

Ατομική Διπλωματική Εργασία

**ΑΝΑΛΥΣΗ ΑΛΓΟΡΙΘΜΩΝ ΦΗΜΗΣ
ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΤΟ ΕΡΓΑΛΕΙΟ PRISM**

Μαρία Φιλίππου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ιανουάριος 2011

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΑΝΑΛΥΣΗ ΑΛΓΟΡΙΘΜΩΝ ΦΗΜΗΣ ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΤΟ ΕΡΓΑΛΕΙΟ PRISM

Μαρία Φιλίππου

Επιβλέπουσα Καθηγήτρια

Άννα Φιλίππου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Ιανουάριος 2011

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά την καθηγήτρια μου κα Άννα Φιλίππου για την υποστήριξή και την καθοδήγηση που μου παρείχε κατά τη διάρκεια της εκπόνησης της διπλωματικής μου εργασίας. Την ευχαριστώ για τις πολύτιμες συμβουλές και την ψυχολογική υποστήριξη που μου παρείχε σε καθοριστικές στιγμές της πορείας μου μέχρι εδώ. Ευχαριστίες θα ήθελα να εκφράσω προς την οικογένεια μου για την υπομονή και κατανόηση που έδειξαν όλο αυτό το διάστημα αλλά και τους φίλους μου Νικόλα Οδυσσέως και Γεωργία Χριστοδουλίδου για την ηθική υποστήριξη.

Περίληψη

Η παρούσα διπλωματική εργασία ασχολείται με το θέμα «Συστήματα φήμης», έχει σαν στόχο τη μελέτη και ανάλυση αλγορίθμων φήμης για συστήματα ηλεκτρονικού εμπορίου καθώς και την αξιολόγηση τους για εξαγωγή συμπερασμάτων .

Στόχος της μελέτης που γίνεται γύρω από τα συστήματα φήμης είναι η κατανόηση της αξιοπιστίας των οντοτήτων που συμμετέχουν αλλά και η εξεύρεση αλγορίθμου υπολογισμού της φήμης κάποιας οντότητας που να ανταποκρίνεται όσο το δυνατόν καλύτερα σε κάθε περίπτωση με σκοπό τη μέγιστη αξιοπιστία. Με τον όρο φήμη εννοούμε την γνώμη που έχει μια ομάδα οντοτήτων προς μια άλλη οντότητα ή οργανισμό βάσει κάποιων κριτηρίων. Για την καλύτερη κατανόηση των συστημάτων φήμης απαιτείται η ανάπτυξη κάποιων μοντέλων τα οποία με γνώμονα κάποια χαρακτηριστικά και στοιχεία που αφορούν τις οντότητες του συστήματος, αλλά και συμπεριφορά του ιδίου του συστήματος θα μπορεί να εξετάζει και να προβλέπει τη φήμη που θα έχει κάποια οντότητα του συστήματος.

Η εργασία εστιάζεται στη μελέτη διάφορων αλγορίθμων υπολογισμού φήμης, που σε συνδυασμό με τα μοντέλα υλοποίησης πιθανοτήτων (Markov chains) και τη χρήση της χρονικής λογικής θα μας επιτρέψει να αναλύσουμε τους αλγορίθμους και να καταλήξουμε σε συμπεράσματα σχετικά με την συμπεριφορά τους. Σε αυτό θα βοηθήσει και το εργαλείο PRISM το οποίο θα χρησιμοποιηθεί για να υλοποιηθούν οι αλγόριθμοι.

Περιεχόμενα

Κεφάλαιο 1 Εισαγωγή.....	1
1.1 Βασικό Υπόβαθρο	1
1.2 Σκοπός της Διπλωματικής Εργασίας	2
1.3 Δομή της Διπλωματικής Εργασίας	2
Κεφάλαιο 2 Συστήματα φήμης και αλγόριθμοι.....	3
2.1 Συστήματα φήμης	3
2.2 Μοντέλα υλοποίησης συστημάτων φήμης	5
2.3 Μέθοδοι υπολογισμού της φήμης και της εμπιστοσύνης	9
Κεφάλαιο 3 Χρονική Λογική και Εργαλείο PRISM.....	15
3.1 Επαλήθευση συστημάτων	15
3.2 Χρονική Λογική	17
3.3 Αλυσίδες Markov	22
3.4 Εργαλείο PRISM	29
Κεφάλαιο 4 Πειράματα και Ανάλυση Αποτελεσμάτων	36
4.1 Γενικά	36
4.2 Υλοποίηση αλγορίθμων φήμης	37
4.3 Αξιολόγηση και Σύγκριση πειραματικών Προσομοιώσεων	41
Κεφάλαιο 5 Συμπεράσματα και Μελλοντικές επεκτάσεις	59
5.1 Γενικά	59
5.2 Μελλοντικές Επεκτάσεις	60
Βιβλιογραφία	61
Παράρτημά	A-1

Κεφάλαιο 1

Εισαγωγή

1.1 Βασικό Υπόβαθρο	1
1.2 Σκοπός της Διπλωματικής Εργασίας	2
1.3 Δομή της Διπλωματικής Εργασίας	2

1.1 Βασικό Υπόβαθρο.

Το ηλεκτρονικό εμπόριο τα τελευταία χρόνια έχει αναπτυχθεί ραγδαία. Ένας παράγοντας που μπορεί να καθορίσει την επιτυχία του ηλεκτρονικού εμπορίου είναι η εμπιστοσύνη γενικότερα αλλά και η δυνατότητα να δημιουργηθεί η εμπιστοσύνη στον αγοραστή. Σε αυτό το είδος του συστήματος η φήμη των πωλητών είναι καθοριστική για τις μελλοντικές κινήσεις των αγοραστών. Η εμπιστοσύνη μπορεί να δομηθεί με διάφορους τρόπους όπως οι εμπειρίες άλλων από ένα προϊόν, οι διάφορες αναφορές από γνωστούς οργανισμούς κ.α. Έτσι για τη σωστή λειτουργία ενός τέτοιου συστήματος απαιτούνται τρία βασικά στοιχεία, η φήμη, η εμπειρία και οι εισηγήσεις [3].

Τα συστήματα φήμης προσδοκούν να διατηρούν το αίσθημα εμπιστοσύνης, αμοιβαιότητας και μελλοντικής ανταποδοτικότητας μεταξύ των χρηστών του συστήματος εισάγοντας ένα σύστημα αξιολόγησης τόσο των χρηστών όσο και του ιδίου του συστήματος [2]. Έτσι είναι πολύ σημαντικό οι μέθοδοι που θα χρησιμοποιηθούν σε αυτά τα συστήματα να δίνουν αξιόπιστα αποτελέσματα, διότι είναι βασικό η γνώμη που έχει μια ομάδα οντοτήτων ως προς μίαν άλλη οντότητα

να είναι αντικειμενική και έγκυρη ούτως ώστε να μπορεί να αναπτυχθεί εμπιστοσύνη μεταξύ τους.

1.2 Σκοπός της διπλωματικής εργασίας

Στόχος της παρούσας διπλωματικής εργασίας είναι η μελέτη και ανάλυση αλγορίθμων φήμης για συστήματα ηλεκτρονικού εμπορίου καθώς και η αξιολόγηση τους για εξαγωγή συμπερασμάτων.

Η εργασία μου επικεντρώνεται στη μελέτη διάφορων αλγορίθμων υπολογισμού φήμης που σε συνδυασμό με τις Αλυσίδες Markov (Markov chains) και τη χρονική λογική (temporal logic) θα γίνει ανάλυση των αλγορίθμων με σκοπό την εξαγωγή συμπερασμάτων που αφορούν την συμπεριφορά τους. Σε αυτό θα συμβάλει και το εργαλείο PRISM το οποίο θα χρησιμοποιηθεί για την υλοποίηση των αλγορίθμων και τον έλεγχο του μοντέλου.

1.3 Δομή της Διπλωματικής Εργασίας

Στο πρώτο κεφάλαιο θα γίνει μια εισαγωγή στο βασικό υπόβαθρο των συστημάτων φήμης και τον σκοπό της διπλωματικής εργασίας. Στο δεύτερο κεφάλαιο θα παρουσιαστούν διάφοροι αλγόριθμοι ανάπτυξης και υλοποίησης συστημάτων φήμης μέσα από έρευνες που έχουν γίνει στο πεδίο αυτό. Στο τρίτο κεφάλαιο εισάγεται η έννοια της χρονικής λογικής και των ιδιοτήτων που ορίζει μέσω των οποίων γίνεται η επαλήθευση του μοντέλου συστήματος που ορίσαμε. Επίσης σε αυτό το κεφάλαιο παρουσιάζεται και το εργαλείο PRISM το οποίο χρησιμοποιείται για την σύγκριση και αξιολόγηση των διαφορετικών αλγορίθμων που παρουσιάστηκαν στο δεύτερο κεφάλαιο. Στο τέταρτο κεφάλαιο παρουσιάζονται τα πειράματα που έχουν γίνει βάσει διάφορων αλγορίθμων που μελετήθηκαν με την χρήση κάποιων ιδιοτήτων χρονικής λογικής. Στο πέμπτο και τελευταίο κεφάλαιο θα γίνει η ανάλυση των αποτελεσμάτων που προέκυψαν από τη διαδικασία καθώς και η παρουσίαση κάποιων πιο γενικών συμπερασμάτων.

Κεφάλαιο 2

Συστήματα φήμης και αλγόριθμοι

2.1 Συστήματα φήμης	3
2.2 Μοντέλα υλοποίησης συστημάτων φήμης	5
2.3 Μέθοδοι υπολογισμού της φήμης και της εμπιστοσύνης	9

2.1 Συστήματα φήμης

Ένα σύστημα φήμης (reputation system) υπολογίζει και δημοσιεύει αποτελέσματα φήμης για ένα σύνολο αντικειμένων τα οποία ανήκουν σε μια ομάδα / κοινότητα και τα οποία υπολογίζονται με βάση τις εκτιμήσεις (ratings) άλλων οντοτήτων του συστήματος χρησιμοποιώντας συγκεκριμένους αλγορίθμους. Οι εκτιμήσεις και τα αποτελέσματα φήμης χρησιμοποιούνται για τη λήψη αποφάσεων (για την αγορά ή όχι κάποιου προϊόντος κτλ) [1]. Ουσιαστικά ένα σύστημα φήμης συλλέγει και ομαδοποιεί αναφορές που αφορούν την προγενέστερη συμπεριφορά των χρηστών ή συμμετεχόντων στο σύστημα και οι οποίοι σπάνια γνωρίζονται μεταξύ τους. Οι εκτιμήσεις αυτών των ατόμων συμβάλλουν στην καλλιέργεια εμπιστοσύνης μεταξύ τους αλλά και στην αποφυγή εκείνων που δεν εμπνέουν εμπιστοσύνη. Έτσι η χρήση των εκτιμήσεων στα συστήματα φήμης βελτιώνει τα συστήματα και βοηθά τους χρήστες του να αποφασίσουν ποιον θα εμπιστευτούν από τους υπόλοιπους χρήστες [2]. Τα συστήματα φήμης είναι πιο χρήσιμα σε περιβάλλοντα όπου οι χρήστες μπορούν εύκολα να αλληλεπιδράσουν με άγνωστους προς αυτούς χρήστες. Έτσι αναλόγως με το πώς αξιολογείται το σύστημα (τι μέθοδος χρησιμοποιείται) υπάρχουν διάφορα είδη / τύποι συστημάτων.

Όμως τα συστήματα φήμης δε διαθέτουν την απαιτούμενη ασφάλεια (τρόποι εισαγωγής και αποδοχής χρηστών στο σύστημα) με αποτέλεσμα να δέχονται επιθέσεις. Παραδείγματα τέτοιων επιθέσεων είναι τα εξής:

- **Sybil attack** [6]: Ένα σύστημα μπορεί να φαίνεται αξιόπιστο σύμφωνα με τα αποτελέσματα του, αλλά αυτό όμως δεν παύει να είναι μια αυταπάτη, διότι κάποιος μπορεί να δημιουργεί ένα μεγάλο αριθμό από ψευδώνυμα που τον βοηθούν να εισβάλλει σε αυτό και να το επηρεάζει θετικά.
- **Unfair ratings** [6]: Η εξεύρεση τρόπων για την αποφυγή ή μείωση της επιρροής των άδικων θετικών ή αρνητικών εκτιμήσεων είναι ένα σημαντικό πρόβλημα στα συστήματα φήμης όπου οι εκτιμήσεις από άλλους λαμβάνονται υπόψη.
- **Ballot Box Stuffing** [6]: Αυτό που συμβαίνει είναι να παρέχονται περισσότερες εκτιμήσεις από το νόμιμο αριθμό, δίνοντας έτσι άδικες εκτιμήσεις.

Τα συστήματα φήμης υπάρχουν σε πολλούς τομείς του ηλεκτρονικού εμπορίου, από τις αναθεωρήσεις ξενοδοχείων μέχρι στα αποτελέσματα αναζήτησης. Υπάρχουν συστήματα που ταξινομούν τους χρήστες, τα προϊόντα, τα ξενοδοχεία, τις υπηρεσίες και τους ιστοσελίδες. Μερικά παραδείγματα τέτοιων συστημάτων είναι τα ακόλουθα:

- **E-Bay** [6][12]: Είναι ένα δημοφιλές χώρος δημοπρασιών ο οποίος επιτρέπει σε πωλητές να ορίσουν προϊόντα που επιθυμούν να πωλήσουν και σε αγοραστές να αγοράσουν προϊόντα που επιθυμούν. Οποτεδήποτε γίνει συναλλαγή στο eBay, ο αγοραστής και ο πωλητής αξιολογούν ό ένας τον άλλο.
- **Epinions.com** [6]: Επιτρέπει σε άτομα να εκφράζουν τις απόψεις τους για όλων των ειδών τα προϊόντα. Διαθέτει δύο ειδών συστήματα φήμης, ένα για τα προϊόντα και ένα για τους χρήστες.

- **Digg com** [9][10]: Είναι μια ιστοσελίδα φτιαγμένη ώστε να μπορούν οι χρήστες να ανακαλύπτουν και να μοιράζονται ιστορίες, εικόνες ή βίντεο από οπουδήποτε στο διαδίκτυο, καταχωρώντας, ψηφίζοντας ή σχολιάζοντας τα αγαπημένα τους είδη. Πολλά από αυτά μπορούν να καταχωρηθούν κάθε μέρα , αλλά μόνο αυτά με το ψηλότερο σκορ εμφανίζονται στην πρώτη σελίδα κάθε κατηγορίας.
- **Yelp** [11] : είναι ένα διαδικτυακός οδηγός πόλεων ο οποίος δίνει βοήθεια σε αυτούς που αναζητούν κάποιο χώρο για να φάνε, να πιούνε , να ψωνίσουν , να διασκεδάσουν και το οποίο είναι βασισμένο στις απόψεις των ανθρώπων που ζουν σε αυτές τις πόλεις ή έχουν επισκεφτεί αυτούς τους χώρους. Οι χρήστες του Yelp έχουν την δυνατότητα να δώσουν κριτικές για εστιατόρια, μπαρ, κτηματομεσιτικά γραφεία, γιατρούς, αθλητικά κέντρα, αλλά και για κάθε είδος χώρου που υπάρχει σε κάποια πόλη ή στην περιοχή τους

2.2 Μοντέλα υλοποίησης συστημάτων φήμης

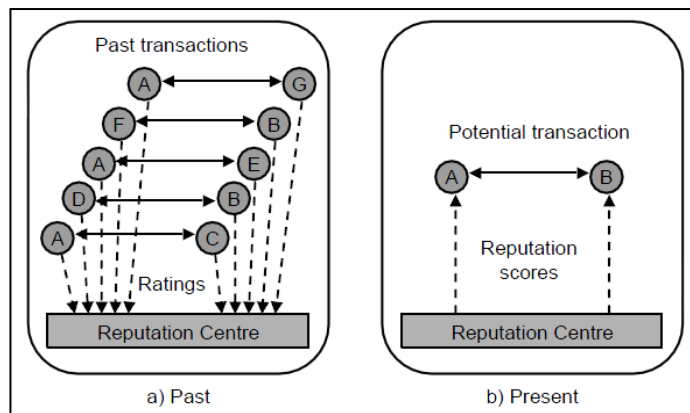
Πολλοί ερευνητές ασχολήθηκαν με την μελέτη των συστημάτων φήμης είτε μέσα από την ανάλυση αλγορίθμων ή την ανάπτυξη μοντέλων. Στόχος τους ήταν η κατανόηση της συμπεριφοράς των οντοτήτων που συμμετέχουν ως προς την αξιολόγηση που κάνουν η μια στην άλλη. Κύριος στόχος όμως ήταν η εύρεση εκείνου του αλγόριθμου που θα επιτυγχάνει την καλύτερη δυνατή αξιοπιστίας και εμπιστοσύνης μεταξύ των δυο οντοτήτων.

Επίσης κατά τη μελέτη μου συνάντησα αρκετά μοντέλα υπολογισμού της εμπιστοσύνης και της φήμης για διαδικτυακές συναλλαγές. Σε ένα από αυτά [11] γίνεται παρουσίαση τέτοιων υφιστάμενων ή προτεινόμενων μοντέλων. Πριν γίνει η παρουσίαση αυτών των μοντέλων θα ήταν καλό να αναφέρουμε τις βασικές αρχιτεκτονικές βάση των οποίων δομείται ένα σύστημα φήμης. Γενικότερα η αρχιτεκτονική του δικτύου ενός τέτοιου συστήματος καθορίζει πως οι αξιολογήσεις

(ratings) και τα σκορ φήμης (reputation scores) μεταδίδονται/μεταβιβάζονται μεταξύ των συμμετεχόντων σε συστήματα φήμης. Οι δύο κύριοι τύποι αρχιτεκτονικών είναι οι **κεντρικοποιημένες (centralized)** [6] και **κατανεμημένες (distributed)** [6].

2.2.1 Κεντρικοποιημένα συστήματα φήμης

Σε αυτά τα συστήματα οι πληροφορίες για την απόδοση ενός συμμετέχοντα καθορίζεται από τις αξιολογήσεις (ratings) από άλλα μέλη του συστήματος τα οποία είχαν άμεση εμπειρία (direct experience) με το συγκεκριμένο συμμετέχοντα. Η κεντρική μονάδα (κέντρο φήμης – reputation centre) μαζεύει όλα τα ratings τα οποία παράγουν το σκορ φήμης για κάθε συμμετέχοντα και δημοσιεύει όλα τα σκορ. Στη συνέχεια οι συμμετέχοντες μπορούν να χρησιμοποιήσουν τα μεταξύ τους σκορ, για παράδειγμα όταν θέλουν να αποφασίσουν αν θα κάνουν ή όχι συναλλαγή με κάποια οντότητα. Η ιδέα είναι ότι οι συναλλαγές (transactions) με συμμετέχοντες με υψηλή καλή φήμη έχουν καλύτερα πιο ευνοϊκά αποτελέσματα από τις συναλλαγές με συμμετέχοντες που έχουν κακή φήμη (Σχήμα 2.1) [6].



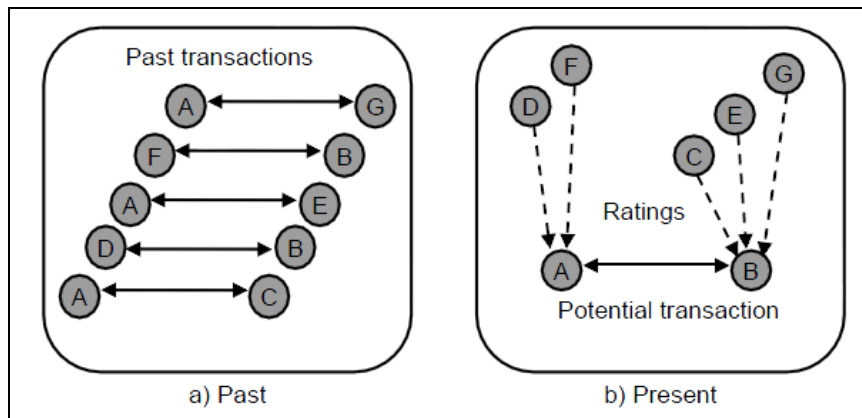
Σχήμα 1.1: Κεντρικοποιημένο σύστημα φήμης, όπου ο A και B είναι χρήστες οι οποίοι είχαν συναλλαγές με άλλους χρήστες στο παρελθόν και οι οποίοι στο παρόν θα κάνουν μία συναλλαγή μεταξύ τους.[6]

Μετά από κάθε συναλλαγή οι πράκτορες/χρήστες δίνουν τις αξιολογήσεις τους (ratings) για τις μεταξύ τους αποδόσεις/συμπεριφορά που αφορούν τη συναλλαγή. Το κέντρο φήμης συγκεντρώνει τα ratings από όλους τους πράκτορες και αναβαθμίζει/ενημερώνει συνεχώς το σκορ φήμης κάθε πράκτορα με βάση τα

ratings που λαμβάνει. Τέλος αυτά τα σκορ χρησιμοποιούνται από τους υπόλοιπους πράκτορες για να αποφασίσουν αν θα κάνουν ή όχι συναλλαγές με τον πράκτορα αυτό.

