



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΙΩΑΝΝΙΝΩΝ

**ΣΧΟΛΗ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΠΜΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ**

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΣΥΓΚΡΙΤΙΚΗ ΜΕΛΕΤΗ ΜΟΝΤΕΛΩΝ ΚΟΣΤΟΥΣ ΣΕ
ΑΡΧΙΤΕΚΤΟΝΙΚΕΣ FOG COMPUTING**

Δόση Νικολέτα

Επιβλέπουσα: Μαργαρίτη Σπυριδούλα,
ΕΔΙΠ, Α

Άρτα, Δεκέμβριος, 2021

**COMPARING THE COST OF MODELS IN ARHITEKTONIC
OF FOG COMPUTING**

Εγκρίθηκε από τριμελή εξεταστική επιτροπή
Άρτα, Ημερομηνία

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1. Επιβλέπουσα καθηγήτρια

Σπυριδούλα Μαργαρίτη,

ΕΔΙΠ, Α

2. Μέλος επιτροπής

Ελευθέριος Στεργίου,

ΔΕΠ, Αναπληρωτής Καθηγητής

3. Μέλος επιτροπής

Χρυσόστομος Στύλιος

ΔΕΠ, Αναπληρωτής Καθηγητής

© Δόση, Νικολέτα, 2021.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Δήλωση μη λογοκλοπής

Δηλώνω υπεύθυνα και γνωρίζοντας τις κυρώσεις του Ν. 2121/1993 περί Πνευματικής Ιδιοκτησίας, ότι η παρούσα μεταπτυχιακή εργασία είναι εκ ολοκλήρου αποτέλεσμα δικής μου ερευνητικής εργασίας, δεν αποτελεί προϊόν αντιγραφής ούτε προέρχεται από ανάθεση σε τρίτους. Όλες οι πηγές που χρησιμοποιήθηκαν (κάθε είδους, μορφής και προέλευσης) για τη συγγραφή της περιλαμβάνονται στη βιβλιογραφία.

Δόση Νικολέτα

Υπογραφή

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα διπλωματική εκπονήθηκε στο τμήμα Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Ιωαννίνων για το μεταπτυχιακό πρόγραμμα Πληροφορικής και Δικτύων. Για την εκπόνησή αυτής θα ήθελα να ευχαριστήσω θερμά τον επιβλέπων καθηγητή κ. Ελευθέριο Στεργίου καθώς επίσης την καθηγήτρια κ. Σπυριδούλα Μαργαρίτη για τη βοήθεια κατά την διάρκεια εκπόνησης της διπλωματικής μου εργασίας. Τέλος, ευχαριστώ τον σύζυγό μου, τα τρία μου παιδιά και τους γονείς μου για την στήριξή τους σε όλα τα έτη των σπουδών μου.

ΠΕΡΙΛΗΨΗ

Το Fog Computing (FC) είναι μια σχετικά πρόσφατη αρχιτεκτονική κατάλληλη για καταναμημένο υπολογισμό. Λειτουργεί ως ενδιάμεσο επίπεδο μεταξύ της υποδομής Cloud και των IoT συσκευών για να παρέχει ένα σύνολο υπηρεσιών στους χρήστες όπως υπολογισμούς, δικτύωση, αποθήκευση. Η απόδοση της αρχιτεκτονικής Fog Computing συνδέεται άμεσα με τον τρόπο ανάπτυξης, την ποιότητα υπηρεσιών και το κόστος από την κατανάλωση των πόρων. Η εκτίμηση του κόστους είναι μια πρόκληση αλλά και συνάμα μια σύνθετη διαδικασία λόγω της ετερογένειας, της κλίμακας και της δυναμικότητας των χρησιμοποιούμενων υποδομών. Σήμερα, υπάρχουν αρκετές εργασίες που επιχειρούν να εκτιμήσουν το κόστος σε Fog Computing αρχιτεκτονικές και προτείνουν αντίστοιχα μοντέλα.

Στη συγκεκριμένη εργασία θα προσδιορίσουμε την έννοια του κόστους στο FC, θα εξετάσουμε και θα συγκρίνουμε διάφορες έρευνες και μοντέλα προσέγγισης του κόστους στα δίκτυα FC. Συγκεκριμένα θα εστιάσουμε στο κόστος υποδομής του δικτύου, στο κόστος επικοινωνίας, στο κόστος υπολογισμού και στο κόστος ασφάλειας. Σκοπός της παρούσας εργασίας είναι: α) να παρουσιάσει συγκριτικά διερευνήσει τα προτεινόμενα μοντέλα κόστους, β) να ανακαλύψει/προσδιορίσει τις παραμέτρους που επηρεάζουν το κόστος και να εντοπίσει τις κοινές παραμέτρους σε όλα τα μοντέλα που οδηγούν στην ελαχιστοποίηση του κόστους και γ) να εξετάσει την ενοποίησή τους για την παροχή μιας συνολικής λύσης υπολογισμού κόστους σε τέτοια περιβάλλοντα.

Με τη βοήθεια του προσομοιωτή FogComputingSim, θα μετρηθεί πειραματικά το κόστος ανάπτυξης με βάση τη συνολική συνάρτηση κόστους, και θα παρουσιασθούν συγκριτικά οι πηγές που συμβάλουν στη διαμόρφωσή του.

Λέξεις κλειδιά: Μετεγκατάσταση, καθυστέρηση, χρήση δικτύου, ποιότητα υπηρεσιών.

ABSTRACT

Fog Computing (FC) is a relatively recent architecture suitable for distributed computing. It acts as an intermediate level between the Cloud infrastructure and IoT devices to provide a set of services to users such as computing, networking, storage. The performance of the Fog Computing architecture is directly related to the way it is developed, the quality of services and the cost of resource consumption. Cost estimation is a challenge but also a complex process due to the heterogeneity, scale and capacity of the infrastructure used. Today, there are several works that attempt to estimate the cost of Fog Computing architectures and suggest corresponding models.

In this paper we will define the concept of cost in FC, examine and compare various surveys and cost approach models in FC networks. Specifically, we will focus on network infrastructure costs, communication costs, calculation costs and security costs. The purpose of this work is: a) to present comparatively investigate the proposed cost models, b) to discover / determine the parameters that affect costs and to identify the common parameters in all models that lead to cost minimization and c) to examine their integration to provide a comprehensive costing solution in such environments.

With the help of the FogComputingSim simulator, development costs will be experimentally measured based on the total cost function. and the sources that contribute to its formation will be presented comparatively.

Keywords: Fog Computing, offloading, migration, latency, network usage, QoS

ΠΕΡΙΕΧΟΜΕΝΑ

1	Εισαγωγή	11
1.1	Η έννοια του κόστους στο FC.....	14
1.2	Το πρόβλημα.....	15
1.3	Το αντικείμενο της εργασίας.....	15
1.4	Δομή της εργασίας.....	16
2	Κόστος υποδομής	18
2.1	CAPEX.....	18
2.2	OPEX.....	19
2.3	Προσεγγίσεις υπολογισμού κόστους.....	20
2.3.1	Γενικό μοντέλο Υπολογισμού του OPEX.....	20
2.3.2	Μοντέλο cost-aware.....	20
2.3.3	Πολυεπίπεδα μοντέλα.....	23
2.3.4	Κόστος ενέργειας.....	24
2.4	Συμπεράσματα.....	26
3	Κόστος Επικοινωνίας	29
3.1	Η έννοια του κόστους επικοινωνίας.....	30
3.2	Διασύνδεση και τεχνολογίες δικτύωσης.....	30
3.3	Προσεγγίσεις υπολογισμού κόστους.....	32
3.3.1	Μοντέλο κόστους δικτυακής κίνησης (IP traffic).....	32
3.3.2	Πολυεπίπεδο μοντέλο κόστους.....	36
3.3.3	Αρχιτεκτονική τριών επιπέδων.....	37
3.3.4	Συσχέτιση απόδοσης και κόστους.....	37
3.4	Συμπεράσματα.....	38
4	Κόστος υπολογισμού (computation cost)	40
4.1	Κόστος επεξεργασίας.....	40
4.1.1	Εκχώρηση πόρων.....	40
4.1.2	Προγραμματισμός εργασιών.....	41
4.1.3	Κόστος Migration.....	44
4.1.4	Κόστος offloading.....	44
4.1.5	Μοντέλο υπολογισμού κόστους σε κινητές συσκευές.....	45
4.2	Ασφάλεια.....	45
4.2.1	Πιστοποίηση Mutual Authentication.....	48

4.3	Συμπεράσματα	48
5	Υλοποίηση προσομοίωσης FogComputingSim	51
5.1	Εισαγωγή.....	51
5.2	Η αρχιτεκτονική FogComputingSim	52
5.2.1	Τα χαρακτηριστικά της προσομοίωσης FogComputingSim	52
5.2.2	Τα συστατικά της προσομοίωσης FogComputingSim.....	53
5.2.3	Οι υπηρεσίες	53
5.2.4	Σχεδιασμός και implementation	54
5.2.5	Δημιουργία προσομοίωσης στο FogComputingSim	56
5.3	Εφαρμογές σε περιβάλλον Fog	58
5.3.1	Το Electroencephalogram (EEG) Tractor Beam Game.....	59
5.3.2	Κατανεμημένο σύστημα από κάμερες.....	59
5.4	Σενάρια προσομοίωσης.....	60
5.4.1	Τιμές παραμέτρων	61
5.4.2	Αξιολόγηση της απόδοσης.....	62
5.4.2.1	Σενάριο 1	63
5.4.2.2	Σενάριο 2.....	63
5.4.2.3	Σενάριο 3.....	64
5.4.2.4	Σενάριο 4.....	65
5.4.2.5	Σενάριο 5.....	66
5.4.2.6	Σενάριο 6.....	67
5.4.2.7	Σενάριο 7.....	68
5.4.2.8	Σενάριο 8.....	69
5.4.2.9	Σενάριο 9.....	70
5.4.2.10	Σενάριο 10.....	71
5.4.2.11	Σενάριο 11	72
5.4.2.12	Σενάριο 12.....	73
5.4.2.13	Σενάριο 13.....	74
5.5	Ανάλυση αποδόσεων	75
5.6	Τιμές κόστους ανά αλγόριθμο.....	79
5.7	Συμπεράσματα	81
6	Συμπεράσματα.....	83

1 Εισαγωγή

Το διαδίκτυο των πραγμάτων (Internet-of-Things - IoT), δηλαδή ένα σύνολο διασυνδεδεμένων αντικειμένων που επικοινωνούν μεταξύ τους δημιουργώντας ένα δίκτυο, αποτελεί μια πτυχή της καθημερινότητας του ανθρώπου, που συμβάλει στην παραγωγή τεράστιου όγκου δεδομένων. Τα δεδομένα συλλέγονται ή δημιουργούνται από τα αντικείμενα και για επεξεργασία, αποθήκευση και διαχείριση χρησιμοποιείται η αρχιτεκτονική Cloud. Το Cloud Computing αποτελεί μια λύση, που έχει υιοθετηθεί ευρέως για την παροχή υπηρεσιών IoT, χάρη στην σχεδόν απεριόριστη ισχύ επεξεργασίας και χώρο αποθήκευσης. Ωστόσο, η αποκλειστική χρήση αυτού του μοντέλου δεν είναι εφικτή σε μεγάλη κλίμακα, καθώς η μετάδοση τεράστιου όγκου δεδομένων IoT στο Cloud δημιουργεί σοβαρά προβλήματα στο διαθέσιμο εύρος ζώνης, όπως σε θέματα που σχετίζονται με τον δυνητικά ανεξέλεγκτο χρόνο καθυστερήσεων, την εκρηκτικά αυξημένη κατανάλωση ενέργειας και την υποβάθμιση των επιδόσεων των υπολογιστικών κόμβων λόγω της αργής μετάδοσης [1]. Τα ζητήματα αυτά εγείρουν την ανάγκη για την αναζήτηση άλλων υπολογιστικών μοντέλων που θα απαλείφουν ή θα μετριάζουν αυτά τα προβλήματα.

Μια τέτοια λύση αποτελεί το Fog Computing (FC) που αναπτύχθηκε ως μια νέα αρχιτεκτονική υπολογισμών που χειρίζεται εφαρμογές ευαίσθητες στην καθυστέρηση, όπου η επεξεργασία, η αποθήκευση, ο έλεγχος και η ανάλυση των δεδομένων λαμβάνουν χώρα κοντά στην πηγή, μειώνοντας την επιβάρυνση στην επικοινωνία και την καθυστέρηση στην παροχή των διαφόρων υπηρεσιών.

Όμως, εξαιτίας της ετερογένειας των συσκευών στο δίκτυο FC, είναι σημαντικό να εφαρμόζονται νέες λύσεις και τρόποι βελτίωσης που θα λαμβάνουν υπόψη την ποικιλομορφία των διαθέσιμων πόρων και τις αυξανόμενες απαιτήσεις των IoT συσκευών για την αποτελεσματική, βελτιωμένη και δυναμική διανομή της επεξεργασίας [1]. Επιπλέον, καθίσταται αναγκαία και υψίστης σημασίας η έξυπνη διαχείριση των πόρων, η αυξανόμενη χρήση εύρους ζώνης, η μείωση στην κατανάλωση ενέργειας και του λανθάνοντα χρόνου από τις συσκευές IoT ώστε να μην οδηγούμαστε σε σφάλματα και αστοχίες επεξεργασίας και απόδοσης που μπορεί να είναι δύσκολο να αντιμετωπιστούν.

Το FC είναι μια σχετικά πρόσφατη αρχιτεκτονική κατάλληλη για κατακευματισμένο υπολογισμό, που μπορεί να αντιμετωπίσει τα παραπάνω προβλήματα. Λειτουργεί ως ενδιάμεσο επίπεδο μεταξύ της υποδομής Cloud και των συσκευών IoT για να παρέχει ένα σύνολο υπηρεσιών στους χρήστες όπως υπολογισμούς, δικτύωση, αποθήκευση [2]. Το FC βρίσκει σήμερα πολλές εφαρμογές όπως στα έξυπνα δίκτυα και έξυπνες πόλεις, συνδεδεμένα οχήματα και αναλύσεις σε πραγματικό χρόνο [3]. Οι εφαρμογές αυτές βασίζονται στην επικοινωνία ενός τεράστιου αριθμού διαφορετικών οντοτήτων. Οι λειτουργίες διαχείρισης δεδομένων τοποθετούνται κοντά στην προέλευσή τους, με αποτέλεσμα την καλύτερη εκμετάλλευση των άκρων του δικτύου. Στο FC η επεξεργασία και η ανάλυση δεδομένων

μπορούν να πραγματοποιηθούν με μειωμένη καθυστέρηση υπηρεσίας και βελτίωση στην ποιότητα υπηρεσιών σε σύγκριση με την απομακρυσμένη χρήση Cloud. Οι τεχνολογίες Cloud και FC μπορούν να χρησιμοποιηθούν από κοινού για να καλύψουν τις ανάγκες διαχείρισης δεδομένων σε περιβάλλοντα IoT.

Το FC ορίζεται ως «ένα σενάριο όπου ένας τεράστιος αριθμός από ετερογενείς (ασύρματων και μερικές φορές αυτόνομων) και αποκεντρωμένες συσκευές με πανταχού παρουσία, επικοινωνούν και ενδεχομένως συνεργάζονται μεταξύ τους και με το δίκτυο για την εκτέλεση εργασιών αποθήκευσης και επεξεργασίας χωρίς την παρέμβαση τρίτων. Αυτές οι εργασίες μπορεί να είναι για την υποστήριξη βασικών λειτουργιών δικτύου ή νέων υπηρεσιών και εφαρμογών που εκτελούνται σε περιβάλλον δοκιμών» [4].

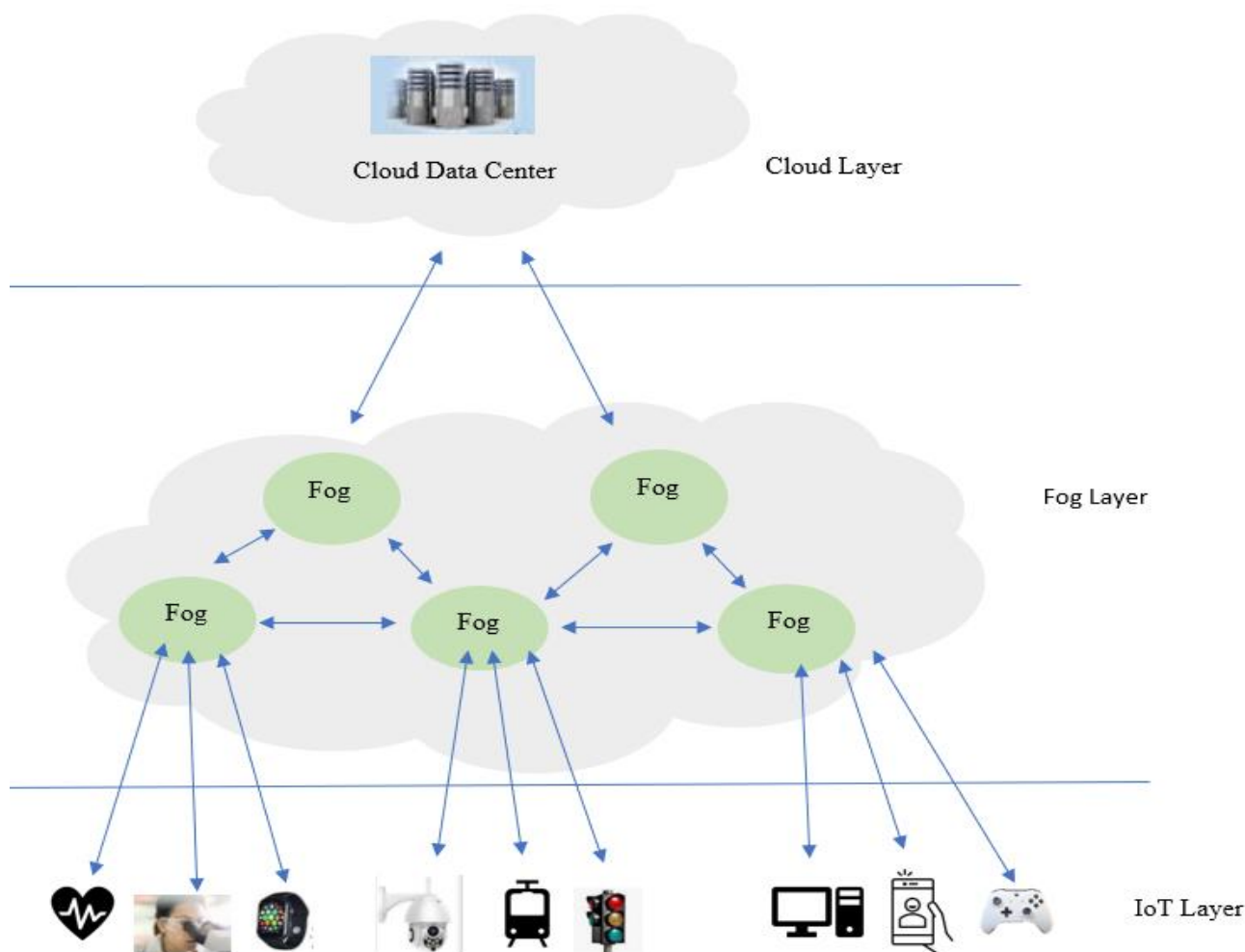
Η αρχιτεκτονική FC συνδέει το Cloud με το IoT, δηλαδή αποτελεί ένα ενδιάμεσο επίπεδο για να παρέχει υπηρεσίες, άμεσα στα άκρα (edges) του δικτύου. Το FC αναδείχθηκε για να καλύψει τις ανάγκες των συσκευών IoT σε αποθήκευση, επεξεργασία, συλλογή και ανταλλαγή δεδομένων, καθώς οι απαιτήσεις για επεξεργασία δεδομένων αυξάνονται καθημερινά. Εξάλλου οι συσκευές IoT αλληλοεπιδρούν με εφαρμογές που είναι ευαίσθητες στην καθυστέρηση και απαιτούν γρήγορη επεξεργασία δεδομένων και χαμηλή κατανάλωση ενέργειας [2].

