



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μεταφορά Δεδομένων με Χρήση Τεχνητής Νοημοσύνης σε Συστήματα Κινητού Υπολογισμού

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

ΣΤΑΥΡΟΥ ΣΤΑΥΡΟΥ

Επιβλέπων: Συμεών Παπαβασιλείου
Καθηγητής Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2020



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Επικοινωνιών, Ηλεκτρονικής και Συστημάτων Πληροφορικής

Μεταφορά Δεδομένων με Χρήση Τεχνητής Νοημοσύνης σε Συστήματα Κινητού Υπολογισμού

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΣΤΑΥΡΟΥ ΣΤΑΥΡΟΥ

Επιβλέπων: Συμεών Παπαβασιλείου
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 23η Σεπτεμβρίου 2020.

(Υπογραφή)

(Υπογραφή)

(Υπογραφή)

.....

Συμεών Παπαβασιλείου
Καθηγητής Ε.Μ.Π.

.....

Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

.....

Ιωάννα Ρουσσάκη
Επίκουρη Καθηγήτρια Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2020



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Επικοινωνιών, Ηλεκτρονικής και Συστημάτων Πληροφορικής

Copyright ©–All rights reserved ΣΤΑΥΡΟΥ ΣΤΑΥΡΟΥ, 2020.

Με επιφύλαξη παντός δικαιώματος.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

(Υπογραφή)

.....
ΣΤΑΥΡΟΥ ΣΤΑΥΡΟΥ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

© 2020– All rights reserved

Η ραγδαία τεχνολογική ανάπτυξη που βιώνει η ανθρωπότητα τις τελευταίες δεκαετίες δεν έχει σταματήσει. Πολλές από τις ενδιαφέρουσες νέες τεχνολογίες, όπως οι Έξυπνες Φάρμες ή τα Αυτόνομα Οχήματα, βασίζονται στο πλαίσιο που προσφέρει το Διαδίκτυο των Πραγμάτων που επιτρέπει στις συσκευές να είναι συνδεδεμένες και να αλληλεπιδρούν μεταξύ τους. Ένα εξίσου σημαντικό πλαίσιο για την υλοποίηση αυτών των τεχνολογιών είναι το Υπολογιστικό Νέφος, το οποίο παρέχει υπηρεσίες υψηλών υπολογιστικών απαιτήσεων σε χρήστες και συσκευές με χαμηλότερους πόρους. Λόγω της μεγάλης ζήτησης για τις υπηρεσίες που παρέχει, παρατηρείται αυξημένη κίνηση, κάτι που αναπόφευκτα αυξάνει και τους χρόνους καθυστέρησης. Αυτό επιβαρύνει σημαντικά Υπηρεσίες Πραγματικού Χρόνου, όπως τα Αυτόνομα Οχήματα, τα οποία απαιτούν άμεση απόκριση χωρίς καθυστερήσεις. Έτσι, ο υπολογισμός κοντά στην άκρη του δικτύου με χρήση Edge Computing αποτελεί μια ενδιαφέρουσα λύση. Με την εμφάνιση αρχιτεκτονικών που το υλοποιούν παρουσιάζονται νέες προκλήσεις και η Μεταφορά Δεδομένων είναι, ίσως, η σημαντικότερη από αυτές. Η χρήση των απομακρυσμένων υπολογιστικών πόρων πρέπει να γίνεται όσο το δυνατό πιο αποδοτικά, ώστε όλοι οι εμπλεκόμενοι να απολαμβάνουν τις καλύτερες δυνατές υπηρεσίες. Στην παρούσα διπλωματική, μετά από μια σύντομη ανασκόπηση στις παραπάνω τεχνολογίες, παρουσιάζουμε ένα μοντέλο Μεταφοράς Δεδομένων, με στόχο τη βέλτιστη αξιοποίηση των πόρων που παρέχονται.

Λέξεις Κλειδιά

Διαδίκτυο των Πραγμάτων, Υπολογιστικό Νέφος, Άκρη του Δικτύου, Μεταφορά Δεδομένων, Απομακρυσμένη Επεξεργασία Δεδομένων, Υπολογιστική Άκρη, Υπηρεσίες Πραγματικού Χρόνου, Υπολογιστική Ομίχλη

Abstract

Over the past few decades humanity has experienced rapid technological advancements and this is still the case. Many of the most interesting new technologies, like Smart Farms or Autonomous Vehicles, are based on the Internet of Things framework, which allows devices to be connected and interact with each other. An equally important framework for realising these technologies is Cloud Computing, which offers services of high computational requirements to users and devices with lower resources. Because of the high demand for the services it offers, higher traffic is observed, which inevitably increases latency experienced by the users. This highly burdens Real-Time Applications, like Autonomous Vehicles, which require low response time. Thus, computing near the edge of the network, with Edge Computing, emerges as an interesting solution. With the appearance of Edge Computing Architectures, new challenges emerge, Data Offloading being, perhaps, the most important of them. The usage of remote computing resources should be done in the most efficient way possible, so that everyone experiences the best available services. In this diploma thesis, after a brief summary of the above technologies, we present a Data Offloading model, aiming to achieve optimal utilization of the provided resources.

Keywords

Internet of Things, Cloud Computing, Network Edge, Data Offloading, Edge Computing, Real-Time Applications, Fog Computing

Ευχαριστίες

Η πραγματοποίηση της παρούσας διπλωματικής δεν θα ήταν εφικτή χωρίς τη βοήθεια και την επίβλεψη του καθηγητή ΕΜΠ, κύριου Συμεών Παπαβασιλείου, ιδιαίτερα κατά την περίοδο που αυτή πραγματοποιήθηκε. Ακόμη, δεν θα μπορούσα να ξεχάσω να ευχαριστήσω την επίκουρη καθηγήτρια του University of New Mexico (USA), κυρία Ειρήνη-Ελένη Τσιροπούλου και τον υποψήφιο διδάκτορα Γιώργο Φράγκο, που παρά την απόσταση και τις υποχρεώσεις τους ήταν πάντα διαθέσιμοι για όποια βοήθεια χρειαζόμουν. Τους ευχαριστώ θερμά.

Αφού η διπλωματική αυτή σηματοδοτεί το τέλος της πενταετούς φοίτησης μου στο Εθνικό Μετσόβιο Πολυτεχνείο θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξη και την αγάπη που μου δείχνουν όλα αυτά τα χρόνια. Τέλος, ένα μεγάλο ευχαριστώ στους φίλους μου Μηνά, Νεόφυτο, Σάββα, Λένο, Ειρήνη, Μαρία, Κωνσταντίνο, Χρυσάνθια, Κατερίνα και Στέλλα που έκαναν τα φοιτητικά μου χρόνια μια από τις ομορφότερες περιόδους της ζωής μου.

Περίληψη	1
Abstract	3
Ευχαριστίες	5
Περιεχόμενα	9
Κατάλογος Σχημάτων	11
1 Εισαγωγή	13
1.1 Πρόλογος	13
1.2 Σχετική Έρευνα	13
1.3 Διάρθρωση της Διπλωματικής Εργασίας	14
2 Εισαγωγικά Στοιχεία για το Internet of Things και το Cloud Computing	15
2.1 Internet of Things (IoT)	15
2.1.1 Πρώτη Γενιά	15
2.1.2 Δεύτερη Γενιά	17
2.1.3 Τρίτη Γενιά	17
2.1.4 Ορισμός του IoT	17
2.2 Απομακρυσμένη Επεξεργασία Δεδομένων	18
2.2.1 Cloud Computing	18
2.2.2 Edge Computing	20
3 Εισαγωγή στο πρόβλημα μεταφοράς δεδομένων	25
3.1 Όροι που Συναντώνται	25
3.2 Ανασκόπηση της Υπάρχουσας Βιβλιογραφίας	26

3.2.1	Μεταφορά δεδομένων σε συτήματα κινητού υπολογισμού με βάση την τεχνολογία SDN	26
3.2.2	Αυτόνομο σύστημα κινητού υπολογισμού με στόχο την ικανοποίηση της Ποιότητας της Υπηρεσίας	27
3.2.3	MEC servers σε drone με χρήση τεχνητής νοημοσύνης	29
3.2.4	Μεταφορά δεδομένων με επίγνωση ρίσκου	30
3.3	Μεταφορά δεδομένων με μεταβαλλόμενη φήμη	31
4	Αναλυτική Παρουσίαση Μοντέλου και Επί Μέρους Συστατικών	33
4.1	Εργαλεία και Μηχανισμοί που χρησιμοποιούνται	33
4.1.1	Κατανεμημένη Μάθηση σε Δίκτυα Επικοινωνίας	33
4.1.2	Τα αυτόματα SLA	34
4.1.3	Μηχανισμοί Φιλαλήθων Δημοσκοπήσεων	35
4.1.4	Ο μηχανισμός Bayesian Truth Serum	35
4.1.5	Ο μηχανισμός RBTS	37
4.1.6	Μοντελοποίηση Φήμης	38
4.2	Αναλυτική Παρουσίαση του Μοντέλου	39
4.2.1	Μεγέθη και Παίχτες του Μοντέλου	40
4.2.2	Πρώτο Επίπεδο - Επιλογή Εξυπηρετητή από Χρήστες	41
4.2.3	Δεύτερο Επίπεδο - Αξιολόγηση Εξυπηρετητών	43
4.2.4	Τρίτο Επίπεδο - Ενημέρωση Φήμης Εξυπηρετητή	44
4.2.5	Σύνοψη Αλγορίθμου	45
5	Αριθμητικά Αποτελέσματα	47
5.1	Αποτελέσματα Μοντέλου	47
5.1.1	Τιμές Σταθερών και Περιβάλλοντος	47
5.1.2	Πρώτο Επίπεδο - Αποτελέσματα Μηχανισμού Επιλογής	48
5.1.3	Χρήστες και Ανταμοιβή	50
5.1.4	Δεύτερο Επίπεδο - Αξιολόγηση Δημοσκόπησης με RBTS	52
5.1.5	Τρίτο Επίπεδο - Εξέλιξη Φήμης	53
5.2	Συγκριτικά Αποτελέσματα	53
5.2.1	SLA και άλλες Μεθόδους Αντιστοίχισης	53
5.2.2	Αποτελέσματα Μηχανισμού BTS	55
5.3	Κλιμακωσιμότητα του Μηχανισμού Επιλογής	56
6	Σύνοψη - Συμπεράσματα	57
6.1	Σύνοψη	57
6.2	Συμπεράσματα	57
6.3	Μελλοντική Εργασία	58
	Βιβλιογραφία	60

Γλωσσάριο

67

Κατάλογος Σχημάτων

1.1	Ο αριθμός των χρηστών του Cloud το 2013 και το 2018	14
2.1	Η απεικόνιση του IoT σαν ένας ανθρώπινος εγκέφαλος	16
2.2	Η συνύπαρξη Edge και Cloud Computing	19
2.3	Η αρχιτεκτονική MEC	23
3.1	Η αρχιτεκτονική MEC σε περιβάλλον SDN	28
3.2	Η αρχιτεκτονική MEC με χρήση UAV (drone)	30
4.1	Οι επαναλήψεις του μηχανισμού επιλογής συγκριτικά με το υπόλοιπο πλαίσιο .	43
4.2	Μια επισκόπηση του αλγορίθμου	45
5.1	Τα χαρακτηριστικά των εξυπηρετητών	48
5.2	Η εξέλιξη των πιθανοτήτων επιλογής ενός Χρήστη	49
5.3	Αποτελέσματα Εξυπηρετητών	50
5.4	Στοιχεία Ανταμοιβής ανά Εξυπηρετητή	51
5.5	Μέση Κανονικοποιημένη Ανταμοιβή	52
5.6	Συνολικά Αποτελέσματα Τρίτου Επιπέδου	53
5.7	SLA και Άλλες Μέθοδοι Επιλογής	54
5.8	Στοιχεία BTS	55
5.9	Κλιμακωσιμότητα Πρώτου Επιπέδου	56

1.1 Πρόλογος

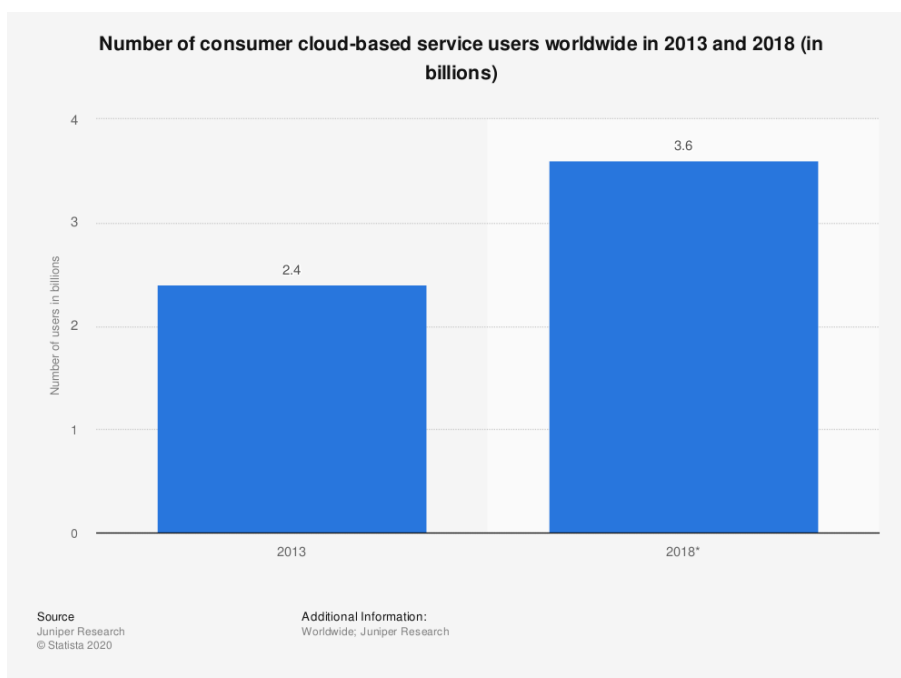
Η εμφάνιση του Cloud Computing και των υπηρεσιών που προσφέρει έλυσε τα χέρια χρηστών με περιορισμένο υλικό. Με τα χρόνια ο αριθμός των χρηστών αυτών έχει αυξηθεί φτάνοντας, σύμφωνα με το Statista τα 3.6 δισεκατομμύρια (Σχήμα 1.1). Ακόμη, ολοένα και περισσότερες συσκευές συνδέονται στο Internet of Things (IoT). Σύμφωνα με την ετήσια έκθεση της Cisco για την πενταετία 2018-2023, μέχρι το 2023 περισσότερες από 14 δισεκατομμύρια συσκευές θα είναι συνδεδεμένες σε αυτό [Cis20]. Όσο το μέγεθος του IoT αυξάνεται, αυξάνεται και η ανάγκη των συνδεδεμένων σε αυτό συσκευών για πρόσβασή τους στο Cloud είτε για αποθήκευση είτε για χρήση υπολογιστικών πόρων. Λόγω της αύξησης αυτής, η μετάδοση δεδομένων στο Cloud γίνεται χρονοβόρα, προκαλώντας επιπλέον καθυστερήσεις, ενώ με την επικείμενη εμφάνιση του 5G η κίνηση αναμένεται να αυξηθεί περαιτέρω. Το Edge Computing επιχρηρεί να εκμηδενίσει τις αποστάσεις και να φέρει τις υπηρεσίες του Cloud στην άκρη του δικτύου, μειώνοντας έτσι την καθυστέρηση που θα είχε η κάθε συσκευή ή ο κάθε χρήστης [ATP20b]. Στη διπλωματική αυτή εξετάζουμε την κατανομή των χρηστών και των συσκευών σε διάφορους Fog Nodes με χρήση διάφορων αλγορίθμων και εργαλείων.

1.2 Σχετική Έρευνα

Το πρόβλημα της Μεταφοράς Δεδομένων είναι, φυσιολογικά, αρκετά πρόσφατο. Παρ' όλ' αυτά υπάρχει αρκετή βιβλιογραφία για το θέμα. Στο [Mit+19] έχουμε τη χρήση της αρχιτεκτονικής SDN για αποδοτική μεταφορά δεδομένων. Στα [Fra+19], [Mes+17], [Mes+19] έχουμε τη διατύπωση ενός περιβάλλοντος όπου η Μεταφορά Δεδομένων γίνεται σε MEC Servers που βρίσκονται πάνω σε Unmanned Aerial Vehicles (UAV). Στο [ATP18] μελετάται η ικανοποίηση προδιαγραφών QoS από τους χρήστες για την εξυπηρέτησή τους από τους MEC servers.

Σε όλα τα πιο πάνω υιοθετείται μια παιχνοδοθεωρητική οπτική με χαρακτηρισμό των παικτών και των κατάλληλων συναρτήσεων ωφελείας, συνοδευόμενα από μια ισορροπία Nash και ένα αλγόριθμο που οδηγεί σε αυτήν. Ωστόσο, θεωρείται πως οι παίχτες που συμμετέχουν σε

Σχήμα 1.1: Ο αριθμός των χρηστών του Cloud το 2013 και το 2018



αυτά είναι απολύτως λογικοί, κάτι που όσο οι συσκευές προσπαθούν να μιμηθούν τις συμπεριφορές των ανθρώπων πιθανόν να μην ισχύει. Λόγω αυτού, βλέπουμε σε διάφορες δημοσιεύσεις τη χρήση της Prospect Theory [Tve00], προσπαθώντας να μοντελοποιήσουν το ρίσκο που λαμβάνει ο χρήστης με πιθανή αποτυχία του server. Στο [MTP20] έχουμε τη χρήση UAV με παίχτες με πιο ανθρώπινη συμπεριφορά. Μια άλλη προσέγγιση, πάλι πιο κοντά στην ανθρώπινη συμπεριφορά, δίνεται στο [ATP20a].

Μερικά από τα πιο πάνω μελετώνται περαιτέρω στο κεφάλαιο 3.

1.3 Διάρθρωση της Διπλωματικής Εργασίας

Στο δεύτερο κεφάλαιο δίνονται μια σειρά από ορισμούς και μια σύντομη περιγραφή για το Internet of Things, το Cloud Computing κ.ά. ώστε ο αναγνώστης να εισαχθεί στο περιβάλλον του προβλήματος που αφορά η εργασία. Στο τρίτο κεφάλαιο παρουσιάζεται το πρόβλημα της Μεταφοράς Δεδομένων. Αρχικά, γίνεται μια ανασκόπηση της βιβλιογραφίας για διάφορες μορφές του προβλήματος και ακολούθως παρουσιάζεται συνοπτικά η συγκεκριμένη μορφή του προβλήματος. Στο τέταρτο κεφάλαιο παρουσιάζονται τα επί μέρους συστατικά του μοντέλου και γίνεται η αναλυτική παρουσίασή του. Στο πέμπτο κεφάλαιο παραθέτουμε τα αριθμητικά αποτελέσματα που λαμβάνουμε από αυτό. Τα τελικά συμπεράσματα συνοψίζονται στο έκτο κεφάλαιο.

Εισαγωγικά Στοιχεία για το Internet of Things και το Cloud Computing

Στο κεφάλαιο αυτό γίνεται μια σύντομη εισαγωγή στο περιβάλλον του προβλήματος. Παρουσιάζεται μέσω μιας μικρής ιστορικής αναδρομής η εξέλιξη του IoT καταλήγοντας σε ένα σύντομο ορισμό του. Ακόμη, παρουσιάζονται οι λειτουργίες που προσφέρει το Cloud Computing, καθώς και οι προσπάθειες μέσω του Edge Computing και των αρχιτεκτονικών του να μεταφέρουμε την επεξεργαστική υπολογιστική δύναμη πιο κοντά στο χρήστη.

2.1 Internet of Things (IoT)

Στο πέρασμα των χρόνων, μεγάλες προσωπικότητες της πληροφορικής, όπως ο Alan Turing και ο Nikola Tesla ανέπτυξαν το όραμα για έξυπνες συσκευές συνδεδεμένες σε ένα μεγάλο παγκόσμιο «εγκέφαλο» [19] (Σχήμα 2.1). Η ιδέα, ωστόσο, ενός δικτύου, όπου διάφοροι υπολογιστές θα μπορούν να επικοινωνούν μεταξύ τους ανταλλάζοντας ακριβή δεδομένα και εξάγοντας πληροφορίες για ανάγκες αντικατάστασης ή/και επισκευής διατυπώθηκε για πρώτη φορά από τον Kevin Ashton το 1999 [Ash09]. Συγκεκριμένα, ο Ashton πρότεινε τη χρήση της νέας (για την εποχή) τεχνολογίας των ετικετών RFID (Radio-frequency identification), για να επιτύχει τη σύνδεση διάφορων αντικειμένων (things) στο διαδίκτυο. Έκτοτε, ο όρος και το όραμα της χρήσης του IoT έχει επεκταθεί σε διάφορους τομείς της κοινωνίας, όπως οι έξυπνες φάρμες (Smart Farms) [Ryu+15], έξυπνα συστήματα στάθμευσης [Tsi+17], και η ανταλλαγή δεδομένων ασθενών σε πραγματικό χρόνο στον τομέα της υγείας [USZ16]. Παρ' όλ' αυτά, η επέκταση του έχει ακόμα αρκετά εμπόδια, τα οποία αναμένεται να ξεπεραστούν στις επόμενες γενιές του. Μέχρι στιγμής μπορούμε να διακρίνουμε τρεις γενιές τις οποίες παρουσιάζουμε συνοπτικά [AIM16].

2.1.1 Πρώτη Γενιά

Η πρώτη γενιά του IoT ξεκινά με την εφεύρεση των ετικετών RFID, ετικέτες μικρής ενεργειακής απαίτησης με δυνατότητα ασύρματης επικοινωνίας χαρακτηριζόμενες, επίσης, από ένα ξεχωριστό ID. Με την εφεύρεση των ετικετών προτείνεται το 1999 από τον Kevin Ashton

2.1.2 Δεύτερη Γενιά

Στη δεύτερη γενιά του IoT το όραμα για διασύνδεση όλων των αντικειμένων στον ίδιο «εγκέφαλο» εξελίσσεται. Προτείνεται η χρήση του υπάρχοντος IP πρωτοκόλλου (IPv4), με σκοπό κάθε αντικείμενο να έχει τη δική του διεύθυνση και δυνατότητα σύνδεσης στο διαδίκτυο. Με την ανάπτυξη του IPv6 υπάρχουν, πλέον, αρκετές διευθύνσεις, ώστε στο μέλλον αυτές να αρκούν για κάθε αντικείμενο (περίπου $3.4 \cdot 10^{38}$ διευθύνσεις ανά τ.μ.). Παράλληλα με τη διευθυνοδοσία όλων των αντικειμένων, η απλότητα που προσφέρει το πρωτόκολλο HTTP, καθώς και η ευρεία χρήση του στην ανάπτυξη εφαρμογών ιστού, προκρίνει τη χρήση του και στο IoT, ώστε τα δεδομένα που συλλέγουν τα αντικείμενα να είναι άμεσα προσβάσιμα μέσω αυτού. Τέλος, εξερευνάται για πρώτη φορά, η δυνατότητα των ανθρώπων να μοιράζονται τα δεδομένα που συλλέγονται στα αντικείμενά τους με άλλους ανθρώπους που εμπιστεύονται.

2.1.3 Τρίτη Γενιά

Η τρίτη γενιά του IoT είναι και η τρέχουσα και, ως εκ τούτου, κάποιες από τις ιδέες της δεν έχουν υλοποιηθεί ακόμη ή βρίσκονται σε πρώιμο στάδιο. Η κυριότερη από αυτές αφορά τη χρήση του Cloud, όχι απλά ως μέσο διεκπεραίωσης κοστοβόρων υπολογισμών που δεν μπορούν να γίνουν στα αντικείμενα, αλλά και για τη δημιουργία εικονικών αναπαραστάσεων των φυσικών αντικειμένων που θα αποθηκεύονται σε αυτό. Εξασφαλίζεται έτσι πως, έστω και εικονικά, τα αντικείμενα είναι πάντα σε λειτουργία. Ακόμη, ένας μεγάλος στόχος είναι η δημιουργία κοινωνικών δικτύων αντικειμένων σε αντιστοιχία με αυτά των ανθρώπων, όπου τα αντικείμενα θα αναπτύσσουν σχέσεις με άλλα αντικείμενα που εμπιστεύονται, ώστε συλλογικά να παίρνουν αποφάσεις και να παράγουν (ακόμη περισσότερη) πληροφορία [Atz+12], [ATP19]. Επιπρόσθετα, επιδιώκεται η πλήρης ενσωμάτωση των λύσεων βασισμένων στις ετικέτες RFID σαν μια μικρή υποκατηγορία των λύσεων που προσφέρει το IoT. Τέλος, γίνεται (σε πιο θεωρητικό υπόβαθρο) η προσπάθεια να καθοριστούν οι λειτουργίες που μπορεί να προσφέρει το IoT, ώστε να προωθηθεί η δυνατότητα διαλειτουργικότητας μεταξύ διαφορετικών συστημάτων που επιθυμούν να το χρησιμοποιήσουν.

