



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Μια πιθανοτική διάλεκτος

Λογισμού Γεγονότων

Ιάσων Φιλίππου

Επιβλέποντες: **Αλέξανδρος Αρτίκης**, Ερευνητής, ΕΚΕΦΕ «Δημόκριτος»
Γεώργιος Παλιούρας, Ερευνητής, ΕΚΕΦΕ «Δημόκριτος»
Παναγιώτης Σταματόπουλος, Επίκουρος Καθηγητής ΕΚΠΑ

ΑΘΗΝΑ

ΑΠΡΙΛΙΟΣ 2012

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Μια πιθανοτική διάλεκτος

Λογισμού Γεγονότων

Ιάσων Φιλίππου

AM: 1115200600142

ΕΠΙΒΛΕΠΟΝΤΕΣ :

Αλέξανδρος Αρτίκης , Ερευνητής, ΕΚΕΦΕ «Δημόκριτος»

Γεώργιος Παλιούρας , Ερευνητής, ΕΚΕΦΕ «Δημόκριτος»

Παναγιώτης Σταματοπουλος, Επίκουρος Καθηγητής ΕΚΠΑ

ΠΕΡΙΛΗΨΗ

Παρουσιάζουμε ένα σύστημα αναγνώρισης ανθρώπινης συμπεριφοράς το οποίο λαμβάνει ως είσοδο μια συμβολική αναπαράσταση από βίντεο. Αναλυτικότερα, η είσοδος του συστήματός μας είναι ένα σύνολο από χρονικά επισημειωμένες δραστηριότητες χαμηλού επιπέδου (Short-Term Activities — STA). Η έξοδος του συστήματος είναι ένα σύνολο αναγνωρισμένων δραστηριοτήτων υψηλού επιπέδου (Long-Term Activities — LTA), οι οποίες είναι χρονικοί συνδυασμοί συγκεκριμένων STA. Οι περιορισμοί επί των STA, οι οποίοι εφ' όσον ικανοποιηθούν οδηγούν στην αναγνώριση μιας LTA, έχουν οριστεί με χρήση μιας διαλέκτου Λογισμού Γεγονότων (Event Calculus). Προκειμένου να αντιμετωπίσουμε την αβεβαιότητα που προκύπτει στην αναγνώριση ανθρώπινης συμπεριφοράς, προσαρμόσαμε την εν λόγω διάλεκτο σε ένα σύγχρονο σύστημα πιθανοτικού λογικού προγραμματισμού. Παρουσιάζουμε μια λεπτομερή εκτίμηση και σύγκριση της ντετερμινιστικής και της πιθανοτικής προσεγγίσης μέσω πειραματισμού σε ένα σύνολο δεδομένων που αποτελείται από βίντεο επιτήρησης ανθρώπινης δραστηριότητας.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Τεχνητή Νοημοσύνη

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Αναγνώριση Γεγονότων, Αναγνώριση Προτύπων, Λογισμός Γεγονότων, ProbLog

ABSTRACT

We present a system for recognising human activity given a symbolic representation of video content. The input of our system is a set of time-stamped short-term activities (STA) detected on video frames. The output is a set of recognised long-term activities (LTA), which are pre-defined temporal combinations of STA. The constraints on the STA that, if satisfied, lead to the recognition of a LTA, have been expressed using a dialect of the Event Calculus. In order to handle the uncertainty that naturally occurs in human activity recognition, we adapted this dialect to a state-of-the-art probabilistic logic programming framework. We present a detailed evaluation and comparison of the crisp and probabilistic approaches through experimentation on a benchmark dataset of human surveillance videos.

SUBJECT AREA: Artificial Intelligence

KEYWORDS: Event Recognition, Pattern Matching, Event Calculus, ProbLog

Στη Ζωή

ΕΥΧΑΡΙΣΤΙΕΣ

Πρωτίστως αισθάνομαι την υποχρέωση να ευχαριστήσω τους επιβλέποντες της παρούσης εργασίας και επιστημονικούς μου συνεργάτες στο Δημόκριτο Γιώργο Παλιούρα και Αλέξανδρο Αρτίκη. Με το να μου επιδεικνύουν συνεχώς όλα μου τα λάθη και να πραγματοποιούν συνεχείς διορθώσεις επί του τελικού κειμένου της εργασίας, μου δίδαξαν το πώς πρέπει κανείς να κάνει έρευνα, σε μια εποχή που στην Ελλάδα κάτι τέτοιο θεωρείται ενασχόληση πολυτελείας.

Η εργασία αυτή δε θα μπορούσε να είχε ολοκληρωθεί χωρίς τη βοήθεια του συναδέλφου Αναστάσιου Σκαρλατίδη, ο οποίος συνεισέφερε ένα μεγάλο κομμάτι κώδικα που κατέστησε εφικτή την παραγωγή ενός μεγάλου όγκου πειραματικών αποτελεσμάτων σε ένα μικρό σχετικά χρονικό διάστημα.

Είμαι καθ' όλα υπόχρεος στους καθηγητές του Τμήματος Πληροφορικής και Τηλεπικοινωνιών ΕΚΠΑ Δρ. Παναγιώτη Σταματόπουλο και Δρ. Ιζαμπώ Καραλη, οι οποίοι στο θερινό εξάμηνο του 2009 με το μάθημά του Λογικού Προγραμματισμού με οδήγησαν στα σημερινά μου ερευνητικά ενδιαφέροντα, όπως επίσης και στον καθηγητή Δρ. Αλέξιο Δελή, ο οποίος με στήριξε και με ενέπνευσε εξίσου στην ακαδημαϊκή μου πορεία μέσα στο Τμήμα.

Οι φίλοι μου, εντός και εκτός σχολής, με κράτησαν ψυχολογικά υγιή όλα αυτά τα χρόνια και η οικογένειά μου μου δίδαξε από νωρίς ότι επιτρέπεται να πέσεις, επιβάλλεται όμως και να σηκωθείς. Χωρίς τη στήριξή όλων τους, η πορεία μου θα ήταν, αδιαμφισβήτητα, διαφορετική.

Η εν λόγω εργασία είναι ερευνητικού περιεχομένου και αποτελεί προϊόν απασχόλησής μου στο ΕΚΕΦΕ «Δημόκριτος» η οποία υποστηρίχτηκε μερικώς από το ερευνητικό πρόγραμμα PRONTO (FP-7 ICT 231738).

ΠΕΡΙΕΧΟΜΕΝΑ

1	Εισαγωγή	19
1.1	Περιγραφή του Προβλήματος	19
1.2	Συνεισφορά	20
1.3	Διάρθρωση εργασίας	20
2	Σχετικές Εργασίες	23
2.1	Λογισμός Γεγονότων	23
2.2	ProbLog	24
2.3	Σχετικές εργασίες σε αναγνώριση δραστηριοτήτων	27
3	Λογισμός Γεγονότων για Αναγνώριση Ανθρώπινης Συμπεριφοράς	33
3.1	Αξιώματα της Crisp-EC	33
3.2	Το σύνολο δεδομένων CAVIAR	35
3.3	Ορισμοί των LTA της Crisp-EC	37
4	Πιθανοτικός Λογισμός Γεγονότων	41
4.1	Από την Crisp-EC στην Prob-EC	41
4.2	Συμπερασμός στην Prob-EC	43
5	Πειράματα	51
5.1	«Ομαλός» θόρυβος	55

5.2	«Οξύς» θόρυβος	58
5.3	Πειραματικά Συμπεράσματα	61
6	Σύνοψη / Μελλοντικές Εργασίες	63
6.1	Συμπεράσματα	63
6.2	Μελλοντικές εργασίες	63
	ΟΡΟΛΟΓΙΑ	65
	ΣΥΝΤΜΗΣΕΙΣ - ΑΡΚΤΙΚΟΛΕΞΑ	67
	ΒΙΒΛΙΟΓΡΑΦΙΑ	69

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

4.1	Η διακύμανση της πιθανότητας του ερωτήματος $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, T)$	44
4.2	Το SLD-δένδρο για το επερώτημα $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, 22)$	45
4.3	Η έξοδος της Prob-EC για το επερώτημα $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, T)$ για διάφορα frames T	47
4.4	Παράδειγμα της διακύμανσης πιθανότητας μιας «ισχυρά εκκινούμενης» (strongly initiated) LTA.	49
5.1	Στιγμιότυπα της STA <i>walking</i> στο CAVIAR ανά μέση τιμή κατανομής Γάμμα. Η μέση τιμή 0.0 αναπαριστά το CAVIAR χωρίς τεχνητό θόρυβο.	54
5.2	F-measure της Crisp-EC και της Prob-EC ανά μέση τιμή κατανομής Γάμμα υπό την υπόθεση «ομαλού θορύβου».	56
5.3	F-measure της Crisp-EC και της Prob-EC ανά μέση τιμή κατανομής Γάμμα υπό την υπόθεση «οξέως θορύβου».	58
5.4	Πιθανότητες αληθών και ψευδών θετικών για ένα πείραμα με την Prob-EC πάνω στο <i>moving</i> , για μέση τιμή της κατανομής Γάμμα ίση με 2.5.	60

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

3.1	Τα κύρια κατηγορήματα της Crisp-EC.	34
5.1	Αληθή Θετικά (True Positives — TP), Ψευδή Θετικά (False Positives — FP), Ψευδή Αρνητικά (False Negatives — FN), Ακρίβεια (Precision), Ανάκληση (Recall) και μέτρο F (F-measure) για την Crisp-EC και την Prob-EC στο CAVIAR, χωρίς τεχνητό θόρυβο.	52

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Περιγραφή του Προβλήματος

Μια διαδεδομένη προσέγγιση στο πρόβλημα της αναγνώρισης ανθρώπινης συμπεριφοράς διαχωρίζει την αναγνώριση «χαμηλού επιπέδου» από την αναγνώριση «υψηλού επιπέδου». Η έξοδος της πρώτης μορφής αναγνώρισης είναι ένα σύνολο από ενέργειες που λαμβάνουν χώρα σε μικρά χρονικά διαστήματα (short-term activities — STA), οι οποίες αποτελούν την είσοδο της αναγνώρισης υψηλού επιπέδου. Η έξοδος της δεύτερης μορφής αναγνώρισης είναι ένα σύνολο από ενέργειες που λαμβάνουν χώρα σε μεγάλα χρονικά διαστήματα (long-term activities — LTA). Στην εργασία αυτή εστιάζουμε στην αναγνώριση «υψηλού επιπέδου».

Ορίζουμε ένα σύνολο από ενδιαφέρουσες LTA, όπως «πάλη» (“fighting”) και «συνάντηση» (“meeting”), ως χρονικούς συνδυασμούς από STA όπως «βάδισμα» (“walking”), «τρέξιμο» (“running”) και «ακινησία» (“inactive”), χρησιμοποιώντας μια υλοποίηση του Λογισμού Γεγονότων (Event Calculus — EC) [22] σε λογικό προγραμματισμό (Prolog). Χρησιμοποιούμε τη γλώσσα EC για να εκφράσουμε τους χρονικούς περιορισμούς επί ενός συνόλου από STA οι οποίοι, εφόσον ικανοποιηθούν, οδηγούν στην αναγνώριση μιας LTA. Το αποτέλεσμα είναι μία διάλεκτος EC βασισμένη σε Prolog, την οποία ονομάζουμε Crisp-EC.

Η αναγνώριση ανθρώπινης συμπεριφοράς γίνεται σχεδόν πάντα υπό την ύπαρξη διαφόρων ειδών θορύβου, οι οποίοι βλάπτουν την ακρίβεια της αναγνώρισης. Σε πρόσφατη εργασία [3] εντοπίσαμε διάφορους τύπους αβεβαιότητας που υπάρχουν στην αναγνώριση γεγονότων, όπως, λόγου χάρη, η εσφαλμένη αναγνώριση κάποιας STA. Η αβεβαιότητα τέτοιας μορφής μπορεί να έχει δραστικά αρνητική επίδραση στην ακρίβεια της Crisp-EC.

1.2 Συνεισφορά

Προκειμένου να αντιμετωπίσουμε το ζήτημα του περιβαλλοντικού θορύβου στην αναγνώριση ανθρώπινης συμπεριφοράς, επεκτείνουμε την υπάρχουσα εργασία μας με την παρουσίαση της Prob-EC, μιας διαλέκτου Λογισμού Γεγονότων κατάλληλης για αναγνώριση ανθρώπινης συμπεριφοράς κάτω από καθεστώς θορύβου. Στο νέο αυτό σύστημα, κάθε δραστηριότητα υψηλού επιπέδου (LTA) που λαμβάνει χώρα μεταξύ δύο ανθρώπων (ή μεταξύ ενός ανθρώπου και ενός αντικειμένου) σε κάποια χρονική στιγμή, αναγνωρίζεται με μια τιμή πιθανότητας, η οποία εκφράζει το βαθμό στον οποίο το σύστημα πιστεύει ότι η εν λόγω συμπεριφορά έχει όντως συντελεστεί. Η Prob-EC χρησιμοποιεί τη γλώσσα πιθανοτικού λογικού προγραμματισμού (probabilistic logic programming) ProbLog [20], η οποία επιτρέπει το συμπερασμό πάνω σε βάσεις γνώσης (knowledge bases) των οποίων τα γεγονότα (facts) συνοδεύονται από πιθανότητες. Σε αντιθεση, η προηγούμενη διάλεκτός μας Λογισμού Γεγονότων (η Crisp-EC) είναι βασισμένη στην Prolog, στην οποία τα γεγονότα παίρνουν τιμή αληθείας (true) ή ψεύδους (false) με πιθανότητα 1. Λόγω της φύσης της ProbLog, η Prob-EC μπορεί να λειτουργήσει σε περιβάλλοντα στα οποία οι εμφανίσεις των STA συνοδεύονται από ένα βαθμό αβεβαιότητας όπως παράγεται, για παράδειγμα, από τους πιθανοτικούς ταξινομητές. Παρουσιάζουμε μια ευρεία πειραματική αξιολόγηση της Prob-EC σε ένα σύνολο δεδομένων σχεδιασμένο για αναγνώριση συμπεριφοράς. Η αξιολόγησή μας επιδεικνύει τις συνθήκες κάτω από τις οποίες η Prob-EC είναι πιο ακριβής από την Crisp-EC.

Δεδομένου του ότι τα συστήματα αναγνώρισης χαμηλού επιπέδου λειτουργούν κατ' εξοχήν μέσα σε περιβάλλον αβεβαιότητας, πιστεύουμε πως η εφαρμοστικότητα του συστήματός μας είναι ευρεία. Η πειραματική μας αξιολόγηση αποδεικνύει ότι, ειδικά κάτω από καθεστώς ισχυρού θορύβου, η Prob-EC είναι σημαντικά πιο ακριβής στην αναγνώριση ανθρώπινης συμπεριφοράς από την Crisp-EC. Σύμφωνα με όσα μέχρι στιγμής γνωρίζουμε, η Prob-EC είναι η πρώτη διάλεκτος Λογισμού Γεγονότων η οποία είναι ικανή να αντιμετωπίσει το θόρυβο στις STA. Η παρούσα εργασία επίσης συνιστά την πρώτη ευρεία πειραματική αξιολόγηση του Λογισμού Γεγονότων μέσα σε περιβάλλον αβεβαιότητας. Ο πλήρης κώδικας της Prob-EC και της Crisp-EC, καθώς και το σύνολο δεδομένων πάνω στο οποίο έχουμε πειραματιστεί, είναι διαθέσιμα στο URL: <http://users.iit.demokritos.gr/~jfilip/files/prob-EC-code.tar.gz>.

1.3 Διάρθρωση εργασίας

Η συνέχεια της εργασίας έχει οργανωθεί σε 5 κεφάλαια.

Στο κεφάλαιο 2 παρουσιάζουμε σχετικές εργασίες πάνω στο Λογισμό Γεγονότων και στη

γλώσσα ProbLog, ενώ συνεχίζουμε με βιβλιογραφία πάνω στην αναγνώριση ανθρώπινης συμπεριφοράς. Στην ανάλυση βιβλιογραφίας ενδιαφερόμαστε πρωτίστως για συστήματα αναγνώρισης συμπεριφοράς τα οποία είναι προσανατολισμένα στην αντιμετώπιση του θορύβου.

Στο κεφάλαιο 3 αναλύουμε την Crisp-EC. Πρώτα εξηγούμε τα αξιώματα Λογισμού Γεγονότων που χρησιμοποιούμε στην παρούσα εργασία. Κατόπιν, παρουσιάζουμε την είσοδο που δέχεται η Crisp-EC κάνοντας λεπτομερή αναφορά στο σύνολο δεδομένων CAVIAR το οποίο και χρησιμοποιούμε στην εν λόγω εργασία. Τέλος, κάνουμε αναφορά στους κανόνες που ορίζουν τις διάφορες LTA της Crisp-EC.

Στο κεφάλαιο 4 αναλύουμε την Prob-EC. Επεξηγούμε τις συντακτικές διαφορές που πρέπει να έχει μια τέτοια διάλεκτος από μια διάλεκτο βασισμένη σε κλασικό λογικό προγραμματισμό (όπως η Crisp-EC) και το πώς αυτές επηρεάζουν τη σημασιολογία των επερωτήσεων. Επίσης, μέσω παραδειγμάτων, δείχνουμε τη διαδικασία συμπερασμού της Prob-EC για δύο LTA διαφορετικής φύσης.

Στο κεφάλαιο 5 παρουσιάζουμε και εξηγούμε τα πειραματικά μας αποτελέσματα. Επειδή το CAVIAR δεν περιέχει πληροφορία για περιβαλλοντικό θόρυβο, εισάγουμε τεχνητό θόρυβο υπό τη μορφή πιθανοτήτων στις δραστηριότητες χαμηλού επιπέδου. Τις πιθανότητες αυτές τις αντλούμε από μια κατανομή πιθανότητας. Χρησιμοποιούμε δύο διαφορετικές μεθόδους εισαγωγής θορύβου («ομαλό» και «οξύ» θόρυβο), και συγκρίνουμε και στις δύο περιπτώσεις την ακρίβεια των Crisp-EC και Prob-EC. Ολοκληρώνουμε το κεφάλαιο συνοψίζοντας τα συμπεράσματα που προέκυψαν από τα πειράματα.

Στο κεφάλαιο 6 συνοψίζουμε την εργασία μας, παρουσιάζουμε τα βασικά συμπεράσματά μας και αναφέρουμε κατευθύνσεις για μελλοντική εργασία.

ΚΕΦΑΛΑΙΟ 2

ΣΧΕΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

2.1 Λογισμός Γεγονότων

Στην ενότητα αυτή θα κάνουμε μια σύντομη αναφορά στο Λογισμό Γεγονότων (Event Calculus). Η Event Calculus, δημιούργημα των Kowalski και Sergot [22], είναι μία τυπική γλώσσα πρώτης τάξης (first-order) που χρησιμοποιείται για την αναπαράσταση και το συμπερασμό συμβάντων (events) και των επιπτώσεών τους. Στην παρούσα εργασία κάνουμε την υπόθεση ότι το μοντέλο του χρόνου είναι γραμμικό και μπορεί να εμπεριέχει πραγματικούς ή ακέραιους αριθμούς. Στην έκφραση $F = V$, το F είναι ένα fluent, δηλαδή μια οντότητα η οποία μπορεί να αλλάζει την τιμή της μέσα στο χρόνο. Η έκφραση $F = V$ υποδεικνύει ότι η τιμή του F είναι V . Τα δυαδικά fluents είναι μια ειδική κατηγορία από fluents στα οποία υπάρχουν μόνο δύο διαφορετικές τιμές: true ή false. Πρακτικά, σε κάποια χρονική στιγμή ισχύει ότι $F = V$ αν το F έχει *εκκινηθεί* στην τιμή V από κάποιο συμβάν σε κάποια προηγούμενη χρονική στιγμή και δεν έχει *τερματιστεί* από κάποιο άλλο event στο μεταξύ. Με τον όρο «εκκίνηση» ενός δυαδικού fluent F αναφέρουμε την παρατήρηση στην είσοδο ενός συνόλου από συμβάντα το οποίο επαρκεί για να ανατεθεί η τιμή true στο fluent F . Για παράδειγμα, η παρατήρηση δύο ανθρώπων που σπρώχνουν ο ένας τον άλλον επαρκεί για να εκχωρηθεί η τιμή true στο fluent *fighting*, που αναπαριστά μια διένεξη μεταξύ δύο ανθρώπων. Κατά αντιστοιχία, η παρατήρηση στην είσοδο ενός συνόλου από συμβάντα τα οποία αναθέτουν τιμή false σε ένα fluent αποτελεί για τη διάλεκτό μας τον «τερματισμό» του εν λόγω fluent.

Σχεδόν όλες οι διάλεκτοι της Event Calculus που υπάρχουν στη βιβλιογραφία εστιάζουν σε κάποιο είδος φορμαλισμού της διαδικασίας ανάθεσης μιας τιμής V σε ένα fluent F . Η ανάθεση αυτή μπορεί να δίνει την τιμή V στο fluent F για μια χρονική στιγμή ή για ένα χρονικό διάστημα.

Στην ενότητα 3.1 εξηγούμε το πώς ακριβώς ορίζουμε αυτή τη διαδικασία στην Crisp-EC.

2.2 ProbLog

Στην ενότητα αυτή κάνουμε μια σύντομη αναφορά στην ProbLog [20], μια πιθανοτική επέκταση της γλώσσας λογικού προγραμματισμού Prolog. Η ProbLog διαφέρει από την Prolog στο ότι επιτρέπει την ύπαρξη πιθανοτικών γεγονότων (probabilistic facts), τα οποία είναι γεγονότα της μορφής $p_i :: f_i$. Στην έκφραση $p_i :: f_i$, το p_i είναι ένας πραγματικός αριθμός εντός του διαστήματος $[0, 1]$ και το f_i είναι ένα γεγονός Prolog. Ο αριθμός p_i εκφράζει την πιθανότητα του να ανήκει το f_i σε έναν τυχαία παραγόμενο (sampled) πιθανό κόσμο (possible world). Αν το $p_i :: f_i$ δεν είναι πλήρως αποτιμημένο (grounded), τότε η πιθανότητα p_i ανατίθεται σε όλες τις πιθανές αποτιμήσεις (groundings) του f_i .

