{x} \text{ σαν ένα αρνητικό παράδειγμα} \end{cases} \quad (1.5)$$

Στην πραγματική ζωή δεν ξέρουμε την $C(\mathbf{x})$, οπότε δεν μπορούμε να αξιολογήσουμε πόσο καλά η $h(\mathbf{x})$ ταιριάζει με την $C(\mathbf{x})$. Αυτό που έχουμε είναι ένα σύνολο εκπαίδευσης X , το οποίο είναι ένα μικρό υποσύνολο του συνόλου όλων των πιθανών παραδειγμάτων \mathbf{x} . Το *εμπειρικό σφάλμα* είναι το ποσοστό των παραδειγμάτων εκπαίδευσης όπου οι *προβλέψεις* της h δεν ταιριάζουν με τις απαιτούμενες τιμές που δίνονται στο X . Το σφάλμα της υπόθεσης h δεδομένου του συνόλου εκπαίδευσης X είναι

$$E(h|X) = \sum_{t=1}^N 1(h(\mathbf{x}^t) \neq r^t) \quad (1.6)$$

Όπου $1(a \neq b)$ είναι 1 αν $(a \neq b)$ και είναι 0 αν $(a = b)$.

Στο παράδειγμά μας, η κλάση υπόθεσης H είναι το σύνολο όλων των πιθανών ορθογωνίων. Κάθε τετράδα $(p_1^h, p_2^h, e_1^h, e_2^h)$ ορίζει μία υπόθεση, $h \in H$, και πρέπει να επιλεγεί η καλύτερη. Συγκεκριμένα θα πρέπει να βρεθούν οι τιμές αυτών των τεσσάρων παραμέτρων, δεδομένου του συνόλου εκπαίδευσης, έτσι ώστε να περιλαμβάνουν όλα τα θετικά παραδείγματα και κανένα από τα αρνητικά παραδείγματα. Πρέπει να σημειώσουμε ότι αν x_1 και x_2 είναι πραγματικές μεταβλητές, υπάρχουν απείρως πολλές τέτοιες υποθέσεις που ικανοποιούνται, δηλαδή, για τις οποίες το σφάλμα, E , είναι μηδέν. Ωστόσο ένα μελλοντικό παράδειγμα, κάπου κοντά στο όριο μεταξύ των θετικών και αρνητικών παραδειγμάτων, θα κάνει τις διαφορετικές υποθέσεις να δώσουν διαφορετικές προβλέψεις. Αυτό είναι το πρόβλημα της γενίκευσης – το οποίο είναι, πόσο καλά η υπόθεσή μας θα ταξινομεί σωστά μελλοντικά παραδείγματα που δεν είναι μέρη του συνόλου εκπαίδευσης.

Μια πιθανότητα είναι να βρεθεί η *πιο ειδική υπόθεση*, S , που είναι το μικρότερο ορθογώνιο που περιλαμβάνει όλα τα θετικά παραδείγματα και κανένα από τα αρνητικά παραδείγματα. Αυτή μας δίνει μία υπόθεση, $h = S$, ως την συμπερασματική μας κλάση. Σημειώστε ότι η πραγματική κλάση C μπορεί να είναι μεγαλύτερη από την S , αλλά δεν είναι ποτέ μικρότερη. Η *πιο γενική υπόθεση*, G , είναι το μεγαλύτερο ορθογώνιο που περιλαμβάνει όλα τα θετικά δείγματα και κανένα από τα αρνητικά παραδείγματα. Οποιαδήποτε $h \in H$ μεταξύ S και G είναι μια έγκυρη υπόθεση χωρίς λάθος, η οποία είναι *συνεπής* με το σύνολο εκπαίδευσης, και συνθέτει το *version space*.

Δεδομένου ενός συνόλου εκπαίδευσης X , μπορούμε να βρούμε τις S ή G , ή ίσως ένα μέσο όρο τους, ή οποιαδήποτε h από το *version space* και να τη χρησιμοποιήσουμε ως την υπόθεση μας, h . Δοσμένου ενός άλλου συνόλου εκπαίδευσης, οι παράμετροι και επομένως η νέα υπόθεση, h , μπορεί να είναι διαφορετική.

Μπορούμε επίσης να πούμε ότι κάθε παράδειγμα που καλύπτεται από την S είναι θετικό, ενώ κάθε παράδειγμα που δεν καλύπτεται από την G είναι αρνητικό, και οποιαδήποτε άλλο παράδειγμα (μεταξύ S και G) είναι μια περίπτωση *αμφιβολίας*, το οποίο δεν μπορούμε να το ταξινομήσουμε με βεβαιότητα, λόγω της έλλειψης των δεδομένων. Σε μια τέτοια περίπτωση, το σύστημα *απορρίπτει* το παράδειγμα και μεταθέτει την απόφαση σε έναν εμπειρογνώμονα.

Εδώ, υποθέτουμε ότι η H περιλαμβάνει την C . Δηλαδή, υπάρχει μία $h \in H$, έτσι ώστε το $E(h|X)$ να είναι μηδέν. Λαμβάνοντας υπόψη μια κλάση υπόθεσης H , είναι πιθανό να μην μπορούμε να μάθουμε την C . Δηλαδή, δεν υπάρχει καμία $h \in H$ για την οποία το σφάλμα να είναι μηδέν. Έτσι, σε κάθε εφαρμογή, πρέπει να βεβαιωθούμε ότι η H είναι αρκετά ευέλικτη, ή έχει αρκετή «χωρητικότητα», για να μάθει την C .

1.2.2. Πολλαπλές Τάξεις

Στο παράδειγμά μας, της μάθησης ενός οικογενειακού αυτοκίνητου, έχουμε θετικά παραδείγματα που ανήκουν στην κλάση οικογενειακό αυτοκίνητο και αρνητικά παραδείγματα που ανήκουν σε όλα τα άλλα αυτοκίνητα. Αυτό είναι ένα πρόβλημα *δύο-κλάσεων*. Στην γενική περίπτωση, έχουμε K τάξεις που συμβολίζονται ως $C_i, i = 1, \dots, K$, και ένα εισερχόμενο παράδειγμα ανήκει σε μία και ακριβώς μία από αυτές. Το σύνολο εκπαίδευσης είναι τώρα της μορφής

$$X = (x^t, r^t)_{t=1}^N \quad (1.7)$$

όπου το r έχει K διαστάσεις και

$$r_i^t = \begin{cases} 1 & \text{αν } x^t \in C_i \\ 0 & \text{αν } x^t \in C_j, j \neq i \end{cases} \quad (1.8)$$

Ένα παράδειγμα δίνεται μέσα από τρεις τάξεις: οικογενειακό αυτοκίνητο, σπορ αυτοκίνητο, και το πολυτελές αυτοκίνητο.

Στην ταξινόμηση μέσω μηχανικής μάθησης, θα θέλαμε να μάθουμε το όριο που χωρίζει τα παραδείγματα μιας κλάσης από τα παραδείγματα όλων των άλλων κλάσεων. Έτσι αναγάγουμε ένα πρόβλημα ταξινόμησης K -κλάσεων ως ένα σύνολο από K δύο-κλάσεων προβλήματα. Τα παραδείγματα εκπαίδευσης που ανήκουν στην κλάση C_i είναι τα θετικά παραδείγματα της υπόθεσης h_i και τα παραδείγματα από όλες τις άλλες τάξεις είναι τα αρνητικά παραδείγματα της h_i . Έτσι, σε ένα πρόβλημα K -κλάσεων, έχουμε K υποθέσεις για να μάθουμε έτσι ώστε

$$h_i(x^t) = \begin{cases} 1 & \text{αν } x^t \in C_i \\ 0 & \text{αν } x^t \in C_j, j \neq i \end{cases} \quad (1.9)$$

Για ένα δεδομένο x , στην ιδανική περίπτωση, μόνο μία υπόθεση $h_i(x)$ είναι 1 και μπορούμε να επιλέξουμε μια κλάση. Αλλά όταν δεν υπάρχει καμία, ή υπάρχουν δύο ή περισσότερες υποθέσεις $h_i(x)$ που είναι 1, δεν μπορούμε να επιλέξουμε μια κλάση. Αυτή είναι η περίπτωση της *αμφιβολίας* και ο ταξινομητής απορρίπτει αυτά τα παραδείγματα.

Στο παράδειγμά μας, της μάθησης ενός οικογενειακού αυτοκινήτου, χρησιμοποιήσαμε μόνο μία υπόθεση και μοντελοποιήσαμε μόνο τα θετικά παραδείγματα. Οποιοδήποτε αρνητικό παράδειγμα είναι εκτός του μοντέλου δεν είναι ένα οικογενειακό αυτοκίνητο. Εναλλακτικά, μερικές φορές μπορεί να προτιμούμε να κατασκευάσουμε δύο υποθέσεις, μία για τα θετικά και μία για τα αρνητικά παραδείγματα. Αυτό προϋποθέτει μια δομή και για τα αρνητικά παραδείγματα που μπορούν να καλυφθούν από άλλη υπόθεση. Ο διαχωρισμός των οικογενειακών αυτοκινήτων από τα σπορ αυτοκίνητα είναι ένα τέτοιο πρόβλημα, στο οποίο η κάθε κλάση έχει μια δική της δομή. Το πλεονέκτημα είναι ότι αν η είσοδος είναι ένα πολυτελές αυτοκίνητο, μπορούμε να έχουμε δύο υποθέσεις που θα αποφασίσουν αρνητικά και θα απορρίψουν το παράδειγμα.

1.2.3. Παλινδρόμηση

Στην ταξινόμηση, δεδομένης μίας εισόδου, η έξοδος που παράγεται είναι Boolean, πρόκειται δηλαδή για μια θετική ή αρνητική απάντηση. Όταν η έξοδος είναι μια αριθμητική τιμή, αυτό που θα θέλαμε να μάθουμε δεν είναι η κλάση, $C(x) \in \{0, 1\}$, αλλά μια συνεχής συνάρτηση. Στη μηχανική μάθηση, η συνάρτηση δεν είναι γνωστή αλλά έχουμε ένα σύνολο εκπαίδευσης από παραδείγματα που προέρχονται από αυτό.

$$X = (x^t, r^t)_{t=1}^N \quad (1.10)$$

Όπου $r^t \in \mathfrak{R}$. Αν δεν υπάρχει θόρυβος, πρόκειται για *παρεμβολή*. Θα θέλαμε να βρούμε την συνάρτηση $f(x)$ που περνά μέσα από αυτά τα σημεία έτσι ώστε να έχουμε

$$r^t = f(x^t) \quad (1.11)$$

Στην πολυωνυμική παρεμβολή, δεδομένων N σημείων, βρίσκουμε το $(N - 1)$ βαθμού πολυώνυμο που μπορούμε να χρησιμοποιήσουμε για να προβλέψουμε το αποτέλεσμα για κάθε x . Αυτό ονομάζεται *παρέκταση*, αν το x είναι έξω από το φάσμα των x^t στο σύνολο εκπαίδευσης. Στην πρόβλεψη χρονοσειρών, για παράδειγμα, έχουμε δεδομένα