2.1.4 Ορισμός του IoT

Βλέποντας τα πιο πάνω, μπορεί κανείς να συμπεράνει ότι νέες τεχνολογίες σηματοδοτούν την έναρξη μιας νέας γενιάς για το IoT. Βάσει αυτού, η τρίτη γενιά (μάλλον) δεν θα είναι η τελευταία. Ειδικότερα, βλέπουμε την άνθιση τεχνολογιών όπως το Cloud Computing και το Fog Computing, την ανάπτυξη των δικτύων 5G, αλλά και την εγγενή ανάγκη του IoT για βελτιώσεις (όπως η ανίχνευση και συσχέτιση όλων των αντικειμένων που ανήκουν στον ίδιο άνθρωπο). Όλες οι πιο πάνω τεχνολογίες αναμένεται να παίξουν σημαντικό ρόλο σε κάποια από τις επομενες γενιές του IoT.

Είναι δύσκολο να αποδώσει κανείς κάποιο ορισμό σε κάτι που εξελίσσεται τόσο ραγδαία. Παρ' όλ' αυτά, από τους Atzori, Iera και Morabito στο [AIM16] δίνεται ο πιο κάτω ορισμός.

Ορισμός 2.1. Το *Internet of Things* είναι ένα εννοιολογικό πλαίσιο που αξιοποιεί το

διαθεσιμότητα ετερογενών συσκευών και λύσεων διασύνδεσης, καθώς και επαυξημένα φυσικά αντικείμενα, παρέχοντας μια κοινή βάση πληροφοριών σε παγκόσμια κλίμακα, για την υποστήριξη του σχεδιασμού εφαρμογών εμπλέκοντας στο ίδιο εικονικό επίπεδο τόσο ανθρώπους όσο και αναπαραστάσεις αντικειμένων.

2.2 Απομακρυσμένη Επεξεργασία Δεδομένων

Με την εξάπλωση του Ηλεκτρονικού Υπολογιστή (Η/Υ) σε κάθε σπίτι και εργασιακό χώρο πολλοί τομείς της καθημερινότητάς μας έχουν αλλάξει. Για την ψυχαγωγία ο Η/Υ είναι πλέον στην κορυφή των προτιμήσεων, ενώ όλο και περισσότερες εταιρείες μεταφέρονται ηλεκτρονικά [Net17]. Οι χρήστες, όμως, παραμένουν περιορισμένοι από τις δυνατότητες του υλικού τους για την ποιότητα της ψυχαγωγίας τους. Την ίδια ώρα οι εργοδότες αναγκάζονται να προσλάβουν άτομα, που πιθανόν να μην έχουν καμία σχέση με τις υπηρεσίες που προσφέρει η εταιρεία, ώστε να επιτυγχάνεται η συντήρηση του εικονικού προσώπου της επιχείρησης.

Πλέον, στη σύγχρονη εποχή, οι ανάγκες των ανθρώπων για γρήγορη και άμεση χρήση υπηρεσιών αυξάνονται, όσο αυξάνονται και οι τομείς τους οποίους καλύπτουν οι προσφερόμενες υπηρεσίες. Καθημερινοί άνθρωποι μπορούν να επιλέξουν να περάσουν τον ελεύθερο τους χρόνο παίζοντας παιχνίδια, όχι με τον παραδοσιακό τρόπο, αλλά μέσω ενός stream με το παιχνίδι να εκτελείται σε ένα απομακρυσμένο server [Heu+17]. Αυτό επιτρέπει σε χρήστες με μια καλή σύνδεση στο διαδίκτυο και περιορισμένη ισχύ υλικού να απολαμβάνουν πολύ καλύτερη εμπειρία παιχνιδιού. Την ίδια ώρα εταιρείες κάθε μορφής και είδους μπορούν να αναθέσουν την αποθήκευση και συντήρηση των δεδομένων των πελατών τους σε απομακρυσμένους servers αποφορτίζοντας το τμήμα IT της κάθε μίας από αυτές [Mao+17].

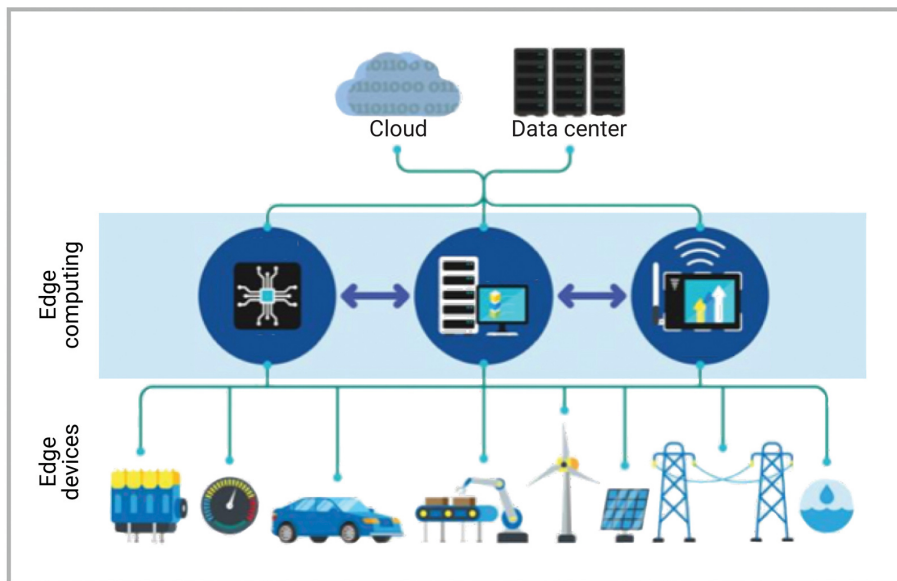
Τα πιο πάνω είναι το αποτέλεσμα της ανάπτυξης του cloud computing, με το edge computing να το ακολουθεί, ώστε να μειωθούν ακόμα περισσότερο οι χρονοκαθυστερήσεις. Μπορούμε να φανταστούμε την ύπαρξη του edge computing ως ένα ενδιάμεσο στάδιο μεταξύ των τερματικών συσκευών και του cloud όπως φαίνεται στο σχήμα 2.2. Στη διπλωματική αυτή, θα μελετήσουμε περισσότερο το δεύτερο, χωρίς όμως να παραλείψουμε να αναφερθούμε συνοπτικά στα σημαντικά σημεία του πρώτου.

2.2.1 Cloud Computing

Η πρώτη εμφάνιση του Cloud Computing χρονολογείται στη δεκαετία του 1960 από τον John McCarthy με τον όρο "Utility Computing". Υπό αυτό το πρίσμα, οι υπολογιστικοί πόροι προσφέρονται σαν μια ακόμα υπηρεσία κοινής ωφελείας, παρά σαν ένα προϊόν [SS14]. Έκτοτε, το όραμα του Cloud Computing έχει πάρει σάρκα και οστά, με διάφορα συγκεκριμένα μοντέλα και αρχιτεκτονικές να καθορίζουν τη λειτουργία του.

Κατ' αρχάς, μπορεί κανείς να διακρίνει τρία είδη συστατικών-ρόλων. Ο βασικότερος από αυτούς είναι οι πελάτες (**Clients**). Ως πελάτη, χαρακτηρίζουμε κάθε τερματική συσκευή η οποία χρησιμοποιείται για διαχείριση ή πρόσβαση στα δεδομένα και τις υπηρεσίες που υπάρχουν στο Cloud. Οι υπηρεσίες αυτές φιλοξενούνται σε μια σειρά από servers, οι οποίοι συνολικά

Σχήμα 2.2: Η συνύπαρξη Edge και Cloud Computing



αποτελούν ένα **Data Center**. Συχνά έχουμε σύνδεση Data Centers σε δίκτυα, τα οποία ονομάζουμε Data Center Networks (DCNs) και τα οποία στον έξω κόσμο εμφανίζονται σαν ένα ενιαίο Data Center με χαμηλή χρονοκαθυστέρηση εντός του δικτύου. Τέλος, για αύξηση της αξιοπιστίας της διαθεσιμότητας της κάθε εφαρμογής ή υπηρεσίας που διατίθεται μέσω αυτών χρησιμοποιούνται γεωγραφικά διάσπαρτοι servers (**Distributed Servers**).

Σύμφωνα με το National Institute of Standards and Technology (NIST) [MG11] οι υπηρεσίες αυτές πρέπει να διέπονται από πέντε αναγκαία χαρακτηριστικά. Αρχικά, κάθε υπηρεσία θα πρέπει να είναι προσβάσιμη στους πελάτες της με μια απλή σύνδεση στο διαδίκτυο (**Broad Network Access**). Επιπρόσθετα, θα πρέπει να φροντίζει για την παροχή υπολογιστικών πόρων στον πελάτη, χωρίς να χρειάζεται ανθρώπινη παρέμβαση για αυτό (**On-Demand Self-Service**), ενώ παράλληλα θα πρέπει να υπάρχει η δυνατότητα για αύξηση (ή μείωση) των υπολογιστικών πόρων που παρέχονται ανάλογα με τη ζήτηση (**Rapid Elasticity**). Ακόμη, η ανάθεση των υπολογιστικών πόρων στους πελάτες πρέπει να γίνεται δυναμικά με βάση τις ανάγκες τους ανά πάσα στιγμή (**Resource Pooling**). Τελικά, η χρήση των πόρων θα καταμετράται (**Measured Service**), ώστε να εξαχθούν συμπεράσματα για τη λειτουργία του συστήματος, όπως η βελτιστοποίηση πόρων ή η εξαγωγή χρέωσης για κάθε πελάτη.

Η έκθεση του NIST συνεχίζει υποδεικνύοντας τρία μοντέλα με βάση τα οποία θα πρέπει οι πόροι να παρέχονται στους πελάτες. Το πιο συνηθισμένο στην καθημερινότητα είναι το **Software as a Service (SaaS)** και αφορά την απλή διάθεση υπηρεσιών μέσω ενός φυλλομετρητή ή κάποιας εφαρμογής (π.χ. Dropbox). Με το δεύτερο μοντέλο, **Platform as a Service (PaaS)**, παρέχονται υπολογιστικοί πόροι σε προγραμματιστές για ανάπτυξη εφαρμογών με χρήση εργαλείων και γλωσσών που υποστηρίζονται από τον πάροχο. Είναι δυνατόν, ο πελάτης να επιλέξει ο ίδιος της γλώσσες και τα εργαλεία στα οποία επιθυμεί να εργαστεί μέσω του **Infrastructure as a Service (IaaS)**, όπου το μόνο που παρέχεται είναι το υλικό το οποίο θα χρησιμοποιήσει ο πελάτης.

Η έκθεση τελειώνει προτείνοντας τέσσερα μοντέλα ανάπτυξης. Οι υπηρεσίες που είναι προσβάσιμες από όλους φιλοξενούνται σε **Public Cloud**. Από την άλλη, με τον όρο **Private Cloud** αναφερόμαστε σε ένα Cloud, στο οποίο έχει αποκλειστική πρόσβαση ένας μόνο οργανισμός. Αν περισσότεροι από ένας οργανισμοί χρησιμοποιούν το ίδιο Cloud χρησιμοποιούμε τον όρο **Community Cloud**. Τέλος, είναι πιθανόν να έχουμε συνδυασμό δύο από τα παραπάνω σε ένα υβριδικό μοντέλο (**Hybrid Cloud**).

Εν κατακλείδει, η άνθιση του Cloud Computing έκανε ευρέως διαθέσιμους ισχυρούς υπολογιστικούς πόρους. Οι υπηρεσίες, που παρέχονται σε αυτό, μπορούν να κλιμακωθούν ανάλογα με τη ζήτησή τους και είναι διαθέσιμες σε κάθε γωνιά του κόσμου με το διαδίκτυο. Υπάρχουν, όμως, και αρνητικά. Για παράδειγμα, ο πελάτης δεν μπορεί να εξασφαλίσει την ασφάλεια των δεδομένων του που φυλάσσονται σε αυτό και βασίζεται στην ασφάλεια του παρόχου του. Ένα άλλο μειονέκτημα είναι οι μεγάλες χρονοκαθυστερήσεις που πιθανόν να παρατηρηθούν αν οι server βρίσκονται σε μεγάλη απόσταση από τον πελάτη. Αυτός είναι και ένας από τους λόγους που τα τελευταία χρόνια έχουμε την εμφάνιση του Edge Computing, το οποίο εξετάζουμε αναλυτικά στην επόμενη παράγραφο.

2.2.2 Edge Computing

Η τεχνολογία του Edge Computing επιχειρεί να φέρει τις δυνατότητες του Cloud Computing πιο κοντά στο χρήστη [ATP20a], [Zha+16]. Η μείωση του χρόνου απόκρισης είναι καίριας σημασίας για εφαρμογές και υπηρεσίες πραγματικού χρόνου. Παρόλο που θεωρείται σαν μια εξέλιξη του Cloud Computing, εν τούτοις δεν προβλέπεται να το αντικαταστήσει, αλλά να το πλαισιώσει εκτελώντας μέρος (ή ακόμα και όλα) των αιτημάτων των συσκευών προς το cloud, προσφέροντας στους χρήστες ακόμα καλύτερο Quality of Experience (QoE) [Kha+19]. Ειδικότερα, με την επικείμενη εμφάνιση των δικτύων 5G και την εκτίμηση για αυξημένη κίνηση, το Edge Computing αναμένεται να παίξει καταλυτικό ρόλο στη διατήρηση καλής εμπειρίας σε διάφορες εφαρμογές [Yu+18].

Με τον όρο Edge Computing αναφερόμαστε σε μια κατανομημένη αρχιτεκτονική επεξεργασίας δεδομένων, όπου η επεξεργασία των δεδομένων γίνεται κοντά στην πηγή παραγωγής τους. Η επεξεργασία των δεδομένων αυτών γίνεται είτε σε edge nodes, όπως μικρά data center, είτε σε edge devices, δηλαδή καθημερινές συσκευές με υπολογιστική ισχύ και χώρο απόθηκευσης που είναι συνδεδεμένες στο IoT, όπως ο προσωπικός μας υπολογιστής [DD17]. Το πρόβλημα της ενεργοποίησης του απαραίτητου αριθμού edge nodes έχει μελετηθεί διεξοδικά στη βιβλιογραφία, υιοθετώντας είτε κατανομημένες [ATP18] είτε κεντροποιημένες λύσεις [Zha+17].

Πέραν του χαμηλού χρόνου απόκρισης, το Edge Computing διαθέτει μια σειρά από χαρακτηριστικά που το διαφοροποιούν από το Cloud Computing. Αρχικά, η εξ ορισμού ανάγκη για ύπαρξή του στην άκρη του δικτύου επιβάλλει την **Πυκνή Γεωγραφική Διασπορά** των κόμβων (Edge Server) που θα εξυπηρετούν τους χρήστες. Οι κόμβοι αυτοί διαφέρουν από υλοποίηση σε υλοποίηση και θα τους δούμε αναλυτικά αργότερα. Προβλέπεται, επίσης, η χρήση πρωτοκόλλων όπως το Locator ID Separation Protocol (LISP), ώστε να προσφέρεται

Υποστήριξη Μετακίνησης, αν ο χρήστης το επιθυμεί. Η **Επίγνωση Θέσης** κάθε χρήστη μπορεί να χρησιμοποιηθεί και για επιλογή του πλησιέστερου Edge Server κάθε φορά και είναι εύκολα υλοποιήσιμη με υφιστάμενες τεχνολογίες, όπως το GPS. Επειδή για την επιλογή του καταλληλότερου Edge Server, μερικές φορές δεν αρκεί μόνο η θέση του χρήστη, το Edge Computing διαθέτει, ακόμη, **Επίγνωση Περιβάλλοντος** με πληροφορίες όπως το φορτίο του δικτύου που μπορούν να χρησιμοποιηθούν για αυτό τον σκοπό. Ως φυσικό επακόλουθο των παραπάνω το Edge Computing διαθέτει **Εγγύτητα** (Proximity) σε σχέση με τις υπολογιστικές υπηρεσίες που προσφέρονται, Τέλος, η ύπαρξη διαφόρων αρχιτεκτονικών και συσκευών με άμεση πρόσβαση στην άκρη του δικτύου, το καθιστούν **Ετερογενές**.

Για υλοποίηση του Edge Computing έχουν προταθεί τρεις αρχιτεκτονικές, τα Cloudlets, το Fog Computing και το Mobile Edge Computing (MEC). Σε αυτή την παράγραφο εξετάζουμε τη λειτουργία κάθε μίας συνοπτικά, καθώς και μια μικρή σύγκριση μεταξύ τους.

Cloudlets

Το 2009 με κίνητρο για χαμηλούς χρόνους απόκρισης που θα ζητούσαν εφαρμογές επαυξημένης πραγματικότητας, Satyanarayanan και λοιποί πρότειναν τη λύση των cloudlets [Sat+09]. Ένα cloudlet ορίζεται σαν ένας υπολογιστής ή ένα δίκτυο υπολογιστών με υψηλή ταχύτητα μεταξύ τους επικοινωνίας, πλούσιο σε υπολογιστικούς πόρους και τοποθετημένο στην άκρη του δικτύου. Η λειτουργία του βασίζεται σε ένα πολύ σημαντικό σύγχρονο εργαλείο, την εικονικοποίηση (virtualization).

Η τοποθέτηση τους προτείνεται να γίνεται σε διάφορους καθημερινούς χώρους όπως καφέ, εστιατόρια κλπ σε αντιστοιχία με την τοποθέτηση των διάφορων Wi-fi Access Points κατά την προηγούμενη δεκαετία. Εξασφαλίζεται, έτσι, η γεωγραφική τους διασπορά και εφαρμογές που χρειάζονται πρόσβαση σε αυτά μπορούν να διακόπτουν τη λειτουργία τους όταν απομακρύνονται από κάποιο και να την συνεχίζουν όταν προσεγγίζουν κάποιο άλλο.

Οι υπολογιστικοί πόροι του cloudlet προσφέρονται μέσω της εικονικοποίησης. Αφού ο πελάτης πλησιάσει το cloudlet και ζητήσει τη χρήση του, ετοιμάζει ένα Virtual Machine (VM) Overlay το οποίο στέλνει σε αυτό. Τότε το cloudlet επιλέγει την κατάλληλη βάση εικονικού μηχανήματος και προσθέτει σε αυτήν το overlay δημιουργώντας έτσι ένα πλήρες εικονικό μηχανήμα με την εφαρμογή του πελάτη έτοιμη προς εκτέλεση. Αφού η εφαρμογή τελειώσει την εκτέλεση της, το αποτέλεσμα στέλνεται στον πελάτη. Κατά την τελική χρήση του cloudlet το αντίστοιχο εικονικό μηχανήμα καταστρέφεται και το cloudlet είναι έτοιμο για χρήση από κάποιον καινούριο πελάτη.

Τα cloudlets είναι "Data-Centers in a Box", ένα μικρό cloud κοντά στο χρήστη δηλαδή. Χαρακτηρίζονται από αυτοδιαχείριση και αποκεντροποιημένη ιδιοκτησία (αφού διάφορες μικροεπιχειρήσεις μπορούν να διαθέτουν από ένα). Ακόμη, είναι προσβάσιμα με μόνο ένα hop με μεγάλο εύρος ζώνης, πετυχαίνοντας, έτσι, χαμηλούς χρόνους απόκρισης [DD17].

Mobile Edge Computing

Το mobile edge computing (MEC), όπως λέει και το όνομά του, είναι μια αρχιτεκτονική του edge computing για κινητούς πελάτες (end users). Οι υπολογιστικές υπηρεσίες του προσφέρονται με τη χρήση MEC servers, υπολογιστών με κοινό εξοπλισμό εμπορίου όσον αφορά τη CPU, τη μνήμη και τις διεπαφές επικοινωνίας τους [Bec+14].

Οι MEC servers τοποθετούνται συνδεδεμένοι φυσικά με τους base stations των δικτύων κινητής τηλεφωνίας και είναι προσβάσιμες στον πελάτη μέσω των υφιστάμενων Radio Access Networks (RANs) (Σχήμα 2.3). Τα RANs είναι το ενδιάμεσο επίπεδο στις υπηρεσίες κινητής τηλεφωνίας από την πρώτη τους εμφάνιση. Οι κινητές τηλεφωνικές συσκευές συνδέονται πρώτα με το RAN της περιοχής που ανήκουν και μέσω αυτού συνδέονται στο βασικό δίκτυο. Με παρόμοιο τρόπο, μια κινητή συσκευή η οποία επιθυμεί να χρησιμοποιήσει κάποια υπηρεσία που προσφέρουν οι MEC servers συνδέεται στο RAN της περιοχής που ανήκει και το αίτημα της μεταβιβάζεται στον αντίστοιχο server όπου και εξυπηρετείται.

Η εξυπηρέτηση των αιτημάτων γίνεται με χρήση του Mobile Edge Orchestrator [Mit+19]. Αφού ο orchestrator λάβει το αίτημα, ελέγχει από ένα κατάλογο, που διατηρεί ο ίδιος, αν η εφαρμογή που περιλαμβάνεται στο αίτημα εκτελείται ήδη σε κάποιον από τους servers, οπότε και την προωθεί σε αυτόν. Σε διαφορετική περίπτωση, εφόσον υπάρχουν διαθέσιμοι πόροι γίνεται αρχικοποίησή της. Σε περίπτωση που δεν υπάρχουν διαθέσιμοι πόροι το αίτημα προωθείται στο cloud [MPV14].

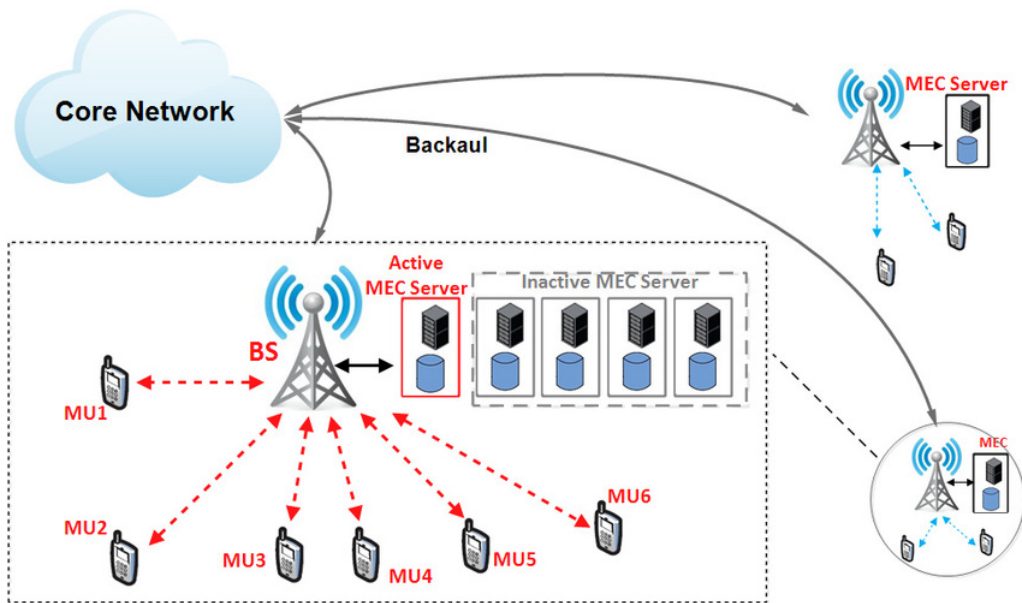
Λόγω της χρήσης των RANs το MEC προσφέρει πρόσβαση σε υπολογιστικούς πόρους με ένα hop και από σχεδόν οποιαδήποτε τοποθεσία. Ένα ακόμη σημαντικό πλεονέκτημα της αρχιτεκτονικής είναι η γνώση πληροφοριών για το περιβάλλον του δικτύου. Μέσω του RAN το οποίο είναι συνδεδεμένο η συσκευή μπορεί να αντλεί πληροφορίες, όπως τοποθεσία και χρόνος, αλλά και στοιχεία του δικτύου, όπως το φορτίο που διαθέτει εκείνη τη στιγμή. Οι πιο πάνω ιδιότητες το καθιστούν ιδανική λύση για εφαρμογές, όπως η αυτοματοποιημένη οδήγηση ή εφαρμογές επαυξημένης πραγματικότητας. Επιπρόσθετες εφαρμογές βρίσκονται στο πεδίο των εφαρμογών των έξυπνων πόλεων που υποστηρίζονται από αυτόματα ιπτάμενα οχήματα [ATT19], [Che+16], σε εφαρμογές κοινωνικής ασφάλειας [Fra+19], [Mes+19], αλλά και φυσικών καταστροφών, όπου η υπολογιστική διαθεσιμότητα και η έγκαιρη επεξεργασία είναι ιδιαίτερα κρίσιμα κριτήρια για την ποιότητα της υπηρεσίας [Mes+17], [FTP20].