Τα πιθανοτικά γεγονότα σε ένα πρόγραμμα ProbLog αναπαριστούν τυχαίες μεταβλητές. Επιπροσθέτως, η ProbLog εφαρμόζει την υπόθεση ανεξαρτησίας επί αυτών των μεταβλητών. Αυτό σημαίνει ότι ένας κανόνας ο οποίος είναι ορισμένος ως η σύζευξη n τέτοιων πιθανοτικών γεγονότων έχει πιθανότητα ίση με το γινόμενο των πιθανοτήτων αυτών των n γεγονότων. Όταν ένα άτομο εμφανίζεται στην κεφαλή περισσότερων από ενός κανόνα, η πιθανότητά του υπολογίζεται ως η πιθανότητα της υπονοούμενης διάζευξης μεταξύ των κανόνων. Για παράδειγμα, για ένα κατηγορημα p με δυο κανόνες $p \leftarrow l_1$ και $p \leftarrow l_2, l_3$, η πιθανότητα $P(p)$ υπολογίζεται ως εξής:

$$\begin{aligned} P(p) &= P((p \leftarrow l_1) \vee (p \leftarrow l_2, l_3)) = \\ &= P(p \leftarrow l_1) + P(p \leftarrow l_2, l_3) - P((p \leftarrow l_1) \wedge (p \leftarrow l_2, l_3)) = \\ &= P(l_1) + P(l_2) \times P(l_3) - P(l_1) \times P(l_2) \times P(l_3) \end{aligned}$$

Δεδομένης της υπόθεσης ανεξαρτησίας, οποιοδήποτε υποπρόγραμμα L της Βάσης Herbrand του προγράμματος έχει πιθανότητα ίση με:

$$P(L) = \prod_{f_i \in L} p_i \cdot \prod_{f_i \notin L} (1 - p_i) \quad (2.1)$$

Με χρήση της εξίσωσης (2.1), μια επερώτηση (query) q έπεται λογικά από ένα πρόγραμμα ProbLog (success probability) με πιθανότητα:

$$P_s(q) = \sum_{L \models q} P(L) \quad (2.2)$$

Δηλαδή, για να υπολογίσουμε την πιθανότητα με την οποία ένα επερώτημα έπεται λογικά από

τη βάση γνώσης μας, αρκεί να υπολογίσουμε το άθροισμα των πιθανοτήτων όλων των υποπρογραμμάτων τα οποία επαρκούν για να αποδειχθεί το επερώτημα. Ο υπολογισμός όμως αυτός είναι υπολογιστικά δύσκολος, καθώς εμπερικλύει ένα άθροισμα με εκθετικά πολλούς όρους ($2^{|B_L|}$, όπου B_L είναι η βάση Herbrand). Ξεκινώντας από την εξίσωση (2.2) και απαλοΐφοντας επαναλαμβανόμενους και περιττούς όρους, καταλήγουμε στον εναλλακτικό ορισμό:

$$P_s(q) = P\left(\bigvee_{e \in \text{Proofs}(q)} \bigwedge_{f_i \in e} f_i\right) \quad (2.3)$$

Δηλαδή το ερώτημα του με ποιά πιθανότητα έπεται λογικά ένα επερώτημα q από μια βάση γνώσης ProbLog έχει μετατραπεί στον υπολογισμό της πιθανότητας του τύπου (2.3), ο οποίος είναι ένας τύπος σε Διαζευκτική Κανονική Μορφή (Disjunctive Normal Form — DNF). Πρακτικά, η εξίσωση (2.3) εκφράζει το γεγονός ότι η πιθανότητα του q είναι ίση με την πιθανότητα του να παραχθεί τουλάχιστον μία από τις αποδείξεις του.

Δυστυχώς, ο υπολογισμός της εξίσωσης (2.3) δεν είναι ευθέως μετατρέψιμος σε ένα άθροισμα γινομένων. Κάθε σύζευξη (απόδειξη) $\bigwedge_{f_i \in e} f_i$ στην εξίσωση (2.3) εκφράζει την πιθανότητα ενός πιθανού κόσμου ο οποίος έπεται λογικά αυτή την απόδειξη. Υπάρχουν πολλοί τέτοιοι κόσμοι, συμπεριλαμβανομένων κόσμων οι οποίοι έπονται και άλλες αποδείξεις του q . Συνεπώς, εάν μεταφράζαμε την εξίσωση (2.3) σε ένα άθροισμα γινομένων, θα υποθέταμε ότι όλες οι διαφορετικές αποδείξεις είναι ανεξάρτητες μεταξύ τους (δηλαδή θα εκφράζον κόσμους οι οποίοι έπονται λογικά ακριβώς μία απόδειξη ο καθένας, και δεν υπάρχει ζεύγος κόσμων που έπονται την ίδια απόδειξη), κάτι που δεν ισχύει σχεδόν ποτέ στην πράξη. Προκειμένου λοιπόν να γίνουν οι αποδείξεις ανεξάρτητες μεταξύ τους, πρέπει σε κάθε σύζευξη της (2.3) να προστεθούν αρνητικά λεκτικά (literals), έτσι ώστε ο υπολογισμός της πιθανότητας μιας απόδειξης να μην περιλαμβάνει κόσμους οι οποίοι έχουν ήδη καλυφθεί από προηγούμενες αποδείξεις του τύπου (2.3). Αυτό το πρόβλημα είναι γνωστό στη βιβλιογραφία ως *disjoint-sum problem* και έχει αποδειχθεί #P-hard [41].

Η προσέγγιση της ProbLog στο συγκεκριμένο πρόβλημα στηρίζεται στη χρήση Δυναδικών Διαγραμμάτων Απόφασης (Binary Decision Diagrams — BDD) [8] για την αναπαράσταση του τύπου (2.3). Ένα BDD είναι μια συμπαγής μορφή δυαδικού δένδρου απόφασης (Binary Decision Tree) στο οποίο έχουμε αφαιρέσει περιττούς κόμβους και συμπύζει ισόμορφα υποδένδρα. Οι κόμβοι του BDD αναπαριστούν τα πιθανοτικά γεγονότα του προγράμματος ProbLog. Κάθε κόμβος έχει μια «θετική» και μια «αρνητική» εξερχόμενη ακμή, οι οποίες οδηγούν είτε σε έναν κόμβο-παιδί είτε σε κάποιον από τους τερματικούς κόμβους “true” και “false”. Η θετική εξερχόμενη ακμή ενός κόμβου είναι επισημειωμένη με την πιθανότητα του αντίστοιχου πιθανοτικού

γεγονότος και η αρνητική είναι επισημειωμένη με το συμπλήρωμα αυτής της πιθανότητας. Οι θετικές και αρνητικές ακμές αναπαριστούν τις αντίστοιχες αποφάσεις σχετικά με το αν ο κόσμος που αναπαριστάται από το παρόν δείγμα (sample) περιλαμβάνει το σχετικό γεγονός ή όχι. Αυτό συμβαίνει με την αντίστοιχη πιθανότητα. Κατ' επέκταση, αφού ένα γεγονός με πιθανότητα 0.9 θα περιλαμβάνεται στους 9 από τους 10 κόσμους που γίνονται sampled, είναι πολύ πιο πιθανό να ακολουθηθεί η «θετική» εξερχόμενη ακμή του σχετικού κόμβου του BDD παρά η αρνητική.

Λόγω της φύσης των BDD, ένα οποιοδήποτε μονοπάτι από τον κόμβο - ρίζα μέχρι τον τερματικό κόμβο "true" αναπαριστά μια σύζευξη του DNF τύπου (2.3). Οι αρνητικές εξερχόμενες ακμές είναι μια συμπαγής αναπαράσταση των αρνητικών λεκτικών τα οποία είναι απαραίτητα για να κάνουμε τον τύπο (2.3) να αναπαριστά μια διάζευξη ανεξάρτητων μεταξύ τους συζεύξεων.

Ο συμπερασμός της ProbLog μπορεί να συμπυκνωθεί σε τρία γενικά βήματα. Το πρώτο βήμα είναι η συγκέντρωση όλων των αποδείξεων του επερωτήματος q . Αυτό συμβαίνει με τη σάρωση του Επιλεκτικού Γραμμικού Οριστικού (Selective Linear Definite — SLD) δένδρου των αποδείξεων του q και την αναπαραστασή του στη μορφή του τύπου (2.3). Στο δεύτερο βήμα, ο τύπος (2.3) μεταφράζεται σε ένα BDD. Στο τρίτο και τελευταίο βήμα, η πιθανότητα αυτού του BDD υπολογίζεται αναδρομικά, ξεκινώντας από τον κόμβο - ρίζα και υποθέτοντας μια πιθανότητα 1 για τον «αληθή» τερματικό κόμβο και 0 για τον «ψευδή» τερματικό κόμβο.

Με τη χρήση των BDD, ο συμπερασμός της ProbLog μπορεί να εφαρμοστεί σε επερωτήματα με χιλιάδες διαφορετικές αποδείξεις [20, κεφ. 5]. Η αποδοτικότητα της ProbLog, καθώς και η απλή προσαρμογή της Crisp-EC σε αυτήν, υπήρξαν οι κύριοι λόγοι για τους οποίους επιλέξαμε την εν λόγω γλώσσα για τη μοντελοποίηση της Prob-EC.

Ένα ακόμη στοιχείο ενδιαφέροντος για την ProbLog είναι η προσέγγιση της γλώσσας στο ζήτημα της άρνησης. Η «άρνηση ως αποτυχία» (negation as failure) [9], η οποία εκφράζεται στην Prolog με το κατηγορημα `not`, έχει παρόμοια σημασιολογία στην ProbLog, η οποία — όπως και η Prolog — κάνει την Υπόθεση Κλειστού Κόσμου (Closed World Assumption). Αν ένα επερωτήμα q είναι μη αποδείξιμο στο λογικό κομμάτι της βάσης γνώσης (δηλαδή, τους κανόνες και τα γεγονότα χωρίς τις πιθανότητές τους), τότε το επερωτήμα `not q` θα είχε πιθανότητα ίση με 1. Αν, όμως, το q είναι αποδείξιμο, ακόμα και με μια πολύ μικρή πιθανότητα, λόγου χάρη 0.01, τότε η πιθανότητα του `not q` θα είναι ίση με 0. Η «άρνηση ως αποτυχία» δεν επηρεάζεται από τις πιθανότητες της βάσης γνώσης: ο μόνος ρόλος της είναι να αποφανθεί αν το επερωτήμα είναι αποδείξιμο στο λογικό μέρος του προγράμματος (δηλαδή τους κανόνες και τα γεγονότα χωρίς τις αντίστοιχες πιθανότητες).

Προκειμένου να υπολογίσει το συμπλήρωμα της πιθανότητας ενός πιθανοτικού γεγονότος, η ProbLog περιλαμβάνει το κατηγορημα `problog_not`. Για οποιοδήποτε πιθανοτικό γεγονός $p_i :: f_i$,

ισχύει:

$$P_s(\text{problog_not}(f_i)) = 1 - P_s(f_i) = 1 - p_i \quad (2.4)$$

Το κατηγορημα `problog_neg` είναι μια επέκταση του `problog_not` που μπορεί να εφαρμοστεί τόσο σε πιθανοτικά γεγονότα όσο και σε κανόνες. Για έναν κανόνα r ισχύει:

$$P_s(\text{problog_neg}(r)) = 1 - P_s(r) \quad (2.5)$$

Τέλος, τα “cuts” (!/0) πρέπει να χρησιμοποιούνται με φειδώ από τον προγραμματιστή ProbLog. Αυτό συμβαίνει γιατί η φύση των cuts είναι να «κλαδεύουν» το SLD-δένδρο του επερωτήματος, συνήθως για λόγους αποδοτικότητας. Αυτό όμως σημαίνει ότι η ProbLog δε θα μπορέσει να συλλέξει όλες τις αποδείξεις του επερωτήματος και κατ’ επέκταση ο τύπος (2.3) θα αναπαριστά ένα υποσύνολο του συνόλου των αποδείξεων.

Η ProbLog έχει ενσωματωθεί πλήρως στο σύστημα Prolog YAP.¹ Πρόσθετες λεπτομέρειες υλοποίησης, παραδείγματα και δείγματα κώδικα είναι διαθέσιμα στο [20] και στην ιστοσελίδα της ProbLog.²

2.3 Σχετικές εργασίες σε αναγνώριση δραστηριοτήτων

Η βιβλιογραφία βρίθει συστημάτων αναγνώρισης γεγονότων και δραστηριοτήτων (event / activity recognition). Στο υποκεφάλαιο αυτό εστιάζουμε σε συστήματα αναγνώρισης δραστηριοτήτων υψηλού επιπέδου (LTA recognition systems), τα οποία, όπως και το δικό μας, παρουσιάζουν μια τυπική και δηλωτική σημασιολογία. Ιδιαίτερου ενδιαφέροντος για εμάς είναι συστήματα τα οποία εμπεριέχουν κάποια μέθοδο αντιμετώπισης θορύβου.

Ένα μεγάλο μέρος των συστημάτων αναγνώρισης είναι βασισμένα στη λογική. Μερικές αξιοσημείωτες προσεγγίσεις είναι το Chronicle Recognition System [11], η διάλεκτος Λογισμού Γεγονότων των Paschke et al. [29, 28] καθώς και η ιεραρχική αναπαράσταση γεγονότων του [15]. Τρεις πρόσφατες περιλήψεις λογικών συστημάτων αναγνώρισης μπορούν να βρεθούν στα [10, 30, 2]. Τα συστήματα αυτά έχουν το κοινό στοιχείο ότι χρησιμοποιούν λογικές μεθόδους για την αναπαράσταση και το συμπερασμό, αλλά δεν είναι ικανά να αντιμετωπίσουν θόρυβο.

Οι Shet et al. [35, 36] έχουν παρουσιάσει ένα σύστημα αναγνώρισης βασισμένο σε λογικό

¹<http://www.dcc.fc.up.pt/~vsc/Yap/>

²<http://dtai.cs.kuleuven.be/problog/>

προγραμματισμό το οποίο επιχειρεί να αντιμετωπίσει το πρόβλημα των δεδομένων που προέρχονται από αναξιόπιστους αισθητήρες. Στην εργασία τους έχουν ορίσει LTA σχετικές με κλοπές, διαρρήξεις, απωλεσθέντα αντικείμενα και ούτω καθ' εξής. Οι συγγραφείς έχουν ενσωματώσει στο σύστημά τους μηχανισμούς για συμπερασμό πάνω σε αβέβαιους κανόνες και γεγονότα. Η αβεβαιότητα στους κανόνες εκφράζει ένα μέτρο της αξιοπιστίας των κανόνων, αποτελεί δηλαδή ένα δείγμα του κατά πόσο ο συμπερασμός που παράγεται από κάποιον κανόνα οφείλει να ληφθεί υπ' όψιν. Από την άλλη, η αβεβαιότητα στα γεγονότα εκφράζει τις πιθανότητες ορθής αναγνώρισης των εκάστοτε δραστηριοτήτων χαμηλού επιπέδου (STA) και είναι το ίδιο πρόβλημα που αντιμετωπίζουμε στην παρούσα εργασία. Στο σύστημα VidMAP [35], δεδομένα που έχουν κατηγοριοποιηθεί λανθασμένα από ένα σύστημα επεξεργασίας εικόνας — για παράδειγμα, ένα δέντρο το οποίο το σύστημα χαμηλότερου επιπέδου θεώρησε άνθρωπο — «απορρίπτονται» αυτόματα. Προκειμένου το σύστημα να μπορεί να αποφαινεται για τέτοιες λανθασμένες κατηγοριοποιήσεις, οι συγγραφείς αναφέρουν [35, σελ. 2]: «... αυτό το επιτυγχάνει με το να παρατηρεί αν το εν λόγω αντικείμενο έχει αναγνωριστεί πολλάκις μέσα στο πρόσφατο παρελθόν». Στο [36] χρησιμοποιείται μια αλγεβρική δομή γνωστή ως «δίπλεγμα» (bilattice) [13] για να αναγνωρισθούν ανθρώπινες οντότητες με βάση την αβέβαιη έξοδο αισθητήρων (καμερών) ειδικευμένων στην αναγνώριση μελών σώματος, όπως για παράδειγμα αισθητήρων χειρών ή ποδιών. Η δομή του διπλέγματος αντιστοιχεί σε κάθε λογικό άτομο (atom) δύο τιμές αβεβαιότητας, εκ των οποίων η μία κωδικοποιεί την ποσότητα της πληροφορίας που υποστηρίζει την υπόθεση ότι το εν λόγω άτομο είναι αληθές, ενώ η άλλη κωδικοποιεί την αξιοπιστία της εν λόγω πληροφορίας. Όσο πιο πολλή αξιόπιστη πληροφορία παρέχεται, τόσο πιο πιθανό είναι το άτομο να είναι αληθές.

Για την αναγνώριση δραστηριοτήτων υπό καθεστώς αβεβαιότητας, ιδιαίτερα δημοφιλή έχουν υπάρξει τα πιθανοτικά γραφικά μοντέλα (probabilistic graphical models). Η αναγνώριση δραστηριοτήτων απαιτεί την επεξεργασία χρονοσειρών από STA, επομένως πολλές σχετικές εφαρμογές είναι βασισμένες σε σειριακές παραλλαγές τέτοιων μοντέλων, όπως είναι, λόγου χάρη, τα Κρυφά Μοντέλα Μαρκόφ (Hidden Markov Models) [31], τα Δυναμικά Μπαεσιανά Δίκτυα (Dynamic Bayesian Networks) [26] και τα Τυχαία Υπό Συνθήκη Πεδία (Conditional Random Fields) [23]. Συγκρινόμενα με τις λογικές μεθόδους, τα γραφικά μοντέλα μπορούν να διαχειριστούν με φυσικό τρόπο την αβεβαιότητα του περιβάλλοντος, η δομή τους όμως περιορίζει ιδιαίτερα τις δυνατότητες αναπαράστασής τους. Προκειμένου ένα γραφικό μοντέλο να μοντελοποιήσει LTA για τον ορισμό των οποίων απαιτούνται πολυάριθμοι συσχετισμοί μεταξύ οντοτήτων, όπως για παράδειγμα είναι οι φυσικές αλληλεπιδράσεις μεταξύ ανθρώπων και αντικειμένων, η δομή του μπορεί να γίνει απαγορευτικά ογκώδης και πολύπλοκη. Για να υπερκεράσουν αυτούς τους περιορισμούς, τα εν λόγω μοντέλα έχουν επεκταθεί για να υποστηρίξουν πιο σύνθετες σχέσεις.

Μερικά παραδείγματα τέτοιων επεκτάσεων είναι η δυνατότητα αναπαράστασης αλληλεπιδράσεων μεταξύ πολλαπλών αντικειμένων [5, 14, 43, 40], η αναπαράσταση μακροπρόθεσμων περιορισμών μεταξύ των εσωτερικών καταστάσεων (states) του μοντέλου [18], καθώς και η περιγραφή ιεραρχικών δομών δραστηριοτήτων [27, 24]. Παρά τις επεκτάσεις, η έλλειψη μιας τυπικής γλώσσας αναπαράστασης κάνει τον ορισμό σύνθετων LTA περίπλοκο και την ενσωμάτωση προϋπάρχουσας γνώσης (domain background knowledge) δύσκολη.

Τα Λογικά Δίκτυα Μαρκόφ (Markov Logic Networks — MLN) [32] είναι άλλος ένας δημοφιλής φορμαλισμός για την αναπαράσταση της αβεβαιότητας στην αναγνώριση δραστηριοτήτων. Τα MLN χρησιμοποιούν μια αναπαράσταση λογικής πρώτης τάξης, στην οποία κάθε κανόνας σχετίζεται με ένα βάρος που εκφράζει το πόσο αξιόπιστος θεωρείται ο εν λόγω κανόνας. Η βάση γνώσης των εν λόγω κανόνων μεταφράζεται σε ένα δίκτυο Μαρκόφ, στο οποίο και λαμβάνει χώρα ο συμπερασμός. Στην εργασία αυτή [4], τα MLN χρησιμοποιούνται για να αναγνωριστούν κάποιες LTA, δεδομένων των αβέβαιων STA που έχουν ανιχνευτεί από ταξινομητές χαμηλότερου επιπέδου. Μία πιο εκφραστική προσέγγιση, η οποία μπορεί να αναπαραστήσει LTA που διατηρούνται αναλλοίωτες στο χρόνο καθώς και LTA που συμβαίνουν ταυτόχρονα, μαζί με τα πρώτα και τελευταία χρονικά σημεία στα οποία συμβαίνουν, έχει προταθεί στο [16]. Η μέθοδος του [33] χρησιμοποιεί υβριδικά MLN (hybrid-MLN) [42] προκειμένου να αναγνωρίσει επιτυχείς και ανεπιτυχείς αλληλεπιδράσεις μεταξύ ανθρώπων, έχοντας ως είσοδο θορυβώδη δεδομένα σχετικά με τη θέση των ανθρώπων. Σε σχέση με τις μεθόδους βασισμένες στα παραδοσιακά MLN, η βάση γνώσης του [33] συντίθεται από ορισμούς (κανόνες) LTA. Επιπροσθέτως, στη μέθοδο αυτή περιλαμβάνονται υβριδικοί κανόνες, οι οποίοι έχουν ως στόχο την αφαίρεση του θορύβου από τα δεδομένα εισόδου. Οι υβριδικοί αυτοί κανόνες ορίζονται ως κλασικοί κανόνες MLN, όμως το κάθε βάρος σχετίζεται με μια πραγματική συνάρτηση, όπως είναι λόγου χάρη η απόσταση μεταξύ δύο διαφορετικών ανθρώπων. Κατ' επέκταση, το μέτρο αξιοπιστίας ενός κανόνα εξαρτάται από το ίδιο το βάρος του, αλλά και από τη σχετιζόμενη με αυτό πραγματική συνάρτηση. Όλες αυτές οι μέθοδοι έχουν το κοινό στοιχείο ότι χρησιμοποιούν την αναπαράσταση της λογικής πρώτης τάξης, η οποία έχει αποδειχθεί στην πράξη ιδιαίτερα εκφραστική καθώς και εύληπτη από μη ειδικούς. Παρ' ολ' αυτά, οι προτεινόμενοι ορισμοί LTA έχουν περιορισμένη χρονική αναπαράσταση. Για παραδειγμα, ο μόνος τρόπος να κωδικοποιηθούν χρονικοί περιορισμοί είναι σε διαδοχικά χρονικά σημεία.