μέχρι και το παρόν και θέλουμε να προβλέψουμε την τιμή στο μέλλον. Στην *παλινδρόμηση*, υπάρχει θόρυβος ο οποίος προστίθεται στην έξοδο της άγνωστης συνάρτησης

$$r^t = f(x^t) + \epsilon \quad (1.12)$$

όπου $f(x) \in \mathfrak{R}$ είναι η άγνωστη συνάρτηση και ϵ είναι ένας τυχαίος θόρυβος. Η εξήγηση για το θόρυβο είναι ότι υπάρχουν επιπλέον κρυμμένες μεταβλητές που δεν μπορούμε να παρατηρήσουμε

$$r^t = f^*(x^t, z^t) \quad (1.13)$$

όπου z^t δηλώνει εκείνες τις κρυφές μεταβλητές. Θα θέλαμε να προσεγγίσουμε την έξοδο με το μοντέλο μας $g(x)$. Το εμπειρικό σφάλμα του συνόλου εκπαίδευσης X είναι

$$E(g|X) = \frac{1}{N} \sum_{t=1}^N [r^t - g(x^t)]^2 \quad (1.14)$$

Επειδή τα r και $g(x)$ είναι αριθμητικές ποσότητες, για παράδειγμα, $\in \mathfrak{R}$, υπάρχει μια διάταξη που ορίζεται πάνω στις τιμές τους και μπορούμε να ορίσουμε μια απόσταση μεταξύ των τιμών, όπως το τετράγωνο της διαφοράς τους, η οποία μας δίνει περισσότερες πληροφορίες από ότι η ισότητα ή η μη ισότητα, όπως χρησιμοποιείται στην ταξινόμηση. Το τετράγωνο της διαφοράς είναι μία συνάρτηση του σφάλματος που μπορεί να χρησιμοποιηθεί, ένα άλλο είναι η απόλυτη τιμή της διαφοράς. Θα δούμε και άλλα παραδείγματα στη συνέχεια του παραρτήματος.

Στόχος μας είναι να βρούμε την $g(\cdot)$ που ελαχιστοποιεί το εμπειρικό σφάλμα. Και πάλι η προσέγγισή μας είναι η ίδια, υποθέτουμε μια κλάση υπόθεσης για την $g(\cdot)$, με ένα μικρό σύνολο παραμέτρων. Αν υποθέσουμε ότι $g(x)$ είναι γραμμική, έχουμε

$$g(x) = w_1 x_1 + \dots + w_d x_d + w_0 = \sum_{j=1}^d w_j x_j + w_0 \quad (1.15)$$

Ας πάμε πίσω στο παράδειγμά μας στην παράγραφο 1.2.1 όπου εκτιμήσαμε την τιμή ενός αυτοκινήτου. Εκεί χρησιμοποιήσαμε ένα γραμμικό μοντέλο απλής εισόδου

$$g(x) = w_1 x + w_0 \quad (1.16)$$

όπου w_1 και w_0 είναι οι παράμετροι που μαθαίνονται από τα δεδομένα. Οι τιμές των w_1 και w_0 θα πρέπει να ελαχιστοποιηθούν

$$E(w_1, w_0|X) = \sum_{t=1}^N [r^t - (w_1 x + w_0)]^2 \quad (1.17)$$

Το ελάχιστο σημείο του μπορεί να υπολογιστεί λαμβάνοντας τις μερικές παραγώγους του E σε σχέση με τα w_1 και w_0 , θέτοντας τες ίσες με 0, και επιλύοντας για τους δύο αγνώστους:

$$\begin{aligned} w_1 &= \frac{\sum_t x^t r^t - \bar{x} \bar{r} N}{\sum_t (x^t)^2 - N \bar{x}^2} \\ w_0 &= \bar{r} - w_1 \bar{x} \end{aligned} \quad (1.18)$$

Όπου $\bar{x} = \sum_t x^t / N$, $\bar{r} = \sum_t r^t / N$, και η λύση του είναι μία ευθεία γραμμή.

Το γραμμικό μοντέλο είναι πολύ απλό, με αποτέλεσμα να είναι περιορισμένο και να πραγματοποιεί ένα μεγάλο σφάλμα προσέγγισης. Σε μια τέτοια περίπτωση, η έξοδος μπορεί να ληφθεί ως συνάρτηση ανώτερης τάξης των δεδομένων εισόδου, για παράδειγμα, την τετραγωνική

$$g(x) = w_2 x^2 + w_1 x + w_0 \quad (I.19)$$

όπου παρόμοια έχουμε μια αναλυτική λύση για τις παραμέτρους. Όταν η τάξη του πολυωνύμου αυξάνεται, το σφάλμα στα δεδομένα εκπαίδευσης μειώνεται. Αλλά ένα υψηλής τάξης πολυώνυμο ακολουθεί στενά τα μεμονωμένα παραδείγματα αντί να συλλάβει τη γενική τάση. Συνεπώς πρέπει να είμαστε προσεκτικοί όταν κάνουμε εξομάλυνση των βραχυχρόνιων διακυμάνσεων της πολυπλοκότητας του μοντέλου σε σχέση με την πολυπλοκότητα της συνάρτησης που κείται κάτω από τα δεδομένα.

I.2.4. Αλγόριθμοι Ταξινόμησης

I.2.4.1 Μη παραμετρικοί Ταξινομητές: *k*-Nearest Neighbor

Στη *μη παραμετρική εκτίμηση*, υποθέτουμε ότι *παρεμφερείς εισροές έχουν παρόμοια αποτελέσματα*. Ως εκ τούτου, ο αλγόριθμος μας αποτελείται από την εύρεση παρόμοιων προηγούμενων παραδειγμάτων από το σύνολο εκπαίδευσης, χρησιμοποιώντας ένα κατάλληλο μέτρο απόστασης, με σκοπό να κάνουμε παρεμβολή και να βρούμε τη σωστή έξοδο. Διαφορετικές μη παραμετρικές μέθοδοι διαφέρουν ως προς τον τρόπο που ορίζουν την ομοιότητα ή κάνουν την παρεμβολή από παρόμοια παραδείγματα εκπαίδευσης. Σε ένα παραμετρικό μοντέλο, το σύνολο των παραδειγμάτων εκπαίδευσης επηρεάζει την τελική συνολική εκτίμηση, ενώ στην μη παραμετρική περίπτωση, δεν υπάρχει ένα ενιαίο συνολικό μοντέλο, τα τοπικά μοντέλα εκτιμώνται όταν είναι απαραίτητο, επηρεάζονται μόνο από τα παραδείγματα εκπαίδευσης που βρίσκονται κοντά. Στην ορολογία της μηχανικής μάθησης, οι μη παραμετρικές μέθοδοι ονομάζονται επίσης *βασισμένοι σε παραδείγματα* ή *βασισμένοι σε μνήμη* αλγόριθμοι εκπαίδευσης, διότι αυτό που κάνουν είναι να αποθηκεύουν τα παραδείγματα εκπαίδευσης σε έναν πίνακα αναζήτησης και να παρεμβάλουν από αυτά.

Ως συνήθως στην εκτίμηση της πυκνότητας, υποθέτουμε ότι το δείγμα $X = \{x^t\}_{t=1}^N$ προέρχεται ανεξάρτητα από κάποια άγνωστη πιθανότητα πυκνότητας $p(\cdot)$. $\hat{p}(\cdot)$ είναι η εκτίμηση του $p(\cdot)$. Στην περίπτωση της ταξινόμησης, χρησιμοποιούμε τη μη παραμετρική προσέγγιση για την εκτίμηση των εξαρτώμενων από την κλάση πυκνοτήτων, $p(x|C_i)$. Ο *k*-η εκτιμητής της εξαρτώμενης από την κλάση πυκνότητας δίνεται ως

$$\hat{p}(x|C_i) = \frac{k_i}{N_i V^k(x)} \quad (I.20)$$

Όπου k_i είναι ο αριθμός των γειτόνων από τον *k* πλησιέστερο που ανήκουν στη κλάση C_i και $V^k(x)$ είναι ο όγκος της *d*-διαστάσεως υπερσφαίρας με κέντρο το x , και με ακτίνα $r = \|x - x_{(k)}\|$ όπου $x_{(k)}$ είναι το *k*-οστό πλησιέστερο παράδειγμα από το x (ανάμεσα σε όλους τους γείτονες από όλες τις τάξεις του x): $V^k = r^d c_d$ με c_d να είναι ο όγκος της σφαίρας στις *d* διαστάσεις, για παράδειγμα, $c_1 = 2$, $c_2 = \pi$, $c_3 = 4\pi/3$, και ούτω καθεξής. Τότε

$$\hat{P}(C_i|x) = \frac{\hat{p}(x|C_i)\hat{P}(C_i)}{\hat{p}(x)} = \frac{k_i}{k} \quad (I.21)$$

Ο k -νη ταξινομητής αναθέτει την είσοδο στην κλάση που έχει τα περισσότερα παραδείγματα μεταξύ των k γειτόνων της εισόδου. Όλοι οι γείτονες έχουν ίση ψήφο, και η κλάση που έχει το μέγιστο αριθμό των ψηφοφόρων μεταξύ των k γειτόνων επιλέγεται. Μια ειδική περίπτωση του k -νη είναι ο ταξινομητής του κοντινότερου γείτονα, όπου $k = 1$ και η είσοδος ανατίθεται στην κλάση του πλησιέστερου προτύπου. Αυτό χωρίζει το χώρο με τη μορφή ενός *Voronoi tessellation*.

1.2.4.2. Δέντρα απόφασης: C4.5

Ένα δέντρο απόφασης είναι ένα ιεραρχικό μοντέλο εποπτευόμενης μάθησης σύμφωνα με το οποίο η τοπική περιοχή προσδιορίζεται σαν μια ακολουθία αναδρομικών διαχωρισμών σε ένα μικρότερο αριθμό βημάτων. Ένα δέντρο απόφασης αποτελείται από εσωτερικούς κόμβους απόφασης και τερματικά φύλλα. Κάθε κόμβος απόφασης m υλοποιεί μία συνάρτηση ελέγχου $f_m(x)$ με διακριτά αποτελέσματα η οποία επισημαίνει την πληροφορία των διακλαδώσεων του δέντρου. Με δεδομένη εισροή, σε κάθε κόμβο, ένας έλεγχος εφαρμόζεται και μία από τις διακλαδώσεις λαμβάνεται ανάλογα με το αποτέλεσμα. Αυτή η διαδικασία ξεκινά από τη ρίζα και επαναλαμβάνεται αναδρομικά μέχρι να φθάσουμε σε έναν κόμβο φύλλου, σε αυτό το σημείο η τιμή που αναγράφεται στο φύλλο αποτελεί την έξοδο.

Σε ένα μονομεταβλητές δέντρο, σε κάθε εσωτερικό κόμβο, ο έλεγχος χρησιμοποιεί μόνο μία από τις διαστάσεις εισόδου. Εάν η χρησιμοποιούμενη διάσταση εισόδου, x_j , είναι διακριτή, λαμβάνοντας μία από τις n πιθανές τιμές, ο κόμβος απόφασης ελέγχει την τιμή της x_j και λαμβάνει την αντίστοιχη διακλάδωση υλοποιώντας έναν n -διαδρομής διαχωρισμό. Για παράδειγμα, αν ένα χαρακτηριστικό είναι το χρώμα με πιθανές τιμές {κόκκινο, μπλε, πράσινο}, τότε ένας κόμβος σε αυτό το χαρακτηριστικό έχει τρεις διακλαδώσεις, που η κάθε μία αντιστοιχεί σε μία από τις τρεις πιθανές τιμές του χαρακτηριστικού.