2.2.2 Κατανεμημένα συστήματα φήμης

Σε ένα κατανεμημένο σύστημα φήμης δεν υπάρχει κέντρο συλλογής/καταχώρησης των αξιολογήσεων(ratings) ή λήψης σκορ φήμης των χρηστών. Αντίθετα μπορεί να υπάρχουν κατανεμημένοι χώροι όπου μπορούν να καταχωρούνται οι αξιολογήσεις ή κάθε συμμετέχοντας καταχωρεί τη γνώμη του για την εμπειρία που είχε άλλους συμμετέχοντες και παρέχει αυτές τις πληροφορίες όταν ζητηθούν από κάποιους άλλους χρήστες. Έτσι όταν κάποιος χρήστης (relying party) επιθυμεί να κάνει συναλλαγές με κάποιο άλλο χρήστη(target party), πρέπει είτε να βρει τους κατανεμημένους χώρους με τις πληροφορίες των χρηστών ή να προσπαθήσει να εξασφαλίσει αξιολογήσεις από όσα περισσότερα μέλη μπορεί τα οποία είχαν άμεση εμπειρία με τον συγκεκριμένο χρήστη (target party). [6]



Σχήμα 2.2: Κατανεμημένο σύστημα φήμης [6]

Ο χρήστης (relying party) υπολογίζει το σκορ φήμης βάσει των αξιολογήσεων που λήφθηκαν. Στην περίπτωση που ο χρήστης (relying party) είχε άμεση επαφή (εμπειρία) με τον άλλο χρήστη (target party), η εμπειρία αυτή μπορεί να ληφθεί σαν

προσωπική πληροφορία και πιθανότατα να μετρά περισσότερο από τις αξιολογήσεις (ratings) που λήφθηκαν.

Πιο κάτω ορίζονται τα μοντέλα τα οποία είναι σημαντικά για την υλοποίηση ενός καταναεμημένου συστήματος φήμης και τα οποία αναφέρονται στην έρευνα που είχαν κάνει οι Michael Kinateder and Kurt Rothermel [3].

2.2.2.1 Μοντέλο συστήματος (System model)

Το μοντέλο αυτό του συστήματος είναι στην ουσία ο σκελετός του. Είναι ένα ιδεατό μοντέλο το οποίο περιγράφει και αναπαριστά ένα σύστημα, το οποίο περιλαμβάνει διάφορες προβολές, όπως σχεδίασης, ανάλυσης, υλοποίησης. Ένα μοντέλο συστήματος πρέπει να το περιγράψει και να αναπαραστήσει όλες τις προβολές. Σε αυτό το μοντέλο οι κόμβοι απεικονίζουν οντότητες οι οποίες έχουν κάποιες ταυτότητες και οι ταυτότητες αυτές μπορούν να πάρουν τιμές από το πραγματικό όνομα του χρήστη (οντότητα) μέχρι ένα από τα ψευδώνυμα του. Η κάθε οντότητα ανήκει σε κάποια κατηγορία και οι σχέσεις των οντοτήτων απεικονίζονται σε ένα μη κατευθυνόμενο γράφο [6].

2.2.2.2 Μοντέλο εμπιστοσύνης (Trust model)

Αποτελείται από ένα σύνολο κατηγοριών με μια τιμή εμπιστοσύνης και ένα αναμενόμενο δείκτη εμπιστοσύνης για κάθε κατηγορία. Η τιμή εμπιστοσύνης κυμαίνεται από 0-1 και δείχνει το βαθμό εμπειρίας μιας οντότητας σε κάποια κατηγορία (0: αντιστοιχεί σε καθόλου εμπειρία, 1: αντιστοιχεί σε μέγιστη εμπιστοσύνη). Έτσι κρατώντας τον αριθμό των εμπειριών αυτών για κάθε κατηγορία βοηθάμε το χρήστη να καταλάβει πόσο έμπιστο είναι κάποιο προϊόν. Αυτό μπορεί να διαφανεί μέσα από τον κατευθυνόμενο γράφο που εισηγούνται οι ερευνητές και ο οποίος αποτελείται από κόμβους (categories) και σταθμισμένες ακμές με τα οποία υποδηλώνεται η εμπιστοσύνη μιας οντότητας Α για μια οντότητα Β μέσα από την εμπειρία που είχαν μεταξύ τους οι δύο οντότητες. [6]

2.2.2.3 Μοντέλο γνώσης (Knowledge model)

Το μοντέλο αυτό χρησιμοποιείται για τη διαφήμιση της πείρας σε κάποιους τομείς μιας οντότητας σε άλλες οντότητες του συστήματος και για την παρακολούθηση της πείρας άλλων. Η πείρα στο μοντέλο γνώσης κάποιου που συμμετέχει στο σύστημα αποτελείται από ένα σύνολο ζευγαριών κατηγορία-τιμή όπου οι κατηγορίες προέρχονται από το μοντέλο εμπιστοσύνης και οι τιμές δείχνουν τον αριθμό των εισηγήσεων που είναι αποθηκευμένες σε αυτή την κατηγορία.

2.3 Μέθοδοι υπολογισμού της φήμης και της εμπιστοσύνης

Τα συστήματα φήμης βασίζονται συνήθως στη δημόσια πληροφορία (public information) δηλαδή στην πληροφορία η οποία προέρχεται από τη γνώμη του κόσμου για ένα αντικείμενο. Με αφορμή αυτό παραθέτονται πιο κάτω διάφορες μεθόδους μέτρησης της φήμης και της εμπιστοσύνης.

2.3.1 Απλό Άθροισμα ή Μέσος όρος των αξιολογήσεων

Ο πιο απλός τρόπος υπολογισμού σκορ φήμης είναι να αθροίσεις το πλήθος των θετικών αξιολογήσεων (positive ratings) και αρνητικών αξιολογήσεων (negative ratings) ξεχωριστά και στη συνέχεια να υπολογίσεις ένα ολικό σκορ αφαιρώντας τα αρνητικά από τα θετικά σκορ (positive – negative). Αυτός ο τρόπος χρησιμοποιείται από το eBay ώστε να υπολογίσει τα σκορ φήμης των χρηστών του. Το eBay ακολουθεί τον εξής αλγόριθμο για να υπολογίζει τη φήμη των χρηστών του. [6]

- **Αλγόριθμος φήμης eBay**

Μετά από κάθε συναλλαγή το eBay ζητά από τον αγοραστή και τον πωλητή να δώσουν feedback ο ένας στον άλλο σχετικά με την συναλλαγή. Με την πάροδο

του χρόνου τα μέλη του eBay αναπτύσσουν ένα feedback profile (ή φήμη) το οποίο βασίζεται στα σχόλια και στις βαθμολογίες που αφήνουν τα υπόλοιπα μέλη μετά από κάποια συναλλαγή. Το feedback score είναι ένα από τα πιο σημαντικά κομμάτια του feedback profile. Οι αγοραστές κτίζουν την φήμη τους βασιζόμενοι σε όλες τις βαθμολογίες και σχόλια τα οποία δίνουν οι πωλητές στο τέλος κάθε συναλλαγής. Το αντίστοιχο γίνεται και για τους πωλητές από τους αγοραστές. Αυτή η πληροφορία είναι διαθέσιμη σε κάθε feedback profile κάθε μέλους και βοηθά τις μελλοντικές αγορές ή πωλήσεις να γίνονται μέσα σε κλίμα εμπιστοσύνης. Έτσι το feedback score υπολογίζεται με βάση τα πιο κάτω:

Για κάθε συναλλαγή, οι αγοραστές και οι πωλητές μπορούν να επιλέξουν να εκτιμήσουν ο ένας τον άλλον αφήνοντας ένα feedback. Οι αγοραστές μπορούν να αφήσουν μια θετική, αρνητική, ή μίαν ουδέτερη εκτίμηση, συν ένα σύντομο σχόλιο. Οι πωλητές μπορούν να αφήσουν μια θετική εκτίμηση και ένα σύντομο σχόλιο. Αυτές οι εκτιμήσεις συχνά ονομάζονται feedback scores. Έτσι τις περισσότερες φορές οι χρήστες παίρνουν:

- +1 βαθμό για κάθε θετική εκτίμηση
- 0 βαθμό για ουδέτερη εκτίμηση
- -1 βαθμό για κάθε αρνητική εκτίμηση

Το feedback score για τους πωλητές υπολογίζεται ως εξής:

- Κάθε φορά που πουλά κάτι, ο αγοραστής αφήνει ένα feedback όπως αναφέρθηκε πιο πάνω.
- Το feedback score θα αυξηθεί κατά 1 μονάδα για κάθε θετικό feedback που παίρνει και θα μειώνεται κατά μια μονάδα για κάθε αρνητικό feedback που παίρνει. Το feedback score μένει το ίδιο αν το feedback είναι ουδέτερο. Αν πωλήσει περισσότερα από ένα προϊόντα στον ίδιο αγοραστή στην ίδια εβδομάδα, τότε το eBay θα συγκρίνει τον συνολικό αριθμό των θετικών feedbacks από αυτό το άτομο, με τον συνολικό αριθμό των αρνητικών feedbacks του. Αν συγκεντρώσει περισσότερα θετικά από αρνητικά feedbacks, τότε το feedback score του θα αυξηθεί κατά 1. Στην περίπτωση

που τα αρνητικά θα είναι περισσότερα από θετικά, το feedback score μειώνεται κατά 1. Τέλος στην περίπτωση που υπάρχει ισότητα μεταξύ των δύο, το feedback score παραμένει το ίδιο.[12]

Το feedback score για τους αγοραστές υπολογίζεται ως εξής [12]:

- Κάθε φορά που αγοράζει κάτι, ο πωλητής αφήνει ένα feedback όπως αναφέρθηκε πιο πάνω. Μπορεί να δώσει θετικό feedback, αλλά μπορεί να δώσει αρνητικό ή ουδέτερο αν ο αγοραστής δεν πληρώσει το προϊόν που έχει αγοράσει.
- Το feedback score αυξάνεται κατά 1 κάθε φορά που παίρνει θετικό feedback ενώ μειώνεται κατά 1 για κάθε αρνητικό feedback που παίρνει. Το feedback score δεν αλλάζει όταν υπάρχει ουδέτερο feedback.
- Αν αγοράσει πολλά προϊόντα από τον ίδιο πωλητή στην ίδια εβδομάδα θα πάρει 1 βαθμό άσχετα από τον αριθμό των προϊόντων που αγόρασε ή τον αριθμό των προϊόντων στα οποία έδωσε feedback.
- Εκτός από το feedback score υπάρχει και positive feedback percentage για κάθε χρήστη και το οποίο υπολογίζεται ως εξής,

$$\frac{\text{positives}}{\text{positives} + \text{negatives}}$$

2.3.2 Μπαγεςιανά Συστήματα

Τα συστήματα αυτά παίρνουν σαν είσοδο δυαδικές αξιολογήσεις (θετικές ή αρνητικές) και βασίζονται σε υπολογιζόμενα σκορ φήμης από στατιστικές αναβαθμίσεις/ ενημερώσεις (statistical updates) της μεθόδου PDF (**Beta Probability Density Functions**). Το ενημερωμένο σκορ φήμης (updated score) υπολογίζεται συγκρίνοντας το προηγούμενο σκορ φήμης με το νέο σκορ. Το σκορ φήμης μπορεί να αναπαρασταθεί στη μορφή beta PDF, ως μια πλειάδα (α,β) όπου

α είναι το σύνολο των θετικών εκτιμήσεων και β το σύνολο των αρνητικών εκτιμήσεων. Η beta PDF ορίζεται για την πιθανότητα p που δηλώνεται από τις τιμές α και β . Η συνάρτηση πυκνότητας (density function) δίνει την πιθανότητα p έχοντας μια συγκεκριμένη τιμή. Λόγω του ότι η beta PDF ορίζεται από την beta ($p|\alpha, \beta$) μπορεί να εκφραστεί χρησιμοποιώντας την συνάρτηση Γ ως εξής:

$$\text{beta}(p|\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1} (1-p)^{\beta-1} \text{ όπου } 0 \leq p \leq 1, \alpha, \beta > 0$$

Η αναμενόμενη πιθανή τιμή της beta δίνεται από τον εξής τύπο:

$$E(p) = \alpha / (\alpha + \beta)$$

Το πλεονέκτημα των συστημάτων αυτών (Bayesian systems) είναι ότι προσφέρουν ένα θεωρητικό τρόπο υπολογισμού των σκορ φήμης, ενώ το μειονέκτημα είναι ότι είναι περίπλοκο να κατανοηθεί από τους περισσότερους ανθρώπους. [6]

2.3.3 Διακριτά μοντέλα εμπιστοσύνης

Οι άνθρωποι συχνά αξιολογούν καλύτερα την απόδοση δίνοντας διακριτούς χαρακτηρισμούς. Έτσι ορίζονται τα διακριτά μοντέλα εμπιστοσύνης (discrete trust models). Για παράδειγμα κάποια οντότητα μπορεί να αναφέρεται ως πολύ αξιόπιστη, αξιόπιστη, αναξιόπιστη, ή πολύ αναξιόπιστη. Ο χρήστης (relying party) μπορεί να δώσει τη δική του γνώμη /αντίληψη για την αξιοπιστία άλλου χρήστη (referring agent) πριν λάβει υπόψη την αναφορά. Οποτεδήποτε ο χρήστης (relying party) x είχε προσωπική εμπειρία με το χρήστη y, αυτό μπορεί να χρησιμοποιηθεί για να καθοριστεί η αξιοπιστία του αναφερόμενου μέλους (referring party). Υποθέτουμε ότι οι αναφορές για το y οι οποίες διαφέρουν από την προσωπική εμπειρία θα δείχνουν κατά πόσο ο αναφερόμενος χρήστης υποτιμά ή υπερτιμά. Αναφορές από το αναφερόμενο χρήστη οι οποίες θα είναι υπερτιμημένες θα υποβιβάζονται. [6]

2.3.4 Μοντέλα πίστης/γνώμης

Η θεωρία γνώμης (belief theory) σχετίζεται με την θεωρία των πιθανοτήτων αλλά υπάρχει μια διαφορά αφού το άθροισμα των πιθανοτήτων των θετικών αποτελεσμάτων δεν αθροίζεται απαραίτητα στο 1 ($\text{sum} \neq 1$) και επίσης η υπολειπόμενη πιθανότητα μεταφράζεται σαν απροσδιόριστη.

Ο Jøsang προτείνει μια μετρική πίστης/εμπιστοσύνης την οποία ονόμασε γνώμη (opinion) και η οποία ορίζεται ως $\omega_x^A = (b, d, u, a)$, το οποίο αναπαριστά την πίστη (b), την δυσπιστία (d), την αβεβαιότητα (u) και μια παράμετρο α η οποία δείχνει πόσο πιστεύει ο A ότι το γεγονός x είναι αληθινό. Οι παράμετροι αυτοί ορίζονται ως εξής [4] [6] [13]:

$$\text{πίστη} = \text{pos} / (\text{pos} + \text{neg} + 2)$$

$$\text{δυσπιστία} = 1 - \text{pos} / (\text{pos} + \text{neg} + 2)$$

$$\text{αβεβαιότητα} = (2.0 / (\text{pos} + \text{neg} + 2))$$

$$\text{base rate } (\alpha) = 1.0 \text{ αν ο χρήστης είναι pre-trusted} \\ 0.5 \text{ αν ο χρήστης δεν είναι pre-trusted}$$

Το μοντέλο αυτό χρησιμοποιεί την άμεση εμπιστοσύνη μεταξύ των οντοτήτων αλλά και τις αναφορές εμπιστοσύνης για κάποιες οντότητες μέσω του μεταβατικού μονοπατιού εμπιστοσύνης. Το μοντέλο αυτό χρησιμοποιείται από τον αλγόριθμο TNA-SL [4][13] οποίος στην συνέχεια της διπλωματικής θα υλοποιηθεί στα πλαίσια της μοντελοποίησης του συστήματος φήμης.

Μια άλλη μέθοδος που χρησιμοποιεί τα μοντέλα αυτά είναι αυτή που προτάθηκε από τους Yu & Singh [6] και η οποία χρησιμοποιεί την θεωρία αυτή (belief theory) για να αναπαραστήσει τα σκορ φήμης. Έτσι το σκορ φήμης Γ για κάποια οντότητα υπολογίζεται χρησιμοποιώντας το αν ο A είναι αξιόπιστος (T_A) ή όχι ($\neg T_A$) αλλά και

τις ξεχωριστές γνώμες για το αν ο A είναι αξιόπιστος ($m(T_A)$) ή όχι ($m(\neg T_A)$). Το σκορ υπολογίζεται ως εξής:

$$\Gamma(A) = m(T_A) - m(\neg T_A), \quad \text{όπου } m(T_A), m(\neg T_A) \in [0,1] \text{ και } \Gamma(A) \in [-1,1]$$

Έτσι γενικότερα το σκορ φήμης καθορίζεται από το αν οι προηγούμενες συμπεριφορές του A κρίνονται αξιόπιστες ή όχι.

2.3.5 Μοντέλα ροής

Τα μοντέλα αυτά υπολογίζουν την εμπιστοσύνη ή τη φήμη με μεταβατικές επαναλήψεις μέσω βρόγχων ή μεγάλων αλυσίδων. Σε κάποια από τα μοντέλα χρησιμοποιείται ένα σταθερό βάρος εμπιστοσύνης/φήμης για όλους τους συμμετέχοντες στο σύστημα και αυτό το βάρος μπορεί να κατανεμηθεί μεταξύ των μελών αυτών. Επίσης οι χρήστες μπορούν μόνο να αυξήσουν την εμπιστοσύνη/φήμη τους σε βάρος των υπολοίπων, όπως γίνεται στα συστήματα του PageRank του Google, το Advogato και άλλα. Γενικότερα σε αυτά τα μοντέλα η φήμη του χρήστη αυξάνεται σαν συνάρτηση των εισερχομένων ροών και μειώνεται σαν συνάρτηση των εξερχομένων ροών [6].

Κεφάλαιο 3

Χρονική Λογική και Εργαλείο PRISM

3.1 Επαλήθευση Συστημάτων	15
3.2 Χρονική Λογική	17
3.3 Αλυσίδες Markov	22
3.4 Εργαλείο PRISM	29

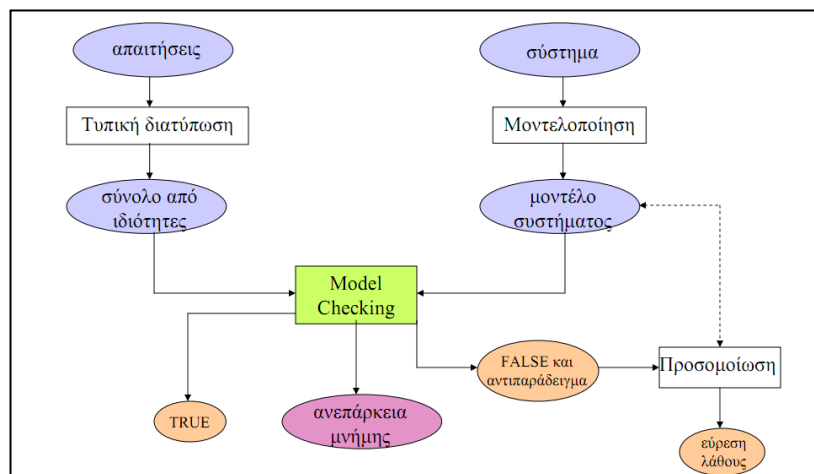
Σε αυτό το κεφάλαιο θα γίνει μια εισαγωγή στην επαλήθευση συστημάτων, θα αναλυθούν η χρονική λογική και οι αλυσίδες Markov που είναι απαραίτητες για την ανάλυση και υλοποίηση των αλγορίθμων στο εργαλείο PRISM. Έπειτα θα γίνει μια περιγραφή του εργαλείου PRISM που θα χρησιμοποιηθεί για την επαλήθευση του συστήματος.

3.1 Επαλήθευση Συστημάτων

Η επαλήθευση συστημάτων ερευνά εάν ένα σύστημα ικανοποιεί τις απαιτήσεις που υπάρχουν από αυτό. Για το λόγο αυτό χρησιμοποιούνται κάποιες τυπικές μέθοδοι που έχουν σαν στόχο την ανάλυση προδιαγραφών, την ανάπτυξη και έλεγχο συστημάτων υλικού και λογισμικού βάση μαθηματικών τεχνικών (Mathematically – based techniques). Οι τυπικές μέθοδοι δίνουν την δυνατότητα επαλήθευσης των συστημάτων κατά τη φάση σχεδιασμού, προσφέρουν αποδοτικές και αυστηρές μεθόδους επαλήθευσης συστημάτων δίνοντας μεγαλύτερη ορθότητα στο σύστημα και τέλος μειώνουν το χρόνο αλλά και το κόστος που απαιτείται για να γίνει η επαλήθευση. Συνήθως οι τυπικές μέθοδοι συνδυάζονται με εργαλεία.