Μια μέθοδος που χρησιμοποιεί χρονικούς συσχετισμούς μεταξύ χρονικών διαστημάτων (αντί σημείων) περιγράφεται στο [25]. Στόχος αυτής της μεθόδου είναι να καθορίσει με όσο το δυνατόν μεγαλύτερη ακρίβεια την πιο συνεπή με τους περιορισμούς αλληλουχία από LTA, βασισμένη στις παρατηρήσεις ταξινομητών χαμηλού επιπέδου. Η εργασία αυτή παρουσιάζει κάποιες

ομοιότητες με τα [39, 19]. Ένα κοινό της στοιχείο με αυτές τις εργασίες είναι ότι η μέθοδος χρησιμοποιεί MLN για να εκφράσει LTA με κανόνες βασισμένους σε «κοινή λογική» (common sense reasoning). Σε αντίθεση όμως με αυτές και προκειμένου να υποστηρίξει συμπερασμό πάνω σε χρονικά διαστήματα, χρησιμοποιεί χρονικές σχέσεις βασισμένες στην Άλγεβρα Διαστημάτων του Allen (Allen's Interval Algebra — IA) [1]. Προκειμένου να αποφευχθεί η εκθετική αύξηση των δυνατών διαστημάτων που συνήθως προκαλείται από την IA, μια μέθοδος «από κάτω προς τα πάνω» (bottom-up) διαγράφει τις λιγότερο πιθανές LTA. Στα [6, 34], παρουσιάζεται μια πιθανοτική επέκταση της Λογικής Γεγονότων (Event Logic) [37] που έχει ως στόχο να πραγματοποιήσει αναγνώριση δραστηριοτήτων σε χρονικά διαστήματα. Παρομοίως με τα MLN, η συγκεκριμένη μέθοδος ξεκινά από ένα σύνολο κανόνων LTA με βάρη για να ορίσει ένα πιθανοτικό μοντέλο. Ο σημαντικότερος περιορισμός της Λογικής Γεγονότων είναι ότι αποφεύγει την απαρίθμηση όλων των πιθανών σχέσεων μεταξύ των διαστημάτων.

Η κυριότερη διαφορά της δικής μας προσέγγισης σε σχέση με τις προαναφερθείσες είναι ότι χρησιμοποιούμε το Λογισμό Γεγονότων για χρονική αναπαράσταση και συμπερασμό. Ο Λογισμός Γεγονότων ορίζει γενικούς κανόνες (αξιώματα) για την αναπαράσταση περίπλοκων χρονικών συσχετίσεων — συμπεριλαμβανομένου του φορμαλισμού της αδράνειας — οι οποίοι βοηθούν σε μεγάλο βαθμό το σχεδιαστή του συστήματος να ορίσει δραστηριότητες υψηλού επιπέδου. Με τη χρήση του Λογισμού Γεγονότων, είναι εύκολο για κάποιον να σχεδιάσει διαισθητικούς και ακριβείς ορισμούς δραστηριοτήτων και κατ' αυτόν τον τρόπο διευκολύνεται η επικοινωνία μεταξύ σχεδιαστή δραστηριοτήτων (activity definition developer) και ειδικού του πεδίου (domain expert), ενώ υποβοηθείται και η συντήρηση του κώδικα. Επιπροσθέτως, δεδομένου του ότι η διάλεκτός μας (Crisp-EC) είναι βασισμένη σε λογικό προγραμματισμό, η μετάφρασή της σε ένα περιβάλλον πιθανοτικού λογικού προγραμματισμού όπως η ProbLog είναι μια απλή διαδικασία συντακτικών αλλαγών, η οποία κάλλιστα μπορεί να αυτοματοποιηθεί μέσω ενός λεκτικού αναλυτή (parser).

Η δουλειά μας συμπληρώνεται επίσης από την πιθανοτική διάλεκτο Λογισμού Γεγονότων που προτείνεται στο [38], η οποία είναι βασισμένη σε MLN. Η διάλεκτος αυτή (DEC-MLN) και η Prob-EC αντιμετωπίζουν διαφορετικά είδη αβεβαιότητας. Η Prob-EC μπορεί να πραγματοποιήσει συμπερασμό δεδομένου του θορύβου στην είσοδο, που εκφράζεται μέσω πιθανοτήτων αναγνώρισης (detection probabilities) των STA. Η DEC-MLN, από την άλλη, εστιάζει στο ότι οι ορισμοί δραστηριοτήτων υψηλού επιπέδου (LTA) ενδέχεται να είναι ατελείς και αυτή η αβεβαιότητα εκφράζεται μέσω βαρών στους ορισμούς αυτούς.

Η ProbLog και τα MLN σχετίζονται σε μεγάλο βαθμό. Μια ειδοποιός διαφορά τους όμως είναι ότι τα MLN, ως επέκταση της κλασικής λογικής πρώτης τάξης, δεν ακολουθούν την υ-

πόθεση κλειστού κόσμου. Στη βιβλιογραφία υπάρχουν κάποιες εργασίες οι οποίες εξερευνούν τη σχέση μεταξύ των δύο αυτών φορμαλισμών. Στο [7], επί παραδείγματι, παρουσιάζεται μια επέκταση της ProbLog η οποία μπορεί να διαχειριστεί τύπους λογικής πρώτης τάξης οι οποίοι συνοδεύονται από περιορισμούς με βάρη. Αυτό έρχεται σε αντίθεση με την κλασική ProbLog, η οποία — όπως και η Prolog — περιορίζεται σε προτάσεις Horn για την αναπαράσταση κανόνων. Στο [12], οι συγγραφείς παρουσιάζουν μια μετατροπή προγραμμάτων ProbLog σε αποτιμημένα (grounded) MLN και κατόπιν χρησιμοποιούν σύγχρονους αλγορίθμους για συμπερασμό σε MLN (για παράδειγμα, Monte Carlo Satisfiability — MC-SAT) πάνω στο αποτιμημένο δίκτυο.

ΚΕΦΑΛΑΙΟ 3

ΛΟΓΙΣΜΟΣ ΓΕΓΟΝΟΤΩΝ ΓΙΑ ΑΝΑΓΝΩΡΙΣΗ ΑΝΘΡΩΠΙΝΗΣ ΣΥΜΠΕΡΙΦΟΡΑΣ

Στο κεφάλαιο αυτό θα αναλύσουμε τη διάλεκτο Λογισμού Γεγονότων που χρησιμοποιούμε για να αναγνωρίσουμε ανθρώπινη συμπεριφορά (την Crisp-EC) σε τρία σκέλη. Πρώτον, θα κάνουμε λόγο για το πώς ορίζουμε τα αξιώματα του Λογισμού Γεγονότων στην Prolog ώστε να καταλήξουμε στη λεγόμενη «μηχανή συμπερασμού» (inference engine) της Crisp-EC. Κατόπιν, θα αναλύσουμε την είσοδο που παίρνει η Crisp-EC στην εργασία μας, εξηγώντας τα επιμέρους στοιχεία του συνόλου δεδομένων CAVIAR. Ουσιαστικά, στο σημείο αυτό θα παρουσιάσουμε τις δραστηριότητες χαμηλού επιπέδου (STA) του συστήματος. Έχοντας αναλύσει την είσοδό μας, κλείνουμε το κεφάλαιο εξηγώντας το πώς η ανίχνευση (tracking) των στοιχείων της εισόδου οδηγεί, μέσω κανόνων Horn, στην αναγνώριση δραστηριοτήτων υψηλού επιπέδου (LTA).

3.1 Αξιώματα της Crisp-EC

Όπως εξηγήσαμε στην Εισαγωγή, σχεδόν όλες οι διάλεκτοι Λογισμού Γεγονότων που έχουν προταθεί στη βιβλιογραφία εστιάζουν σε κάποιο μηχανισμό απόδοσης μιας τιμής V σε ένα fluent F . Στο υποκεφάλαιο αυτό εξηγούμε πώς μοντελοποιούμε αυτή τη διαδικασία στα πλαίσια της διαλέκτου Crisp-EC, χρησιμοποιώντας κατηγορήματα της γλώσσας Prolog.

Μια *περιγραφή γεγονότος* στην Crisp-EC περιλαμβάνει κανόνες οι οποίοι ορίζουν, μεταξύ άλλων, τις απαιτούμενες εμφανίσεις στιγμιαίων συμβάντων (με τη χρήση του κατηγορήματος

Κατηγορήμα	Σημασιολογία
$\text{happensAt}(E, T)$	Το γεγονός E λαμβάνει χώρα τη χρονική στιγμή T .
$\text{initially}(F = V)$	Η τιμή του fluent F είναι V τη χρονική στιγμή 0.
$\text{holdsAt}(F = V, T)$	Η τιμή του fluent F είναι V τη χρονική στιγμή T .
$\text{initiatedAt}(F = V, T)$	Τη χρονική στιγμή T ξεκινά μια περίοδος χρόνου για την οποία ισχύει $F = V$.
$\text{terminatedAt}(F = V, T)$	Τη χρονική στιγμή T ολοκληρώνεται μια περίοδος χρόνου για την οποία ισχύει $F = V$.

Πίνακας 3.1: Τα κύρια κατηγορήματα της Crisp-EC.

happensAt), τις επιπτώσεις αυτών των συμβάντων (με τη χρήση των κατηγορημάτων initiatedAt και terminatedAt) και τις τιμές των fluents (με τη χρήση των κατηγορημάτων initially και holdsAt). Ο Πίνακας 3.1 συνοψίζει τα κύρια κατηγορήματα της Crisp-EC. Οι μεταβλητές, οι οποίες και ξεκινούν με κεφαλαίο γράμμα, θεωρούνται καθολικά ποσοτικοποιημένες, εκτός και αν κατι τέτοιο αναφέρεται ρητά πως δεν ισχύει. Τα κατηγορήματα, τα συναρτησιακά σύμβολα και οι σταθερές ξεκινούν με πεζό γράμμα.

Οι γενικοί (domain-independent) κανόνες που ορίζουν το κατηγορήμα holdsAt είναι οι εξής:

$$\begin{aligned} \text{holdsAt}(F = V, T) \leftarrow & \\ & \text{initiatedAt}(F = V, T_s), \\ & T_s < T, \end{aligned} \tag{3.1}$$

$$\begin{aligned} \text{broken}(F = V, T_s, T) \leftarrow & \\ & \text{terminatedAt}(F = V, T_f), \\ & T_s < T_f < T \end{aligned} \tag{3.2}$$

$$\begin{aligned} \text{broken}(F = V_1, T_s, T) \leftarrow & \\ & \text{initiatedAt}(F = V_2, T_f), \\ & V_1 \neq V_2, \\ & T_s < T_f < T \end{aligned} \tag{3.3}$$

Η χρήση της «άρνησης ως αποτυχία» στον κανόνα (3.1) έχει ως στόχο να υλοποιήσει μια αντοχή στο χρόνο — «αδράνεια» — των fluents. Σύμφωνα με τον εν λόγω κανόνα, ισχύει (holds) ότι $F = V$ τη χρονική στιγμή T αν το fluent F έχει λάβει την τιμή V σε μια προηγούμενη χρονική στιγμή T_s , και δεν έχει έκτοτε «σπάσει» (broken). Σύμφωνα με τον κανόνα (3.2), μια περίοδος χρόνου

για την οποία ισχύει $F = V$ «σπάει» στη χρονική στιγμή T_f αν η ισότητα $F = V$ τερματίζεται τη χρονική στιγμή T_f . Ο κανόνας (3.3) ορίζει ότι αν τη χρονική στιγμή T_f το fluent F λάβει την τιμή V_2 , τότε ταυτόχρονα τερματίζεται η χρονική περίοδος για την οποία ισχύει $F = V_1$. Κατ' επέκταση, ο εν λόγω κανόνας δεν επιτρέπει σε ένα fluent να έχει δύο διαφορετικές τιμές την ίδια χρονική στιγμή. Πρέπει να σημειωθεί ότι επιτρέπεται σε ένα fluent να μην έχει τιμή σε κάποια χρονική στιγμή. Στην Crisp-EC υπάρχει διαφορά μεταξύ της ανάθεσης $F = \text{false}$ και του τερματισμού του $F = \text{true}$: το πρώτο υπονοεί το δεύτερο, όχι όμως και αντιστρόφως.

Σύμφωνα με τους κανόνες (3.1)-(3.3), τη χρονική στιγμή στην οποία ανατίθεται στο F η τιμή V , το F δεν έχει ακόμα τιμή V . Η ανάθεση της τιμής θεωρείται ότι ισχύει από την επόμενη χρονική στιγμή. Αντιστοίχως, το F διατηρεί την τιμή του τη χρονική στιγμή στην οποία τερματίζεται.

Πέραν του γενικού αξιώματος $\text{initiatedAt}(F = V, 0) \leftarrow \text{initially}(F = V)$, οι ορισμοί των κατηγορημάτων initiatedAt και terminatedAt εξαρτώνται από την εφαρμογή. Μια τυπική μορφή κανόνα που ορίζεται μέσω του initiatedAt είναι η εξής:

$$\begin{aligned} \text{initiatedAt}(F = V, T) \leftarrow \\ \text{happensAt}(E, T), \\ \text{Conditions}[T] \end{aligned} \tag{3.4}$$

όπου η έκφραση $\text{Conditions}[T]$ αναφέρεται σε ένα επιπλέον σύνολο από συνθήκες (conditions) που αφορούν στη χρονική στιγμή T . Παρομοίως εκφράζουμε τους κανόνες που χρησιμοποιούν το κατηγορημα terminatedAt . Στην ενότητα 3.3 παρουσιάζουμε τη χρήση των initiatedAt και terminatedAt για τον ορισμό των LTA που αναγνωρίζει το σύστημά μας.

3.2 Το σύνολο δεδομένων CAVIAR

Στην εργασία αυτή χρησιμοποιούμε το πρώτο σύνολο δεδομένων του ερευνητικού προγράμματος CAVIAR¹ για την αναγνώριση LTA. Το σύνολο δεδομένων αυτό περιέχει 28 βίντεο επιτήρησης ενός στεγασμένου δημόσιου χώρου. Τα βίντεο είναι σκηνοθετημένα: ηθοποιοί περπατούν, κάθονται, συναντιούνται, αφήνουν αντικείμενα, καυγαδίζουν και ούτω καθ' εξής. Κάθε βίντεο έχει επισημειωθεί από την υπεύθυνη ομάδα του CAVIAR σε επίπεδο STA και LTA. Δεδομένου αυτού του συνόλου δεδομένων, η είσοδος που παίρνει το σύστημα αναγνώρισής μας είναι:

- (i) Οι STA «βάδισμα» (*walking*), «τρέξιμο» (*running*), «κίνηση» (*active* — εδώ υπονοείται κίνηση

¹<http://homepages.inf.ed.ac.uk/rbf/CAVIAR/>

μελών σώματος χωρίς αλλαγή της θέσης του ανθρώπου) και «ακινήσια» (*inactive*), μαζί με τις χρονικές στιγμές στις οποίες συμβαίνουν, δηλαδή το σχετικό «καρέ» (*frame*) του βίντεο. Οι δραστηριότητες αυτές είναι αμοιβαίως αποκλειόμενες (δηλαδή, ένας άνθρωπος σε κάποια χρονική στιγμή πραγματοποιεί ακριβώς μία από αυτές) και αναπαριστώνται με το κατηγορήμα *happensAt*. Επί παραδειγματι, *happensAt(active(id₆), 15560)* σημαίνει ότι το άτομο που αντιστοιχεί στον *id₆* παρουσίασε «κίνηση» τη χρονική στιγμή *15560*. Η περιγραφή των STA ως στιγμιαίων γεγονότων αποσκοπεί στη χρήση των κανόνων *initiatedAt* και *terminatedAt* — όπως περιγράφεται από τον τύπο (3.4) — για να εκφραστούν οι περιορισμοί που πρέπει να ικανοποιούν οι STA προκειμένου να εκκινηθεί ή να τερματιστεί μια LTA.

(ii) Οι συντεταγμένες (*coordinates*, συντμ.: “*coord*”) των αναγνωρισμένων ανθρώπων και αντικειμένων ως θέσεις εικονοσημείων (*pixels*) σε κάθε χρονική στιγμή, καθώς και ο προσανατολισμός τους (*orientation*). Τα δύο αυτά στοιχεία αναπαριστώνται με τη χρήση του κατηγορήματος *holdsAt*. Επί παραδειγματι, το γεγονός *holdsAt(coord(id₂)=(14, 55), 10600)* εκφράζει ότι οι συντεταγμένες (σε *pixels*) του *id₂* είναι *(14, 55)* τη χρονική στιγμή *10600*. Από την άλλη, το γεγονός *holdsAt(orientation(id₂)=120, 10600)* εκφράζει ότι την ίδια χρονική στιγμή, ο ίδιος άνθρωπος σχηματίζει μια γωνία *120°* με τον άξονα των *x* στη δισδιάστατη προβολή του βίντεο. Αυτός ο τύπος πληροφορίας περιλαμβάνεται στο σύνολο όρων *Conditions[T]* του τύπου (3.4) που πρέπει να ικανοποιηθούν για την αναγνώριση μιας LTA.

(iii) Η πρώτη και η τελευταία φορά που το σύστημα παρατηρεί ένα συγκεκριμένο άνθρωπο ή αντικείμενο. Αυτός ο τύπος δεδομένων αναπαρίσταται με τη χρήση του κατηγορήματος *happensAt*. Για παράδειγμα, το γεγονός *happensAt(appear(id₁₀), 300)* δηλώνει ότι ο άνθρωπος (ή το αντικείμενο) *id₁₀* εμφανίστηκε πρώτη φορά στο βίντεο τη χρονική στιγμή *300*.

Δεδομένης αυτής της εισόδου, η Crisp-EC αναγνωρίζει τις εξής LTA: ένας άνθρωπος αφήνει ένα αντικείμενο (*leaving_object*), δύο άνθρωποι συναντιώνται (*meeting*), βαδίζουν μαζί (*moving*), ή καυγαδίζουν (*fighting*). Οι δραστηριότητες αυτές αναπαριστώνται ως *fluents*. Για παράδειγμα, το γεγονός *holdsAt(moving(id₁, id₃)=true, 140)* δηλώνει ότι ο άνθρωπος *id₁* περπατούσε μαζί με τον *id₃* τη χρονική στιγμή *140*.

Η αναγνώριση υψηλού επιπέδου βασίζεται σε ένα σύνολο ορισμών LTA οι οποίοι κωδικοποιούνται με τη χρήση των κατηγορημάτων *initiatedAt* και *terminatedAt*. Στην επόμενη ενότητα, παραθέτουμε χαρακτηριστικά τμήματα των ορισμών των LTA που περιλαμβάνονται στον κώδικα της Crisp-EC (σύνδεσμος προς τον κώδικα διαθέσιμος στην Εισαγωγή).

3.3 Ορισμοί των LTA της Crisp-EC

Η δραστηριότητα «άφεσης αντικειμένου» (*leaving_object*) ορίζεται ως ακολούθως:

$$\begin{aligned} \text{initiatedAt}(\text{leaving_object}(P, \text{Obj}) = \text{true}, T) \leftarrow \\ \text{happensAt}(\text{appear}(\text{Obj}), T), \\ \text{happensAt}(\text{inactive}(\text{Obj}), T), \\ \text{holdsAt}(\text{close}(P, \text{Obj}, 30) = \text{true}, T), \\ \text{holdsAt}(\text{person}(P) = \text{true}, T) \end{aligned} \quad (3.5)$$

$$\begin{aligned} \text{terminatedAt}(\text{leaving_object}(P, \text{Obj}) = \text{true}, T) \leftarrow \\ \text{happensAt}(\text{disappear}(\text{Obj}), T) \end{aligned} \quad (3.6)$$

Οι κάμερες του CAVIAR δεν ανιχνεύουν ένα αντικείμενο ενώ αυτό κρατείται από κάποιον άνθρωπο — μόνο ο εν λόγω άνθρωπος είναι ορατός σε αυτές σαν μια ολότητα. Το αντικείμενο θα γίνει ορατό (θα «εμφανιστεί» — “appear”) αν και μόνο αν ο άνθρωπος το αφήσει κάπου. Επιπροσθέτως, τα αντικείμενα στα βίντεο του CAVIAR μπορούν να επιδειξουν μόνο «ακινησία» (*inactive*), σε αντίθεση με τους ανθρώπους. Ο κανόνας (3.5) εκφράζει τις συνθήκες υπό τις οποίες η Crisp-EC αναγνωρίζει το *leaving_object*. Το fluent που καταγράφει αυτή τη δραστηριότητα, το *leaving_object(P, Obj)*, λαμβάνει τιμή αληθείας (true) τη χρονική στιγμή *T* αν η οντότητα *Obj* εμφανιστεί την *T*, η STA που παρατηρείται για την *Obj* είναι η *inactive* και υπάρχει ένας άνθρωπος *P* «κοντά» (“close”) στην *Obj* τη στιγμή *T*. Το fluent *close(ID₁, ID₂, Threshold)* εκφράζει ότι η απόσταση μεταξύ των οντοτήτων *ID₁* και *ID₂* είναι το πολύ *Threshold* pixels. Αυτό το fluent ορίζεται ως εξής:

$$\begin{aligned} \text{holdsAt}(\text{close}(\text{ID}_1, \text{ID}_2, \text{Threshold}) = \text{true}, T) \leftarrow \\ \text{holdsAt}(\text{distance}(\text{ID}_1, \text{ID}_2) = \text{Dist}, T), \\ \text{Dist} \leq \text{Threshold} \end{aligned} \quad (3.7)$$

Η απόσταση (distance) μεταξύ δύο αναγνωρισμένων ανθρώπων ή αντικειμένων υπολογίζεται ως η Ευκλείδεια Απόσταση μεταξύ των συντεταγμένων τους στην εικόνα ενός καρέ (υπενθυμίζουμε ότι οι συντεταγμένες αυτές δίνονται ως είσοδος στο σύστημά μας). Η στατική απόσταση των 30 pixels που φαίνεται στον κανόνα (3.5) επιλέχτηκε μετά από μια εμπειρική ανάλυσή μας του CAVIAR.