Ένας κόμβος απόφασης έχει διακριτές διακλαδώσεις και οι αριθμητικές εισοδοί θα πρέπει να διακριτοποιούνται. Αν x_j είναι αριθμός (διατεταγμένος), ο έλεγχος είναι η σύγκριση

$$f_m(x): x_j \geq w_{m0} \quad (1.22)$$

όπου w_{m0} είναι μία κατάλληλα επιλεγμένη τιμή κατωφλίου. Ο κόμβος απόφασης χωρίζει το χώρο εισόδου σε δύο: $L_m = \{x | x_j \geq w_{m0}\}$ και $R_m = \{x | x_j < w_{m0}\}$, αυτό ονομάζεται *δυναμικός διαχωρισμός*. Διαδοχικοί κόμβοι απόφασης σε μια διαδρομή από τη ρίζα σε ένα φύλλο διαιρούν περαιτέρω αυτούς σε δύο χαρακτηριστικά και δημιουργούν διαχωρισμούς ορθογώνιους μεταξύ τους. Οι κόμβοι φύλλα καθορίζουν υπερορθογώνια στο χώρο εισόδου.

Στην περίπτωση ενός δέντρου απόφασης για την ταξινόμηση, δηλαδή, ένα δέντρο ταξινόμησης, η βελτιστοποίηση ενός διαχωρισμού ποσοτικοποιείται από ένα μέτρο ακαθαρσίας. Ο διαχωρισμός είναι καθαρός, αν μετά το διαχωρισμό, για όλες τις διακλαδώσεις, όλα τα παραδείγματα, επιλέγοντας μία διακλάδωση ανήκουν στην ίδια κλάση. Ας πούμε για τον κόμβο m , N_m είναι ο αριθμός των παραδειγμάτων εκπαίδευσης που φθάνουν στον κόμβο m . Για τον κόμβο ρίζα, είναι N . N_m^i από N_m ανήκουν στην κλάση C_i , με $\sum_i N_m^i = N_m$. Δεδομένου ότι ένα παράδειγμα φτάνει τον κόμβο m , η εκτίμηση για την πιθανότητα της κλάσης C_i είναι

$$\hat{P}(C_i|\mathbf{x}, m) \equiv p_m^i = \frac{N_m^i}{N_m} \quad (1.23)$$

Ο κόμβος m είναι καθαρός, αν p_m^i για κάθε i είναι είτε 0 ή 1. Είναι 0 όταν κανένα από τα παραδείγματα που φθάνουν τον κόμβο m δεν είναι της κλάσης C_i , και είναι 1 αν όλα αυτά τα παραδείγματα είναι της κλάσης C_i . Αν ο διαχωρισμός είναι καθαρός, δεν χρειάζεται να διαχωρίσουμε περαιτέρω και μπορούμε να προσθέσουμε έναν κόμβο φύλλου επισημαίνοντας τον με την κλάση για την οποία p_m^i είναι 1. Μια πιθανή συνάρτηση για να μετρήσουμε την ακαθαρσία είναι η εντροπία.

$$I_m = - \sum_{i=1}^K p_m^i \log_2 p_m^i \quad (1.24)$$

όπου $0 \log 0 \equiv 0$. Η εντροπία στη θεωρία της πληροφορίας καθορίζει τον ελάχιστο αριθμό των bits που απαιτούνται για την κωδικοποίηση της ακρίβειας ταξινόμησης ενός παραδείγματος.

Εάν ο κόμβος m δεν είναι καθαρός, τότε τα παραδείγματα θα πρέπει να διαχωριστούν για να μειωθεί η ακαθαρσία και υπάρχουν πολλά πιθανά χαρακτηριστικά στα οποία μπορούμε να διαχωρίσουμε. Για ένα αριθμητικό χαρακτηριστικό πολλαπλές θέσεις διαχωρισμού είναι δυνατές. Μεταξύ όλων, ψάχνουμε για το διαχωρισμό που ελαχιστοποιεί την ακαθαρσία μετά τη διάσπαση, επειδή θέλουμε να δημιουργήσουμε το μικρότερο δέντρο. Εάν τα υποσύνολα μετά το διαχωρισμό είναι πιο καθαρά θα χρειαστούν λιγότεροι διαχωρισμοί (αν υπάρχουν) στη συνέχεια. Φυσικά αυτό είναι τοπικά βέλτιστο, και δεν έχουμε καμία εγγύηση για την εύρεση του μικρότερου δένδρου απόφασης. Ας πούμε ότι στον κόμβο m , N_{mj} από N_m επιλέγουν την j διακλάδωση, αυτά είναι \mathbf{x}^t παραδείγματα για τα οποία η συνάρτηση ελέγχου $f_m(\mathbf{x}^t)$ επιστρέφει ως αποτέλεσμα τη διακλάδωση j . Για ένα διακριτό χαρακτηριστικό με n τιμές υπάρχουν n δυνατά αποτελέσματα και για ένα αριθμητικό χαρακτηριστικό, υπάρχουν δύο δυνατά αποτελέσματα ($n = 2$), και στις δύο περιπτώσεις ικανοποιείται η συνθήκη $\sum_{j=1}^n N_{mj} = N_m$. N_{mj}^i από N_{mj} ανήκουν στην κλάση C_i : $\sum_{i=1}^K N_{mj}^i = N_{mj}$. Ομοίως, $\sum_{j=1}^n N_{mj}^i = N_m^i$.

Έτσι, δεδομένου ότι στον κόμβο m , ο έλεγχος επιστρέφει αποτέλεσμα j , η εκτίμηση για την πιθανότητα της κλάσης C_i είναι

$$\hat{P}(C_i|\mathbf{x}, m, j) \equiv p_{mj}^i = \frac{N_{mj}^i}{N_{mj}} \quad (1.25)$$

Και η συνολική ακαθαρσία μετά το διαχωρισμό δίνεται ως

$$\hat{I}_m = - \sum_{j=1}^n \frac{N_{mj}}{N_m} \sum_{i=1}^K p_{mj}^i \log_2 p_{mj}^i \quad (1.26)$$

Έτσι, για όλα τα χαρακτηριστικά διακριτά και αριθμητικά, και για ένα αριθμητικό χαρακτηριστικό για όλες τις θέσεις διαχωρισμού, υπολογίζουμε την ακαθαρσία και επιλέγουμε αυτή που έχει την ελάχιστη εντροπία, για παράδειγμα, όπως μετράται από την εξίσωση 1.26. Στη συνέχεια, η κατασκευή του δένδρου γίνεται αναδρομικά και παράλληλα για όλες τις διακλαδώσεις που δεν είναι καθαρές, έως ότου όλα να είναι καθαρά. Αυτή είναι η αρχή των αλγορίθμων των *Δέντρων ταξινόμησης και Παλινδρόμησης* (CART), του αλγόριθμου ID3, και της επέκτασής του, C4.5.

1.2.4.3. Σύνθετοι Ταξινομητές: Voting

Κάθε μαθησιακός αλγόριθμος υπαγορεύει ένα συγκεκριμένο μοντέλο που έχει μια σειρά υποθέσεων. Αυτή η επαγωγική κλίση οδηγεί σε σφάλμα, αν οι υποθέσεις δεν ισχύουν για τα συγκεκριμένα δεδομένα. Η μάθηση είναι ένα ιδιαίτερο πρόβλημα όπου με πεπερασμένα στοιχεία κάθε αλγόριθμος συγκλίνει σε διαφορετική λύση και είναι πιθανό να αποτύχει κάτω από διαφορετικές συνθήκες. Η απόδοση ενός μαθησιακού αλγόριθμου μπορεί να τελειοποιηθεί έτσι ώστε να επιτευχθεί η υψηλότερη δυνατή ακρίβεια σε ένα σύνολο επικύρωσης, αλλά αυτή η τελειοποίηση είναι μια πολύπλοκη διαδικασία και εξακολουθούν να υπάρχουν περιπτώσεις στις οποίες ακόμη και ο καλύτερος μαθησιακός αλγόριθμος δεν είναι αρκετά ακριβής. Η κύρια ιδέα σε αυτήν την περίπτωση είναι ότι μπορεί να υπάρχει ένας άλλος μαθησιακός αλγόριθμος που μπορεί να είναι ακριβής σε αυτό το σύνολο επικύρωσης. Αν συνδυαστούν κατάλληλα πολλαπλοί μαθησιακοί αλγόριθμοι τότε, η ακρίβεια μπορεί να βελτιωθεί. Πρόσφατα, όπου η ταχύτητα υπολογισμού έχει βελτιωθεί και η χρηματική αξία της μνήμης έχει γίνει φθηνή, τα συστήματα αυτά που αποτελούνται από πολλαπλούς μαθησιακούς αλγόριθμους έχουν γίνει ιδιαίτερα δημοφιλή.

Ας υποθέσουμε ότι έχουμε L βασικούς μαθησιακούς αλγόριθμους. Δηλώνουμε με $d_j(x)$ την πρόβλεψη του M_j βασικού μαθησιακού αλγόριθμου δεδομένης της αυθαίρετης διάστασης εισόδου x . Στην περίπτωση πολλαπλών αναπαραστάσεων κάθε M_j χρησιμοποιεί διαφορετική x_j αναπαράσταση εισόδου. Η τελική πρόβλεψη υπολογίζεται με βάση τις προβλέψεις των βασικών μαθησιακών αλγόριθμων:

$$y = f(d_1, d_2, \dots, d_L | \Phi) \quad (1.27)$$

Όπου $y = f(\cdot)$ είναι η συνδυαστική συνάρτηση και όπου Φ δηλώνονται οι παράμετροι της. Όταν υπάρχουν K έξοδοι, κάθε μαθησιακός αλγόριθμος έχει K εξόδους, $d_{ji}(x)$, $i = 1, \dots, K$, $j = 1, \dots, L$, και ο συνδυασμός τους, θα παράξει επίσης K τιμές y_i , $i = 1, \dots, K$ και στη συνέχεια, για παράδειγμα σε μια ταξινόμηση επιλέγουμε την κλάση με τη μέγιστη τιμή y_i .

Ο απλούστερος τρόπος να συνδυαστούν πολλαπλοί ταξινομητές είναι με την *ψηφοφορία*, η οποία αντιστοιχεί σε έναν γραμμικό συνδυασμό των μαθησιακών αλγορίθμων. Αν ορίσουμε με w_j το βάρος του μαθησιακού αλγόριθμου j , τότε το τελικό αποτέλεσμα υπολογίζεται ως

$$y = \sum_{j=1}^L w_j d_j \quad (1.28)$$

Ικανοποιώντας

$$w_j \geq 0, \forall j \text{ and } \sum_{j=1}^L w_j = 1$$

Εδώ, η συνάρτηση $f(\cdot)$ της εξίσωσης 1.27 αντιστοιχεί σε ένα σταθμισμένο άθροισμα όπου Φ είναι το σύνολο των βαρών, w_1, \dots, w_L .