Το FC μπορεί να βελτιώσει την επικοινωνία μειώνοντας την καθυστέρηση, εξοικονομώντας εύρος ζώνης, υποστήριξη κινητικότητας (mobility), γεωγραφικά κατανεμημένη και αποκεντροποιημένη ανάπτυξη, και υποστήριξη ετερογένειας, διαλειτουργικότητας, ασφάλειας δεδομένων και προστασία της ιδιωτικότητας [2].

Το FC συνδέει το Cloud με τους χρήστες και τα things με τη βοήθεια του δικτύου (core και metro). Το FC μπορεί να παρέχει επιπλέον ενίσχυση, συγκέντρωση, διαχείριση και ασφάλεια των πόρων και των λειτουργιών που διανέμονται στο Cloud, ή βρίσκονται στη διαδρομή μεταξύ Cloud και things, και υποστηρίζει από άκρο σε άκρο τις υπηρεσίες και εφαρμογές. Το FC προσβλέπει στην πραγματοποίηση μια απρόσκοπτης συνέχειας υπολογιστικών υπηρεσιών από το Cloud στα αντικείμενα αντί να θεωρεί τα άκρα του δικτύου ως μεμονωμένες πλατφόρμες υπολογισμού. Επίσης, υποστηρίζει τις κοινές λειτουργίες υπολογισμού για πολλές βιομηχανίες και τομείς εφαρμογών, συμπεριλαμβανομένων και αυτές των παραδοσιακών υπηρεσιών τηλεπικοινωνίας. Το FC είναι μια ευέλικτη εφαρμογή, ικανή να λειτουργήσει σε ενσύρματα και ασύρματα δίκτυα [5].

Η αρχιτεκτονική Fog (Εικόνα 1.1) συνδέει το Cloud με το IoT, δηλαδή αποτελεί ένα ενδιάμεσο επίπεδο για να παρέχει υπηρεσίες, άμεσα στα άκρα (edges) του δικτύου. Το FC σχηματίζει μία ιεραρχική αρχιτεκτονική μεταξύ του Cloud και των things, με τους κόμβους Fog που είναι

διασκορπισμένοι σε διαφορετικά επίπεδα, εξασφαλίζοντας συνεργασία μεταξύ τους, υποστήριξη από άκρο σε άκρο, κατανέμουν τις υπολογιστικές λειτουργίες και διαχειρίζονται, συγκεντρώνουν, ενορχηστρώνουν και προστατεύουν τους κατανεμημένους πόρους και τις λειτουργίες τους. Η επικοινωνία FC ελέγχει πως θα πραγματοποιηθεί η σύνδεση μεταξύ των συσκευών ώστε να μπορέσουν να επικοινωνήσουν παρά τις όποιες συνθήκες του δικτύου εξασφαλίζοντας μειωμένη καθυστέρηση και επιτρέποντας την ανατροφοδότηση για τον έλεγχο των βρόχων [5].



Εικόνα 1.1: Αρχιτεκτονική FC.

Τα πλεονεκτήματα που παρέχονται από την αρχιτεκτονική Fog αναφέρονται με το ακρωνύμιο “SCALE”, αφορούν νέες υπηρεσίες και επιχειρηματικά μοντέλα που αναμένεται να επιφέρουν νέα έσοδα, με μειωμένο κόστος και επιτάχυνση στη διάθεση των προϊόντων.

Security: Μειώνοντας την απόσταση μετακίνησης των πληροφοριών εξαλείφεται η πιθανότητα κακόβουλων εισβολέων.

Cognition: Καθώς οι εφαρμογές FC είναι πιο κοντά τους χρήστες, μπορούν να παρέχουν καλύτερη υποστήριξη των αναγκών των καταναλωτών.

Agility: Είναι μια γρήγορη, καινοτόμος, κλιμάκωση οικονομική και προσιτή λύση για πειραματισμό ή και ανάπτυξη εφαρμογών. Επιτρέπει τη χρήση ανοικτών προτύπων και διευκολύνει την ανάπτυξη νέων καινοτόμων εφαρμογών.

Latency: Παρέχει επεξεργασία σε πραγματικό χρόνο και cyber- physical έλεγχο συστήματος.

Efficiency: Καθώς το FC είναι κοντά στους χρήστες βελτιώνεται η απόδοση του συστήματος. Επίσης μπορεί να κατανέμει πόρους που δε χρησιμοποιούνται ανάλογα με τις ανάγκες και τις απαιτήσεις των χρηστών.

Για παράδειγμα, το σύστημα FC επιτρέπει σε μία πολύπλοκη υπηρεσία να κατανεμηθεί σε ένα σύνολο από στοιχειώδη στοιχεία λογισμικού τα οποία λειτουργούν σε ετερογενείς κόμβους, όπως τερματικά τελικών χρηστών και κέντρα δεδομένων. Αυτό αποτελεί μία πρόκληση για το δυναμικό FC περιβάλλον που περιπλέκεται με τις πολλαπλές FC-προσβάσιμες εφαρμογές που είναι εγκατεστημένες στις τελικές συσκευές των χρηστών, την αναγκαιότητα παροχής διαφορετικής διαχείρισης σε κάθε απαίτηση, την προσαρμογή των υπηρεσιών στην ετερογενή υποδομή του δικτύου και την προσαρμογή των υπηρεσιών στο εξωτερικό περιβάλλον. Επιπλέον, το FC είναι ικανό να ενορχηστρώνει υπηρεσίες ακόμη και σε καταστάσεις οριακών υπολογιστικών δυνατοτήτων με ισχυρό βαθμό ανθεκτικότητας, αλλά και να παρέχει εφαρμογές που προσαρμόζουν αυτόματα την συμπεριφορά τους στο εξωτερικό περιβάλλον για παράδειγμα σε περίπτωση που κάποιες υπηρεσίες δεν είναι προσβάσιμες (π.χ μεγάλος χώρος αποθήκευσης ή υψηλής απόκρισης αισθητήρας), θα καταναίμει την υπηρεσία αυτή στον χρήστη έστω και με κάποια υποβάθμιση της [5].

1.1 Η έννοια του κόστους στο FC

Το FC είναι μία κατανεμημένη αρχιτεκτονική, που αξιοποιεί με επιτυχία τους αποκεντρωμένους πόρους, για την παροχή υπηρεσιών υπολογισμού, μεταφοράς και αποθήκευσης των εφαρμογών IoT μεταξύ Cloud και IoT. Με τη χρήση των έξυπνων αισθητήρων κατανέμονται οι διεργασίες υπολογισμού, εκτέλεσης και αποθήκευσης των εφαρμογών IoT μεταξύ του FC και του κέντρου δεδομένων στο Cloud, παρέχοντας τη βέλτιστη εξυπηρέτηση των αιτημάτων με το ελάχιστο κόστος [6].

Το FC αποτελείται από διάσπαρτους κόμβους Fog που αναμένουν να εξυπηρετήσουν τα αιτήματα των χρηστών κάνοντας χρήση των τοπικών υπολογιστικών πόρων αντί των πόρων του Cloud, μειώνοντας την καθυστέρηση και αυξάνοντας την απόδοση του συστήματος. Η γεωγραφική εγγύτητα

του FC μειώνει τον λανθάνοντα χρόνο μετάδοσης, κάνοντας κατάλληλη αξιοποίηση των τοπικών πόρων των υπολογιστικών συστημάτων για την επεξεργασία των εργασιών που αιτείται [7].

Όμως η ποικιλομορφία των υπολογιστικών πόρων λόγω της διαφορετικής σύνθεσης τους σε διάφορες συσκευές FC, οι περιορισμένοι πόροι και οι αυξανόμενες IoT απαιτήσεις κάνουν την αποτελεσματική εκχώρηση των πόρων, με την επίτευξη της μέγιστης απόδοσης και της ικανοποίησης των αιτημάτων των χρηστών να είναι πολύπλοκο πρόβλημα με αυξημένο κόστος επικοινωνίας, ενέργειας και εκχώρησης των πόρων στα περιβάλλοντα Fog. Ο στόχος του FC είναι να βρεθεί η βέλτιστη εκχώρηση των πόρων για τη βέλτιστη χρήση τους και την ελαχιστοποίηση του κόστους επεξεργασίας, μετακίνησης των δεδομένων των εφαρμογών με την ελάχιστη κατανάλωση ενέργειας και επιδιώκοντας τη μέγιστη απόδοση του συστήματος [8].

1.2 Το πρόβλημα

Ένας κρίσιμος παράγοντας για την αποδοτική λειτουργία ενός τέτοιου σχήματος είναι το κόστος. Η εκτίμηση του κόστους είναι μια πρόκληση αλλά και συνάμα μια σύνθετη διαδικασία λόγω της ετερογένειας, της κλίμακας και της δυναμικότητας των χρησιμοποιούμενων υποδομών. Το κόστος σχετίζεται με την απόκτηση υπηρεσιών ή πόρων, εξαρτάται από τις οντότητες που συμμετέχουν και τους ρόλους τους (π.χ. πάροχοι ή χρήστες). Στο FC το κόστος έχει διάφορες εκφάνσεις, όπως για παράδειγμα το κόστος που αντιστοιχεί στην εκχώρηση των πόρων, τα χρησιμοποιούμενα πρότυπα και πρωτόκολλα, τις υποδομές και την διαθεσιμότητά τους, υπολογίζεται από διαφορετικούς τύπους και συνδέεται με διάφορες παραμέτρους [2] **Error! Reference source not found..**

Σήμερα, υπάρχουν αρκετές εργασίες που επιχειρούν να εκτιμήσουν το κόστος στις αρχιτεκτονικές Fog και προτείνουν αντίστοιχα μοντέλα. Όμως, τα μοντέλα αυτά εστιάζουν κατά κύριο λόγο μόνο σε μια διάσταση του συστήματος, για παράδειγμα το δίκτυο ή τη διαχείριση των πόρων.

Δεδομένου ότι η FC τεχνολογία γνωρίζει ραγδαία εξέλιξη, υπάρχει ανάγκη για την ανάπτυξη ενός μοντέλου που θα εξετάσει και θα αποτιμήσει συνολικά το κόστος μιας τέτοιας υλοποίησης και με απώτερο στόχο να: α) να συμβάλει στην επιλογή της βέλτιστης λύσης σε συνδυασμό με την απόδοση του συστήματος, β) την λήψη αποφάσεων για την τοποθέτηση των υπολογισμών.

1.3 Το αντικείμενο της εργασίας

Το αντικείμενο της παρούσας διπλωματικής εργασίας είναι η διερεύνηση προτεινόμενων μοντέλων κόστους. Γίνεται μελέτη και παρουσίαση των παραμέτρων υπολογισμού του κόστους σε FC και εξετάζεται η πιθανή ενοποίησή τους για την παροχή μιας συνολικής λύσης υπολογισμού κόστους σε τέτοια περιβάλλοντα με τη βοήθεια προσομοίωσης.

Ένα σημαντικό ζήτημα που απαιτείται για την επιτυχία της νέας αυτής τεχνολογίας είναι η βελτιστοποίηση του κόστους της δικτύωσης, με τον εντοπισμό και την καταγραφή των παραμέτρων που το επηρεάζουν, καθώς αυτά μπορούν να χρησιμοποιηθούν για την εφαρμογή του FC στις εφαρμογές με αυξημένες απαιτήσεις.

Στόχος αυτής της εργασίας είναι να προσδιορίσει την έννοια κόστος στο FC, να διερευνήσει τα προτεινόμενα μοντέλα κόστους και να εξετάσει την πιθανή ενοποίησή τους για την παροχή μιας συνολικής λύσης υπολογισμού κόστους σε τέτοια περιβάλλοντα με τη βοήθεια προσομοίωσης.

Στα πλαίσια αυτής της εργασίας η συνεισφορά μας συνοψίζεται στα ακόλουθα:

- Εισάγουμε τις βασικές έννοιες του κόστους στο FC και διερευνούμε το πρόβλημα της βελτιστοποίησης σε συνδυασμό με την απόδοση του συστήματος
- Παρουσιάζουμε και συγκρίνουμε τις υπάρχουσες μελέτες για το προσδιορισμό των παραμέτρων του κόστους στο FC.
- Εξετάζουμε το συνολικό κόστος δικτύου χρησιμοποιώντας διάφορες μετρικές, όπως τα κόστη καθυστέρησης, ενέργειας, επεξεργασίας, εύρους ζώνης και μετεγκατάστασης, δημιουργώντας διάφορες τοπολογίες δικτύου, με διαφορετικό αριθμό εφαρμογών και κόμβων Fog.
- Συγκρίνουμε πειραματικά, τις πηγές κόστους και εξετάζουμε την βελτιστοποίησή του εφαρμόζοντας βασικούς αλγορίθμους βελτιστοποίησης όπως Brute force, Random και Genetic.

1.4 Δομή της εργασίας

Η εργασία αποτελείται από έξι επιμέρους κεφάλαια. Στο πρώτο κεφάλαιο γίνεται αναφορά στην αρχιτεκτονική FC, και ορίζεται το πρόβλημα προσδιορισμού της έννοιας του κόστους, της διερεύνησης και παρουσίασης των παραμέτρων που επηρεάζουν το κόστος του συστήματος, όπου ο προσδιορισμός της βέλτιστης λύσης θα μας οδηγήσει και στην ελαχιστοποίησή του. Το δεύτερο κεφάλαιο εισάγονται τα κόστη της υποδομής και της υλοποίησης των συστημάτων FC, παρουσιάζονται τα βασικά τους στοιχεία και εφαρμόζονται μεθοδολογίες αναζήτησης της βέλτιστης λύσης που θα οδηγήσει στην ελαχιστοποίηση του κόστους. Στο τρίτο κεφάλαιο παρουσιάζεται το κόστος επικοινωνίας και πως η δικτυακή υποδομή, που συμβάλει στη κατανομή των βασικών υπηρεσιών του FC (υπολογισμό, δικτύωση, αποθήκευση) στις τελικές συσκευές, επηρεάζει το κόστος και ποιες είναι βασικές παράμετροι που το επηρεάζουν.

Στο τέταρτο κεφάλαιο γίνεται αναφορά στο κόστος υπολογισμού, που διαχωρίζεται στο κόστος επεξεργασίας/υπολογισμού, στο κόστος εκχώρησης πόρων, στον προγραμματισμό εργασιών, το

κόστος migration (μετεγκατάστασης) των εργασιών από έναν κόμβο Fog σε έναν άλλο, το κόστος offloading, και τις παραμέτρους που το επηρεάζουν. Επίσης αναφερόμαστε στο κόστος για την ασφάλεια στα συστήματα Fog.

Στο πέμπτο κεφάλαιο με τον προσομοιωτή FogComputingSim δημιουργούμε διαφορετικά σενάρια συστήματος Fog από τα οποία καταγράφουμε το συνολικό κόστος δικτύου και το αξιολογούμε.

Στο έκτο και τελευταίο κεφάλαιο ολοκληρώνεται η μελέτη και παρουσιάζονται τα βασικά συμπεράσματα που έχουν προκύψει.

2 Κόστος υποδομής

Στο FC, το κόστος μπορεί να αναλυθεί από την πλευρά των κεφαλαιουχικών δαπανών (Capital Expenditure - Capex) και από την πλευρά των λειτουργικών εξόδων (Operational Expenditure - Opex). Τα κόστη αυτά λαμβάνονται υπόψη κατά τη σχεδίαση και την υλοποίηση των συστημάτων FC και εφαρμόζονται μεθοδολογίες ώστε να βρεθεί η βέλτιστη λύση που να ελαχιστοποιεί το συνολικό κόστος, ενώ εξακολουθούν να ικανοποιούνται οι απαιτήσεις των χρηστών.

2.1 CAPEX

Το CAPEX αφορά τα έξοδα για την απόκτηση ή αναβάθμιση ενός πάγιου, φυσικού και μη αναλώσιμου περιουσιακού στοιχείου μιας εταιρείας και απαιτούνται για να επεκτείνουν στους πελάτες τους. Το CAPEX ενός δικτύου, χωρίζεται σε τέσσερα κύρια μέρη [9]:

- *Το κόστος του κτιρίου ή του κέντρου δεδομένων του δικτύου.*
- *Το κόστος φυσικής υποδομής, που αφορά τα έξοδα υλικού, υποδομής ενός δικτύου, την αγορά οπτικών ινών, δρομολογητών, διακόπτες και ούτω καθεξής.*
- *Το κόστος λογισμικού.*
- *Δάνεια*

Στο μοντέλο αυτό, οι Karakus και Duresi [9], υποθέτουν ότι το κόστος στέγασης και δανείων είναι μηδενικά. Οι ίδιοι συγγραφείς, σε επόμενη εργασία τους, εντάσσουν στο Capex το κόστος αναβάθμισης του εξοπλισμού, δηλαδή την αναβάθμιση των δικτυακών συσκευών ή/και των συνδέσμων.