Ένα αντικείμενο που κάποιος σηκώνει παύει να ανιχνεύεται («εξαφανίζεται» — “disappears”), τερματίζοντας το *leaving_object*— πρβλ. κανόνα (3.6).

Στο CAVIAR δεν υπάρχει κάποια σαφής ενδειξη σχετικά με το αν κάποια αναγνωρισμένη οντότητα είναι άνθρωπος ή αντικείμενο. Συνεπώς, στους ορισμούς μας των δραστηριοτήτων υψηλού επιπέδου, προσπαθούμε να συμπεράνουμε αν μια οντότητα είναι άνθρωπος ή αντικείμενο χρησιμοποιώντας τις STA που έχουν ανιχνευτεί. Για το λόγο αυτό εισάγουμε στη βάση γνώσης μας το *fluent person(P)* και ορίζουμε το *fluent* αυτό να έχει τιμή *true* αν η οντότητα *P* έχει παρουσιάσει κάποια από τις STA *walking*, *running* ή *active* σε μια χρονική στιγμή μετά την εμφάνισή της στο βίντεο:

$$\begin{aligned}
 & \text{initiatedAt}(\text{person}(P) = \text{true}, T) \leftarrow \\
 & \quad \text{happensAt}(\text{walking}(P), T) \\
 & \text{initiatedAt}(\text{person}(P) = \text{true}, T) \leftarrow \\
 & \quad \text{happensAt}(\text{active}(P), T) \\
 & \text{initiatedAt}(\text{person}(P) = \text{true}, T) \leftarrow \\
 & \quad \text{happensAt}(\text{running}(P), T) \\
 & \text{terminatedAt}(\text{person}(P) = \text{true}, T) \leftarrow \\
 & \quad \text{happensAt}(\text{disappear}(P), T)
 \end{aligned} \tag{3.8}$$

Η τιμή του *person(P)* είναι χρονικά εξαρτώμενη επειδή στο CAVIAR ο δείκτης *P* μιας οντότητας η οποία «εξαφανίζεται» μπορεί κάποια στιγμή αργότερα να χρησιμοποιηθεί για να περιγράψει μια άλλη οντότητα που «εμφανίζεται» και η εν λόγω οντότητα μπορεί να μην είναι άνθρωπος. Αξιοσημείωτο είναι επίσης ότι ο κανόνας (3.5) κάνει μια — διαισθητικά προφανή — υπόθεση, δηλαδή ότι ένας άνθρωπος που εμφανίζεται σε κάποια χρονική στιγμή σε ένα βίντεο δε μπορεί να είναι, ταυτόχρονα, ακίνητος (*inactive*). Αν μια οντότητα είναι ακίνητη τη στιγμή που εμφανίζεται, μπορεί να θεωρηθεί με ασφάλεια ότι είναι ένα αντικείμενο που κάποιος άνθρωπος μόλις άφησε (πρβλ. πρώτες δύο συνθήκες του κανόνα (3.5)).

Με παρόμοιο τρόπο μπορούμε να ορίσουμε τους ορισμούς άλλων LTA. Η χρήση του Λογισμού Γεγονότων, σε συνδυασμό με την εκφραστικότητα του λογικού προγραμματισμού, μας επιτρέπει να εκφράσουμε ορισμούς LTA οι οποίοι εμπεριέχουν σύνθετους χρονικούς, χωρικούς και άλλου είδους περιορισμούς. Στη συνέχεια παρουσιάζουμε τμήματα των υπόλοιπων ορισμών LTA.

Η «συνάντηση» (*meeting*) δύο ανθρώπων P_1 και P_2 αναγνωρίζεται όταν δύο άνθρωποι «αλληλεπιδρούν»: τουλάχιστον ένας τους είναι σε κίνηση ή ακινησία ενώ ο άλλος δεν τρέχει και η μεταξύ τους απόσταση είναι το πολύ 25 pixels (όλες οι απαιτούμενες αποστάσεις εκτιμήθηκαν μέσω εμπειρικής ανάλυσης του συνόλου δεδομένων). Στο CAVIAR, αυτή η αλληλεπίδραση αντιστοιχεί συνήθως σε μια μορφή χαιρετισμού, όπως μια χειραψία. Οι κανόνες (3.9) και (3.10)

σκιαγραφούν τις συνθήκες υπό τις οποίες ξεκινά η αναγνώριση του *meeting*:

$$\begin{aligned}
 \text{initiatedAt}(\text{meeting}(P_1, P_2) = \text{true}, T) \leftarrow \\
 & \text{happensAt}(\text{inactive}(P_1), T), \\
 & \text{holdsAt}(\text{close}(P_1, P_2, 25) = \text{true}, T), \\
 & \text{holdsAt}(\text{person}(P_1) = \text{true}, T), \\
 & \text{holdsAt}(\text{person}(P_2) = \text{true}, T), \\
 & \text{not happensAt}(\text{running}(P_2), T), \\
 & \text{not happensAt}(\text{active}(P_2), T)
 \end{aligned} \tag{3.9}$$

$$\begin{aligned}
 \text{initiatedAt}(\text{meeting}(P_1, P_2) = \text{true}, T) \leftarrow \\
 & \text{happensAt}(\text{active}(P_1), T), \\
 & \text{holdsAt}(\text{close}(P_1, P_2, 25) = \text{true}, T), \\
 & \text{holdsAt}(\text{person}(P_2) = \text{true}, T), \\
 & \text{not happensAt}(\text{running}(P_2), T)
 \end{aligned} \tag{3.10}$$

Η LTA *meeting* τερματίζεται υπό μια πληθώρα διαφορετικών συνθηκών, για παράδειγμα όταν κάποιος από τους δύο ανθρώπους αρχίζει να τρέχει ή «εξαφανίζεται».

Η «συμβάδιση» (*moving*) ορίστηκε για να αναγνωρίζει το σύστημά μας χρονικές στιγμές στις οποίες δύο άνθρωποι περπατούν μαζί:

$$\begin{aligned}
 \text{initiatedAt}(\text{moving}(P_1, P_2) = \text{true}, T) \leftarrow \\
 & \text{happensAt}(\text{walking}(P_1), T), \\
 & \text{happensAt}(\text{walking}(P_2), T), \\
 & \text{holdsAt}(\text{close}(P_1, P_2, 34) = \text{true}, T), \\
 & \text{holdsAt}(\text{orientation}(P_1) = Or_1, T), \\
 & \text{holdsAt}(\text{orientation}(P_2) = Or_2, T), \\
 & |Or_1 - Or_2| < 45
 \end{aligned} \tag{3.11}$$

Προκειμένου να αναγνωρίσει η Crisp-EC το *moving*, οι δύο εμπλεκόμενοι άνθρωποι οφείλουν να περπατούν (*walking*) ενώ είναι «κοντά» ο ένας στον άλλον. Ταυτόχρονα, πρέπει να βαδίζουν προς την ίδια, περίπου, κατεύθυνση (αν δεν υπήρχε αυτός ο περιορισμός, το σύστημα μπορεί να αναγνώριζε τη δραστηριότητα *moving* μεταξύ δύο ανθρώπων οι οποίοι βαδίζουν σε αντίρροπες κατευθύνσεις, κάτι μη θεμιτό). Αυτό επιτυγχάνεται με το να περιορίζουμε τους προσανατολισμούς (*orientation*) τους εντός ενός οκταμορίου (45°) καθώς αυτοί βαδίζουν.

Η δραστηριότητα *moving* τερματίζεται όταν οποιοσδήποτε από τους δύο ανθρώπους απομα-

κρύνεται (“walks away”) από τον άλλον σε σχέση με το προκαθορισμένο όριο των 34 pixels:

$$\begin{aligned} \text{terminatedAt}(\text{moving}(P_1, P_2) = \text{true}, T) \leftarrow \\ \text{happensAt}(\text{walking}(P_1), T), \\ \text{holdsAt}(\text{close}(P_1, P_2, 34) = \text{false}, T) \end{aligned} \quad (3.12)$$

Σύμφωνα με άλλες τερματικές συνθήκες του *moving*, η LTA τερματίζεται όταν κάποιος από τους δύο ανθρώπους αρχίζει να τρέχει μακριά από τον άλλο, ή όταν κάποιος ή και οι δύο άνθρωποι «εξαφανίζονται» από τον επιτηρούμενο χώρο.

Ο τελευταίος ορισμός μας αφορά την LTA «διαμάχη» (*fighting*). Σε προηγούμενα πειράματά μας [3], ορίζαμε το *fighting* με χρήση των STA *running* και *active*. Αυτός ο φορμαλισμός μας οδηγούσε σε πολύ ανακριβή αναγνώριση, λόγω του ότι η ποικιλία των STA του CAVIAR είναι ιδιαίτερα περιορισμένη. Προκειμένου να υπερκεράσουμε αυτό το πρόβλημα, προσθέσαμε την STA *abrupt*, η οποία ανταποκρίνεται σε βίαιη κίνηση των χεριών και των ποδιών ενός ανθρώπου. Η STA *abrupt* είναι μια μορφή δραστηριότητας χαμηλού επιπέδου η οποία μπορεί να αναγνωριστεί από κάποια σύγχρονα συστήματα εντοπισμού, όπως αυτό που περιγράφεται στο [21]. Με τη χρήση αυτής της νέας δραστηριότητας, ορίζουμε το *fighting* ως εξής:

$$\begin{aligned} \text{initiatedAt}(\text{fighting}(P_1, P_2) = \text{true}, T) \leftarrow \\ \text{happensAt}(\text{abrupt}(P_1), T), \\ \text{holdsAt}(\text{close}(P_1, P_2, 44) = \text{true}, T), \\ \text{not happensAt}(\text{inactive}(P_2), T) \end{aligned} \quad (3.13)$$

Για να αναγνωρίσουμε το *fighting*, απαιτούμε από τους δύο ανθρώπους να είναι αρκετά κοντά ο ένας στον άλλο και τουλάχιστον ένας από αυτούς να παρουσιάζει τη δραστηριότητα *abrupt*, ενώ ο άλλος να μην είναι ακίνητος (*inactive*), υπονοώντας ότι πρέπει με κάποιον τρόπο να συμμετέχει στη διαμάχη. Η Crisp-EC σταματά να αναγνωρίζει το *fighting* όταν τουλάχιστον ένα από τα δύο άτομα απομακρύνεται από τον τόπο της διαμάχης (περπατώντας ή τρέχοντας), ή αποχωρεί από τον επιτηρούμενο χώρο.

ΚΕΦΑΛΑΙΟ 4

ΠΙΘΑΝΟΤΙΚΟΣ ΛΟΓΙΣΜΟΣ ΓΕΓΟΝΟΤΩΝ

Στο κεφάλαιο αυτό παρουσιάζουμε την Prob-EC, μια πιθανοτική επέκταση της Crisp-EC για χρήση σε αβέβαια περιβάλλοντα. Η Prob-EC, όπως αναφέραμε στην Εισαγωγή, είναι βασισμένη στη γλώσσα πιθανοτικού λογικού προγραμματισμού ProbLog. Στο πρώτο σκέλος της παρουσίασής μας απαριθμούμε και εξηγούμε τις συντακτικές μετατροπές που πρέπει να λάβουν χώρα σε μια διάλεκτο Λογισμού Γεγονότων βασισμένη σε λογικό προγραμματισμό (όπως η Crisp-EC) προκειμένου να την κάνουμε συμβατή με την ProbLog. Στο δεύτερο σκέλος, εξηγούμε το πώς λειτουργεί ο συμπερασμός (inference) στην Prob-EC, χρησιμοποιώντας δύο χαρακτηριστικά σενάρια αναγνώρισης ανθρώπινης συμπεριφοράς.

4.1 Από την Crisp-EC στην Prob-EC

Στην ενότητα 2.2 αναφέραμε τα βασικά χαρακτηριστικά της ProbLog, συμπεριλαμβανομένων των ενσωματωμένων κατηγορημάτων άρνησης (`problog_not`, `problog_neg`). Για να εκφράσουμε την Prob-EC με ακρίβεια, χρειάστηκε να συνδυάσουμε τα κατηγορήματα αυτά με το κλασικό `not`.

Μια πρόκληση την οποία έπρεπε να αντιμετωπίσουμε ήταν ότι τα κατηγορήματα `problog_not` και `problog_neg` δε συμπεριφέρονταν πάντα όπως περιμέναμε. Συγκεκριμένα, όταν εφαρμόζονται σε ένα άτομο (πιθανοτικό γεγονός για το `problog_not` και κεφαλή οποιουδήποτε στόχου για το `problog_neg`) το οποίο δεν είναι αποδείξιμο από το λογικό κομμάτι της βάσης γνώσης μας, αντί να παράγουν πιθανότητα 1 (όπως επιτυγχάνεται μέσω του `not`), «αποτυγχάνουν» (`fail`) σιωπηρά και επιστρέφουν πιθανότητα 0. Κατ' επέκταση, η ευθεία αντικατάσταση του `not` στα γενικά αξιώματα της Crisp-EC (ενότητα 3.1) και στους ορισμούς των LTA (ενότητα 3.3) με τα `problog_neg` και `problog_not` αντίστοιχα δε θα επετύγχανε αυτό ακριβώς που θέλουμε. Για παράδειγμα, όταν

περιλαμβάνουμε στον ορισμό κάποιας LTA την άρνηση κάποιας STA (όπως είναι λόγου χάρη η άρνηση των *running* και *active* στον τύπο (3.9)), θέλουμε να παράγουμε πιθανότητα 1 όταν η STA δεν υπάρχει στην είσοδό μας. Θέλουμε όμως και το συμπλήρωμα της πιθανότητας όταν η STA υπάρχει στην είσοδό μας με κάποια πιθανότητα. Για να αποκτήσουμε αυτή τη λειτουργικότητα στην Prob-EC, ορίσαμε τα κατηγορήματα $negate_1$ και $negate_2$:

$$\begin{aligned} negate_1(Fact) &\leftarrow problog_not(Fact) \\ negate_1(Fact) &\leftarrow not\ Fact \end{aligned} \quad (4.1)$$

$$\begin{aligned} negate_2(Goal) &\leftarrow problog_neg(Goal) \\ negate_2(Goal) &\leftarrow not\ Goal \end{aligned} \quad (4.2)$$

Έτσι μπορούμε να μετατρέψουμε τους ορισμούς των LTA ώστε να πάρουμε τη συμπληρωματική πιθανότητα ενός πιθανοτικού λεκτικού όταν αυτή υπάρχει στην είσοδο, καθώς και τη διαισθητική πιθανότητα 1 όταν το λεκτικό στο οποίο επιβάλλουμε την άρνηση δεν υπάρχει στην είσοδο (πρακτικά, αυτό ισοδυναμεί με το να υπάρχει στην είσοδο με πιθανότητα 0). Για παράδειγμα, ο κανόνας (3.13) θα γραφτεί στην Prob-EC ως εξής:

$$\begin{aligned} initiatedAt(fighting(P_1, P_2) = true, T) &\leftarrow \\ happensAt(abrupt(P_1), T), \\ holdsAt(close(P_1, P_2, 44) = true, T), \\ negate_1(happensAt(inactive(P_2), T)) \end{aligned} \quad (4.3)$$

Αξίζει να σημειωθεί ότι η μόνη περίπτωση στην οποία το $negate_1$ θα παράξει μια πιθανότητα ίση με 0 είναι όταν το άτομο στο οποίο εφαρμόζεται έχει πιθανότητα ίση με 1. Αυτό είναι διαισθητικά προφανές και σύμφωνα με τη λογική της Prob-EC.

Εκτός από τη μετάφραση των ορισμών των LTA της Crisp-EC, παρουσιάστηκε η ανάγκη να προσαρμόσουμε τα γενικά αξιώματα της Crisp-EC ((3.1) - (3.3)) στη νέα μέθοδο άρνησης. Το αξίωμα (3.1) χρησιμοποιεί «άρνηση ως αποτυχία» για να δηλώσει ότι ένα *fluent* κρατάει την τιμή του μέχρι να τερματιστεί από κάποια συνθήκη τερματισμού (*termination condition*). Στην Prob-EC, τόσο οι συνθήκες αρχικοποίησης (*initiation conditions*, ουσιαστικά οι *initiatedAt* κανόνες που δείξαμε στην ενότητα 3.3) όσο και οι συνθήκες τερματισμού (οι *terminatedAt* κανόνες της ενότητας 3.3) μπορεί να έχουν κάποια πιθανότητα. Θα θέλαμε να χρησιμοποιήσουμε αυτήν την πληροφορία ώστε να εκτιμήσουμε την ακριβή πιθανότητα των εκάστοτε επερωτήσεών μας τύπου

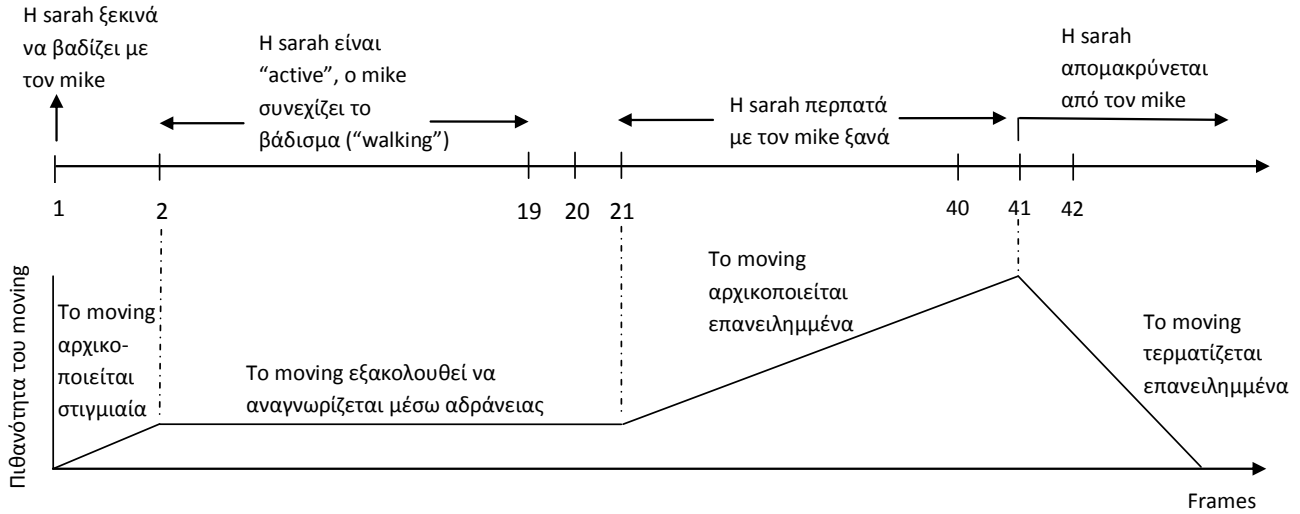
$\text{holdsAt}(F = V, T)$. Για το λόγο αυτό, μεταφράσαμε το αξίωμα (3.1) ως εξής:

$$\begin{aligned} \text{holdsAt}(F = V, T) \leftarrow & \\ & \text{initiatedAt}(F = V, T_s), \\ & T_s < T, \\ & \text{negate}_2(\text{broken}(F = V, T_s, T)) \end{aligned} \quad (4.4)$$

Η σημασιολογία του αξιώματος (4.4) είναι η εξής: η πιθανότητα ότι $F = V$ τη χρονική στιγμή T προκύπτει ως συνδυασμός της πιθανότητας εκκίνησης του $F = V$ στο παρελθόν (T_s) και της πιθανότητας μη - τερματισμού του έκτοτε. Ο υπολογισμός της συνδυασμένης πιθανότητας θα παρουσιαστεί στην επόμενη ενότητα. Η χρήση του negate_2 στο αξίωμα (4.4) δίνει μια ενδιαφέρουσα διάσταση στον τερματισμό των δραστηριοτήτων που αναγνωρίζει η Prob-EC: αν κατά τη διάρκεια της ανάλυσης του βίντεο ικανοποιούνται αλληπάλληλα οι τερματικές συνθήκες για κάποια LTA (για παράδειγμα, η συνθήκη (3.12) που αφορά την LTA *moving*) και οι συνθήκες αυτές είναι πιθανοτικές (στο ίδιο παράδειγμα, πιθανότητα στα *happensAt* της STA *walking* οδηγεί σε μια πιθανότητα στην ίδια τη συνθήκη τερματισμού), τότε η πιθανότητα του να ισχύει $F = V$ δε θα πέσει κατευθείαν στο 0, αλλά σταδιακά, με μια κλίση που εξαρτάται από τις πιθανότητες των αλληπάλληλων τερματικών συνθηκών.