Στην παλινδρόμηση, παίρνουμε το σταθμισμένο μέσο όρο των ατομικών προβλέψεων. Ο όρος *ψηφοφορία* προέρχεται από τη χρήση του στην ταξινόμηση

$$y_i = \sum_{j=1}^L w_j d_{ji} \quad (1.29)$$

Όπου d_{ji} είναι η ψήφος του μαθησιακού αλγόριθμου j για την κλάση C_i και w_j είναι το βάρος της ψήφου του. Στην απλούστερη περίπτωση, έχουμε *απλή ψηφοφορία*, στην οποία όλοι οι ψηφοφόροι έχουν την ίδια βαρύτητα, δηλαδή, $w_j = 1/L$. Στην ταξινόμηση αυτό ονομάζεται *ψηφοφορία πολλαπλότητας* όπου η κλάση που έχει το μέγιστο αριθμό ψήφων είναι και ο νικητής. Όταν υπάρχουν δύο τάξεις, έχουμε την *ψηφοφορία πλειοψηφίας*, όπου η κλάση νικητής παίρνει περισσότερο από το ήμισυ των ψήφων. Εάν οι ψηφοφόροι παρέχουν και πρόσθετες πληροφορίες για το πόσο θα ψηφίσουν για κάθε κλάση (για παράδειγμα, με την ύστερη πιθανότητα), τότε μετά από την κανονικοποίηση αυτές μπορούν να χρησιμοποιηθούν ως βάρη σε μία *σταθμισμένη ψηφοφορία*. Αντίστοιχα, αν d_{ji} είναι οι ύστερες πιθανότητες της κλάσης, $P(C_i|x, M_j)$, τότε μπορούμε να τις αθροίσουμε ($w_j = 1/L$) και να επιλέξουμε την κλάση με τη μεγαλύτερη δυνατή y_i .

Μια άλλη δυνατότητα είναι να αποκτηθεί πρόσβαση στην ακρίβεια των μαθησιακών αλγορίθμων (παλινδρομητών ή ταξινομητών), σε ένα ξεχωριστό σύνολο επικύρωσης και να χρησιμοποιηθεί αυτή η πληροφορία για να υπολογιστούν τα βάρη, έτσι ώστε να δοθεί περισσότερο βάρος στους πιο ακριβείς μαθησιακούς αλγορίθμους.

1.2.5. Αξιολόγηση και Σύγκριση Αλγόριθμων Ταξινόμησης

Οι ταξινομητές της μηχανικής μάθησης εξαρτώνται από το σύνολο εκπαίδευσης, και υπάρχει ανάγκη για στατιστικό έλεγχο, πρώτον για να αξιολογηθεί το αναμενόμενο σφάλμα ενός αλγορίθμου ταξινόμησης, και δεύτερον για να συγκριθεί το αναμενόμενο ποσοστό σφάλματος δύο αλγορίθμων ταξινόμησης για να βρεθεί ο βέλτιστος από αυτούς.

1.2.5.1. *K*-Φορές Διασταυρωμένη Επικύρωση (*K*-Fold Cross-Validation)

Στην *K*-φορές διασταυρωμένη επικύρωση, το σύνολο δεδομένων X χωρίζεται τυχαία σε K μέρη ίσου μεγέθους, X_i , $i = 1, \dots, K$. Για να δημιουργήσουμε κάθε ζευγάρι (δηλαδή, ένα σύνολο εκπαίδευσης και ένα σύνολο επικύρωσης), κρατάμε ένα από τα K μέρη έξω ως το σύνολο επικύρωσης, V , και όλα τα υπόλοιπα $K - 1$ μέρη σχηματίζουν το σύνολο εκπαίδευσης, T . Κάνοντας αυτήν την ενέργεια K φορές, κάθε φορά αφήνοντας έξω ένα άλλο από τα K μέρη, θα έχουμε K ζεύγη:

$$\begin{aligned} V_1 &= X_1 & T_1 &= X_2 \cup X_3 \cup \dots \cup X_K \\ V_2 &= X_2 & T_2 &= X_1 \cup X_3 \cup \dots \cup X_K \\ &\vdots & &\vdots \\ V_K &= X_K & T_K &= X_1 \cup X_2 \cup \dots \cup X_{K-1} \end{aligned}$$

Υπάρχουν δύο προβλήματα με αυτό: Πρώτον, για να διατηρηθεί το σύνολο εκπαίδευσης μεγάλο, επιτρέπουμε στα σύνολα επικύρωσης να είναι μικρά. Δεύτερον, τα σύνολα εκπαίδευσης επικαλύπτονται σε μεγάλο βαθμό, δηλαδή, κάθε δυο σύνολα εκπαίδευσης μοιράζονται $K - 2$ μέρη.

Το K είναι συνήθως 10 ή 30. Καθώς το K αυξάνεται, το ποσοστό των παραδειγμάτων εκπαίδευσης αυξάνεται και έχουμε ως αποτέλεσμα πιο ισχυρούς εκτιμητές, ωστόσο το σύνολο επικύρωσης γίνεται μικρότερο. Επιπλέον, το κόστος εκπαίδευσης του ταξινομητή για K φορές αυξάνεται καθώς το K αυξάνει. Όσο το N αυξάνεται, το K μπορεί να είναι μικρότερο, αν το N είναι μικρό, τότε το K θα πρέπει να είναι μεγάλο για να έχουμε αρκετά μεγάλα σύνολα εκπαίδευσης. Μια ακραία περίπτωση της *K*-φορές διασταυρωμένης επικύρωσης είναι το *leave-one-out*, όπου δοσμένου ενός σύνολο δεδομένων με N παραδείγματα, μόνο ένα παράδειγμα μένει έξω ως το σύνολο επικύρωσης και η εκπαίδευση χρησιμοποιεί τα $N - 1$ παραδείγματα. Εμείς στη συνέχεια

παίρνουμε N ξεχωριστά ζεύγη αφήνοντας έξω ένα διαφορετικό παράδειγμα σε κάθε επανάληψη. Αυτή η μέθοδος συνήθως χρησιμοποιείται σε εφαρμογές όπως η ιατρική διάγνωση, όπου τα δεδομένα είναι δύσκολο να βρεθούν.

1.2.5.2. McNemar's Test

Δοθέντος ενός συνόλου εκπαίδευσης και ενός συνόλου επικύρωσης, χρησιμοποιούμε δύο αλγόριθμους για να εκπαιδύσουμε δύο ταξινομητές πάνω στο σύνολο εκπαίδευσης και τους ελέγχουμε στο σύνολο επικύρωσης με αποτέλεσμα να υπολογίσουμε τα λάθη τους. Ένας πίνακας *contingency table*, όπως αυτός που παρουσιάζεται εδώ, είναι μια σειρά φυσικών αριθμών σε μορφή πίνακα που αντιπροσωπεύουν μετρήσεις, ή συχνότητες:

e_{00} : Πλήθος παραδειγμάτων που έχουν ταξινομηθεί εσφαλμένα και από τους δύο	e_{01} : Πλήθος παραδειγμάτων που έχουν ταξινομηθεί εσφαλμένα από τον 1 αλλά όχι και από τον 2
e_{10} : Πλήθος παραδειγμάτων που έχουν ταξινομηθεί εσφαλμένα από τον 2 αλλά όχι και από τον 1	e_{11} : Πλήθος παραδειγμάτων που έχουν ταξινομηθεί σωστά και από τους δύο

Σύμφωνα με την μηδενική υπόθεση, η οποία είναι, ότι οι αλγόριθμοι ταξινόμησης έχουν το ίδιο ποσοστό σφάλματος, αναμένουμε ότι $e_{01} = e_{10}$ και αυτό να είναι ίσο με $(e_{01} + e_{10})/2$. Έχουμε το chi-square στατιστικό στοιχείο με ένα βαθμό ελευθερίας

$$\frac{(|e_{01} - e_{10}| - 1)^2}{e_{01} + e_{10}} \sim \chi_1^2 \quad (1.30)$$

Το *McNemar's test* αποδέχεται την υπόθεση ότι οι δύο αλγόριθμοι ταξινόμησης έχουν το ίδιο ποσοστό σφάλματος στο α επίπεδο σημαντικότητας, αν αυτή η τιμή είναι μικρότερη ή ίση με $\chi_{\alpha,1}^2$. $\chi_{0.05,1}^2 = 3.84$.

1.2.6. Αλγόριθμοι Παλινδρόμησης

1.2.6.1. Μη παραμετρικοί Παλινδρομητές: Kernel Smoother

Στην παλινδρόμηση, λαμβάνοντας υπόψη το σύνολο εκπαίδευσης $X = \{x^t, r^t\}$ όπου $r^t \in \mathfrak{R}$, υποθέτουμε

$$r^t = g(x^t) + \epsilon \quad (1.31)$$

Στην παραμετρική παλινδρόμηση, χρησιμοποιούμε ένα πολυώνυμο μιας συγκεκριμένης κλάσης και υπολογίζουμε τους συντελεστές που ελαχιστοποιούν το άθροισμα των τετραγωνικών σφαλμάτων στο σύνολο εκπαίδευσης. Η μη παραμετρική παλινδρόμηση χρησιμοποιείται όταν δεν υπάρχει ένα τέτοιο πολυώνυμο, το μόνο που μπορούμε να υποθέσουμε είναι ότι κοντινά x έχουν κοντινές $g(x)$ τιμές. Δεδομένων των x , η προσέγγισή μας είναι να βρούμε τη γειτονιά των x και να υπολογίσουμε το μέσο όρο των r τιμών στη γειτονιά τους έτσι ώστε να υπολογίσουμε την $\hat{g}(x)$. Ο μη παραμετρικός εκτιμητής παλινδρόμησης ονομάζεται επίσης και *smoother* και η εκτίμηση του ονομάζεται *smooth*. Υπάρχουν διάφορες μέθοδοι για τον προσδιορισμό της γειτονιάς

και τον υπολογισμό του μέσου όρου στη γειτονιά. Θα συζητήσουμε τις μεθόδους για την μονομεταβλητή x , όπου μπορούν να γενικευτούν και στην πολυμεταβλητή περίπτωση, κατά απλό τρόπο κάνοντας χρήση πολυμεταβλητών πυρήνων.

Όπως και στον εκτιμητή του πυρήνα, μπορούμε να χρησιμοποιήσουμε έναν πυρήνα που να δίνει μικρότερη βαρύτητα στα απομακρυσμένα σημεία, και παίρνουμε τον *kernel smoother*

$$\hat{g}(x) = \frac{\sum_t K\left(\frac{x-x^t}{h}\right)r^t}{\sum_t K\left(\frac{x-x^t}{h}\right)} \quad (1.32)$$

Συνήθως, ένας Gaussian πυρήνας $K(\cdot)$ χρησιμοποιείται. Αντί για τον καθορισμό του h , μπορούμε να καθορίσουμε το k , που είναι ο αριθμός των γειτόνων, προσαρμόζοντας την εκτίμηση στην πυκνότητα γύρω από το x , με αποτέλεσμα να έχουμε τον *k-nn smoother*.