Fog Computing

Ο όρος Fog Computing θεμελιώθηκε για πρώτη φορά το 2012 από τη Cisco [BM12] και με αυτόν ορίζουμε μια γεωγραφικά κατανεμημένη αρχιτεκτονική με χρήση πόρων από ένα σύνολο ετερογενών συσκευών τοποθετημένη κοντά στην άκρη του δικτύου. Όπως και οι υπόλοιπες αρχιτεκτονικές edge computing τοποθετείται στο επίπεδο μεταξύ των τοπικών συσκευών και του cloud.

Σε αντίθεση με τα cloudlets και το MEC η τοποθέτηση των edge nodes μπορεί να είναι οπουδήποτε στο επίπεδο του fog, αν και τυπικά αυτοί βρίσκονται πιο κοντά στις τοπικές συσκευές. Οι κόμβοι αυτοί μπορεί να είναι οποιαδήποτε συσκευή με υπολογιστική ισχύ και

Σχήμα 2.3: Η αρχιτεκτονική MEC



δυνατότητες αποθήκευσης και σύνδεσης σε δίκτυα, όπως μεταγωγείς και κάμερες ασφαλείας.

Η παροχή πόρων στο fog γίνεται με τη βοήθεια του Fog Orchestrator. Ο orchestrator λαμβάνει το αίτημα συνοδευόμενο από μια σειρά από απαιτούμενες προδιαγραφές (πχ QoS). Ακολουθώντας, με βάση το αίτημα και τις προδιαγραφές που έχει ζητήσει ο πελάτης, ο orchestrator ετοιμάζει μια ταξινομημένη λίστα από κατάλληλους κόμβους και ο πελάτης επιλέγει με ποιον θα συνδεθεί για να εκτελέσει το αίτημά του [DD17].

Λόγω των διάφορων συσκευών που μπορούν να λειτουργήσουν σαν edge nodes, το fog computing χαρακτηρίζεται από ετερογένεια, η οποία με τη σειρά της επιτρέπει την εξυπηρέτηση ενός μεγάλου αριθμού πρωτοκόλλων. Οι κόμβοι αυτοί επιτρέπουν, επίσης, την άντληση πληροφοριών για το περιβάλλον. Από την άλλη, η ετερογένεια αυτή, μπορεί σε ακραίες περιπτώσεις να προκαλέσει αύξηση της χρονοκαθυστερήσης, αφού είναι πιθανό κανείς από τους διαθέσιμους edge nodes να μην μπορεί να εξυπηρετήσει το αίτημα του πελάτη. Σε αυτήν την περίπτωση το αίτημα προωθείται με αποτέλεσμα την αύξηση του αριθμού των hops.

Σύγκριση των Αρχιτεκτονικών

Οι τρεις αρχιτεκτονικές που εξετάσαμε έχουν αρκετά κοινά χαρακτηριστικά, όπως η μείωση των χρονοκαθυστερήσεων, η γεωγραφική διασπορά, η επίγνωση τοποθεσίας και περιβάλλοντος κ.ά. Θα εξετάσουμε σύντομα μερικές διαφορές τους. Μια συνοπτική σύγκριση δίνεται στον πίνακα 2.1 (από το [DD17]).

Η πρώτη διαφορά μπορεί να σημειωθεί στο είδος των edge nodes. Ενώ στο MEC και στα cloudlets υπάρχουν συγκεκριμένες συσκευές που λειτουργούν ως κόμβοι, στο fog computing έχουμε πολλές διαφορετικές συσκευές που αναλαμβάνουν αυτό το ρόλο, κάτι που ευνοεί την εξάπλωσή του. Παρ' όλ' αυτά, η χρήση συγκεκριμένων συσκευών στις δύο πρώτες αρχιτεκτονικές, συνήθως, σημαίνει την ύπαρξη καλύτερων πόρων σε αυτές.

Πίνακας 2.1: Σύγκριση των αρχιτεκτονικών edge computing

	Fog Computing	MEC	Cloudlets
Τύπος Κόμβων	Συσκευές με υπολογιστικές δυνατότητες	MEC Servers	Data Center in a Box
Τοποθεσία Κόμβων	Όπουδήποτε ανάμεσα στο Cloud και τους χρήστες	Radio Network Controller	Τοπική Εγκατάσταση
Αρχιτεκτονική Λογισμικού	Fog Orchestrator	Mobile Edge Orchestrator	Cloudlet Agent
Επίγνωση Περιβάλλοντος	Μέτρια	Υψηλή	Χαμηλή
Απόσταση	1 ή περισσότερα hops	1 hop	1 hop
Πρόσβαση	Wi-Fi, Bluetooth, RANs	RANs	Wi-Fi
Διακομβική Επικοινωνία	Ναι	Μερικώς	Μερικώς

Ακόμη, λόγω των διαφορετικών τρόπων πρόσβασης και δυνατοτήτων των edge nodes είναι πιθανό στο fog computing να έχουμε απόσταση μεγαλύτερη του ενός hop σε αντίθεση με την εγγυημένη απόσταση ενός hop στις άλλες δύο αρχιτεκτονικές. Τέλος, η ετερογένεια αυτή προσδίδει στο fog computing ευελιξία για τη χρήση διαφορετικών πρωτοκόλλων για επικοινωνία, σε αντίθεση με τα, περιορισμένα σε πρωτόκολλα Wi-Fi, cloudlet και τους, περιορισμένους σε κινητά δίκτυα, MEC servers.

Εισαγωγή στο πρόβλημα μεταφοράς δεδομένων

Με την άνοδο και την εξέλιξη των τεχνολογιών που περιγράφονται στο κεφάλαιο 2, προκύπτει μια σειρά από προβλήματα τα οποία ζητούν τα κατάλληλα μοντέλα για την επίλυσή τους. Ένα από τα σημαντικότερα από αυτά είναι το πρόβλημα της μεταφοράς δεδομένων, δηλαδή η αποστολή δεδομένων στους απομακρυσμένους servers που προσφέρουν τις υπηρεσίες του edge computing. Στο κεφάλαιο αυτό επιχειρείται μια ανασκόπηση στην υπάρχουσα βιβλιογραφία, ώστε να διαπιστωθούν τα περιβάλλοντα για τα οποία το πρόβλημα έχει μελετηθεί. Ακολούθως, δίνεται το περιβάλλον με το οποίο εργαζόμαστε στα επόμενα κεφάλαια της διπλωματικής. Πριν την ανασκόπηση προηγείται μια σειρά σύντομων ορισμών για τεχνολογίες που συναντώνται στη βιβλιογραφία που μελετείται.

3.1 Όροι που Συναντώνται

Κατηγορίες Μεταφοράς Δεδομένων

Είναι δυνατόν να διακρίνουμε τρεις κατηγορίες Μεταφοράς Δεδομένων (Data Offloading) [ATP20a] βάσει του ποσοστού της εφαρμογής το οποίο οι πελάτες επιθυμούν να στείλουν στο server:

- **Always Offload:** Ο χρήστης πάντα επιλέγει να στείλει ολόκληρη την εφαρμογή.
- **All or Nothing Offloading:** Ο χρήστης επιλέγει είτε τον εαυτό του είτε τον server και ο υπολογισμός γίνεται εξ ολοκλήρου εκεί.
- **Partial Offloading:** Ο χρήστης στέλνει ένα κομμάτι της εφαρμογής για εξυπηρέτηση και εκτελεί το υπόλοιπο τοπικά.

Στη συνέχεια θα χρησιμοποιούμε τους πιο πάνω όρους για να χαρακτηρίσουμε τη βιβλιογραφία που μελετείται.

Software-Defined Networking (SDN)

Η αρχιτεκτονική SDN είναι μια αρχιτεκτονική διαχείρισης δικτύων που σκοπό έχει να προσφέρει ευελιξία στο διαχειριστή του δικτύου. Κύριο της χαρακτηριστικό είναι η υλοποίηση

των τοπολογιών στο λογικό επίπεδο (πάνω από ήδη υπάρχον κατάλληλο υλικό), σε αντίθεση με τις παλαιότερες τεχνικές όπου οι τοπολογίες υλοποιούνταν στο φυσικό επίπεδο. Επιτρέπεται, έτσι, στο διαχειριστή του δικτύου να ρυθμίζει ευκολότερα την κίνηση στο δίκτυο, καθώς και να έχει ανά πάσα στιγμή μια εικόνα για την κατάστασή του. Η κίνηση στα δίκτυα SDN ελέγχεται από τον SDN Controller [Xia+15].

Μη-Συνεργατικά Παίγνια και Ισορροπία Nash

Με το όρο «Μη-Συνεργατικά Παίγνια» αναφερόμαστε σε παίγνια όπου δύο ή περισσότεροι παίκτες ανταγωνίζονται μεταξύ τους με στόχο ο καθένας να μεγιστοποιήσει το δικό του κέρδος. Οι συνεργασίες μεταξύ παικτών απαγορεύονται. Όταν κανένας παίκτης δεν μπορεί να βελτιώσει το κέρδος του αλλάζοντας αποκλειστικά τη δική του στρατηγική λέμε πως το παίγνιο έχει καταλήξει σε μια ισορροπία Nash [TVP13; ZG11]. Είναι πιθανό ένα παίγνιο να έχει περισσότερες από μία ισορροπίες.

Παίγνια Μειονότητας

Με τον όρο «Παίγνια Μειονότητας» αναφερόμαστε σε ένα παίγνιο (συνεργατικό ή μη-συνεργατικό) στο οποίο ένας περιττός αριθμός από παίκτες καλείται να πάρει μια δυαδική απόφαση. Νικητές αναδουκνούνται οι παίκτες που πήραν την απόφαση με τους λιγότερους υποστηρικτές. Ένα τέτοιο παίγνιο διαθέτει πάντα τουλάχιστον μία ισορροπία Nash [RMH17].

Θεωρία Προοπτικής (Prospect Theory)

Η θεωρία προοπτικής [Tve00] προσπαθεί να περιγράψει την πραγματική συμπεριφορά των ανθρώπων σε ένα επενδυτικό χώρο όπου κέρδη και ζημιές είναι πιθανά. Με βάση την θεωρία εκτιμώμενης ωφελείας (expected utility theory) οι αποφάσεις σε ένα τέτοιο χώρο θεωρείται ότι λαμβάνονται με λογική. Η θεωρία προοπτικής υποδεικνύει πως διαφορετικοί άνθρωποι έχουν διαφορετική οπτική για το κέρδος ή τις ζημιές τους με αποτέλεσμα οι αποφάσεις από άνθρωπο σε άνθρωπο να διαφέρουν. Η θεωρία προοπτικής έχει βρει πολλές εφαρμογές στο πεδίο των ασύρματων επικοινωνιών με στόχο τη συντηρητική χρήση των πόρων του δικτύου [VTP19b; VTP19a; VTP19c], στο πεδίο των κοινωνικών κυβερνοχώρων [TTP19; TTP20], και των μηχανισμών προστασίας των ασύρματων δικτύων [Xia+14]. Με βάση τις προηγούμενες μελέτες αποδεικνύεται ότι η θεωρία προοπτικής καταλήγει σε πιο ρεαλιστικά αποτελέσματα σε σχέση με τη θεωρία μεγιστοποίησης της ευχαρίστησης των χρηστών [Lew17].

3.2 Ανασκόπηση της Υπάρχουσας Βιβλιογραφίας

3.2.1 Μεταφορά δεδομένων σε συστήματα κινητού υπολογισμού με βάση την τεχνολογία SDN

Στο [Mit+19] έχουμε τη χρήση της αρχιτεκτονικής SDN ώστε ο κάθε χρήστης να επιλέγει τον κατάλληλο MEC server για να εξυπηρετηθεί. Ακόμη, θεωρείται πως οι χρήστες

ακολουθούν τακτική partial offloading. Στο μοντέλο που προτείνεται έχουμε τη σύνδεση των MEC servers στον SDN Controller, ενώ και οι πελάτες επικοινωνούν με αυτόν για την εξυπηρέτησή τους. Είναι ευθύνη του SDN Controller να καθορίσει τόσο την κατανομή (και το ποσοστό υπολογισμού) των χρηστών στους servers, όσο και την τιμή που καλούνται να πληρώσουν για τις υπηρεσίες που λαμβάνουν, μέσω του αλγορίθμου Data Offloading and MEC Server Selection (DO-MECS) που παρουσιάζεται στη δημοσίευση με τις λειτουργίες του οποίου είναι εφοδιασμένος. Η προτεινόμενη αρχιτεκτονική φαίνεται στο Σχήμα 3.1 από την ίδια δημοσίευση).

Το περιβάλλον παρουσιάζεται ως εξής. Το σύνολο S αποτελείται από όλους τους servers, με τον καθένα να χαρακτηρίζεται από τα μεγέθη $c_s^{(t)}$ και $f_s^{(t)}$, το κόστος του server ανά bit επεξεργασίας και την έκπτωση προς τους πελάτες, αντίστοιχα. Το σύνολο U αποτελείται από τους χρήστες οι οποίοι χαρακτηρίζονται από τους όρους $d_u^{(t)}$, $\alpha_u^{(t)}$ και $\beta_u^{(t)}$, με τον πρώτο να συμβολίζει την οικονομική δυνατότητα του χρήστη και τους άλλους δύο την ευκολία ικανοποίησής του. Το πρόβλημα παρουσιάζεται σε δύο στάδια. Στην αρχή οι χρήστες καλούνται να αποφασίσουν τον server τον οποίο θα χρησιμοποιήσουν και τον αριθμό των bits που θα στείλουν σε αυτόν. Ακολούθως, οι servers αποφασίζουν την τιμή που θα καθορίσουν για τις υπηρεσίες τους.

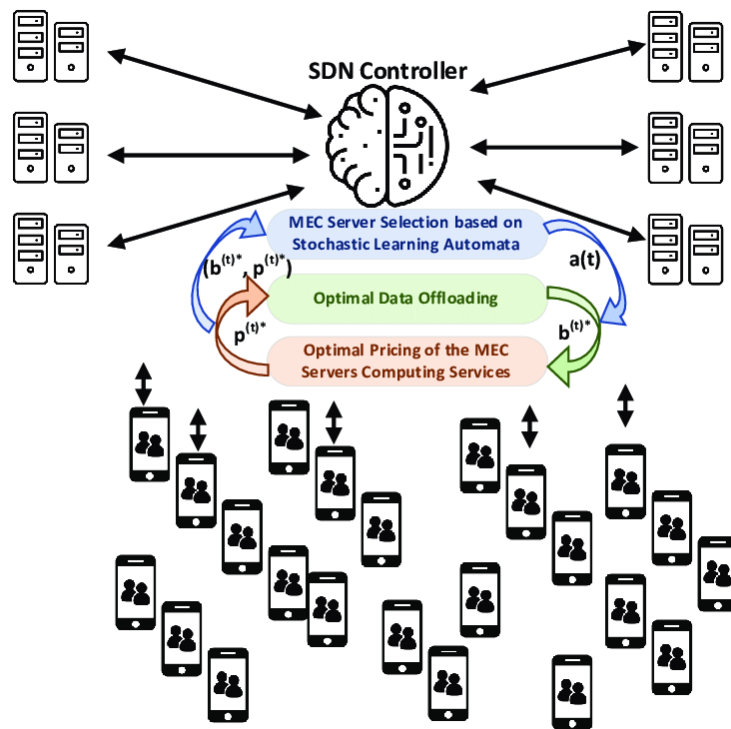
Στην αρχή οι παίχτες-χρήστες προσομοιώνονται σαν Stochastic Learning Automata (SLA) (παρουσιάζονται αναλυτικά στο κεφάλαιο 4) από τον SDN Controller και αποφασίζεται έτσι με ποιο server θα συσχετιστούν. Ακολούθως, το ποσό σε bits που θα στείλει ο κάθε παίκτης για επεξεργασία δεν είναι παρά ένα μη-συνεργατικό παίγνιο, όπου ο κάθε παίκτης προσπαθεί να μεγιστοποιήσει την (κατάλληλα διατυπωμένη) utility function του, καταλήγοντας σε ισορροπία Nash. Το παιχνίδι μελετάται και η λύση του δίνεται σε κλειστή μορφή στη δημοσίευση, οπότε ο SDN Controller ενημερώνει κατάλληλα τους παίχτες-χρήστες. Τέλος, αυτό που απομένει είναι η λύση ενός προβλήματος μεγιστοποίησης για την τιμή που θα θέσει ο κάθε server ξεχωριστά. Με χρήση και πάλι του κατάλληλου κλειστού τύπου ο SDN Controller ενημερώνει τον κάθε server για την τιμή που θα μεγιστοποιήσει το κέρδος του.

Η απεικόνιση του προβλήματος προσφέρει μια αποδοτική λύση σε ένα σύγχρονο περιβάλλον συνδυάζοντας πρόσφατες και ήδη υπάρχουσες τεχνολογίες όπως το SDN και το MEC. Είναι σημαντικό ότι το μοντέλο περιλαμβάνει τον υπολογισμό τιμής, κάτι που το κάνει σημαντικό για χρήση στην καθημερινότητα σε ένα ρεαλιστικό πλαίσιο. Η συμπερίληψη της partial offloading συμπεριφοράς το καθιστά επίσης ευέλικτο ως προς τις πιθανές αποφάσεις των χρηστών.

3.2.2 Αυτόνομο σύστημα κινητού υπολογισμού με στόχο την ικανοποίηση της Ποιότητας της Υπηρεσίας

Στο [ATP18] μελετάται η ενεργοποίηση και αξιοποίηση των MEC servers, και πάλι, σε δύο επίπεδα. Σε αντίθεση με το [Mit+19] οι πελάτες ακολουθούν τακτική Always Offload. Ως εκ τούτου για το επίπεδο των πελατών μελετάται η ανάγκη τους σε σπατάλη ενέργειας για την αποστολή των δεδομένων τους στο server. Το σύνολο S αποτελείται από όλους τους servers, ενώ το σύνολο N αποτελεί το αντίστοιχο σύνολο για τους χρήστες. Ο κάθε χρήστης

Σχήμα 3.1: Η αρχιτεκτονική MEC σε περιβάλλον SDN



συνοδεύεται από μια τιμή QoS_n η οποία χαρακτηρίζει τις προδιαγραφές που ο ίδιος θέτει για ικανοποίησή του, Η δημοσίευση καταλήγει παρουσιάζοντας τον αλγόριθμο Autonomous operation and QoS satisfaction (AOQOSS) τον οποίο βλέπουμε συνοπτικά.

Το πρώτο επίπεδο αφορά την απόφαση για ενεργοποίηση (ή όχι) του κάθε server. Πρώτα υπολογίζεται ένα κατώφλι $|S|_{th}$, το οποίο αντιπροσωπεύει τη μέγιστο αριθμό servers που η ενεργοποίησή τους παραμένει κερδοφόρα, βάση των κόστων για ενεργοποίησή τους και της αμοιβής που λαμβάνουν από κάθε χρήστη που εξυπηρετείται. Ως νικήτρια στρατηγική θεωρείται η ενεργοποίηση, εφόσον ο αριθμός των server που την επέλεξε ήταν κάτω από το κατώφλι, διαφορετικά νικήτρια στρατηγική θεωρείται η μη-ενεργοποίηση. Με χρήση των πιο πάνω διατυπώνεται ένας καταναμημένος αλγόριθμος, ο οποίος τρέχει στους servers. Σε κάθε επανάληψη ο κάθε server επιλέγει μια από τις δύο στρατηγικές βάσει μιας πιθανότητας. Εφόσον, η στρατηγική που επέλεξε ήταν η νικήτρια αυξάνει την πιθανότητα να κάνει την ίδια επιλογή στο μέλλον, διαφορετικά τη μειώνει. Ο αλγόριθμος συγκλίνει και τερματίζει όταν όλοι οι servers καταλήξουν με μεγάλη πιθανότητα σε μια στρατηγική, οπότε και ανακοινώνονται το σύνολο των ενεργών server.

Στο δεύτερο επίπεδο, αρχικά, οι μοντελοποιημένοι σαν SLA χρήστες επιλέγουν τον server με τον οποίο συσχετίζονται. Ακολούθως, μελετάται η (ελάχιστη) ισχύς μετάδοσης που χρειάζεται να ασκήσει ο κάθε χρήστης ώστε να ικανοποιήσει το QoS_n του. Για τον σκοπό αυτό διατυπώνεται ένα μη-συνεργατικό παίγνιο με παίχτες τους χρήστες που θέλουν να ελαχιστοποιήσουν το κόστος τους για εκπομπή των δεδομένων τους στον server της επιλογής τους.

Εισάγεται, επίσης, η έννοια του Efficient Satisfaction Equilibrium (ESE), της ισορροπίας δηλαδή που απαιτεί την ελάχιστη ισχύ από τους χρήστες και δίνεται ένας κατανομημένος αλγόριθμος που οδηγεί σε αυτήν, ο οποίος τρέχει σε κάθε χρήστη [Pro+19; FTP19; Goo+16; Mér+12].

Το πιο πάνω framework είναι πλήρως κατανομημένο τόσο στους servers όσο και στους χρήστες. Δεν υπάρχει, λοιπόν, η ανάγκη για κάποια κεντρική αρχή για συντονισμό των αποφάσεων των εμπλεκόμενων κάτι που το κάνει αποτελεσματικό για εφαρμογή του στα ευμετάβλητα περιβάλλοντα της καθημερινής ζωής. Ακόμη, σημαντική είναι η επίτευξη της ελαχιστοποίησης του κόστους των χρηστών και η μέγιστη αξιοποίηση των server χωρίς περιττές ενεργοποιήσεις.

3.2.3 MEC servers σε drone με χρήση τεχνητής νοημοσύνης

Μια προσέγγιση που έχει προσελκύσει αρκετό ερευνητικό ενδιαφέρον είναι η προσάρτιση των MEC servers σε Unmanned Aerial Vehicles (UAV), ή όπως έχουμε συνηθίσει να τα αποκαλούμε πλέον, drones. Ο προσαρτημένος στο drone server κινείται πάνω από το κέντρο μιας περιοχής και λαμβάνει τα αιτήματα των χρηστών κανονικά. Η προσέγγιση αυτή προσφέρει αρκετά πλεονεκτήματα όπως η εύκολη και φθηνή εγκατάστασή τους και η δυνατότητα επικοινωνίας με τις συσκευές χωρίς παρεμβολές [MTP20]. Από την ίδια δημοσίευση μπορούμε να δούμε μια απεικόνιση του μοντέλου στο σχήμα 3.2.

Μια τέτοια προσέγγιση μελετάται στο [FTP20]. Το μοντέλο θεωρεί ένα σύνολο D από χρήστες, οι οποίες βρίσκονται διασκορπισμένες σε ένα ορθογώνιο χώρο και ένα MEC server ο οποίος πετάει πάνω από την περιοχή. Οι χρήστες ακολουθούν πολιτική partial offloading και το ποσοστό των δεδομένων που θα στείλουν για επεξεργασία στο server είναι το αντικείμενο της έρευνας.

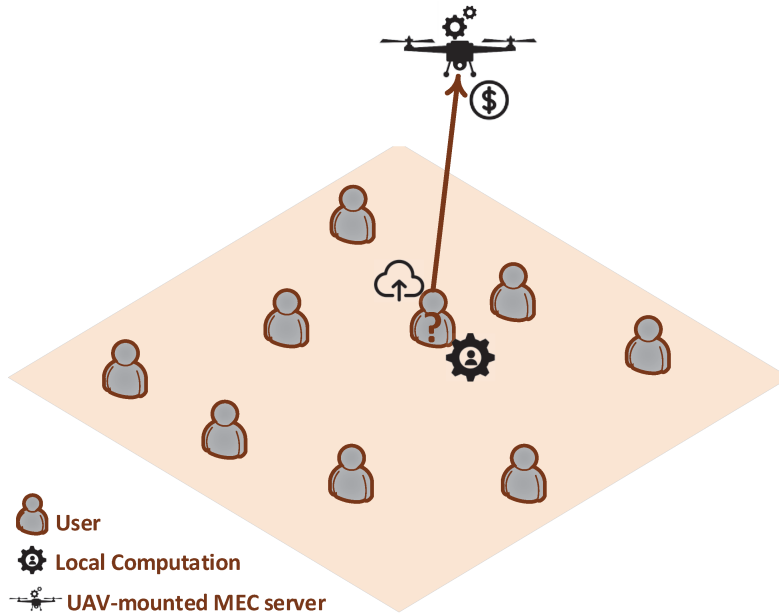
Έχουμε, ακολούθως την ανάλυση του μοντέλου, όπου εκφράζονται οι εξισώσεις που περιγράφουν τα overheads, καθώς και τις συναρτήσεις ωφελείας των χρηστών από την εξυπηρέτησή τους. Η πιο πάνω διατύπωση οδηγεί φυσιολογικά στη διατύπωση ενός μη-συνεταιρικού παιχνιδιού, για το οποίο αποδεικνύεται η ύπαρξη ισορροπίας Nash και μάλιστα δίνονται οι συναρτήσεις δυναμικής βέλτιστης απόκρισης (best response dynamics) για τους παίκτες οι οποίες όμως είναι αλληλοεξαρτώμενες.