Στην παρούσα διπλωματική εργασία θα χρησιμοποιηθεί η μέθοδος του έλεγχου μοντέλου. Στην προσέγγιση αυτή γίνεται ένας συστηματικός έλεγχος της ιδιότητας φ στις καταστάσεις του μοντέλου και εφαρμόζεται αν το σύστημα παράγει ένα πεπερασμένο μοντέλο συμπεριφοράς. Στη συνέχεια θα γίνει ανάλυση και επεξήγηση της τεχνικής ελέγχου του μοντέλου που θα χρησιμοποιηθεί για την επαλήθευση του μοντέλου που θα αναπτυχθεί.

Η τεχνική ελέγχου του μοντέλου είναι επαναστατική μέθοδος για την αυτοματοποιημένη επαλήθευση συστημάτων. Ελέγχει κατά πόσο ένα μοντέλο ικανοποιεί μια τροπική ιδιότητα (πχ το σύστημα δίνει σωστό αποτέλεσμα; , το σύστημα περιέχει αδιέξοδα;) και βασίζεται σε συστηματικό έλεγχο του συνόλου των καταστάσεων ενός συστήματος. Η τεχνική αυτή προϋποθέτει μια ακριβή και ξεκάθαρη δήλωση των ιδιοτήτων που θα ελεγχτούν. Αυτό γίνεται με την χρήση της χρονικής λογικής η οποία θα αναλυθεί στη συνέχεια αυτού του κεφαλαίου. Το Σχήμα 3.1 δείχνει το πώς λειτουργεί η τεχνική αυτή.



Σχήμα 3.1 : Διάγραμμα της τεχνικής του ελέγχου του μοντέλου [16]

3.2 Χρονική Λογική

Σε αυτή την υποενότητα θα γίνει μια εισαγωγή στη χρονική λογική αλλά και ανάλυση των διάφορων κατηγοριών χρονικής λογικής που χρησιμοποιούνται στην επαλήθευση συστημάτων.

Στη λογική, ο όρος **χρονική λογική** χρησιμοποιείται για να περιγράψει κάθε σύστημα κανόνων και συμβολισμού για την αναπαράσταση και συλλογιστική προτάσεων που χρησιμοποιούν το χρόνο. Κάθε λογική που θεωρεί το χρόνο σαν μια ακολουθία από καταστάσεις είναι χρονική λογική [8].

Ο έλεγχος μοντέλου βασίζεται στη χρονική λογική. Τα μοντέλα της χρονικής λογικής περιλαμβάνουν πολλές καταστάσεις και μια πρόταση μπορεί να είναι αληθής σε κάποιες καταστάσεις και λανθασμένη σε κάποιες άλλες. Στον έλεγχο μοντέλου τα μοντέλα M είναι μεταβατικά συστήματα και οι ιδιότητες φ είναι προτάσεις σε χρονική λογική. Για να εξακριβωθεί κατά πόσο ένα σύστημα ικανοποιεί μια ιδιότητα πρέπει να γίνουν τα εξής [14]:

1. Μοντελοποίηση του συστήματος χρησιμοποιώντας μια περιγραφική γλώσσα ενός εργαλείου ελέγχου μοντέλου (model checker), φτάνοντας σε ένα μοντέλο M .
2. Κωδικοποίηση της ιδιότητας χρησιμοποιώντας την γλώσσα προδιαγραφών του εργαλείου ελέγχου μοντέλου (model checker), καταλήγοντας έτσι σε μια πρόταση χρονικής λογικής φ .
3. Χρήση του εργαλείου ελέγχου μοντέλου με εισόδους τα M και φ .

Η χρονική λογική χωρίζεται σε δύο κατηγορίες αναλόγως του πως χρησιμοποιείται ο χρόνος μέσα στο μοντέλο. Αν ο χρόνος είναι γραμμικός τότε έχουμε την Γραμμική Χρονική Λογική (LTL), ενώ όταν ο χρόνος είναι διακλαδωμένος τότε έχουμε τη Διακλαδωμένη Χρονική Λογική (CTL) [13].

3.2.1 Γραμμική Χρονική Λογική (LTL)

Η Γραμμική Χρονική Λογική (LTL) είναι μια χρονική λογική με συνδέσμους οι οποίοι μας επιτρέπουν να αναφερόμαστε στο μέλλον. Η λογική αυτή μοντελοποιεί το χρόνο σαν μια σειρά καταστάσεων η οποία επεκτείνεται άπειρα στο μέλλον και η οποία συνήθως ονομάζεται και υπολογιστικό μονοπάτι. Η γλώσσα αυτή χρησιμοποιεί ένα σταθερό σύνολο ατομικών προτάσεων που ονομάζεται *Atoms* και το οποίο ισχύει στο σύστημα και ακολουθεί μια συγκεκριμένη σύνταξη που έχει ως εξής [14]:

$$\begin{aligned} \phi ::= & T \mid \perp \mid p \mid (\neg\phi) \mid (\phi \wedge \phi) \mid (\phi \vee \phi) \mid \\ & (\phi \rightarrow \phi) \mid (X\phi) \mid (F\phi) \mid (G\phi) \mid (\phi \cup \phi) \mid (\phi W \phi) \mid (\phi R \phi) \end{aligned}$$

Η LTL αποτελείται από ένα σύνολο από προτασιακές μεταβλητές P_1, P_2 , τους γνωστούς λογικούς συνδέσμους $\neg, \vee, \wedge, \rightarrow$ και τους ακόλουθους χρονικούς τροπικούς τελεστές [14]:

- **X φ** για το "επόμενο" (*neXt*) : η φ πρέπει να ισχύει στην επόμενη κατάσταση
- **G φ** για το "πάντα" (*Globally*) : η φ πρέπει να ισχύει σε όλο το υπόλοιπο μονοπάτι
- **F φ** για το "τελικά" (*Future*) : η φ πρέπει τελικά να ισχύει (κάπου στο υπόλοιπο μονοπάτι).
- **ψ U φ** για το "μέχρι" (*Until*) : η ψ πρέπει να ισχύει *τουλάχιστον* μέχρι τη φ, η οποία ισχύει στην τρέχουσα θέση ή σε κάποια θέση στο μέλλον.
- **ψ R φ** για το "απελευθέρωση" (*Release*) : η φ πρέπει να είναι αληθής μέχρι και τη στιγμή που η ψ γίνεται αληθής για πρώτη φορά - αν η ψ δε γίνει ποτέ αληθής, η φ πρέπει να μείνει αληθής για πάντα.
- **ψ W φ** για "αδύναμο - μέχρι" (*Weak until*) : σημασιολογία του οποίου μοιάζει με αυτήν του τελεστή **U** αλλά η συνθήκη τερματισμού δε χρειάζεται

να ισχύει (όπως στην *απελευθέρωση*), δηλαδή δεν απαιτεί η φ να ικανοποιείται κατά μήκος του μονοπατιού όπως ισχύει για το U .

Οι πρώτοι τρεις τελεστές είναι μοναδιαίοι, δηλαδή η $X\varphi$ είναι καλά ορισμένη πρόταση όταν η φ είναι μια καλά ορισμένη πρόταση. Οι τρεις τελευταίοι τελεστές είναι δυαδικοί, δηλαδή η $\varphi U \psi$ είναι καλά ορισμένη πρόταση όταν η φ και η ψ είναι καλά ορισμένες.

Σύστημα Μεταβάσεων

Ένα σύστημα μεταβάσεων $M = (S, \rightarrow, L)$ είναι ένα σύνολο καταστάσεων S ακολουθούμενο από μια μεταβατική σχέση $\rightarrow$ (μια δυαδική σχέση πάνω στο S), έτσι ώστε κάθε $s \in S$ έχει κάποιο $s' \in S$, με $s \rightarrow s'$ και μια συνάρτηση μετονομασίας $L: S \rightarrow P(\text{Atoms})$. Η σχέση $\rightarrow$, ορίζει πως το σύστημα μπορεί να κινηθεί από κατάσταση σε κατάσταση ενώ το $P(\text{Atoms})$ ορίζεται σαν το δυναμοσύνολο των ατομικών προτάσεων (Atoms) [14].

Σημασιολογία της LTL

Έστω ότι $\sigma = (S, \rightarrow, L)$ είναι σύστημα μεταβάσεων για την LTL, $s \in S$ και φ είναι ένας LTL τύπο. Τότε η σημασιολογική σχέση $(\sigma, i \models \varphi)$ ορίζεται όπως φαίνεται πιο κάτω:

1. $\pi \models \text{true}$
2. $\pi \models \text{false}$
3. $\pi \models p$ αν $p \in L(s_1)$
4. $\pi \models \neg\varphi$ αν $\pi \not\models \varphi$
5. $\pi \models \varphi_1 \wedge \varphi_2$ αν $\pi \models \varphi_1$ και $\pi \models \varphi_2$
6. $\pi \models \varphi_1 \vee \varphi_2$ αν $\pi \models \varphi_1$ ή $\pi \models \varphi_2$
7. $\pi \models \varphi_1 \rightarrow \varphi_2$ αν $\pi \models \varphi_2$ οποτεδήποτε το $\pi \models \varphi_1$
8. $\pi \models X\varphi$ αν $\pi^2 \models \varphi$

9. $\pi \models \mathbf{G}\phi$ ανν για όλα τα $i \geq 1$, $\pi^i \models \phi$
10. $\pi \models \mathbf{F}\phi$ ανν υπάρχει κάποιο $i \geq 1$ έτσι ώστε $\pi^i \models \phi$
11. $\pi \models \phi \cup \psi$ ανν υπάρχει κάποιο $i \geq 1$ έτσι ώστε $\pi^i \models \psi$ και για όλα τα $j = 1, \dots, i-1$ να ισχύει $\pi^j \models \phi$
12. $\pi \models \phi \mathbf{W} \psi$ ανν υπάρχει είτε κάποιο $i \geq 1$ έτσι ώστε $\pi^i \models \psi$ και για όλα τα $j = 1, \dots, i-1$ ισχύει $\pi^j \models \phi$, ή για όλα τα $k \geq 1$ ισχύει $\pi^k \models \phi$
13. $\pi \models \phi \mathbf{R} \psi$ ανν υπάρχει είτε κάποιο $i \geq 1$ έτσι ώστε $\pi^i \models \phi$ και για όλα τα $j = 1, \dots, i-1$ ισχύει $\pi^j \models \psi$, ή για όλα τα $k \geq 1$ ισχύει $\pi^k \models \psi$

3.2.2 Διακλαδωμένη Χρονική Λογική (CTL)

Η λογική υπολογιστικού δένδρου (Computation tree logic, CTL) ή διακλαδωμένη χρονική λογική είναι μια λογική διακλαδιζόμενου χρόνου (branching - time), το οποίο σημαίνει ότι το μοντέλο της για το χρόνο είναι μια δενδρική δομή, στην οποία το μέλλον δεν έχει οριστεί αλλά υπάρχουν διαφορετικά μονοπάτια σε αυτό και κάθε ένα από αυτά τα μονοπάτια μπορεί να είναι ένα μονοπάτι που θα πραγματοποιηθεί. Χρησιμοποιείται στην τυπική επαλήθευση λογισμικού ή υλικού, συνήθως από λογισμικό που ονομάζεται ελεγκτής μοντέλων (model checker). Το λογισμικό αυτό προσδιορίζει αν το λογισμικό ή το υλικό που ελέγχεται έχει κάποιες συγκεκριμένες ιδιότητες ασφάλειας (safety) ή ζωτικότητας (liveness).

Σαν κατασκευαστικά στοιχεία χρησιμοποιεί ατομικές προτάσεις για τη διατύπωση στοιχειωδών ισχυρισμών, που σε δεδομένη κατάσταση έχουν συγκεκριμένη τιμή αλήθειας (true ή false) [13]. Όπως και στην LTL υπάρχει ένα σταθερό σύνολο ατομικών προτάσεων $p_1, p_2, \dots$ και έχει την ακόλουθη σύνταξη:

$$\begin{aligned}\Phi, \Psi &::= p \mid \neg \Phi \mid \Phi \mid \Phi \vee \Psi \mid \mathbf{A} \phi \mid \mathbf{E} \phi \\ \phi &::= \mathbf{X} \Phi \mid \Phi \cup \Psi\end{aligned}$$

Η CTL στη συνέχεια συνδυάζει τις ατομικές προτάσεις σε τύπους χρησιμοποιώντας λογικούς τελεστές και χρονικές λογικές. Οι λογικοί τελεστές είναι οι συνήθειες: $\neg, \vee, \wedge, \Rightarrow$ και $\Leftrightarrow$. Οι τύποι της CTL μπορούν επιπλέον να χρησιμοποιούν μαζί με αυτούς τις λογικές σταθερές true (αληθές) και false (ψευδές). Οι χρονικοί τελεστές που ορίζονται είναι οι ακόλουθοι:

- **A φ** - Όλα (All): η φ πρέπει να ισχύει σε όλα τα μονοπάτια που αρχίζουν από την τρέχουσα κατάσταση.
- **E φ** - Υπάρχει (Exists): υπάρχει τουλάχιστον ένα μονοπάτι που αρχίζει από την τρέχουσα κατάσταση, στο οποίο να ισχύει η φ.
- **X φ** - Επόμενο (Next): η φ πρέπει να ισχύει στην επόμενη κατάσταση (ο τελεστής αυτός συνήθως συμβολίζεται και σαν **N** αντί για **X**).
- **φ U ψ** - Μέχρι (Until): η φ πρέπει να ισχύει *τουλάχιστον* μέχρι κάποια θέση στην οποία να ισχύει η ψ. Αυτό σημαίνει ότι η ψ θα επαληθευτεί στο μέλλον.

Στη CTL, οι χρονικοί τελεστές μπορούν να αναμιγνύονται ελεύθερα. Στη CTL, ο τελεστής πρέπει πάντα να ομαδοποιείται σε δύο θέσεις: ένας τελεστής μονοπατιού που ακολουθείται από έναν τελεστή κατάστασης. Οι τύποι της CTL ερμηνεύονται με συστήματα μεταβάσεων (Transition Systems) όπως ορίζεται πιο κάτω. Ένα σύστημα μεταβάσεων $M = (S, \rightarrow, L)$, έχει ένα σύνολο μεταβάσεων S , μια σχέση $\rightarrow$ και μια συνάρτηση L η οποία συνδέει κάθε κατάσταση με τις ατομικές προτάσεις τις οποίες ικανοποιεί, δηλώνοντας πως το σύστημα μπορεί να κινηθεί από μια κατάσταση σε μια άλλη κατάσταση. Έστω ότι $M = (S, \rightarrow, L)$ είναι σύστημα μεταβάσεων για την CTL, $s \in S$ και φ είναι ένας CTL τύπο. Τότε η σημασιολογική σχέση $(M, s \models \varphi)$ ορίζεται όπως φαίνεται στον πίνακα 3.2.[14] [8]

Ορίζουμε τη σχέση $\models$ όπου,

$M, s \models \varphi$ αν και μόνο εάν η ιδιότητα φ ικανοποιείται στην κατάσταση s του M ως εξής:

$M, s \models p$ αν και μόνο εάν $p \in L(s)$, $L(s)$: το σύνολο των ατομικών προτάσεων που ισχύουν στην κατάσταση s

$M, s \models \neg \varphi$ αν και μόνο εάν δεν ισχύει ότι $M, s \models \varphi$

$M, s \models \varphi \vee \psi$ αν και μόνο εάν $(M, s \models \varphi)$ ή $(M, s \models \psi)$

$M, s \models E \varphi$ αν και μόνο εάν $M, w \models \varphi$ για κάποιο μονοπάτι w που ξεκινά από την s

$M, s \models A \varphi$ αν και μόνο εάν $M, w \models \varphi$ για κάθε μονοπάτι w που ξεκινά από την s

Έστω το μονοπάτι $w = s_0 s_1 s_2 \dots$ του συστήματος μεταβάσεων M και η ιδιότητα φ .

Ορίζουμε τη σχέση $\models$ όπου

$M, w \models \varphi$ αν και μόνο αν η ιδιότητα φ ικανοποιείται στο μονοπάτι w του συστήματος μεταβάσεων M ως εξής:

$M, w \models X \Phi$ αν και μόνο αν $M, w[1] \models \Phi$

$M, w \models \Phi \cup \Psi$ αν και μόνο αν υπάρχει $j \geq 0$ τέτοιο ώστε $M, w[j] \models \Psi$ και για κάθε $0 \leq k < j$, $M, w[k] \models \Phi$

Όπου αν $w = s_0 s_1 s_2 \dots$, $w[k]$ είναι η κατάσταση s_k

Πίνακας 3.2: Σύνολο ιδιοτήτων της γλώσσας CTL [8]

3.3 Αλυσίδες Markov

Οι αλυσίδες Markov χρησιμοποιήθηκαν για πρώτη φορά από τον Andrey Markov το 1906. Αποτελούν ένα σημαντικό μαθηματικό εργαλείο για στατιστική ανάλυση και μοντελοποίηση. Στη θεωρία πιθανοτήτων οι αλυσίδες Markov είναι μοντέλα απεικόνισης καταστάσεων ενός συστήματος σε δεδομένες χρονικές στιγμές (ή περιόδους). Το κύριο χαρακτηριστικό στις αλυσίδες Markov είναι ότι δεν έχουν μνήμη δηλαδή δεν λαμβάνεται υπόψη το παρελθόν του συστήματος για τη μελλοντική εξέλιξη του, αλλά μόνο τον παρόν. Η πιθανότητα μετάβασης του συστήματος από την κατάσταση i στην κατάσταση j συμβολίζεται με το P_{ij} . Έτσι διαμορφώνεται ο πίνακας πιθανοτήτων μετάβασης από μια χρονική στιγμή στην

αμέσως επόμενη, όπου το P_{ij} είναι το στοιχείο της i γραμμής και της j στήλης του πίνακα P . [16]

Οι αλυσίδες Markov διακρίνονται σε 3 κατηγορίες [15][16]:

- Discrete-time Markov chain (DTMC)
- Continuous-time Markov chain (CTMC)
- Markov decision process (MDP)

Discrete-time Markov chain (DTMC)

Η αλυσίδα **Markov** είναι μια τυχαιοποιημένη διαδικασία με την ιδιότητα ότι η επόμενη κατάσταση εξαρτάται μόνο από την υφιστάμενη κατάσταση. Μια διακριτή χρονική τυχαιοποιημένη διαδικασία (DTMC), είναι ένα σύστημα το οποίο σε μια συγκεκριμένη κατάσταση σε κάθε βήμα αλλάζει κατάσταση μεταξύ των βημάτων με τυχαιοποιημένο τρόπο, μετακινείται δηλαδή από μια κατάσταση σε μια άλλη, με βάση κάποια πιθανότητα. [15][16]. Ο ορισμός της DTMC φαίνεται πιο κάτω.

Ορισμός DTMC

Μια αλυσίδα DTMC D είναι μια πλειάδα $(S, \bar{s}, P, L)$, όπου

- S είναι ένα πεπερασμένο σύνολο καταστάσεων
- $\bar{s} \in S$ είναι η αρχική κατάσταση
- $P: S \times S \rightarrow [0,1]$ είναι ο πίνακας μεταβάσεων πιθανοτήτων όπου

$$\sum_{s' \in S} P_{(s,s')} = 1 \text{ για όλες τις } s \in S;$$

- $L: S \rightarrow 2^{AP}$ είναι μια δηλωμένη συνάρτηση η οποία αποδίδει σε κάθε κατάσταση $s \in S$, το σύνολο $L(s)$ των ατομικών προτάσεων που ισχύουν στην κατάσταση.

Η Markov ιδιότητα ορίζει ότι η συνεχόμενη κατανομή πιθανότητας για το σύστημα για το επόμενο βήμα το οποίο δίνεται στην παρούσα κατάσταση, εξαρτάται μόνο από την παρούσα κατάσταση του συστήματος και όχι από προηγούμενες καταστάσεις του συστήματος. Έτσι, αφού το σύστημα αλλάζει πιθανοτικά, είναι αδύνατον να υπάρξει πρόβλεψη της ακριβούς κατάστασης του συστήματος μελλοντικά.[16]

Markov Decision Process (MDP)

Παρέχει μια μαθηματική δομή για την μοντελοποίηση λήψης αποφάσεων σε καταστάσεις όπου το αποτέλεσμα είναι μερικώς τυχαία και μερικώς κάτω από τον έλεγχο αυτού που παίρνει τις αποφάσεις. Σε κάθε χρονικό βήμα, η διεργασία είναι σε κάποια κατάσταση s , και ο λαμβάνων την απόφαση μπορεί να επιλέξει κάποια ενέργεια a η οποία είναι διαθέσιμη στην κατάσταση s . Η διεργασία ανταποκρίνεται στο επόμενο χρονικό βήμα με το να μετακινηθεί τυχαία σε μια νέα κατάσταση S' και δίνει στον λαμβάνων την απόφαση ένα αποτέλεσμα / ανταμοιβή $R_a(s, s')$. Η πιθανότητα η διεργασία να επιλέξει το s' σαν την νέα της κατάσταση επηρεάζεται από την επιλεγόμενη ενέργεια και δίνεται από $P_a(s, S')$. Έτσι η επόμενη κατάσταση s' εξαρτάται από την υφιστάμενη κατάσταση s και την ενέργεια a που λαμβάνει ο λήπτης αποφάσεων. Όμως τα S και a είναι υπό όρους ανεξάρτητα από όλες τις προηγούμενες καταστάσεις και ενέργειες. Κατέχει την ιδιότητα Markov. [15][16]

Στη συνέχεια γίνεται ανάλυση της πιθανοτικής χρονικής λογικής η οποία χρησιμοποιείται από την DTMC και με την οποία θα γίνει η ανάλυση του μοντέλου.