1.3. Μη εποπτευόμενη μάθηση

1.3.1. Ομαδοποίηση

1.3.1.1. Offline k-Means

Ας υποθέσουμε ότι έχουμε μια εικόνα που είναι αποθηκευμένη με 24 bits / pixel όπου μπορούμε να έχουμε έως και 16 εκατομμύρια χρώματα. Ας υποθέσουμε ότι έχουμε μια έγχρωμη οθόνη με 8 bits / pixel που μπορεί να εμφανίσει μόνο 256 χρώματα. Θέλουμε να βρούμε τα καλύτερα 256 χρώματα από τα 16 εκατομμύρια χρώματα, έτσι ώστε η εικόνα, χρησιμοποιώντας μόνο τα 256 χρώματα στην παλέτα να μοιάζει όσο το δυνατόν πλησιέστερα στην αρχική εικόνα. Αυτό ονομάζεται *κβαντοποίηση χρώματος* όπου γίνεται απεικόνιση από υψηλή σε χαμηλή ανάλυση. Στην γενική περίπτωση, ο στόχος είναι να κάνουμε μια απεικόνιση από έναν συνεχή χώρο σε ένα διακριτό χώρο, αυτή η διαδικασία ονομάζεται *κβαντισμός διανύσματος*.

Φυσικά, μπορούμε πάντα να κβαντοποιήσουμε ομοιόμορφα, αλλά αυτό φθείρει τον χάρτη χρώματος καθώς γίνεται ανάθεση καταχωρήσεων σε χρώματα που δεν υπάρχουν στην εικόνα, ή δεν θα εκχωρηθούν επιπλέον καταχωρήσεις σε χρώματα που χρησιμοποιούνται συχνά στην εικόνα. Για παράδειγμα, αν η εικόνα είναι ένα θαλασσίνο τοπίο, περιμένουμε να δούμε πολλές αποχρώσεις του μπλε και ίσως όχι το κόκκινο. Έτσι, η κατανομή των εγγραφών στον χάρτη χρώματος θα πρέπει να αντικατοπτρίζει την αρχική πυκνότητα όσο το δυνατόν πλησιέστερα τοποθετώντας πολλές καταχωρήσεις σε υψηλής πυκνότητας περιοχές, απορρίπτοντας τις περιοχές όπου δεν υπάρχουν καθόλου δεδομένα.

Ας υποθέσουμε ότι έχουμε ένα δείγμα $X = \{x^t\}_{t=1}^N$. Έχουμε k διανύσματα αναφοράς, $m_j, j = 1, \dots, k$. Στο παράδειγμά μας της κβαντοποίησης χρώματος, x^t είναι οι τιμές των pixel της εικόνας σε 24 bits και m_j οι καταχωρίσεις στον χάρτη χρώματος, επίσης, σε 24 bits, με $k = 256$.

Υποθέτουμε προς το παρόν ότι γνωρίζουμε κατά κάποιο τρόπο τις m_j τιμές, θα συζητήσουμε πώς θα γίνει η διαδικασία μάθησης τους σύντομα. Στη συνέχεια, για την εμφάνιση της εικόνας, δεδομένων των pixel, x^t , την αναπαριστούμε με την πιο όμοια είσοδο, m_i στο χάρτη χρώματος, ικανοποιώντας

$$\|x^t - \mathbf{m}_i\| = \min_j \|x^t - \mathbf{m}_j\|$$

Δηλαδή, αντί της αρχικής τιμής των δεδομένων, χρησιμοποιούμε την πλησιέστερη τιμή που έχουμε στο αλφάβητο των διανυσμάτων αναφοράς. Τα \mathbf{m}_i καλούνται επίσης *διανύσματα βιβλίου κωδικών* ή *συνθηματικές λέξεις*, γιατί αυτή είναι μια διαδικασία κωδικοποίησης / αποκωδικοποίησης. Η μετάβαση από το x^t στο i είναι μια διαδικασία κωδικοποίησης των δεδομένων χρησιμοποιώντας το βιβλίο κωδικών των \mathbf{m}_i , $i = 1, \dots, k$ και στη πλευρά λήψης, αποκωδικοποιούμε τα \mathbf{m}_i από τα i . Ο κβαντισμός επιτρέπει επίσης και τη *συμπύεση*. Για παράδειγμα, αντί να χρησιμοποιήσουμε 24 bit για την αποθήκευση (ή για την μεταφορά πάνω από μια γραμμή επικοινωνίας) για κάθε x^t , μπορούμε μόνο να αποθηκεύσουμε / μεταφέρουμε μόνο τον δείκτη του, i , στον χάρτη χρώματος χρησιμοποιώντας 8 bit για να δείξουμε ένα από τα 256 χρώματα, και θα πάρουμε ένα ρυθμό συμπίεσης σχεδόν ίσο με 3.

Ας δούμε πώς μπορούμε να υπολογίσουμε τα \mathbf{m}_i . Όταν τα x^t αναπαριστώνται από τα \mathbf{m}_i , υπάρχει ένα σφάλμα που είναι ανάλογο με την απόσταση, $\|x^t - \mathbf{m}_i\|$. Για να μοιάζει η νέα εικόνα με την αρχική εικόνα, θα πρέπει να έχουμε αυτές τις αποστάσεις όσο το δυνατόν μικρότερες για όλα τα pixels. Το συνολικό *σφάλμα ανοικοδόμησης* ορίζεται ως

$$E(\{\mathbf{m}_i\}_{i=1}^k | X) = \sum_t \sum_i b_i^t \|x^t - \mathbf{m}_i\|^2 \quad (1.33)$$

όπου

$$b_i^t = \begin{cases} 1, & \text{if } \|x^t - \mathbf{m}_i\| = \min_j \|x^t - \mathbf{m}_j\| \\ 0, & \text{otherwise} \end{cases} \quad (1.34)$$

Τα καλύτερα διανύσματα αναφοράς είναι εκείνα που ελαχιστοποιούν το συνολικό σφάλμα ανοικοδόμησης. Τα b_i^t εξαρτώνται επίσης από τα \mathbf{m}_i και δεν μπορούμε να λύσουμε αυτό το πρόβλημα βελτιστοποίησης αναλυτικά. Γι αυτό έχουμε μια επαναληπτική διαδικασία που ονομάζεται *k-means clustering*. Κατ αρχάς, ξεκινάμε με κάποια \mathbf{m}_i που αρχικοποιούνται τυχαία. Στη συνέχεια, σε κάθε επανάληψη, χρησιμοποιούμε στην αρχή την εξίσωση 1.34 και υπολογίστε τα b_i^t για όλα τα x^t , τα οποία είναι οι εκτιμώμενες ετικέτες, αν το b_i^t είναι 1, λέμε ότι το x^t ανήκει στην ομάδα των \mathbf{m}_i . Στη συνέχεια, αφού έχουμε αυτές τις ετικέτες, θα ελαχιστοποιήσουμε την εξίσωση 1.33. Λαμβάνοντας την παράγωγο της σε σχέση με τα \mathbf{m}_i και θέτοντας την ίση με 0, παίρνουμε

$$\mathbf{m}_i = \frac{\sum_t b_i^t x^t}{\sum_t b_i^t} \quad (1.35)$$

Το διάνυσμα αναφοράς έχει οριστεί ως ο μέσος όρος για όλα τα παραδείγματα που αντιπροσωπεύει. Αυτή είναι μια επαναληπτική διαδικασία, διότι τη στιγμή που θα υπολογιστεί το νέο \mathbf{m}_i , το b_i^t αλλάζει και πρέπει να υπολογιστεί εκ νέου το καινούργιο το οποίο με τη σειρά του επηρεάζει το \mathbf{m}_i . Αυτά τα δύο βήματα επαναλαμβάνονται έως ότου το \mathbf{m}_i σταθεροποιηθεί. Ένα μειονέκτημα είναι ότι αυτή είναι μια τοπική διαδικασία αναζήτησης και το τελικό \mathbf{m}_i εξαρτάται σε μεγάλο βαθμό από το αρχικό \mathbf{m}_i . Μετά από τη σύγκλιση, όλα τα κέντρα θα πρέπει να καλύπτουν κάποιο υποσύνολο από τα παραδείγματα των δεδομένων, ως εκ τούτου είναι καλύτερο να προετοιμάζουμε τα κέντρα, όπου πιστεύουμε ότι υπάρχουν τα δεδομένα.

I.4. Ανταγωνιστική Μάθηση

I.4.1. Online Ομαδοποίηση

Οι μαθησιακοί μέθοδοι που προτείνουμε σε αυτήν την ενότητα είναι *online*. Δεν έχουμε το σύνολο του δείγματος κατά τη διάρκεια της εκπαίδευσης, αντιθέτως λαμβάνουμε τα παραδείγματα ένα προς ένα και ενημερώνουμε τις παραμέτρους του μοντέλου, όπως τα λαμβάνουμε. Ο όρος *ανταγωνιστική μάθηση* χρησιμοποιείται διότι αυτές οι ομάδες, ή μάλλον οι μονάδες που αντιπροσωπεύουν αυτές οι ομάδες, ανταγωνίζονται μεταξύ τους για να βρεθεί αυτή που είναι πλησιέστερη, δηλαδή υπεύθυνη, για την εκπροσώπηση ενός παραδείγματος. Το μοντέλο ονομάζεται επίσης και *winner-take-all*, διότι όταν μια ομάδα κερδίσει ενημερώνεται μόνο αυτή, και οι άλλες δεν ενημερώνονται καθόλου.

Αυτές οι μέθοδοι μπορούν να χρησιμοποιηθούν από μόνες τους για online ομαδοποίηση, σε αντίθεση με τις offline μεθόδους που συζητήθηκαν στην ενότητα 3. Μια online μέθοδος έχει τα συνήθη πλεονεκτήματα: (1) δεν χρειάζεται επιπλέον μνήμη για να αποθηκεύσει όλο το σύνολο εκπαίδευσης, (2) η ενημέρωση σε κάθε βήμα είναι απλή στην εφαρμογή της, και (3), η κατανομή των εισροών μπορεί να αλλάξει στο χρόνο και το μοντέλο προσαρμόζεται σε αυτές τις αλλαγές αυτόματα. Αν επρόκειτο να χρησιμοποιήσουμε έναν offline αλγόριθμο, πρέπει να συλλέξουμε ένα νέο δείγμα και να τρέξουμε με την offline μέθοδο από την αρχή σε όλο το δείγμα.

I.4.1.1. Online k-Means

Στην εξίσωση I.33, ορίσαμε το σφάλμα ανοικοδόμησης σαν

$$E(\{\mathbf{m}_i\}_{i=1}^k | X) = \frac{1}{2} \sum_t \sum_i b_i^t \|\mathbf{x}^t - \mathbf{m}_i\|^2 \quad (I.36)$$

όπου

$$b_i^t = \begin{cases} 1, & \text{if } \|\mathbf{x}^t - \mathbf{m}_i\| = \min_l \|\mathbf{x}^t - \mathbf{m}_l\| \\ 0, & \text{otherwise} \end{cases} \quad (I.37)$$

$X = \{\mathbf{x}^t\}_t$ είναι το δείγμα και \mathbf{m}_i , $i = 1, \dots, k$ είναι τα κέντρα των ομάδων. b_i^t είναι 1 αν \mathbf{m}_i είναι το κοντινότερο κέντρο στο \mathbf{x}^t με βάση την Ευκλείδεια απόσταση. Είναι σαν όλα τα \mathbf{m}_l , $l = 1, \dots, k$ να ανταγωνίζονται και το \mathbf{m}_i να κερδίσει το διαγωνισμό, επειδή είναι το πιο κοντινό.