Στην περίπτωση του FC, οι παραπάνω κατηγορίες κόστους μπορεί να μην εμφανίζονται, ή και να διαφοροποιούνται. Η δόμηση των FC δικτύων αποτελείται από ένα σύνολο από Cloud, gateways, συσκευές Fog, τελικές συσκευές και αισθητήρες που συνήθως, διασυνδέονται ιεραρχικά μεταξύ τους. Σύμφωνα με το [10], το κόστος CAPEX στο FC προέρχεται κυρίως από τη χρήση διαφορετικών τύπων κόμβων Fog και διαφορετικών συνδέσμων, όπως π.χ. ασύρματους και ενσύρματους συνδέσμους.

Οι Zhang κ.ά.[11], εξετάζουν το ζήτημα ανάπτυξης και βελτιστοποίησης της υποδομής για το δίκτυο FC. Για τη μοντελοποίηση του κόστους εστιάζουν στην υποδομή του δικτύου FC που περιλαμβάνει τις συσκευές Fog (π.χ. micro data centers, MicroDCs), τις οπτικές ίνες και τον κατάλληλο εξοπλισμό (π.χ. switches, ενισχυτές, κ.ά.) για τη διασύνδεσή τους. Οι Lin και Yang [12], επιπλέον υπολογίζουν και το κόστος από το Cloud, τις τελικές συσκευές και τους αισθητήρες.

Οι Brogi κ.ά. [13], για την εκτίμηση του κόστους συμπεριλαμβάνουν τις υποδομές Cloud, FC και IoT. Το πλήθος των οντοτήτων από κάθε ομάδα είναι θέμα του σχεδιαστή συστήματος και επιλέγονται με βάση τις απαιτήσεις για την κάλυψη των απαιτήσεων των καταναλωτών της εφαρμογής. Οι κόμβοι Cloud θεωρούνται ως εικονικές μηχανές (VMs) τυπικές ή οριζόμενες κατά απαίτηση. Οι κόμβοι Fog παρέχουν δυνατότητες λογισμικού και περιορισμένους πόρους υλικού. Το υλικό, τόσο στους κόμβους Cloud όσο και στους κόμβους Fog περιλαμβάνει τη μνήμη RAM, τον αποθηκευτικό χώρο (HDD) και τον επεξεργαστική ισχύς (CPUs). Το κόστος των VMs είναι ανάλογο των διατιθέμενων πόρων, το οποίο καταβάλλεται στους παρόχους μηνιαίως. Στην περίπτωση αυτή, το συγκεκριμένο κόστος λογίζεται ως OPEX. Επίσης, το υλικό και το λογισμικό που διαθέτουν οι κόμβοι, έχει ένα μηνιαίο κόστος. Για το υλικό, το κόστος υπολογίζεται ανά μονάδα χρήσης ενώ για το λογισμικό υπολογίζεται σε σχέση με το είδος του λογισμικού που ζητούν για χρήση οι κόμβοι.

2.2 OPEX

Τα έξοδα OPEX ενός δικτύου χωρίζονται σε τρεις κατηγορίες [9]:

A) *OPEX για τη λειτουργία του δικτύου*: Αυτή η κατηγορία σχετίζεται με το κόστος λειτουργίας και συντήρησης ενός δικτύου. Και αφορά:

- *Τα έξοδα διατήρησης του δικτύου σε λειτουργία* όπως το κόστος του χώρου, το κόστος ενέργειας και το κόστος μίσθωσης του εξοπλισμού του δικτύου.
- *Κόστος συντήρησης που σχετίζεται με τις πληροφορίες της κατάστασης του δικτύου, διαχείρισης των αποθεμάτων και διαχείρισης του λογισμικού* (π.χ. παρακολούθηση εκδόσεων λογισμικού και εγκατάσταση ενημερώσεων).
- *Κόστος αποζημίωσης*. Πρόκειται για έξοδα που προκύπτουν από την επισκευή της βλάβης, έξοδα των τεχνικών, κλπ.
- *Κόστος παροχής και διαχείρισης των υπηρεσιών*. Η παροχή υπηρεσιών αφορά ένα αίτημα από έναν δυνητικό πελάτη για έναρξη, μετακίνηση και/ή διακοπή μιας υπηρεσίας, επεξεργασία μίας παραγγελίας από τον διαχειριστή του δικτύου και εκτέλεση της απαραίτητης δοκιμής για την υπηρεσία που παρέχεται.
- *Τιμολόγηση και κόστος χρέωσης*. Δηλαδή την αποστολή λογαριασμών σε πελάτες και εξασφάλιση των πληρωμών τους.
- *Κόστος επιχειρησιακού σχεδιασμού*, που αφορά τις δραστηριότητες προγραμματισμού του δικτύου, όπως ο καθημερινός προγραμματισμός του και οι προγραμματισμένες αναβαθμίσεις.

- *Κόστος μάρκετινγκ.*

B) *OPEX για τη δημιουργία δικτύου που σχετίζονται με το κόστος που απαιτείται πριν τη σύνδεση του πρώτου πελάτη και διακρίνεται σε:*

- *Κόστος εγκατάστασης.* Αυτά τα κόστη σχετίζονται με την αρχική εγκατάσταση του εξοπλισμού, τις συνδέσεις και τον έλεγχο του νέου στοιχείου στο δίκτυο.
- *Κόστος Προγραμματισμού* για την αξιολόγηση ενός νέου δικτύου, τυχόν αλλαγές στη τοπολογία του, εισαγωγή νέας τεχνολογίας ή μίας νέας υπηρεσίας πλατφόρμας.

Γ) *Μη συγκεκριμένο κόστος OPEX.*

- *Αφορά το κόστος όπως η θέρμανση, ο καθαρισμός κτιρίων κλπ.*
- *Μη ειδικό κόστος διαχείρισης που σχετίζεται με τις διοικητικές ενέργειες όπως την πληρωμή στη διοίκηση των εργαζομένων, το ανθρώπινο δυναμικό τμήμα κλπ.*
- *Ειδικά κόστη που προκύπτουν κατά τη λειτουργία του δικτύου, όπως για παράδειγμα το κόστος που εισάγουν τα πλεονάζοντα μηνύματα σε ένα δίκτυο [9].*

2.3 Προσεγγίσεις υπολογισμού κόστους

2.3.1 Γενικό μοντέλο Υπολογισμού του OPEX

Παρόλο που το μοντέλο [9], δεν αναφέρεται σε δίκτυα FC, μπορεί να βρει εφαρμογή στην εκτίμηση του OPEX, ως ένα γενικό μοντέλο. Το OPEX συναθροίζει τα έξοδα λειτουργίας του δικτύου, τα έξοδα εγκατάστασης/ανάπτυξης του δικτύου και τα μη-συγκεκριμένα έξοδα. Τα έξοδα αυτά περιλαμβάνουν την αμοιβή των παρόχων υπηρεσιών υποδομής (ISP) για τη χρήση των υπηρεσιών επικοινωνίας και υπολογισμού. Το [10] συνυπολογίζει στο OPEX το κόστος χρήσης των κόμβων Fog ή/και των κόμβων Cloud που δεσμεύονται για την εξυπηρέτηση μιας διεργασίας στο IoT. Όπως αναφέρθηκε και παραπάνω το OPEX εξαρτάται από τον τύπο του κόμβου Fog και το είδος του συνδέσμου επικοινωνίας που θα χρησιμοποιηθεί. Μια ευέλικτη ανάπτυξη, με την επιλογή παρόχου κόμβων, δεδομένου ότι το κόστος ποικίλει ανά πάροχο και τύπο κόμβου, θα συμβάλει στη μείωση του κόστους OPEX του FC [14].

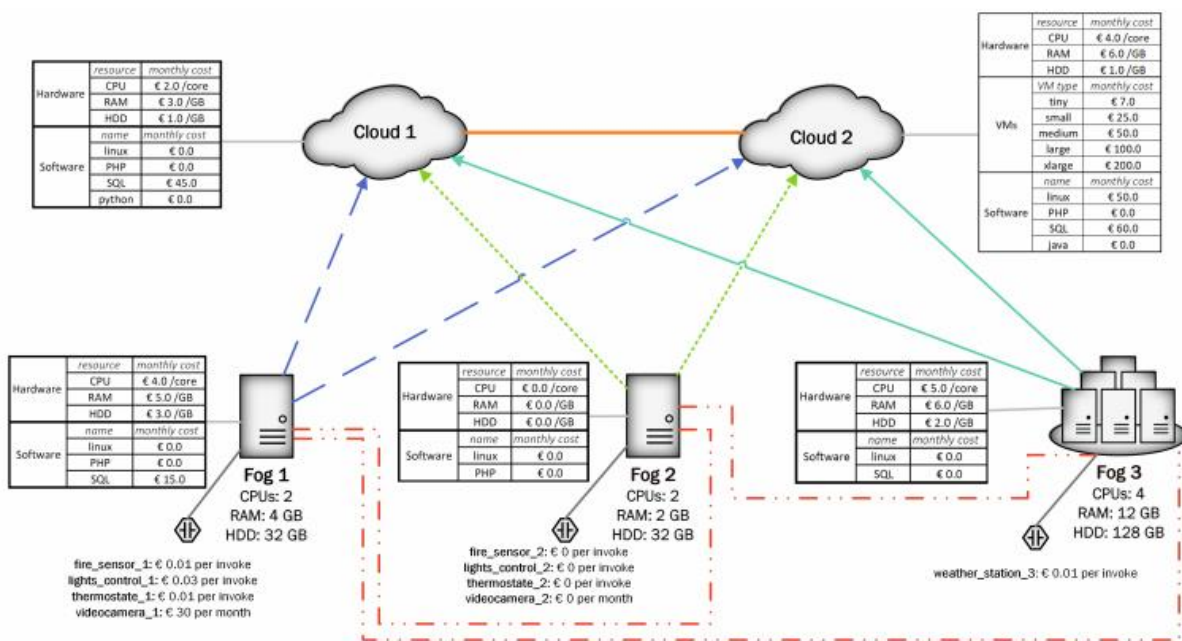
2.3.2 Μοντέλο cost-aware

Οι Brogi κ.ά.[13], προτείνουν ένα μοντέλο για την εκτίμηση του μηνιαίου κόστους που απαιτείται για την ανάπτυξη εφαρμογών σε υποδομές IoT, Fog και Cloud. Το μοντέλο κόστους, προστίθεται ως

επέκταση στον προσομοιωτή FogTorchΠ και δίνει έμφαση στον τρόπο ανάπτυξης των εφαρμογών στην υφιστάμενη υποδομή σε σχέση με:

- α) *QoS-awareness*, με την οποία επιτυγχάνεται μείωση της καθυστέρησης, εξοικονόμηση εύρους ζώνης και επιβάλλει επιχειρηματικές πολιτικές,
- β) *Context-awareness* για καλύτερη εκμετάλλευση των τοπικών και απομακρυσμένων πόρων και
- γ) *Cost-awareness* για την προσέγγιση οικονομικά αποδοτικών εφαρμογών.

Στην *Εικόνα 2.1* φαίνεται ένα χαρακτηριστικό παράδειγμα υπολογισμού κόστους σε υποδομή του δικτύου FC, όπως εφαρμόζεται στον προσομοιωτή FogTorchΠ.



Εικόνα 2.1: Υποδομή δικτύου Fog στον προσομοιωτή FogTorch, πηγή [13].

Η υποδομή του δικτύου αποτελείται από δύο κέντρα δεδομένων Cloud, τρεις κόμβους Fog και εννιά things, και επιλέχθηκαν από το σχεδιαστή του συστήματος για την κάλυψη των απαιτήσεων της εφαρμογής. Οι χρήστες της εφαρμογής αξιοποιούν τις τελικές συσκευές και εκμεταλλεύονται τους πόρους των κόμβων Cloud και Fog. Οι κόμβοι Fog παρέχουν δυνατότητες λογισμικού και περιορισμένους πόρους υλικού (RAM, HDD, CPUs). Οι κόμβοι Cloud παρέχουν λογισμικό και υλικό με αντίστοιχη χρέωση. Οι κόμβοι Cloud μπορούν να διαθέσουν “άπειρους” πόρους στους χρήστες, αρκεί να τους μισθώσουν.

Για παράδειγμα, η αγορά ενός μικρού στοιχείου στον κόμβο Cloud2 στοιχίζει €7 ανά μήνα. Στον κόμβο Fog2 τα στοιχεία διατίθενται χωρίς κόστος για την κάλυψη των απαιτήσεων των χρηστών.

Το μοντέλο κόστους FogTorchΠ κάνει χρήση της μεθόδου Monte Carlo για να γίνει η εκτίμηση του QoS-assurance αφού εξετάσει τις πιθανές εκδοχές δεδομένης της ποικιλομορφίας στις συνδέσεις επικοινωνίας.

Στο συγκεκριμένο μοντέλο κόστους, ο υπολογισμός του κόστους περιλαμβάνει τα έξοδα μίσθωσης για το λογισμικό και τις συσκευές IoT. Για κάθε κόμβο FC ή Cloud, υπολογίζεται το κόστος υλικού και λογισμικού. Το κόστος υλικού καθορίζεται σε σχέση με την ποσοστιαία κατανάλωση των πόρων (CPU, RAM και HDD) που καταναλώνει ένας κόμβος, είτε με προκαθορισμένη μηνιαία χρέωση των VMs, είτε κυμαίνεται ανάλογα με τη χρήση (on demand) των πόρων από τις VMs.

Συγκεκριμένα, το μηνιαίο κόστος του υλικού (H) που παρέχεται στον κόμβο n είναι:

$$p(H, n) = \begin{cases} p(H, n), & \text{για προεπιλεγμένη χρέωση των VMs} \\ \sum_{p \in R} [H \cdot p \times c(p, n)], & \text{για χρέωση των VMs ανάλογα με τη χρήση (on - demand)} \end{cases}$$

Όπου, R είναι το σύνολο των πόρων CPU, RAM και HDD, $H \cdot p$ ορίζει τον αριθμό των πόρων ($p \in R$) που χρειάζεται η VM και $c(p, n)$ είναι το μηνιαίο κόστος στον κόμβο n για τον πόρο p .

Αντίστοιχα, το μηνιαίο κόστος του λογισμικού (S) στον κόμβο n είναι:

$$p(S, n) = \begin{cases} c(S, n), & \text{για προεπιλεγμένη χρέωση των VMs} \\ \sum_{p \in R} c(s, n), & \text{για χρέωση των VMs ανάλογα με τη χρήση (on - demand)} \end{cases}$$

Όπου, $c(S, n)$ είναι η χρέωση για προκαθορισμένη χρήση πακέτων λογισμικού στον κόμβο n και $c(s, n)$ είναι η μηνιαία χρέωση ανάλογα με τη χρήση του λογισμικού στον κόμβο n , με $s \in S$.

Τέλος, σε περιπτώσεις χρήσης αντικειμένων σύμφωνα με το μοντέλο Sensing-as-a-Service, η χρέωση ενός αντικειμένου υπολογίζεται είτε με βάση μια μηνιαία συνδρομή, είτε σε σχέση με το πλήθος των αιτημάτων που έγιναν για αυτό το αντικείμενο.

Συγκεκριμένα:

$$p(T, t) = \begin{cases} c(T, t), & \text{για προεπιλεγμένη χρέωση των VMs} \\ T \cdot k * c(t), & \text{για χρέωση των VMs ανάλογα με τη χρήση (on - demand)} \end{cases}$$

Όπου $c(T, t)$ είναι η μηνιαία συνδρομή για το T στο t , $T \cdot k$ είναι ο αριθμός των μηνιαίων κλήσεων για t και $c(t)$ είναι το κόστος κλήσης των t (συμπεριλαμβανομένου τα things και/ή το κόστος μεταφοράς των δεδομένων).

Για να είναι επιλέξιμη η ανάπτυξη μιας εφαρμογής (eligible deployment), θα πρέπει να ικανοποιούνται οι απαιτήσεις σε πόρους υλικού και λογισμικού, η απρόσκοπτη πρόσβαση στα αντικείμενα και η επικοινωνία μεταξύ των αντικειμένων της εφαρμογής και να είναι δυνατή υπό τον περιορισμό της χωρητικότητας των συνδέσεων. Τότε, το συνολικό μηνιαίο κόστος ανάπτυξης της εφαρμογής A , υπολογίζεται συνδυάζοντας τις προηγούμενες εξισώσεις κόστους με τις απαιτήσεις των εφαρμογών σε υλικό, λογισμικό και υποδομή με την εξίσωση:

$$cost_{(H,S,I)} = \sum_{\gamma \in A} \left[p_m(H_\gamma^A) + p_m(S_\gamma^A) + \sum_{r \in T} p_m(r_s) \right]$$

Όπου, A είναι η επιλέξιμη ανάπτυξη μιας εφαρμογής, $\gamma \in A$, είναι ένα στοιχείο της εφαρμογής A και $r \in T$ και H , S και T είναι το υλικό, το λογισμικό και τα αντικείμενα αντίστοιχα για τη κάλυψη των απαιτήσεων του συστήματος.

2.3.3 Πολυεπίπεδα μοντέλα

Οι Lin κ.ά. [12], θεωρούν ότι μια αρχιτεκτονική Fog αναπτύσσεται σε επίπεδα στα οποία τοποθετούνται από πάνω προς τα κάτω οι κόμβοι Cloud, οι πύλες, οι συσκευές Fog, οι τελικές συσκευές και οι συσκευές ανίχνευσης (αισθητήρες). Για τον υπολογισμό του κόστους μιας τέτοιας ανάπτυξης, λαμβάνεται υπόψη το κόστος επικοινωνίας μεταξύ των κόμβων καθώς και το κόστος υποδομής που περιλαμβάνει την εγκατάσταση των πυλών, των συσκευών Fog και των τελικών συσκευών.

Η θέση των συσκευών καθορίζεται από ένα μαθηματικό μοντέλο και χρησιμοποιώντας ακέραιο προγραμματισμό, οι συγγραφείς προσπαθούν να ελαχιστοποιήσουν το συνολικό κόστος θέτοντας μια σειρά από περιορισμούς που σχετίζονται με:

- Το πλήθος των συνδέσεων μεταξύ των κόμβων.
- Το πλήθος των συσκευών που αντιστοιχούν σε κάθε κόμβο Fog.
- Τη δυνατότητα ανάπτυξης ή όχι σε έναν κόμβο (πύλη, κόμβο Fog).
- Τη χωρητικότητα που καθορίζεται από ένα μέγιστο όριο.
- Την καθυστέρηση που πρέπει να είναι μικρότερη από ένα μέγιστο όριο.
- Την κάλυψη του δικτύου σε περίπτωση ασύρματης επικοινωνίας.
- Τη μέγιστη χωρητικότητα συσκευών.