Επιπρόσθετα της θεωρητικής ανάλυσης δίνεται μια σύγχρονη προσέγγιση με χρήση αλγορίθμων μάθησης. Όπως σημειώνεται, αυτό γίνεται για δύο λόγους. Αφενός, είναι πιθανόν οι χρήστες να μην γνωρίζουν τις συναρτήσεις βέλτιστης απόκρισης. Αφετέρου, τα δεδομένα που έχουν για αποστολή στους servers οι χρήστες, σχεδόν βέβαια δεν μπορούν να χωριστούν αυθαίρετα και γι'αυτό το λόγο πρέπει να ληφθεί υπόψη ένα διακριτό σύνολο στρατηγικών, σε αντίθεση με το παιχνιδιοθεωρητικό μοντέλο που θεωρεί οποιαδήποτε διαμέριση πιθανή. Συγκεκριμένα, εξετάζονται οι μεθόδοι Gradient Ascent Learning μέσω του κανόνα Linear Reward Inaction (LRI), Log-Linear Learning μέσω του κανόνα Binary Log-Linear Learning και Q-Learning για διάφορες τιμές της παραμέτρου μάθησης.

Η ποιότητα των λύσεων που προσφέρουν οι προαναφερθείσες μέθοδοι εξετάζεται πειραμα-

τικά και τα αποτελέσματα όσο αφορά το Social Welfare (το άθροισμα δηλαδή των ωφελειών όλων των παικτών) είναι αρκετά κοντά στη λύση που δίνει η θεωρητική προσέγγιση, τουλάχιστον στα περισσότερα από αυτά. Στα αρνητικά είναι ότι οι καλύτερες ποιοτικά λύσεις αποδεικνύονται και πιο χρονοβόρες.

Σχήμα 3.2: Η αρχιτεκτονική MEC με χρήση UAV (drone)



3.2.4 Μεταφορά δεδομένων με επίγνωση ρίσκου

Μέχρι στιγμής έχουμε δει χρήστες οι οποίοι συμπεριφέρονται λογικά. Κάτι τέτοιο (ειδικότερα αν οι χρήστες είναι άνθρωποι) πολλές φορές δεν ισχύει και στο [ATP20b] ακολουθείται αυτή η προσέγγιση με χρήση της θεωρίας προοπτικής. Οι χρήστες του μοντέλου ακολουθούν τακτική partial offloading, όμως δεν χρειάζεται να επιλέξουν αποκλειστικά ένα server για αυτό. Μπορούν να στείλουν δεδομένα για εκτέλεση σε όσους servers το επιθυμούν.

Όσο αφορά τις οντότητες του προβλήματος, έχουμε και πάλι τα σύνολα U και S για τους χρήστες και τους servers αντίστοιχα. Οι χρήστες συνοδεύονται από μια σειρά από χαρακτηριστικά τους: το μέγεθος της εφαρμογής, οι κύκλοι CPU που χρειάζονται για εκτέλεσή της, η μέγιστη επιτρεπτή χρονοκαθυστέρηση και η διαθέσιμη μπαταρία τους. Σε αντίθεση με προηγούμενες προσεγγίσεις, οι servers συνοδεύονται από ένα κατώφλι μέγιστου αριθμού δεδομένων που μπορούν να λάβουν, καθώς και από μια πιθανότητα αποτυχίας ο καθένας.

Η ανάλυση που ακολουθεί στη δημοσίευση αρχικά δεν διαφέρει από μια ανάλυση με λογικές οντότητες. Γίνεται διατύπωση και υπολογισμός των μεγεθών που χρειάζονται και ακολούθως λαμβάνοντας υπόψη την πιθανότητα αποτυχίας του κάθε server δίνεται η μέση τιμή των overheads, η οποία και χρησιμοποιείται στο υπόλοιπο της ανάλυσης. Βάσει αυτών και με τη θεωρία προοπτικής (με χρήση σημείων αναφοράς για το ρίσκο που είναι πρόθυμος να λάβει ο κάθε χρήστης) διατυπώνεται η ευχαρίστησή του κάθε χρήστη σαν συνάρτηση ωφελείας

καταλήγοντας σε ένα πρόβλημα μεγιστοποίησης το οποίο ισοδύναμα μπορεί να γραφτεί σαν ένα μη-συνεργατικό παίγνιο. Αφού αποδεικνύεται η ύπαρξη της ισορροπίας Nash δίνεται ο κατανεμημένος αλγόριθμος Distributed Algorithm for Convergence to NPE (DACP) που καταλήγει σε αυτήν.

Το γεγονός ότι ο αλγόριθμος είναι κατανεμημένος απαλλάσει το σύστημα από την ανάγκη για ύπαρξη κάποιας έμπιστης κεντρικής αρχής. Ακόμη, η χρήση της θεωρίας προοπτικής κάνει το μοντέλο πιο κοντά στην πραγματικότητα, όπου οι έξυπνες συσκευές επιδιώκουν να λάβουν αποφάσεις όπως θα τις έπαιρνε ο ιδιοκτήτης τους, ο οποίος πιθανόν να μην συμπεριφέρεται λογικά.

3.3 Μεταφορά δεδομένων με μεταβαλλόμενη φήμη

Στην ανασκόπηση που προηγήθηκε είδαμε διαφορετικές προσεγγίσεις για τη μεταφορά δεδομένων. Οι περισσότερες προσεγγίσεις αφορούν Partial Offloading και επικεντρώνονται στο ποσοστό της εφαρμογής το οποίο οι χρήστες θα στείλουν για επεξεργασία. Γενικότερα, επιδιώκεται, κυρίως μέσα από χρήση της θεωρίας παιγνίων, η βελτιστοποίηση της εμπειρίας των χρηστών από τις εφαρμογές τους ανεξάρτητα από που αυτές εκτελούνται.

Η δικιά μας διατύπωση του προβλήματος δεν παρεκκλίνει από την πιο πάνω πεπατημένη. Στόχος μας είναι η βελτίωση της εμπειρίας των χρηστών και η μείωση των χρονοκαθυστερήσεων (και των overheads) που επιβαρύνονται μέσω της κατανομής τους στους servers. Θεωρούμε ότι το edge computing υλοποιείται με την αρχιτεκτονική του fog computing που αναπτύχθηκε στο κεφάλαιο 2, ενώ οι χρήστες λειτουργούν με πολιτική always offload.

Το μοντέλο μας αποτελείται από τρία επίπεδα. Αφού γίνει η επιλογή server στο πρώτο επίπεδο, στο δεύτερο οι χρήστες ερωτώνται για την ικανοποίησή τους. Βάση των δημοσκοπήσεων (μία δημοσκόπηση για κάθε ένα server) λαμβάνεται μια τελική απόφαση και η φήμη του κάθε ενός ενημερώνεται ανάλογα. Μέσω αυτού στοχεύουμε στην κατανομή στους servers ανάλογα με τα χαρακτηριστικά τους, καθώς καλύτερα χαρακτηριστικά και πιο αποδοτικός υπολογισμός σημαίνει πιο ευχαριστημένοι χρήστες και κατ' επέκταση καλύτερη φήμη. Αυτό θα επιφέρει αύξηση των χρηστών στον «καλύτερο server» σε μελλοντικές αναθέσεις, το οποίο με τη σειρά του ίσως φέρει χαμηλότερα επίπεδα απόδοσης και μείωση της φήμης, μέχρι το σύστημα να καταλήξει σε μιας μορφής ισορροπία, όπου οι server θα αποκτήσουν επίπεδα φήμης ανάλογα με τα χαρακτηριστικά τους.

Το μοντέλο και οι επί μέρους μηχανισμοί που το απαρτίζουν παρουσιάζεται αναλυτικά στο κεφάλαιο 4.

Αναλυτική Παρουσίαση Μοντέλου και Επί Μέρους Συστατικών

Στο κεφάλαιο 3 είδαμε, με μια σύντομη ανασκοπήση, διάφορες μορφές του προβλήματος μεταφοράς δεδομένων που έχουν μελετηθεί. Αν και το πρόβλημα εμφανίστηκε μόλις την προηγούμενη δεκαετία, εν τούτοις έχει ήδη τοποθετηθεί σε αρκετά διαφορετικά πλαίσια και έχει αξιολογηθεί τόσο πειραματικά όσο και θεωρητικά. Σε αυτό το κεφάλαιο αναλύουμε το δικό μας πλαίσιο το οποίο παρουσιάσαμε συνοπτικά στο τέλος του κεφαλαίου 3. Πριν από αυτό θα δούμε μια σειρά από εργαλεία και μηχανισμούς τα οποία χρησιμοποιούνται στο μοντέλο.

4.1 Εργαλεία και Μηχανισμοί που χρησιμοποιούνται

4.1.1 Κατανεμημένη Μάθηση σε Δίκτυα Επικοινωνίας

Όπως είδαμε και στο κεφάλαιο 3, η θεωρία παιγνίων είναι ένα από τα σημαντικότερα εργαλεία στην ανάλυση ανταγωνιστικών καταστάσεων που συναντώνται στα δίκτυα επικοινωνίας. Η διατύπωση του παιγνίου και η εύρεση της ισορροπίας του (αν είναι δυνατό όσο πιο κοντά στη βέλτιστη λύση γίνεται) είναι η συνηθισμένη τακτική στα περισσότερα προβλήματα, όπου οι παίκτες-χρήστες καλούνται να λάβουν, αυτόνομα, αποφάσεις. Ωστόσο, οι εγγενείς περιορισμοί που υπάρχουν όσον αφορά την ποιότητα της πληροφορίας την οποία δέχονται οι συμμετέχοντες στο δίκτυο [XWW15] δυσχεραίνουν το εγχείρημα για θεωρητική ανάλυση του. Συγκεκριμένα, η πληροφορία στα δίκτυα επικοινωνιών μπορεί να είναι:

- **Ελλιπής:** Ο συμμετέχων στο δίκτυο γνωρίζει μόνο μερικές πληροφορίες για το περιβάλλον. Επίσης, γνωρίζει πληροφορίες μόνο για τον ίδιο και τίποτα για τους υπόλοιπους χρήστες.
- **Δυναμική:** Οι μεταβλητές και καταστάσεις του συστήματος αλλάζουν στο χρόνο.
- **Αβέβαιη:** Οι παρατηρήσεις που γίνονται από το χρήστη δεν ανταποκρίνονται (απαραιτήτως) στις πραγματικές τιμές.

Οι μηχανισμοί μάθησης χαρακτηρίζονται από την ικανότητα που δίνουν στους χρήστες να λαμβάνουν αποφάσεις βάσει του ιστορικού του συστήματος. Λόγω και της πιθανότητας

ελλιπούς πληροφορίας, οι κατανεμημένοι αλγόριθμοι μάθησης (όπου κάθε χρήστης τρέχει τον αλγόριθμο τοπικά) καθίστανται ιδανικοί για χρήση σε δυναμικά περιβάλλοντα, όπως τα δίκτυα επικοινωνιών. Ουσιαστικά, η διαφορά μεταξύ των ποικίλων μηχανισμών και αλγορίθμων που υπάρχουν έγκειται στον τρόπο με τον οποίο ο μηχανισμός αξιοποιεί τις πληροφορίες που παίρνει για να αποφασίσει την στρατηγική του.

Μαθηματικά έχουμε το εξής. Για τη μοντελοποίηση θεωρούμε ότι οι αποφάσεις λαμβάνονται σε διακριτούς γύρους. Έστω N χρήστες, ο καθένας από τους οποίους για το γύρο k έχει επιλέξει μια στρατηγική $a_i(k)$ λαμβάνοντας ανταμοιβή $r_i(k)$. Θεωρούμε τα διανύσματα $a(k) = [a_1(k), \dots, a_N(k)]$ και $r(k) = [r_1(k), \dots, r_N(k)]$ στα οποία έχουμε πληροφορία για όλο το σύστημα. Έστω, ακόμη, ότι ο χρήστης n καλείται να πάρει μια απόφαση στον γύρο $k + 1$. Αφού ο χρήστης n λάβει την ανταμοιβή του, εφαρμόζει τον κανόνα μάθησης F και αποφασίζει την στρατηγική του για τον επόμενο γύρο. Έχουμε, λοιπόν, το γενικό κανόνα:

$$a_n(k + 1) = F([a(k); r(k)])$$

Και αφού ο κανόνας εφαρμοστεί από όλους, μπορούμε να γράψουμε (όπου $a(k+1)$ το αντίστοιχο διάνυσμα για τον επόμενο γύρο):

$$a(k + 1) = F([a(k); r(k)])$$

4.1.2 Τα αυτόματα SLA

Τα αυτόματα Stochastic Learning Automata (SLA) είναι ένας πιθανοτικός μηχανισμός μάθησης [XWW15]. Κάθε χρήστης που συμμετέχει στο δίκτυο μοντελοποιείται σαν ένα SLA και, στο τέλος κάθε γύρου, εκτελείται ο κανόνας μάθησης. Το αποτέλεσμα του κανόνα, ωστόσο, δεν είναι ντετερμινιστικό, αλλά, αντίθετα, έχουμε την αλλαγή στην κατανομή πιθανότητας σε ένα διάνυσμα από πιθανές στρατηγικές. Βάσει της νέας κατανομής πιθανότητας, ο χρήστης n παίρνει την απόφασή του για τον επόμενο γύρο.

Ένας από τους πιο δημοφιλείς κανονες μάθησης για SLA (και αυτός που χρησιμοποιούμε και εμείς) είναι ο κανόνας Linear Reward Inactive (LRI). Ποιοτικά, αφού ο χρήστης λάβει την ανταμοιβή του για το γύρο που προηγήθηκε ενημερώνει το διάνυσμα των πιθανοτήτων του ως εξής. Αν λάβει θετική ανταμοιβή αυξάνει την πιθανότητα να επιλέξει ξανά την ίδια στρατηγική και μειώνει την πιθανότητα να επιλέξει κάποια άλλη. Σε περίπτωση αρνητικής ανταμοιβής έχουμε το αντίθετο.

Ας θεωρήσουμε και πάλι την περίπτωση ενός συνόλου \mathcal{U} με N χρήστες και $n \in \mathcal{U}$. Έστω $p_n(k) = \{p_{n1}(k), \dots, p_{ni}(k), \dots, p_{nm}(k)\}$ το (πεπερασμένο μήκους m) διάνυσμα πιθανοτήτων για τη στρατηγική του χρήστη n κατά τη διάρκεια του k -οστού γύρου, $a_n(k)$ η στρατηγική που εν τέλει επιλέχθηκε και $\bar{r}_n(k)$ η κανονικοποιημένη ανταμοιβή του. Τότε οι πιθανότητες επιλογής ενημερώνονται ως εξής:

$$\begin{aligned} p_{ni}(k + 1) &= p_{ni}(k) + b\bar{r}_n(k)(1 - p_{ni}(k)) & i = a_n(k) \\ p_{ni}(k + 1) &= p_{ni}(k) - b\bar{r}_n(k)p_{ni}(k) & i \neq a_n(k) \end{aligned}$$

Όπου b η παράμετρος μάθησης. Όπως μπορούμε να δούμε ο LRI είναι πλήρως κατανεμημένος, καθώς η ενημέρωση γίνεται μόνο με χρήση της ανταμοιβής του κάθε χρήστη. Επίσης,

όπως αποδείχθηκε στο [SPT94], για πολύ μικρές τιμές του b , $b \rightarrow 0$, συγκλίνει ασυμπτωτικά σε ισορροπία Nash για παιχνίδια δυναμικού.

4.1.3 Μηχανισμοί Φιλαλήθων Δημοσκοπήσεων

Η λήψη αποφάσεων πολλές φορές, δεν εξαρτάται αποκλειστικά από τις εμπειρίες ενός συγκεκριμένου ατόμου. Για παράδειγμα, αν κάποιος έχει μια κακή εμπειρία με ένα συγκεκριμένο προϊόν ή υπηρεσία δεν σημαίνει απαραίτητα πως δεν θα το ξαναδοκίμασει ποτέ, αν οι υπόλοιποι που θα το αξιολογήσουν έμειναν ικανοποιημένοι. Αυτό μπορεί να επεκταθεί και σε άλλες περιπτώσεις στις οποίες θέλουμε να μάθουμε μια τιμή και δεν έχουμε κάποιο τρόπο να μάθουμε την πραγματική τιμή της (στο προηγούμενο παράδειγμα, η πραγματική ποιότητα ενός προϊόντος).

Οι δημοσκοπήσεις αυτού του είδους αντιμετωπίζουν δύο εγγενείς προβλήματα. Αφενός, η σύνταξη μιας ποιοτικής αξιολόγησης είναι χρονοβόρα και είναι ωφέλιμη μόνο για τους άλλους, οπότε έχουμε μιας μορφής προχειρότητα στις αξιολογήσεις που συναντώνται. Αφετέρου, οι αξιολογητές δεν είναι απαραίτητα ειλικρινείς. Είναι πιθανόν, αξιολογητές να καταθέτουν ψευδείς αξιολογήσεις από φόβο, ευγένεια ή ανάγκη τους για συμφωνία με την πλειοψηφία. Συσσωρεύονται έτσι ψευδείς αξιολογήσεις που επηρεάζουν την κοινή γνώμη.

Μια λύση που θα μπορούσε να σκεφτεί κανείς είναι η ανταμοιβή των αξιολογητών για τις κριτικές τους. Με αυτόν τον τρόπο, ωστόσο, δεν καταπολεμάται απαραίτητα το ψεύδος των αξιολογήσεων. Αν, για παράδειγμα, η ανταμοιβή δίνεται σε αυτούς που εξέφρασαν τη γνώμη της πλειοψηφίας, τότε πολλοί από την αντίθετη άποψη θα δώσουν ψευδή αξιολόγηση, ώστε να λάβουν την ανταμοιβή. Είναι καίριας σημασίας, λοιπόν, η δημιουργία μηχανισμών, τέτοιων ώστε ο κάθε κριτής να καταθέτει την πραγματική του άποψη για το αντικείμενο που αξιολογείται. Οι μηχανισμοί αυτοί χαρακτηρίζονται ως **φιλαλήθεις** (truthful).

Η πιο διαδεδομένη οικογένεια φιλαλήθων μηχανισμών είναι οι μηχανισμοί VCG (Vickrey-Clarke-Groves, από τα ονόματα των δημιουργών του), με ευρεία εφαρμογή σε δημοπρασίες [KP17]. Ένας άλλος (ο πρώτος ο οποίος λαμβάνει υπόψη και τις απάντησεις άλλων για καθορισμό της ανταμοιβής του κάθε αξιολογητή) είναι ο Peer-Prediction που παρουσιάστηκε το 2005 στο [MRZ05]. Στις επόμενες παραγράφους θα μελετήσουμε άλλους δύο μηχανισμούς αυτού του είδους, συγγενείς μάλιστα μεταξύ τους.

4.1.4 Ο μηχανισμός Bayesian Truth Serum

Ο μηχανισμός Bayesian Truth Serum (BTS) προτάθηκε το 2004 από τον Drazen Prelec στο [Pre04]. Είναι ένας φιλαλήθης μηχανισμός για διάφορους τύπους δημοσκοπήσεων, όπως αυτούς που περιεγράψαμε παραπάνω, και στοχεύει στην αξιοποίηση της ιδιότητας «παραδόξως κοινός» που παρουσιάζουν οι προσωπικές απόψεις, ώστε να πάρει την αλήθεια από τους ερωτηθέντες. Συγκεκριμένα, ο μηχανισμός ζητά δύο απαντήσεις από κάθε συμμετέχοντα. Πρώτα, ζητά την προσωπική του άποψη σε μια ερώτηση πολλαπλής επιλογής (με m το πλήθος επιλογές) και ακολούθως, ζητά την εκτίμησή του για το ποσοστό των ερωτηθέντων που έδωσε την κάθε απάντηση. Με βάσει τα πιο πάνω ο μηχανισμός αποφασίζει μια ανταμοιβή για τον κάθε ερωτηθέντα.

Σαν παραδόξως κοινή ορίζεται μια απάντηση η οποία συγκεντρώνει πραγματικό ποσοστό μεγαλύτερο από το μέσο εκτιμώμενο των συμμετοχόντων από την πιο πάνω διαδικασία. Με τον ίδιο τρόπο ορίζεται και η παραδόξως σπάνια απάντηση, όπου το πραγματικό ποσοστό είναι μικρότερο από το μέσο εκτιμώμενο. Σύμφωνα με το «δείγμα του ενός» [Daw89], ο κάθε ερωτηθέντας τείνει να πιστέψει πως η άποψή του συναντάται σε μικρότερο μέρος του πληθυσμού από το πραγματικό. Αυτό συμβαίνει, διότι, ενώ αρχικά θεωρεί πως ένα ποσοστό X του πληθυσμού μοιράζεται την ίδια άποψη με αυτόν, εν τούτοις, καταλήγει στο συμπέρασμα πως τα δικά του θετικά συναισθήματα για την άποψη που έδωσε τον έχουν οδηγήσει σε μια υπερεκτίμηση του πραγματικού ποσοστού και ως αποτέλεσμα δίνει ένα ποσοστό $\bar{X} < X$ ως τελική εκτίμηση. Ως εκ τούτου, οι πραγματικές προσωπικές απόψεις τείνουν να είναι παραδόξως κοινές και ο μηχανισμός τις ανταμοίβει περισσότερο. Είναι επομένως, στο συμφέρον του κάθε συμμετέχοντα στη δημοσκόπηση να εκφράσει την πραγματική προσωπική του άποψη.

Η σύγκλιση του μηχανισμού σε ισορροπία βασίζεται σε δύο υποθέσεις για το περιβάλλον. Αφενός, το πλήθος n των ερωτηθέντων θα πρέπει να είναι αρκετά μεγάλο, $n \rightarrow \infty$, ώστε οι μεμονωμένες απαντήσεις να μην επηρεάζουν σημαντικά την κατανομή των απαντήσεων. Σημειώνεται, πως ο μηχανισμός λειτουργεί όχι μόνο για πεπερασμένο πλήθος, αλλά και για απείρως μετρήσιμο. Αφετέρου, οι συμμετέχοντες, θα πρέπει να θεωρούν τις προσωπικές τους απόψεις σαν απρόσωπη πληροφορία για την κατανομή των αξιολογήσεων στον πληθυσμό.

Εφόσον, τα παραπάνω ισχύουν ο μηχανισμός τίθεται σε λειτουργία για ένα άπειρο μετρήσιμο σύνολο ερωτηθέντων. Θεωρούμε τον πληθυσμό των ερωτηθέντων συμβολίζοντας με $r \in \{1, 2, \dots\}$ κάθε έναν από αυτούς. Αν η πολλαπλής επιλογής ερώτηση έχει m πιθανές απαντήσεις, ορίζουμε για κάθε ερωτηθέντα την τούπλα $x^r = (x_1^r, x_2^r, \dots, x_m^r)$ με $x_k^r \in \{0, 1\}$ για κάθε $k \in \{1, 2, \dots, m\}$ και $\sum_{i=1}^m x_i^r = 1$, η οποία περιέχει όλη την πληροφορία για την απάντηση που έδωσε ο ερωτηθέντας. Ακολούθως, δίνει τις εκτιμήσεις του για το ποσοστό του πληθυσμού που έδωσαν την κάθε απάντηση, ήτοι $y^r = (y_1^r, y_2^r, \dots, y_m^r)$ με $y_k^r \in [0, 1]$ για κάθε $k \in \{1, 2, \dots, m\}$ και $\sum_{i=1}^m y_i^r = 1$.

Αφού, ο μηχανισμός συλλέξει όλες τις απαντήσεις υπολογίζει την πραγματική συχνότητα με την οποία εμφανίζεται κάθε απάντηση και το γεωμετρικό μέσο για την πρόβλεψη της κάθε απάντησης. Έχουμε:

$$\bar{x}_k = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n x_k^i$$

$$\bar{y}_k = \lim_{n \rightarrow \infty} \sqrt[n]{y_1 y_2 \cdots y_n}$$

Με χρήση των οποίων προκύπτει για κάθε ερώτηση η βαθμολογία πληροφορίας της ως $\log \frac{\bar{x}_k}{\bar{y}_k}$. Ο κάθε χρήστης παίρνει τη βαθμολογία πληροφορίας της απάντησης που έδωσε, ενώ επιβαρύνεται με ένα μέγεθος ανάλογο της απόστασης Kullback-Leibler της πραγματικής κατανομής με την εκτίμηση που έδωσε (γνωστή και ως σχετική εντροπία μεταξύ δύο κατανομών) [KL51], τη βαθμολογία πρόβλεψης. Συνεπώς η ανταμοιβή κάθε συμμετέχοντα γίνεται:

$$SCORE_r = \sum_{i=1}^m x_i^r \log \frac{\bar{x}_i}{y_i} + \alpha \sum_{i=1}^m \bar{x}_i \log \frac{y_i}{\bar{x}_i}$$

όπου με $0 \leq \alpha \leq 1$ συμβολίζουμε το ύψος της επιβάρυνσης. Η καλύτερη βαθμολογία πρόβλεψης που μπορεί να πάρει κανείς είναι το 0 και επιτυγχάνεται, εφόσον η πρόβλεψη ενός ερωτηθέντα συμπίπτει με την πραγματική κατανομή. Συνεπώς, η υψηλότερη πιθανή ανταμοιβή ισούται με τη βαθμολογία πληροφορίας της απάντησης που έδωσε ο ερωτηθέντας.