Ο offline αλγόριθμος, *k-means*, ενημερώνει τα κέντρα, ως

$$\mathbf{m}_i = \frac{\sum_t b_i^t \mathbf{x}^t}{\sum_t b_i^t} \quad (I.38)$$

η οποία ελαχιστοποιεί την εξίσωση I.36, τη στιγμή που οι νικητές επιλέγονται από την εξίσωση I.37. Όπως είδαμε και πριν, αυτά τα δύο στάδια του υπολογισμού των b_i^t και της ενημέρωσης των \mathbf{m}_i επαναλαμβάνονται μέχρι τη σύγκλιση.

Μπορούμε να τρέξουμε τον *online k-means* κάνοντας *stochastic gradient descent*, λαμβάνοντας υπόψη τα παραδείγματα ένα προς ένα, και να κάνουμε μια μικρή ενημέρωση σε κάθε βήμα, χωρίς να ξεχνάμε το αποτέλεσμα των προηγούμενων ενημερώσεων. Το σφάλμα ανοικοδόμησης για ένα μόνο παράδειγμα, είναι

$$E^t(\{\mathbf{m}_i\}_{i=1}^k | \mathbf{x}^t) = \frac{1}{2} \sum_i b_i^t \|\mathbf{x}^t - \mathbf{m}_i\|^2 = \frac{1}{2} \sum_i \sum_{j=1}^d b_i^t (x_j^t - m_{ij})^2 \quad (I.39)$$

όπου b_i^t ορίζεται όπως στην εξίσωση I.37. Χρησιμοποιώντας gradient descent σε αυτό, θα έχουμε τον ακόλουθο κανόνα ενημέρωσης για κάθε παράδειγμα, \mathbf{x}^t :

$$\Delta m_{ij} = -\eta \frac{\partial E^t}{\partial m_{ij}} = \eta b_i^t (x_j^t - m_{ij}) \quad (I.40)$$

Αυτό μετακινεί το πλησιέστερο κέντρο (για το οποίο ισχύει $b_i^t = 1$) προς την είσοδο με ένα συντελεστή που δίνεται από το η . Τα άλλα κέντρα έχουν το $b_l^t, l \neq i$ τους ίσο με 0 και δεν έχουν ενημερωθεί. Μια offline διαδικασία μπορεί επίσης να οριστεί με άθροιση της εξίσωσης I.40 πάνω από όλα τα t . Όπως σε κάθε διαδικασία gradient descent, ένας όρος ορμής μπορεί επίσης να προστεθεί. Για να έχουμε σύγκλιση, το η σταδιακά μειώνεται στο 0. Αλλά αυτό προϋποθέτει το *stability-plasticity dilemma*. Εάν το η μειώνεται προς το 0, το δίκτυο γίνεται σταθερό, αλλά χάνουμε την προσαρμοστικότητα σε νέα πρότυπα που μπορεί να συμβούν προϊόντος του χρόνου επειδή οι ενημερώσεις γίνονται πάρα πολύ μικρές. Αν κρατάμε το η μεγάλο, το \mathbf{m}_i μπορεί να ταλαντώνεται.

I.4.1.2. Θεωρία Προσαρμοστικού Συντονισμού (Adaptive Resonance Theory)

Στον offline και online k-means ο αριθμός των ομάδων, k , θα πρέπει να είναι γνωστός και καθορίζεται πριν να υπολογιστούν οι παράμετροι. Μια άλλη προσέγγιση είναι η *προοδευτική*, όπου ξεκινάμε με μια απλή ομάδα και προσθέτουμε νέες ομάδες, όταν είναι απαραίτητο. Συζητάμε τον *adaptive resonance theory* (ART) αλγόριθμο ως παράδειγμα ενός προοδευτικού αλγορίθμου. Στον ART, δεδομένης μιας εισόδου, όλες οι μονάδες εξόδου υπολογίζουν τις τιμές τους και η πιο όμοια με την είσοδο επιλέγεται. Αυτή είναι η μονάδα με τη μέγιστη τιμή, εάν η μονάδα χρησιμοποιεί το dot product, ή είναι η μονάδα με την ελάχιστη τιμή, εάν η μονάδα χρησιμοποιεί την Ευκλείδεια απόσταση.

Ας υποθέσουμε ότι χρησιμοποιούμε την Ευκλείδεια απόσταση. Εάν η ελάχιστη τιμή είναι μικρότερη από ένα ορισμένο όριο, που ονομάζεται *ευαισθησία* (*vigilance*), η ενημέρωση γίνεται όπως και στον online k-means. Εάν αυτή η απόσταση είναι μεγαλύτερη από την *ευαισθησία*, μια νέα μονάδα εξόδου προστίθεται και το κέντρο της αρχικοποιείται με το συγκεκριμένο παράδειγμα. Αυτό ορίζει μια υπερσφαίρα της οποίας η ακτίνα δίνεται από την *ευαισθησία* και καθορίζει τον όγκο του πεδίου εφαρμογής της κάθε μονάδας, προσθέτουμε μια νέα μονάδα κάθε φορά που έχουμε μια είσοδο που δεν καλύπτεται από καμία μονάδα. Δηλώνοντας την *ευαισθησία* ως ρ , χρησιμοποιούμε τις παρακάτω εξισώσεις σε κάθε ενημέρωση:

$$b_i = \|\mathbf{m}_i - \mathbf{x}^t\| = \min_{l=1}^k \|\mathbf{m}_l - \mathbf{x}^t\| \quad (I.41)$$

$$\begin{cases} \mathbf{m}_{k+1} \leftarrow \mathbf{x}^t, & \text{if } b_i > \rho \\ \Delta \mathbf{m}_i = \eta(\mathbf{x} - \mathbf{m}_i), & \text{otherwise} \end{cases}$$

Θέτοντας ένα κατώτατο όριο για την απόσταση, αυτό είναι ισοδύναμο με την τοποθέτηση ενός ορίου για το σφάλμα ανοικοδόμησης για κάθε παράδειγμα, και αν η απόσταση είναι Ευκλείδεια και το σφάλμα ορίζεται από την εξίσωση I.39, αυτό δείχνει ότι το μέγιστο σφάλμα ανοικοδόμησης που επιτρέπεται ανά παράδειγμα είναι το τετράγωνο της ευαισθησίας.

ΠΑΡΑΡΤΗΜΑ II – ΘΕΩΡΙΑ ΒΕΛΤΙΣΤΗΣ ΠΑΥΣΗΣ

II.1. Ορισμός της Θεωρίας Βέλτιστης Παύσης

Το βασικό πλαίσιο πολλών προβλημάτων αφορά στη λήψη μιας απόφασης έπειτα από την παρατήρηση μιας διαδικασίας που εξελίσσεται χρονικά με τυχαίο τρόπο. Ο παρατηρητής καλείται, βασισμένος μόνο σε διαδοχικά παρατηρηθείσες τυχαίες μεταβλητές, να επιτύχει το καλύτερο αποτέλεσμα. Δεδομένης, λοιπόν, της ακολουθίας των παρατηρήσεων X_1, X_2, \dots, X_n ο χρόνος τερματισμού αποτελεί τη φόρμουλα που ορίζει αν ο παρατηρητής πρέπει να σταματήσει στο βήμα n . Ως παραδείγματα χρόνου τερματισμού μπορούν να αναφερθούν τα παρακάτω :

1. η 5^η επίσκεψη στην κατάσταση X ,
2. 10 λεπτά μετά τη 2^η επίσκεψη στην κατάσταση y , ή
3. η στιγμή κατά την οποία το άθροισμα $X_1 + X_2 + \dots + X_n$ ξεπερνά το 100 κ.ο.κ.

Με την θεωρία του βέλτιστου χρονικού τερματισμού μελετάται το πρόβλημα της επιλογής της χρονικής στιγμής για τη λήψη μιας απόφασης, ή την εκτέλεση μιας συγκεκριμένης λειτουργίας, ώστε να μεγιστοποιηθεί το αναμενόμενο κέρδος ή να ελαχιστοποιηθεί το αναμενόμενο κόστος. Προβλήματα αυτού του είδους συναντώνται στον τομέα της στατιστικής, όπου η απόφαση μπορεί να αφορά στην εξέταση μιας υπόθεσης ή στην εκτίμηση μιας παραμέτρου, καθώς και στον τομέα της λειτουργικής έρευνας, όπου η απόφαση μπορεί να αφορά στην αντικατάσταση μιας μηχανής, την πρόσληψη μιας γραμματέως ή την επαναπαραγγελία αποθέματος κ.λ.π. [31].

Τα προβλήματα τερματισμού ορίζονται από δύο αντικείμενα [9] :

1. Μια ακολουθία τυχαίων μεταβλητών X_1, X_2, \dots , των οποίων η κοινή κατανομή θεωρείται γνωστή, και
2. Μια ακολουθία συναρτήσεων ανταμοιβής με πραγματικές τιμές,
 $y_0, y_1(x_1), y_2(x_1, x_2), \dots, y_\infty(x_1, x_2, \dots)$.

Με βάση τα αντικείμενα αυτά, το πρόβλημα τερματισμού περιγράφεται ως ακολούθως [9]. Παρατηρείται η ακολουθία X_1, X_2, \dots , για ένα συγκεκριμένο χρονικό διάστημα. Για κάθε $n = 1, 2, \dots$, μετά τις παρατηρήσεις $X_1 = x_1, X_2 = x_2, \dots, X_n = x_n$, ο παρατηρητής είτε επιλέγει να σταματήσει και να λάβει το γνωστό αντίτιμο $y_n(x_1, \dots, x_n)$, είτε επιλέγει να συνεχίσει και να λάβει την παρατήρηση X_{n+1} . Αν η επιλογή του είναι να μην διαλέξει καμία παρατήρηση, τότε λαμβάνει το σταθερό ποσό y_0 . Αν δεν σταματήσει ποτέ, τότε λαμβάνει το ποσό $y_\infty(x_1, x_2, \dots)$.

Το πρόβλημα αφορά στην επιλογή της κατάλληλης χρονικής στιγμής στην οποία πρέπει να σταματήσει ο παρατηρητής ώστε να μεγιστοποιηθεί το αναμενόμενο κέρδος. Επιτρέπεται η χρήση τυχαίων αποφάσεων. Δηλαδή, αν ο παρατηρητής φτάσει στο στάδιο n έχοντας παρατηρήσει τα $X_1 = x_1, X_2 = x_2, \dots, X_n = x_n$ μπορεί να επιλέξει να σταματήσει με μια πιθανότητα που να εξαρτάται από αυτές τις παρατηρήσεις. Η πιθανότητα αυτή δηλώνεται ως $\varphi_n(x_1, \dots, x_n)$.

Ένας κανόνας τερματισμού αποτελείται από την ακολουθία των συναρτήσεων $\varphi = (\varphi_0, \varphi_1(x_1), \varphi_2(x_1, x_2), \dots)$ για όλα τα n και τα x_1, x_2, \dots, x_n , με $0 \leq \varphi_n(x_1, \dots, x_n) \leq 1$. Συνεπώς το φ_0 αναπαριστά την πιθανότητα κατά την οποία δεν έχει γίνει καμία παρατήρηση. Στην περίπτωση που λαμβάνεται η πρώτη παρατήρηση $X_1 = x_1$, τότε το $\varphi_1(x_1)$ αναπαριστά την πιθανότητα να σταματήσει ο παρατηρητής μετά την πρώτη παρατήρηση, κ.ο.κ.. Με βάση τον κανόνα τερματισμού φ και την ακολουθία παρατηρήσεων $X = (X_1, X_2, \dots) = (x_1, x_2, \dots)$ καθορίζεται η τυχαία χρονική στιγμή N , με $0 < N \leq \infty$, στην οποία θα πραγματοποιηθεί ο βέλτιστος τερματισμός.