4.2 Συμπερασμός στην Prob-EC

Ένα τυπικό παράδειγμα συμπερασμού της Prob-EC σκιαγραφείται στο Σχήμα 4.1. Στο παράδειγμα αυτό, δύο άνθρωποι, ο mike και η sarah, εμπλέκονται στην LTA *moving* για έναν αριθμό από frames. Η δραστηριότητα εκκινείται για πρώτη φορά στο frame 1, ένα frame στο οποίο ο mike και η sarah βαδίζουν σε μια περίπου ομόρροπη κατεύθυνση, όπως απαιτείται από τον κανόνα (3.11). Στο δεύτερο frame, η sarah σταματά να βαδίζει (*walking*), στοιχείο το οποίο απαιτείται από τον κανόνα (3.11) για να αναγνωριστεί το *moving*, και αντ' αυτού παρουσιάζει «κίνηση» (*active*). Προκειμένου να τερματιστεί η LTA *moving* μέσω της STA *active*, θα πρέπει και τα δύο άτομα να πραγματοποιούν την εν λόγω STA. Η κατάσταση αυτή συνεχίζει μέχρι το frame 21, όταν η sarah για μια ακόμα φορά ξεκινά το βάδισμα (*walking*) και έτσι το *moving* εκκινείται ξανά. Στο frame 41 ο mike σταματά να βαδίζει, παρουσιάζοντας «ακίνησία» (*inactive*), ενώ η sarah συνεχίζει να βαδίζει. Στο σημείο αυτό κάνουμε την υπόθεση ότι η sarah απομακρύνεται από τον mike περισσότερο από 34 pixels, ενεργοποιώντας τη συνθήκη τερματισμού (3.12) (“walk away”). Για να απλοποιήσουμε το παράδειγμα, ας υποθέσουμε ότι όλες οι τιμές των context variables

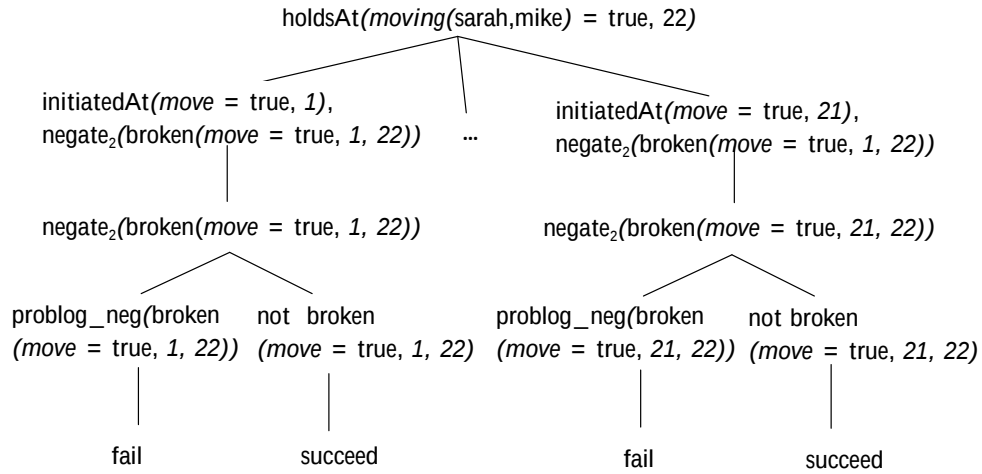


Σχήμα 4.1: Η διακύμανση της πιθανότητας του ερωτήματος $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, T)$

(coordinates, orientation) υπολογίζονται με βεβαιότητα (πιθανότητα 1), ενώ οι STA *walking*, *active* και *inactive* αναγνωρίζονται με τις παρακάτω πιθανότητες αναγνώρισης (detection probabilities):

$0.70 :: \text{happensAt}(\text{walking}(\text{mike}), 1)$	$0.46 :: \text{happensAt}(\text{walking}(\text{sarah}), 1)$
$0.73 :: \text{happensAt}(\text{walking}(\text{mike}), 2)$	$0.55 :: \text{happensAt}(\text{active}(\text{sarah}), 2)$
...	
$0.69 :: \text{happensAt}(\text{walking}(\text{mike}), 21)$	$0.58 :: \text{happensAt}(\text{walking}(\text{sarah}), 21)$
...	
$0.18 :: \text{happensAt}(\text{inactive}(\text{mike}), 41)$	$0.32 :: \text{happensAt}(\text{walking}(\text{sarah}), 41)$

Στο δεύτερο frame, το ερώτημα $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, 2)$ έχει πιθανότητα ίση με την πιθανότητα της συνθήκης εκκίνησης που ικανοποιήθηκε στο πρώτο frame. Η πιθανότητα αυτή, σύμφωνα με τον τύπο (3.11) και με δεδομένο ότι οι συντεταγμένες και κατευθύνσεις υπολογίστηκαν με πλήρη βεβαιότητα, ισούται με το γινόμενο των πιθανοτήτων με τις οποίες θεωρούμε ότι το σύστημα χαμηλού επιπέδου έχει ανιχνεύσει ότι ο mike και η sarah περπατούν αμφότεροι, δηλαδή $0.70 \times 0.46 = 0.322$. Αυτό φαίνεται στο αριστερο κομμάτι της εικόνας 4.1, όπου η πιθανότητα της LTA *moving* αυξάνεται από το 0 στο 0.322. Από το frame 2 μέχρι το frame 20, ο ταξινομητής λέει ότι ο mike συνεχίζει να περπατά, όχι όμως και η sarah. Κατ' επέκταση, στο χρονικό αυτό διάστημα δεν ικανοποιείται καμία συνθήκη εκκίνησης ή τερματισμού και η πιθανότητα του *moving* μένει



Σχήμα 4.2: Το SLD-δένδρο για το επερώτημα $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, 22)$.

σταθερή. Αυτό συμβαίνει λόγω του «νόμου της αδράνειας» και αναπαρίσταται στο Σχήμα 4.1 με την οριζόντια γραμμή μεταξύ των frames 2 και 20. Στο frame 21, η sarah ξεκινά πάλι να περπατά μαζί με τον mike. Συνεπώς, στο frame 22, το επερώτημα $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, 22)$ συνδυάζει δύο διαφορετικές συνθήκες εκκίνησης στο συμπέρασμα, η μία ενεργοποιημένη στο frame 1 και η άλλη στο frame 21. Αυτό συμβαίνει γιατί ο κανόνας (4.4) σαρώνει όλα τα frames από το πρώτο μέχρι και το τρέχον για να βρει συνθήκες εκκίνησης μιας LTA.

Όπως αναφέραμε στην ενότητα 2.2, η ProbLog ξεκινά τον υπολογισμό της πιθανότητας ενός επερωτήματος με τη σάρωση του SLD-δένδρου του επερωτήματος. Στο Σχήμα 4.2 μπορούμε να δούμε το δένδρο για το επερώτημα $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, 22)$. Το δεύτερο βήμα του συμπερασμού της ProbLog αναπαριστά αυτό το δένδρο ως μια φόρμουλα σε μορφή DNF. Στο συγκεκριμένο παράδειγμα, η DNF είναι η εξής:

$$\underbrace{\text{initiatedAt}(\text{moving}(\text{sarah}, \text{mike}) = \text{true}, 1)}_{init_1} \vee \underbrace{\text{initiatedAt}(\text{moving}(\text{sarah}, \text{mike}) = \text{true}, 21)}_{init_{21}} \quad (4.5)$$

Απλοποιούμε την αναπαράσταση με την απαλοιφή των δύο όρων που περιέχουν το κατηγορημα negate_2 , καθώς — όπως φαίνεται στο Σχήμα 4.2 — είναι και οι δύο αποδείξεις μέσω της «άρνησης ως αποτυχία», συνεπώς έχουν πιθανότητα ίση με 1. Αυτό συμβαίνει γιατί μεταξύ των frames 1 και 22 δεν έχει λάβει χώρα καμία συνθήκη τερματισμού, επομένως το negate_2 λειτουργεί σαν το not (κανόνας (4.2)).

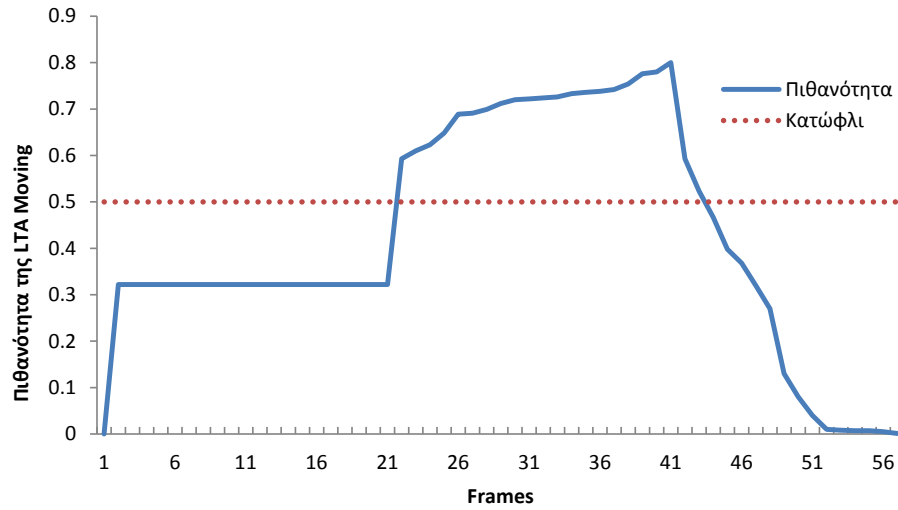
Μέχρι το frame 22, ικανοποιούνται μόνο δύο φορές οι συνθήκες εκκίνησης της LTA moving , συγκεκριμένα οι $init_1$ και $init_{21}$ (πρβλ. τύπο (4.5)). Στη γενική όμως περίπτωση, ενδέχεται να

υπάρχουν πολλές χρονικές στιγμές στις οποίες ικανοποιούνται οι συνθήκες εκκίνησης από την αρχή του βίντεο έως και την τρέχουσα χρονική στιγμή. Για κάθε συνθήκη που ικανοποιείται, ο κανόνας (4.4) θα εξετάσει το χρονικό διάστημα μεταξύ της εκκίνησης και της τρέχουσας χρονικής στιγμής για να διαπιστώσει αν το *fluent* που αναπαριστά την LTA έχει «σπάσει» (*broken*), δηλαδή αν η LTA έχει τερματιστεί. Η διαδικασία αυτή επαναλαμβάνεται στο επόμενο *frame*. Έτσι καταλήγουμε να έχουμε πολλούς περιττούς υπολογισμούς. Προκειμένου να αποφύγουμε αυτό το πρόβλημα, υλοποιήσαμε μια απλή μέθοδο «προσωρινής αποθήκευσης» (*caching*), σύμφωνα με την οποία η πιθανότητα του επερωτήματος $\text{holdsAt}(F = V, T - 1)$ αποθηκεύεται στη μνήμη. Δεδομένης αυτής της αποθηκευμένης πληροφορίας, το επερώτημα $\text{holdsAt}(F = V, T)$ χρειάζεται μόνο να εξετάσει αν την προηγούμενη χρονική στιγμή ικανοποιήθηκαν συνθήκες εκκίνησης ή τερματισμού οι οποίες επηρεάζουν την πιθανότητα της εν λόγω LTA. Αυτή η τεχνική μπορεί να λειτουργήσει κάτω από την υπόθεση ότι το σύστημα αναγνώρισης δραστηριοτήτων λαμβάνει τα *frames* του βίντεο χρονικά ταξινομημένα, μια υπόθεση που ισχύει στο CAVIAR. Η μέθοδος αυτή έχει υλοποιηθεί στον κώδικα που παραθέτουμε στην ενότητα 1.2.

Το τρίτο σκέλος του συμπερασμού της ProbLog περιλαμβάνει τη μετάφραση του DNF τύπου σε ένα BDD και τον αναδρομικό υπολογισμό της πιθανότητας. Το παράδειγμά μας είναι αρκετά απλό ώστε να μας επιτρέπει να πραγματοποιήσουμε υπολογισμούς με το χέρι, καθώς υπάρχουν μόνο δύο αποδείξεις για το *holdsAt*. Η πιθανότητα του DNF τύπου (4.5) μπορεί να υπολογιστεί ως η πιθανότητα της διάζευξης δυο κανόνων, όπως περιγράφηκε στην ενότητα 2.2:

$$\begin{aligned} P(\text{init}_1 \vee \text{init}_{21}) &= P(\text{init}_1) + P(\text{init}_{21}) - P(\text{init}_1 \wedge \text{init}_{21}) = \\ &= 0.70 \times 0.46 + 0.69 \times 0.58 - 0.7 \times 0.46 \times 0.69 \times 0.58 = 0.593 \end{aligned}$$

Η πιθανότητα του $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, 22)$ έχει αυξηθεί σε σχέση με το αντίστοιχο ερώτημα του δεύτερου *frame*, λόγω της επιπρόσθετης εκκίνησης του *moving* στο *frame* 21. Αυτό είναι ένα από τα χαρακτηριστικά της Prob-EC: η συνεχής ικανοποίηση των συνθηκών εκκίνησης μιας συγκεκριμένης LTA προκαλεί την αύξηση της πιθανότητάς της. Αυτή η συμπεριφορά είναι σύμφωνη με την ανθρώπινη διαίσθηση: δεδομένων συνεχών ενδείξεων για το ότι μια δραστηριότητα λαμβάνει χώρα, το σύστημα πιστεύει ολοένα και περισσότερο ότι η δραστηριότητα όντως συμβαίνει, ακόμα και αν η αξιοπιστία κάθε μίας από τις ατομικές ενδείξεις (εκφρασμένη από την αντίστοιχη πιθανότητά της) είναι σχετικά χαμηλή. Για το λόγο αυτό, από το *frame* 22 και μέχρι το *frame* 41, η πιθανότητα του *moving* αυξάνεται μονότονα (πρβλ. Σχήμα 4.1). Στο εν λόγω παράδειγμα, στο *frame* 41 η πιθανότητα έχει φτάσει στην τιμή 0.8.



Σχήμα 4.3: Η έξοδος της Prob-EC για το επερώτημα $\text{holdsAt}(\text{moving}(\text{mike}, \text{sarah}) = \text{true}, T)$ για διάφορα frames T .

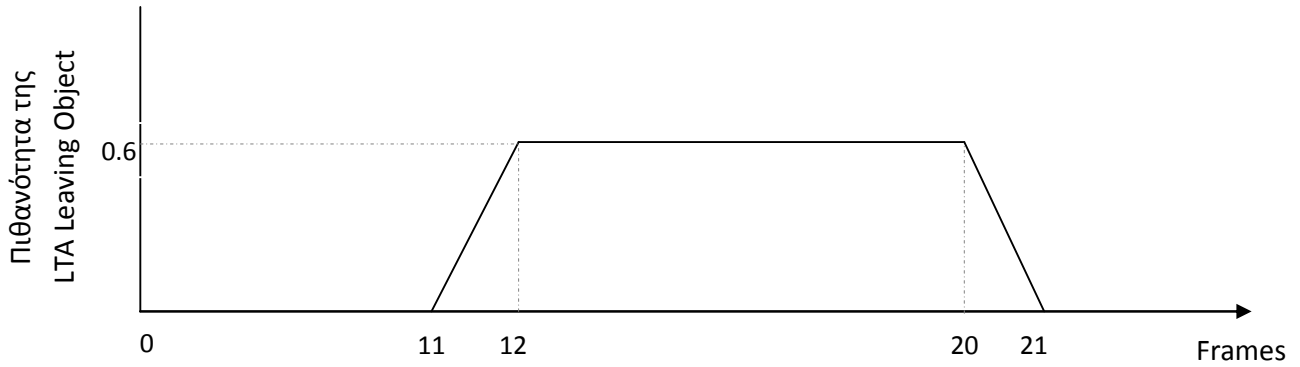
Στο frame 42, η Prob-EC πρέπει να λάβει υπ' όψιν της την τερματική συνθήκη που ενεργοποιείται στο frame 41. Αυτή η συνθήκη, η οποία αντιστοιχεί στον κανόνα (3.12), είναι πιθανοτική· η πιθανότητά της ισούται με την πιθανότητα του να απομακρύνθηκε η *sarah* από τον *mike*. Η πιθανότητα αυτή, σύμφωνα με τον κανόνα (3.12) και δεδομένου του ότι θεωρούμε ότι το κατηγορήμα *close* έχει πιθανότητα 1 (αφού οι συντεταγμένες του *mike* και της *sarah* ανιχνεύονται με βεβαιότητα), ισούται με την ίδια την πιθανότητα της STA *walking*, η οποία είναι 0.32. Συνεπώς, καθώς υπολογίζει την πιθανότητα του να εμπλέκονται ο *mike* και η *sarah* στην LTA *moving* στο frame 42, η Prob-EC πρέπει να συνεκτιμήσει την πιθανότητα όλων των πιθανών κόσμων στους οποίους η *sarah* δεν έχει απομακρυνθεί από τον *mike*. Η πιθανότητα όλων αυτών των κόσμων υπολογίζεται με τη χρήση του κατηγορήματος negate_2 στον κανόνα (4.4) και ισούται με $1 - 0.32 = 0.68$. Επομένως, η πιθανότητα του ότι η *sarah* και ο *mike* περπατούν ακόμα μαζί (*moving*) στο frame 42, παραβλέποντας την πιθανότητα το *moving* να τερματίστηκε στο frame 41, είναι ίση με $0.8 \times 0.68 = 0.544$. Παρομοίως με την σταθερή αύξηση της πιθανότητας δεδομένων συνεχών συνθηκών αρχικοποίησης, όταν αντιμετωπίζουμε συνεχείς τερματικές συνθήκες, η πιθανότητα μιας LTA μειώνεται σταθερά. Η μέση κλίση της ανόδου (καθόδου) ορίζεται από τις πιθανότητες των σχετικών συνθηκών αρχικοποίησης (τερματισμού). Στο παράδειγμά μας υποθέτουμε ότι η *sarah* συνεχίζει να απομακρύνεται από τον *mike* μέχρι το τέλος του βίντεο και ως εκ τούτου η πιθανότητα πλησιάζει ολόένα και περισσότερο το μηδέν, όπως φαίνεται στο Σχήμα 4.1.

Στο σχήμα 4.3 φαίνεται η ακριβής έξοδος της Prob-EC για το παράδειγμά μας. Μετά από μια απότομη άνοδο από το 0 στο 0.322, η πιθανότητα μένει σταθερή μεταξύ των frames 2 και

21, υποδεικνύοντας ότι για αυτή τη χρονική περίοδο η LTA εξακολουθεί να υφίσταται μέσω του νόμου της αδράνειας. Μεταξύ των frames 22 και 41 η πιθανότητα της LTA αυξάνεται μονότονα, ανταποκρινόμενη στην παρουσία συνεχών αρχικοποιήσεων του *moving* που συμβαίνουν σε αυτή τη χρονική περίοδο. Μετά το frame 41, μειώνεται, καθώς οι επαναλαμβανόμενες συνθήκες τερματισμού μειώνουν την πιθανότητα του να λαμβάνει χώρα το *moving*. Η οριζόντια στικτή γραμμή στην τιμή πιθανότητας 0.5 αντιπροσωπεύει το «κατώφλι» (threshold) πιθανότητας που χρησιμοποιούμε προκειμένου να διαχωρίσουμε τα «θετικά» (positives) των LTA τα οποία και θεωρούμε αρκετά αξιόπιστα — αυτά τα ονομάζουμε «αναγνώσεις» (recognitions) — και αυτά τα οποία θεωρούμε μη αξιόπιστα. Σε άλλες εφαρμογές, το κατώφλι αναγνώρισης μπορεί να είναι διαφορετικό από το 0.5.

Αυτό το παράδειγμα παρουσίασε μια περίπτωση LTA η οποία αναγνωρίζεται μετά από ένα συγκεκριμένο αριθμό συνεχών αρχικοποιήσεων. Αναφερόμαστε σε τέτοιες LTA ως «αδύναμα αρχικοποιημένες» (*weakly initiated*). Κατ' αντιστοιχία με αυτές τις δραστηριότητες, υπάρχουν LTA οι οποίες είναι «ισχυρά αρχικοποιημένες» (*strongly initiated*). Αυτές οι LTA αναγνωρίζονται μέσω μίας μονάχα συνθήκης αρχικοποίησης και η περαιτέρω αναγνώριση της δραστηριότητας επιτυγχάνεται μέσω του νόμου της αδράνειας. Η LTA *leaving_object* είναι μια τέτοια δραστηριότητα, καθώς δεν έχει ιδιαίτερο νόημα στον πραγματικό κόσμο να υποθέσει κανείς ότι ένας άνθρωπος μπορεί να αφήνει ένα αντικείμενο για πολλά συνεχή frames.

Η επιρροή της ισχυρής αρχικοποίησης στην έξοδο της Prob-EC είναι σημαντική. Αυτό συμβαίνει λόγω της έλλειψης επαναλαμβανόμενων αρχικοποιήσεων οι οποίες — όπως εξηγήσαμε στο προηγούμενο παράδειγμα — προκαλούν μια αύξηση της πιθανότητας της LTA. Η πιθανότητα μιας ισχυρά αρχικοποιημένης δραστηριότητας υψηλού επιπέδου είναι εξ' ολοκλήρου εξαρτώμενη από την πιθανότητα της μοναδικής της συνθήκης αρχικοποίησης. Αν η πιθανότητα αυτή είναι πάνω από το κατώφλι αναγνώρισης (0.5 στην περίπτωση μας), τότε τα frames στα οποία η LTA θα εξακολουθεί να υφίσταται θα μετρούν όλα σαν αναγνώσεις. Αυτή η συμπεριφορά μπορεί να εξηγηθεί μέσω άλλου ενός παραδείγματος — πρβλ. Σχήμα 4.4. Ας υποθέσουμε ότι η Sarah περπατάει ενώ ταυτόχρονα κρατάει μια βαλίτσα για 10 frames. Στο ενδέκατο frame, αφήνει τη βαλίτσα στο πάτωμα και απομακρύνεται από αυτήν. Στο ίδιο λοιπόν frame η βαλίτσα «εμφανίζεται» (“appear”) στο σύστημα αναγνώρισης χαμηλού επιπέδου, ενεργοποιώντας τη συνθήκη αρχικοποίησης (3.5). Ας υποθέσουμε ότι η εν λόγω συνθήκη αρχικοποίησης έχει πιθανότητα ίση με 0.6. Στο εικοστό frame, η Sarah σηκώνει τη βαλίτσα, προκαλώντας την «εξαφάνισή» της μέσω της τερματικής συνθήκης (3.6). Για λόγους απλότητας της παρουσίασης, θεωρούμε ότι και αυτή η συνθήκη τερματισμού έχει πιθανότητα ίση με 0.6. Όπως φαίνεται στο σχήμα 4.4, λόγω της έλλειψης συνθηκών αρχικοποίησης μετά το frame 11, η δραστηριότητα *leaving_object*



Σχήμα 4.4: Παράδειγμα της διακύμανσης πιθανότητας μιας «ισχυρά εκκινούμενης» (strongly initiated) LTA.

αναγνωρίζεται μόνο μέσω του νόμου της αδράνειας. Η πιθανότητα αυτής της LTA είναι ίση με την πιθανότητα της συνθήκης αρχικοποίησης, δηλαδή 0.6. Επειδή η πιθανότητα αυτή είναι πάνω από το 0.5, όλα τα frames μέχρι αυτό στο οποίο η *sarah* σηκώνει τη βαλίτσα θα αναγνωριστούν επιτυχώς. Αν, όμως, η πιθανότητα της συνθήκης αρχικοποίησης ήταν κάτω από το 0.5, το *leaving_object* δε θα αναγνωριζόταν.