4.1.5 Ο μηχανισμός RBTS

Είδαμε πιο πάνω τον μηχανισμό BTS, ο οποίος, όμως, παρουσιάζει δύο προβλήματα. Αφενός, ο μηχανισμός χρειάζεται ένα μεγάλο αριθμό συμμετοχόντων για να λειτουργήσει, κάτι που τον κάνει μη εφαρμόσιμο για μικρότερους πληθυσμούς. Αφετέρου, η ανταμοιβή που δίνεται από το μηχανισμό μπορεί να είναι αρνητική (ανάλογα και με την τιμή της τιμής α). Αυτοί είναι οι λόγοι που οδήγησαν στη δημιουργία του μηχανισμού Robust BTS (RBTS) [WP12]. Ο μηχανισμός που παρουσιάζεται είναι δυαδικός (ως προς τις απαντήσεις), δηλαδή η ερώτηση έχει δύο πιθανές απαντήσεις ($m = 2$). Μια επέκταση του μηχανισμού για περισσότερες επιλογές παρουσιάζεται στο [RF13]. Μιας και στην περίπτωση μας χρησιμοποιούμε το δυαδικό μηχανισμό, τον παρουσιάζουμε συνοπτικά.

Ο μηχανισμός χρησιμοποιεί τους Αυστηρά Κατάλληλους Κανόνες Βαθμολόγησης (Strictly Proper Scoring Rules) [GR07] για να υπολογίσει την ανταμοιβή για τον κάθε συμμετέχοντα. Η πιο πάνω οικογένεια συναρτήσεων-κανόνων μπορεί να χρησιμοποιηθεί, ώστε να δώσει κίνητρα στους παίκτες για φιλαλήθη αποκάλυψη των απόψεών τους.

Ορισμός 4.2. Δεδομένης μιας τυχαίας μεταβλητής ω με τιμές στο $\Omega = \{0, 1\}$ και μιας πρόβλεψης $y \in [0, 1]$ για το ενδεχόμενο $\omega = 1$, ένας **Δυαδικός Κανόνας Βαθμολόγησης**, R , δίνει μια βαθμολόγηση $R(y, \omega)$ βάσει της πρόβλεψης και της τελικής τιμής της τυχαίας μεταβλητής.

Ορισμός 4.3. Ένας δυαδικός κανόνας βαθμολόγησης είναι κατάλληλος, εφόσον η μεγιστοποίηση της βαθμολόγησης ενός παίκτη επιτυγχάνεται μέσω της ειλικρινής πρόβλεψης του, $y \in [0, 1]$. Ένας κατάλληλος κανόνας είναι **Αυστηρά Κατάλληλος Κανόνας Βαθμολόγησης** αν η ειλικρινής πρόβλεψη είναι ο μόνος τρόπος μεγιστοποίησης της βαθμολόγησης του.

Ο κανόνας που χρησιμοποιείται για την παρουσίαση του μηχανισμού είναι ο δυαδικός τετραγωνικός κανόνας βαθμολόγησης που ορίζεται ως εξής:

$$\begin{aligned} R_q(y, \omega = 1) &= 2y - y^2 \\ R_q(y, \omega = 0) &= 1 - y^2 \end{aligned}$$

Με όλα τα παραπάνω είμαστε έτοιμοι να παρουσιάσουμε το μηχανισμό. Θεωρούμε ένα πεπερασμένο πλήθος σύνολο ερωτηθέντων, $\mathcal{N} = \{1, 2, \dots, N\}$, $N \geq 3$, οι οποίοι καλούνται να απαντήσουν στην ίδια ερώτηση (με δύο πιθανές απαντήσεις) και να δώσουν την εκτίμηση τους για την κατανομή του πληθυσμού. Ακολούθως, ο μηχανισμός ορίζει για κάθε ερωτηθέντα

i τον παίκτη αναφοράς, $j = (i + 1) \bmod N$, και τον παίκτη ομοτιμίας, $k = (i + 2) \bmod N$. Ορίζουμε την τιμή $\delta = \min(y_j, 1 - y_j)$ και βάσει αυτής υπολογίζουμε:

$$\begin{aligned} y'_i &= y_j + \delta, & x_i &= 1 \\ y'_i &= y_j - \delta, & x_i &= 0 \end{aligned}$$

Τελικά, δίνεται η βαθμολογήση RBTS ως:

$$SCORE_{RBTS} = R_q(y'_i, x_k) + R_q(y_i, x_k)$$

Και πάλι, όπως και τον BTS, ο πρώτος όρος ορίζεται ως βαθμολογία πληροφορίας και ο δεύτερος ως βαθμολογία πρόβλεψης.

Ο μηχανισμός μπορεί να εφαρμοστεί για οποιοδήποτε αριθμό πληθυσμού, καθώς το μόνο που είναι απαραίτητο είναι η ύπαρξη τουλάχιστον άλλων δύο ατόμων για υπολογισμό της βαθμολογίας του τρίτου. Ακόμη, ο κανόνας R_q δίνει τιμές στο διάστημα $[0, 1]$ και κατ' επέκταση η βαθμολογήση κάθε συμμετέχοντα παίρνει τιμές στο διάστημα $[0, 2]$ σαν άθροισμα δύο όρων του. Δεν παρουσιάζεται, λοιπόν, το φαινόμενο αρνητικών βαθμολογήσεων που υπάρχει στον BTS. Τέλος, ο μηχανισμός είναι συμβατός κινήτρου αφού βασίζεται στον κανόνα R_q ο οποίος με ένα σύντομο λήμμα αποδεικνύεται, επίσης, συμβατός κινήτρου.

Λήμμα 4.1. Έστω μια δημοσκόπηση στην οποία χρησιμοποιείται ο κανόνας R_q για βαθμολογήση και έστω $p \in [0, 1]$ η πραγματική πρόβλεψη ενός συμμετέχοντα σε αυτή. Έστω, ακόμη πως οι προβλέψεις πρέπει να ανήκουν στο σύνολο $Y \subseteq [0, 1]$. Τότε, η βέλτιστη στρατηγική για το συμμετέχοντα είναι να δώσει πρόβλεψη, $p \in Y$, η οποία ελαχιστοποιεί την ποσότητα $|p - y|$.

Απόδειξη. Η μέση βαθμολογήση για κάθε συμμετέχοντα με πραγματική πρόβλεψη p είναι $E[y] = p(2y - y^2) + (1 - p)(1 - y^2)$. Συνεπώς, η αναμενόμενη βαθμολογήση εάν δοθεί η πραγματική πρόβλεψη είναι $E[p] = p(2p - p^2) + (1 - p)(1 - p^2) = p^2 - p + 1$. Επομένως, η ζημιά ενός συμμετέχοντα αν δεν δώσει ειλικρινή απάντηση δίνεται από τη σχέση $E[p] - E[y] = p^2 - p + 1 - 2yp + y^2p - 1 + y^2 + p - y^2p = p^2 - 2yp + y^2 = (p - y)^2$, η οποία ελαχιστοποιείται στο 0 για $y = p$ και είναι ισοδύναμο με ελαχιστοποίηση της ποσότητας $|p - y|$. ■

4.1.6 Μοντελοποίηση Φήμης

Στα οικονομικά μοντέλα και στην καθημερινή ζωή χαρακτηριζόμαστε από φήμη ως προς την αξιοπιστία μας για στον τομέα που εργαζόμαστε ή προσφέρουμε υπηρεσίες, αλλά ακόμα και για τα προσωπικά χαρακτηριστικά του καθενός [Cab05]. Για παράδειγμα, μια εταιρία κρίνεται ως προς τη φήμη της για την ποιότητα των προϊόντων που παράγει.

Η φήμη μαθηματικά μοντελοποιείται με δύο μηχανισμούς. Πρώτα, είναι ο μηχανισμός Bootstrap ο οποίος βασίζεται στις πολλές αλληλεπιδράσεις μεταξύ των συμμετοχόντων, ώστε να αναπτυχθεί μεταξύ τους εμπιστοσύνη, η οποία εν προκειμένω ισοδυναμεί με τη φήμη. Το δεύτερο μοντέλο (και αυτό που χρησιμοποιούμε σε αυτή τη διπλωματική) είναι η πίστη κατά Bayes (Bayesian Belief). Συγκεκριμένα, στο παράδειγμα με το προϊόν και την ποιότητά του,

έχουμε ένα πωλητή ο οποίος ξεκινά με μια αρχική φήμη για την ποιότητα του προϊόντος του. Όσο περισσότεροι καταναλωτές το αγοράζουν, η φήμη του πωλητή ενημερώνεται ανάλογα με τους ευχαριστημένους από την ποιότητα του προϊόντος καταναλωτές. Περισσότεροι ευχαριστημένοι καταναλωτές δίνουν ψηλότερη φήμη από την αρχική και περισσότεροι δυσαρεστημένοι δίνουν χαμηλότερη φήμη από την αρχική. Το μοντέλο αυτό δεν είναι τίποτα περισσότερο από ενημέρωση πιθανοτήτων κατά Bayes (Bayesian Updating), στο οποίο η πιθανότητα ένα προϊόν να είναι «καλό» ενημερώνεται με βάση το δείγμα των καταναλωτών που το έχουν αγοράσει και το οποίο καταλήγει σε μια κατάλληλα διατυπωμένη δεσμευμένη πιθανότητα.

Συνεχίζοντας το πιο πάνω παράδειγμα, ας θεωρήσουμε ότι ένας πωλητής μπορεί να ανήκει σε μια από δύο κατηγορίες. Είτε είναι «καλός» και παράγει προϊόντα που δουλεύουν με μια πιθανότητα a_H , είτε είναι «κακός» και τα προϊόντα του δουλεύουν με πιθανότητα a_L , $0 < a_L < a_H < 1$. Θεωρούμε λογικούς αγοραστές που προσφέρουν τιμή αγοράς ίση με την πιθανότητα του προϊόντος να δουλεύει. Έστω μ η φήμη του πωλητή. Τότε οι αγοραστές είναι πρόθυμοι να αγοράσουν το προϊόν με τιμή $p(\mu) = a_H\mu + (1 - \mu)a_L$. Παραγωγίζοντας το πιο πάνω, έχουμε $p'(\mu) = a_H - a_L > 0$ και συνεπώς η τιμή είναι γνησίως αύξουσα ως προς το μ , δηλαδή οι αγοραστές είναι πρόθυμοι να πληρώσουν περισσότερα για ένα προϊόν από καλής φήμης πωλητή.

Αυτό που μένει είναι να δούμε πως αλλάζει η φήμη κατόπιν των αγορών που γίνονται. Ας συμβολίσουμε με μ_0 την αρχική φήμη ενός πωλητή, με S τον αριθμό ευχαριστημένων αγοραστών και με F τον αριθμό δυσαρεστημένων αγοραστών. Τότε η φήμη, μ του πωλητή γίνεται:

$$\mu = \frac{\mu_0 a_H^S (1 - a_H)^F}{\mu_0 a_H^S (1 - a_H)^F + (1 - \mu_0) a_L^S (1 - a_L)^F}$$

Είναι εύκολο να δει κανείς πως η φήμη είναι αύξουσα συναρτήσει των θετικών απαντήσεων, S και φθίνουσα συναρτήσει των αρνητικών απαντήσεων, F . Μια σύντομη απόδειξη παρατίθεται στο Παράρτημα Α.

Τα πιο πάνω, ασφαλώς, γενικεύονται δημιουργώντας ένα μοντέλο φήμης, όπου οντότητες κατηγορίας Α (πχ αγοραστές) έχουν μια άποψη (πχ προϊόντα καλής ποιότητας) για μια οντότητα κατηγορίας Β (πχ πωλητής).

4.2 Αναλυτική Παρουσίαση του Μοντέλου

Αφού έχουμε παρουσιάσει αναλυτικά τα εργαλεία που χρησιμοποιούμε στο μοντέλο είμαστε έτοιμοι να το παρουσιάσουμε ολοκληρωμένα. Παρουσιάζουμε, αρχικά, τους διάφορους παίχτες του μοντέλου και στη συνέχεια τα διάφορα μεγέθη που τους χαρακτηρίζουν. Ακολούθως, παρουσιάζεται η λειτουργία του μοντέλου διαιρεμένη σε τρία επίπεδα. Υπενθυμίζουμε, πως το μοντέλο αφορά ένα περιβάλλον με Fog Computing στο οποίο οι χρήστες ακολουθούν στρατηγική Always Offload και έχουν να επιλέξουν ανάμεσα στους διαθέσιμους servers του περιβάλλοντος για μεταφορά των δεδομένων τους. Επισημαίνουμε, πως το παραπάνω μοντέλο επαναλαμβάνεται συνεχώς με τη μορφή γύρων (timeslots).

4.2.1 Μεγέθη και Παίχτες του Μοντέλου

Ορίζουμε πρώτα το σύνολο των servers (από εδώ και στο εξής οι όροι servers, Fog Nodes και Εξυπηρετητής θεωρούνται ισοδύναμοι και χρησιμοποιούνται αυθαίρετα) ως $N = \{1, 2, \dots, |N|\}$. Για κάθε $k \in N$ ορίζουμε τα εξής μεγέθη:

- F_k : Η υπολογιστική ισχύς που διαθέτει ο κάθε Εξυπηρετητής [$\frac{CPU Cycles}{sec}$].
- \overline{B}_k : Η μέγιστη χωρητικότητα δεδομένων που μπορεί να δεχτεί ο Εξυπηρετητής [$bits$].

Οι άλλοι παίχτες του μοντέλου είναι οι χρήστες (ή Fog Users) που θέλουν να χρησιμοποιήσουν τους πόρους που προσφέρουν οι Fog Nodes και ανήκουν στο σύνολο $U = \{1, 2, \dots, |U|\}$. Οι ίδιοι δεν χαρακτηρίζονται από κάποιο μέγεθος, ωστόσο θεωρούμε πως όλοι οι χρήστες έχουν σε κάθε γύρο ένα νέο πακέτο δεδομένων (εφαρμογή) για επεξεργασία. Το κάθε πακέτο δεδομένων χαρακτηρίζεται από την τούπλα $tu = (I_{tu}, C_{tu}, \Phi_{tu})$ ως εξής:

- I_{tu} : Το μέγεθος της εφαρμογής [$bits$].
- C_{tu} : Οι πόροι που χρειάζεται η εφαρμογή για επεξεργασία [$CPU Cycles$].
- Φ_{tu} : Η παράμετρος εντατικότητας που χαρακτηρίζει το πόσο απαιτητική είναι η εφαρμογή, Ορίζεται σαν το πηλίκο $\frac{C_{tu}}{I_{tu}}$ [$\frac{CPU Cycles}{bits}$].

Θεωρούμε, ακόμη, ένα πίνακα \mathcal{D} , διαστάσεων $|N| \times |U|$, στον οποίο περιέχονται οι αποστάσεις μεταξύ Εξυπηρετητών και Χρηστών του μοντέλου:

$$\mathcal{D} = \begin{bmatrix} d_{11} & \dots & d_{1|U|} \\ \vdots & \ddots & \vdots \\ a_{|N|1} & \dots & d_{|N||U|} \end{bmatrix}$$

Τέλος, ορίζουμε, τα χαρακτηριστικά επικοινωνίας (η ισχύς μετάδοσης, $p_{u,k}$ και το κέρδος του καναλιού, $g_{u,k}$) μεταξύ κάθε πιθανού ζευγαριού Εξυπηρετητή, k και Χρήστη, u , ως εξής:

$$p_{u,k} = \left[\frac{d_{ku}}{\max_i(d_{ki})} \right]^{\theta_1} \quad g_{u,k} = \left(\frac{1}{d_{ku}} \right)^{\theta_2}$$

Και ανάλογα με τον πίνακα αποστάσεων μπορούμε να ορίσουμε δύο πίνακες (με τις ίδιες διαστάσεις) που περιέχουν τα πιο πάνω χαρακτηριστικά:

$$\mathcal{P} = \begin{bmatrix} p_{11} & \dots & p_{1|U|} \\ \vdots & \ddots & \vdots \\ p_{|N|1} & \dots & p_{|N||U|} \end{bmatrix} \quad \mathcal{G} = \begin{bmatrix} g_{11} & \dots & g_{1|U|} \\ \vdots & \ddots & \vdots \\ g_{|N|1} & \dots & g_{|N||U|} \end{bmatrix}$$

Τα πιο πάνω μεγέθη είναι σταθερά και ορίζονται είτε στο ξεκίνημα κάθε γύρου (η εφαρμογή προς επεξεργασία) είτε στο ξεκίνημα του μοντέλου (όλα τα υπόλοιπα). Οι τιμές τους χρησιμοποιούνται κατάλληλα στα τρία επίπεδα, για καθορισμό των αναγκαίων τιμών κάθε επιπέδου.

4.2.2 Πρώτο Επίπεδο - Επιλογή Εξυπηρετητή από Χρήστες

Στο πρώτο επίπεδο του αλγορίθμου έχουμε την επιλογή του κατάλληλου Εξυπηρετητή από τους Χρήστες. Οι Χρήστες μοντελοποιούνται σαν SLA εφαρμόζοντας τον κανόνα μάθησης LRI που είδαμε στην παράγραφο 4.1.2. Όπως είδαμε, χρειαζόμαστε ένα ορισμό της κανονικοποιημένης ανταμοιβής του κάθε χρήστη, ώστε να γίνεται η ενημέρωση των πιθανοτήτων του μηχανισμού.

Ορίζουμε, πρώτα, μια σειρά από μεγέθη, τα οποία εξαρτώνται από την κατανομή των Χρηστών στους Εξυπηρετητές. Έτσι, για την πρώτη επανάληψη του μηχανισμού επιλογής, θεωρούμε πως ο κάθε Χρήστης επιλέγει τυχαία από μια ομοιόμορφη κατανομή τον Εξυπηρετητή του. Για το ρυθμό μετάδοσης μεταξύ Χρήστη και Εξυπηρετητή έχουμε:

$$R_{u,k} = W_k \log_2 \left(1 + \frac{p_{u,k} \cdot g_{u,k}}{G_0 + \sum_{u' \in U_k, u' \neq u} p_{u',k} \cdot g_{u',k}} \right)$$

Όπου με W_k συμβολίζουμε τη χωρητικότητα του καναλιού, με G_0 τη διασπορά του Προσθετικού Λευκού Γκαουσιανού Θρυβού (AWGN) και με U_k των σύνολο των Χρηστών που εξυπηρετούνται από τον Fog Server k . Ακολούθως υπολογίζουμε το μερίδιο υπολογισμού που λαμβάνει ο Χρήστης u που έχει συσχετιστεί με τον Εξυπηρετητή k . Συμβολίζουμε με $f_{u,k}$ και γράφουμε:

$$f_{u,k} = \frac{\Phi_{tu}}{\sum_{u \in U_k} \Phi_{tu}} \cdot \left[1 - \frac{\sum_{u \in U_k} I_{tu}}{B_k} \right] \cdot \frac{\sum_{i=0}^{t-1} RBTS_{iu}}{\sum_{i=0}^{t-1} \sum_{u \in U} RBTS_{iu}} \cdot F_k$$

Βλέπουμε πως στο γινόμενο υπάρχουν τέσσερις όροι. Ο τελευταίος όρος, F_k , είναι η υπολογιστική ισχύς που διαθέτει ο Εξυπηρετητής k σε πλήρη λειτουργία. Οι υπόλοιποι όροι μας δίνουν το ποσοστό αυτής της ισχύος που αντιστοιχεί στον κάθε Χρήστη. Ο πρώτος όρος είναι ο παράγοντας δικαιοσύνης και αντιστοιχεί στον κάθε εξυπηρετούμενο Χρήστη ένα ποσοστό ισχύος ανάλογα με το πόσο απαιτητική είναι η εφαρμογή του. Ο δεύτερος όρος συμβολίζει την απώλεια ισχύος λόγω συμφόρησης στον Εξυπηρετητή από τα δεδομένα των Χρηστών που αποθηκεύονται σε αυτόν. Τέλος, ο τρίτος όρος συμβολίζει τη συνεισφορά του χρήστη στη βαθμολόγηση RBTS (δεύτερο επίπεδο) και εκφράζει το πηλίκο της συνολικής βαθμολογίας του Χρήστη u μέχρι το γύρο $t - 1$ προς τη συνολική βαθμολογία όλων των Χρηστών μέχρι το γύρο $t - 1$. Ποιοτικά, όσο πιο ειλικρινείς είναι οι απαντήσεις του κάθε χρήστη, τόσο μεγαλύτερο και το ποσοστό ισχύος που παίρνει. Σημειώνεται ότι στο γύρο 0, όπου ακόμα κανείς Χρήστης δεν έχει βαθμολόγηση RBTS, ο τρίτος όρος αγνοείται. Μπορούμε, πλέον, να προσδιορίσουμε την επιβάρυνση χρόνου (time overhead) την οποία δέχεται ο Χρήστης u . Έχουμε:

$$O_t^u = \frac{I_{tu}}{R_{u,k}} + \frac{C_{tu}}{f_{u,k}}$$

Ο πρώτος όρος συμβολίζει το χρόνο που χρειάζεται η εφαρμογή για να μεταδοθεί στον Εξυπηρετητή k , ενώ ο δεύτερος όρος συμβολίζει το χρόνο που χρειάζεται για να εξυπηρετηθεί

η εφαρμογή στον Εξυπηρετητή. Με παρόμοιο τρόπο ορίζουμε και την ενεργειακή επιβάρυνση του Χρήστη u :

$$O_e^u = \frac{I_{tu} \cdot P_{u,k}}{R_{u,k}}$$

Όπως είναι λογικό, η ενεργειακή επιβάρυνση ορίζεται ως το γινόμενο του χρόνου που χρειάζεται για μετάδοση της εφαρμογής επί την ισχύ της συσκευής που την αποστέλνει. Τέλος, η συνολική επιβάρυνση δίνεται από τη σχέση $O^u = O_e^u + O_t^u$.

Είμαστε, πλέον, έτοιμοι να ορίσουμε την ανταμοιβή των παικτών και τα συστατικά της. Δίνουμε πρώτα τη σχέση υπολογισμού της ανταμοιβής (πραγματικής και κανονικοποιημένης) του Χρήστη u που εξυπηρετείται από τον Fog Node k και, ακολούθως, επεξηγούμε τη συνεισφορά του κάθε όρου. Έχουμε:

$$R_k^u = \frac{\mu_k \cdot RF_k}{Congestion_k \cdot UCCP_k} \quad \hat{R}_k^u = \sqrt[4]{\frac{R_k^u}{\sum_k \sum_{u \in U_k} R_k^u}}$$

Όπως μπορούμε να δούμε από την πιο πάνω σχέση, ο υπολογισμός της ανταμοιβής εξαρτάται αποκλειστικά από μεγέθη που έχουν να κάνουν με τον Εξυπηρετητή. Οπότε, όλοι οι Χρήστες που βρίσκονται στον ίδιο Εξυπηρετητή λαμβάνουν την ίδια ανταμοιβή/ευχαριστήση. Με τον όρο μ_k συμβολίζουμε τη φήμη του κάθε Εξυπηρετητή (η ενημέρωσή της γίνεται στο τρίτο επίπεδο) και όσο καλύτερη είναι τόσο πιο ευχαριστημένος είναι ο Χρήστης. Όπως θα δούμε και στα επόμενα επίπεδα η φήμη εξαρτάται άμεσα από τις απαντήσεις των Χρηστών, οπότε αυτό τους δίνει ένα ακόμα κίνητρο να είναι ειλικρινείς στην αξιολόγηση που γίνεται στο δεύτερο επίπεδο. Με τον όρο RF_k συμβολίζουμε την ευχαρίστηση του Χρήστη από την υπολογιστική ισχύ που διαθέτει ο Εξυπηρετητής στους Χρήστες του σε σχέση με τους υπόλοιπους Εξυπηρετητές:

$$RF_k = \frac{\sum_{u \in U_k} f_{u,k}}{|U_k|} = \frac{\sum_k \sum_{U_k} \frac{f_{u,k}}{|U_k|}}{\sum_k \sum_{U_k} \frac{f_{u,k}}{|U_k|}}$$

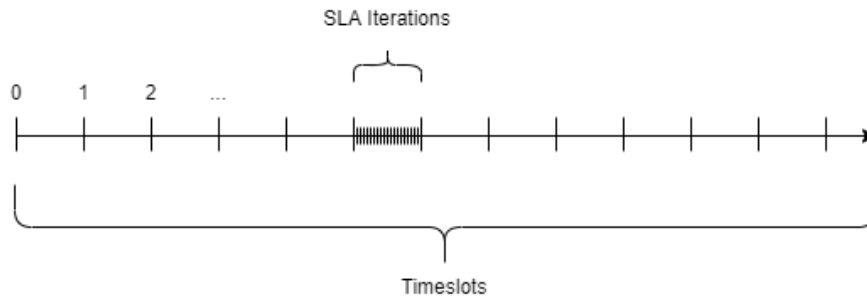
Με τον όρο $UCCP_k$ (User-Centric Cluster Performance), συμβολίζουμε την επιβάρυνση που έχουν οι Χρήστες από τις επιβαρύνσεις άλλων Χρηστών με τους οποίους μοιράζονται τον ίδιο Εξυπηρετητή. Προφανώς, όσο μεγαλύτερο αυτό το μέγεθος, τόσο μεγαλύτερη και η δυσαρέσκεια που προκαλεί. Συμβολίζουμε με T τη διάρκεια του κάθε γύρου και με e_u την ενεργειακή διαθεσιμότητα κάθε Χρήστη (πχ την μπαταρία του) και γράφουμε:

$$UCCP_k = \frac{\sum_{u \in U_k} \frac{O_t^u}{T} + \frac{O_e^u}{e_u}}{|U_k|}$$

Όπως βλέπουμε το μέγεθος εκφράζει τη μέση κανονικοποιημένη επιβάρυνση που έχουν οι Χρήστες που εξυπηρετούνται στον Εξυπηρετητή k .