Η συνάρτηση μάζας πιθανότητας του N δηλώνεται με $\psi = (\psi_0, \psi_1, \psi_2, \dots, \psi_\infty)$, όπου $\psi_n = (x_1, x_2, \dots, x_n) = P(N = n | X = x)$ για $n = 0, 1, 2, \dots$ και $\psi_\infty = (x_1, x_2, \dots) = P(N = \infty | X = x)$. Η συσχέτισή της με τον κανόνα τερματισμού φ γίνεται ως εξής:

$$\begin{aligned} \psi_0 &= \varphi_0 \\ \psi_1(x_1) &= (1 - \varphi_0) \cdot \varphi_1(x_1) \\ &\vdots \\ \psi_n(x_1, x_2, \dots, x_n) &= \left[\prod_{j=1}^{n-1} (1 - \varphi_j(x_1, x_2, \dots, x_j)) \right] \cdot \varphi_n(x_1, x_2, \dots, x_n) \\ &\vdots \\ \psi_\infty(x_1, x_2, \dots) &= 1 - \sum_0^\infty \psi_j(x_1, x_2, \dots, x_j), \end{aligned}$$

όπου $\psi_\infty(x_1, x_2, \dots)$ είναι η πιθανότητα να μην γίνει τερματισμός, αφού έχουν ληφθεί όλες οι παρατηρήσεις.

Επομένως το πρόβλημα μετασχηματίζεται στην επιλογή του κανόνα τερματισμού φ που θα οδηγήσει σε μεγιστοποίηση του κέρδους $V(\varphi)$, που ορίζεται ως $V(\varphi) = E_{yN}(X_1, \dots, X_n) = E \sum_{j=0}^\infty \psi_j(X_1, \dots, X_j) y_j(X_1, \dots, X_j)$. Σαν συνάρτηση του τυχαίου χρόνου τερματισμού, ο κανόνας τερματισμού φ μπορεί να εκφραστεί ως $\varphi_n = (X_1, X_2, \dots, X_n) = P(N = n | N \geq n, X = x)$ για $n = 0, 1, 2, \dots$

Σε κάποιες εφαρμογές, η ακολουθία ανταμοιβής περιγράφεται πιο ρεαλιστικά ως μια ακολουθία από τυχαίες μεταβλητές $Y_0, Y_1, \dots, Y_\infty$ των οποίων η κοινή κατανομή με τις παρατηρήσεις X_1, X_2, \dots είναι γνωστή. Η πραγματική τιμή του Y_n μπορεί να μην είναι γνωστή τη χρονική στιγμή n όταν πρέπει να ληφθεί η απόφαση για παύση ή συνέχιση των παρατηρήσεων.

Ωστόσο, το να επιτρέπεται να είναι τυχαίο το κέρδος δεν αποτελεί γενικότητα, καθώς, δεδομένου ότι η απόφαση για τερματισμό τη χρονική στιγμή n μπορεί να εξαρτάται από τα X_1, \dots, X_n , μπορεί να αντικατασταθεί η ακολουθία των τυχαίων ανταμοιβών Y_n από την ακολουθία των συναρτήσεων ανταμοιβής $y_n(x_1, \dots, x_n)$, με $n = 0, 1, \dots, \infty$, όπου $y_n(x_1, \dots, x_n) = E\{Y_n | X_1 = x_1, \dots, X_n = x_n\}$. Οποιοσδήποτε κανόνας παύσης φ για την ακολουθία κέρδους $Y_0, Y_1, \dots, Y_\infty$ θα έδινε την ίδια αναμενόμενη τιμή για την ακολουθία $y_0, y_1, \dots, y_\infty$.

II.2. Προβλήματα Πεπερασμένου Ορίζοντα - Πρόβλημα της Γραμματέως

Μια κατηγορία προβλημάτων τερματισμού είναι τα προβλήματα πεπερασμένου ορίζοντα. Ένα πρόβλημα τερματισμού θεωρούμε ότι έχει πεπερασμένο ορίζοντα όταν υπάρχει ένα γνωστό άνω όριο ως προς τον αριθμό των σταδίων στα οποία μπορεί να σταματήσει ο παρατηρητής. Εάν λαμβάνει τις παρατηρήσεις X_1, \dots, X_T και πρέπει να σταματήσει μετά από την παρατήρηση X_T , τότε λέμε ότι το πρόβλημα έχει ορίζοντα T . Ένα πρόβλημα πεπερασμένου ορίζοντα μπορεί να ληφθεί ως μια ειδική περίπτωση του γενικού προβλήματος που παρουσιάστηκε προηγουμένως θέτοντας $y_{T+1} = \dots = y_\infty = -\infty$.

Σε γενικές γραμμές, τα προβλήματα αυτά μπορούν να λυθούν με τη μέθοδο της οπισθοδρομικής επαγωγής (backward induction). Δεδομένου ότι ο παρατηρητής πρέπει να σταματήσει στο στάδιο T , βρίσκεται αρχικά ο βέλτιστος κανόνας στο στάδιο $T - 1$. Κατόπιν, γνωρίζοντας τον κανόνα αυτό, βρίσκεται ο βέλτιστος κανόνας στο προηγούμενο στάδιο $T - 2$, κ.ο.κ μέχρι το αρχικό στάδιο 0.

Διάφορα προβλήματα αξιολογούνται ως προβλήματα πεπερασμένου ορίζοντα, με πιο γνωστό από αυτά το πρόβλημα της γραμματέως. Το πρόβλημα της γραμματέως και οι παραλλαγές του αποτελούν μια σημαντική τάξη των προβλημάτων πεπερασμένου ορίζοντα. Το πρόβλημα αυτό αναφέρεται, συνήθως, στην επιλογή της καλύτερης γραμματέως από ένα σύνολο υποψηφίων. Πολλές φορές, όμως, αναφέρεται και στην επιλογή της καλύτερης συζύγου ή του μεγαλύτερου αριθμού από ένα σύνολο άγνωστων αριθμών (Googol).

Το κλασσικό πρόβλημα της γραμματέως μπορεί να περιγραφεί ως εξής:

1. Υπάρχει μόνο μια διαθέσιμη θέση γραμματέως.
2. Υπάρχουν n αιτούντες για την θέση, όπου το n είναι γνωστός αριθμός.
3. Είναι δυνατή η γραμμική ταξινόμηση των αιτούντων από τον καλύτερο στον χειρότερο χωρίς να υπάρχουν ισοβαθμίες.
4. Οι αιτούντες περνούν από συνέντευξη διαδοχικά με τυχαία διάταξη και όλες οι $n!$ διατάξεις είναι ισοπίθανες.
5. Μετά από κάθε συνέντευξη, ο αιτών είτε προσλαμβάνεται για τη θέση, οπότε το πρόβλημα απόφασης λήγει, είτε απορρίπτεται, οπότε έρχεται η σειρά του επόμενου εξεταζόμενου.
6. Το αν θα γίνει αποδεκτός ή θα απορριφθεί ένας υποψήφιος βασίζεται μόνο στη σχετική διάταξη των υποψηφίων που έχουν περάσει από συνέντευξη μέχρι εκείνη τη στιγμή.
7. Από τη στιγμή που ο αιτών απορριφθεί δεν μπορεί αργότερα να γίνει αποδεκτός.
8. Ο στόχος είναι η επιλογή του καλύτερου αιτούντος, οπότε στην περίπτωση αυτή το κέρδος είναι 1, διαφορετικά είναι 0.

Θεωρούνται οι παρατηρήσεις X_1, X_2, \dots, X_n όπου X_j είναι η κατάταξη του j αιτούντα στην μέχρι τώρα διάταξη, με το 1 να είναι ο βέλτιστος αιτών. Σύμφωνα με την υπόθεση 4, οι τυχαίες μεταβλητές X_j είναι ανεξάρτητες και έχουν ομοιόμορφη κατανομή από το 1 μέχρι το j . Δηλαδή, για 2 αιτούντες X_1 και X_2 η πιθανότητα ο X_2 να είναι ο 1^{ος} στην κατάταξη είναι ίδια με την πιθανότητα να είναι 2^{ος} στην κατάταξη $P(X_2 = 1) = P(X_2 = 2) = 1/2$ κ.τ.λ.

Πρέπει να σημειωθεί ότι ένας αιτών γίνεται αποδεκτός μόνο αν είναι σχετικά καλύτερος από αυτούς που έχουν ήδη περάσει. Ένας σχετικά καλύτερος (βέλτιστος) αιτών ονομάζεται υποψήφιος. Επομένως, ο j αιτών είναι υποψήφιος αν και μόνο αν $X_j = 1$. Η πιθανότητα ο υποψήφιος στο στάδιο j να είναι η συνολικά βέλτιστη επιλογή σε σύνολο n αιτούντων είναι j/n . Επομένως

$$y_j(x_1, \dots, x_j) = \begin{cases} j/n & \text{Αν ο εξεταζόμενος } j \text{ είναι υποψήφιος} \\ 0 & \text{διαφορετικά} \end{cases}$$

Σημειώνουμε ότι για $y_0 = 0$ και για $j \geq 1$, το y_j εξαρτάται μόνο από το x_j .

Ένας βέλτιστος κανόνας για την επιλογή του καλύτερου υποψηφίου είναι, για κάποιο $r \geq 1$, να απορριφθούν οι πρώτοι $r - 1$ αιτούντες και στη συνέχεια να προσληφθεί ο επόμενος σχετικά βέλτιστος υποψήφιος (αν υπάρχει). Αποδεικνύεται ότι για ένα μεγάλο αριθμό αιτούντων είναι βέλτιστο να απορριφθεί ένα ποσοστό ίσο με το $e^{-1} = 36.8\%$ και να επιλεγεί ο πρώτος υποψήφιος με τη σχετικά καλύτερη επίδοση μέχρι εκείνη τη στιγμή. Η πιθανότητα να επιλεγεί ο καλύτερος υποψήφιος είναι τότε ίση με e^{-1} .