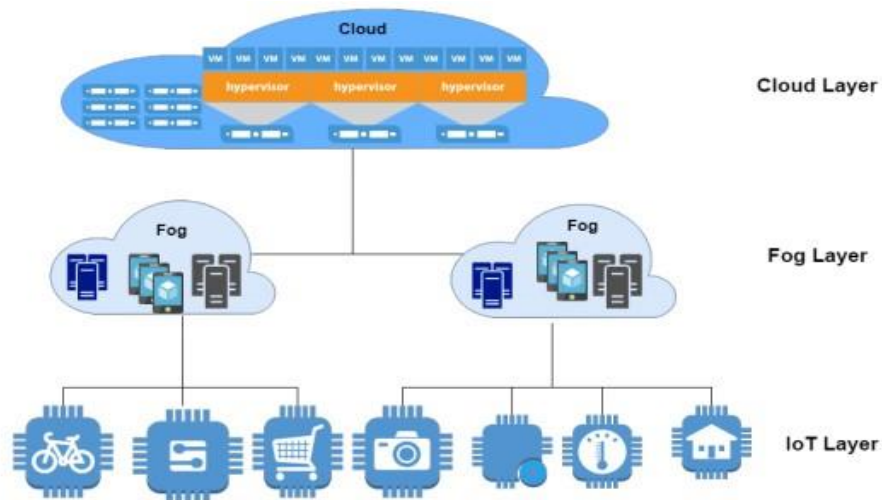
Οι Zhang κ.ά. [11], διερευνούν μια οικονομικά αποδοτική ανάπτυξη και βελτιστοποίηση υποδομής FC που η δόμησή του περιλαμβάνει Micro Data Center και Long Reach Passive Optical Network (MicroDC-LRPON). Το FC ακολουθεί μια πολυεπίπεδη δεντρική δομή. Το κόστος απαρτίζεται από το κόστος του εξοπλισμού και ειδικότερα από το κόστος των splitters, των MicroDCs, των ενισχυτών οπτικής ίνας και της οπτικής ίνας. Για την επίλυση του προβλήματος και την ελαχιστοποίηση του κόστους τίθενται μια σειρά από περιορισμούς. Συγκεκριμένα, οι περιορισμοί είναι:

- Διατήρηση της τοπολογίας του δέντρου ολόκληρου του δικτύου
- Οι θυγατρικοί κόμβοι ενός splitter (διακλαδωτή) δεν μπορούν να έχουν μεγαλύτερο λόγο διαίρεσης διαχωρισμού συγκριτικά με τον δικό του
- Η συνολική εξασθένιση ισχύος πρέπει να είναι μικρότερη από το προκαθορισμένο όριο.
- Τα MicroDCs πρέπει να προσαρμόζονται στους περιορισμούς χωρητικότητας.
- Η καθυστέρηση θα πρέπει να είναι μικρότερη από το καθοριζόμενο μέγιστο όριο.
- Κάθε μονάδα οπτικού δικτύου μπορεί να εξυπηρετηθεί από ένα MicroDC.
- Περιορισμός αρχιτεκτονικής: μπορεί να τοποθετηθεί ένας διαχωριστής/MicroDC με συγκεκριμένο τύπο σε πιθανή τοποθεσία για την ανάπτυξη διαχωριστών ή MicroDC αντίστοιχα.

2.3.4 Κόστος ενέργειας

Οι Battula κ.ά. [8], προτείνουν ένα μοντέλο κόστους χαμηλής κατανάλωσης ενέργειας το οποίο εξαρτάται από το συνολικό αριθμό των συνδεδεμένων αισθητήρων κατά τη διάρκεια της επεξεργασίας (Εικόνα 2.2). Συγκεκριμένα το κόστος της ενέργειας υπολογίζεται σε σχέση με:

- 1) με το συνολικό αριθμό των συσκευών Fog,
- 2) το σύνολο των αισθητήρων που συνδέονται στις συσκευές Fog,
- 3) το χρόνο αδράνειας που είναι ο χρόνος καθυστέρησης μεταξύ ανίχνευσης και αποστολής
- 4) το κόστος της μπαταρίας.



Εικόνα 2.3: Fog Computing, πηγή [8].

Κατά τους Liu κ.ά. [16], η κατανάλωση ενέργειας της κινητής συσκευής mobile device (MD) i, χωρίζεται σε δύο μέρη:

- 1) κατανάλωση ενέργειας της MD για την επεξεργασία αιτημάτων τοπικών υπηρεσιών και
- 2) κατανάλωση ενέργειας για τη μετακίνηση δεδομένων στο σταθμό βάσης (Base Station).

Κατά τους Suryadevara κ.ά. [17], η συνολική κατανάλωση ενέργειας επηρεάζεται από το συνολικό μήκος των πακέτων, το εύρος ζώνης που απαιτείται για μεταφόρτωση και λήψη, την παροχή ρεύματος και το συνολικό κύκλο που κάνει μία διεργασία για την εκτέλεση της. Ο τύπος υπολογισμού της συνολικής ενέργειας είναι το άθροισμα της ενέργειας που καταναλώνεται:

1. από τους αισθητήρες,
2. τον αλγόριθμο ταξινόμησης και
3. την εκτέλεση υπολογισμών και μεταφορά των δεδομένων.

Οι Bellendorf κ.ά. [18], επιτυγχάνουν την ελαχιστοποίηση της κατανάλωσης ενέργειας κατά την εκτέλεση των διεργασιών υπολογισμού των κινητών συσκευών λαμβάνοντας υπόψη τους περιορισμούς των πόρων και την προκαθορισμένη ανώτερη καθυστέρηση. Οι διεργασίες από τις τελικές συσκευές γίνονται offloading στους κόμβους Fog ή στο κέντρο δεδομένων στο Cloud ανάλογα με το επίπεδο της μπαταρίας, την υπολογιστική ισχύ του επεξεργαστή στις τελικές συσκευές και τους περιορισμούς στο φορτίο των κόμβων Fog.

Κατά τους Ahvar κ.ά. [19], η κατανάλωση ενέργειας χωρίζεται σε στατική και σε δυναμική.

Η στατική ενέργεια είναι το αφορά την στατική κατανάλωσης ενέργειας των Physical Machines (PMs) και των switch στα Data Center (DC), όταν αυτοί είναι σε λειτουργία αλλά δεν εκτελούν καμία διεργασία. Ακόμη αφορά την προμήθεια ενέργειας στα DCs και στους ανεμιστήρες. Στο δίκτυο επικοινωνίας η στατική κατανάλωση ενέργειας αφορά την κατανάλωση ενέργειας των δρομολογητών όταν είναι σε αδράνεια και εργάζονται για τη σύνδεση των DCs με τους τελικούς χρήστες.

Η δυναμική ενέργεια εξαρτάται από τη χρήση της CPU των PMs σε συγκεκριμένη χρονική στιγμή και τη δυναμική ενέργεια στο εσωτερικό του DC. Για τον υπολογισμό της λαμβάνεται υπόψη η κατανάλωση ενέργειας σε κάθε switch, όταν αυτός μεταφέρει δεδομένα, όπου και καταναλώνει επιπλέον ενέργεια κατά την επεξεργασία πακέτων, την αποθήκευση και την μετακίνησή τους.

Οι Ren κ.ά. [20], στο μοντέλο ελαχιστοποιούν την κατανάλωση ενέργειας προτείνοντας έναν αλγόριθμο προγραμματισμού εργασιών που προγραμματίζει τους πόρους στο Cloud συνδυάζοντας τους με τα ασύρματα κανάλια και θέτοντας περιορισμό στο χρόνο ολοκλήρωσης.

2.4 Συμπεράσματα

Το κόστος υποδομής στο FC, αφορά τη σχεδίαση και την υλοποίηση του συστήματος και διακρίνεται στα κεφαλαιουχικά και λειτουργικά έξοδα. Στην ενότητα αυτή παρουσιάστηκαν διάφορες έρευνες και μοντέλα και έγινε μια συνοπτική ανάλυση των παραγόντων που συμβάλουν στον υπολογισμό του κόστους υποδομής και στη μείωση της κατανάλωσης ενέργειας, θέτοντας μια σειρά από περιορισμούς με στόχο να βρεθεί η βέλτιστη λύση που θα ελαχιστοποιηθεί το κόστος της υποδομής και την ενέργεια του δικτύου (Πίνακας 2.1).

Τα κεφαλαιουχικά έξοδα σε ένα περιβάλλον FC συνδέονται με τα κόστη του κτιρίου, της φυσικής υποδομής, το λογισμικό, τα δάνεια και το κόστος αναβάθμισης του εξοπλισμού [9]. Μερικοί ερευνητές υπολογίζουν το κόστος υποδομής στο FC σύμφωνα με τη χρήση διαφορετικών τύπων κόμβων Fog και διαφορετικών συνδέσμων, όπως π.χ. ασύρματους και ενσύρματους [10]. Άλλες ερευνητικές εργασίες εστιάζουν στις συσκευές Fog (π.χ. MicroDCs), στις οπτικές ίνες και στον κατάλληλο εξοπλισμό (π.χ. switch, ενισχυτές, κ.ά.) για την διασύνδεσή τους. Ενώ κάποιοι συνυπολογίζουν στο κόστος το Cloud, τις τελικές συσκευές και τους αισθητήρες [11][12][13].

Το κόστος λειτουργίας του FC είναι τα έξοδα διατήρησης, συντήρησης, αποζημίωσης, παροχής και διαχείρισης των υπηρεσιών, τιμολόγησης και κόστη χρέωσης, επιχειρησιακού σχεδιασμού και μάρκετινγκ, το κόστος σύνδεσης του πρώτου πελάτη στο δίκτυο και τα κόστη που αφορούν τη θέρμανση, καθαρισμό των κτιρίων κλπ., η πληρωμή των εργαζομένων και ειδικά κόστη όπως το κόστος που εισάγουν τα πλεονάζοντα μηνύματα σε ένα δίκτυο [9][13].

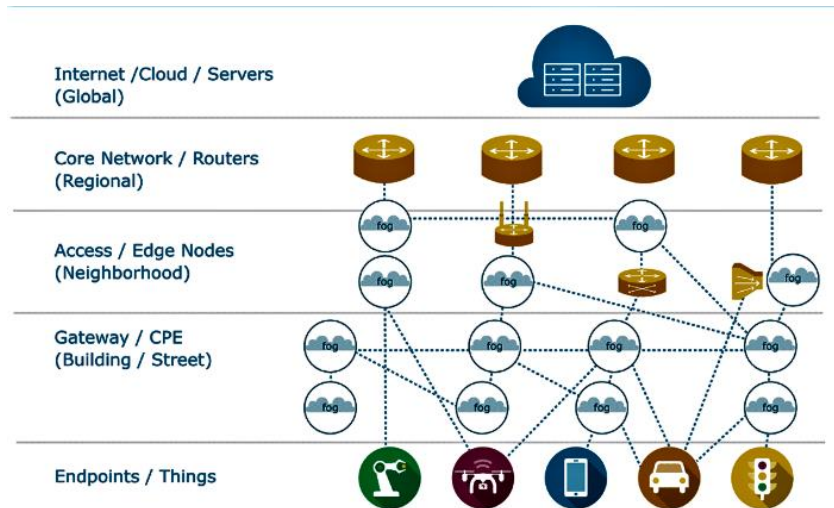
Αρκετοί ερευνητές εστιάζουν στο κόστος χρήσης των κόμβων Fog ή/και των κόμβων Cloud που δεσμεύονται για την εξυπηρέτηση μιας διεργασίας στο IoT, συνυπολογίζοντας το κόστος υλικού και λογισμικού για κάθε κόμβο Fog και Cloud, το κόστος επικοινωνίας μεταξύ των κόμβων καθώς και το κόστος υποδομής για την εγκατάσταση των πυλών, των συσκευών Fog, και των τελικών συσκευών [12][13].

Σε άλλες εργασίες προτείνεται το μοντέλο κόστους χαμηλής κατανάλωσης ενέργειας υπολογίζοντας το κόστος των μπαταριών, των αισθητήρων που χρησιμοποιούνται κατά την εκτέλεση μίας διεργασίας [2][8][16] [18]. Άλλοι ερευνητές υπολογίζουν τη μέγιστη χωρητικότητα απαιτήσεων, το μέγιστο χρόνο καθυστερήσεων και τη μέγιστη χωρητικότητα των συσκευών [11][12][18][20]. Ενώ κάποιοι άλλοι μειώνουν το κόστος ενέργειας λαμβάνοντας υπόψη το μήκος των πακέτων, το εύρος ζώνης, τη χωρητικότητα του επεξεργαστή, την ενέργεια που καταναλώνεται κατά την επεξεργασία, αποθήκευση και μετακίνηση των πακέτων, ακόμη και την ενέργεια που καταναλώνουν οι συσκευές όταν είναι σε αδράνεια [17][18][19].

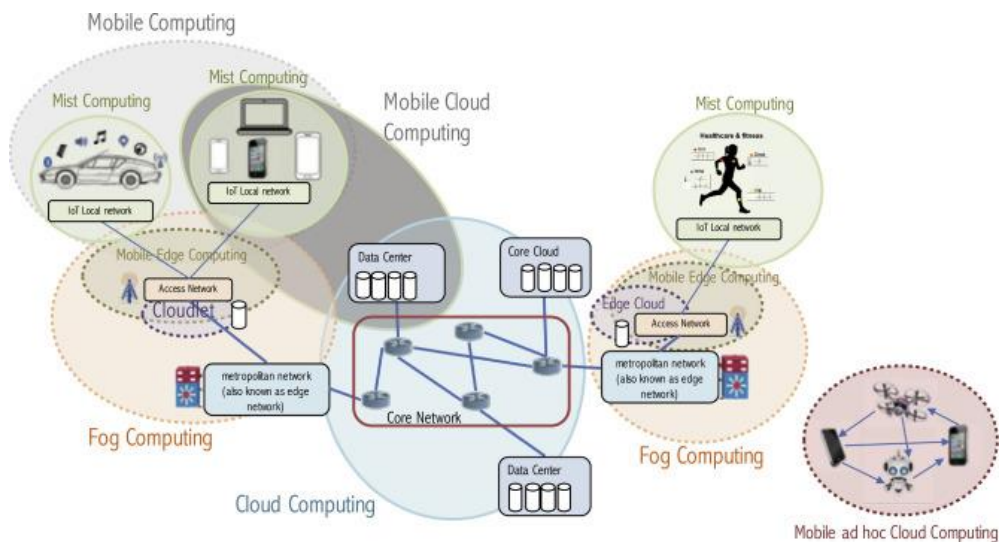
Πίνακας 2.1: Παράμετροι υπολογισμού κόστους υποδομής.

Αναφορές	cost factor	Αιτούμενη χωρητικότητα	Κόστος αναβάθμισης της λειτουργίας του δικτύου	Προγραμματισμό εργασιών	Coverage	Μήκος των πακέτων	Εύρος ζώνης ανεβάσιματος και κατεβάσιματος	Εκτέλεση μιας διεργασίας	Κατανάλωση ρεύματος	Επεξεργασία πακέτων, αποθήκευση και μετακίνησή τους	Κόστος μπαταρίας	Συνολικός αριθμός συσκευών Fog και αισθητήρων που συνδέονται σε αυτές	Χρόνος αναμονής ανάμεσα σε αισθητήρα και το χρόνο αποστολής	Chassis power και ανεμιστήρες	Τοποθεσία	Σταθερά έξοδα υπηρεσιών υποδομής	Τοπολογία	Έξοδα απόζημίωσης \ Έξοδα εκτέλεσης δικτύου	Περιορισμούς των πόρων	Έξοδα εγκατάστασης δικτύου	Επίπεδο της μπαταρίας	Έξοδα αρχικού λογισμικού και αρχικής φυσικής υποδομής	Δοκιμές Υπηρεσιών	Σχεδιασμός και Λογισμικό	Χρόνος καθυστέρησης	Μηνιαίο κόστος για πόρους (CPU, RAM, HDD)	Tuning up	Χωρητικότητα συσκευών, επεξεργαστή, αιτήσεων και συνδέσεων	Έξοδα εγκατάστασης δικτύου	Έξοδα συντήρησης	Αρχιτεκτονική δικτύων και εξοπλισμός (splitters, MicroDCs, OAs και οπτική ίνα)			
(Karakus 2018), [9]		✓													✓			✓		✓		✓	✓			✓			✓					
Broggi 2018, [13]																										✓								
Lin 2018, [12]		✓			✓																				✓			✓						
Zhang 2017, [11]														✓			✓																✓	
Battula 2019, [8]									✓	✓	✓																							
Suryadevara 2021, [17]						✓	✓	✓	✓																									
Bellendorf 2020, [18]																			✓		✓				✓			✓						
Ahvar 2019, [19]								✓	✓	✓				✓																				
Ren 2019, [20]				✓																					✓									

3 Κόστος Επικοινωνίας



Εικόνα 3.1: Διανομή Fog Υπηρεσιών, πηγή [21].



Εικόνα 3.2: Υποδομή Δικτύωσης Fog, πηγή [22].

Το FC, όπως αναφέρεται στο OpenFog Consortium, έχει ιεραρχική δόμηση και παρέχει διάφορες υπηρεσίες (υπολογισμού, δικτύωσης, αποθήκευσης, ελέγχου και επιτάχυνσης) οπουδήποτε από το

Cloud στα αντικείμενα IoT [22]. Η όλη προσπάθεια για απρόσκοπτη συνέχεια των υπολογιστικών υπηρεσιών μεταξύ αυτών βασίζεται στη δικτύωση. Οι υπηρεσίες δικτύωσης κατανέμονται μεταξύ του κεντρικού δικτύου, δηλαδή σε επίπεδο κορμού που χρησιμοποιούνται IP/MPLS τεχνολογίες, το επίπεδο πρόσβασης, δηλαδή των κόμβων πρόσβασης που χρησιμοποιούν 3G/4G/Ethernet, και το κατώτερο επίπεδο που περιλαμβάνει τις τελικές συσκευές του δικτύου (Εικόνα 3.1: Διανομή Fog Υπηρεσιών, πηγή). Ο Yousefrouf κ.ά. [22] παρουσιάζει την υποδομή δικτύωσης του FC συγκριτικά με άλλες παρόμοιες τεχνολογίες όπως απεικονίζεται (Εικόνα 3.2: Υποδομή Δικτύωσης Fog, πηγή). Η δικτυακή υποδομή περιλαμβάνει βασικούς πόρους (συσκευές, συνδέσμους, κλπ.) με διαφορετικές δυνατότητες όπως ταχύτητα σύνδεσης, χωρητικότητα, τεχνολογία πρόσβασης στο δίκτυο, λειτουργικά συστήματα, κ.ά.. Η δικτυακή υποδομή συμβάλει στη διανομή των βασικών υπηρεσιών του FC (υπολογισμό, δικτύωση, αποθήκευση) στις τελικές συσκευές. Ουσιαστικά, η επικοινωνία στο FC αφορά το πώς οι συσκευές μπορούν να επικοινωνούν μεταξύ τους παρά τη διαλείπουσα παγκόσμια συνδεσιμότητα [11].