Τέλος, με τον όρο $Congestion_k$ συμβολίζουμε το ποσοστό της συμφόρησης του Εξυπηρετητή σε σχέση με τους υπόλοιπους. Όσο πιο χαμηλή η συμφόρηση στον Εξυπηρετητή, τόσο

Σχήμα 4.1: Οι επαναλήψεις του μηχανισμού επιλογής συγκριτικά με το υπόλοιπο πλαίσιο



πιο χαρούμενοι είναι οι Χρήστες που εξυπηρετεί:

$$Congestion_k = \frac{\sum_{U_k} I_{tu}}{\sum_k \sum_{U_k} I_{tu}}$$

Με χρήση της κανονικοποιημένης ανταμοιβής μπορούμε να χρησιμοποιήσουμε τον κανόνα μάθησης LRI για να ενημερώσουμε τις πιθανότητες επιλογής Εξυπηρετητή. Όπως επισημάναμε, το διάνυσμα επιλογής Εξυπηρετητή για κάθε χρήστη ξεκινά με μια ομοιόμορφη κατανομή και στη συνέχεια ενημερώνεται κατάλληλα. Ο μηχανισμός ανακοινώνει σύγκλιση όταν όλοι οι Χρήστες έχουν ένα Fog Node με πιθανότητα επιλογής > 0.95 . Συνεπώς, ο μηχανισμός επιλογής απαιτεί την εκτέλεση πολλών επαναλήψεων για τη σύγκλιση του στον ίδιο γύρο. Στο σχήμα 4.1 βλέπουμε μια σχηματική απεικόνιση του πιο πάνω. Για λόγους πληρότητας παραθέτουμε εκ νέου τον κανόνα μάθησης, συμβολίζοντας με Pr_{uk} την πιθανότητα επιλογής του Εξυπηρετητή k από το Χρήστη u και με m τον Εξυπηρετητή που επέλεξε ο Χρήστης κατά την προηγούμενη επανάληψη:

$$\begin{aligned} Pr_{uk}(i+1) &= Pr_{uk}(i) + b \cdot \hat{R}_k^u \cdot (1 - Pr_{uk}(i)) & k = m \\ Pr_{uk}(i+1) &= Pr_{uk}(i) - b \cdot \hat{R}_k^u \cdot Pr_{uk}(i) & k \neq m \end{aligned}$$

4.2.3 Δεύτερο Επίπεδο - Αξιολόγηση Εξυπηρετητών

Στο δεύτερο επίπεδο του αλγορίθμου οι Χρήστες καλούνται να αξιολογήσουν τους Εξυπηρετητές που επέλεξαν στο πρώτο επίπεδο. Συγκεκριμένα, καλούνται να απαντήσουν στην ερώτηση «Είστε ευχαριστήμενοι από τον Fog Node που σας εξυπηρέτησε». Ακολουθούμε λοιπόν τα βήματα του μηχανισμού RBTS, όπως αυτά περιγράφησαν στην παράγραφο 4.1.5.

Μια σημαντική διαφορά είναι η εξής. Είναι βέβαιο πως στην αρίθμηση του συνόλου $U = \{1, 2, \dots, |U|\}$ διαδοχικοί Χρήστες θα ανατεθούν σε διαφορετικούς Εξυπηρετητές. Προφανώς, δεν έχει νόημα οι Χρήστες να βαθμολογηθούν για την ειλικρίνιά τους από το μηχανισμό, εφόσον η απάντηση που δίνουν αφορά διαφορετικούς Εξυπηρετητές. Οπότε, για κάθε Εξυπηρετητή k εκτελούμε ξεχωριστά το μηχανισμό (συνολικά k φορές) επί του συνόλου U_k των Χρηστών που έχουν συσχετιστεί μαζί του και των απαντήσεων τους.

Και πάλι για λόγους πληρότητας παρουσιάζουμε συνοπτικά τα βήματα του μηχανισμού για τυχαίο Εξυπηρετητή k .

1. Θεωρούμε μια αρίθμηση των Χρηστών που εξυπηρετούνται στον Εξυπηρετητή, $U_k = \{1, 2, \dots, |U_k|\}$.
2. Ο κάθε Χρήστης, u , δίνει την άποψη του για τον Εξυπηρετητή, $x_k^u \in \{0, 1\}$ και την εκτίμηση του για την κατανομή των θετικών απαντήσεων, $y_k^u \in [0, 1]$.
3. Για κάθε Χρήστη, u , επιλέγεται ο παίκτης αναφοράς, $j_u = (u + 1) \bmod |U_k|$ και ο παίκτης ισοτιμίας, $p_u = (u + 2) \bmod |U_k|$ (με βάση την αρίθμηση του βήματος 1).
4. Υπολογίζεται η τιμή $\delta = \min(y_{j_u}, 1 - y_{j_u})$.
5. Υπολογίζεται η τιμή y'_u σύμφωνα με τις σχέσεις:

$$\begin{aligned} y'_u &= y_{j_u} + \delta, & x_u &= 1 \\ y'_u &= y_{j_u} - \delta, & x_u &= 0 \end{aligned}$$

6. Ανακοινώνεται η τελική βαθμολόγηση του Χρήστη u για το γύρο t :

$$RBTS_{tu} = R_q(y'_u, x_{p_u}) + R_q(y_u, x_{p_u})$$

Όπου R_q ο δυαδικός τετραγωνικός κανόνας βαθμολόγησης

4.2.4 Τρίτο Επίπεδο - Ενημέρωση Φήμης Εξυπηρετητή

Στο τρίτο και τελευταίο επίπεδο έχουμε την ενημέρωση της φήμης των Εξυπηρετητών ανάλογα με τις αξιολογήσεις που πήραν στο δεύτερο επίπεδο. Όπως είδαμε στην παράγραφο 4.1.6 η φήμη ενός πωλητή (εν προκειμένω του Εξυπηρετητή που «πωλεί» τους υπολογιστικούς πόρους στους Χρήστες) θα πρέπει να εξαρτάται από τις καλές ή τις κακές απόψεις που έχουν οι πελάτες του γι' αυτόν.

Για να εκμηδενίσουμε την επιρροή κακόβουλων και ανειλικρινών Χρηστών δεν λαμβάνουμε υπόψη μας όλες τις απαντήσεις ισοδύναμα, αλλά επιχειρούμε να ανακηρύξουμε ένα νικητή από τη δημοσκόπηση που πραγματοποιήθηκε στο δεύτερο επίπεδο λαμβάνοντας υπόψη την ποιότητα της κάθε απάντησης. Έτσι, επιλέγουμε την άποψη για την οποία η μέση βαθμολογία που έλαβαν οι υποστηρικτές της είναι η ψηλότερη. Γράφουμε:

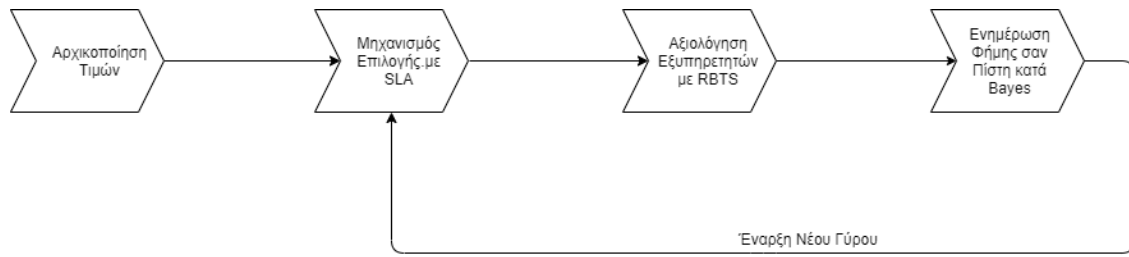
$$\overline{U}_{k,1} = \frac{\sum_{u \in U_k, x_k^u=1} RBTS_u}{\sum_{u \in U_k} x_k^u} \quad \overline{U}_{k,0} = \frac{\sum_{u \in U_k, x_k^u=0} RBTS_u}{|U_k| - \sum_{u \in U_k} x_k^u}$$

Και τελικά:

$$ans_k^* = \arg\{\max_i \overline{U}_{k,i}\}$$

Αφού λάβουμε, έτσι, μια ειλικρινή άποψη για την ποιότητα του Εξυπηρετητή k ενημερώνουμε τη φήμη του σύμφωνα με την εξίσωση της παραγράφου 4.1.6. Για $ans_k^* = 1$ έχουμε αύξηση του μετρητή επιτυχιών, $s_k = s_k + 1$, διαφορετικά για $ans_k^* = 0$ έχουμε αύξηση του

Σχήμα 4.2: Μια επισκόπηση του αλγορίθμου



μετρητή αποτυχιών $f_k = f_k + 1$. Τέλος, η φήμη του Εξυπηρετητή ενημερώνεται σύμφωνα με τη σχέση:

$$\mu_k = \frac{\mu_{0_k} a_{H_k}^{s_k} (1 - a_{H_k})^{f_k}}{\mu_{0_k} a_{H_k}^{s_k} (1 - a_{H_k})^{f_k} + (1 - \mu_{0_k}) a_{L_k}^{s_k} (1 - a_{L_k})^{f_k}}$$

Όπου με a_{H_k} συμβολίζουμε την πιθανότητα ο Εξυπηρετητής k να αφήσει ευχαριστημένο ένα Χρήστη και με a_{L_k} διαφορετικά, ενώ με μ_{0_k} συμβολίζουμε την αρχική φήμη του.

4.2.5 Σύνοψη Αλγορίθμου

Έχουμε ολοκληρώσει την παρουσίαση του μοντέλου. Ανακεφαλαιώνουμε συνοπτικά. Μια επισκόπηση ολόκληρου του μοντέλου φαίνεται στο σχήμα 4.2.

1. Αρχικοποιούνται τα σύνολα και οι σταθερές του συστήματος.
2. Ξεκινά ένας νέος γυρός. Κάθε Χρήστης δημιουργεί ένα πακέτο δεδομένων το οποίο επιθυμεί να επεξεργαστεί στον Εξυπηρετητή.
3. Ενεργοποιείται ο μηχανισμός επιλογής. Σημειώνεται, και πάλι, ότι ο μηχανισμός εκτελεί πολλαπλές επαναλήψεις μέσα στον ίδιο γύρο μέχρι να καταλήξει σε ισορροπία.
4. Αφού ο κάθε Χρήστης συνδεθεί με ένα Εξυπηρετητή καταθέτει τη γνώμη του για την ευχαρίστησή του και λαμβάνει μια βαθμολόγηση.
5. Η φήμη του Εξυπηρετητή ενημερώνεται βάσει των αποτελεσμάτων του βήματος 5.
6. Επιστροφή στο βήμα 2.

Αριθμητικά Αποτελέσματα

Στο κεφάλαιο 4 παρουσιάσαμε το μοντέλο αναλυτικά, καθώς και τους μηχανισμούς που χρησιμοποιούνται σε αυτό. Σε αυτό το κεφάλαιο παρουσιάζουμε αριθμητικά αποτελέσματα, τα οποία χρησιμοποιούμε για αξιολόγησή του. Ακόμη, σε διάφορα σημεία, εξετάζουμε τη χρήση εναλλακτικών επιλογών και παρουσιάζουμε συγκριτικά αποτελέσματα για την ποιότητα αυτών.

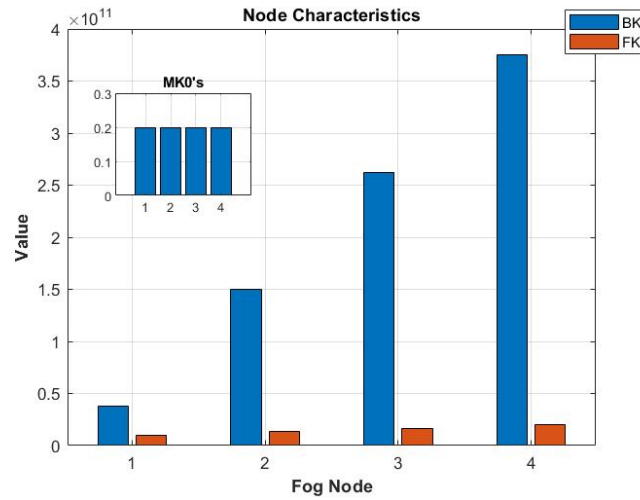
5.1 Αποτελέσματα Μοντέλου

5.1.1 Τιμές Σταθερών και Περιβάλλοντος

Στο μοντέλο που παρουσιάσαμε υπάρχουν σταθερές που επηρεάζουν τα αποτελέσματα. Είναι σημαντικό να αναφέρουμε τις τιμές αυτές, ώστε να υπάρχει μια καλή εικόνα για τα αποτελέσματα που εξάγουμε. Συγκεκριμένα, θέτουμε τις εξής τιμές:

- Όσον αφορά τον αριθμό των παικτών έχουμε $|N| = 4$ για τους Εξυπηρετητές και $|U| = 80$ για τους Χρήστες.
- Στα χαρακτηριστικά των Χρηστών, σε κάθε γύρο του μοντέλου έχουμε τη δημιουργία του πακέτου δεδομένων για μεταφορά στον Εξυπηρετητή. Αυτό γίνεται με τη δημιουργία τυχαίων τιμών για τις τιμές I_{tu} από το σύνολο $[50, 150] \cdot 10^6$ και C_{tu} από το σύνολο $[1, 2] \cdot 10^9$. Ακόμη, η ενεργειακή διαθεσιμότητα του κάθε Χρήστη είναι μια τυχαία τιμή από το σύνολο $[200, 20000]$, υποθέτοντας ότι με γεμάτη μπαταρία ισοδυναμεί με 20000 Joules.
- Στα χαρακτηριστικά των Εξυπηρετητών επιτρέπουμε τιμές για τη χωρητικότητα, \overline{B}_k στο σύνολο $[37.5, 375] \cdot 10^9$. Επειδή, στο μοντέλο μας δεν θεωρούμε την πιθανότητα αποτυχίας ενός Εξυπηρετητή είναι σημαντικό να γνωρίζουμε ότι ακόμα και αν όλοι οι Χρήστες επιλέξουν τον ίδιο Εξυπηρετητή, αυτός θα πετύχει. Με τη ρύθμιση αυτή εξασφαλίζουμε πως κανένας Εξυπηρετητής δεν αποτυγχάνει εξυπηρετώντας μέχρι και 250 Χρήστες μέγιστων απαιτήσεων. Για την υπολογιστική ισχύ, F_k , έχουμε τιμές από το σύνολο $[10, 20] \cdot 10^9$. Για τις ανάγκες σχηματισμού «καλών» και «κακών» Εξυπηρετητών, επιλέξαμε να αντιστοιχίσουμε γραμμικά τον κάθε Εξυπηρετητή με τιμές από

Σχήμα 5.1: Τα χαρακτηριστικά των εξυπηρετητών



τα δύο σύνολα. Ακόμη, η αρχική φήμη $\mu_{k_0} = 0.2$ για όλους τους Εξυπηρετητές. Τα χαρακτηριστικά των εξυπηρετητών φαίνονται στο Σχήμα 5.1.

- Οι αποστάσεις μεταξύ Χρηστών και Εξυπηρετητών λαμβάνονται τυχαία από το σύνολο $(10, 400)$ [m].
- Για τα χαρακτηριστικά του καναλιού ορίζουμε τις τιμές $\theta_1 = 2$ και $\theta_2 = 3$.
- Για τον υπολογισμό της τιμής $UCCP_k$ θεωρούμε $T = 1800$ δευτερόλεπτα, δηλαδή κάθε γύρος διαρκεί μισή ώρα.
- Για τον υπολογισμό του ρυθμού μετάδοσης θεωρούμε χωρητικότητα καναλιού, $W = 5$ Mbps και τη διάσπορα του θορύβου, $G_0 = 1 \cdot 10^{-13}$.
- Για τον κανόνα μάθησης LRI θέτουμε $b = 0.7$.
- Τέλος, για την ενημέρωση της φήμης των Εξυπηρετητών θεωρούμε $a_{H_k} = 0.51$ και $a_{L_k} = 0.49$ για όλους τους Εξυπηρετητές.

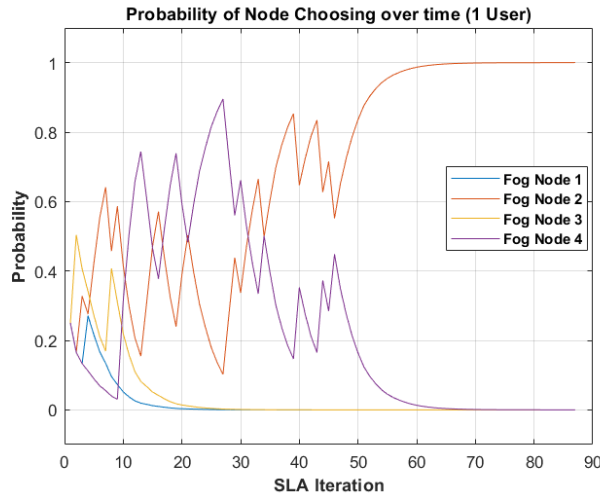
Επισημαίνουμε, πως τα αποτελέσματα για το πρώτο και το τρίτο επίπεδο προέκυψαν από την ίδια εκτέλεση του μοντέλου. Αφού το αφήσαμε να τρέξει κανονικά για εκατό γύρους (λαμβάνοντας έτσι τις μετρήσεις για το τρίτο επίπεδο), πήραμε στον επόμενο γύρο τα αποτελέσματα για το μηχανισμό επιλογής. Είναι σημαντικό πως στο τέλος των εκατό γύρων το διάστημα φήμης των εξυπηρετητών ήταν $[0.1814, 0.206, 0.296, 0.367]$, ακολουθώντας πλήρως την κατάταξη των χαρακτηριστικών που θέσαμε στο ξεκίνημα.

5.1.2 Πρώτο Επίπεδο - Αποτελέσματα Μηχανισμού Επιλογής

Το πρώτο επίπεδο αποτελεί την καρδιά του μοντέλου, καθώς σε αυτό το επίπεδο οι Χρήστες καλούνται να επιλέξουν τον Εξυπηρετητή, στον οποίο θα ανατεθεί η εφαρμογή τους και τον

οποίο θα αξιολογήσουν στη συνέχεια. Όπως περιγράψαμε διατηρούμε ένα διάνυσμα πιθανοτήτων (για κάθε Χρήστη), στο οποίο ενημερώνουμε τις πιθανότητες μέχρι ο μηχανισμός να συγκλίνει (πιθανότητα επιλογής ≥ 0.95) για όλους. Στο Σχήμα 5.2 βλέπουμε την εξέλιξη της πιθανότητας για ένα τυχαίο Χρήστη.

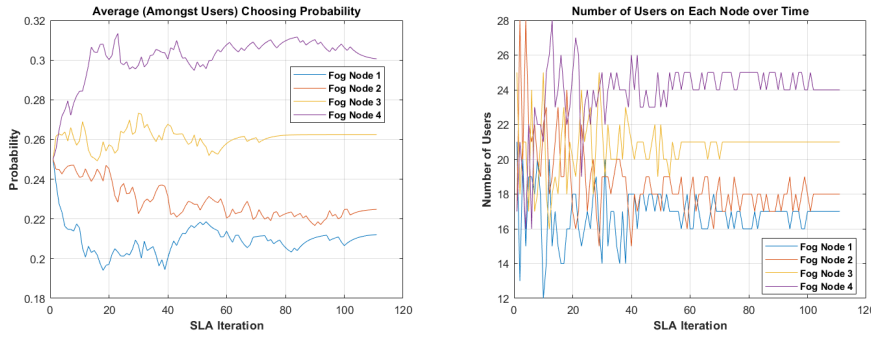
Σχήμα 5.2: Η εξέλιξη των πιθανοτήτων επιλογής ενός Χρήστη



Μπορούμε να δούμε πως εξελίσσονται οι πιθανότητες επιλογής, ξεκινώντας από την ομοιόμορφη κατανομή. Ο Χρήστης του οποίου τις πιθανότητες μελετούμε έχει συγκλίνει αρκετή ώρα πριν το τέλος του μηχανισμού, καθώς από την 60η επανάληψη έχει πιθανότητα επιλογής για τον Εξυπηρετητή 2 σχεδόν ίση με 1. Για τις τελευταίες, περίπου, 30 επαναλήψεις ο Χρήστης περιμένει μέχρι και όλοι οι υπόλοιποι να αποφασίσουν, ενώ αυτός είναι σίγουρος για την επιλογή του. Ένα γράφημα με παρόμοια μορφή ισχύει για όποιο άλλο Χρήστη επιλέξουμε να μελετήσουμε.

Το παραπάνω αποτέλεσμα μας δίνει μια αρκετά καλή εικόνα, για το πως συμπεριφέρονται οι Χρήστες ατομικά. Εξίσου σημαντικό, είναι να δούμε πως συμπεριφέρονται οι Χρήστες στο σύνολο τους και με ποιο τρόπο αντιδρούν οι Εξυπηρετητές. Στο Σχήμα 5.3 βλέπουμε συγκεντρωμένα μερικά στοιχεία. Με διαφορετικά χρώματα είναι σημειωμένοι οι τέσσερις εξυπηρετητές με αύξουσα, ως προς τα χαρακτηριστικά τους, σειρά (1- Μπλε, 2 - Πορτοκαλί, 3 - Κίτρινο, 4 - Μωβ).