ΑΝΑΦΟΡΕΣ

- [1] A. Dey, Understanding and using context, *Personal and Ubiquitous Computing*, 5(1), pp. 4-7, 2001.
- [2] J. Hightower, G. Borriello, *Location Systems for Ubiquitous Computing*, *IEEE Computer*, 34(8), August, 2001.
- [3] I. Priggouris, E. Zervas and S. Hadjiefthymiades, "Location Based Network Resource Management", in "Handbook of Research on Mobile Multimedia" (editor: Ismail Khalil Ibrahim), Idea Group Inc., May 2006.
- [4] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, "Efficient Location Prediction in Mobile Cellular Networks", *International Journal of Wireless Information Networks*, Springer, to appear, Springer Verlag, December, 2011.
- [5] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, "An Adaptive Machine Learning Algorithm for Location Prediction", *International Journal of Wireless Information Networks* (2011) 18:88-99, Springer, April, 2011
- [6] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, "An Adaptive Location Prediction Model based on Fuzzy Control", *Computer Communications* 34 (2011) 816-834, Elsevier, Elsevier, January, 2011
- [7] C. Anagnostopoulos, S. Hadjiefthymiades, T. Anagnostopoulos, "Movement Prediction: A Delay Classification Approach for Improving Prediction Score", Submitted to *IEEE Transactions on Mobile Computing*, November, 2011.
- [8] E. Alpaydin, *Introduction to Machine Learning*, The MIT Press, 2004.
- [9] R. Duda, P. Hart, D. Stork, *Pattern Classification*, Wiley-Interscience, 2001.
- [10] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning, Data mining, Inference and Prediction*, Springer, (Second Edition 2009).
- [11] Quinlan, *C4.5: Programs for Machine Learning*, MK Series in Machine Learning, 1993.
- [12] J. Killer, M. Hatef, R. Duin, J. Matas, On combining classifiers, *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 20(3), 1998.
- [13] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, M. Kyriakakos and A. Kalousis, Predicting the location of mobile users: a machine learning approach, *Proceedings of International Conference on Pervasive Services (ICPS) 2009*, London, UK.
- [14] T. Anagnostopoulos, C. Anagnostopoulos, S. Hadjiefthymiades, A. Kalousis and M. Kyriakakos, Path Prediction through Data Mining, *Proceedings of the IEEE International Conference on Pervasive Services (ICPS)*, pp. 128-135, Istanbul, Turkey, 2007.
- [15] G. Liu, G. Maguire Jr., A Class of Mobile Motion Prediction Algorithms for Wireless Mobile Computing and Communications, *MONET*, 1, pp. 113-121, 1996.
- [16] A. Bhattacharya, S. Das, LeZi Update: An Information Theoretic Approach to Track Mobile Users In PCS Networks, *proceedings of ACM/IEEE Mobicom*, 1999.
- [17] F. Giannoti, M. Nanni, D. Pedreschi and F. Pinelli, Trajectory Pattern Mining, *KDD Int. Conf.*, 2007.
- [18] A. Aljadhari and T.F. Znati, Predictive Mobility Support for QoS Provisioning in Mobile Wireless Environments, *IEEE JSAC*, Vol. 19, No. 10, October 2001.
- [19] M. Poulakis, V. Vassaki, and S. Hadjiefthymiades, Proactive Radio Resource Management Using Optimal Stopping Theory, *proceedings of WoW MoM 2009, 10th IEEE International Symposium on World of Wireless, Mobile and Multimedia Networks*, Kos, Greece, June 2009.
- [20] J-M Francois, G. Leduc, and S. Martin, Learning Movement Patterns in Mobile Networks: a generic method, *proceedings of European Wireless*, pp. 128-134, 2004.
- [21] Cheng, R. Jain, E. van den Berg, Location prediction algorithms for mobile wireless systems, *Wireless Internet handbook: technologies, standards, and application*, 245-263, CRC Press, 2003.
- [22] K. M. Curewitz, P. Krishnan, and J. S. Vitter, Practical Prefetching via Data Compression, *Proceedings of ACM SIG MOD*, 1993, pp. 257-266.
- [23] K. Narendra and M. A. L. Thathachar, *Learning Automata – An Introduction*, Prentice Hall, 1989.
- [24] E. Belogay, C. Cabrelli, U. Molter, and R. Shonkwiler, Calculating the Hausdorff Distance between Curves, *Information Processing Letters*, vol. 64, no. 1, pp. 17-22, 1997.
- [25] Site: <http://www.openstreetmap.org/traces/>
- [26] T. C. Bell, J. G. Cleary, I.H. Witten, *Text Compression*, Prentice Hall, Englewood cliffs, NJ, 1990.

- [27] Z.-Y. Zhao, M. Tomizuka, S. Isaka, Fuzzy gain scheduling of PID controllers, *IEEE Trans. Syst., Man, Cybern.*, vol. 23, pp. 1392-1398, 1993.
- [28] C. W. de Silva, *Intelligent Control: Fuzzy Logic Applications*. New York: CRC, 1995.
- [29] T. Anagnostopoulos, C. Anagnostopoulos, and S. Hadjiefthymiades, "An Online Adaptive Model for Location Prediction", in *Proc. 3rd International ICST Conference on Autonomic Computing and Communication Systems (Autonomics 2009)*, LNICST 23, pp. 64-78, 2010.
- [30] C. Loader, *Local Regression and Likelihood*, New York, Springer, 1999.
- [31] T. S. Ferguson, Mathematics Department, UCLA, "Optimal Stopping and Applications", [Online], Available: <http://www.math.ucla.edu/~tom/Stopping/Contents.html> , [Accessed: Aug. 25, 2010].
- [32] A. Shiryaev, "Optimal Stopping Rules", New York: Springer, 1978.
- [33] G. Cardillo, K. Fu, "A dynamic programming procedure for sequential pattern classification and feature selection", *Mathematical Biosciences*, 1(3), Sept. 1967, pp. 463-491.
- [34] Y. Chien, K. Fu, "A modified sequential recognition machine using timevarying stopping boundaries", *IEEE Trans. On Information Theory* 12(2), 206-214, 1966.
- [35] K. R. Evensen, A. Petlund, H. Riiser, P. Vigmostad, D. Kaspar, C. Griwodz, and P. Halvorsen. "Mobile video streaming using location-based network prediction and transparent handover", *ACM NOSSDAV*, pp. 21-26, 2011.
- [36] D. Siegmund, "Sequential Analysis. Test and Confidence Intervals", *Springer Series in Statistics*, Springer-Verlag, New York, 1985.
- [37] J. Francois, "Performing and Making use of Mobility Prediction", PhD thesis (Ch. 4), University of Liege, 2007.
- [38] Y. Chen, K. Jiang, Y. Zheng, C. Li, N. Yu, "Trajectory simplification method for location-based social networking services", In *Proc. Of the Intl. Workshop on Location Based Social Networks*, ACM, 33-40, 2009.
- [39] H. Cao, O. Wolfson, G. Trajcevski, "Spatio-temporal data reduction with deterministic error bounds", In *Proc. Of the 2003 ACM joint workshop on Foundations of mobile computing (DIALM-POMC '03)*, 2003.
- [40] M. Vlachos, G. Kollios, and D. Gunopoulos, "Elastic Translation Invariant Matching of Trajectories", *Machine Learning*, vol. 58, Issue 2-3, 2005, pp. 301-334.
- [41] A. Croitoru, P. Agouris, A. Stefanidis, "3D trajectory matching by pose normalization", In *Proc. Of the 13th ACM Intl. workshop on GIS*, New York, USA: ACM Press, 2005, pp.153-162.
- [42] D. Berndt, J. Clifford, "Using dynamic time warping to find patterns in time series", In *Proc. of ACM SIGKDD*, 1994, pp. 359-370.
- [43] S. Grossberg, "Competitive learning: From interactive activation to adaptive resonance", *Cognitive science*, vol. 11, pp. 23-63, 1987.
- [44] Sang-Jun Han, Sung-Bae Cho, "Predicting User's Movement with a Combination of Self-Organizing Map and Markov Model", In *Proc. of 16th Intl. Conf. Artificial Neural Networks (ICANN'06)*, LNCS 4132, pp. 884-893, 2006, Springer.
- [45] S. Choi, K. G. Shin, "Predictive and adaptive bandwidth reservation for hand-offs in QoS-sensitive cellular networks", *ACM SIGCOMM*, 1998.
- [46] S. Hadjiefthymiades, L. Merakos, "Proxies + Path Prediction: Improving Web Service Provision in Wireless-Mobile Communications", *ACM/Kluwer Mobile Networks and Applications*, S.I. Mobile and Wireless Data Management, 8(4), 2003.
- [47] A. Karmouch and N. Samaan, "A Mobility Prediction Architecture Based on Contextual Knowledge and Spatial Conceptual Maps", *IEEE Trans. on Mobile Computing*, 4(6), 2005.
- [48] H. Jeung, Q. Liu, H. Tao Shen, X. Zhou, "A Hybrid Prediction Model for Moving Objects", In *Proc. of the 24th IEEE Intl. Conf. on Data Engineering*, pp. 70-79, 2008.
- [49] K. Majumdar, N. Das, "Mobile User Tracking Using A Hybrid Neural Network", *Springer Wireless Networks*, vol. 11, pp. 275-284, 2005.
- [50] H. Jeung, M. L. Yiu, X. Zhou, C. S. Jensen, "Path prediction and predictive range querying in road network databases", *Springer VLDB*, 19:585-602, 2010.
- [51] C. Hermes, C. Woehler, K. Schenk, F. Kummert, "Long-term vehicle motion prediction", In *Proc. of the IEEE Intelligent Vehicles Symposium*, pp. 652-657, 2009.

- [52] S. Akoush, A. Sameh, "Mobile User Movement Prediction Using Bayesian Learning for Neural Networks", In Proc. of the ACM Intl. Wireless Communications and Mobile Computing Conf., 2007.
- [53] I. Burbey, T. L. Martin, "Predicting Future Locations Using Prediction-by-Partial-Match", In Proc. of the 1st ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments, 2008.
- [54] D. Ashbrook and T. Starner, "Learning Significant Locations and Predicting User Movement with GPS", In Proc. of the 6th Intl. Symp. Wearable Computers (ISWC 2002), pp. 101-108, 2002.
- [55] Y. Xiao, H. Zhang, H. Wang, "Location Prediction for Tracking Moving Objects Based on Grey Theory", In Proc. of the IEEE Fuzzy Systems and Knowledge Discovery, pp. 390-394, 2007.
- [56] R. Viayan and J. Holtman, "A model for analyzing handoff algorithms", IEEE Trans. on Veh. Technol., 42(3), Aug. 1993.
- [57] W. Wanalertlak, B. Lee, C. Yu, M. Kim, Seung-Min Park, Won-Tae Kim, "Behavior-based mobility prediction for seamless handoffs in mobile wireless networks", Springer, Wireless Networks, 17(3):645-658, 2011.
- [58] M. Varsta, J. Heikkopen, and J. Del Ruiz Mill, "Context learning with the self-organizing map", In Proc. of Workshop on Self-Organizing Maps, pages 197-202, 1997.
- [59] C. Bettini, X. Sean Wang, and Sushil Jajodia, "Protecting Privacy Against Location-Based Personal Identification", Springer, Secure Data Management, 2005.
- [60] J. Krumm, "A survey of computational location privacy", Personal and Ubiquitous Computing, 13:391-399, Springer, 2009.
- [61] G. A. Carpenter and S. Grossberg, "Adaptive Resonance Theory", In Michael A. Arbib (Ed.), The Handbook of Brain Theory and Neural Networks, Cambridge, MA: MIT Press, 2003.
- [62] F. T. Bruss, G. Louchard, "The Odds-algorithm based on sequential updating and its performance", Adv. App. Prob. 41(1):131-153, 2009.
- [63] K. Ano, M. Tamaki, M. Hu, "A secretary problem with uncertain employment when the number of offers is restricted", J. Operation Research Society of Japan, 39:307-315, 1996.
- [64] M. Kyriakakos, N. Frangiadakis, S. Hadjiefthymiades, L. Merakos, "RMPG: A Realistic Mobility Pattern Generator for the Performance Assessment of Mobility Functions", Simulation Modeling Practice And Theory, 12(1), Elsevier, 2004.