Πιθανοτική Χρονική Λογική (PCTL)

Η πιθανοτική Χρονική Λογική (PCTL: Probabilistic Computation Tree Logic) είναι μια επέκταση της CTL που αφορά τις πιθανότητες και η οποία σου επιτρέπει να θέσεις όρια στην πιθανότητα όλων των εκτελέσεων να ικανοποιούν συγκεκριμένη ιδιότητα.[15]

Η σύνταξη της PCTL ορίζεται με ατομικές προτάσεις, χρησιμοποιώντας τις καθολικές συνδέσεις (Boolean connectives) και πως τελεστές «Next» και «Until» που έχει ως εξής: [15]

$$\Phi ::= \text{true} \mid a \mid \neg\phi \mid \phi \wedge \phi \mid P_{\sim p}[\phi]$$

$$\phi ::= x\phi \mid \phi U_{\leq k} \phi \quad \text{όπου}$$

Το a είναι μια ατομική πρόταση $\sim \in \{<, \leq, >, \geq\}$, $p \in [0,1]$ και $k \in \mathbb{N} \cup \{\infty\}$

Το Φ αντιπροσωπεύει φόρμουλα κατάστασης ενώ το ϕ ορίζει φόρμουλα μονοπατιού.

Για να ορίσουμε μια ιδιότητα σε DTMC συνήθως χρησιμοποιούμε έκφραση κατάστασης. Μια κατάσταση S του D ικανοποιεί την ιδιότητα $P_{\sim p}[\Phi]$ εάν η πιθανότητα να ικανοποιείται το ϕ ξεκινά από την κατάσταση s και ακολουθώντας ολόκληρο το μονοπάτι ικανοποιεί τη συνθήκη $\sim p$. Για παράδειγμα η ιδιότητα,

$$P < 0.01 \quad [X \ y=1]$$

είναι true σε κάποια κατάσταση εάν η πιθανότητα να έχουμε $y=1$ στην επόμενη κατάσταση είναι μικρότερη από 0.01.

Ο όρος $X\phi$ είναι αληθής εάν το Φ ικανοποιείται στη επόμενη κατάσταση και ο όρος $\phi U_{\leq k} \psi$ είναι αληθής εάν το ψ ικανοποιείται σε k χρονικά βήματα και το Φ είναι αληθής μετά από εκείνο το σημείο.

Για να έχουμε αποτέλεσμα true για την έκφραση $P_{\sim p}[\Phi]$ σε κάποια κατάσταση θα πρέπει για όλα τα βήματα s του DTMC να υπολογιστεί η πιθανότητα $\text{Prob}^D(s, \varphi)$ και στη συνέχεια να συγκριθούν αυτές οι τιμές με την τιμή συνθήκης ($\sim p$) της έκφρασης. Η έκφραση $\text{Prob}^D(s, \varphi)$ μας δίνει την πιθανότητα στη κατάσταση s να ικανοποιείται η ιδιότητα φ . [15]

Πιο κάτω δίνουμε μερικά παραδείγματα χρήσης αυτής της έκφρασης για να καταλάβουμε πως λειτουργεί. [15]

- $P_{\leq 0.1} [\neg \text{done } U_{\leq 10} \text{ fault}] = \text{Prob}^D(\neg \text{done } U_{\leq 10} \text{ fault}) < 0.1$

Η πιθανότητα να προκύψει λάθος πριν ολοκληρωθεί το πρωτόκολλο μέσα σε 10 χρονικές στιγμές είναι αυστηρά μικρότερη από 0.1

- $P_{\geq 0.9} [X (\neg \text{try} \vee \text{succ})] = \text{Prob}^D(X \neg \text{try} \vee \text{succ}) \geq 0.9$

Η πιθανότητα στην επόμενη κατάσταση να μην περιμένει για να στείλει ένα μήνυμα ή να αποστείλει με επιτυχία το μήνυμα είναι τουλάχιστο 0.9

- $P_{> 0.98} [\text{true } U^{\leq 2} \text{ succ}] = P_{> 0.98} [\Diamond^{\leq 2} \text{ succ}]$

Η πιθανότητα να αποστέλλονται μήνυμα με επιτυχία στις πρώτες 2 χρονικές στιγμές είναι 0.98.

Η έκφραση $P_{\sim p} [f]$ εκτός από Boolean τιμή (true, false) επιστρέφει και αριθμητική τιμή εάν το $\sim p$ αντικατασταθεί με $=?$. Άρα η έκφραση $P_{\sim p}[\varphi]$ εκφράζεται σαν $P=?[\varphi]$ και επιστρέφει την πιθανότητα να ικανοποιείται η έκφραση φ . Στην συνέχεια θα δούμε πως ορίζονται οι steady-state συμπεριφορές αλλά και ανταμοιβές (rewards) στην PCTL.[15]

Steady-state –συμπεριφορά

Η ιδιότητα αυτή χρησιμοποιείται όταν θέλουμε να κάνουμε αναφορά σε μια σταθερή κατάσταση συμπεριφοράς του μοντέλου σε μακράς διάρκειας προσομοίωση. Η steady-state πιθανότητα ορίζει την πιθανότητα σε μακρά εκτέλεση μιας DTMC ή CTMC και ενώ να ξεκίνησα από μια κατάσταση s , τώρα να βρίσκομαι σε μια κατάσταση s' . Η σύνταξη της έχει ως εξής:[15]

$$S_{-p} [\Phi]$$

Η ιδιότητα αυτή ορίζει ότι σε μακράς διάρκειας εκτέλεση η πιθανότητα το σύστημα να βρίσκεται σε μια κατάσταση η οποία ικανοποιεί τη συνθήκη ϕ φτάνει το όριο $\sim p$.

Παράδειγμα

- $S_{<0.02} [\text{insufficient_routers}]$

Η μακράς εκτέλεσης πιθανότητα (long run probability) να υπάρχει ένας ανεπαρκής αριθμός δρομολογητών οι οποίοι να είναι λειτουργικοί είναι μικρότεροι από 0.02.[15]

Ανταμοιβές (Rewards)

Η ανταμοιβή μπορεί να χρησιμοποιηθεί για να δώσει επιπρόσθετες πληροφορίες για ένα σύστημα DTMC, όπως για παράδειγμα την κατανάλωση ενέργειας, των αριθμό των πακέτων που έχουν σταλεί ή των αριθμό των χαμένων αιτήσεων. Για ένα DTMC υπάρχουν δυο τύποι ανταμοιβών, η ανταμοιβή κατάστασης (state reward) και ανταμοιβή μετάβασης (transition reward). Η ανταμοιβή κατάστασης $r(s)$ είναι η ανταμοιβή η οποία αναλογεί σε μια κατάσταση s ανά ένα χρονικό βήμα ενώ η ανταμοιβή μετάβασης $i(s,s')$ δίνεται κάθε φορά που υπάρχει μετάβαση από το s στο s' . Στην PCLT υπάρχουν οι εξής τύποι ιδιοτήτων ανταμοιβών:[15]

- **Προσεγγιστικές Ιδιότητες Κόστους "reachability reward"**

$$R_{\sim r} [F \Phi]$$

Η ιδιότητα $R_{\sim r} [F \Phi]$ είναι αληθής αν από μια κατάσταση s η αναμενόμενη ανταμοιβή που υπολογίζεται πριν προκύψει κάποια κατάσταση που να ικανοποιεί την Φ φτάνει το όριο $\sim r$. [15]

- **Συσσωρευτικές Ιδιότητες Κόστους "cumulative reward"**

$$R_{\sim r} [C^{\leq k}]$$

Μια κατάσταση s ικανοποιεί την ιδιότητα $R_{\sim r} [C^{\leq k}]$, αν από την κατάσταση s , η αναμενόμενη ανταμοιβή μετά από k βήματα ικανοποιεί τη συνθήκη $\sim r$. [15]

- **Στιγμιαίες Ιδιότητες Κόστους "instantaneous reward"**

$$R_{\sim r} [I=t]$$

Η ιδιότητα είναι αληθής αν από κάποια κατάσταση s , η αναμενόμενη ανταμοιβή στην χρονική στιγμή t ικανοποιεί το $\sim r$. [15]

Παραδείγματα

- $R=? [C \leq 24]$

Η αναμενόμενη κατανάλωση ενέργειας κατά τη διάρκεια των πρώτων 24 ωρών της εργασίας.

- $R=? [I = 100]$

Ο αναμενόμενος αριθμός πακέτων που περιμένουν να παραδοθούν μετά από 100 χρονικές στιγμές.

- $R=? [F \text{ elected}]$

Ο αναμενόμενος αριθμός βημάτων που απαιτούνται για να εκλεγεί κάποιος πρόεδρος.

3.4 Εργαλείο PRISM

Το PRISM είναι ένα εργαλείο ελέγχου μοντέλου που βασίζεται στις πιθανότητες και με το οποίο γίνεται επίσημη μοντελοποίηση και ανάλυση συστημάτων που εμφανίζουν τυχαία ή πιθανοτική συμπεριφορά. Ο πιθανοτικός έλεγχος μοντέλου είναι μια επίσημη τεχνική επαλήθευσης η οποία βασίζεται στην κατασκευή ενός ακριβή μαθηματικού μοντέλου ενός συστήματος το οποίο θα αναλυθεί. Το PRISM υποστηρίζει τρεις τύπους πιθανοτικών μοντέλων οι οποίοι αναλύθηκαν στην προηγούμενη ενότητα και είναι οι εξής [7]:

- Discrete-time Markov chains (DTMCs)
- Continuous-Time Markov Chains (CTMCs)
- Markov Decision Processes (MDPs).

Επιπρόσθετα με αυτούς τους τύπους επεκτείνει τα μοντέλα με το κόστος (cost) και τα οφέλη (rewards). Οι ιδιότητες του μοντέλου είναι γραμμένες στη γλώσσα προδιαγραφών ιδιοτήτων του PRISM (PRISM property specification language) η οποία βασίζεται στην χρονική λογική. Επίσης η γλώσσα αυτή ενσωματώνει πολλές διαφορετικές πιθανοτικές χρονικές λογικές όπως: PCTL (probabilistic computation tree logic), CSL (continuous stochastic logic), LTL (linear time logic) κ.α. [7]

Ένα μοντέλο στο PRISM αποτελείται από *modules*, τα οποία είναι διεργασίες οι οποίες μπορούν και αλληλεπιδρούν μεταξύ τους. Ένα *module* τοπικές μεταβλητές, οι οποίες σε κάθε δεδομένη στιγμή αποτελούν την κατάσταση του *module* και ένα σύνολο εντολών (commands), το οποίο περιγράφει τη συμπεριφορά του *module*.

Οι εντολές έχουν την εξής μορφή:

$$[] \ g \rightarrow \text{prob}_1: \text{update}_1 + \text{prob}_2: \text{update}_2 + \dots + \text{prob}_n: \text{update}_n$$

, όπου το g (*guard*) είναι ένα κατηγορημα πάνω από όλες τις μεταβλητές του μοντέλου, το update_n είναι η μετάβαση που θα κάνει το *module* αν το

κατηγορήμα g είναι αληθές και οι εκφράσεις $prob_n$ ορίζουν την πιθανότητα με την οποία θα πραγματοποιηθεί η μετάβαση. [7]

Για ένα μοντέλο πιθανοτήτων που έχει υλοποιηθεί με τη γλώσσα PRISM, δομείται σαν την παράλληλη σύνθεση των modules του. Σε κάθε κατάσταση του μοντέλου υπάρχει ένα πλήθος εντολών που μπορεί να πραγματοποιηθεί. Η εντολές αυτές όπως έχει προαναφερθεί μπορεί να ανήκουν σε οποιαδήποτε από τα modules του μοντέλου. Η επιλογή ανάμεσα στο ποιά εντολή θα εκτελεστεί εξαρτάται από τον τύπο του μοντέλου. Για ένα MDP μοντέλο η επιλογή είναι μη ντετερμινιστική, ενώ για ένα DTMC μοντέλο η επιλογή είναι βασισμένη σε πιθανότητες. Κάθε εντολή επιλέγεται με ίση πιθανότητα. Πιο κάτω φαίνεται ένα παράδειγμα το οποίο περιγράφει τα βασικά στοιχεία της γλώσσας του PRISM.

```
// Example 1
// Two process mutual exclusion

mdp

module M1

    x : [0..2] init 0;

    [] x=0 -> 0.8:(x'=0) + 0.2:(x'=1);
    [] x=1 & y!=2 -> (x'=2);
    [] x=2 -> 0.5:(x'=2) + 0.5:(x'=0);

endmodule

module M2

    y : [0..2] init 0;

    [] y=0 -> 0.8:(y'=0) + 0.2:(y'=1);
    [] y=1 & x!=2 -> (y'=2);
    [] y=2 -> 0.5:(y'=2) + 0.5:(y'=0);

endmodule
```

Πίνακας 3.3: Παράδειγμα κώδικα στο εργαλείο PRISM [7]

Οι ιδιότητες στο PRISM ορίζονται χρησιμοποιώντας μια γλώσσα που βασίζεται στη χρονική λογική PCTL. Η βασική σύνταξη της γλώσσας που θα χρησιμοποιηθεί για να εκφράσει τις ιδιότητες που χρειαζόμαστε ορίζονται στο Σχήμα 3.4, όπου :

- $\langle \text{exp} \rangle$ είναι μια έκφραση η οποία αποτιμάται σε true.
- $F \leq T \text{ prop}$ είναι η ιδιότητα “Eventually” ενός μονοπατιού, η οποία είναι αληθής για ένα μονοπάτι αν το prop ξαφνικά γίνει αληθής σε κάποιο σημείο του μονοπατιού μέσα στο χρόνο T.

```

<prop> ::= <expr> | <prop> & <prop>
pathdrop ::= F<=T prop | prop

```

Σχήμα 3.4: Σύνταξη της γλώσσας ιδιοτήτων του PRISM [7]

Ο κύριος τύπος ιδιότητας που χρησιμοποιείται για την επαλήθευση του μοντέλου είναι η ποσοτική (quantitative property) η οποία υπολογίζει την πραγματική πιθανότητα να εντοπιστεί κάποια συμπεριφορά του μοντέλου. Αυτή η ιδιότητα χρησιμοποιείται με τους τελεστές P και S.

Τελεστής P

Ο τελεστής P δίνει την πιθανότητα να προκύψει κάποιο συμβάν και ορίζεται ως εξής:

$$P=? [\text{pathdrop}]$$

Παραδείγματα ιδιοτήτων

- $P \geq 0.4[X \text{ delivered}]$

Η πιθανότητα το μήνυμα να έχει παραδοθεί μετά από μια χρονική στιγμή είναι τουλάχιστο 0.4.

Μερικά παραδείγματα χρήσης αυτής της ιδιότητας είναι τα εξής:

- $P=? [\text{!proc2_terminate} \cup \text{proc1_terminate}]$

Μας δίνει την πιθανότητα η διεργασία 1 να τερματίσει πριν τερματίσει η διεργασία?

- $P=? [true \ U \leq T \ (message_lost \ \rangle \ 10)]$

Μας δίνει την πιθανότητα να έχουν χαθεί περισσότερα από 10 μηνύματα μέχρι τη χρονική στιγμή T.

Τελεστής S

Ο τελεστής S δίνει την μακροπρόθεσμη πιθανότητα να ικανοποιηθεί μια κατάσταση / συνθήκη (steady-state συμπεριφορά) και ορίζεται ως εξής:

$$S_{bound} [pathdrop]$$

Η ιδιότητα $S_{bound} [prop]$, είναι αληθής για μια κατάσταση s ενός CTMC, όταν ξεκινώντας από την s, η πιθανότητα στην μακρά πορεία της προσομοίωσης να βρεθεί σε μια κατάσταση που ικανοποιεί την ιδιότητα prop, είναι εντός του ορίου bound.

Στην συνέχεια παρουσιάζονται οι ιδιότητες που μπορούν να οριστούν βάση του τελεστή S. [7]

Steady-state –συμπεριφορά

Η ιδιότητα αυτή χρησιμοποιείται όταν θέλουμε να κάνουμε αναφορά σε μια σταθερή κατάσταση συμπεριφοράς του μοντέλου σε μακράς διάρκειας προσομοίωση.

Για παράδειγμα: $S < 0.05 [queue_size / max_size > 0.75]$, που ερμηνεύεται ως την πιθανότητα (long-run probability) της ουράς να είναι άνω του 75% γεμάτη, είναι μικρότερη από 0.05. [7]

Ορισμός Ιδιοτήτων Κόστους και Ανταμοιβών

Στη γλώσσα ιδιοτήτων του PRISM οι ιδιότητες που σχετίζονται με ανταμοιβές ορίζονται χρησιμοποιώντας των τελεστή R . Ο τελεστής R μπορεί να χρησιμοποιηθεί είτε σε ένα Boolean αίτημα $S_{\text{bound}} [\text{rewardprop}]$ είτε σε ένα υπολογίσιμο αίτημα $S_{\text{bound}} [\text{rewardprop}]$. Η ιδιότητα $S_{\text{bound}} [\text{rewardprop}]$ είναι αληθής σε μια κατάσταση του μοντέλου αν η αναμενόμενη ανταμοιβή που σχετίζεται με την έκφραση rewardprop του μοντέλου ξεκινώντας από αυτή την κατάσταση ικανοποιεί τη συνθήκη bound . Η ιδιότητα $S_{\text{query}} [\text{rewardprop}]$ επιστρέφει την πραγματική αναμενόμενη ανταμοιβή βάση της έκφρασης rewardprop . [7]

Στο PRISM υπάρχουν τέσσερεις τύποι ιδιοτήτων reward [7] :

- Προσεγγιστικές Ιδιότητες Κόστους "reachability reward": $F \text{ prop}$
- Συσσωρευτικές Ιδιότητες Κόστους "cumulative reward" : $C \leq t$
- Στιγμιαίες Ιδιότητες Κόστους "instantaneous reward" : $I = t$
- Ιδιότητες Κόστους "steady- state reward" : S .

Προσεγγιστικές Ιδιότητες Κόστους

Για ένα MDP και ένα DTMC μοντέλο, η συνολική ανταμοιβή που μαζεύεται είναι ίσο με το άθροισμα των ανταμοιβών των καταστάσεων του μονοπατιού, συν τις ανταμοιβές των μεταβάσεων μεταξύ των καταστάσεων του μονοπατιού. Η ιδιότητα F που σχετίζεται με reward ($F \text{ prop}$) αναφέρεται σε ένα κόστος που έχει μαζευτεί κατά μήκος ενός μονοπατιού μέχρι να ικανοποιείται η συνθήκη prop . Τα rewards μαζεύονται όπως περιγράφηκε παραπάνω. Τα κόστη της κατάστασης prop που ικανοποιεί το ερώτημα PCTL, δεν υπολογίζονται στο τελικό αποτέλεσμα. Στην περίπτωση που δεν υπάρχει μονοπάτι που φτάνει στην κατάσταση prop το αποτέλεσμα είναι άπειρο.

Παρακάτω φαίνεται ένα τυπικό ερώτημα της ιδιότητας F . $R \leq 9.5$ [F $z=2$] ,το οποίο είναι αληθές αν το κόστος που μαζεύεται κατά μήκος του μονοπατιού έως την κατάσταση $z=2$ είναι μικρότερο του 9.5. [7]

Συσσωρευτικές Ιδιότητες Κόστους

Η ιδιότητα τύπου συσσωρευτικού κόστους επίσης αντιστοιχούν ένα κόστος με κάθε μονοπάτι του μοντέλου αλλά μέχρι ένα ορισμένο χρονικό σημείο. Η ιδιότητα $C \leq t$ αντιστοιχεί στο κόστος που έχει μαζευτεί κατά μήκος ενός μονοπατιού μέχρι t μονάδες χρόνου να έχουν περάσει. Για μοντέλα τύπου DTMC και MDP το t πρέπει να είναι ένας ακέραιος. Για μοντέλα τύπου CTMC το t μπορεί να είναι double.[7]

Στιγμιαίες Ιδιότητες Κόστους

Η ιδιότητα αυτού του τύπου αναφέρεται σε κάποια ανταμοιβή του μοντέλου σε συγκεκριμένη χρονική στιγμή. Η ιδιότητα $I=t$ συσχετίζει ένα μονοπάτι με την ανταμοιβή μιας κατάστασης του μονοπατιού όταν έχουν περάσει ακριβώς t μονάδες χρόνου. Για μοντέλα τύπου DTMC και MDP το t πρέπει να είναι ένας ακέραιος. Για μοντέλα τύπου CTMC το t μπορεί να είναι double.