Στο σχήμα 5.3α' φαίνεται η μέση (ως προς τους Χρήστες) πιθανότητα επιλογής κάθε Εξυπηρετητή υπολογιζόμενη ως $avgPr_k(t) = \frac{\sum_{i=1}^{|U|} Pr_{ik}(t)}{|U|}$ για κάθε γύρο t . Παρατηρούμε την τάση οι Χρήστες, όντως να προτιμούν την επιλογή του «καλύτερου», από άποψη φήμης και χαρακτηριστικών, Εξυπηρετητή, ενώ και η υπόλοιπη κατάταξη ακολουθεί αυτό το μοτίβο. Στο σχήμα 5.3β' βλέπουμε την κατανομή των Χρηστών στους Εξυπηρετητές ανά γύρο και το αποτέλεσμα είναι άμεσα συγγενικό με το προηγούμενο. Η μέση πιθανότητα επιλογής Εξυπηρετητή σε κάθε γύρο, όπως φαίνεται στο προηγούμενο σχήμα, είναι μια πολύ καλή εκτίμηση για την ποσοστιαία κατανομή του πληθυσμού που επιλέγει τον κάθε Εξυπηρετητή. Για παράδειγμα,



α': Μέσες Πιθανότητες Επιλογής β': Αριθμός Χρηστών ανά Εξυπηρετητή

Σχήμα 5.3: Αποτελέσματα Εξυπηρετητών

μια μέση πιθανότητα επιλογής $avgPr_k(t) = 0.25$ μας υποδεικνύει ότι στον Εξυπηρετητή k για το γύρο t κατά την τρέχουσα επανάληψη του Μηχανισμού Επιλογής υπάρχουν περίπου $|U| \times 0.25$ χρήστες. Αυτή η παρατήρηση, ισχύει με μεγάλη ακρίβεια στις τελευταίες επαναλήψεις του μηχανισμού επιλογής κατά τη σύγκλιση του, αφού αρκετοί Χρήστες παρουσιάζουν, πλέον, πιθανότητα περίπου ίση με 1.0 για επιλογή κάποιου συγκεκριμένου Εξυπηρετητή. Οπότε, σύμφωνα και με τη σχέση υπολογισμού μέσης πιθανότητας, αρκετοί από τους όρους που αθροίζουμε στον αριθμητή είναι ίσοι με 1 και το αποτέλεσμα είναι ακριβώς η πραγματική κατανομή του πληθυσμού. Ενδεικτικά, υπολογίζουμε τις τιμές κατά τη σύγκλιση του μηχανισμού:

- **Εξυπηρετητής 1:** Εκτίμηση = $0.21 \times 80 = 16.8$, Πραγματική τιμή = 17
- **Εξυπηρετητής 2:** Εκτίμηση = $0.23 \times 80 = 18.4$, Πραγματική τιμή = 18
- **Εξυπηρετητής 3:** Εκτίμηση = $0.26 \times 80 = 20.8$, Πραγματική τιμή = 21
- **Εξυπηρετητής 4:** Εκτίμηση = $0.3 \times 80 = 24$, Πραγματική τιμή = 24

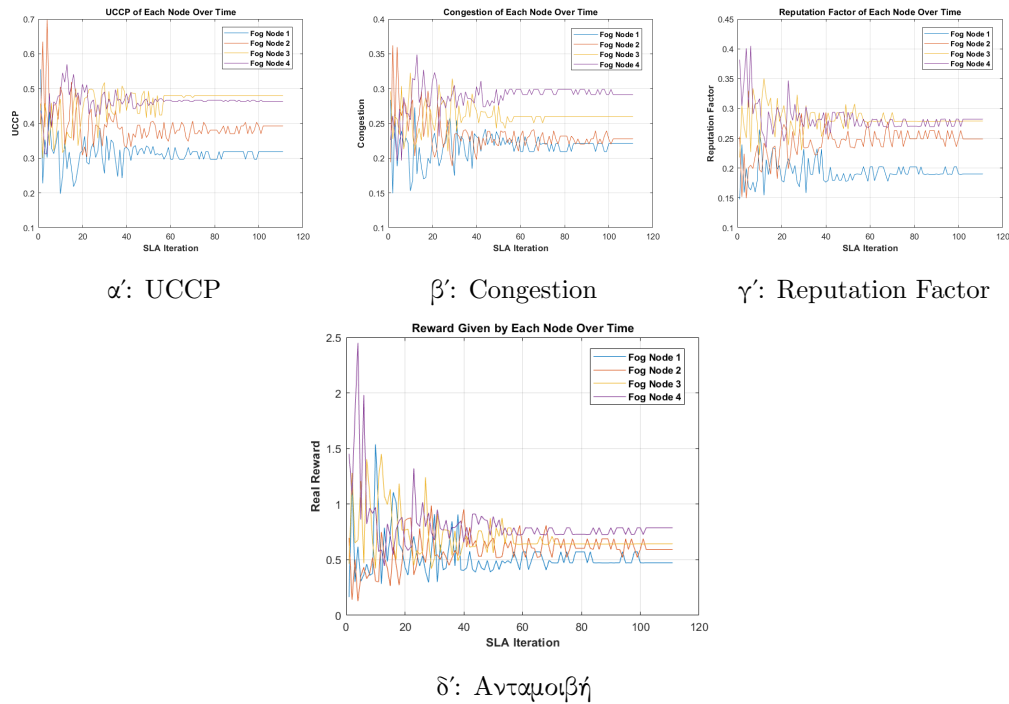
5.1.3 Χρήστες και Ανταμοιβή

Όπως έχουμε δει στο κεφάλαιο 4, όλοι οι Χρήστες, που βρίσκονται στον ίδιο Εξυπηρετητή λαμβάνουν την ίδια ανταμοιβή. Σε αυτή την παράγραφο θα δούμε αποτελέσματα για την ανταμοιβή που δίνει ο κάθε Εξυπηρετητής αλλά και για τα επί μέρους συστατικά της. Υπενθυμίζουμε σε αυτό το σημείο την σχέση υπολογισμού.

$$R_k^u = \frac{\mu_k \cdot RF_k}{Congestion_k \cdot UCPC_k}$$

Ο όρος μ_k , που αντιπροσωπεύει τη φήμη του κάθε Εξυπηρετητή, είναι σταθερός καθ' όλη τη διάρκεια του μηχανισμού επιλογής και δεν επηρεάζεται από τις επιλογές των Χρηστών. Στο Σχήμα 5.4 βλέπουμε τη διακύμανση των τιμών για τα υπόλοιπα συστατικά της ανταμοιβής, αλλά και της ίδιας της ανταμοιβής.

Σε συνέχεια των αποτελεσμάτων της παραγράφου 5.1.2 βλέπουμε τα συστατικά $Congestion$ και RF να ακολουθούν την ίδια σειρά με τα αποτελέσματα εκείνης της ενότητας. Αυτό είναι



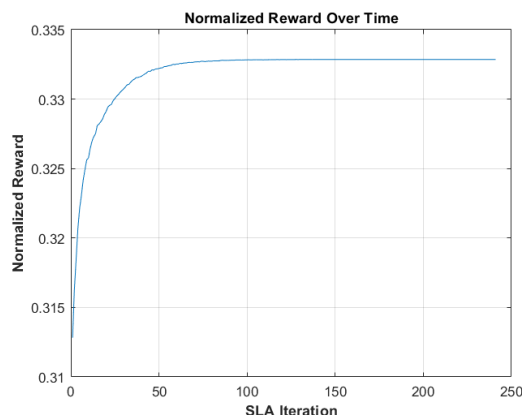
Σχήμα 5.4: Στοιχεία Ανταμοιβής ανά Εξυπηρετητή

ανεμενόμενο, καθώς, όπως είδαμε στις εξισώσεις της ενότητας 4.2.2, τα μεγέθη εξαρτώνται έμμεσα από τον αριθμό των Χρηστών που βρίσκονται σε κάθε Εξυπηρετητή. Οπότε, φυσιολογικά, ο μεγαλύτερος αριθμός Χρηστών στους Εξυπηρετητές 3 και 4 τους φέρνει και με τις μεγαλύτερες τιμές σε αυτά τα μεγέθη.

Σχετικά με το συστατικό *UCCP*, ακολουθεί και αυτό το μοτίβο των προηγούμενων μεγεθών με τη διαφορά πως η σειρά των δύο «καλύτερων» κόμβων είναι αντεστραμμένη. Όπως έχουμε αναφέρει, το μέγεθος εκφράζει ένα μέσο όρο των κανονικοποιημένων επιβαρύνσεων που έχουν οι Χρήστες. Οπότε, παρά την έμμεση εξάρτησή του από τον αριθμό των Χρηστών η επιρροή στην τιμή δεν είναι τόσο μεγάλη και ως εκ τούτου τείνουν να παρουσιάζονται κάποιες «ανωμαλίες» στην κατάταξή του.

Τα πιο πάνω δίνουν την ανταμοιβή που φαίνεται στο σχήμα 5.4δ'. Φυσιολογικά, οι Εξυπηρετητές που διαθέτουν τόσο την καλύτερη φήμη όσο και τα καλύτερα χαρακτηριστικά, δίνουν τη μεγαλύτερη ανταμοιβή στους Χρήστες τους. Η μεγαλύτερη ανταμοιβή είναι ακόμα ένα στοιχείο που ωθεί τους Χρήστες να επιλέξουν τους συγκεκριμένους Εξυπηρετητές. Όσο μεγαλύτερη είναι η πραγματική ανταμοιβή, τόσο μεγαλύτερη είναι και η κανονικοποιημένη ανταμοιβή και, συνεπώς, στην ενημέρωση των πιθανοτήτων οι Εξυπηρετητές με μεγάλη ανταμοιβή τείνουν να επιλέγονται συχνότερα.

Το τελευταίο αποτέλεσμα για το πρώτο επίπεδο αφορά την ανταμοιβή που παίρνουν οι Χρήστες. Στο Σχήμα 5.5 βλέπουμε την εξέλιξη της μέσης (ανάμεσα στους Χρήστες) κανονικοποιημένης ανταμοιβής. Το αποτέλεσμα δείχνει σταθερά αυξητική τάση μέχρι τη σύγκλιση και είναι πολύ σημαντικό, καθώς μας υποδεικνύει την τάση των παιχτών να λαμβάνουν αποφάσεις βέλτιστες για τους ίδιους. Επισημαίνουμε πως το σημείο σύγκλισης δεν είναι απαραίτητα



Σχήμα 5.5: Μέση Κανονικοποιημένη Ανταμοιβή

το βέλτιστο συνολικά για το σύστημα, αλλά αποτελεί σημείο Ισοροπίας Nash και είναι βέλτιστο για τον κάθε παίκτη ατομικά.

Σημείωση: Το τελευταίο αποτέλεσμα λήφθηκε με χρήση *Monte Carlo Smoothing* για 1000 επαναλήψεις, ώστε να εξαληφθεί όσο το δυνατό περισσότερο ο τυχαίος παράγοντας.

5.1.4 Δεύτερο Επίπεδο - Αξιολόγηση Δημοσκόπησης με RBTS

Για το δεύτερο επίπεδο, τη δημοσκόπηση για την ποιότητα του κόμβου, έχουμε απλά την υλοποίηση του Robust BTS, όπως περιγράφεται στο [WP12]. Για αυτό το λόγο, θα επιδείξουμε απλά με ένα παράδειγμα τη λειτουργία του μηχανισμού. Έστω ένας Εξυπηρετητής, ο οποίος εξυπηρετεί 10 Χρήστες. Οι απαντήσεις και οι αξιολογήσεις τους φαίνονται στον πίνακα 5.1.

Θα εφαρμόσουμε το μηχανισμό που περιγράψαμε για να πάρουμε τη συνολική απάντηση γι' αυτό τον Εξυπηρετητή. Έχουμε $avgYes = 1.187$ και $avgNo = 0.689$. Οπότε, σε αυτή την περίπτωση έχουμε νικήτρια άποψη το «ΝΑΙ» και η φήμη του Εξυπηρετητή ενημερώνεται βάσει αυτού.

Χρήστης	Προσωπική Απάντηση	Πρόβλεψη ΝΑΙ	Πρόβλεψη ΟΧΙ	Βαθμολογία Πληροφορίας	Βαθμολογία Πρόβλεψης	Βαθμολογία RBTS
1	0	50%	50%	0	0.75	0.75
2	0	1%	99%	0	0.1990	0.01990
3	1	27%	73%	1	0.46710	1.46710
4	1	79%	21%	1	0.95590	1.95590
5	1	67%	33%	0	0.55110	0.55110
6	1	63%	37%	0.66360	0.86310	1.52670
7	0	21%	79%	0.92160	0.37590	1.29750
8	1	86%	14%	1	0.98040	1.98040
9	1	72%	28%	0	0.48160	0.48160
10	1	81%	19%	0	0.34390	0.34390

Πίνακας 5.1: Αποτελέσματα Robust Bayesian Truth Serum (RBTS)

5.1.5 Τρίτο Επίπεδο - Εξέλιξη Φήμης

Στο τρίτο επίπεδο, την εξέλιξη της φήμης των Εξυπηρετητών βάσει του μοντέλου που αναλύσαμε, παρουσιάζουμε τις απαντήσεις των Χρηστών και πως αυτές επηρεάζουν τη φήμη του κάθε Εξυπηρετητή. Στο Σχήμα 5.6 φαίνεται η εξέλιξη των απαντήσεων και της φήμης για κάθε Εξυπηρετητή κατά τη διάρκεια των εκατό γύρων που εκτελέσαμε το μοντέλο.

Αφού όλοι οι Εξυπηρετητές ξεκινούν από την ίδια αρχική φήμη, $m_{0k} = 0.2$ και έχουν τις ίδιες πιθανότητες ικανοποίησης, $a_{H_k} = 0.51$ και $a_{L_k} = 0.49$, μια ίδια κατανομή θετικών και αρνητικών απαντήσεων θα τους αναθέσει και την ίδια φήμη. Όπως βλέπουμε στα αποτελέσματα μεγαλύτερος αριθμός θετικών απαντήσεων, δίνει ψηλότερη φήμη από την αρχική (Εξυπηρετητές 2,3 και 4), ενώ ο μεγαλύτερος αριθμός αρνητικών απαντήσεων δίνει χαμηλότερη φήμη από την αρχική (Εξυπηρετητής 2). Επισημαίνουμε, και πάλι, πως η φήμη των Εξυπηρετητών ακολουθεί την ποιότητα τους, όπως τη θέσαμε στο πρώτο βήμα του μοντέλου.

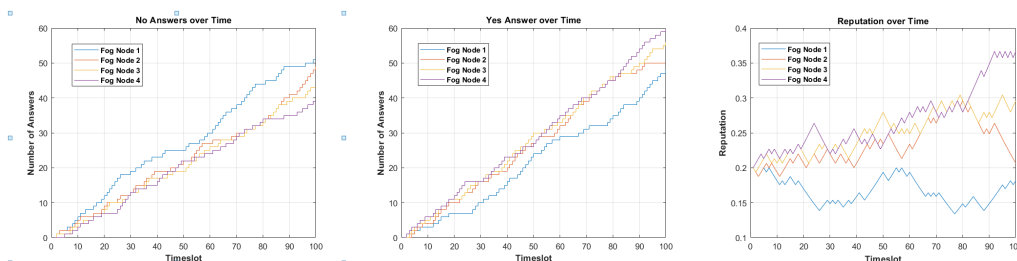
5.2 Συγκριτικά Αποτελέσματα

Σε αυτή την παράγραφο επιχειρούμε να δούμε εναλλακτικές επιλογές για τους μηχανισμούς που έχουμε καθορίσει στο μοντέλο, παίρνοντας κατάλληλες μετρήσεις όπου αυτό είναι δυνατό. και συγκρίνοντάς τις με τις επιλεγμένες μεθόδους.

5.2.1 SLA και άλλες Μεθόδοι Αντιστοίχισης

Στον αλγόριθμο που προτείνουμε μοντελοποιήσαμε τους Χρήστες σαν SLA, αφήνοντας τους να «μάνουν» μέσα από τις παρατηρήσεις τους το περιβάλλον τους και μέσω αυτής της γνώσης να επιλέξουν τον Εξυπηρετητή τους. Θα εξετάσουμε άλλες πέντε μεθόδους. Οι μετρήσεις για τις μεθόδους που εξετάζουμε πάρθηκαν με 80 Χρήστες και 4 Εξυπηρετητές με τυχαία χαρακτηριστικά:

- **Closest:** Ο κάθε Χρήστης επιλέγει τον κοντινότερο σε αυτόν Εξυπηρετητή.
- **Average Closest:** Υπολογίζεται για κάθε Εξυπηρετητή η μέση του απόσταση από όλους τους Χρήστες. Όλοι οι Χρήστες επιλέγουν τον Εξυπηρετητή με τη μικρότερη μέση τιμή απόστασης.



α': Αρνητικές Απαντήσεις

β': Θετικές Απαντήσεις

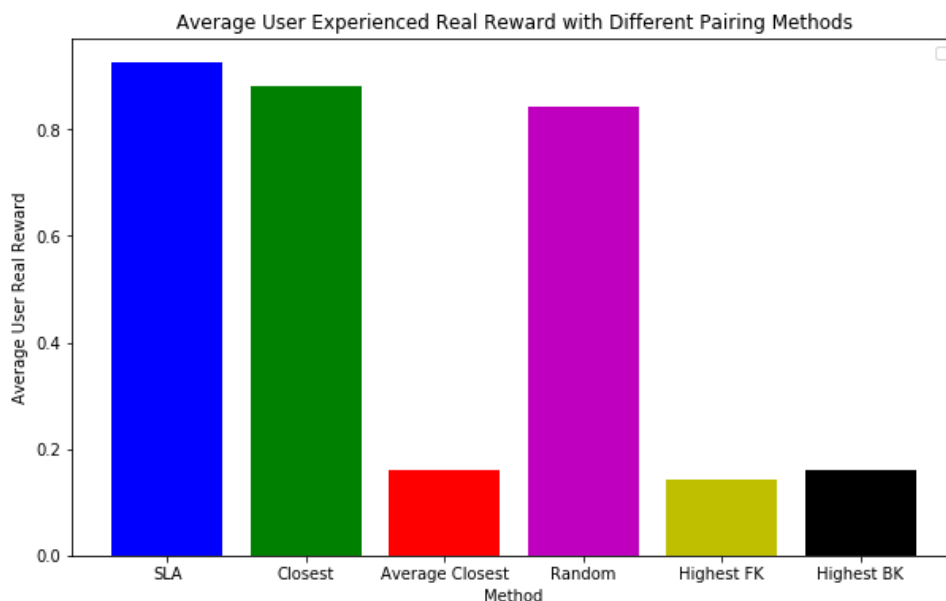
γ': Φήμη

Σχήμα 5.6: Συνολικά Αποτελέσματα Τρίτου Επιπέδου

- **Random:** Ο κάθε Χρήστης επιλέγει τυχαία έναν Εξυπηρετητή.
- **Highest FK:** Όλοι οι Χρήστες επιλέγουν τον Εξυπηρετητή με τη μεγαλύτερη υπολογιστική ισχύ.
- **Highest BK:** Όλοι οι Χρήστες επιλέγουν τον Εξυπηρετητή με το μεγαλύτερο αποθηκευτικό χώρο.

Τα αποτελέσματα της σύγκρισης φαίνονται στο σχήμα 5.7. Όπως μπορούμε να δούμε, ο μηχανισμός επιλογής με χρήση SLA είναι αυτός που δίνει και το καλύτερο αποτέλεσμα, αφού μέσω αυτού οι Χρήστες μαθαίνουν το περιβάλλον τους μέχρι να λάβουν την τελική απόφαση. Για την περίπτωση Closest βλέπουμε ένα αρκετά καλό αποτέλεσμα. Οι Χρήστες επιλέγοντας τον κοντινότερο γι' αυτούς Εξυπηρετητή κατανέμονται αρκετά ομοιόμορφα στους Εξυπηρετητές με αποτέλεσμα να μην παρουσιάζονται υψηλές τιμές Congestion πουθενά και σαν αποτέλεσμα να έχουμε ψηλές τιμές ανταμοιβής. Το ίδιο συμβαίνει και με τη μέθοδο Random, όπου η επιλογή γίνεται τυχαία.

Στις άλλες τρεις μεθόδους επιλογής (Average Closest, Highest FK, Highest BK) έχουμε το κοινό χαρακτηριστικό, πως όλοι οι Χρήστες επιλέγουν τον ίδιο Εξυπηρετητή για ανάθεση της εργασίας τους. Αυτό προκαλεί μεγάλα επίπεδα συμφόρησης στον Εξυπηρετητή που επωμίζεται το κόστος εξυπηρέτησης όλων των Χρηστών με αποτέλεσμα η ανταμοιβή να μένει σε πολύ χαμηλά επίπεδα. Αυτό ισχύει και σε περιπτώσεις που επιλέγουμε τον Εξυπηρετητή με τα «καλύτερα» χαρακτηριστικά όπως οι δύο τελευταίες.



Σχήμα 5.7: SLA και Άλλες Μέθοδοι Επιλογής

5.2.2 Αποτελέσματα Μηχανισμού BTS

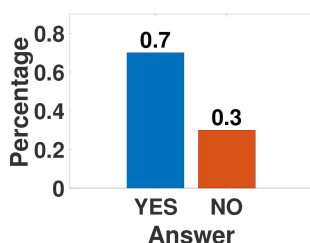
Σε αυτή την παράγραφο εξετάζουμε τη λειτουργία του μηχανισμού BTS η οποία μπορεί να αντιπαραβληθεί με τα αντίστοιχα αποτελέσματα για το μηχανισμό RBTS. Συγκεκριμένα, θα προσπαθήσουμε μέσω των αποτελεσμάτων να εξηγήσοουμε καλύτερα την έννοια του «παραδόξως κοινό» που αναφέραμε συνοπτικά στην παράγραφο 4.1.4.

Ετοιμάζουμε, αρχικά, τον πίνακα 5.8α' (θέτοντας για τον υπολογισμό της συνολικής βαθμολογίας, $\alpha = 1$), ο οποίος είναι παρόμοιος μορφής με τον πίνακα 5.1 και μάλιστα περιέχουν τις ίδιες απαντήσεις και προβλέψεις. Σύμφωνα με τα βήματα του μηχανισμού υπολογίζεται ο γεωμετρικός μέσος των προβλέσεων, όπως φαίνεται στο Σχήμα 5.8γ'. Η πραγματική κατανομή του πληθυσμού, με βάσει τις απαντήσεις φαίνεται στο Σχήμα 5.8β'. Διαπιστώνουμε πως η απάντηση για το «ΝΑΙ» έχει χαμηλότερη εκτίμηση από το πραγματικό (και άρα παραδόξως κοινή) και η απάντηση για το «ΟΧΙ» ελαφρώς ψηλότερη εκτίμηση από το πραγματικό (και άρα παραδόξως σπάνια). Ακόμη, διαπιστώνουμε πως οι συμμετέχοντες που απάντησαν «ΟΧΙ» είναι μάλιστα αρκετά ακραίοι στις προβλέψεις τους, προβλέποντας πως ο πληθυσμός συμφωνεί μαζί τους. Ως εκ τούτου, βλέπουμε τους υποστηρικτές του «ΟΧΙ» να λαμβάνουν εξαιρετικά χαμηλές βαθμολογίες.

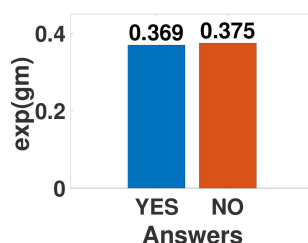
Με το υφιστάμενο δείγμα μπορούμε να δούμε γιατί ο μηχανισμός BTS δεν λειτουργεί για μικρά δείγματα. Με τέσσερις από τους υποστηρικτές του «ΝΑΙ» να ψηφίζουν «ΟΧΙ» τα αποτελέσματα αντιστρέφονται και έχουμε πλέον το «ΟΧΙ» σαν παραδόξως κοινή απάντηση.

Χρήστης	Προσωπική Απάντηση	Πρόβλεψη ΝΑΙ	Πρόβλεψη ΟΧΙ	Βαθμολογία Πληροφορίας	Βαθμολογία Πρόβλεψης	Βαθμολογία BTS
1	0	50%	50%	-0.22227	-0.08228	0.30456
2	0	1%	99%	-0.22227	-2.61577	-2.83804
3	1	27%	73%	0.63898	-0.40008	0.23890
4	1	79%	21%	0.63898	-0.02234	0.61665
5	1	67%	33%	0.63898	-0.00207	0.63691
6	1	63%	37%	0.63898	-0.01083	0.62815
7	0	21%	79%	-0.22227	-0.55231	-0.77458
8	1	86%	14%	0.63898	-0.08455	0.55444
9	1	72%	28%	0.63898	-0.00098	0.63800
10	1	81%	19%	0.63898	-0.03486	0.60412

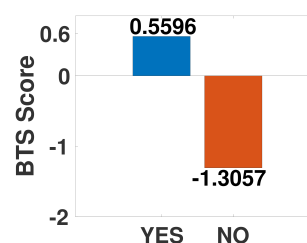
α': Αποτελέσματα Bayesian Truth Serum (BTS)



β': Κατανομή Απαντήσεων



γ': Γ.Μ. Προβλέψεων



δ': Μέση Βαθμολογία

Σχήμα 5.8: Στοιχεία BTS

5.3 Κλιμακωσιμότητα του Μηχανισμού Επιλογής

Τέλος, εξετάζουμε την απόδοση του μηχανισμού επιλογής ως προς τις τρεις παραμέτρους που επηρεάζουν την ταχύτητα σύγκλισής του. Τον αριθμό των Χρηστών, τον αριθμό των Εξυπηρετητών και την τιμή της παραμέτρου μάθησης, b . Εξετάζουμε τόσο την ταχύτητα σύγκλισης του μηχανισμού, όσο και την ανταμοιβή την οποία παίρνουν οι Χρήστες με διάφορες τιμές των παραμέτρων. Στα αποτελέσματα του σχήματος 5.9 η μπλε καμπύλη συμβολίζει την μέση ανταμοιβή ανάμεσα στους Χρήστες και η κόκκινη καμπύλη τον αριθμο των επαναλήψεων που εκτελεί ο μηχανισμός μέχρι να φτάσει στη σύγκλιση

Στο σχήμα 5.9α' βλέπουμε την απόδοση του μηχανισμού επιλογής για διάφορες τιμές της παραμέτρου μάθησης, b . Τόσο ο αριθμός, όσο και τα χαρακτηριστικά Χρηστών και Εξυπηρετητών παραμένουν σταθερά ($|U| = 80$ και $|N| = 4$ με τυχαία χαρακτηριστικά). Όπως μπορούμε να δούμε, μικρότερες τιμές της παραμέτρου δίνουν πιο αργή σύγκλιση. Το αποτέλεσμα είναι αναμενόμενο, αφού με χαμηλότερη παράμετρο οι Χρήστες μαθαίνουν πιο «αργά» και η σύγκλιση αργή. Σαν αποτέλεσμα της αργής μάθησης, ωστόσο, οι χρήστες μαθαίνουν καλύτερα και κάνουν τη βέλτιστη επιλογή, μεγιστοποιώντας την ανταμοιβή τους.