3.1 Η έννοια του κόστους επικοινωνίας

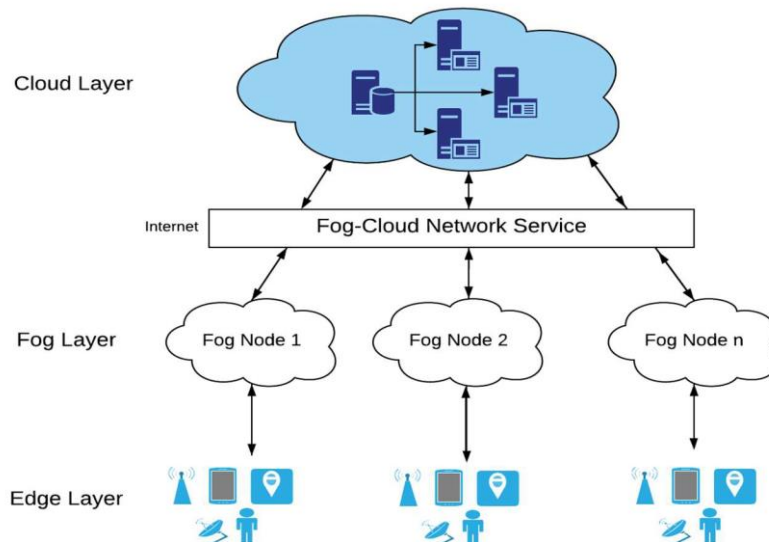
Το κόστος επικοινωνίας αναφέρεται από τη διαβίβαση των δεδομένων από και προς τις συσκευές Fog και εξαρτάται από το συνολικό αριθμό των μεταδιδόμενων μηνυμάτων, το μέγεθος του κάθε μηνύματος και το κόστος ανά μονάδα μηνύματος [8]. Το κόστος επικοινωνίας μιας εφαρμογής σε αυτό το περιβάλλον υπολογίζεται με βάση την ταχύτητα και τον όγκο των δεδομένων που διαβιβάζονται από και προς τις συσκευές Fog. Το κόστος επικοινωνίας εξαρτάται επίσης και από τα χαρακτηριστικά της σύνδεσης, όπως χωρητικότητα, συνθήκες δικτύου ή επιθυμία για ιδιωτικότητα [23]. Για να εξετάσουμε το κόστος επικοινωνίας σε ένα σύστημα FC θα πρέπει να κατανοήσουμε τον τρόπο δόμησης του και τις παραμέτρους που επηρεάζουν το κόστος.

3.2 Διασύνδεση και τεχνολογίες δικτύωσης

Ένα σύστημα FC τοποθετείται μεταξύ του Cloud και των τελικών συσκευών (Εικόνα 3.3). Διασυνδέει γεωγραφικά κατανεμημένες συσκευές και αξιοποιεί την υπάρχουσα δικτυακή υποδομή, όπως για παράδειγμα το δίκτυο πρόσβασης και το metro δίκτυο [22] προκειμένου να παρέχει δυνατότητες επικοινωνίας. Η επικοινωνία μπορεί να βασίζεται σε ενσύρματες ή ασύρματες συνδέσεις και υποστηρίζεται από τις σύγχρονες τεχνολογίες δικτύωσης ενσύρματες ή και ασύρματες. Τυπικά, η επικοινωνία μεταξύ των τελικών κόμβων και των κόμβων Fog είναι ασύρματη, ενώ οι κόμβοι Fog για τις μεταξύ τους επικοινωνίες ή τις επικοινωνίες με Clouds χρησιμοποιούν ενσύρματες συνδέσεις(IP/MPLS).

Γενικά, οι συσκευές Fog διατηρούν τρεις διαφορετικούς τύπους σύνδεσης [24][25]88:

- Επικοινωνία των κόμβων Fog με τις κινητές συσκευές (χρήστες).
- Επικοινωνία μεταξύ των κόμβων Fog.
- Επικοινωνία μεταξύ Cloud και FC.



Εικόνα 3.3: Αρχιτεκτονική Fog computing, πηγή [25].

Για τον πρώτο τύπο επικοινωνίας, υπάρχουν βασικά στοιχεία FC, όπως για παράδειγμα Fog servers και αποθηκευτικές συσκευές, που παρέχουν την απαραίτητη υποδομή για την επικοινωνία των κόμβων Fog με τους τελικούς χρήστες. Οι τεχνολογίες που χρησιμοποιούνται για την επίτευξη της επικοινωνίας σε αυτό το επίπεδο είναι WiFi, Bluetooth, NFC (near-field communications), RFID (radio frequency identification), LTE (long-term evolution) και το 5G, που είναι συνδυασμός της τεχνολογίας LTE με νεότερες radio-access τεχνολογίες. Πρόκειται για τεχνολογίες επικοινωνίας με διαφορετικούς ρυθμούς μετάδοσης δεδομένων, συχνότητα λειτουργίας, χωρητικότητα καναλιών και μέγιστο εύρος μετάδοσης δεδομένων, όπου τα χαρακτηριστικά τους φαίνονται αναλυτικά.

Η τεχνολογία NFC έχει εύρος μετάδοσης και εύρος δεδομένων πολύ χαμηλό κάτι που την καθιστά κατάλληλη για ανταλλαγή εφαρμογών σε μικρό όγκο πληροφοριών, π.χ. ηλεκτρονικές πληρωμές και πιστοποίηση πρόσβασης σε υπηρεσίες. Η RFID είναι παρόμοια με τη προηγούμενη αλλά υποστηρίζει μόνο one-way επικοινωνία. Το Bluetooth είναι δυναμική τεχνολογία επικοινωνίας και ανταλλαγής δεδομένων ενώ το WiFi και LTE ή 5G είναι τεχνολογίες προσαρμόσιμες σε μεγάλες αποστάσεις και έχουν χωρητικότητα μεγάλου εύρους δεδομένων.

Σε συνδυασμό με την ανάπτυξη των ασύρματων τεχνολογιών μετάδοσης, τα πρωτόκολλα επικοινωνίας και δικτύωσης πρέπει να είναι επανασχεδιασμένα για την ενσωμάτωση τόσο των υπολογιστών όσο των υποδομών επικοινωνίας και την αποτελεσματική βελτίωση του υπολογισμού αποδοτικότητας και μετάδοσης δεδομένων [26].

Για παράδειγμα η ενσύρματη επικοινωνία μπορεί να βελτιωθεί με την υιοθέτηση της Long Reach Passive Optical Networks (LR-PON) τεχνολογίας [25]. Η τεχνολογία LR-PON παρέχει εξοικονόμηση κόστους με την χρήση οπτικών ινών σε γραφεία, κτίρια σπιτιών, πεζοδρόμια κλπ. έως και 100 km. Η εφαρμογή τους στα δίκτυα FC παρέχει υποστήριξη σε εφαρμογές που είναι ευαίσθητες σε εύρος ζώνης και πλατφόρμες, όπως έξυπνες βιομηχανίες και έξυπνο σπίτι. Κάθε τύπος επικοινωνίας (όπως αναφέρθηκαν παραπάνω), μπορεί να έχει διαφορετικά χαρακτηριστικά (π.χ. χωρητικότητα) και να χρησιμοποιεί διαφορετικές τεχνολογίες μετάδοσης.

Η αρχιτεκτονική Fog υποστηρίζει επίσης σύγχρονες τεχνολογίες επικοινωνίας όπως Network Function Virtualization (NFV) και Software Define Networking (SDN) [25]. Το NFV παρέχει μια νέα μέθοδο για τη δημιουργία, τη διαχείριση και την παροχή υπηρεσιών δικτύωσης που βασίζεται στην τεχνολογία της εικονικοποίησης. Πολλά βασικά στοιχεία του FC, όπως gateways και switches μπορούν να εικονοποιηθούν και να διατηρηθούν στους κόμβους Fog εξασφαλίζοντας ευελιξία, γρήγορη ανάπτυξη και παροχή νέων υπηρεσιών. Το SDN είναι μια νέα κεντρικοποιημένη αρχιτεκτονική που επιτρέπει στους χρήστες του να ορίζουν τους δικούς τους κανόνες δρομολόγησης. Η χρήση SDN σε περιβάλλον FC συμβάλει στην επίλυση προβλημάτων όπως υψηλός ρυθμός απώλειας πακέτων, συγκρούσεις, μη συνεπή συνδεσιμότητα, μείωση καθυστέρησης και βελτιστοποίηση στη χρήση πόρων.

3.3 Προσεγγίσεις υπολογισμού κόστους

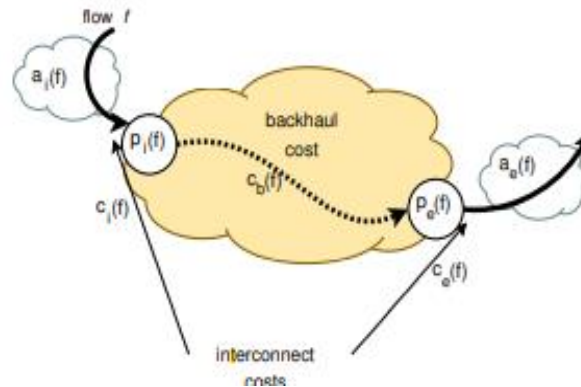
Σήμερα γίνονται αρκετές ερευνητικές προσπάθειες, διερευνώντας το θέμα του κόστους στα FC δίκτυα. Για το σκοπό αυτό πρότειναν διάφορα μοντέλα που βασίζονται στην αρχιτεκτονική αυτών των συστημάτων και λαμβάνουν υπόψη τα διάφορα στοιχεία που τα συνθέτουν. Η δικτύωση μεταξύ των επιπέδων επιτρέπει την επικοινωνία και τις όποιες αλληλοεπιδράσεις απαιτούν οι εφαρμογές για να λειτουργήσουν. Στη συνέχεια παρουσιάζονται τα πιο γνωστά μοντέλα υπολογισμού κόστους.

3.3.1 Μοντέλο κόστους δικτυακής κίνησης (IP traffic)

Το μοντέλο κόστους δικτυακής κίνησης [28], προτάθηκε για να υπολογίσει το κόστος ως αποτέλεσμα της κίνησης IP σε ένα δίκτυο. Το μοντέλο αυτό δεν αφορά περιβάλλοντα FC όμως, όπως αναφέρθηκε παραπάνω, μέρος της δικτυακής κίνησης που βασίζεται σε δίκτυα IP. Σύμφωνα με τους συγγραφείς της εργασίας το κόστος της δικτυακής κίνησης συντίθεται από δύο βασικές συνιστώσες:

- Το κόστος διασύνδεσης (interconnect) και
- Το κόστος *backhaul* (οπισθόζευξης),

όπως φαίνεται και στην Εικόνα 3.4.



Εικόνα 3.4: Ταξινόμηση του κόστους κυκλοφορίας για την ροή f . Μια ροή συγκεντρώνει την κίνηση από πολλές μεμονωμένες συνδέσεις, πηγή [28].

Το κόστος διασύνδεσης αφορά την κίνηση και τον όγκο των πακέτων στο δίκτυο, σε συγκεκριμένη γεωγραφική περιοχή. Το κόστος των δικτύων εξαρτάται από τη συμφωνία που έχει γίνει με τον πάροχο του κάθε δικτύου σχετικά, με τον όγκο των δεδομένων που διακινούνται στο δίκτυο και τις χρεώσεις που καταβάλλονται στο δίκτυο από τους παρόχους τους. Επίσης, βασικός παράγοντας διαμόρφωσης του κόστους είναι η γεωγραφική θέση του δικτύου διακίνησης, π.χ οι χρεώσεις στην Ασία είναι πολύ υψηλές σε σχέση με αυτές των ΗΠΑ.

Ένας ακόμη, βασικός παράγοντας διαμόρφωσης του κόστους διασύνδεσης είναι το κόστος των ports, που αφορά την προκαθορισμένη χρέωση εγκατάστασης των ports, και η τιμή τους διαμορφώνεται σύμφωνα με τη χωρητικότητα των ports. Εάν η προκαθορισμένη χωρητικότητα των ports ξεπεραστεί τότε η χρέωση θα αυξηθεί, καθώς θα πρέπει να προστεθούν επιπλέον ports στη συναλλαγή.

Το κόστος *backhaul* εξαρτάται από τρεις παράγοντες:

- Τα κυκλώματα(circuits),
- Τον εξοπλισμό για τις συνδέσεις (π.χ. routers, switches) και
- Το operational κόστος του δικτύου.

Το κόστος του κυκλώματος χωρίζεται σε δύο κατηγορίες: *Metro-range cost* και *Regional cost*.

- Το *metro-range* κόστος αφορά τις δαπάνες σύνδεσης στη μητροπολιτική περιοχή του δικτύου, το κόστος αυτό είναι αμελητέο για μεγάλα δίκτυα εκτός κι αν ένα δίκτυο εκμισθώνει τη τοπική συνδεσιμότητα του από άλλα δίκτυα.
- Το *regional cost* αφορά δαπάνες οι οποίες εξαρτώνται από τη γεωγραφική θέση που καλύπτει το δίκτυο και την απόσταση του περιφερειακού κυκλώματος, καθώς κάποιες περιοχές είναι πιο ακριβές από κάποιες άλλες.

Ο *εξοπλισμός για τις συνδέσεις* είναι το σταθερό κόστος και αφορά τα κεφαλαιουχικά κόστη (capital costs) δηλαδή την αγορά, τη συντήρηση και την αναβάθμιση των δρομολογητών και των switches.

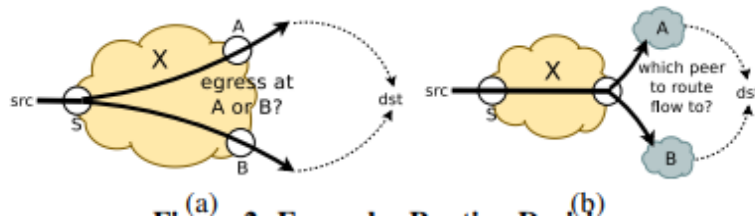
Το *λειτουργικό κόστος (operational cost)* είναι η μίσθωση των φορέων του δικτύου για την ψύξη και την κατανάλωση ισχύος και συνήθως, ενσωματώνεται στο σταθερό κόστος του δικτύου, το οποίο είναι το βασικό συστατικό των συνδέσεων υποδομής του.

Το προτεινόμενο μοντέλο κόστους συναθροίζει το *σταθερό κόστος (Fixed Costs)* του δικτύου και το κόστος χρήσης του δικτύου για διασύνδεση, που περιλαμβάνει το *κόστος χρήσης της σύνδεσης (Usage-based Cost)*, το *κόστος χρήσης των διασυνδέσεων (Usage-based Interconnect)* (στα δύο άκρα της σύνδεσης) και το *κόστος χρήσης του δικτύου backhaul (Usage-based Backhaul Cost)*.

- Το *σταθερό κόστος* ορίζεται από την τοπολογία του δικτύου (backbone) και τη διασύνδεση με τα γειτονικά δίκτυα. Πιο συγκεκριμένα το σταθερό κόστος τοπολογίας απαριθμεί το σταθερό κόστος κάθε δυνατής διαδρομής μεταξύ δύο *PoP*¹.
- Το *κόστος χρήσης της σύνδεσης* αφορά το κόστος χρήσης του συνδέσμου που υπολογίζεται για κάθε ροή εισερχόμενη στο δίκτυο και αφορά, το κόστος διάσχισης του δικτύου (στο backbone σύνδεσμο) και το κόστος διασύνδεσης στα άκρα του συνδέσμου (εισόδου εξόδου). Το κόστος αυτό εξαρτάται από τον όγκο των δεδομένων ή το εύρος και τη διαδρομή που θα επιλέξει μια ροή *f* στο δίκτυο.
- Το *κόστος χρήσης των διασυνδέσεων* αφορά το κόστος που αντιστοιχεί στα δύο άκρα της σύνδεσης, τα σημεία ανταλλαγής δεδομένων. Το κόστος στα δύο σημεία διασύνδεσης είναι συμμετρικό. Εξαρτάται από τον όγκο δεδομένων στη ροή και κοστολογείται ανά μονάδα όγκου για κάθε ροή που διέρχεται από το δίκτυο. Η σύνηθες τιμή χρέωσης του είναι μεταξύ 0.4 και

¹ Ένα point of presence (PoP) είναι ένα σημείο οριοθέτησης, σημείο πρόσβασης ή φυσική τοποθεσία στην οποία δύο ή περισσότερα δίκτυα ή συσκευές επικοινωνίας μοιράζονται μια σύνδεση.

0.75, τιμή που καθορίζεται από τον πάροχο του δικτύου, εξαρτώμενη από την προσφορά και την ζήτηση.



Εικόνα 3.5: Παράδειγμα δρομολόγησης, πηγή [28].

- Το κόστος χρήσης του δικτύου οπισθόζευξης αφορά το δίκτυο backhaul και εξαρτάται από την γεωγραφική θέση των PoP και την μεταξύ τους απόσταση.

Παράδειγμα

Ένα παράδειγμα υπολογισμού κόστους από την οπτική γωνιά του δικτύου φαίνεται στην Εικόνα 3.5. Στην Εικόνα 3.5a το δίκτυο “X” δρομολογεί τη ροή από το PoP S προς το PoP A ή B. Αυτό συμβαίνει όταν το δίκτυο επικοινωνεί με δύο συγκεκριμένους γείτονες σε δύο τοποθεσίες A και B και για τη δρομολόγηση της ροής μπορεί να επιλέξει οποιαδήποτε διαδρομή PoP χωρίς να αλλάζει το κόστος.