Η ιδιότητα: $R < 4.4$ [$I=100$], σημαίνει ότι μετά από 100 μονάδες χρόνου ακριβώς, η ανταμοιβή της κατάστασης που βρίσκεται το μοντέλο πρέπει να είναι μικρότερο του 4.4. [7]

Ιδιότητες Κόστους

Σε αντίθεση με τους προηγούμενους τρεις τύπους ιδιοτήτων δεν σχετίζονται με μονοπάτια, αλλά με κόστος στη διάρκεια της εκτέλεσης του προγράμματος.

Ένα τυπικό παράδειγμα αυτής της ιδιότητας είναι το $R < 0.7$ [S], που αν υποθεθεί πως τα κόστη του μοντέλου σχετίζονται με κατανάλωση ενέργειας, τότε η

παραπάνω ιδιότητα σημαίνει η κατανάλωση ενέργειας στην εκτέλεση του προγράμματος να μην ξεπερνά τις 0.7 μονάδες. [7]

Κεφάλαιο 4

Πειράματα και ανάλυση αποτελεσμάτων

4.1 Γενικά	36
4.2 Υλοποίηση αλγορίθμων φήμης στο PRISM	37
4.3 Αξιολόγηση και σύγκριση πειραματικών προσομοιώσεων	41

4.1 Γενικά

Η διαχείριση φήμης υιοθετείται στα συστήματα ηλεκτρονικού εμπορίου για να βοηθήσει τους αγοραστές του συστήματος να αποφασίσουν κατά πόσο είναι ασφαλές να αγοράσουν κάποιο προϊόν από ένα πωλητή. Η φήμη κάποιου πωλητή παίζει καθοριστικό ρόλο στην ανάπτυξη της εμπιστοσύνης μεταξύ των χρηστών του συστήματος. Όσο μεγαλύτερο είναι το σκορ φήμης που έχει κάποιος πωλητής τόσο περισσότερη θα είναι και εμπιστοσύνη που θα δείχνουν προς το πρόσωπο του οι υπόλοιποι αγοραστές του συστήματος. Άρα θα πρέπει να διασφαλιστεί ότι το σκορ φήμης που παίρνει ένας πωλητή είναι το σωστό. Αυτό θα γίνει με τη χρήση κατάλληλων αλγορίθμων.

Σε αυτό το κεφάλαιο, σκοπός μας είναι να προσομοιώσουμε πειραματικά τέτοιους αλγορίθμους (όπως παρουσιάστηκαν στο κεφάλαιο 2) και στη συνέχεια να τους συγκρίνουμε βάση συγκεκριμένων ιδιοτήτων.

4.2 Υλοποίηση αλγορίθμων φήμης στο PRISM

Αρχικά μελετώντας δύο αλγορίθμους φήμης σκοπός μας είναι να βρούμε ποιος αλγόριθμος μας δίνει το καλύτερο και πιο αξιόπιστο σκορ φήμης για ένα πωλητή δεδομένου των βαθμολογιών που λαμβάνει μετά από κάθε συναλλαγή από ένα αγοραστή. Για να το επιτύχουμε αυτό θα πρέπει να γίνουν τα ακόλουθα:

1. Μοντελοποίηση του αλγορίθμου φήμης στο πιθανοτικό εργαλείο ελέγχου μοντέλου PRISM.
2. Ανάλυση του μοντέλου στο PRISM με τη χρήση της χρονικής λογικής PCTL

Η υλοποίηση του αλγορίθμου φήμης αποτελείται από τρία μέρη. Το πρώτο μέρος αφορά στην ενημέρωση των θετικών ή αρνητικών σκορ κάθε πωλητή. Αρχικά για κάθε πωλητή ορίζεται μια σταθερή τιμή που χαρακτηρίζει τη φύση του, το τι στην πραγματικότητα αντιπροσωπεύει. Αυτή η τιμή ορίζεται από την σταθερά *s_nature* και χαρακτηρίζει την πιθανότητα ένας πωλητής να πωλήσει σε ένα αγοραστή καλό προϊόν. Η αντίστοιχη πιθανότητα για κακό προϊόν ορίζεται από τη έκφραση *1-s_nature*. Έτσι κάθε πωλητής (seller) στέλνει ένα καλό ή κακό προϊόν βάσει των δύο αυτών πιθανοτήτων. Δεδομένου ότι η μεταβλητή *av11* ορίζει τον τύπο που προϊόντος που θα σταλεί (1: καλό, 2: κακό) ορίζουμε το πιο πάνω χρησιμοποιώντας την πιο κάτω εντολή του PRISM:

```
[ ]    b1=1 & s=1 -> s_nature : ( av11'=1 ) & ( s'=2 ) +  
                                ( 1 -_nature ) : ( av11'=2 ) & ( s'=3 );
```

Μετά από κάθε συναλλαγή ο αγοραστής (buyer) στέλνει θετικό ή αρνητικό feedback σκορ στον πωλητή με τον οποίο έκανε τη συναλλαγή με αποτέλεσμα να μεταβάλλεται το σύνολο των θετικών ή αρνητικών feedback σκορ του πωλητή (πίνακας 4.1).

<pre>[] f1=1 & s=2 & (good_score + bad_score < no_transc) & (transc < no_transc) -> (good_score'=good_score + 1) & (transc'= transc +1) & (avll'=0) & (s'=1);</pre>	1
<pre>[] f1=2 & s=3 & (good_score + bad_score < no_transc) & (transc < no_transc) -> (bad_score'=bad_score + 1) & (transc'= transc +1) & (avll'=0) & (s'=1);</pre>	2

Πίνακας 4.1: κώδικας συνθήκης μέτρησης θετικών και αρνητικών εκτιμήσεων του πωλητή

Πιο κάτω αναλύονται οι δύο περιπτώσεις που εμφανίζονται στον πίνακα 4.1:

(Περίπτωση 1) Εάν ο αγοραστής έδωσε θετική εκτίμηση ($f1=1$) και δεν έχουν γίνει ακόμα 100 συναλλαγές, τότε αύξησε το σύνολο των θετικών του και τον αριθμό των συναλλαγών κατά 1.

(Περίπτωση 2) Εάν ο αγοραστής έδωσε αρνητική εκτίμηση και δεν έχουν γίνει ακόμα 100 συναλλαγές τότε αύξησε το σύνολο των αρνητικών εκτιμήσεων του πωλητή.

Το σκορ που παίρνει ο πωλητής από τον αγοραστή εξαρτάται από το τι προϊόν έλαβε ο αγοραστής. Το αρχικό μοντέλο του αλγορίθμου κινείται στη λογική που εφαρμόζει το eBay, ότι δηλαδή αν ο αγοραστής έλαβε καλό προϊόν τότε βαθμολογεί θετικά αλλιώς αρνητικά. Το δεύτερο μέρος αφορά στον υπολογισμό του σκορ φήμης κάθε πωλητή βάση κάποιου αλγόριθμου λαμβάνοντας υπόψη τα θετικά και αρνητικά feedback σκορ που λαμβάνει.

Πριν την παρουσίαση και ανάλυση των αλγορίθμων φήμης που μοντελοποιήθηκαν αξίζει να σημειωθεί ότι καθένας από τους αλγορίθμους που αναφέρθηκε προηγουμένως αναλύθηκε σε δυο υποπεριπτώσεις. Αναλόγως του αν ο αγοραστής δίνει ειλικρινείς ή όχι εκτιμήσεις για τον συγκεκριμένο πωλητή έχουμε τις εξής υποπεριπτώσεις:

- **Ειλικρινές μοντέλο:** δίνουν θετικό feedback σκορ αν το προϊόν είναι καλό ενώ δίνουν αρνητικό feedback σκορ αν το προϊόν είναι κακό.

Η μεταβλητή avl1 δηλώνει κατά πόσο ένας πωλητής έχει στείλει καλό ή κακό προϊόν, ενώ όταν πάρει τιμή ίση με 2 τότε έχει στείλει κακό προϊόν.

```
[ ] avl1=1 & b1=2 -> (f1'=1) & (b1'=1);
[ ] avl1=2 & b1=2 -> (f1'=2) & (b1'=1);
```

- **Ανειλικρινές μοντέλο:** στο μοντέλο αυτό ο αγοραστής δίνει την εκτίμηση του βάσει τριών πιθανοτήτων που αναλογούν σε κάθε μια από τις τρεις κατηγορίες αγοραστή που ακολουθούν:

- **Ειλικρινής υποστήριξη (honest buyers):** αν το προϊόν είναι καλό τότε δίνει θετικό feedback σκορ ενώ δίνει αρνητικό feedback σκορ αν το προϊόν είναι κακό.
- **Θετική υποστήριξη (positive buyers):** οποιασδήποτε ποιότητας και να είναι το προϊόν (καλό ή κακό) ο αγοραστής δίνει μόνο θετικό feedback σκορ.
- **Αρνητική υποστήριξη (negative buyers):** οποιασδήποτε ποιότητας και να είναι το προϊόν (καλό ή κακό) ο αγοραστής δίνει μόνο αρνητικό feedback σκορ.

Οι πιο πάνω περιπτώσεις εφαρμόζονται βάσει πιθανοτήτων που ορίζονται στην αρχή του μοντέλου και έχουν ως εξής:

```
[ ] b1=1 -> honest : ( b1'=2) + positive : ( b1'=3) + negative : ( b1'=4);
[ ] avl1=1 & b1=2 -> (f1'=1) & (b1'=1);
[ ] avl1=2 & b1=2 -> (f1'=2) & (b1'=1);
[ ] b1=3 & avl1 > 0 -> (f1'=1) & (b1'=1);
[ ] b1=4 & avl1 > 0 -> (f1'=2) & (b1'=1);
```

Όπως φαίνεται πιο πάνω ο αγοραστής βάσει των τριών παραμέτρων (honest, positive, negative) οι οποίες ορίζουν το ποσοστό ειλικρινούς, μόνο θετικής ή αρνητικής υποστήριξης, δίνει την εκτίμηση του στον πωλητή.

Στη συνέχεια της ενότητας προχωρούμε στην υλοποίηση δύο αλγορίθμων φήμης που παρουσιάστηκαν στο Κεφάλαιο 2, τον αλγόριθμο φήμης του eBay και τον TNA-SL.

4.2.1 Αλγόριθμός φήμης του eBay

Ο αλγόριθμος που χρησιμοποιήσαμε εμείς χρησιμοποιεί τη λογική που ακολουθεί το eBay, ότι δηλαδή μετά από κάθε συναλλαγή μεταξύ ενός αγοραστή και ενός πωλητή ο αγοραστής δίνει θετικό ή αρνητικό σκορ στον πωλητή. Οι βαθμολογίες αυτές είναι είτε θετικές είτε αρνητικές. Κάθε φορά που δίνει θετική εκτίμηση ο αγοραστής, το σύνολο των θετικών εκτιμήσεων (positives) του πωλητή αυξάνεται κατά μία μονάδα, ενώ όταν δίνει αρνητική εκτίμηση το σύνολο των αρνητικών εκτιμήσεων (negatives) αυξάνεται κατά μία μονάδα.

Στη συνέχεια αφού υπολογιστούν τα δύο σύνολα ο πωλητής λαμβάνει σαν σκορ φήμης το λεγόμενο positive feedback percentage που έχει το e-bay και υπολογίζεται ως εξής:

$$\text{Positive feedback percentage} = \frac{\text{positive}}{\text{positive} + \text{negatives}}$$

$$\text{Formula rep} = \text{good_score} / (\text{good_score} + \text{bad_score})$$

4.2.2 Αλγόριθμος TNA-SL

Όπως και στον προηγούμενο αλγόριθμο χρησιμοποιούμε τη λογική αυτού του αλγόριθμου που έχει να κάνει με την έννοια της εμπιστοσύνης. Ο αλγόριθμος

αυτός βασίζεται επίσης στο σύνολο των θετικών και το σύνολο των αρνητικών εκτιμήσεων που έχει κάποιος πωλητής και χρησιμοποιούνται για να υπολογιστεί η εμπιστοσύνη. Όπως προαναφέρθηκε σε προηγούμενη ενότητα η εμπιστοσύνη κάποιου αγοραστή εκφράζεται συνάρτηση τριών παραμέτρων της πίστης (belief), αβεβαιότητας (uncertainty) και μιας παραμέτρου α (base rate) που καθορίζει την βεβαιότητα του αποτελέσματος. Το α παίρνει τιμές 0.5 όταν έχουμε non pre trusted χρήστες και 1.0 όταν έχουμε pre trusted.

Με βάση αυτά και αφού υπολογιστούν τα σκορ των θετικών και αρνητικών εκτιμήσεων υπολογίζεται η φήμη του κάθε πωλητή βάσει του expected value που ορίζεται από τον κύριο αλγόριθμο του TNA-SL και έχει ως εξής:

$$EV = \text{belief} + a * \text{uncertainty}$$

```
Formula belief = good score / (good score + bad _ score + 2);  
Formula uncertainty = 2 / (good _ score + bad _ score + 2)  
Formula rep = belief + uncertainty * a;
```

Η υλοποίηση αυτή διαφέρει από την προηγούμενη μόνο στον αλγόριθμο που χρησιμοποιεί για να υπολογίσει τη φήμη.

Στη συνέχεια του κεφαλαίου αυτού δίνονται τα σενάρια με τα οποία έγιναν τα πειράματα και εξάχθηκαν τα αποτελέσματα.

4.3 Αξιολόγηση και σύγκριση πειραματικών προσομοιώσεων

Στην ενότητα αυτή θα εκτιμήσουμε τους αλγορίθμους φήμης eBay και TNA-SL θα τους συγκρίνουμε μεταξύ τους βάσει κάποιων ιδιοτήτων που αφορούν το σκορ φήμης που υπολογίζουν για ένα seller. Για την ορθότερη αξιολόγηση των

αλγορίθμων φήμης, θα εξετάσουμε τις ίδια ιδιότητα για διαφορετικά σενάρια κάθε φορά.

Όπως αναφέραμε και πιο πάνω κάθε πωλητής έχει ένα reputation score αλλά και ένα σταθερό nature score, το οποίο είναι μέτρο σύγκρισης για το αν ο αλγόριθμος μας δίνει έγκυρο και έμπιστο reputation score για κάποιο πωλητή. Άρα όσο το reputation score (φήμη) του πωλητή πλησιάζει την πραγματική φύση του τόσο πιο αξιόπιστο και έγκυρο είναι το σκορ φήμης του πωλητή. Αξίζει να αναφέρουμε ότι όλα τα πειράματα πραγματοποιούνται με ένα αγοραστή και ένα πωλητή. Το μοντέλο χρησιμοποιεί μόνο έναν αγοραστή διότι η φήμη του πωλητή δεν μεταβάλλεται από τον αριθμό των αγοραστών που συμμετέχουν αλλά από το τι βαθμολογίες λαμβάνουν από αυτούς. Βασιζόμενοι σε αυτό θα ξεκινήσουμε με τη σύγκριση των δύο πρώτων αλγορίθμων υπολογίζοντας την πιθανότητα το reputation score να φτάσει το nature score. Στα πειράματα που πραγματοποιήθηκαν μελετήθηκαν δύο ιδιότητες.

Ιδιότητα 1

Η ιδιότητα αυτή μελετήθηκε για διαφορετικά σενάρια και εκφράζεται σε χρονική λογική ως εξής:

$$P=? [F \leq T \text{ (rep} = \text{s_nature)}]$$

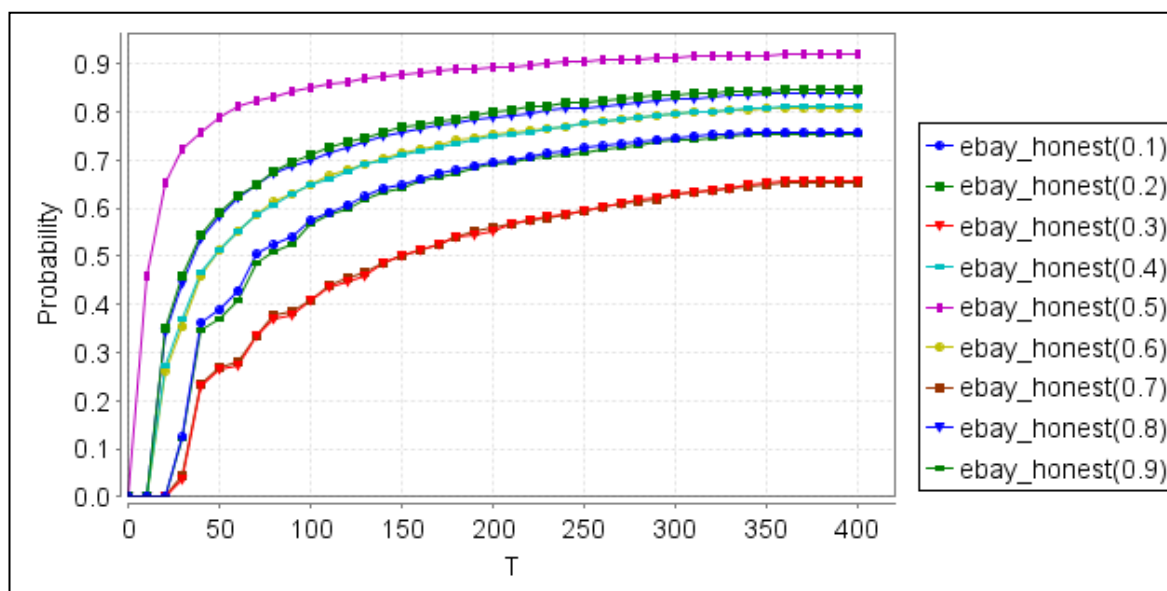
όπου το rep αντιστοιχεί στο reputation score του πωλητή και το οποίο υπολογίζεται κάθε φορά για καθένα από τους αλγορίθμους και s_nature είναι η πραγματική φύση του πωλητή. Η ιδιότητα αυτή εκφράζει την πιθανότητα η φήμη του πωλητή να φτάσει την πραγματική του φύση σε T χρονικές στιγμές. Στα σενάρια φαίνεται η πιθανότητα αυτή συναρτήσει του χρόνου T ανά 10 βήματα.

Σε εμπειρικές μελέτες που γίνονται κατά καιρούς [5] σε σύστημα ηλεκτρονικού εμπορίου και αφορούν τη συμπεριφορά των χρηστών, διαπιστώθηκε ότι περίπου

το 90% των χρηστών μετά το τέλος κάθε συναλλαγής με άλλο χρήστη δίνουν πάντα θετική εκτίμηση όσο ευχαριστημένοι ή δυσαρεστημένοι μπορεί να είναι από την συναλλαγή που έκαναν. Αυτό συμβαίνει διότι φοβούνται τη μελλοντική αντιμετώπιση (ίσως αν δώσουν αρνητική εκτίμηση να λάβουν στο μέλλον αρνητική εκτίμηση) που μπορεί να έχουν από τους χρήστες με τους οποίους έκαναν συναλλαγές στο παρελθόν. Επίσης οι μελέτες αυτές δείχνουν ότι μόνο το 10% των συναλλαγών βασίζονται σε ειλικρινείς εκτιμήσεις από τους χρήστες. Με βάση τα πιο πάνω αποφάσισαμε όπως στις περιπτώσεις που γίνεται αναφορά στο ανειλικρινές μοντέλο οι παράμετροι *honest*, *positive* και *negative* που προαναφέρθηκαν να παίρνουν τιμές ίσες με 0.1, 0.9 και 0.

Σενάριο 1

Στο σενάριο 1 έχουμε ένα σύστημα με ένα αγοραστή και ένα πωλητή όπου γίνονται 100 συναλλαγές μεταξύ τους. Σκοπός του Σεναρίου 1 είναι να παρατηρήσουμε τη συμπεριφορά του ειλικρινούς μοντέλου του e-bay για διαφορετικές τιμές της φύσης του πωλητή (οι οποίες αναγράφονται με μέσα στην παρένθεση στο δεξί μέρος της γραφικής παράστασης).