Σημειώστε ότι στην Prob-EC είναι εφικτό να μετατρέψουμε οποιαδήποτε ισχυρά αρχικοποιημένη LTA σε μια αδύναμα αρχικοποιημένη έκδοση της ίδιας LTA, με το να υποθέσουμε ότι κάθε frame στο οποίο ισχύει η αδύναμα αρχικοποιημένη LTA είναι μια συνθήκη αρχικοποίησης για την αδύναμα αρχικοποιημένη έκδοση. Ο ακόλουθος κανόνας δείχνει το πώς θα μπορούσε αυτό να είχε υλοποιηθεί στην περίπτωση του *leaving_object*:

$$\begin{aligned} \text{initiatedAt}(\text{leaving_object_weakly_initiated}(P, \text{Obj}) = \text{true}, T) \leftarrow \\ \text{holdsAt}(\text{leaving_object}(P, \text{Obj}) = \text{true}, T) \end{aligned} \quad (4.6)$$

Παρά το γεγονός ότι αυτή η μεταβολή θα ήταν ευεργετική για τα πειράματά μας στο CAVIAR, αφού θα μπορούσαμε να αυξήσουμε την πιθανότητα του *leaving_object* μέσω της ανωτέρω αδύναμα αρχικοποιημένης έκδοσής του, υπάρχουν και κάποιοι παράπλευροι κίνδυνοι. Ας θεωρήσουμε, για παράδειγμα, ένα σενάριο στο οποίο το *leaving_object* αναγνωρίζεται με μια πολύ μικρή πιθανότητα, λόγου χάρη 0.01, η οποία υποδεικνύει ότι η πεποίθηση του συστήματος σχετικά με το αν έχει όντως συμβεί αυτή η δραστηριότητα στο συγκεκριμένο frame είναι πολύ χαμηλή. Τότε, η Prob-EC θα εκτιμήσει ότι το *leaving_object* συμβαίνει με πιθανότητα 0.01 για όλα τα frames μέχρι εκείνο στο οποίο το αντικείμενο «εξαφανίζεται». Δεδομένου του κατωφλίου μας, όλα αυτά τα «θετικά» του *leaving_object* θα απορριφθούν, λόγω της πολύ μικρής πιθανό-

τητάς τους. Η αδύναμα αρχικοποιημένη έκδοση του *leaving_object*, όμως (πρβλ. κανόνα (4.6)), θα συνεχίσει να αυξάνει την πιθανότητα της συγκεκριμένης LTA η οποία εν τέλει θα ξεπεράσει το κατώφλι του 0.5. Έτσι θα παραχθούν πολλά Ψευδή Θετικά (False Positives - FP), ειδικά αν τα frames στα οποία το αντικείμενο ήταν αφύλακτο ήταν πολυάριθμα. Παρά το γεγονός ότι τέτοιες συνθήκες δε συμβαίνουν στο CAVIAR, η παρουσία τους σε άλλα σύνολα δεδομένων είναι πολύ πιθανή, ειδικά σε εφαρμογές όπου δίνεται μεγάλη έμφαση στην αναγνώριση αφύλακτων αντικειμένων.

Παρομοίως με την «ισχυρή» ή «αδύναμη» αρχικοποίηση, μια LTA μπορεί να είναι «ισχυρά» ή «αδύναμα» *τερματιζόμενη*. Για παράδειγμα, η δραστηριότητα *moving* είναι αδύναμα τερματιζόμενη, γιατί ο τερματισμός της λαμβάνει χώρα σε πολλά συνεχή frames, ενώ η *leaving_object* είναι ισχυρά τερματιζόμενη στο frame στο οποίο ένας άνθρωπος σηκώνει το αντικείμενο ενδιαφέροντος. Στη γενική περίπτωση, δεν υπάρχει κάποια προκαθορισμένη σχέση μεταξύ της αρχικοποίησης και του τερματισμού ενός fluent — για παράδειγμα, μια LTA μπορεί να είναι αδύναμα αρχικοποιημένη και ισχυρά τερματιζόμενη. Στην περίπτωση του ισχυρού τερματισμού — δηλαδή, εν τη απουσία συνεχώς ενεργοποιημένων τερματικών συνθηκών — η πιθανότητα μιας LTA μπορεί να μην πέσει κάτω από το κατώφλι πιθανότητας της εφαρμογής, οδηγώντας σε πολλά ψευδή θετικά.

ΚΕΦΑΛΑΙΟ 5

ΠΕΙΡΑΜΑΤΑ

Για τα πειράματά μας χρησιμοποιήσαμε τα 28 βίντεο επιτήρησης ανθρώπων του συνόλου δεδομένων CAVIAR τα οποία περιέχουν, στο σύνολο, 26419 frames. Τα εν λόγω frames έχουν επισημειωθεί χειροκίνητα από την ομάδα του CAVIAR ώστε να δωθεί ένα σημείο αναφοράς για το ποιές STA και LTA συμβαίνουν σε κάθε χρονική στιγμή. Πραγματοποιήσαμε μια μικρή αλλαγή στην επισημείωση του CAVIAR με την εισαγωγή της STA *abrupt* (πρβλ. υποκεφάλαιο 3.3). Ο Πίνακας 5.1 δείχνει τα αποτελέσματα της Crisp-EC και της Prob-EC στο CAVIAR. Αυτά τα αποτελέσματα έχουν προκύψει με τον υπολογισμό ερωτημάτων της μορφής $\text{holdsAt}(LTA = \text{true}, T)$ και τη σύγκριση των αποτελεσμάτων με την επισημείωση των εκάστοτε LTA. Λόγω της υλοποίησης της άρνησης της Prob-EC μέσω των κατηγορημάτων negate_1 και negate_2 , τα αποτελέσματά της είναι παρομοια με αυτά της Crisp-EC.

Ιδιαίτερα αξιοσημείωτη στα πειραματικά αποτελέσματα είναι η χαμηλή τιμή Ακρίβειας (Precision) για την LTA *leaving_object*, η οποία οφείλεται σ' ένα μεγάλο αριθμό από Ψευδή Θετικά (False Positives — FP). Αυτά τα FP δημιουργούνται λόγω της προβληματικής επισημείωσης του CAVIAR για αυτήν τη δραστηριότητα. Για παράδειγμα, στο βίντεο 14, ένας άνθρωπος αφήνει ένα αντικείμενο στο frame 946, προκαλώντας την ενεργοποίηση του κανόνα (3.5) για το *leaving_object*. Ο ίδιος άνθρωπος σηκώνει το αντικείμενο στο frame 1354, όμως η επισημείωση αναφέρει ότι το *leaving_object* σταματά να συμβαίνει στο frame 996. Έτσι καταλήγουμε να έχουμε 358 FP τα οποία θα μπορούσαμε να είχαμε αποφύγει με μια πιο συνεπή επισημείωση του εν λόγω βίντεο. Παρομοίως, στην επισημείωση των βίντεο 17 και 18, ένα μεγάλο κομμάτι των frames λείπει. Το βίντεο 16 περιέχει μια πολύ ενδιαφέρουσα ειδική περίπτωση του *leaving_object*. Στο συγκεκριμένο βίντεο, ένα άνθρωπος αφήνει μια τσάντα δίπλα σε μια καρέκλα και αποχωρεί από τη σκηνή. Μετά από κάποια δευτερόλεπτα εισέρχεται ξανά σε αυτήν και σηκώνει την τσάντα.

LTA	TP	FP	FN	Precision	Recall	F-measure
meeting	3099	1910	525	0.619	0.855	0.718
moving	4008	2162	2264	0.650	0.639	0.644
fighting	531	97	729	0.421	0.845	0.562
leaving object	143	1539	55	0.085	0.722	0.152

Πίνακας 5.1: Αληθή Θετικά (True Positives — TP), Ψευδή Θετικά (False Positives — FP), Ψευδή Αρνητικά (False Negatives — FN), Ακρίβεια (Precision), Ανάκληση (Recall) και μέτρο F (F-measure) για την Crisp-EC και την Prob-EC στο CAVIAR, χωρίς τεχνητό θόρυβο.

Όταν όμως ο άνθρωπος εισέρχεται ξανά στη σκηνή, το σύστημα χαμηλού επιπέδου του αποδίδει έναν καινούριο περιγραφέα (αυτό είναι ιδιαίτερα σύνθηες στο CAVIAR). Αυτό οδηγεί σε διάφορες περιπλοκές. Πρώτον, το αρχικό στιγμιότυπο του *leaving_object* δεν τερματίζεται από τους κανόνες μας όταν ο άνθρωπος που αφήνει το αντικείμενο «εξαφανίζεται» (αποχωρεί δηλαδή από τη σκηνή). Αυτό είναι σκόπιμο από την πλευρά μας: επιλέγουμε να τερματίζουμε το *leaving_object* όταν κάποιος άνθρωπος σηκώνει το αντικείμενο, παρά όταν ο άνθρωπος που το άφησε αποχωρήσει από το πεδίο επιτήρησης. Με αυτόν τον τρόπο δίνουμε έμφαση σε χρονικά σημεία στα οποία ένα αντικείμενο είναι αφύλακτο. Η επισημείωση του CAVIAR, όμως, βλέπει τη δραστηριότητα *leaving_object* από διαφορετική οπτική γωνία, υποθέτοντας ότι τερματίζεται όταν ο άνθρωπος που άφησε το αντικείμενο εγκαταλείψει το οπτικό πεδίο της κάμερας. Αυτή η σημασιολογική διαφορά έχει ως αποτέλεσμα να έχουμε ένα σημαντικό αριθμό από FP, ένα για κάθε frame στο οποίο ο άνθρωπος που άφησε το αντικείμενο δεν είναι στη σκηνή. Όταν ο άνθρωπος ξαναμπει στη σκηνή, το CAVIAR του προσδίδει έναν καινούριο περιγραφέα και συνεχίζει την επισημείωση του *leaving_object* με μια πλειάδα της μορφής $\langle new_person_id, same_object_id \rangle$. Μετά από κάποια frames, ο άνθρωπος (που περιγράφεται με το *new_person_id*) σηκώνει το αντικείμενο, τερματίζοντας το fluent *leaving_object(new_person_id, same_object_id)* το οποίο η Crisp-EC δεν αρχικοποίησε ποτέ. Επομένως, μαζί με ένα σχετικά μεγάλο αριθμό από FP, έχουμε στην προκείμενη περίπτωση και 55 Ψευδή Αρνητικά (False Negatives — FN, πρβλ. Πίνακα 5.1) επειδή η Crisp-EC δεν αναγνωρίζει ποτέ τη νέα δραστηριότητα *leaving_object*.

Σύμφωνα με την επισημείωση του συνόλου δεδομένων CAVIAR, όλες οι STA έχουν πιθανότητα ίση με 1, κάτι που υποδηλώνει ότι το σύστημα αναγνώρισης χαμηλού επιπέδου είναι 100% βέβαιο για την ορθότητα των ανιχνεύσεων. Σε ρεαλιστικές εφαρμογές αναγνώρισης δραστηριοτήτων, αυτή η υπόθεση δε μπορεί να γίνει. Προκειμένου λοιπόν να πειραματιστούμε υπό ρεαλιστικές συνθήκες, προσθέσαμε τεχνητό θόρυβο στο σύνολο δεδομένων μας, με τη μορφή πι-

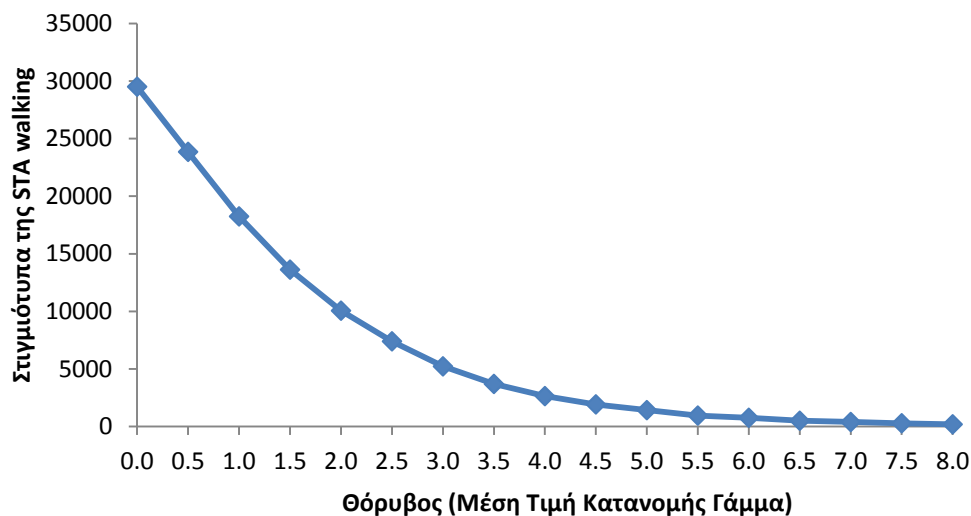
θανοτήτων στις αναγνωρισμένες STA. Για να αναπαραστήσουμε διαφορετικά επίπεδα θορύβου, για τις πιθανότητες αυτές χρησιμοποιήσαμε μια κατανομή Γάμμα [17] με μια ποικιλία τιμών για τη μέση τιμή.

Η πειραματική μας προσέγγιση μπορεί να συνοψιστεί ως ακολούθως:

- Χρησιμοποιούμε δύο διαφορετικές προσεγγίσεις για την πρόσθεση πιθανοτήτων στα γεγονότα εισόδου:
 1. Προσθέτουμε πιθανότητες μόνο στις STA (υπόθεση «ομαλού θορύβου»).
 2. Μαζί με τις STA, προσθέτουμε πιθανότητες και στα σχετικά fluents συντεταγμένων (coord) και κατεύθυνσης (orientation) (υπόθεση «οξέως θορύβου»). Δεν απαιτούμε οι STA να έχουν τις ίδιες πιθανότητες με τα fluents που περιγράφουν συντεταγμένες και κατεύθυνση.

Καταλήγουμε έτσι με δύο διαφορετικές «αλλοιωμένες» εκδόσεις του CAVIAR.

- Δίνουμε αυτές τις εκδόσεις στην Prob-EC και φιλτράρουμε την έξοδό της — η οποία είναι μια σειρά θετικών της μορφής $Prob :: holdsAt(LTA = true, T)$ — κρατώντας μόνο τα θετικά με πιθανότητα από 0.5 και πάνω, τα οποία και θεωρούμε αρκετά ακριβή.
- Φιλτράρουμε και τις δύο εκδόσεις του συνόλου δεδομένων διαγραφοντας όλα τα γεγονότα με πιθανότητες κάτω από 0.5. Κρατάμε τα γεγονότα με πιθανότητες από 0.5 και πάνω, διαγράφοντας την τιμή της πιθανότητάς τους. Κάνουμε με αυτόν τον τρόπο την υπόθεση ότι τα γεγονότα αυτά έχουν αναγνωριστεί με βεβαιότητα. Για την περίπτωση του ομαλού θορύβου, αυτό το βήμα διαγράφει κάποιες STA, ενώ για την περίπτωση του οξέως θορύβου, αυτό το βήμα διαγράφει STA καθώς και fluents σχετικά με συντεταγμένες και κατευθύνσεις.
- Δίνουμε αυτές τις εκδόσεις σαν είσοδο στην Crisp-EC. Με αυτό το βήμα επιχειρούμε να εκτιμήσουμε την επιρροή του περιβαλλοντικού θορύβου στην Crisp-EC, κάνοντας την υπόθεση ότι το σύστημα μπορεί να συμπεριλάβει στο συμπέρασμό μόνο τα γεγονότα τα οποία έχουν αναγνωριστεί με μια σχετική βεβαιότητα, δηλαδή γεγονότα με πιθανότητα τουλάχιστον 0.5.
- Συγκρίνουμε την απόδοση των Crisp-EC και Prob-EC.



Σχήμα 5.1: Στιγμιότυπα της STA *walking* στο CAVIAR ανά μέση τιμή κατανομής Γάμμα. Η μέση τιμή 0.0 αναπαριστά το CAVIAR χωρίς τεχνητό θόρυβο.

Επαναλάβαμε αυτήν την πειραματική διαδικασία 16 φορές, ξεκινώντας από την τιμή 0.5 για τη μέση τιμή της κατανομής Γάμμα και φθάνοντας μέχρι την τιμή 8.0, με βήμα 0.5. Ο θόρυβος που προσθέτουμε είναι τυχαίος· για παράδειγμα, μια STA που διαγράφεται από την είσοδο της Crisp-EC για μέση τιμή ίση με 6.0 μπορεί να μη διαγραφεί από την είσοδο της Crisp-EC για μέση τιμή 6.5. Στην υλοποίησή μας, όσο μεγαλύτερη η μέση τιμή, τόσο πιο μικρές οι πιθανότητες που προσθέτονται στα γεγονότα εισόδου, κάτι που υποδεικνύει ένα «σκληρότερο» επίπεδο θορύβου. Η επίπτωση της σταδιακής αύξησης του μέτρου του θορύβου στις STA του CAVIAR φαίνεται γραφικά στο Σχήμα 5.1, το οποίο δείχνει τον αριθμό των στιγμιοτύπων της STA *walking* με πιθανότητα τουλάχιστον 0.5. Όπως φαίνεται στο σχήμα, ο αριθμός των στιγμιοτύπων του *walking* μειώνεται εκθετικά με την αύξηση του επιπέδου του θορύβου. Όλες οι υπόλοιπες STA ακολουθούν το ίδιο μοτίβο. Στην περίπτωση του «οξέως θορύβου», τα στιγμιότυπα των fluents συντεταγμένων και κατεύθυνσης μειώνονται επίσης εκθετικά.

Παρά το γεγονός πως ο θόρυβος που προσθέτουμε μπορεί να φαίνεται ιδιαίτερα δυνατός (ιδιαίτερα για τις υψηλές τιμές της μέσης τιμής της κατανομής Γάμμα) είναι επίσης ιδιαίτερα ρεαλιστικός. Στην αναγνώριση δραστηριοτήτων από βίντεο, οι STA συνήθως έχουν πολύ χαμηλές πιθανότητες, γιατί η ακρίβεια των συστημάτων αναγνώρισης χαμηλού επιπέδου υποφέρουν σημαντικά από τον περιβαλλοντικό θόρυβο.¹

Η απώλεια των STA από την είσοδο της Crisp-EC μπορεί να έχει ιδιαίτερα σημαντική επίδρα-

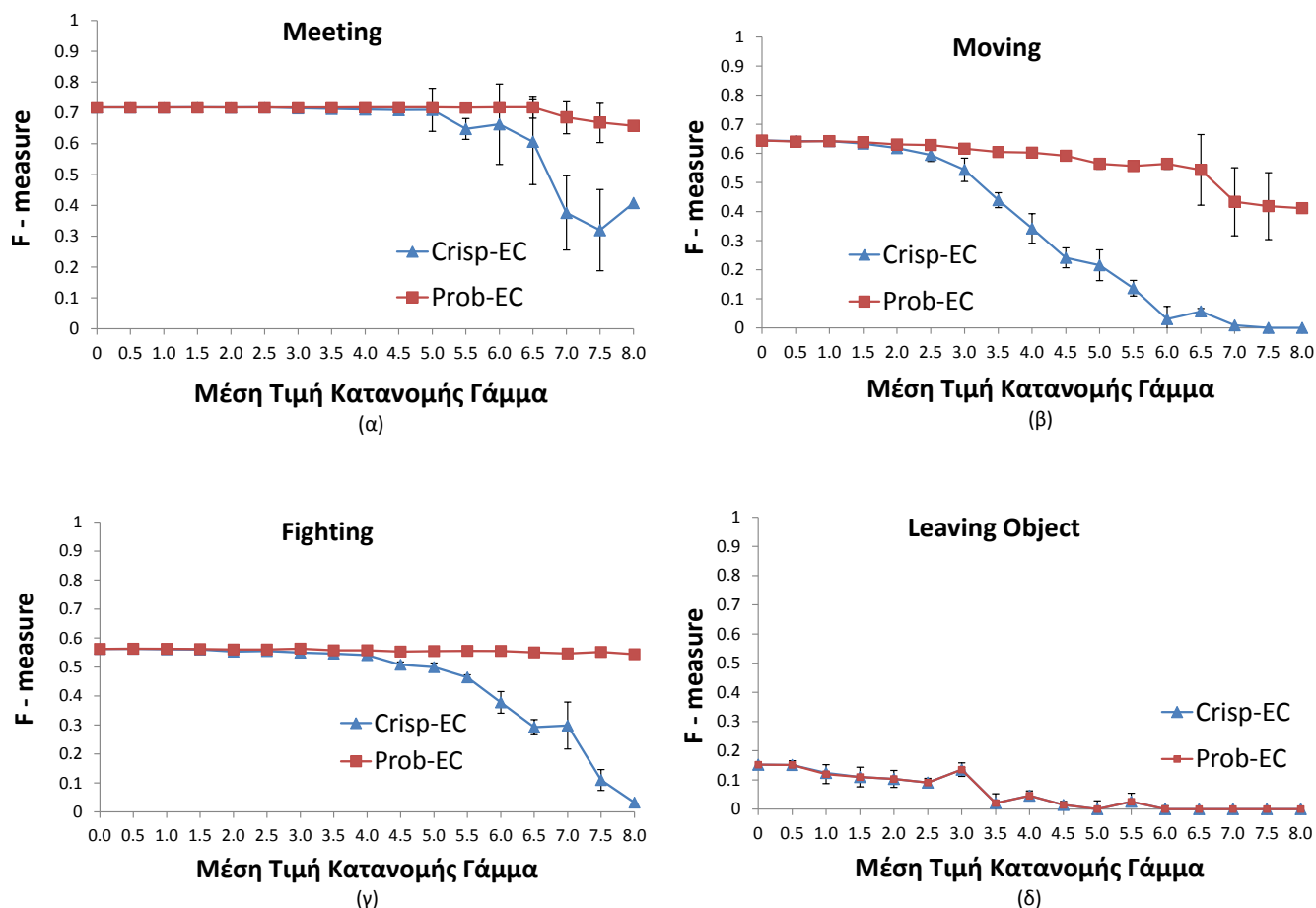
¹Δείτε, για παράδειγμα, το σύνολο δεδομένων Mind's Eye: <http://www.visint.org/datasets.html>

ση στην ακρίβεια της αναγνώρισης δραστηριοτήτων. Ορισμένες STA ή τα σχετιζόμενα με αυτές fluents μπορεί να ορίζουν συνθήκες αρχικοποίησης οι οποίες θα οδηγούσαν στην αναγνώριση μιας δραστηριότητας αν δεν είχαν διαγραφεί από την είσοδο της Crisp-EC. Αυτό μπορεί να οδηγήσει σε πολλά FN. Ακόμη, ορισμένες τερματικές συνθήκες μπορεί να μην αναγνωριστούν, κάτι που μπορεί να οδηγήσει σε πολλά FP. Δεδομένου πάντως του ότι η Crisp-EC κάνει λάθη στο πρωτότυπο σύνολο δεδομένων (δηλαδή στο CAVIAR χωρίς τεχνητό θόρυβο), σε μερικές περιπτώσεις μπορεί ένα FP στο πρωτότυπο CAVIAR να μετατραπεί σε ένα Αληθές Αρνητικό (True Negative — TN) στο «θορυβώδες» CAVIAR — παρομοίως με τα FN. Για να δείξουμε τί εννοούμε μέσω ενός παραδείγματος, ας υποθέσουμε ότι δύο άνθρωποι κινούνται μαζί (*moving*) σε ένα πεζοδρόμιο. Ξαφνικά, πρέπει να απομακρυνθούν ο ένας από τον άλλο για να επιτρέψουν σε ένα άτομο σε αναπηρική καρέκλα να κινηθεί απρόσκοπτα στο πεζοδρόμιο. Αυτό δε σημαίνει ότι σταματούν να επιδεικνύουν την LTA *moving*: απλώς πρέπει για λίγα frames να απομακρυνθούν ο ένας από τον άλλο. Στην Crisp-EC όμως, αυτή η απομάκρυνση θα ενεργοποιούσε την τερματική συνθήκη «απομάκρυνσης» (“walk away”, πρβλ. κανόνα (3.12)), επειδή η απόσταση μεταξύ των δύο ανθρώπων θα υπερέβαινε το προκαθορισμένο όριο των 34 pixels. Η λανθασμένη ενεργοποίηση της τερματικής συνθήκης απομάκρυνσης θα δημιουργήσει κάποια FN στην Crisp-EC. Αν, όμως, η STA *walking* δεν αναγνωριστεί λόγω θορύβου για τα frames στα οποία λαμβάνει χώρα η απομάκρυνση μεταξύ των δύο ανθρώπων, η τερματική συνθήκη δε θα ενεργοποιηθεί ποτέ, άρα τα εν λόγω FN μετατρέπονται σε Αληθή Θετικά (True Positives — TP).