Στην Εικόνα 3.5b το δίκτυο μπορεί να δρομολογήσει τη ροή σε μία συγκεκριμένη διαδρομή μέσω των γειτονικών δικτύων. Ο διαχειριστής του δικτύου έχει να επιλέξει μεταξύ των διαδρομών A ή B δηλαδή να επιλέξει τη διαδρομή με το μικρότερο κόστος, για τη δρομολόγηση μιας συγκεκριμένης ροής. Λαμβάνοντας υπόψη το κόστος μεταφοράς, τη τρέχουσα δρομολόγηση, τις πληροφορίες τοπολογίας του δικτύου και υπολογίζοντας το συνολικό κόστος (διασύνδεσης και οπισθόζευξης) δρομολογεί τη ροή επιλέγοντας τη διαδρομή A ή B με το μικρότερο κόστος. Η επιλεγούσα διαδρομή δρομολόγησης πρέπει να ικανοποιεί τους περιορισμούς της χωρητικότητας και τη δυναμικότητα απόδοσης οπισθόζευξης της σύνδεσης του δικτύου. Ο διαχειριστής επίσης καταγράφει τους περιορισμούς της χωρητικότητας διασύνδεσης στο δίκτυο και τη δυναμική απόδοση κάθε δρομολόγησης ώστε να αποφευχθούν περιπτώσεις επαναδρομολόγησης της ροής στην ίδια διαδρομή όπου έχει προκληθεί συμφόρηση ή έχει επιβληθεί ποινή υψηλής απόδοσης.

Σύμφωνα με τους συγγραφείς της εργασίας [28], το κόστος επικοινωνίας στο δίκτυο μπορεί να μειωθεί με:

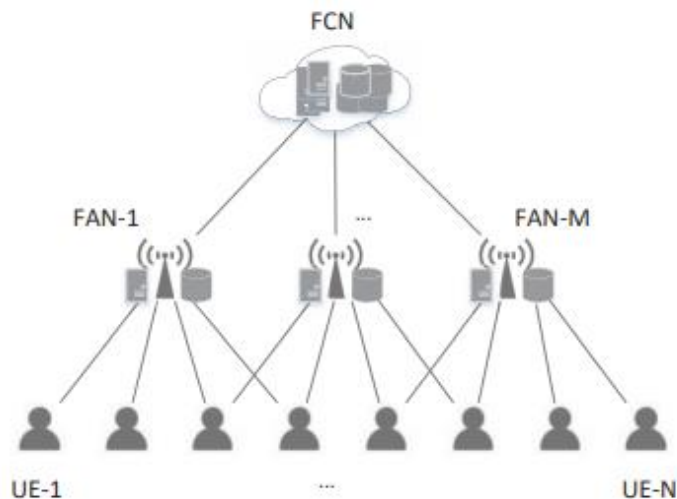
- Τη σωστή επιλογή δρομολόγησης της ροής (ingress-egress flows).

- Την ισότιμη σύνδεση (peering) με έναν γείτονα.

3.3.2 Πολυεπίπεδο μοντέλο κόστους

Οι Liu κ.ά. [29], θεωρούν ότι το δίκτυο FC αποτελείται από επίπεδα που τοποθετούνται το ένα μετά το άλλο και περιλαμβάνουν: έναν κόμβο ελέγχου (Fog Control Node, FCN), κόμβους πρόσβασης Fog (Fog Access Nodes, FANs) και τον εξοπλισμό των χρηστών (user equipments, UE). Για παράδειγμα, όπως φαίνεται και στην Εικόνα 3.6, το δίκτυο FC με τα πολλαπλά επίπεδα αποτελείται από: 1 FCN, M FANs και N UEs. Το FCN υποστηρίζεται από έναν πάροχο τηλεπικοινωνιών, ο οποίος παρέχει τον εξοπλισμό και τις υπηρεσίες στους N συνδρομητές (N UEs), ενώ οι FANs έχουν πόρους τους οποίους κατανέμουν στους χρήστες του δικτύου. Η πολιτική πληρωμής που ακολουθείται είναι δυναμική (Service billing pay per service), όπου το FCN χρεώνεται από τον πάροχό του ανάλογα με τον όγκο των υπηρεσιών και τον εξοπλισμό που παρέχει στους χρήστες (Εικόνα 3.6).

Το μοντέλο υπολογίζει το κόστος του δικτύου με παραμέτρους την καθυστέρηση (Service Delay), την εξυπηρέτηση των αιτούμενων υπηρεσιών των χρηστών, τη διαμόρφωση της δυναμικής πληρωμής (Liner in bed dynamic payment scheme) - δηλαδή η πληρωμή εξαρτάται και διαμορφώνεται σύμφωνα με τη ζήτηση, τη χρήση του εξοπλισμού από τους χρήστες (Service billing pay per use) και τον προαπαιτούμενο προγραμματισμένο αριθμό εξυπηρετούμενων χρηστών έχοντας ως απώτερο στόχο την ελαχιστοποίηση του κόστους του δικτύου.



Εικόνα 3.6: Απεικόνιση πολυεπίπεδου μοντέλου κόστους πηγή[29].

Οι He κ.ά. [30], στην πρότασή τους για μια πολυεπίπεδη ιεραρχική δομή του FC εκτιμούν το κόστος επικοινωνίας ως συνάρτηση του όγκου των δεδομένων, τη χρέωση ανά μονάδα δεδομένων (byte) και

τον τύπο του υπολογιστικού περιβάλλοντος (π.χ. Cloud, FC). Το κόστος επικοινωνίας λαμβάνεται υπόψη κατά τη σχεδίαση του σχήματος ποιότητας υπηρεσιών.

3.3.3 Αρχιτεκτονική τριών επιπέδων

Κατά τους Battula κ.ά. [8], το κόστος επικοινωνίας είναι μια συνιστώσα του συνολικού κόστους σε ένα σύστημα FC και εξετάζεται ανά επίπεδο της αρχιτεκτονικής Fog. Το κόστος προέρχεται από:

- a) Τη μετακίνηση των δεδομένων από και προς τις συσκευές Fog για την επεξεργασία τους από αυτές. Το κόστος επικοινωνίας δεν επιβαρύνεται από το κόστος του λογισμικού ή το κόστος migration (μετεγκατάστασης) και εξαρτάται από τον συνολικό αριθμό των ληφθέντων μηνυμάτων από τις συσκευές Fog, το μέγεθος του κάθε μηνύματος και το κόστος ανά μονάδα μηνύματος.
- b) Την αποστολή και λήψη δεδομένων από μια συσκευή Fog στο Cloud για τη διενέργεια υπολογισμών, όταν αυτό χρειασθεί. Το κόστος αυτό εξαρτάται από το εύρος ζώνης της σύνδεσης δικτύου με το Cloud και περιλαμβάνει την καθυστέρηση επεξεργασίας των δεδομένων στο Cloud.
- c) Το λειτουργικό κόστος του δικτύου.

Στο μοντέλο τους οι Pham κ.ά. [31], ορίζουν το οικονομικό κόστος επικοινωνίας εξετάζοντας αν:

- Οι υπολογισμοί γίνονται σε κόμβο Cloud: το κόστος αυτό αφορά τη μεταφορά των δεδομένων από άλλους κόμβους Cloud στον κόμβο Cloud όπου γίνεται η επεξεργασία των δεδομένων,
- Οι υπολογισμοί γίνονται σε κόμβο Fog: το κόστος αυτό αφορά τη μεταφορά των δεδομένων από άλλους κόμβους Cloud στον τοπικό κόμβο Fog όπου γίνεται η επεξεργασία των δεδομένων

Για το υπολογισμό του κόστους γίνεται χρέωση ανά μονάδα δεδομένων που μετακινούνται.

3.3.4 Συσχέτιση απόδοσης και κόστους

Οι Ali και Pasha κ.ά. [15], συσχετίζουν το κόστος με την απόδοση σε ένα σύστημα FC. Προτείνουν ένα μαθηματικό μοντέλο που το κόστος απαρτίζεται από τρεις συνιστώσες: την κατανάλωση ενέργειας, την καθυστέρηση εκτέλεσης μιας διεργασίας, και τη χρήση του δικτύου (network usage). Οι δύο τελευταίες συνιστώσες αφορούν το δίκτυο.

Πιο συγκεκριμένα, η καθυστέρηση είναι το σύνολο των καθυστερήσεων από την αναμονή στην ουρά, τη διάδοση (propagation) και τη μετάδοση των πακέτων. Η χρήση του δικτύου αναφέρεται στο φόρτο του δικτύου και στον όγκο των δεδομένων που αποστέλλονται από και προς το δίκτυο λόγω απαίτησης διαμοίρασης δεδομένων μεταξύ των εφαρμογών, διακομιστών, συσκευών και χρηστών

δικτύου. Για τη διαμοίραση δεδομένων μεταξύ των οντοτήτων χρησιμοποιείται το δίκτυο και απασχολούνται οι συνδέσεις του καθόλη τη διάρκεια της ανταλλαγής δεδομένων. Αυτό έχει επίπτωση στο φόρτο του δικτύου, καθώς κατά τη μεταφορά οι συνδέσεις είναι απασχολημένες μέχρι το πακέτο να φτάσει με επιτυχία στον προορισμό του.

3.4 Συμπεράσματα

Το FC διασυνδέει γεωγραφικά κατανεμημένες συσκευές, με διαφορετικούς τρόπους διασύνδεσης και αξιοποιεί την υπάρχουσα δικτυακή υποδομή. Κάθε τρόπος διασύνδεσης και επικοινωνίας έχει διαφορετικά χαρακτηριστικά και δυνατότητες, που επηρεάζουν το κόστος επικοινωνίας (Πίνακας 3.1).

Το κόστος επικοινωνίας σε ένα περιβάλλον FC συνδέεται με τη χωρητικότητα ή το εύρος ζώνης και τις αντίστοιχες δαπάνες που συνδέονται με τη χρήση του δικτύου [8][22][23][28][15]. Άλλες ερευνητικές εργασίες συνδέουν το κόστος με την καθυστέρηση (είτε delay είτε latency) [8][29][15]. Μερικοί ερευνητές λαμβάνουν υπόψη και τα έξοδα αγοράς συσκευών για τη διαμοίραση των δεδομένων και εντάσσουν αυτή τη δαπάνη στο κόστος δικτύου [22][28][29]88. Κάποιοι άλλοι υπολογίζουν την ταχύτητα [22][23] και την κατανάλωση ενέργειας [28][15]. Τα διαθέσιμα μοντέλα υπολογισμού κόστους εστιάζουν στις παραμέτρους που επηρεάζουν το κόστος δικτύου και επιχειρούν να βρουν λύσεις που να ελαχιστοποιούν το κόστος. Στον πίνακα 3.1 περιγράφονται αναλυτικά οι παράμετροι του κόστους όπως προέκυψαν από την μελέτη της βιβλιογραφίας. Από αυτές διακρίνουμε το συνολικό αριθμό των μεταδιδόμενων μηνυμάτων, το μέγεθος του κάθε μηνύματος, την αρχιτεκτονική Fog και την τοπολογία του δικτύου.

Από το σύνολο των μοντέλων, που παρουσιάζονται συνοπτικά παραπάνω, συμπεραίνουμε ότι το FC είναι μία νέα τεχνολογία, με δυνατότητα πολλών και ποικίλων υλοποιήσεων που βασίζεται σε διάφορους δικτυακούς πόρους. Αυτό έχει ως συνέπεια, ο υπολογισμός του κόστους να είναι δύσκολη και πολύπλοκη διαδικασία, λόγω του πολλών και διαφορετικών συνιστωσών που μετέχουν στην υλοποίησή του. Είναι όμως βέβαιο πως το κόστος δικτύου εξαρτάται από τις απαιτήσεις χρήσης, αλλά και από την διαθεσιμότητα (ή προσφορά) των δικτυακών πόρων.

Πίνακας 3.1: Παράμετροι υπολογισμού κόστους επικοινωνίας.

Αναφορές cost factor	Όγκος δεδομένων	Χρόνος διάδοσης πακέτων	Χωρητικότητα Καναλιού	Αριθμός Χρηστών	Χρήση δικτύου (time, data size)	Διαμόρφωση πληροφορίας	Η απόσταση μεταξύ ingress and egress PoP	Αριθμός μηνυμάτων	Καθυστέρηση (queuing, propagation and transmission)	Ταχύτητα	Διατάξεις σύνδεσης (switches, routers)	Κατανάλωση ενέργειας
Battula 2019, [8]	✓		✓			✓		✓	✓			
Motiwala 2019, [28]	✓	✓	✓			✓	✓				✓	✓
Liu 2019, [16]	✓			✓	✓	✓			✓		✓	
Ali 2020, [15]		✓	✓	✓	✓				✓			✓
He 2012, [10]	✓					✓						
Pham 2017, [31]	✓						✓					
Wang 2021, [23]	✓		✓							✓		
Yousefpour 2018, [22]		✓	✓			✓				✓	✓	

4 Κόστος υπολογισμού (computation cost)

Ο υπολογισμός, όπως η αποθήκευση, η δικτύωση, η λήψη αποφάσεων και η διαχείριση δεδομένων συμβαίνουν καθώς τα δεδομένα μετακινούνται στο Cloud από τις συσκευές IoT, με τη βοήθεια του FC [22]. Ο υπολογισμός αναλύει, φιλτράρει, περικόπτει και ανακατασκευάζει τα συλλεγόμενα δεδομένα, όταν αυτό είναι απαραίτητο. Οι διαδικασίες αυτές γίνονται σε πραγματικό χρόνο και ειδικά εάν απαιτείται οι παραγόμενες πληροφορίες να γίνουν άμεσα διαθέσιμες. Μετά από τον υπολογισμό και την επεξεργασία τα δεδομένα αποθηκεύονται τοπικά ή στέλνονται στο Cloud για μόνιμη αποθήκευση, έχοντας ληφθεί οι ανάλογες αποφάσεις, και λαμβάνοντας υπόψη τις απαιτήσεις των χρηστών.

Το κόστος υπολογισμού περιλαμβάνει το κόστος επεξεργασίας/υπολογισμού, το κόστος εκχώρησης πόρων (resource allocation), το κόστος migration (μετεγκατάστασης) εργασιών από έναν κόμβο Fog σε έναν άλλο και το κόστος offloading. Το συνολικό κόστος εκτέλεσης (execution cost) υπολογίζεται με βάση το χρόνο ολοκλήρωσης της εργασίας και το κόστος χρήσης ανά μονάδα χρόνου [14].

Το FC παρέχει τη δυνατότητα συγκέντρωσης βασικών υπηρεσιών (υπολογισμό, δικτύωση, αποθήκευση) μεγιστοποιώντας τη χρήση των αδρανών υπολογιστικών και αποθηκευτικών πόρων για να ολοκληρωθούν δραστηριότητες με υψηλές απαιτήσεις σε πόρους. Η εκχώρηση πόρων για υπολογισμό, επεξεργασία και αποθήκευση κατά την εκτέλεση εφαρμογών ή εργασιών στους κόμβους Fog απαιτεί ένα σύνολο δαπανών περιλαμβάνοντας τις χρεώσεις ή τις δαπάνες που προκύπτουν από την εκμετάλλευση των πόρων. Ο μηχανισμός offloading βελτιώνει την απόδοση του συστήματος όταν μεταξύ των συσκευών είναι διαθέσιμες υψηλού εύρους ζώνης δικτυακές συνδέσεις.

Στη συνέχεια παρουσιάζονται βασικά μοντέλα κόστους επεξεργασίας/υπολογισμού.

4.1 Κόστος επεξεργασίας

4.1.1 Εκχώρηση πόρων

Ορισμένοι ερευνητές, που μελετούν την εκχώρηση πόρων σε ένα σύστημα FC, εξετάζουν τον παράγοντα του κόστους επεξεργασίας, ενώ άλλοι δεν το συμπεριλαμβάνουν στις μελέτες τους, ή το αναφέρουν χωρίς όμως να προτείνουν συγκεκριμένα μοντέλα υπολογισμού [8].

Οι Battula κ.ά. [8], προτείνουν ένα μοντέλο υπολογισμού κόστους σε μικροεπίπεδο όπου το κόστος επεξεργασίας της εφαρμογής εξαρτάται από τον αριθμό των ενεργειών που ενεργοποιήθηκαν και εκτελέστηκαν, το καθοριζόμενο επίπεδο ποιότητας υπηρεσιών (QoS), την αποστολή μηνυμάτων, τους

διαθέσιμους πόρους σε κάθε συσκευή και το χρόνο που χρειάστηκε για την εξυπηρέτηση του αιτήματος του χρήστη διατηρώντας παράλληλα την απαιτούμενη ποιότητα υπηρεσιών.

Ο χρήστης σε κάθε περίπτωση αιτείται την εκτέλεση της εφαρμογής ή το σύνολο των πόρων που απαιτούνται για την εκτέλεση της εφαρμογής (π.χ. container, VMs). Αν ο χρήστης ζητήσει συγκεκριμένους πόρους για την εκτέλεση της εφαρμογής, το κόστος επεξεργασίας εξαρτάται από το ποσοστό χρήσης των πόρων (π.χ. χρήσης εικονικής CPU) που δεσμεύει και χρησιμοποιεί, την παρεχόμενη ποιότητα υπηρεσιών και το χρόνο χρήσης.

Οι Yu κ.ά. [32], αποσκοπούν στη βέλτιστη εκχώρηση των πόρων στο FC και στο Cloud υπολογίζοντας τη μέγιστη ανεκτή καθυστέρηση μιας εφαρμογής στη συσκευή Fog, ορίζοντας την εξισορρόπηση των εργασιών (workload balance) στο FC, τη μέγιστη χωρητικότητα των συνδέσεων στο κέντρο δεδομένων Cloud και τον υπολογισμό της μέγιστης ανεκτής καθυστέρησης μιας εφαρμογής στο κέντρο δεδομένων του Cloud.

Όταν η συσκευή Fog, λάβει αίτημα από τους χρήστες (UR) του Cloud, θα επεξεργαστεί ένα μέρος τοπικά και θα διαμοιράσει τις υπόλοιπες αιτήσεις στο Cloud. Ως κόστος υπολογίζεται η καθυστέρηση, που πρέπει να είναι μικρότερη από τη μέγιστη ανεκτή καθυστέρηση της εφαρμογής, και εξαρτάται από το αιτούμενο από το χρήστη εύρος, το εύρος της εφαρμογής που υποστηρίζεται από την συσκευή Fog και το μέγεθος της εφαρμογής [32].