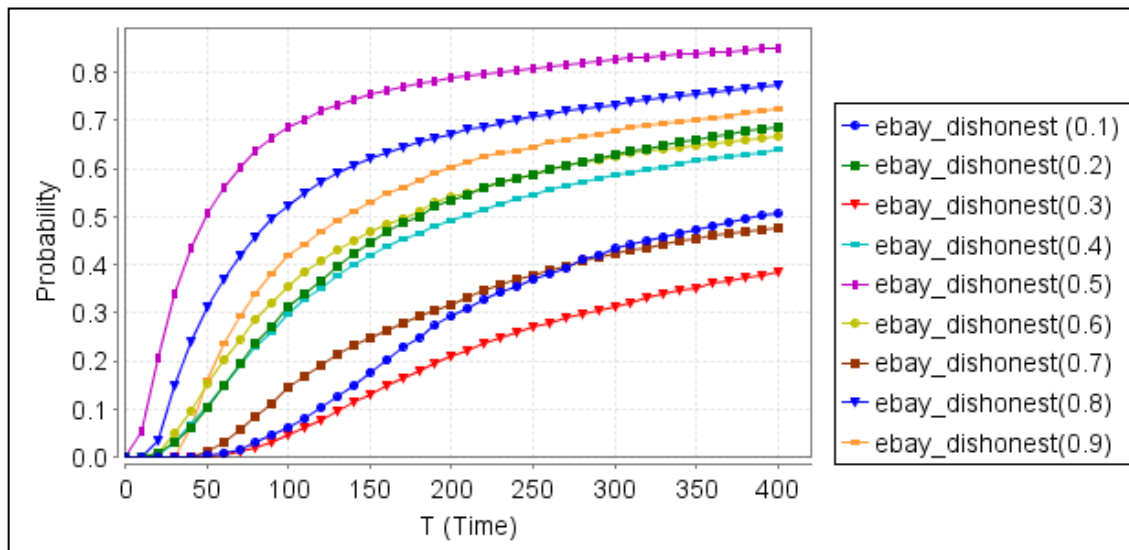


Σχήμα 4.1: Σενάριο 1 προσομοίωση ειλικρινούς μοντέλου του e-bay

Παρατηρούμε ότι ο αλγόριθμος αυτός μας δίνει καλύτερα αποτελέσματα όταν η φύση του πωλητή είναι 0.5, διότι λαμβάνοντας υπόψη τη φύση του πωλητή ο οποίος δίνει 50% καλό προϊόν και 50% κακό προϊόν είναι πιθανότερο να λάβει φήμη ίση με 50%. Επίσης παρατηρούμε ότι για ψηλές τιμές της φύσης του πωλητή η πιθανότητα επίσης κυμαίνεται σε ψηλά επίπεδα κάτι που αναμέναμε να συμβεί αφού οι πωλητές με ψηλή φύση έχουν μεγαλύτερη πιθανότητα να πάρουν ψήλο σκορ φήμης με την πάροδο του χρόνου.

Σενάριο 2

Στο σενάριο 2 έχουμε ένα σύστημα με ένα αγοραστή και ένα πωλητή όπου γίνονται 100 συναλλαγές μεταξύ τους. Σκοπός του Σεναρίου 2 είναι να παρατηρήσουμε τη συμπεριφορά του μη ειλικρινούς μοντέλου του e-bay για διαφορετικές τιμές της φύσης του πωλητή (οι οποίες αναγράφονται με μέσα στην παρένθεση στο δεξί μέρος της γραφικής παράστασης)..



Σχήμα 4.2: Σενάριο 2- προσομοίωση ανειλικρινούς μοντέλου του e-bay

Παρατηρούμε ότι για τιμές της φύσης 0.5 έχουμε καλύτερα αποτελέσματα, αφού η φήμη του πωλητή πλησιάζει την πραγματική του φύση κατά 80% και αυτό

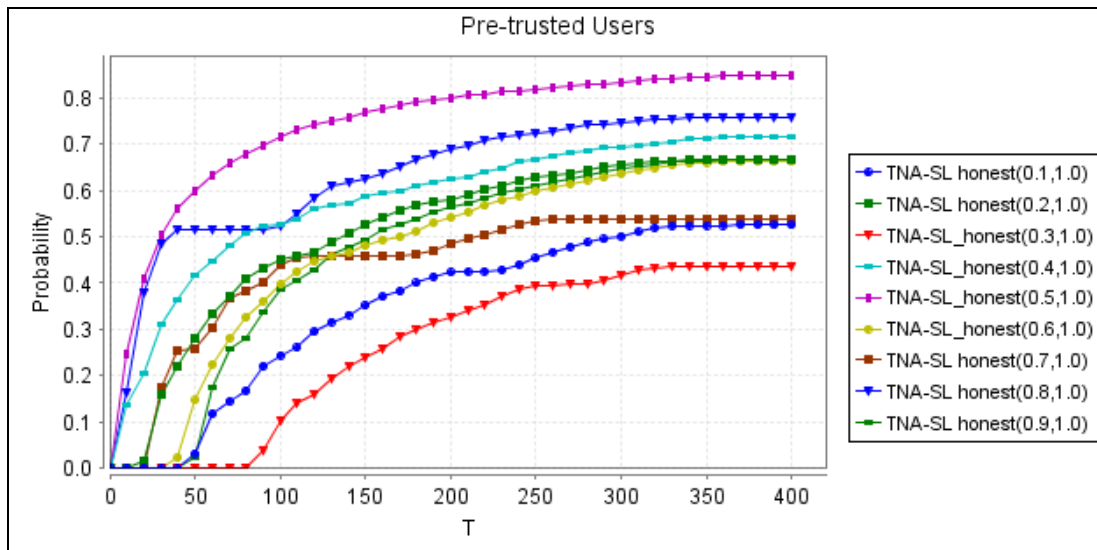
συμβαίνει (μειώθηκε σε σχέση με το προηγούμενο), διότι όπως και στο προηγούμενο σενάριο ο πωλητής δίνει 50% καλό προϊόν και 50% κακό προϊόν είναι πιθανότερο να λάβει φήμη ίση με 50%.

Σε αυτό το σενάριο βλέπουμε ότι η πιθανότητα μειώθηκε σε σύγκριση με το σενάριο 1 αλλά παραμένει σε ψηλά επίπεδα όταν έχουμε καλούς πωλητές (με ψηλή φύση), διότι πλέον οι αγοραστές δίνουν στις πλείστες περιπτώσεις θετικές εκτιμήσεις ανεξάρτητα από το προϊόν που παίρνουν κάτι που λογικά θα αυξήσει τη φήμη τους μειώνοντας την πιθανότητα να έχουν φήμη ίση με τη φύση τους. Κάτι άλλο που παρατηρούμε είναι ότι σε χαμηλές τιμές της φύσης κάποιου πωλητή η πιθανότητα επίσης μειώθηκε, κάτι που σημαίνει ότι κάποιος κακός πωλητής μπορεί να αυξήσει τη φήμη του κάτι που φαίνεται λογικό αφού λαμβάνει συνεχώς θετικές εκτιμήσεις.

Σενάριο 3

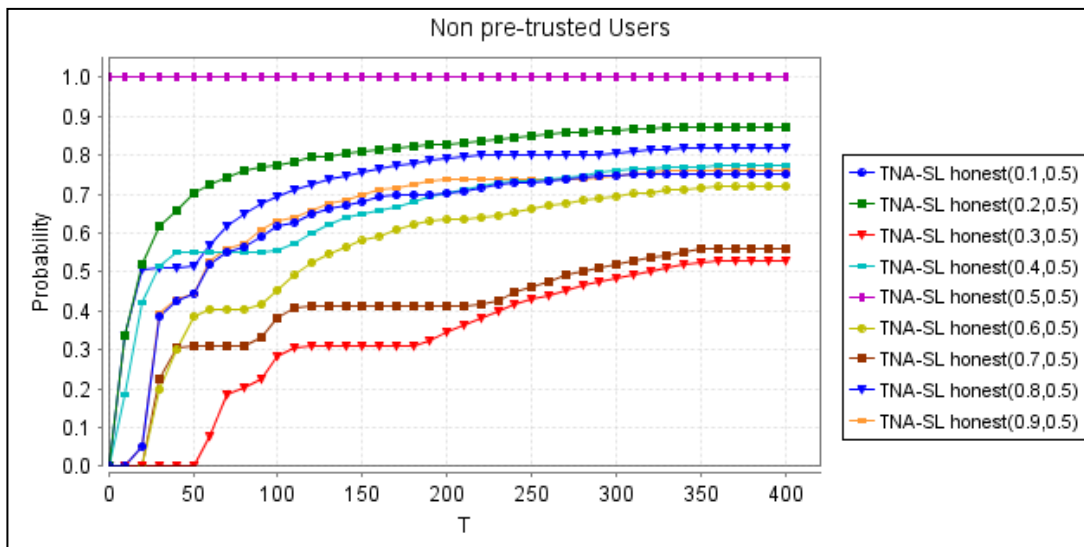
Στο σενάριο αυτό έχουμε ένα σύστημα που πραγματοποιούνται 100 συναλλαγές μεταξύ ενός αγοραστή και ενός πωλητή. Σκοπός του Σεναρίου 3 είναι να συγκρίνουμε τον αλγόριθμο TNA-SL για διαφορετικές τιμές της φύσης του πωλητή, όταν το μοντέλο είναι ειλικρινές. και για δυο διαφορετικές καταστάσεις του συστήματος. Η πρώτη είναι να έχουμε pre-trusted πωλητή ενώ στη δεύτερη ο πωλητής δεν είναι pre-trusted.

Pre-Trusted χρήστες



Σχήμα 4.3: Σενάριο 3-προσομοίωση ειλικρινούς μοντέλου του TNA-SL

Non pre-trusted Users



Σχήμα 4.4: Σενάριο 3-προσομοίωση ειλικρινούς μοντέλου του TNA-SL

Παρατηρώντας τις πιο πάνω γραφικές παραστάσεις διαπιστώνουμε τα αποτελέσματα για τον αλγόριθμο TNA-SL είναι όπως τα περιμέναμε με βάση τον τρόπο που λειτουργεί στις περιπτώσεις που η φύση του πωλητή κυμαίνεται από 0.1-0.2 και 0.9. Αυτό που περιμένουμε από τον αλγόριθμο είναι στις χαμηλές τιμές

της φύσης ενός πωλητή, η φήμη του πωλητή να μειώνεται ακόμη περισσότερο, αφού υπάρχει ο παράγοντας της αβεβαιότητας του αποτελέσματος. Το γεγονός ότι δεν έχουμε pre-trusted χρήστες, δηλαδή οι αγοραστές δεν εμπιστεύονται του πωλητές και επιπλέον οι πωλητές είναι κακοί μειώνει ακόμη πιο πολύ τη φήμη τους. Άρα η πιθανότητα η φήμη τους να πλησιάσει την πραγματική φύση του μεγαλώνει.

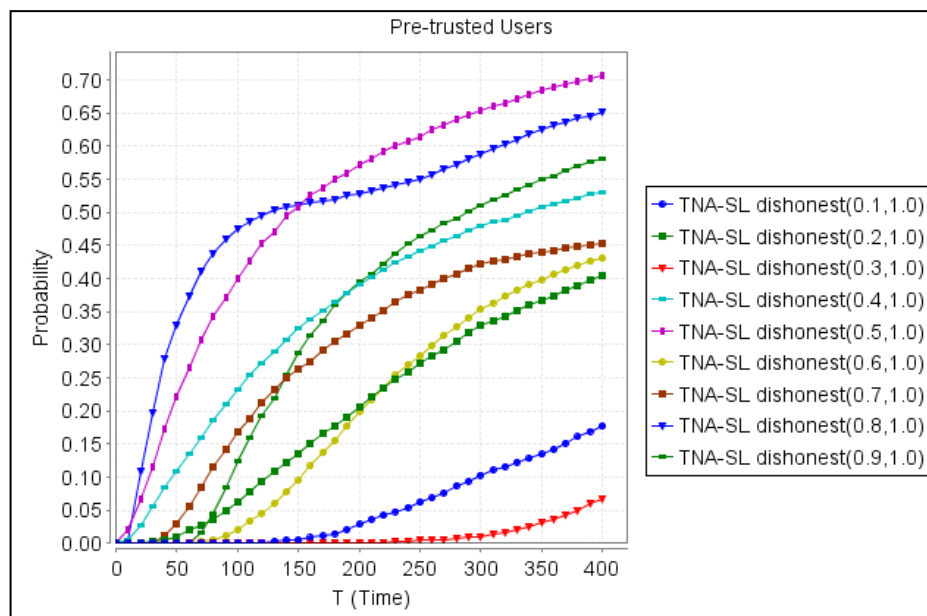
Παρατηρούμε ότι στην περίπτωση που ένας πωλητής είναι κακός (έχει χαμηλό nature) και είναι pre-trusted (τυγχάνει εμπιστοσύνης από τους υπόλοιπους χρήστες του συστήματος), η πιθανότητα να έχει χαμηλή φήμη κυμαίνεται σε χαμηλά επίπεδα. Αυτό συμβαίνει διότι η φήμη κάποιου πωλητή ο οποίος είναι pre-trusted αυξάνεται, αφού δεν υπάρχει πλέον αβεβαιότητα για το εάν το σκορ που του δίνει κάποιος αγοραστής είναι σωστό ή λάθος. Μια άλλη παρατήρηση που μπορούμε να κάνουμε είναι η κατανομή της πιθανότητας στο γράφημα δεν είναι ομοιόμορφη σε συνάρτηση με το χρόνο όπως συμβαίνει στα σενάρια 1 και 2, ίσως επειδή εισάγεται ο παράγοντας της αβεβαιότητας στον υπολογισμό της φήμης.

Ο TNA –SL non pre-trusted φαίνεται να είναι καλύτερος από τον TNA –SL pre-trusted αφού απομονώνει τους κακούς χρήστες (αυτούς με χαμηλή φύση) διατηρώντας τα επίπεδα της φήμης τους σε χαμηλά επίπεδα, ενώ παράλληλα αναδεικνύει τους καλούς χρήστες κρατώντας τη φήμη τους σε ψηλά επίπεδα.

Σενάριο 4

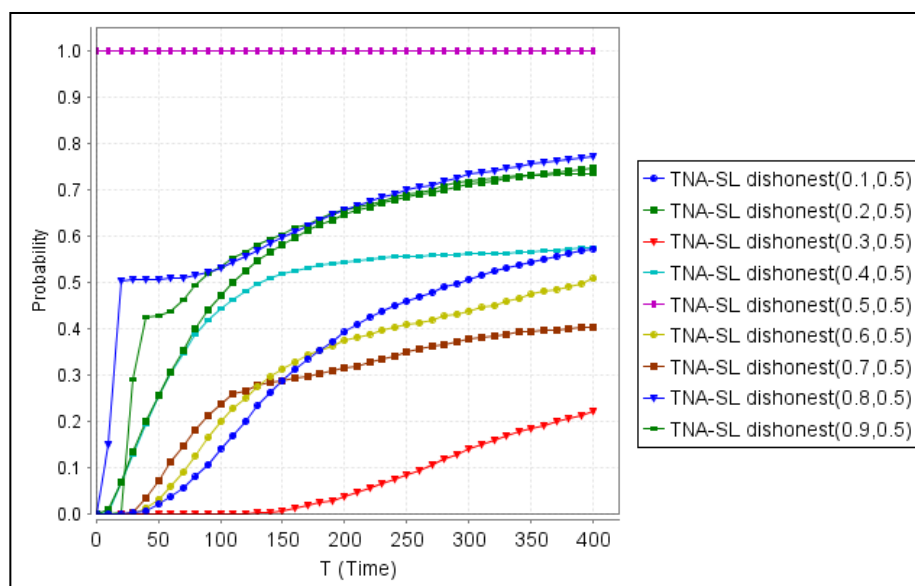
Στο σενάριο αυτό έχουμε ένα σύστημα που πραγματοποιούνται 100 συναλλαγές μεταξύ ενός αγοραστή και ενός πωλητή. Σκοπός του Σεναρίου 4 είναι να συγκρίνουμε το ανειλικρινή μοντέλο TNA-SL (υποενότητα 4.2) για διαφορετικές τιμές της φύσης του πωλητή. και για δυο διαφορετικές καταστάσεις του συστήματος. Η πρώτη είναι να έχουμε pre-trusted πωλητή ενώ στη δεύτερη ο πωλητής δεν είναι pre-trusted.

Pre-trusted



Σχήμα 4.5: Σενάριο 4- προσομοίωση ανειλικρινούς μοντέλου του TNA-SL για Pre-trusted χρήστες

Non pre-trusted



Σχήμα 4.6: Σενάριο 4- προσομοίωση ανειλικρινούς μοντέλου του TNA-SL για Non pre-trusted χρήστες

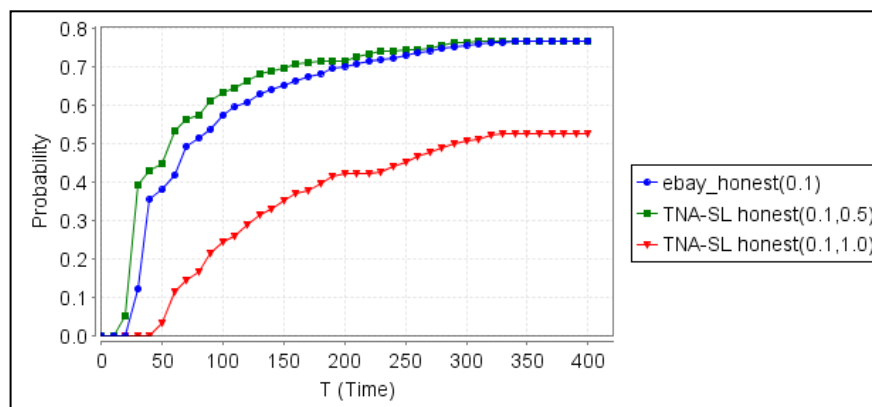
Παρατηρώντας τις πιο πάνω γραφικές παραστάσεις συμπεραίνουμε ότι ο πρώτος αλγόριθμος που αφορά pre-trusted χρήστες δίνει καλύτερα αποτελέσματα σε βάθος χρόνου στις περιπτώσεις που έχουμε κακούς πωλητές. Ενώ ο δεύτερος αλγόριθμος που αφορά non-pre-trusted χρήστες δίνει καλύτερα αποτελέσματα όταν οι πωλητές είναι καλοί. Αυτό συμβαίνει, διότι αφενός στον πρώτο αλγόριθμο υπάρχει βεβαιότητα για το αποτέλεσμα οπότε η φήμη κάποιου πωλητή αυξάνεται περισσότερο σε σύγκριση με το 2^ο αλγόριθμο και επιπλέον οι αγοραστές δίνουν στις πλείστες περιπτώσεις καλή εκτίμηση. Έτσι αυτό επηρεάζει περισσότερο τη φήμη των κακών πωλητών.

Στο δεύτερο αλγόριθμο έχουμε καλύτερα αποτελέσματα από τον 1^ο αλγόριθμο όταν οι πωλητές είναι καλοί, η φήμη τους δεν επηρεάζεται τόσο πολύ ούτε από το γεγονός ότι παίρνουν συνέχεια θετικές εκτιμήσεις από τους αγοραστές, αλλά ούτε από το γεγονός ότι δεν είναι pre-trusted. Σε αυτή την περίπτωση το γεγονός ότι είναι non-pre-trusted του δίνει πλεονέκτημα σε σχέση με τους κακούς, αφού ο αλγόριθμος αυτός απομακρύνει τους κακούς χρήστες.

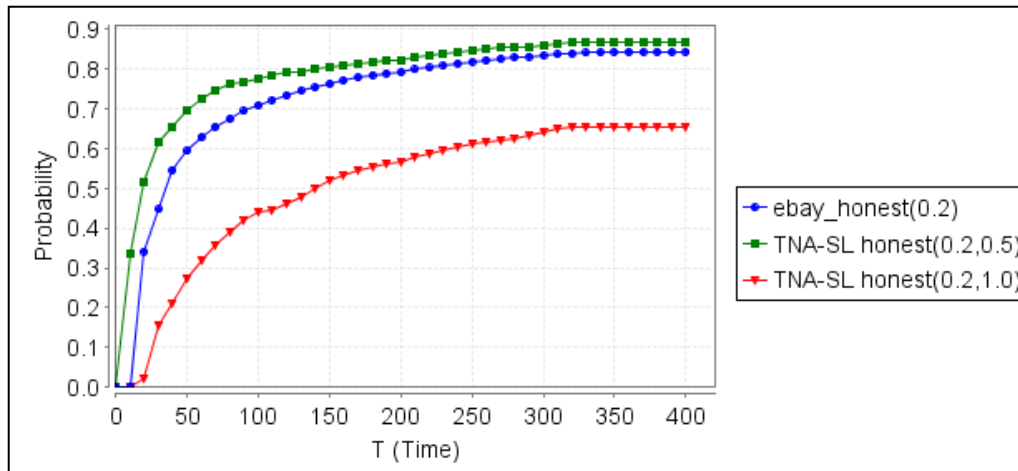
Σύγκριση αλγορίθμων βάσει της ιδιότητας 1

- Ειλικρινές μοντέλο

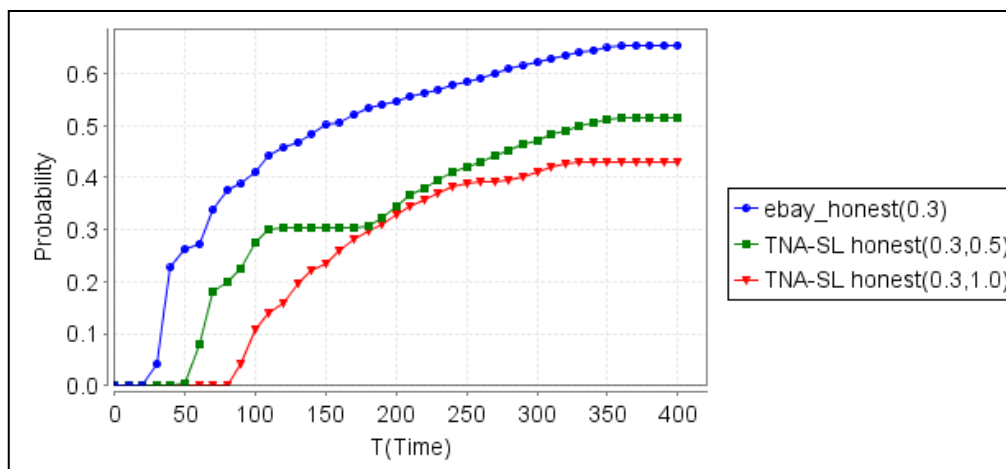
Πιο κάτω γίνεται οι σύγκριση των διαφορετικών αλγορίθμων μεταξύ τους για διαφορετική τιμή της φύσης του πωλητή που χρησιμοποιούν το ειλικρινές μοντέλο.