Στο σχήμα 5.9β' έχουμε την απόδοση του μηχανισμού συναρτήσει του αριθμού των Εξυπηρετητών που συμμετέχουν. Θέτουμε $b = 0.7$ και $|U| = 240$ με σταθερά χαρακτηριστικά. Διαπιστώνουμε, πως μεγαλύτερος αριθμός Εξυπηρετητών σημαίνει και αργότερη σύγκλιση, αφού οι Χρήστες σπαταλούν περισσότερο χρόνο μέχρι να αποφασίσουν για την επιλογή τους. Παράλληλα, με αύξηση του αριθμού Εξυπηρετητών βλέπουμε και αύξηση στις ανταμοιβές των Χρηστών. Αυτό συμβαίνει, διότι ο μεγαλύτερος αριθμός Εξυπηρετητών σημαίνει και μικρότερο αριθμό Χρηστών ανα Εξυπηρετητή, το οποίο επιφέρει πολύ χαμηλό *Congestion* και το οποίο, τελικά, ανεβάζει την τιμή της ανταμοιβής.

Τέλος, στο σχήμα 5.9γ' έχουμε την απόδοση του μηχανισμού με βάσει τον αριθμό των Χρηστών. Θέτουμε $b = 0.7$ και $|N| = 4$ με σταθερά χαρακτηριστικά. Όπως βλέπουμε, η αύξηση στον αριθμό των Χρηστών φέρνει πιο αργή σύγκλιση, καθώς περισσότεροι μεμονωμένοι Χρήστες πρέπει να συγκλίνουν. Ακόμη, ο μεγαλύτερος αριθμός Χρηστών φέρνει συμφόρηση στους Εξυπηρετητές μειώνοντας έτσι την ανταμοιβή που προσφέρουν.



α': Παράμετρος Beta

β': Αριθμός Εξυπηρετητών

γ': Αριθμός Χρηστών

Σχήμα 5.9: Κλιμακωσιμότητα Πρώτου Επιπέδου

6.1 Σύνοψη

Η τεχνολογία κάνει ραγδαία βήματα πρόοδου και γίνεται ολοένα και πιο ξεκάθαρο ότι αυτό δεν πρόκειται να αλλάξει σύντομα. Οι νέες τεχνολογίες, όπως η τεχνολογία 5G σε συνδυασμό με το ήδη υφιστάμενο πλαίσιο του Internet of Things θα συνεχίσουν να δημιουργούν μεγάλη κίνηση στα δίκτυα επικοινωνιών. Οι υπολογιστικές ανάγκες των χρηστών θα αυξηθούν και μαζί τους η ανάγκη για άμεση απόκριση στις απαιτήσεις τους για υπηρεσίες. Το Cloud Computing και το Edge Computing αποτελούν καταλυτικούς παράγοντες στην πραγματοποίηση νέων και σύγχρονων εφαρμογών για τις ανάγκες της καθημερινότητάς μας.

Στην παρούσα διπλωματική μελετήσαμε το πρόβλημα της Μεταφοράς Δεδομένων στα, συνεχώς εξελισσόμενα, μοντέλα της αρχιτεκτονικής Edge Computing. Αφού εισαγάγαμε τις νέες τεχνολογίες και αρχιτεκτονικές, προχωρήσαμε στον καθορισμό του προβλήματος και ακολούθως κάναμε μια ανασκόπηση της βιβλιογραφίας που υπήρχε. Προτείναμε ένα μοντέλο κατάλληλο για χρήση σε ένα περιβάλλον όπου πολλοί χρήστες θέλουν να εξυπηρετηθούν σε μια κοινή ομάδα servers και το αξιολογήσαμε, παρουσιάζοντας συνοπτικά τα αποτελέσματα και δίνοντας τις παρατηρήσεις μας.

6.2 Συμπεράσματα

Με τις μετρήσεις που πήραμε καταλήγουμε στο συμπέρασμα πως το μοντέλο λειτουργεί καλά. Ο κάθε Χρήστης ατομικά βελτιώνει την ανταμοιβή του μέχρι που ο ίδιος δεν μπορεί να την βελτιώσει άλλο, αλλάζοντας απλά Εξυπηρετητή. Ακολούθως, με την αξιολόγηση των Εξυπηρετητών από τους Χρήστες, οι Εξυπηρετητές αποχτούν φήμη ανάλογη των υπηρεσιών που προσφέρουν. Επιτυγχάνεται έτσι μια αρκετά δίκαιη κατανομή Χρηστών στους Εξυπηρετητές, κάτι που επιτρέπει την καλύτερη αξιοποίησή τους. Δυστυχώς, η σύγκλιση που πετυχαίνει η μοντελοποίηση των Χρηστών σαν SLA είναι σύγκλιση σε ισορροπία Nash και όχι βέλτιστη, οπότε η αξιοποίηση των Εξυπηρετητών έχει περιθώρια βελτίωσης.

6.3 Μελλοντική Εργασία

Ο σημαντικότερος περιορισμός που θέσαμε στον αλγόριθμό μας, ήταν η πολιτική Always Offload, την οποία ακολουθούσαν οι Χρήστες κατά τη Μεταφορά Δεδομένων στους Εξυπηρετητές. Μια ενδιαφέρουσα επέκταση θα ήταν να μελετούσαμε το πρόβλημα και με την πολιτική Partial Offloading, όπου εκτός από την επιλογή Εξυπηρετητή, οι Χρήστες έχουν να αποφασίσουν και το ποσοστό της εφαρμογής που θα στείλουν για επεξεργασία. Το πρόβλημα βελτιστοποίησης σε αυτή την περίπτωση θα είχε δύο επίπεδα.

Μια ακόμα πιθανή επέκταση θα ήταν η θεωρητική επίλυση του μοντέλου που παρουσιάσαμε και ο υπολογισμός κάποιας ισορροπίας Nash σε αυτό. Το μη-συνεργατικό παίγνιο μπορεί να εκφραστεί με παίχτες τους Χρήστες, στρατηγική του κάθε Χρήστη να είναι η επιλογή ανάμεσα στους εξυπηρετητές και ως συναρτήση ωφελείας να τειθεί η συνάρτηση ανταμοιβής.

Τέλος, μια ενδιαφέρουσα επέκταση θα μπορούσε να ήταν ο καθορισμός τιμών στους Εξυπηρετητές, τέτοιων ώστε να πετυχαίνεται η βέλτιστη λύση, δηλαδή η μετατροπή της βέλτιστης λύσης σε μια ισορροπία Nash. Η συγκεκριμένη επέκταση θα χρειαζόταν τη μετατροπή του παίγνιου που περιγράφηκε πιο πάνω σε ένα παίγνιο σύγκρισης με τους Εξυπηρετητές σαν τους μοιραζόμενους πόρους και της κατάλληλης επίλυσής του.

Παράρτημα Α

Θα δώσουμε εδώ την απόδειξη, πως η εξίσωση ενημέρωσης φήμης που δίνεται στην παράγραφο 4.1.6 είναι αύξουσα ως προς τον αριθμό των θετικών απαντήσεων, S και φθίνουσα προς τον αριθμό των αρνητικών απαντήσεων, F . Η συνάρτηση υπολογισμού φήμης δίνεται από τον τύπο:

$$\mu = \frac{\mu_0 a_H^S (1 - a_H)^F}{\mu_0 a_H^S (1 - a_H)^F + (1 - \mu_0) a_L^S (1 - a_L)^F}$$

Για λόγους εξοικονόμησης χώρου και πιο ευανάγνωστων υπολογισμών θέτουμε:

$$\Gamma = \mu_0 a_H^S (1 - a_H)^F \quad \Delta = (1 - \mu_0) a_L^S (1 - a_L)^F$$

Υπογραμμίζουμε σε αυτό το σημείο πως ισχύει $\Gamma, \Delta > 0$, καθώς $0 < \mu_0, a_H, a_L < 1$ και $S, F \geq 0$. Για να εξετάσουμε τη μονοτονία της συνάρτησης μ θα μελετήσουμε τις μερικές παραγώγους της μ ως προς τα S και F ξεχωριστά. Ξεκινώντας με τον αριθμό των θετικών απαντήσεων, S έχουμε:

$$\frac{\partial \mu}{\partial S} = \frac{\ln(a_H) \cdot \Gamma \cdot (\Gamma + \Delta) - (\ln(a_H) \cdot \Gamma - \ln(a_L) \cdot \Delta) \cdot \Gamma}{(\Gamma + \Delta)^2} = \frac{\Gamma \cdot \Delta \cdot \ln\left(\frac{a_H}{a_L}\right)}{(\Gamma + \Delta)^2}$$

Όπως έχουμε επισημάνει $\Gamma, \Delta > 0$. Οπότε, το πρόσημο της πιο πάνω παράστασης καθορίζεται από το πρόσημο της τιμής $\ln\left(\frac{a_H}{a_L}\right)$. Αφού $a_H > a_L$, έχουμε $\frac{a_H}{a_L} > 1$ και συνεπώς $\ln\left(\frac{a_H}{a_L}\right) > 0$. Συνεπώς, $\frac{\partial \mu}{\partial S} > 0$ και μ αύξουσα προς το S .

Με τον ίδιο τρόπο εργαζόμαστε και για τον αριθμό των αρνητικών απαντήσεων F . Γράφουμε:

$$\frac{\partial \mu}{\partial F} = \frac{\ln(1 - a_H) \cdot \Gamma \cdot (\Gamma + \Delta) - (\ln(1 - a_H) \cdot \Gamma - \ln(1 - a_L) \cdot \Delta) \cdot \Gamma}{(\Gamma + \Delta)^2} = \frac{\Gamma \cdot \Delta \cdot \ln\left(\frac{1 - a_H}{1 - a_L}\right)}{(\Gamma + \Delta)^2}$$

Και πάλι, όπως μπορούμε να δούμε, το πρόσημο της παράστασης καθορίζεται από το πρόσημο της τιμής $\ln\left(\frac{1 - a_H}{1 - a_L}\right)$. Αφού $a_H > a_L$, τότε $1 - a_H < 1 - a_L$ και $\frac{1 - a_H}{1 - a_L} < 1$. Συνεπώς, $\ln\left(\frac{1 - a_H}{1 - a_L}\right) < 0$ και άρα, $\frac{\partial \mu}{\partial F} < 0$ και μ φθίνουσα προς το F .

Βιβλιογραφία

- [19] *Internet of Things (IoT) History*. Nov. 2019. URL: <https://www.postscapes.com/iot-history/>.
- [AIM16] Luigi Atzori, Antonio Iera, and Giacomo Morabito. “Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm”. In: *Ad Hoc Networks* 56 (Dec. 2016). DOI: 10.1016/j.adhoc.2016.12.004.
- [Ash09] Kevin Ashton. *That 'Internet of Things' Thing*. June 2009. URL: <https://www.rfidjournal.com/that-internet-of-things-thing>.
- [ATP18] Pavlos Athanasios Apostolopoulos, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Game-theoretic learning-based QoS satisfaction in autonomous mobile edge computing”. In: *2018 Global Information Infrastructure and Networking Symposium (GIIS)*. IEEE. 2018, pp. 1–5.
- [ATP19] Pavlos Athanasios Apostolopoulos, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Risk-aware Social Cloud Computing based on Serverless Computing Model”. In: *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE. 2019, pp. 1–6.
- [ATP20α] Pavlos Athanasios Apostolopoulos, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Cognitive Data Offloading in Mobile Edge Computing for Internet of Things”. In: *IEEE Access* 8 (2020), pp. 55736–55749.
- [ATP20β] Pavlos Athanasios Apostolopoulos, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Risk-Aware Data Offloading in Multi-Server Multi-Access Edge Computing Environment”. In: *IEEE/ACM Transactions on Networking* (2020).
- [ATT19] Pavlos Athanasios Apostolopoulos, Marcos Torres, and Eirini Eleni Tsiropoulou. “Satisfaction-aware Data Offloading in Surveillance Systems”. In: *Proceedings of the 14th Workshop on Challenged Networks*. 2019, pp. 21–26.

- [Atz+12] Luigi Atzori et al. “The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization”. In: *Computer networks* 56.16 (2012), pp. 3594–3608.
- [Bec+14] Michael Beck et al. “Mobile Edge Computing: A Taxonomy”. In: Jan. 2014.
- [BM12] Flavio Bonomi and Rodolfo Milito. “Fog Computing and its Role in the Internet of Things”. In: *Proceedings of the MCC workshop on Mobile Cloud Computing* (Aug. 2012). DOI: 10.1145/2342509.2342513.
- [Cab05] Luis M. B. Cabral. “The Economics of Trust and Reputation : A Primer”. In: 2005.
- [Che+16] Ning Chen et al. “Dynamic urban surveillance video stream processing using fog computing”. In: *2016 IEEE second international conference on multimedia big data (BigMM)*. IEEE. 2016, pp. 105–112.
- [Cis20] Cisco. “Cisco Annual Internet Report (2018–2023)”. In: Jan. 2020. URL: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf>.
- [Daw89] Robyn Dawes. In: *Journal of Experimental Social Psychology* 25.1 (1989).
- [DD17] K. Dolui and S. K. Datta. “Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing”. In: *2017 Global Internet of Things Summit (GIoTS)*. 2017, pp. 1–6.
- [Far+18] Haleem Farman et al. “Multi-criteria based zone head selection in Internet of Things based wireless sensor networks”. In: *Future Generation Computer Systems* 87 (2018), pp. 364–371.
- [FK17] Mohammad Fouladlou and Ahmad Khademzadeh. “An energy efficient clustering algorithm for Wireless Sensor devices in Internet of Things”. In: *2017 Artificial Intelligence and Robotics (IRANOPEN)*. IEEE. 2017, pp. 39–44.
- [Fra+19] Georgios Fragkos et al. “Artificial Intelligence Empowered UAVs Data Offloading in Mobile Edge Computing”. In: *IEEE International Conference on Communications (ICC)* (2019), pp. 1–6.
- [FTP19] Michail Fasoulakis, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Satisfy instead of maximize: Improving operation efficiency in wireless communication networks”. In: *Computer Networks* 159 (2019), pp. 135–146.
- [FTP20] Georgios Fragkos, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Artificial Intelligence Enabled Distributed Edge Computing for Internet of Things Applications”. In: *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE Computer Society. 2020, pp. 450–457.
- [Goo+16] Mathew Goonewardena et al. “Generalized satisfaction equilibrium: A model for service-level provisioning in networks”. In: *European Wireless 2016; 22th European Wireless Conference*. VDE. 2016, pp. 1–5.

- [GR07] Tilmann Gneiting and Adrian E Raftery. “Strictly Proper Scoring Rules, Prediction, and Estimation”. In: *Journal of the American Statistical Association* 102.477 (2007), pp. 359–378. DOI: 10.1198/016214506000001437.
- [Heu+17] Niklas Heuveloop et al. “Ericsson mobility report”. In: *Ericsson, Stockholm* (2017).
- [Kha+19] Wazir Khan et al. “Edge computing: A survey”. In: *Future Generation Computer Systems* 97 (Feb. 2019). DOI: 10.1016/j.future.2019.02.050.
- [KL51] S. Kullback and R. A. Leibler. “On Information and Sufficiency”. In: *Ann. Math. Statist.* 22.1 (Mar. 1951), pp. 79–86. DOI: 10.1214/aoms/1177729694.
- [KP17] Anna Karlin and Yuval Peres. “Truthful auctions in win/lose settings”. In: *Game Theory, Alive*. American Mathematical Society, Apr. 2017, pp. 267–284. ISBN: 9781470419820.
- [Lew17] Michał Lewandowski. “Prospect theory versus expected utility theory: Assumptions, predictions, intuition and modelling of risk attitudes”. In: *Central European Journal of Economic Modelling and Econometrics* (2017), pp. 275–321.
- [Mao+17] Yuyi Mao et al. “A survey on mobile edge computing: The communication perspective”. In: *IEEE Commun, Surveys & Tutorials* 19.4 (2017), pp. 2322–2358.
- [Mér+12] François Mériaux et al. “Achievability of efficient satisfaction equilibria in self-configuring networks”. In: *International Conference on Game Theory for Networks*. Springer. 2012, pp. 1–15.
- [Mes+17] Mohamed-Ayoub Messous et al. “Computation offloading game for an UAV network in mobile edge computing”. In: *2017 IEEE International Conference on Communications (ICC)*. IEEE. 2017, pp. 1–6.
- [Mes+19] Mohamed-Ayoub Messous et al. “A game theory based efficient computation offloading in an uav network”. In: *IEEE Transactions on Vehicular Technology* 68.5 (2019), pp. 4964–4974.
- [MG11] Peter M. Mell and T. Grance. “The NIST Definition of Cloud Computing”. In: 2011.
- [Mit+19] Giorgos Mitsis et al. “Intelligent dynamic data offloading in a competitive mobile edge computing market”. In: *Future Internet* 11.5 (2019), p. 118.
- [MPV14] Olga Munoz, Antonio Pascual-Iserte, and Josep Vidal. “Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading”. In: *IEEE Transactions on Vehicular Technology* 64.10 (2014), pp. 4738–4755.

- [MRZ05] Nolan Miller, Paul Resnick, and Richard Zeckhauser. “Eliciting Informative Feedback: The Peer-Prediction Method”. In: *Management Science* 51.9 (2005), pp. 1359–1373. DOI: 10.1287/mnsc.1050.0379. URL: <https://doi.org/10.1287/mnsc.1050.0379>.
- [MTP20] Giorgos Mitsis, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Data Offloading in UAV-Assisted Multi-Access Edge Computing Systems: A Resource-Based Pricing and User Risk-Awareness Approach”. In: *Sensors* 20 (Apr. 2020), p. 2434. DOI: 10.3390/s20082434.
- [Net17] Cisco Visual Networking. “Cisco Global Cloud Index: forecast and methodology, 2015-2020”. In: *White paper by Cisco* (2017).
- [Pre04] Drazen Prelec. “A Bayesian Truth Serum for Subjective Data”. In: *Science (New York, N.Y.)* 306 (Nov. 2004), pp. 462–6. DOI: 10.1126/science.1102081.
- [Pro+19] Panagiotis Promponas et al. “Redesigning Resource Management in Wireless Networks based on Games in Satisfaction Form”. In: *2019 12th IFIP Wireless and Mobile Networking Conference (WMNC)*. IEEE. 2019, pp. 24–31.
- [RF13] Goran Radanovic and Boi Faltings. “A Robust Bayesian Truth Serum for Non-binary Signals”. In: Jan. 2013, pp. 833–839.
- [RMH17] Shermila Ranadheera, Setareh Maghsudi, and Ekram Hossain. “Minority games with applications to distributed decision making and control in wireless networks”. In: *IEEE Wireless Communications* 24.5 (2017), pp. 184–192.
- [Ryu+15] Minwoo Ryu et al. “Design and implementation of a connected farm for smart farming system”. In: *2015 IEEE SENSORS*. IEEE. 2015, pp. 1–4.
- [Sat+09] M. Satyanarayanan et al. “The Case for VM-Based Cloudlets in Mobile Computing”. In: *IEEE Pervasive Computing* 8.4 (2009), pp. 14–23.
- [SBE01] Sanjay Sarma, David Brock, and Daniel Engels. “Radio frequency identification and the electronic product code”. In: *IEEE micro* 21.6 (2001), pp. 50–54.
- [SPT94] P. S. Sastry, V. V. Phansalkar, and M. A. L. Thathachar. “Decentralized learning of Nash equilibria in multi-person stochastic games with incomplete information”. In: *IEEE Transactions on Systems, Man, and Cybernetics* 24.5 (1994), pp. 769–777.
- [SS14] S. B. Shaw and A. K. Singh. “A survey on cloud computing”. In: *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*. 2014, pp. 1–6.
- [TKP18] E Tsiropoulou, K Koukas, and S Papavassiliou. “A socio-physical and mobility-aware coalition formation mechanism in public safety networks”. In: *EAI Endorsed Trans. Future Internet* 4 (2018), p. 154176.

- [TL16] Utku Tefek and Teng Joon Lim. “Clustering and radio resource partitioning for machine-type communications in cellular networks”. In: *2016 IEEE Wireless Communications and Networking Conference*. IEEE. 2016, pp. 1–6.
- [TMP18] Eirini Eleni Tsiropoulou, Giorgos Mitsis, and Symeon Papavassiliou. “Interest-aware energy collection & resource management in machine to machine communications”. In: *Ad Hoc Networks* 68 (2018), pp. 48–57.
- [TPB17] Eirini Eleni Tsiropoulou, Surya Teja Paruchuri, and John S Baras. “Interest, energy and physical-aware coalition formation and resource allocation in smart IoT applications”. In: *2017 51st Annual Conference on Information Sciences and Systems (CISS)*. IEEE. 2017, pp. 1–6.
- [Tsi+17] Eirini Eleni Tsiropoulou et al. “RFID-based smart parking management system”. In: *Cyber-Physical Systems* 3.1-4 (2017), pp. 22–41.
- [TTP19] Athina Thanou, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Quality of experience under a prospect theoretic perspective: A cultural heritage space use case”. In: *IEEE Transactions on Computational Social Systems* 6.1 (2019), pp. 135–148.
- [TTP20] Athina Thanou, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “A Sociotechnical Approach to the Museum Congestion Management Problem”. In: *IEEE Transactions on Computational Social Systems* 7.2 (2020), pp. 563–568.
- [Tve00] Amos Tversky. “Prospect Theory: An Analysis of Decision under Risk”. In: Sept. 2000, pp. 17–43. ISBN: 9780511803475. DOI: 10.1017/CB09780511803475.003.
- [TVP13] Eirini Eleni Tsiropoulou, Panagiotis Vamvakas, and Symeon Papavassiliou. “Joint utility-based uplink power and rate allocation in wireless networks: A non-cooperative game theoretic framework”. In: *Physical Communication* 9 (2013), pp. 299–307.
- [USZ16] Kaleem Ullah, Munam Ali Shah, and Sijing Zhang. “Effective ways to use Internet of Things in the field of medical and smart health care”. In: *2016 International Conference on Intelligent Systems Engineering (ICISE)*. IEEE. 2016, pp. 372–379.
- [VTP19α] Panagiotis Vamvakas, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Exploiting prospect theory and risk-awareness to protect UAV-assisted network operation”. In: *EURASIP Journal on Wireless Communications and Networking* 2019.1 (2019), p. 286.
- [VTP19β] Panagiotis Vamvakas, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “On the prospect of uav-assisted communications paradigm in public safety networks”. In: *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE. 2019, pp. 762–767.

- [VTP19 ζ] Panagiotis Vamvakas, Eirini Eleni Tsiropoulou, and Symeon Papavassiliou. “Risk-aware resource control with flexible 5G access technology interfaces”. In: *2019 IEEE 20th International Symposium on “A World of Wireless, Mobile and Multimedia Networks”(WoWMoM)*. IEEE. 2019, pp. 1–9.
- [WP12] Jens Witkowski and David Parkes. “A Robust Bayesian Truth Serum for Small Populations (Technical Report)”. In: 2 (May 2012).
- [Xia+14] Liang Xiao et al. “Prospect theoretic analysis of anti-jamming communications in cognitive radio networks”. In: *2014 IEEE Global Communications Conference*. IEEE. 2014, pp. 746–751.
- [Xia+15] Wenfeng Xia et al. “A Survey on Software-Defined Networking”. In: *Communications Surveys Tutorials, IEEE* 17 (Jan. 2015), pp. 27–51. DOI: 10.1109/COMST.2014.2330903.
- [XWW15] Yuhua Xu, Jinlong Wang, and Qihui Wu. “Distributed Learning of Equilibria with Incomplete, Dynamic, and Uncertain Information in Wireless Communication Networks”. In: Jan. 2015, pp. 63–86. DOI: 10.4018/978-1-4666-8642-7.ch003.
- [Yu+18] Wei Yu et al. “A Survey on the Edge Computing for the Internet of Things”. In: *IEEE Access* 6 (2018), pp. 6900–6919.
- [ZG11] Yan Zhang and Mohsen Guizani. *Game theory for wireless communications and networking*. CRC press, 2011.
- [Zha+16] Ke Zhang et al. “Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks”. In: *IEEE access* 4 (2016), pp. 5896–5907.
- [Zha+17] Ke Zhang et al. “Optimal delay constrained offloading for vehicular edge computing networks”. In: *2017 IEEE International Conference on Communications (ICC)*. IEEE. 2017, pp. 1–6.

Απόδοση

Διαδίκτυο των Πραγμάτων
Υπολογιστικό Νέφος
Φυλλομετρητής
Επαυξημένη Πραγματικότητα
Παίγνιο Μειονότητας
Μη-Συνεργατικό Παίγνιο
Θεωρία Προοπτικής
Φιλαλήθης
Κατά Bayes Ενημέρωση
Αυστηρά Κατάλληλοι Κανόνες Βαθμολόγησης
Πίστη κατά Bayes
Χωρητικότητα Καναλιού
Κέρδος Καναλιού
Επιβάρυνση
Άκρη του Δικτύου
Εφαρμογές Πραγματικού Χρόνου
Υπολογιστική Άκρη

Ξενόγλωσσος όρος

Internet of Things
Cloud Computing
Web Browser
Augmented Reality
Minority Game
Non-Cooperative Game
Prospect Theory
Truthful
Bayesian Updating
Strictly Proper Scoring Rules
Bayesian Belief
Bandwidth
Channel Gain
Overhead
Network Edge
Real-Time Applications
Edge Computing