4.1.2 Προγραμματισμός εργασιών

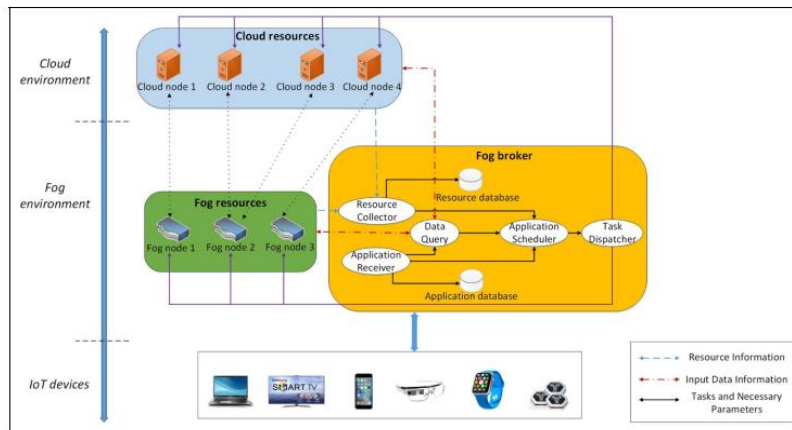
Οι Pham κ.ά. [31], εξετάζουν τον προγραμματισμό εργασιών (task scheduling) σε υποδομές FC/Cloud και αναλύουν ένα μοντέλο FC με σκοπό την επίλυση προβλημάτων, όπως:

- Ελαχιστοποίηση της διάρκειας εκτέλεσης εφαρμογών με υψηλής κλίμακας υπολογιστικών απαιτήσεων,
- Ικανοποίηση των απαιτήσεων σε ποιότητα υπηρεσιών των εφαρμογών στις καταναμημένες πλατφόρμες του Cloud και του FC και
- Βελτιστοποίηση στον προγραμματισμό των εργασιών.

Η αρχιτεκτονική Fog που περιγράφει έχει 3 επίπεδα (

Εικόνα 4.1) όπου το FC περιβάλλον είναι στο μεσαίο επίπεδο. Σε αυτό το επίπεδο η συσκευή *Fog broker* λαμβάνει τα αιτήματα των χρηστών και προγραμματίζει τις διάφορες εργασίες. Συγκεκριμένα ο Fog broker λαμβάνει τα αιτήματα των χρηστών, διαχειρίζεται τους διαθέσιμους πόρους στους Cloud και κόμβους Fog (π.χ. χωρητικότητα επεξεργασίας, διαδικτυακό εύρος), συνδυάζοντας το κόστος επεξεργασίας και επικοινωνίας με τα αποτελέσματα της κίνησης των δεδομένων στους διάφορους

κόμβους. Ακόμα παίρνει αποφάσεις για τον κατάλληλο προγραμματισμό των εργασιών προς τον κόμβο που θα κατευθυνθεί η κάθε ροή.

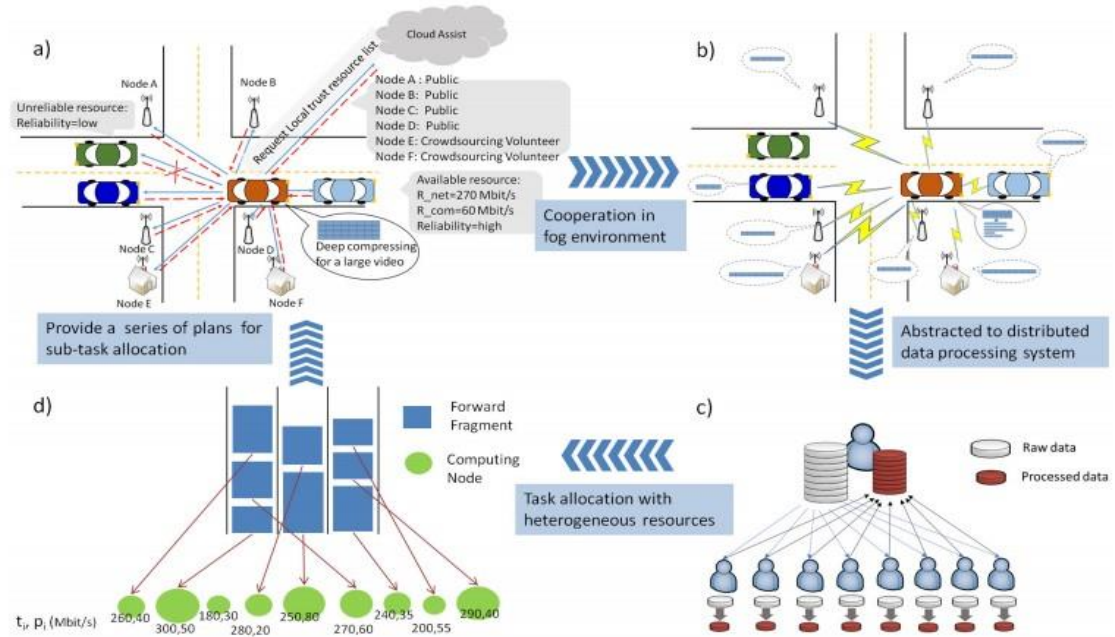


Εικόνα 4.1: Αρχιτεκτονική Συστήματος, πηγή [31].

Σύμφωνα με τους συγγραφείς το κόστος έχει δύο βασικές συνιστώσες: το κόστος δικτύου και το κόστος υπολογισμού. Το κόστος υπολογισμού εξαρτάται από τον τύπο του κόμβου (Fog, Cloud) και υπολογίζεται ανά μονάδα χρόνου. Στην περίπτωση του κόμβου Cloud, το κόστος περιλαμβάνει το κόστος επεξεργασίας, αποθήκευσης, μνήμης και το κόστος επικοινωνίας των δεδομένων στον κόμβο Cloud. Στην περίπτωση του κόμβου Fog, προσμετράτε μόνο το κόστος για τη μεταφορά των δεδομένων στον τοπικό κόμβο Fog από τον Cloud.

Οι Rah κ.ά. [33], στο μοντέλο τους χρησιμοποιούν διαμεσολαβητές (brokers), εφαρμογές και συσκευές Fog τα οποία έχουν σχέσεις αλληλεξάρτησης και καλούν τον χρονοπρογραμματιστή για την εκχώρηση πόρων όταν και όπου κρίνεται απαραίτητο. Το κόστος που προκύπτει υπολογίζεται σε συνάρτηση με το κόστος εκτέλεσης, το χρόνο εκτέλεσης, τον ρυθμό των MIPS και τον συνολικό αριθμό MIPS του host.

Στο σενάριο τους οι Liu κ.ά. [7], εξετάζουν ένα μοντέλο κατανομής πόρων, προγραμματισμού εργασιών και βελτιστοποίησης ή μείωση της καθυστέρησης μετάδοσης, σε μία εφαρμογή δικτύωσης αυτοκινήτων (vehicular adhoc networks application, VANET), η οποία αναλύει τον τρόπο κατανομής, επεξεργασίας και μετάδοσης των εργασιών σε περιβάλλον FC.



Εικόνα 4.2: Εφαρμογή μοντέλων κατανομής πόρων, προγραμματισμού εργασιών και βελτιστοποίησης ή μείωση της καθυστέρησης μετάδοσης μέσω 3 MU-MIMO καναλιών, πηγή [7].

Ένας εκκινητής διεργασιών (task initiator) συλλέγει τα σχόλια και επιλέγει τους εφικτούς κόμβους του συνασπισμού (coalition). Στη συνέχεια, το βίντεο χωρίζεται σε τμήματα, σύμφωνα με μια συγκεκριμένη στρατηγική και διασκορπίζεται στους κατανεμημένους συνεργάτες, όπως την Εικόνα 4.2 (b).

Η στρατηγική που ακολουθείται για την διεκπεραίωση του αιτήματος περιλαμβάνει:

1. Τον διαχωρισμό της εργασίας σε δευτερεύουσες εργασίες και
2. Τον προγραμματισμό των δευτερευουσών εργασιών ώστε να ελαχιστοποιηθεί η καθυστέρηση σε όλη τη διαδικασία.

Η διαδικασία αυτή είναι η κατανεμημένη επεξεργασία δεδομένων όπως φαίνεται στην Εικόνα 4.2(γ). Αποτελείται από τρία μέρη: διανομή, επεξεργασία και επιστροφή. Πρόκειται για κατανομή πόρων, με επίλυση του προβλήματος προγραμματισμού εργασιών (Εικόνα 4.2 (δ)).

Η εφαρμογή κατάλληλου αλγορίθμου (Multi user - Multiple Input Multiple Output (MU-MIMO)) θέτει σε λειτουργία στο framework καθώς παρέχει χωριστά κανάλια με πλήρη εύρος δικτύου, κάτι που δίνει μεγάλο πλεονέκτημα στο ασύρματο κατανεμημένο δίκτυο για την επίτευξη της αποτελεσματικής μοντελοποίησης και βελτιστοποίησης της στρατηγικής μειώνοντας τον λανθάνοντα χρόνο και ελαχιστοποιώντας την καθυστέρηση.

Η βελτιστοποίηση αυτή παρέχει μια σειρά σχεδίων για το πως η αρχική εργασία θα τμηματοποιηθεί, που θα αποσταλούν τα κομμάτια και πως θα τοποθετηθούν στη σειρά.

Στόχος αυτής της διαδικασίας είναι η ελαχιστοποίηση της καθυστέρησης με τον προσδιορισμό ενός αποτελεσματικού σχεδίου κατανομής ανά το μέγεθος των εργασιών (chunk size), τη σειρά παράδοσης τους, την επιλογή των κατάλληλων κόμβων και καναλιών.

4.1.3 Κόστος Migration

Η κινητικότητα της συσκευής είτε σε επίπεδο IoT είτε σε επίπεδο Fog είναι ένα πολύ πιθανό ενδεχόμενο. Σε επίπεδο Fog, κινητές μπορεί να είναι όχι μόνο οι τελικές συσκευές, αλλά και υπολογιστικές και αποθηκευτικές συσκευές που τοποθετούνται στο επίπεδο αυτό. Αυτό δημιουργεί μια σειρά από ζητήματα, όπως η διαχείριση κινητών συσκευών σε διαφορετικά επίπεδα και ο συγχρονισμός τους [34]. Η διανομή εργασιών, η δημιουργία αντιγράφων και το migration είναι ενέργειες που μπορούν να δώσουν λύση και να διασφαλίσουν τη διαθεσιμότητα των πόρων και την επιτυχή ολοκλήρωση των εργασιών. Το migration θεωρείται ως μια τεχνική διαχείρισης πόρων και έχει απασχολήσει τους ερευνητές στην περιοχή του FC στα πλαίσια του offloading και της αναδιανομής φορτίου [35]. Η λήψη απόφασης για το migration γίνεται αφού εκτιμηθούν δυο διαφορετικά κόστη: το κόστος migration και το κόστος μετάδοσης. Το κόστος migration οφείλεται στην μετακίνηση μιας υπηρεσίας σε μια νέα τοποθεσία και εξαρτάται από το μέγεθος της υπηρεσίας, το κόστος δέσμευσης/αποδέσμευσης της υπηρεσίας και το εύρος ζώνης που καταναλώνεται για το migration της υπηρεσίας. Το κόστος μετάδοσης αφορά το κόστος του δικτύου backhaul για την πρόσβαση στην υπηρεσία εφόσον δεν γίνει migration της υπηρεσίας. Βέβαια πρέπει να ληφθεί υπόψη και η αύξηση της καθυστέρησης που συνεπάγεται αυτή η επιλογή.

Οι Rejiba κ.ά. [35], στην αναφορά τους, διερευνούν αναλυτικά το πρόβλημα του migration και τις προτεινόμενες μεθόδους για τη βελτιστοποίηση του σε σχέση με το κόστος. Οι παράγοντες που λαμβάνονται υπόψη για τη βελτιστοποίηση είναι το κόστος migration, το κόστος μετάδοσης δεδομένων, το κόστος δημιουργίας αντιγράφων, η καθυστέρηση στο δίκτυο κλπ.

Οι Battula κ.ά. [8], ορίζουν ως κόστος migration το κόστος που θα πρέπει να καταβληθεί από έναν κόμβο Fog σε έναν άλλο για την συνεισφορά του στις υπηρεσίες σε περίπτωση migration. Το κόστος αυτό εξαρτάται από το συνολικό χρόνο εκτέλεσης των εργασιών migration και το κόστος ανά μονάδα επεξεργασίας.

4.1.4 Κόστος offloading

Η τεχνική offloading διευκολύνει τις τελικές συσκευές με περιορισμένους πόρους με την αποστολή των υπολογιστικών εργασιών σε συσκευές με διαθέσιμους ή σε αδράνεια πόρους για την εκτέλεση τους. Σε αυτή την περίπτωση υπάρχουν ζητήματα προς διευθέτηση όπως η κατανομή των πόρων στους διαθέσιμους κόμβους Fog, υπολογίζοντας παραμέτρους όπως η μέγιστη καθυστέρηση αναμονής των

εργασιών, ο χρονικός περιορισμός στη διανομή μιας υπηρεσίας, η ενεργοποίηση δευτερευόντων βοηθητικών προγραμμάτων υπολογισμού των διεργασιών και ο χρόνος επικοινωνίας για τη μετακίνηση των δεδομένων από κόμβο σε κόμβο.

Οι Wang κ.ά. [23], στους μοντέλο τους υπολογίζουν το συνολικό κόστος του offloading ως το άθροισμα από τα κόστη επεξεργασίας και των λαθών. Ως κόστος offloading θεωρούν το κόστος επικοινωνίας κατά την μετακίνηση των δεδομένων από τον κόμβο i στον κόμβο j , σε συγκεκριμένο χρόνο t . Το κόστος λαθών ποσοτικοποιεί την επίδραση των λαθών που συμβαίνουν λόγω της μετακίνησης. Το κόστος γενικά, εξαρτάται από τον όγκο των δεδομένων, το κόστος συνδεσιμότητας και το κόστος επεξεργασίας.

4.1.5 Μοντέλο υπολογισμού κόστους σε κινητές συσκευές

Στο μοντέλο των Mao et al. [26], υπολογίζεται το κόστος υπολογισμού των κινητών συσκευών και αναλύεται η μεθοδολογία αξιολόγησης της υπολογιστικής απόδοσης. Στο μοντέλο αυτό, το κόστος συνδέεται με την απόδοση και εξαρτάται από τα χαρακτηριστικά της CPU της κινητής συσκευής. Η απόδοση της ελέγχεται από τη συχνότητα της CPU και την κλίμακα τάσης (scaling voltage) οι οποίες επιτρέπουν την άνοδο ή τη μείωση της συχνότητας της CPU επιδρώντας στην αύξηση ή την μείωση της κατανάλωσης ενέργειας αντίστοιχα.

Η καθυστέρηση εκτέλεσης και η κατανάλωση ενέργειας ορίζουν εάν η κινητή συσκευή είναι ικανή να ολοκληρώσει την εργασία υπολογισμού στην προκαθορισμένη προθεσμία ολοκλήρωσης της, ή η κατανάλωση ενέργειας που απαιτείται για την εκτέλεση της διεργασίας είναι τόσο υψηλή όπου η ενσωματωμένη μπαταρία θα εξαντληθεί γρηγορότερα. Σε αυτές τις δύο περιπτώσεις απαιτείται το offloading και η συνέχεια εκτέλεσης της διεργασίας σε έναν mobile edge computing (MEC) διακομιστή [26].

4.2 Ασφάλεια

Τα συστήματα ασφάλειας που είναι εγκατεστημένα στο FC παρέχουν ολοκληρωμένη, αποτελεσματική, αξιόπιστη και εφαρμόσιμη λύση ασφάλειας, γι' αυτό γίνεται αναφορά στην επικοινωνία και στο computation. Σημαντικό για τον σχεδιαστή του FC είναι να αντιλαμβάνεται πόσο σημαντικά είναι τα δεδομένα και να καλεί τα μέτρα ασφαλείας που είναι απαραίτητα σε κάθε περίπτωση, χωρίς να μειώνεται η απόδοση του συστήματος Fog.

Τα οχτώ ζητήματα που σχετίζονται με την ασφάλεια και οι λύσεις τους στο FC είναι οι παρακάτω [36]:

- *Κρυπτογράφηση (Encryption) δεδομένων*

Στο FC αποκρυπτογραφούνται οι ευαίσθητες πληροφορίες, όπως η ταυτότητα των χρηστών στις κινητές συσκευές, τα δεδομένα των ασθενών στα συστήματα υγείας, τα δεδομένα της κρυφής μνήμης κ.α. Ο AES αλγόριθμος με 256-bit μέγεθος κλειδιού χρησιμοποιείται για την διασφάλιση της ιδιωτικότητας, ενώ το Secure Socket Layer (SSL) πρωτόκολλο εγκαθιστά ασφαλή επικοινωνία μεταξύ server και client.

➤ *Αποτροπή επιθέσεων στην προσωρινή μνήμη*

Στην προσωρινή μνήμη αποθηκεύονται πληροφορίες οι οποίες χρησιμοποιούνται συχνά και περιέχουν προσωπικές πληροφορίες του χρήστη, όπου για την προστασία τους σχεδιάζονται λύσεις όπως η μνήμη Partition Locked και η μνήμη Random Permutation που μπορούν να αποκλείουν ενδεχόμενες επιθέσεις. Ο μηχανισμός αποτροπής των τροποποιήσεων σε μικρό μέγεθος πληροφοριών είναι η διατήρηση τους στον κόμβο Fog για συγκεκριμένο χρονικό διάστημα.

➤ *Παρακολούθηση δικτύου (Network monitoring)*

Κάθε πλατφόρμα Fog περιέχει μηχανισμούς ελέγχου της κίνησης του δικτύου σε καταστάσεις ομαλής λειτουργίας αλλά και σε περιπτώσεις υπερφόρτωσης, ώστε να αποφεύγεται οποιαδήποτε κακόβουλη δραστηριότητα πριν προκύψει πραγματική ζημιά. Τα Artificial Neural Networks (ANNs) σε συνδυασμό με τους κανόνες ανίχνευσης απειλών και την εφαρμογή των Virtual Private Networks (VPNs) προστατεύουν από εξωτερικές επιθέσεις στο δίκτυο, καθώς μεγάλος αριθμός ετερογενών συσκευών (IoT) εκπέμπουν και επεξεργάζονται δεδομένα σε πολλαπλά επίπεδα (hypervisor, operating system και apps).

➤ *(Zero day) Malware protection*

Οι malware επιθέσεις είναι αυτές που προκαλούνται από ιούς, trojan, rootkit, spyware και worms δημιουργώντας σοβαρές ζημιές και μειώνοντας την απόδοση του δικτύου, οι οποίες είναι αναπόφευκτες καθώς είναι συνεχής η ζήτηση σε δικτύωση, υπολογισμό και πόρους. Η χρήση της συσκευής ανίχνευσης malware αποτελεί ιδανική λύση προστασίας καθώς η απαίτηση της σε πόρους Fog είναι μηδαμινή και σε συνδυασμό με το εργαλείο bareCloud μπορεί να εντοπιστεί εγκαίρως η επίθεση malware και να αποτραπεί, παρέχοντας υψηλά επίπεδα ασφάλειας και εγγύησης στο δίκτυο.