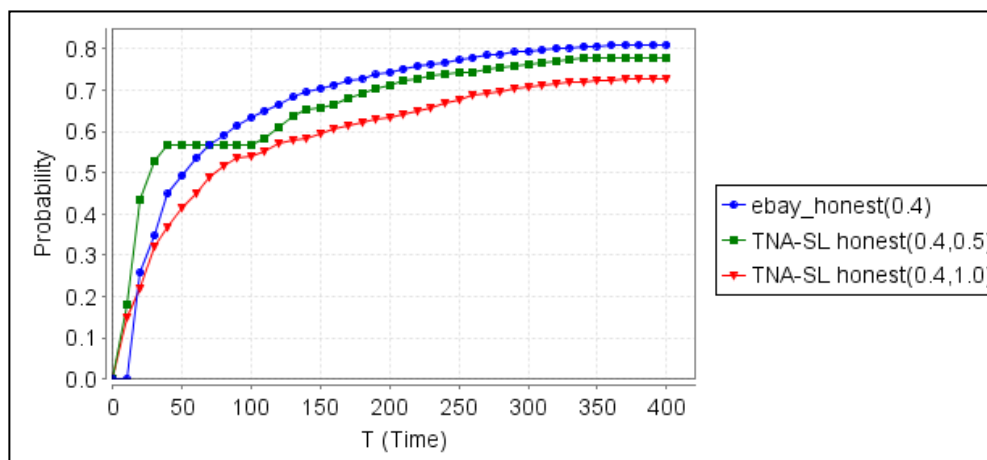
Σχήμα 4.7: σύγκριση αλγορίθμων με nature=0.1



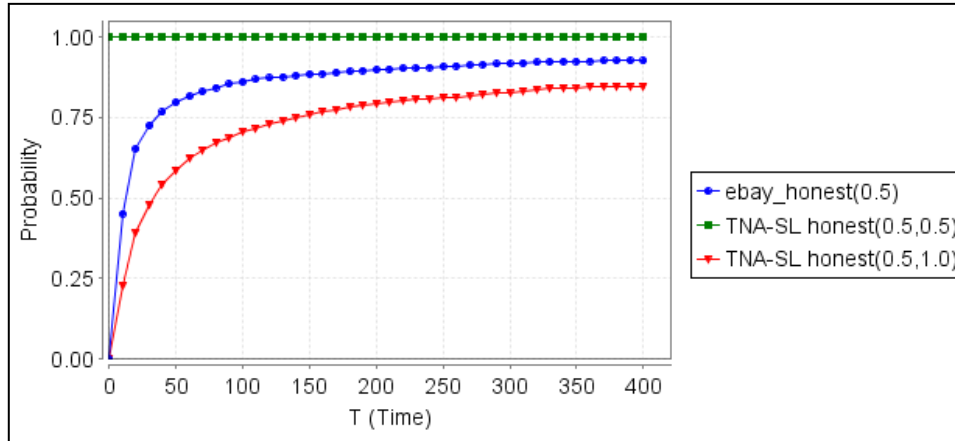
Σχήμα 4.8: σύγκριση αλγορίθμων με nature=0.2



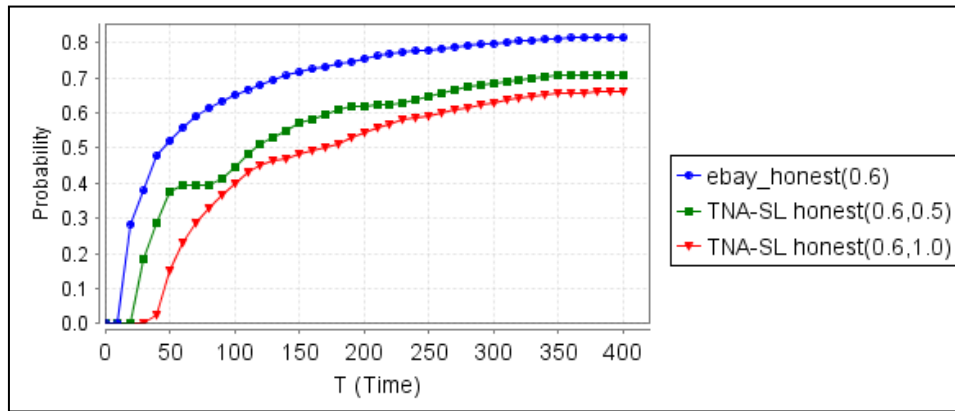
Σχήμα 4.9: σύγκριση αλγορίθμων με nature=0.3



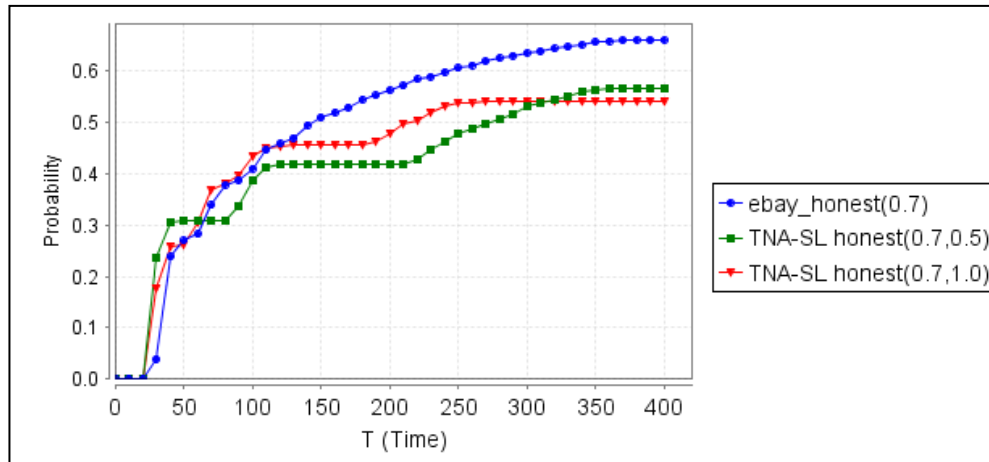
Σχήμα 4.10: σύγκριση αλγορίθμων με nature=0.4



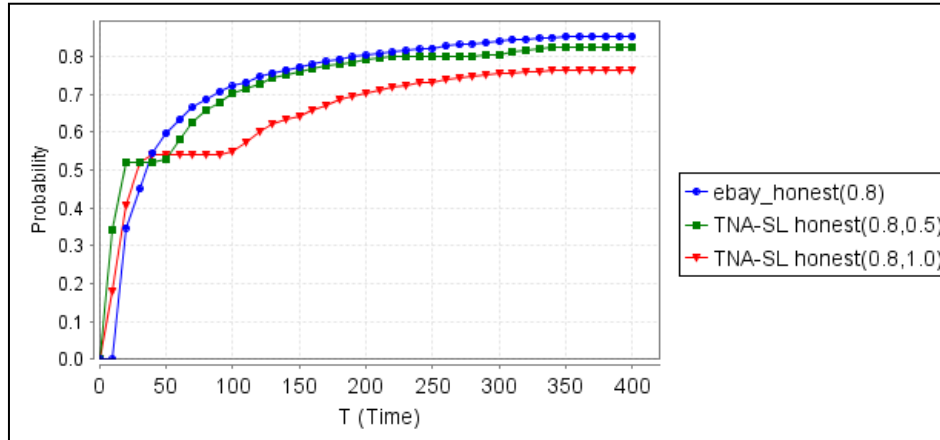
Σχήμα 4.11: σύγκριση αλγορίθμων με nature=0.5



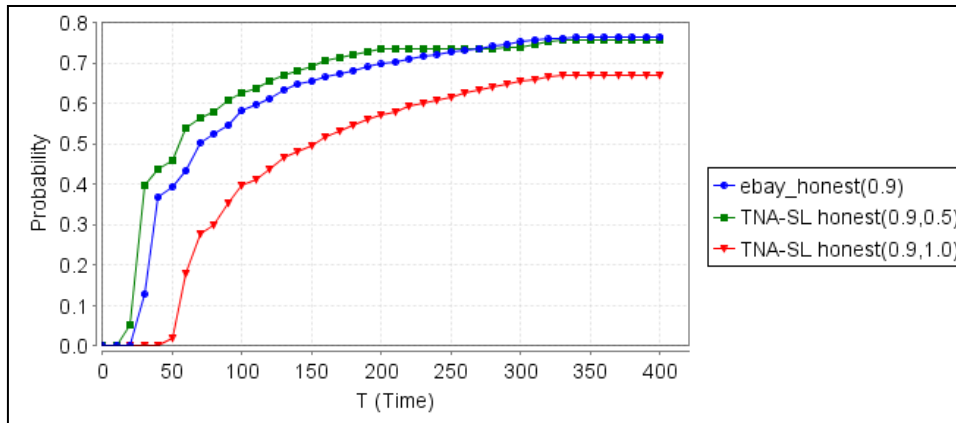
Σχήμα 4.12: σύγκριση αλγορίθμων με nature=0.6



Σχήμα 4.13: σύγκριση αλγορίθμων με nature=0.7



Σχήμα 4.14: σύγκριση αλγορίθμων με nature=0.8

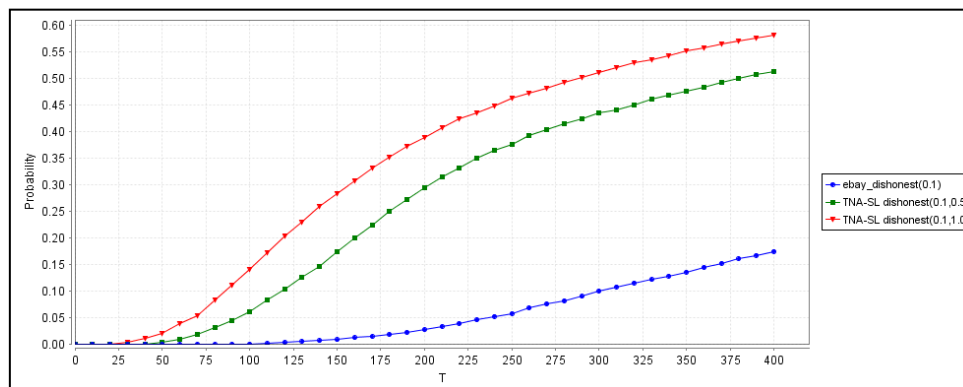


Σχήμα 4.15: σύγκριση αλγορίθμων με nature=0.9

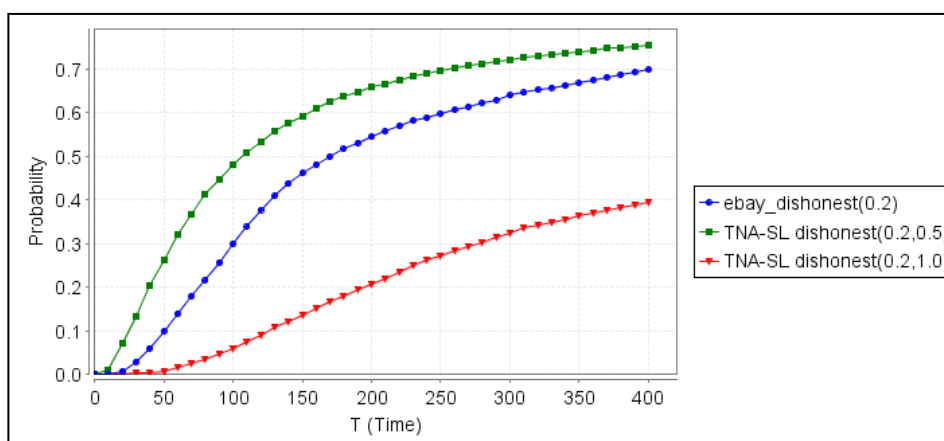
Με βάση τις πιο πάνω γραφικές παραστάσεις παρατηρούμε ότι στις πολύ χαμηλές τιμές (για 0.1 και 0.2) αλλά και στις πολύ ψηλές τιμές (για 0.8 και 0.9) τη φύσης του πωλητή οι αλγόριθμοι του eBay και του TNA-SL για non pre-trusted χρήστες δίνουν τα καλύτερα αποτελέσματα αφού στην περίπτωση που ο πωλητής είναι καλός η φήμη διατηρείται σε ψηλά επίπεδα ενώ στην περίπτωση που ο πωλητής είναι κακός η φήμη του βρίσκεται σε χαμηλά επίπεδα.

- **Ανελικρινές μοντέλο**

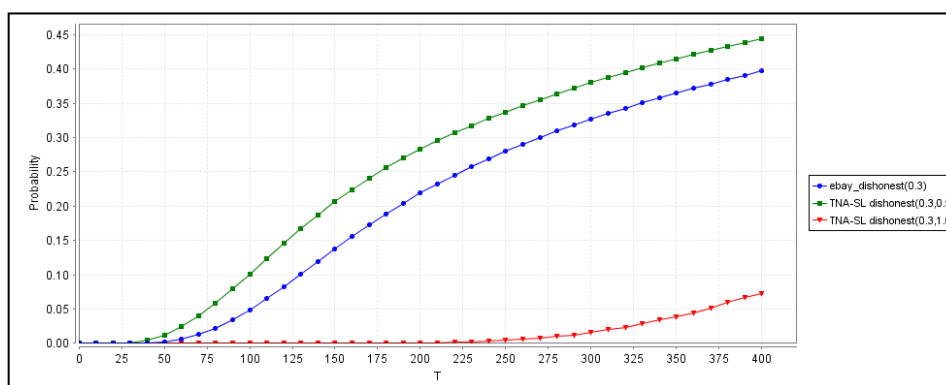
Πιο κάτω γίνεται η σύγκριση των διαφορετικών αλγορίθμων μεταξύ τους για διαφορετική τιμή της φύσης του πωλητή που χρησιμοποιούν το ειλικρινές μοντέλο.



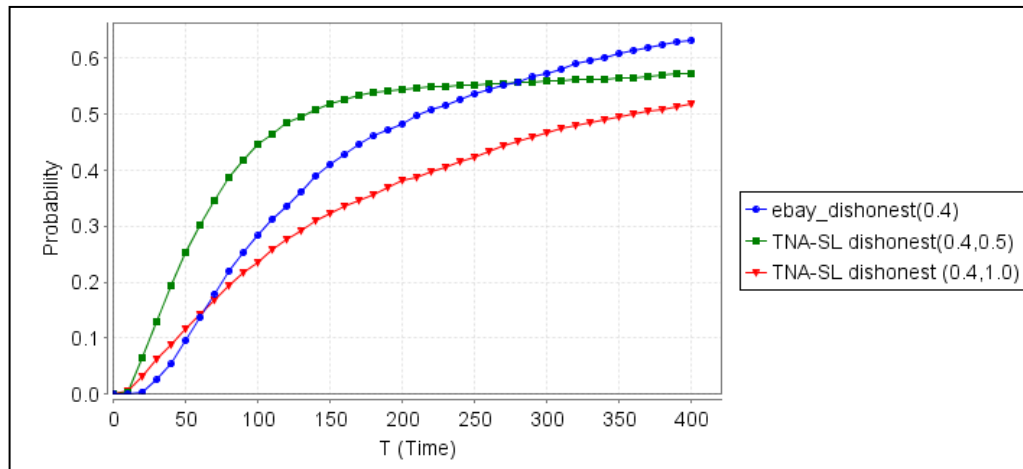
Σχήμα 4.16: σύγκριση αλγορίθμων με nature=0.1



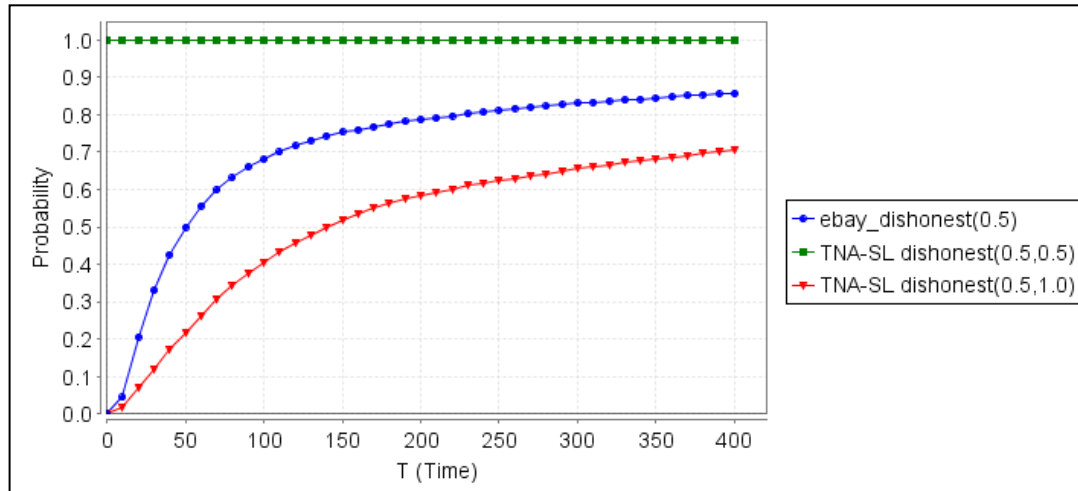
Σχήμα 4.17: σύγκριση αλγορίθμων με nature=0.2



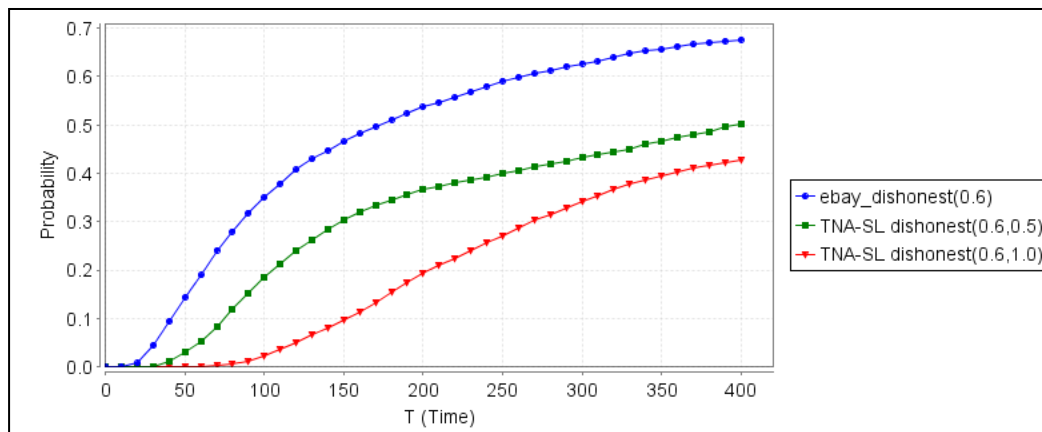
Σχήμα 4.18: σύγκριση αλγορίθμων με nature=0.3



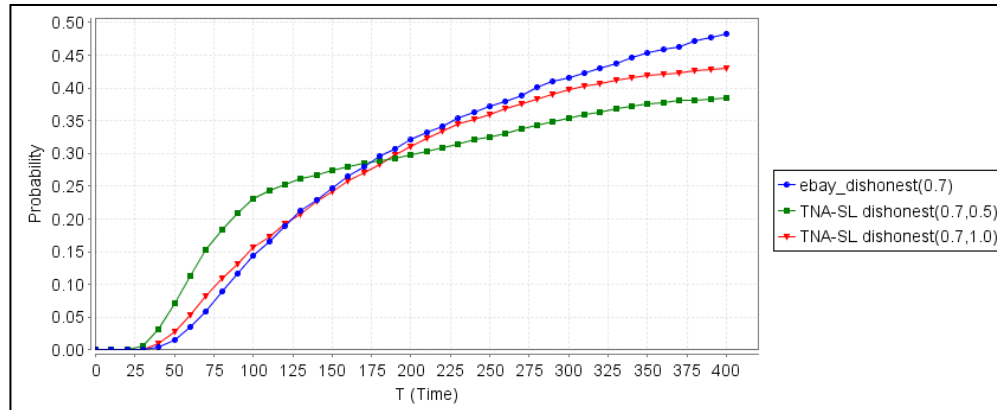
Σχήμα 4.19: σύγκριση αλγορίθμων με nature=0.4