5.1 «Ομαλός» θόρυβος

Στο σχήμα 5.2 συγκρίνουμε την ακρίβεια των Crisp-EC και Prob-EC για όλες τις LTA που ορίσαμε στο υποκεφάλαιο 3.3, κάτω από την υπόθεση του «ομαλού» θορύβου. Για κάθε σημείο (τετράγωνο / τρίγωνο) των διαγραμμάτων, εκτελέσαμε 5 διαφορετικά πειράματα και πήραμε τις εκάστοτε μέσες τιμές του μέτρου F (F-measure) ανά μέση τιμή της κατανομής Γάμμα. Οι κάθετες γραμμές σφάλματος αναπαριστούν την τυπική απόκλιση των τιμών αυτών.

Η ακρίβεια αναγνώρισης της Prob-EC είναι ανώτερη της Crisp-EC για τις LTA *meeting*, *moving* και *fighting*. Για τη δραστηριότητα *meeting* (πρβλ. Σχήμα 5.2(α), η ακρίβεια της Crisp-EC ξεκινά να πέφτει από τη μέση τιμή 5.0 κι έπειτα, καθώς οι STA *active* και *inactive* τείνουν να εξαφανίζονται από την είσοδό της, αφού λαμβάνουν τιμές πιθανότητας κάτω από το 0.5 κατά την παραγωγή του τυχαίου θορύβου. Από την άλλη, η Prob-EC μπορεί να αρχικοποιήσει το *meeting* με κάποια πιθανότητα (θυμηθείτε ότι οι κανόνες (3.9) και (3.10) κωδικοποιούν τις συνθήκες αρχικοποίησης του *meeting*). Μετά από έναν αριθμό από frames, οι συνεχείς αρχικοποιήσεις



Σχήμα 5.2: F-measure της Crisp-EC και της Prob-EC ανά μέση τιμή κατανομής Γάμμα υπό την υπόθεση «ομαλού θορύβου».

του *meeting* οδηγούν σε μια πιθανότητα των ερωτημάτων τύπου *holdsAt* που υπερβαίνει την τιμή του 0.5, επομένως η Prob-EC εν τέλει αναγνωρίζει αυτή τη δραστηριότητα (πρβλ. υποκεφάλαιο 4.2). Οι συνθήκες αρχικοποίησης του *meeting* συμβαίνουν αρκετά συχνά στο CAVIAR. Κατ'επέκταση, το *meeting* τείνει να αναγνωρίζεται με πιθανότητες κοντά στο 1.

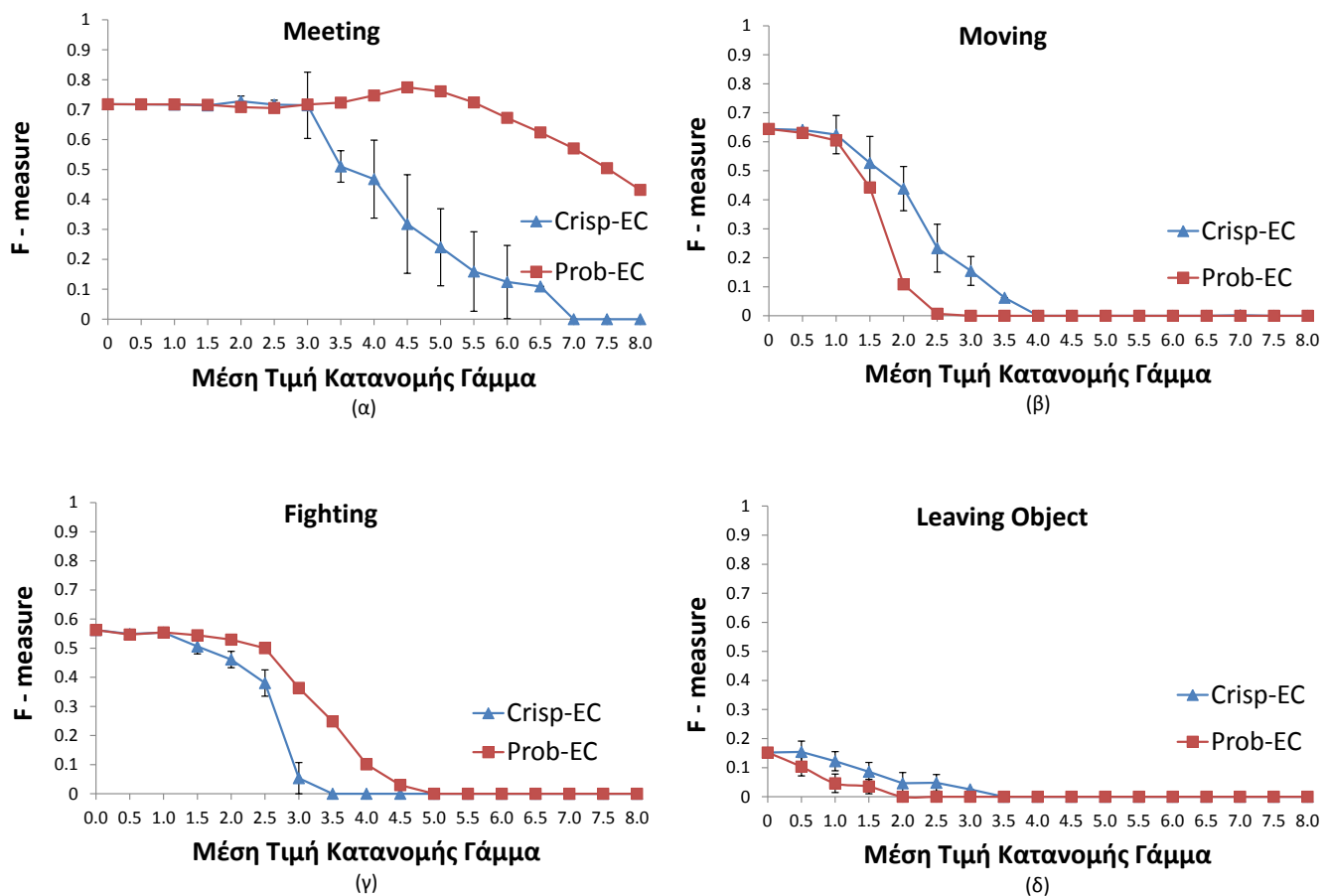
Στην περίπτωση του *moving* (πρβλ. Σχήμα 5.2(β)), η απώλεια πολλαπλών στιγμιotypών της STA *walking* (λόγω θορύβου) προκαλεί μια κατακόρυφη αύξηση των FN για την Crisp-EC (πρβλ. κανόνα (3.11) για την περιγραφή της συνθήκης αρχικοποίησης του *moving*). Η Prob-EC, από την άλλη, χρησιμοποιεί επαναλαμβανόμενες αρχικοποιήσεις προκειμένου να υπερκεράσει το κατώφλι του 0.5. Όπως φαίνεται στο Σχήμα 5.2(β), η ακρίβεια της Crisp-EC αρχίζει να υποχωρεί ταχύτερα για το *moving* απ' ό,τι για το *meeting*. Αυτό συμβαίνει γιατί η συνθήκη αρχικοποίησης του *moving* απαιτεί την ταυτόχρονη ύπαρξη δύο πιθανοτικών γεγονότων υπό τη μορφή δύο

διαφορετικών στιγμιοτύπων του *walking* στο σώμα του κανόνα (3.11). Συνεπώς, η εν λόγω LTA επηρεάζεται από το θόρυβο δις. Οι συνθήκες αρχικοποίησης του *meeting* (πρβλ. κανόνες (3.10) και (3.9)) μπορεί να έχουν κι αυτές δύο διαφορετικά πιθανοτικά γεγονότα στο σώμα τους, όμως αυτό συνήθως δε συμβαίνει — οι συνθήκες που ορίζονται μέσω του κατηγορήματος *negative* συνήθως αναγνωρίζονται με πιθανότητα 1 μέσω της «άρνησης ως αποτυχία».

Παρομοίως με τα *meeting* και *moving*, η Prob-EC είναι πιο ακριβής από την Crisp-EC για το *fighting* (πρβλ. Σχήμα 5.2(γ)). Για υψηλά επίπεδα θορύβου — δηλαδή για μεγάλες τιμές της μέσης τιμής της κατανομής Γάμμα — η Crisp-EC δυσκολεύεται να αρχικοποιήσει την εν λόγω δραστηριότητα (πρβλ. κανόνα (3.13)), ενώ η Prob-EC χρησιμοποιεί επανειλημμένες αρχικοποιήσεις για να περάσει το κατώφλι αναγνώρισης. Η Prob-EC καταφέρνει να περάσει το 0.5 παρά το γεγονός ότι, σε σύγκριση με τις δραστηριότητες *meeting* και *moving*, η μέση χρονική διάρκεια (άρα και το πλήθος διαθέσιμων αρχικοποιήσεων) μιας «πάλης» (*fighting*) είναι μικρότερη.

Η δραστηριότητα *leaving_object* (Σχήμα 5.2(δ)) είναι μια ενδιαφέρουσα ειδική περίπτωση, λόγω της ισχυρής αρχικοποίησής της. Θυμηθείτε από τη σχετική συζήτηση της υποενότητας 4.2 ότι, με την έννοια «ισχυρή αρχικοποίηση», αναφερόμαστε στην περίπτωση μιας LTA η οποία αναγνωρίζεται μέσω μιας μοναδικής συνθήκης αρχικοποίησης και του νόμου της αδράνειας. Στα βίντεο του συνόλου δεδομένων CAVIAR, ένας άνθρωπος αφήνει ένα αντικείμενο αφού περάσουν κάποια frames στα οποία περπατά κουβαλώντας το αντικείμενο. Σε σπάνιες περιπτώσεις είναι εφικτό για ένα από τα δύο συστήματα αναγνώρισης να μην αναγνωρίσει την αρχικοποίηση του *leaving_object* λόγω θορύβου και, λόγω της ισχυρής αρχικοποίησης, να μην μπορέσει να αναγνωρίσει ολόκληρη την LTA, ενώ το άλλο την αναγνωρίζει. Στα πειράματα που αναλύουμε στην παρούσα υποενότητα, τέτοια σενάρια δεν έλαβαν χώρα. Συνεπώς, η Prob-EC και η Crisp-EC είναι εξίσου ακριβείς στην αναγνώριση της LTA αυτής. Όταν το *fluent person* αναγνωρίζεται από την Crisp-EC, αναγνωρίζεται με μία αρκετά μεγάλη πιθανότητα από την Prob-EC και αντιστρόφως. Όπως αναφέραμε στο υποκεφάλαιο 4.2, θα μπορούσαμε να επιτύχουμε καλύτερη απόδοση για την Prob-EC στο *leaving_object* με την εισαγωγή μιας αδύναμα αρχικοποιημένης έκδοσης του *leaving_object*, όμως το να κάνουμε κάτι τέτοιο θα μπορούσε να προκαλέσει προβληματική συμπεριφορά για την Crisp-EC σε άλλα σύνολα δεδομένων.

Οι LTA στο CAVIAR συνήθως τερματίζονται όταν κάποια από τις εμπλεκόμενες οντότητες «εξαφανίζεται» από τη σκηνή ή όταν δύο άνθρωποι απομακρύνονται ο ένας από τον άλλο. Στην τελευταία περίπτωση, η πιθανότητα των συνθηκών τερματισμού εξαρτάται εξ' ολοκλήρου από την πιθανότητα του *walking* (πρβλ. κανόνα (3.12)). Γενικά, οι πιθανοτικές τερματικές συνθήκες έχουν παρόμοια συμπεριφορά με τις πιθανοτικές συνθήκες αρχικοποίησης. Όταν τερματικές



Σχήμα 5.3: F-measure της Crisp-EC και της Prob-EC ανά μέση τιμή κατανομής Γάμμα υπό την υπόθεση «οξέως θορύβου».

συνθήκες με πιθανότητες κάτω από 0.5 ενεργοποιούνται επανειλημμένα, η Prob-EC εν τέλει τερματίζει την LTA, σε αντίθεση με την Crisp-EC, η οποία παράγει FP.

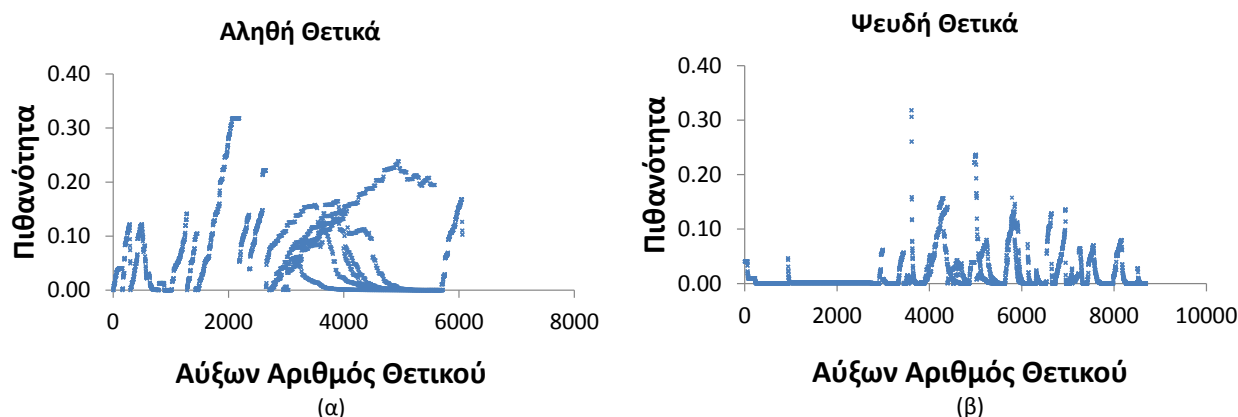
5.2 «Οξύς» θόρυβος

Στο Σχήμα 5.3 φαίνονται τα αποτελέσματά μας κάτω από την υπόθεση του «οξέως θορύβου». Σε σύγκριση με την υπόθεση του «ομαλού» θορύβου, η ακρίβεια και των δύο συστημάτων αναγνώρισης επηρεάζεται σε μεγαλύτερο βαθμό. Η Prob-EC υπερτερεί της Crisp-EC στο *meeting* (πρβλ. Σχήμα 5.3(α) ακόμα και για χαμηλότερα επίπεδα θορύβου (μικρότερες μέσες τιμές) σε σύγκριση με την υπόθεση ομαλού θορύβου. Αυτό συμβαίνει γιατί, πέρα από την απώλεια των STA *active* και *inactive* λόγω θορύβου, η Crisp-EC δυσκολεύεται στο να αποδείξει ότι δύο οντότητες είναι αρκετά κοντά (*close*) για να αρχικοποιήσουν το *meeting*, γιατί στην υπόθεση του οξέως θορύβου

διαγράφουμε και πληροφορία σχετική με τις συντεταγμένες των οντοτήτων στο βίντεο. Η Prob-EC, από την άλλη, χρησιμοποιεί επαναλαμβανόμενες αρχικοποιήσεις του *meeting* και εν τέλει ξεπερνάει το κατώφλι του 0.5.

Σχετικά με τη δραστηριότητα *moving* (Σχήμα 5.3(β)), η ακρίβεια της Crisp-EC πέφτει για μία ακόμα φορά θύμα της γενικευμένης εξαφάνισης του *walking*. Στον οξύ θόρυβο, το πρόβλημα εντείνεται· όπως γίνεται σαφές στο σχήμα, μετά τη μέση τιμή 3.5 η Crisp-EC δε μπορεί να παράξει ούτε ένα θετικό. Αυτό συμβαίνει γιατί μαζί με τα δύο διαφορετικά στιγμιότυπα του *walking* που υπάρχουν στον κανόνα (3.11), η Crisp-EC πρέπει να αντιμετωπίσει την απώλεια πληροφορίας σχετική με συντεταγμένες (που επηρεάζει το fluent *close*) αλλά και για την κατεύθυνση (orientation) των δύο ανθρώπων. Κατ' επέκταση, ακόμα και σε περιπτώσεις που το *walking* αναγνωρίζεται από το σύστημα αναγνώρισης χαμηλού επιπέδου, υπάρχουν ακόμα 4 διαφορετικά στοιχεία, απαραίτητα για την αναγνώριση του *moving*, τα οποία μπορεί να μην υπάρχουν στην είσοδο.

Κάτι ιδιαίτερα ενδιαφέρον στην περίπτωση του *moving* είναι ότι η ακρίβεια της Prob-EC είναι τουλάχιστον χαμηλότερη από της Crisp-EC για όλες τις διαφορετικές μέσες τιμές της κατανομής Γάμμα. Αυτό συμβαίνει γιατί οποτεδήποτε βρίσκεται αντιμέτωπη με μια συνθήκη αρχικοποίησης του *moving*, η Prob-EC πρέπει να υπολογίσει τη σχετική πιθανότητα ως ένα γινόμενο 6 πιθανοτήτων στο σύνολο. (Θυμηθείτε από τον κανόνα (3.7) ότι το fluent *close* ορίζεται μέσω των συντεταγμένων και των δύο οντοτήτων, οπότε συνεισφέρει δις στον αριθμό των πιθανοτικών γεγονότων στον κανόνα (3.11)) για το *moving*.) Συνεπώς, η Prob-EC δυσκολεύεται στην παραγωγή πιθανοτήτων αρκετά μεγάλων ώστε να περάσουν το κατώφλι του 0.5, ακόμα και σε περιπτώσεις όπου υπάρχουν πολλές αρχικοποιήσεις του *moving*. Ακόμα και σε περιπτώσεις υψηλής βεβαιότητας για κάθε ένα από τα στοιχεία του κανόνα (3.11) ξεχωριστά, η υπόθεση ανεξαρτησίας της ProbLog οδηγεί σε πολύ χαμηλές τιμές γινομένου. Για παράδειγμα, ας θεωρήσουμε δύο οντότητες id_1 και id_2 , οι STA των οποίων (στην περίπτωσή μας, το *walking*) καθώς και οι context variables (coordinates, orientation) αναγνωρίζονται όλες με μια πιθανότητα της τάξης του 0.8. Η Crisp-EC, στην περίπτωση αυτή, θα μπορέσει να συμπεράνει ότι το *moving* αρχικοποιείται, γιατί η τιμή της πιθανότητας με την οποία έχουν αναγνωριστεί όλα τα στοιχεία είναι επαρκής. Η Prob-EC, όμως, θα πρέπει να υπολογίσει την πιθανότητα της σχετικής συνθήκης αρχικοποίησης και θα τη βρει ίση με $(0.8)^6 = 0.262$. Επειδή η πιθανότητα αυτή είναι κάτω από 0.5, στην πειραματική μας αξιολόγηση θα πρέπει να απορρίψουμε το εν λόγω θετικό. Το φαινόμενο αυτό απεικονίζεται στο Σχήμα 5.4, όπου σχεδιάζουμε τις πιθανότητες όλων των θετικών για ένα πείραμα πάνω στο *moving* με μέση τιμή 2.5 για την κατανομή Γάμμα. Στο σχήμα αυτό, τα θετικά μας είναι πλήρως αποτιμημένα (grounded) fluents της μορφής $\text{holdsAt}(\text{moving}(ID1, ID2) = \text{true}, T)$.



Σχήμα 5.4: Πιθανότητες αληθών και ψευδών θετικών για ένα πείραμα με την Prob-EC πάνω στο *moving*, για μέση τιμή της κατανομής Γάμμα ίση με 2.5.

Μερικά από αυτά τα θετικά είναι αληθή (TP, πρβλ. Σχήμα 5.4(α)) και μερικά ψευδή (FP, πρβλ. Σχήμα 5.4(β)). Στα εν λόγω διαγράμματα, ο οριζόντιος άξονας αναπαριστά τον αύξοντα αριθμό (index) του θετικού. Ο αριθμός αυτός αντιστοιχεί στη σειρά με την οποία ένα θετικό παράχθηκε· για παράδειγμα, το πρώτο θετικό του πρώτου βίντεο που εξετάστηκε θα έχει αριθμό 1, το πρώτο θετικό του δεύτερου βίντεο μπορεί να έχει αριθμό 300, και ούτω καθ' εξής. Όπως φαίνεται στα διαγράμματα, ούτε ένα θετικό που παράγει η Prob-EC δεν έχει πιθανότητα πάνω από 0.5, παρά την ύπαρξη πολλαπλών αρχικοποιήσεων στην επισημείωση των αντίστοιχων βίντεο. Συνεπώς, το Precision, το Recall, άρα και το F-measure, είναι όλα ίσα με 0.