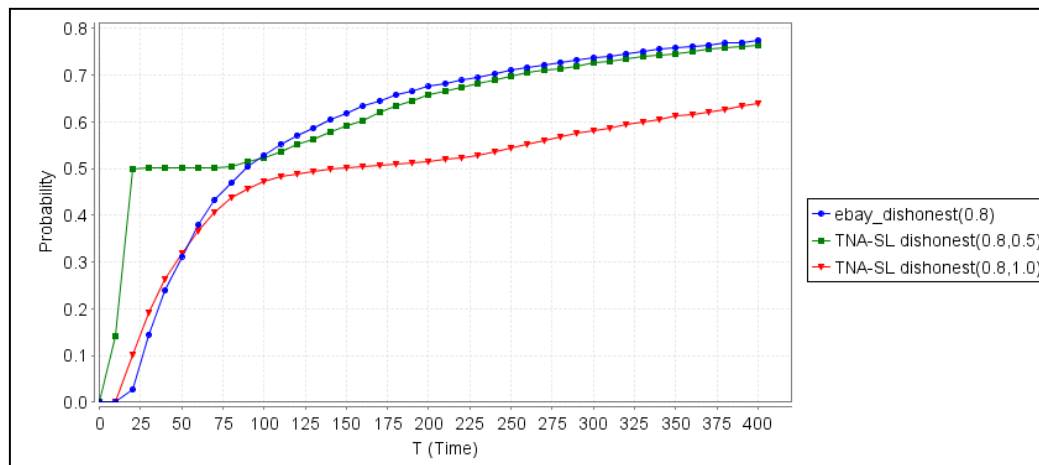
Σχήμα 4.20: σύγκριση αλγορίθμων με nature=0.5



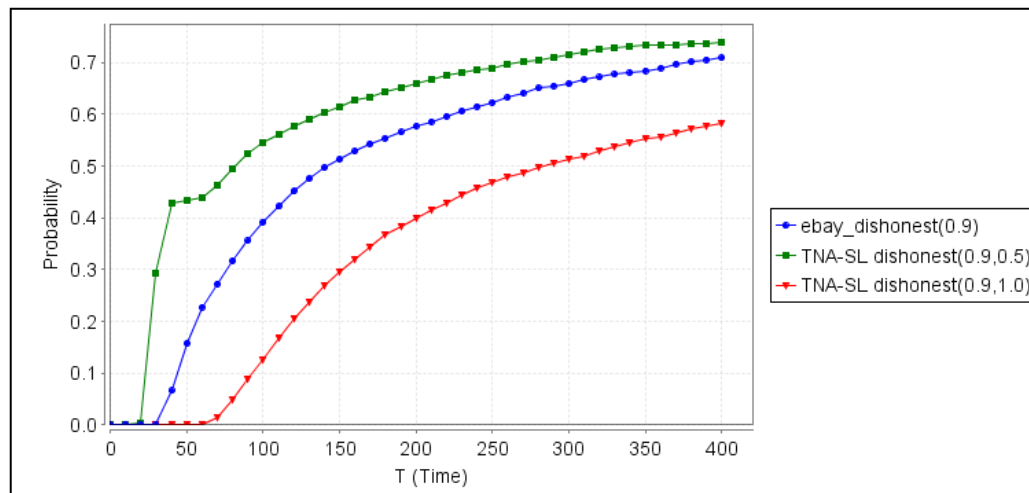
Σχήμα 4.21: σύγκριση αλγορίθμων με nature=0.6



Σχήμα 4.22: σύγκριση αλγορίθμων με nature=0.7



Σχήμα 4.23: σύγκριση αλγορίθμων με nature=0.8



Σχήμα 4.24: σύγκριση αλγορίθμων με nature=0.9

Παρατηρούμε ότι όταν η φύση των πωλητών είναι χαμηλή ο αλγόριθμος TNA-SL με non pre-trusted χρήστες δίνει καλύτερα αποτελέσματα από τους υπόλοιπους αλγορίθμους γιατί η φήμη τους αναμένεται να πλησιάσει τη φύση τους σε μεγάλο ποσοστό άρα η φήμη που θα έχουν θα είναι αξιόπιστη. Για περιπτώσεις όπου η φύση του πωλητή είναι ψηλή τα καλύτερα αποτελέσματα τα δίνει ο αλγόριθμος του eBay διότι η πιθανότητα κυμαίνεται σε ψηλά επίπεδα που σημαίνει ότι η φήμη του πωλητή θα πλησιάσει τη φύση του, άρα θα είναι πιο αξιόπιστη.

Ιδιότητα 2

Η ιδιότητα αυτή θα μελετηθεί για όλους τους αλγορίθμους που αναλύθηκαν χωρίς τη χρήση κάποιων σεναρίων και εκφράζεται σε χρονική λογική ως εξής:

$$S=? \text{ [exp} \leq \text{bound]}$$

όπου το exp ορίζεται σαν η διαφορά της φήμης του πωλητή από την φύση του. Η ιδιότητα αυτή δίνει την πιθανότητα (long run probability) η διαφορά της φήμης από τη φύση του πωλητή να είναι μικρότερη ή ίση από κάποια τιμή bound.

Αρχικά αποφασίσαμε να μελετήσουμε την ιδιότητα $S=? \text{ (exp} \leq 0.1)$ με σκοπό να διερευνήσουμε κατά πόσο η φήμη πωλητή φτάνει τη φύση του. Έτσι θέσαμε σαν πρώτο όριο στο πείραμα η διαφορά των δύο τιμών, της φήμης και της φύσης, να φτάνει μέχρι και 10%. Εκτελώντας το πείραμα αυτό πήραμε τα αποτελέσματα που φαίνονται στο πίνακα. Παρατηρώντας το πίνακα βλέπουμε ότι η φήμη του πωλητή και στους δύο αλγόριθμους κατά την εκτέλεση της ιδιότητας 1 πλησιάζει τη φύση του σε ποσοστό μεγαλύτερο του 96%. Για να βεβαιωθούμε ότι η φήμη του πωλητή κυμαίνεται σε αυτά τα επίπεδα αποφασίσαμε να ελέγξουμε την ιδιότητα για τιμή μικρότερη ή ίση του 5%.

Παρατηρώντας την $S=? \text{ (exp} \leq 0.05)$ του πίνακα 4.2 βλέπουμε ότι τα ποσοστά έχουν μειωθεί σε σχέση με την πρώτη ιδιότητα μέχρι και 20%. Λαμβάνοντας

υπόψη τις πιο πάνω παρατηρήσεις συμπεραίνουμε ότι η φήμη του πωλητή κυμαίνεται στα ίδια επίπεδα με τη φύση του με τη πάροδο του χρόνου άσχετα από το γεγονός ότι η πρώτη ιδιότητα μας δίνει συγκριτικά μικρότερες πιθανότητες.

Μελετώντας τις πιο πάνω περιπτώσεις αυτού του πειράματος παρατηρούμε ότι στην περίπτωση που ο πωλητής έχει φύση ίση με 0.5 παίρνουμε τα χαμηλότερα ποσοστά σε σχέση με τις υπόλοιπες τιμές της φύσης του πωλητή. Αυτό έρχεται σε αντίθεση με τα αποτελέσματα που πάρθηκαν από τα πειράματα της ιδιότητας 1 αφού σε αυτό στην ιδιότητα 2 φαίνεται ότι το 0.5 δίνει τα χειρότερα αποτελέσματα, έτσι το 0.5 δεν είναι η καλύτερη στρατηγική. Όποτε η ιδιότητα αυτή μας δίνει τα καλύτερα αποτελέσματα όταν οι τιμή της φύσης του πωλητή βρίσκεται στα άκρα (πχ 0.9 και 0.1) και στους τέσσερις αλγόριθμους. Συγκρίνοντας και τους τέσσερις αλγορίθμους μεταξύ τους παρατηρούμε ότι ο αλγόριθμος του ebay όταν το μοντέλο είναι ειλικρινές παρουσιάζει λίγο καλύτερα αποτελέσματα από τους υπόλοιπους.

			TNA-SL honest	eBay honest	TNA-SL dishonest	eBay dishonest
Property	s_nature	a	Probability	Probability	Probability	Probability
S=? [exp<=0.1]	0.1	0.5	0.998	0.999	0.997	0.997
	0.2	0.5	0.986	0.991	0.986	0.991
	0.3	0.5	0.97	0.978	0.97	0.978
	0.4	0.5	0.968	0.968	0.968	0.968
	0.5	0.5	0.964	0.964	0.964	0.964
	0.6	0.5	0.968	0.968	0.968	0.968
	0.7	0.5	0.97	0.978	0.97	0.978
	0.8	0.5	0.986	0.991	0.986	0.991
	0.9	0.5	0.998	0.991	0.998	0.999
S=? [exp<=0.1]	0.1	1	0.995	-	0.995	-
	0.2	1	0.979	-	0.979	-
	0.3	1	0.961	-	0.961	-
	0.4	1	0.964	-	0.964	-
	0.5	1	0.961	-	0.961	-
	0.6	1	0.965	-	0.965	-
	0.7	1	0.971	-	0.971	-
	0.8	1	0.988	-	0.988	-
	0.9	1	0.999	-	0.999	-
S=? [exp<=0.05]	0.1	0.5	0.937	0.936	0.903	0.936
	0.2	0.5	0.788	0.832	0.788	0.832
	0.3	0.5	0.723	0.77	0.723	0.77
	0.4	0.5	0.69	0.738	0.69	0.738
	0.5	0.5	0.728	0.728	0.728	0.728
	0.6	0.5	0.69	0.738	0.69	0.738
	0.7	0.5	0.723	0.77	0.723	0.77
	0.8	0.5	0.788	0.832	0.788	0.832
	0.9	0.5	0.903	0.936	0.903	0.936
S=? [exp<=0.05]	0.1	1	0.868	-	0.868	-
	0.2	1	0.764	-	0.763	-
	0.3	1	0.703	-	0.703	-
	0.4	1	0.672	-	0.672	-
	0.5	1	0.719	-	0.719	-
	0.6	1	0.689	-	0.689	-
	0.7	1	0.72	-	0.72	-
	0.8	1	0.784	-	0.784	-
	0.9	1	0.902	-	0.902	-

Πίνακας 4.2: Αποτελέσματα πειράματος με την ιδιότητα 2

Κεφάλαιο 5

Συμπεράσματα και μελλοντικές επεκτάσεις

5.1 Γενικά	59
5.2 Μελλοντικές επεκτάσεις	60

5.1 Συμπεράσματα

Στις μέρες μας η χρήση εφαρμογών ηλεκτρονικού εμπορίου έγινε απαραίτητη. Σε αυτές τις εφαρμογές η επιτυχία ή όχι ενός συστήματος κρίνεται σε μεγάλο βαθμό από την εμπιστοσύνη γενικότερα αλλά και την δυνατότητα να δημιουργηθεί η εμπιστοσύνη στον αγοραστή. Με γνώμονα αυτό θα πρέπει να υπάρχει ένα σύστημα φήμης το οποίο χρησιμοποιώντας κατάλληλο αλγόριθμο θα παρέχει την κατάλληλη εμπιστοσύνη και αξιοπιστία στο σύστημα μέσα από την αξιολόγηση των χρηστών του συστήματος.

Έτσι με βάση αυτό έγινε μελέτη διάφορων αλγορίθμων φήμης και επιλογή δυο από αυτούς για μοντελοποίηση με σκοπό την εύρεση εκείνου του αλγορίθμου που δίνει αξιοπιστία και εμπιστοσύνη στο σύστημα. Στη συνέχεια αφού έγινε η υλοποίηση των αλγορίθμων στο εργαλείο PRISM πραγματοποιήθηκαν πειραματικές μελέτες και εξήχθηκαν κάποια συμπεράσματα.

Συμφώνα με μελέτες που έγιναν στα εν ενεργεία συστήματα φήμης φάνηκε ότι οι αγοραστές φοβούνται να δώσουν αρνητική εκτίμηση σε κάποιον πωλητή που είχε κακή συμπεριφορά επειδή φοβούνται την μελλοντική συμπεριφορά του πωλητή προς εκείνους [5]. Έτσι συμπεραίνουμε βάσει των πειραμάτων που έγιναν και των αποτελεσμάτων που εξάχθηκαν ότι και οι δυο αλγόριθμοι (TNA-SL με no pre-

trusted και eBay) μας δίνουν παρόμοια αποτελέσματα της φήμης των χρηστών και μπορούν να χρησιμοποιηθούν στα εν ενεργεία συστήματα φήμης. Ο TNA-SL με non pre-trusted χρήστες αλλά και ο αλγόριθμος του eBay απομονώνουν τους κακούς χρήστες διατηρώντας τη φήμη τους σε χαμηλά επίπεδα και προστατεύουν του πολύ καλούς χρήστες ανταμείβοντας τους με καλή φήμη. Ίσως ο αλγόριθμος του TNA-SL να προτιμηθεί περισσότερο από αυτόν του eBay λόγω του ότι υπολογίζει την φήμη του κάθε χρήστη όχι μόνο μέσω των εκτιμήσεων που δίνονται αλλά λαμβάνοντας υπόψη και την εμπιστοσύνη μεταξύ των χρηστών κάτι που στο eBay δεν γίνεται.

5.2 Μελλοντικές επεκτάσεις

Επιπλέον επέκταση της παρούσας εργασίας θα μπορούσε να είναι η υλοποίηση ενός συστήματος φήμης χρησιμοποιώντας άλλους αλγορίθμους όπως ο Eigen Trust ο οποίος χρησιμοποιεί την έννοια της μεταβατικής εμπιστοσύνης, δηλαδή βασίζει την φήμη του πωλητή ή κάποιου κόμβου στο δίκτυο στην γνώμη που έχουν οι ενδιαμέσες οντότητες που συναλλάχτηκαν μαζί του στο παρελθόν. Έτσι αναμένουμε ότι ο EigenTrust θα είναι καλύτερος από τους δύο αλγορίθμους που αναλύθηκαν, διότι βασίζεται στην γνώμη που έχουν οι ενδιαμέσοι χρήστες του συστήματος για κάποιο τελικό χρήστη Α ο οποίος θα κάνει συναλλαγή με κάποιο άλλο χρήστη Β. Με βάση αυτό ο αλγόριθμος ενδεχομένως να δώσει πιο αληθινές και αξιόπιστες τιμές εμπιστοσύνης προς τους χρήστες.

Βιβλιογραφία

- [1] http://en.wikipedia.org/wiki/Reputation_system
- [2] Resnick P., Kuwabara K., Zeckhauser R., Friedman E., “Reputation Systems”, *Communication of the ACM* , vol.43 (12), pp. 45-48, 2000.
- [3] M. Kinader and K. Rothermel, “Architecture and Algorithms for a Distributed Reputation System,” *in Proc. of the 1st International Conference on Trust Management*, pp.1–16, 2003.
- [4] Andrew G. West, Adam J. Aviv, Jian Chang, Vinayak S. Prabhu, Matt Blaze, Sampath Kannan, Insup Lee, Jonathan M. Smith, and Oleg Sokolsky, “Quantm: a quantitative trust management system”, *in Proc. of the 2nd European Workshop on System Security of the ACM*, pp. 28-35, 2009.
- [5] P. Resnick and R. Zeckhauser, “Trust Among Strangers in Internet Transactions: Empirical Analysis of eBay’s Reputation System”, *The Economics of the Internet and E-Commerce of Advances in Applied Microeconomics*, Emerald Group Publishing Limited, 2002, vol. 11, pp. 127-157.
- [6] A. Josang, R. Ismail, and C. Boyd, “A Survey of Trust and Reputation Systems for Online Service Provision,” *Decision Support Systems*, 2005.
- [7] Εγχειρίδιο χρήσης του εργαλείου PRISM
- [8] Σημειώσεις από το μάθημα «ΕΠΛ 654 Ανάλυση και Επαλήθευση συστημάτων ΕΠΛ 654» ,Άννα Φιλίππου, 2010
- [9] <http://en.wikipedia.org/wiki/Digg>
- [10] <http://web.archive.org/web/20080329121633/http://digg.com/about/>
- [11] http://www.yelp.com/faq#what_is_yelp
- [12] <http://pages.ebay.com/help/feedback/allaboutfeedback.html>
- [13] A. G. West, S. Kannan, I. Lee, and O. Sokolsky, “An evaluation framework for reputation management systems”. Working chapter for Trust Modeling and Management in Digital Environments: From Social Concept to System Development, (Zheng Yan, and ed.).

- [14] M. Huth and A. Ryan. Logic in Computer Science: Modeling and Reasoning about Concurrent Systems. Cambridge University Press, 2000.
- [15] M. Kwiatkowska, G. Norman, D. Parker, Stochastic model checking, in: M. Bernardo, J. Hillston (Eds.), Formal Methods for the Design of Computer, Communication and Software Systems: Performance Evaluation (SFM'07), Vol. 4486 of Lecture Notes in Computer Science (Tutorial Volume), Springer, 2007, pp.220–270
- [16] http://en.wikipedia.org/wiki/Markov_chain

Παράρτημα

Ειλικρινές μοντέλο

dtmc

```
const      double s_nature;  
const int no_transc=100;  
const double a;
```

```
module Buyer1
```

```
b1: [0..10] init 1; // states of module  
f1:[0..2]; // 0: no feedback, 1:good feedback,  
           2: bad feedback  
  
// if buyer receives good item sends bad feedback  
[] avl1=1 & b1=1 -> (f1'=1) & (b1'=1);  
// if buyer receives bad item sends bad feedback  
[] avl1=2 & b1=1 -> (f1'=2) & (b1'=1);
```

```
endmodule
```

```
module Seller
```

```
s:[0..10] init 1;  
avl1:[0..2];  
good_score:[0..no_transc];  
bad_score:[0..no_transc];  
transc:[0..no_transc] init 0;
```

```

[] b1=1 & s=1 -> s_nature:(av11'=1) & (s'=2) + (1-
s_nature):(av11'=2) & (s'=3);
[] f1=1 & s=2 & (good_score + bad_score < no_transc) & (transc <
no_transc) ->(good_score'=good_score + 1) & (transc'= transc +1)
& (av11'=0) & (s'=1); // if seller receives good feedback from
buyer then his/her good score increase by 1
[] f1=2 & s=3 & (good_score + bad_score < no_transc) & (transc <
no_transc) -> (bad_score'=bad_score + 1) & (transc'= transc +1)
&(av11'=0) & (s'=1); // if seller receives bad feedback from buyer
then his/her good score increase by 1

endmodule

// eBay algorithm
formula rep = good_score / (good_score + bad_score);

// TNA-SL lagorithm
formula belief = good_score / (good_score + bad_score + 2);
formula uncertainly = 2 / (good_score + bad_score + 2);
formula repl=belief + uncertainly*a;

formula exp= (s_nature - repl)>0 ? s_nature - repl: - (s_nature -
repl);

```

Ανελικρινές μοντέλο

dtmc

```
const      double s_nature;  
const int no_transc=100;  
const      double honest=1/10; // honest buyer  
const double positive = 9/10; // dishonest good buyer  
const double negative = 0/10; // dishonest bad buyer  
const double a;  
  
// eBay algorithm  
formula rep = good_score / (good_score + bad_score);  
  
// TNA-SL lagorithm  
formula belief = good_score / (good_score + bad_score + 2);  
formula uncertainly = 2 / (good_score + bad_score + 2);  
formula repl=belief + uncertainly*a;  
formula exp= (s_nature - repl)>0 ? s_nature - repl: - (s_nature -  
repl);  
  
module Buyer1  
  
b1:[0..10] init 1; // states of module  
f1:[0..2]; // 0: no feedback, 1:good feedback, 2: bad feedback  
honest1:[0..no_transc];  
dishonest1:[0..no_transc];  
  
[] b1=1 -> honest:(b1'=2) + positive:(b1'=3) + negative:(b1'=4);  
[] avl1=1 & b1=2 -> (f1'=1) & (b1'=1); // if buyer receives good  
item sends bad feedback  
[] avl1=2 & b1=2 -> (f1'=2) & (b1'=1); // if buyer receives bad  
item sends bad feedback  
[] b1=3 & avl1 > 0-> (f1'=1) & (b1'=1);
```

```

[] b1=4 & av11 > 0-> (f1'=2) & (b1'=1);

endmodule

module Seller

s:[0..10] init 1;
av11:[0..2]init 0;
good_score:[0..no_transc]init 0;
bad_score:[0..no_transc]init 0;
//transc:[0..no_transc] init 0;

[] b1>1 & s=1 -> s_nature:(av11'=1) & (s'=2) + (1-
s_nature):(av11'=2) & (s'=3); // if seller transact with
                                buyer1 then he/she sends a good
                                or bad item depends on
                                his/her nature

[] f1=1 & s=2 & (good_score + bad_score < no_transc) ->
(good_score'=good_score + min(1,10)) & (av11'=0) & (s'=1); // if
seller receives good feedback from buyer then his/her good score
increase by 1

[] f1=2 & s=3 & (good_score + bad_score < no_transc) ->
(bad_score'=bad_score + min(1,10)) & (av11'=0) & (s'=1); // if
seller receives bad feedback from buyer then his/her good score
increase by 1

endmodule

```