Σχετικά με το *fighting* (πρβλ. διάγραμμα 5.3(γ)), η Prob-EC υπερτερεί της Crisp-EC για ορισμένες μέσες τιμές της κατανομής Γάμμα. Για υψηλές μέσες τιμές, οι Crisp-EC και Prob-EC είναι εξίσου (αν)ακριβείς. Στην περίπτωση του οξέως θορύβου, υπάρχουν τουλάχιστον τρία πιθανοτικά γεγονότα ανά συνθήκη αρχικοποίησης (πρβλ. κανόνα (4.3)). Λόγω του ότι η δραστηριότητα *fighting* συμβαίνει για λιγότερα frames απ' ό,τι οι δραστηριότητες *meeting* και *moving*, για τις υψηλές μέσες τιμές της κατανομής Γάμμα δεν υπάρχουν επαρκείς αρχικοποιήσεις για να ανεβάσουν την πιθανότητα των θετικών του *fighting* πάνω από το 0.5

Όσον αφορά την LTA *leaving_object* (πρβλ. 5.3(δ)), η Crisp-EC φαίνεται να υπερτερεί κατά ένα μικρό βαθμό της Prob-EC στα χαμηλά επίπεδα του θορύβου. Κάτω από την υπόθεση του οξέως θορύβου, η Prob-EC πρέπει να συμπεριλάβει περισσότερα πιθανοτικά γεγονότα στον υπολογισμό της πιθανότητας των αρχικοποιήσεων του *leaving_object* (αναφερόμαστε στα fluents των συντεταγμένων, που επηρεάζουν την πιθανότητα του *close* στον κανόνα (3.5)). Συνεπώς, δεδομένης και της ισχυρής αρχικοποίησης του *leaving_object*, η Prob-EC τείνει να παράγει πιθανότητες κάτω από 0.5 για τη συγκεκριμένη LTA, ακόμα και όταν τα στιγμιότυπα των επιμέρους στοιχείων

έχουν πιθανότητες υψηλότερες από 0.5. Για μεγαλύτερες μέσες τιμές της κατανομής Γάμμα (οι οποίες και ανταποκρίνονται σε δυνατότερα επίπεδα θορύβου) οι Crisp-EC και Prob-EC υποφέρουν εξίσου από την παρουσία του θορύβου. Αυτό συμβαίνει από τη μία γιατί οι πιθανότητες που παράγει η Prob-EC είναι πολύ μικρές και από την άλλη γιατί τα στοιχεία που απαιτούνται για την αρχικοποίηση της εν λόγω LTA, είτε αυτά είναι συντεταγμένες, είτε είναι η STA *inactive*, είτε είναι στοιχεία απαραίτητα για την απόδειξη του *fluent person* (πρβλ. κανόνα (3.8)) λείπουν από την είσοδο της Crisp-EC.

5.3 Πειραματικά Συμπεράσματα

Σύμφωνα με την πειραματική μας αξιολόγηση, η Prob-EC αποδεικνύεται πιο ακριβής από την Crisp-EC, όταν:

- Η LTA στην οποία εφαρμόζονται τα δύο συστήματα είναι αδύναμα αρχικοποιημένη και
- Η LTA εξαρτάται από ένα μικρό αριθμό πιθανοτικών γεγονότων.

Η Prob-EC υπερτερεί της Crisp-EC υπό τις προαναφερθείσες συνθήκες από μέσα έως μεγάλα επίπεδα θορύβου. Σε χαμηλά επίπεδα θορύβου, η Prob-EC είναι τουλάχιστον όσο ακριβής είναι και η Crisp-EC. Η Prob-EC υπερτερεί της Crisp-EC (υπό τις ίδιες συνθήκες) ακόμα και αν η διάρκεια της LTA είναι σχετικά μικρή, δηλαδή υπάρχει ένας μικρός αριθμός από συνθήκες αρχικοποίησης (όπως για παράδειγμα συμβαίνει με τη δραστηριότητα *fighting*, πρβλ. Σχήματα 5.2(γ) και 5.3(γ)).

Όταν μια LTA εξαρτάται από ένα μεγάλο αριθμό πιθανοτικών στοιχείων, η Prob-EC ενδέχεται να είναι πιο ακριβής από την Crisp-EC, αυτό όμως δε συμβαίνει πάντα. Η ProbLog κάνει υπόθεση ανεξαρτησίας για τα γεγονότα εισόδου, επομένως το γινόμενο αυτών των γεγονότων μπορεί να είναι πολύ χαμηλό, ακόμα και σε περιπτώσεις όπου το κάθε ένα γεγονός έχει μεγάλη πιθανότητα. Όσο μεγαλώνει το πλήθος αυτών των στοιχείων, μεγαλώνει και το πλήθος των αρχικοποιήσεων που χρειάζεται η Prob-EC για να περάσει το κατώφλι πιθανότητας (που στη συγκεκριμένη εργασία είναι ίσο με 0.5). Όταν τα πιθανοτικά στοιχεία είναι πολυάριθμα, σε χαμηλά επίπεδα θορύβου η Crisp-EC είναι τουλάχιστον όσο ακριβής είναι και η Prob-EC, ενώ σε υψηλά επίπεδα θορύβου και τα δύο συστήματα υποφέρουν εξίσου.

Στην περίπτωση της ισχυρής αρχικοποίησης, υπάρχουν περιπτώσεις όπου η Prob-EC μπορεί να τα πάει πολύ καλύτερα από την Crisp-EC και αντιστρόφως. Τέτοιες περιπτώσεις όμως δε συνέβησαν στα πειράματά μας. Στη γενική περίπτωση, η Prob-EC αναμένεται να έχει παρόμοια

απόδοση με την Crisp-EC σε ισχυρά αρχικοποιημένες LTA όταν ο αριθμός των πιθανοτικών γεγονότων είναι μικρός, ενώ όταν ο αριθμός τους είναι μεγάλος, η Crisp-EC είναι πιθανότερο να έχει καλύτερη απόδοση.

Τα πειραματικά μας συμπεράσματα όσον αφορά τον ισχυρό / αδύναμο τερματισμό είναι παρόμοια με αυτά που αφορούν την ισχυρή / αδύναμη αρχικοποίηση. Για παράδειγμα, η Prob-EC υπερτερεί της Crisp-EC σε περιπτώσεις αδύναμα τερματιζόμενων LTA των οποίων οι τερματισμοί εξαρτώνται από ένα μικρό αριθμό από πιθανοτικά γεγονότα. Ο επανειλημμένος τερματισμός επιτρέπει στην Prob-EC να τερματίσει μια LTA την οποία η Crisp-EC αποτυγχάνει να τερματίσει, παράγοντας έτσι FP.

Σαν ένα τελικό συμπέρασμα, επισημαίνουμε ότι, σε όλες τις προαναφερθείσες περιπτώσεις, η Prob-EC είναι τουλάχιστον όσο ακριβής είναι και η Crisp-EC εν τη παρουσία υψηλών επιπέδων θορύβου. Κατ' επέκταση, πιστεύουμε ότι η Prob-EC θα έχει καλύτερη απόδοση σε ρεαλιστικές εφαρμογές αναγνώρισης δραστηριοτήτων, στις οποίες τα συστήματα αναγνώρισης χαμηλού επιπέδου καλούνται να αντιμετωπίσουν ιδιαίτερα ισχυρό θόρυβο.

ΚΕΦΑΛΑΙΟ 6

ΣΥΝΟΨΗ / ΜΕΛΛΟΝΤΙΚΕΣ ΕΡΓΑΣΙΕΣ

6.1 Σύνοψη

Παρουσιάσαμε την Prob-EC, μια διάλεκτο Λογισμού Γεγονότων (Event Calculus) βασισμένη σε πιθανοτικό λογικό προγραμματισμό. Η Προβ-ΕΣ είναι καταλληλή για αναγνώριση γεγονότων σε περιβάλλον θορύβου. Πραγματοποιήσαμε εκτεταμένη πειραματική αξιολόγηση της νέας διαλέκτου σε ένα σύνολο δεδομένων σχεδιασμένο ειδικά για αναγνώρισης ανθρώπινης συμπεριφοράς. Η αξιολόγηση αυτή έδειξε τις συνθήκες κάτω από τις οποίες η Prob-EC αποδίδει καλύτερα από μια ντετερμινιστική διάλεκτο Λογισμού Γεγονότων, όπως η βασισμένη στην Prolog διάλεκτός μας (η Crisp-EC). Η Prob-EC είναι η πρώτη διάλεκτος Λογισμού Γεγονότων η οποία μπορεί να αντιμετωπίσει την αβεβαιότητα στα δεδομένα εισόδου. Ακόμη, η παρούσα δουλειά συνιστά την πρώτη αξιολόγηση του Λογισμού Γεγονότων σε ένα πιθανοτικό περιβάλλον.

6.2 Μελλοντικές εργασίες

Υπάρχουν διάφορες κατευθύνσεις για μελλοντική εργασία. Πρώτον, σκοπεύουμε να πειραματιστούμε με διαφορετικά είδη θορύβου — επεξεργαστείτε, για παράδειγμα, την περίπτωση ενός συστήματος χαμηλού επιπέδου το οποίο «συγχέει» δύο «παρόμοιες» STA, όπως τις *walking* και *running*. Δεύτερον, σκοπεύουμε να επεκτείνουμε την Prob-EC ώστε να περιλαμβάνει χρονικά διαστήματα, όπως περιγράφουμε στο [3]. Κατ' αυτόν τον τρόπο, επιχειρούμε να μπορούμε να απαντήσουμε σε ερωτήματα της μορφής: «Με ποιά πιθανότητα περπατούσαν μαζί (*moving*) ο mike και η sarah (πρβλ. Σχήμα 4.1) για το χρονικό διάστημα $[21, 41]$;» Τρίτον, επιθυμούμε να εμπλουτίσουμε την Prob-EC με ανεπτυγμένες μεθόδους προσωρινής αποθήκευσης (cach-

ing) έτσι ώστε να κάνουμε τη διάλεκτο κατάλληλη για δυναμική αναγνώριση δραστηριοτήτων, όπου οι STA φτάνουν στην είσοδο με μια μεταβαλλόμενη καθυστέρηση και υπάρχει η ανάγκη δυναμικής ανανέωσης των διαστημάτων των ήδη αναγνωρισμένων LTA. Τέλος, σκοπεύουμε να αντιμετωπίσουμε το ζήτημα των ανακριβών ορισμών δραστηριοτήτων. Θα θέλαμε να συνδυάσουμε τη μηχανή συμπερασμού της Prob-EC και της διαλέκτου Λογισμού Γεγονότων βασισμένης σε Markov Logic Networks [38] προκειμένου να παρουσιάσουμε ένα ενοποιημένο σύστημα το οποίο θα μπορεί να αντιμετωπίσει τόσο το θόρυβο στα δεδομένα όσο και ανακριβείς ορισμούς LTA.

ΟΡΟΛΟΓΙΑ

Ξενόγλωσσος Όρος	Ελληνικός Όρος
Event/Activity Recognition	Αναγνώριση Γεγονότων/Δραστηριοτήτων
Event Calculus	Λογισμός Γεγονότων
Short-Term Activity	Βραχυπρόθεσμη Δραστηριότητα
Long-Term Activity	Μακροπρόθεσμη Δραστηριότητα
fluent	«Μεταβαλλόμενο»: Ένα σύμβολο του οποίου η τιμή μεταβάλλεται με το χρόνο.
image processing	Επεξεργασία Εικόνας
Dataset	Σύνολο Δεδομένων
Probabilistic Logic Programming	Πιθανοτικός Λογικός Προγραμματισμός
Sample	Δείγμα
Disjunctive Normal Form	Διαζευκτική Κανονική Μορφή
Grounded fact	Αποτιμημένο γεγονός (χωρίς μεταβλητές)
Query	Επερώτημα
Binary Decision Diagram	Δυαδικό Διάγραμμα Απόφασης
Binary Decision Tree	Δυαδικό Δέντρο Απόφασης
Negation as Failure	Άρνηση ως Αποτυχία
Closed-World Assumption	Υπόθεση Κλειστού Κόσμου
Probabilistic Graphical Models	Πιθανοτικά Γραφικά Μοντέλα
Markov Logic Networks	Λογικά Δίκτυα Μάρκοφ
(Dynamic) Bayesian Networks	(Δυναμικά) Μπασιανά Δίκτυα
Conditional Random Fields	Τυχαία Υπό Συνθήκη Πεδία
Allen's Interval Algebra	Άλγεβρα Διαστημάτων του Allen

Συνέχεια στην επόμενη σελίδα

Συνέχεια από τη προηγούμενη σελίδα

Ξενόγλωσσος Όρος	Ελληνικός Όρος
Event Logic	Λογική Γεγονότων
Inference Engine	Μηχανή Συμπερασμού
Domain-Independent	Καθολικός
Domain-Dependent	Εξαρτώμενος από την εφαρμογή
Frame	“Καρέ” (ενός βίντεο)
Pixel	Εικονοστοιχείο
Coordinates	Συντεταγμένες
Orientation	Κατεύθυνση
Identifier	Περιγραφέας
Abrupt	Απότομος
Initiation Condition	Συνθήκη Αρχικοποίησης
Termination Condition	Συνθήκη Τερματισμού / Τερματική Συνθήκη
Context Variable	Μεταβλητή Περιβάλλοντος
Caching	Προσωρινή Αποθήκευση
Probability Threshold	Κατώφλι Πιθανότητας
Weak Initiation / Termination	Αδύναμη/ος Αρχικοποίηση / Τερματισμός
Strong Initiation / Termination	Ισχυρή/ός Αρχικοποίηση / Τερματισμός
True Positive	Αληθές Θετικό
False Positive	Ψευδές Θετικό
True Negative	Αληθές Αρνητικό
False Negative	Ψευδές Αρνητικό
Occurence / Instance	Στιγμιότυπο
Index	Αύξων Αριθμός ¹
Recall	Ανάκτηση
Precision	Ακρίβεια
F-measure	Μέτρο F
Gamma Distribution	Κατανομή Γάμμα

¹Σε άλλες ερμηνείες, η λέξη index μπορεί να έχει τη σημασία του ευρετηρίου.

ΣΥΝΤΜΗΣΕΙΣ - ΑΡΚΤΙΚΟΛΕΞΑ

Σύντμηση	Πλήρης Ανάπτυξη
STA	Short-Term Activity / Activities
LTA	Long-Term Activity / Activities
EC	Event Calculus
Crisp-EC	Crisp Event Calculus
Prob-EC	Probabilistic Event Calculus
URL	Uniform Resource Locator
CAVIAR	Context Aware Vision using Image-based Active Recognition
SLD	Selective Linear Definite
DNF	Disjunctive Normal Form
BDD	Binary Decision Diagram(s)
MLN	Markov Logic Network(s)
IA	Interval Algebra
DEC-MLN	Discrete Event Calculus based on Markov Logic Networks
MC-SAT	Monte Carlo Satisfiability
TP	True Positive(s)
FP	False Positive(s)
TN	True Negative(s)
FN	False Negative(s)

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] J. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.
- [2] A. Artikis, G. Paliouras, F. Portet, and A. Skarlatidis. Logic-based representation, reasoning and machine learning for event recognition. In *Proceedings of Conference on Distributed Event-Based Systems (DEBS)*, pages 282–293. ACM Press, 2010.
- [3] A. Artikis, M. Sergot, and G. Paliouras. A logic programming approach to activity recognition. *ACM International Workshop on Events in Multimedia*, 2010.
- [4] Rahul Biswas, Sebastian Thrun, and Kikuo Fujimura. Recognizing activities with multiple cues. In Ahmed M. Elgammal, Bodo Rosenhahn, and Reinhard Klette, editors, *Workshop on Human Motion*, volume 4814 of *Lecture Notes in Computer Science*, pages 255–270. Springer, 2007.
- [5] Matthew Brand, Nuria Oliver, and Alex Pentland. Coupled hidden markov models for complex action recognition. In *CVPR*, pages 994–999. IEEE Computer Society, 1997.
- [6] W. Brendel, A. Fern, and S. Todorovic. Probabilistic event logic for interval-based event recognition. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3329–3336. IEEE, 2011.
- [7] Maurice Bruynooghe, Theofrastos Mantadelis, Angelika Kimmig, Bernd Gutmann, Joost Vennekens, Gerda Janssens, and Luc De Raedt. Problog technology for inference in a probabilistic first order logic. In *ECAI 2010 - 19th European Conference on Artificial Intelligence*, 2010.
- [8] R. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, 35(8):677–691, 1986.
- [9] K. Clark. Negation as failure. In H. Gallaire and J. Minker, editors, *Logic and Databases*, pages 293–322. Plenum Press, 1978.
- [10] G. Cugola and A. Margara. Processing flows of information: From data stream to complex event processing. *ACM Computing Surveys*, 2011.
- [11] C. Dousson and P. Le Maigat. Chronicle recognition improvement using temporal focusing and hierarchisation. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 324–329, 2007.
- [12] Daan Fierens, Guy Van den Broeck, Ingo Thon, Bernd Gutmann, and Luc De Raedt. Inference in probabilistic logic programs using weighted cnf's. In *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence (UAI), July 2011*, 2011.

- [13] Matthew L. Ginsberg. Bilattices and modal operators. In *TARK*, pages 273–287, 1990.
- [14] S. Gong and T. Xiang. Recognition of group activities using dynamic probabilistic networks. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 742–749. IEEE, 2003.
- [15] A. Hakeem and M. Shah. Learning, detection and representation of multi-agent events in videos. *Artificial Intelligence*, 171(8–9):586–605, 2007.
- [16] Rim Helaoui, Mathias Niepert, and Heiner Stuckenschmidt. Recognizing interleaved and concurrent activities: A statistical-relational approach. In *PerCom*, pages 1–9. IEEE, 2011.
- [17] R. V. Hogg and A. T. Craig. *Introduction to Mathematical Statistics*. New York: Macmillan, 4th edition, 1978.
- [18] S. Hongeng and R. Nevatia. Large-scale event detection using semi-hidden markov models. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1455–1462. IEEE, 2003.
- [19] Aniruddha Kembhavi, Tom Yeh, and Larry S. Davis. Why did the person cross the road (there)? scene understanding using probabilistic logic models and common sense reasoning. In *ECCV (2)*, pages 693–706, 2010.
- [20] A. Kimmig, B. Demoen, L. De Raedt, V. Santos Costa, and R. Rocha. On the implementation of the probabilistic logic programming language problog. *Theory and Practice of Logic Programming*, 2010.
- [21] D. Kosmopoulos, P. Antonakaki, K. Valasoulis, A. Kesidis, and S. Perantonis. Human behaviour classification using multiple views. In *Proceedings of Hellenic Conference on Artificial Intelligence*, volume 5138. Springer, 2008.
- [22] R. Kowalski and M. Sergot. A logic-based calculus of events. *New Generation Computing*, 4(1):67–96, 1986.
- [23] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Carla E. Brodley and Andrea Pohoreckyj Danyluk, editors, *ICML*, pages 282–289. Morgan Kaufmann, 2001.
- [24] L. Liao, D. Fox, and H. Kautz. Hierarchical conditional random fields for gps-based activity recognition. *Robotics Research*, pages 487–506, 2007.
- [25] Vlad I. Morariu and Larry S. Davis. Multi-agent event recognition in structured scenarios. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.
- [26] K.P. Murphy. *Dynamic Bayesian Networks: representation, inference and learning*. PhD thesis, University of California, 2002.
- [27] P. Natarajan and R. Nevatia. Hierarchical multi-channel hidden semi markov models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI’07)*, pages 2562–2567, 2007.
- [28] A. Paschke. ECA-RuleML: An approach combining ECA rules with temporal interval-based KR event/action logics and transactional update logics. Technical Report 11, Technische Universität München, 2005.

- [29] A. Paschke and M. Bichler. Knowledge representation concepts for automated SLA management. *Decision Support Systems*, 46(1):187–205, 2008.
- [30] A. Paschke and A. Kozlenkov. Rule-based event processing and reaction rules. In *Proceedings of RuleML*, volume LNCS 5858, pages 53–66. Springer, 2009.
- [31] L. Rabiner and B. Juang. An introduction to hidden markov models. *ASSP Magazine, IEEE*, 3(1):4–16, 1986.
- [32] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.
- [33] A. Sadilek and H. Kautz. Location-based reasoning about complex multi-agent behavior. *Journal of Artificial Intelligence Research*, 43:87–133, 2012.
- [34] J. Selman, M. Amer, A. Fern, and S. Todorovic. Pel-cnf: Probabilistic event logic conjunctive normal form for video interpretation. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 680–687. IEEE, 2011.
- [35] V. Shet, D. Harwood, and L. Davis. VidMAP: video monitoring of activity with Prolog. In *Proceedings of International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 224–229. IEEE, 2005.
- [36] V. Shet, J. Neumann, V. Ramesh, and L. Davis. Bilattice-based logical reasoning for human detection. In *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE, 2007.
- [37] J.M. Siskind. Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic. *Journal of Artificial Intelligence Research*, 15:31–90, 2001.
- [38] Anastasios Skarlatidis, Georgios Paliouras, George A. Vouros, and Alexander Artikis. Probabilistic event calculus based on markov logic networks. In *RuleML America*, pages 155–170, 2011.
- [39] Son D. Tran and Larry S. Davis. Event modeling and recognition using markov logic networks. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 610–623, 2008.
- [40] D.L. Vail, M.M. Veloso, and J.D. Lafferty. Conditional random fields for activity recognition. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, page 235. ACM, 2007.
- [41] L. G Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8:410–421, 1979.
- [42] J. Wang and P. Domingos. Hybrid markov logic networks. In *Proceedings of the 23rd national conference on Artificial intelligence*, volume 2, pages 1106–1111, 2008.
- [43] T. Wu, C. Lian, and J.Y. Hsu. Joint recognition of multiple concurrent activities using factorial conditional random fields. In *Proc. 22nd Conf. on Artificial Intelligence (AAAI-2007)*, 2007.