➤ *Ασφάλεια ασύρματης επικοινωνίας*

Εξαιτίας του όγκου των πληροφοριών, της ορατότητας των ασύρματων συνδεδεμένων συσκευών στις πλατφόρμες Fog και της συνεχόμενης κίνησης των ιδιωτικών πληροφοριών

μέσα σε αυτό, είναι απαραίτητο να εξασφαλίζεται η ασφάλεια στο δίκτυο Fog, η ελαχιστοποίηση του packet sniffing, rouge access points και άλλες παρόμοιες προκλήσεις, εφαρμόζοντας τόσο κρυπτογράφηση όσο και μεθόδους πιστοποίησης. Η επικοινωνία στην ασύρματη σύνδεση κρυπτογραφείται με Wi-Fi αλγόριθμους ασφαλείας όπως το WiFi Protected Access (WPA), WPA2, κ.α. Εάν η σύνδεση δεν είναι προστατευμένη ο εισβολέας μπορεί να γίνει μέλος της σύνδεσης (Sybil attack) ή να χρησιμοποιεί το εύρος του δικτύου παράνομα (Flood Attack) ή ακόμη και να διακόψει ή να μετατρέψει την διαδικτυακή επικοινωνία (MiTM). Ιδιαίτερα σε περιπτώσεις όπου οι πληροφορίες είναι χρηστικής σημασίας απαραίτητη είναι η χρήση των πρωτοκόλλων 802.11 και των μεταγενέστερων του για τη μέγιστη ασφάλεια.

➤ *Secured vehicular networks*

Η ζήτηση σε ασφάλεια είναι απαιτητική κατά την κίνηση των πληροφοριών και των εφαρμογών σε πραγματικό χρόνο, τόσο από εσωτερικούς όσο και εξωτερικούς εισβολείς. Στο δίκτυο vehicular η σύνδεση είναι ευπαθή ειδικότερα όταν πραγματοποιείται για σύντομη χρονική διάρκεια, καθώς είναι δύσκολη η πιστοποίηση της ταυτότητας. Στις περιπτώσεις αυτές η μεθοδολογία κίνησης πρέπει να είναι ανώνυμη και προσωρινή, η λεγόμενη STAMP, όπου η τοποθέτηση του χρήστη και η ταυτοποίηση του κρατείται προσωρινά στο δίκτυο Fog.

➤ *Secured multi-tenancy*

Το FC είναι ένα εξειδικευμένο εικονοποιημένο περιβάλλον που υποστηρίζει την πολυχρηστικότητα (multi-tenancy) και παρέχει εγκαταστάσεις διαχείρισης των πόρων ανάλογα με τις απαιτήσεις των εφαρμογών και των συνδεδεμένων συσκευών. Όταν αυξάνεται το πλήθος των αιτούντων χρηστών για υπηρεσίες στο FC και προκειμένου να διατηρηθεί η ισορροπία, η εμπιστευτικότητα και η ακεραιότητα στην εκχώρηση των πόρων πραγματοποιείται ένας έλεγχος ταυτότητας πολλαπλών παραγόντων με μηχανισμούς που βασίζονται στην ταυτότητα και στον ρόλο των τελικών χρηστών και αναλύουν τις δραστηριότητες του διαχειριστή και του ενοικιαστή. Ένας τέτοιος μηχανισμός είναι η υπηρεσία Secure and Resilient Networking (SeReNe) η οποία προγραμματίζει και προσαρμόζει την τοπολογία, το εύρος ζώνης, και τις πολιτικές κίνησης, σύμφωνα με την κινητικότητα και τον αριθμό των συσκευών που συνδέονται στο δίκτυο.

➤ *Backup και recovery*

Οι τρόποι διατήρησης του Backup και του recovery είναι το Fibre Channel, η High Security Distribution, η τεχνολογία Rake (HS-DRT), η τεχνική υπηρεσιών Parity Cloud (PCS), η

στρατηγική υπηρεσιών Cold και Hot Backup Replacement (CBSRS) και Shared backup router resources (SBBR). Η βελτίωση του backup και το recovery επιτυγχάνεται με το SSD-υποβοηθούμενο σύστημα βάσης δεδομένων (assisted database), όπου σε περίπτωση φυσικής καταστροφής ή εισβολής κακόβουλου χρήστη τα δεδομένα θα παραμείνουν διαθέσιμα και ακέραια στον χρήστη και στο σύστημα [36].

4.2.1 Πιστοποίηση Mutual Authentication

Οι Loffi κ.ά. [37], αναλύουν τον τρόπο λειτουργίας της μεθόδου πιστοποίησης Mutual Authentication multi-factor κάνοντας χρήση των ασύμμετρων και συμμετρικών κλειδιών.

Στην εργασία τους [37] παρουσιάζουν μια επισκόπηση για το κόστος των προτεινόμενων μεθόδων πιστοποίησης τόσο των ίδιων των συγγραφέων όσο και άλλων ερευνητών. Το κόστος εκτιμάται θεωρητικά και εξαρτάται από τον αλγόριθμο ή την τεχνική κρυπτογράφησης (π.χ., κρυπτογράφηση με τα ασύμμετρα κλειδιά Elliptic Curve Cryptography (ECC) και Rivest Shamir Adleman (RSA), ή με τη χρήση συμμετρικών κλειδιών). Το συνολικό κόστος της ασφάλειας περιλαμβάνει το κόστος επικοινωνίας και το κόστος υπολογισμού.

Το κόστος υπολογισμού (computation) σε κρυπτογράφηση με ασύμμετρο κλειδί ECC και RSA έχει κόστος κρυπτογράφησης T_e/T_{ecm} (asymmetric encryption times). Ενώ η κρυπτογράφηση με συμμετρικό κλειδί έχει κόστος κρυπτογράφησης T_s (symmetric encryption times).

Το κόστος επικοινωνίας υπολογίζεται με τις μετρικές nonce (τυχαίοι αριθμοί), hash (ονομάζεται T_h και είναι ίση με $1T_h$) και timestamps και οι τιμές τους είναι 160 bits, 160 bits και 32 bits αντίστοιχα. Μελέτες που έγιναν μεταξύ του αισθητήρα και το Cloud της google προέκυψε η πληροφορία ότι ο αλγόριθμος καταναλώνει περίπου 1400 bits για κάθε ζήτηση πιστοποίησης κάνοντας χρήση του RSA κλειδιού, και 1280 bits με το ECC κλειδί [37].

4.3 Συμπεράσματα

Στο σύστημα FC απαιτείται οι παραγόμενες πληροφορίες να γίνουν άμεσα διαθέσιμες σε πραγματικό χρόνο, γι' αυτό ο υπολογισμός αναλύει, φιλτράρει, περικόπτει, ανακατασκευάζει, αποθηκεύει και παίρνει αποφάσεις για τα συλλεγόμενα δεδομένα, όταν αυτό είναι απαραίτητο και λαμβάνοντας υπόψη τις απαιτήσεις των χρηστών. Το migration των εργασιών από τον έναν κόμβο Fog στον άλλο, η απαιτούμενη εκμετάλλευση ή η εκχώρηση των πόρων, το offloading, η επεξεργασία και η αποθήκευση κατά την εκτέλεση των εργασιών είναι οι διαφορετικές απαιτήσεις και τα χαρακτηριστικά της κάθε διεργασίας που επηρεάζουν το κόστος υπολογισμού (Πίνακας 4.1).

Το κόστος υπολογισμού σε ένα περιβάλλον FC συνδέεται με το κόστος επεξεργασίας της εφαρμογής εξαρτώμενο από τον συνολικό αριθμό των ενεργειών που ενεργοποιήθηκαν και

εκτελέστηκαν, το κόστος επικοινωνίας της κάθε ενέργειας ή κίνηση των δεδομένων στους διάφορους κόμβους και το καθοριζόμενο επίπεδο ποιότητας υπηρεσιών [8][31]. Σε κάποιες έρευνες επιδιώκεται η βέλτιστη εκχώρηση πόρων στο σύστημα με γνώμονα τη μέγιστη ανεκτή καθυστέρηση μιας εφαρμογής [6][7][35], τον κατάλληλο χρονοδρομολόγηση (προγραμματισμό) και την εξισορρόπηση των εργασιών, ορίζοντας τη μέγιστη χωρητικότητα των συνδέσεων και ικανοποιώντας τις απαιτήσεις στην ποιότητα υπηρεσιών των εφαρμογών στις καταναμημένες πλατφόρμες του Cloud και του FC [23][31].

Κάποιοι άλλοι ερευνητές διαχωρίζουν το κόστος υπολογισμού σε κόστος επικοινωνίας, κόστος μετάδοσης και κόστος δημιουργίας αντιγράφων [23][31][35]. Ενώ σε άλλες έρευνες υπολογίζουν το τη χωρητικότητα της συνδεσιμότητας και το κόστος του εύρους ζώνης δικτύου [17][23]. Ενώ κάποιοι άλλοι στο μοντέλο τους συνδέουν το κόστος με την απόδοση και η οποία εξαρτάται από τα χαρακτηριστικά της CPU και το μέγεθος χρήσης της [4][12]. Το κόστος επικοινωνίας και το κόστος υπολογισμού περιλαμβάνετε στον υπολογισμό του κόστους για την ασφάλεια [37].

Από το σύνολο των μοντέλων, που παρουσιάζονται συνοπτικά παραπάνω, συμπεραίνουμε ότι το FC είναι μία νέα τεχνολογία, με δυνατότητα αναπροσαρμογής και ανακατεύθυνσης των εργασιών σύμφωνα με τις τρέχουσες απαιτήσεις των αιτούμενων εργασιών από τις τελικές συσκευές αλλά και τις δυνατότητες των πόρων που παρέχει ο κάθε κόμβος Fog ή Cloud. Αυτό έχει ως συνέπεια, ο υπολογισμός του κόστους να είναι δύσκολη και πολύπλοκη διαδικασία, λόγω του πολλών και διαφορετικών συνιστωσών που χρειάζονται για την υλοποίησή του. Είναι όμως βέβαιο πως το κόστος υπολογισμού εξαρτάται από τις απαιτήσεις των εφαρμογών, αλλά και την υποδομή των διαθέσιμων πόρων στους κόμβους Fog.

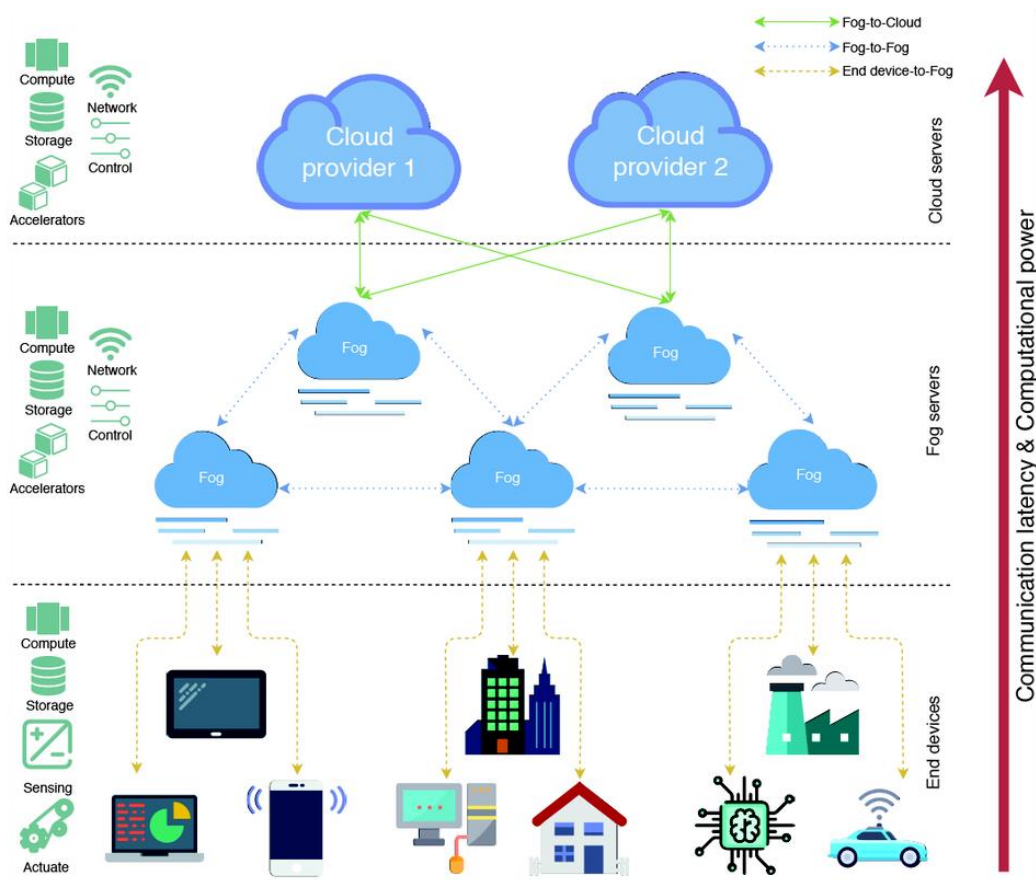
Πίνακας 4.1: Παράμετροι υπολογισμού κόστους υπολογισμού.

Αναφορές cost factor	Ποιότητα υπηρεσιών	Ρυθμό μετάδοσης MIPS	Λάθη κατά τη μετακίνηση δεδομένων	Κατανάλωση ενέργειας	Χαρακτηριστικά της CPU της κινητής συσκευής	Κόστος δημιουργίας αντιγράφων ασφαλείας	Χαρακτηριστικά της CPU της κινητής συσκευής	Όγκος δεδομένων	Διαμόρφωση κόστους	Καθυστερήση υπηρεσιών	Χρόνος εκτέλεσης	Μέγεθος αποθήκευσης	Συνολικό αριθμό κινήσεων	Συνολικό αριθμό MIPS στο host	Απόλαβα εσόδων εξαιτίας της καθυστέρησης της WAN μετάδοσης	Αριθμός διεργασιών και κόστος επεξεργασίας κινήσεων, ή το μέγεθος χρήσης των VCPU και CPU ανά ώρα	Μέγεθος της εφαρμογής
Battula 2019, [8]	✓								✓				✓			✓	
Yu 2017, [32]				✓					✓	✓				✓			✓
Pham 2017, [31]	✓		✓						✓			✓	✓				
Liu 2017, [29]						✓			✓	✓							
Mao 2021, [26]				✓	✓		✓									✓	
Rah 2017, [33]		✓									✓			✓			
Wang 2021, [23]			✓					✓	✓				✓				
Rejiba 2019, [3]						✓				✓	✓						

5 Υλοποίηση προσομοίωσης FogComputingSim

5.1 Εισαγωγή

Ο προσομοιωτής iFogSim [38] είναι μια υψηλής απόδοσης εργαλειοθήκη ανοιχτού κώδικα κατάλληλη για προσομοίωση FC, edge computing και IoT. Το iFogSim ενσωματώνει τις τεχνικές διαχείρισης πόρων και συνεργάζεται με το CloudSim, μια ευρέως χρησιμοποιούμενη βιβλιοθήκη για την προσομοίωση περιβάλλοντος που βασίζεται σε Cloud. Το περιβάλλον προσομοίωσης επιτρέπει τον καθορισμό του πλήθους των Fog, και συσκευών IoT (π.χ., αισθητήρων, actuators). Το iFogSim έχει τη δυνατότητα να αξιολογεί την καθυστέρηση στην από άκρο-σε-άκρο σύνδεση, τη συμφόρηση στο δίκτυο, τη χρησιμοποιούμενη ενέργεια, τα έξοδα λειτουργίας και την ικανοποίηση σε ποιότητα υπηρεσιών. Σε αρκετές εργασίες το iFogSim έχει χρησιμοποιηθεί για την προσομοίωση πόρων, κινητικότητας, καθυστέρησης, ποιότητα εμπειρίας (QoE), ενέργειας, ασφάλειας και ποιότητα υπηρεσιών στο περιβάλλον FC [38].



Εικόνα 5.1: Αρχιτεκτονική FogComputingSim, πηγή [39].

Ο προσομοιωτής FogComputingSim [39] είναι μια επέκταση του iFogSim κατάλληλη για προσομοίωση διάφορων σεναρίων συστημάτων FC. Το βασικό του πλεονέκτημα είναι το σύνολο των υλοποιημένων αλγορίθμων βελτιστοποίησης για την επίλυση διατυπωμένου προβλήματος κόστους στο FC και η εξεύρεση βέλτιστης λύσης. Πρόκειται για παραμετροποιήσιμο σύστημα που ο χρήστης μπορεί να ορίσει τα δικά του σενάρια.

5.2 Η αρχιτεκτονική FogComputingSim

5.2.1 Τα χαρακτηριστικά της προσομοίωσης FogComputingSim

- *Υποστήριξη της ετερογένειας*: Υποστηρίζει την επεξεργασία δεδομένων που συλλέχθηκαν από διαφορετικούς παράγοντες και είναι επηρεασμένα από τα πολλαπλά είδη δικτυακής επικοινωνίας σε ένα σύνολο από διαφορετικά είδη κόμβων με μεγάλη ποικιλία εφαρμογών και υπηρεσιών.
- *Γεωγραφική κατανομή*: Σε αντίθεση με το Cloud, χρησιμοποιεί όλους τους πόρους μεταξύ του Cloud και των things για να παρέχει τους πανταχού παρόντες υπολογιστικούς πόρους, επιτρέποντας τη συνέχεια των υπηρεσιών σε περιβάλλοντα κινητής τηλεφωνίας.
- *Ενυαισθητοποίηση σχετικά με την τοποθεσία και τη χαμηλή καθυστέρηση*: Παρέχει χαμηλή καθυστέρηση λόγω της εγγύτητας μεταξύ των συσκευών IoT και των κόμβων Fog. Επίσης, η τοποθεσία τους επιτρέπει να έχουν επίγνωση του κόστους της καθυστέρησης επικοινωνίας τόσο με άλλους κόμβους Fog όσο και με τις τελικές συσκευές, επιτρέποντας την διανομή εφαρμογών σε όλο το δίκτυο που οργανώνεται με ισοσταθμισμένο τρόπο.
- *Υποστήριξη κινητικότητας*: Υποστηρίζει την ετερογένεια, τη γεωγραφική κατανομή και την ενυαισθητοποίηση σχετικά με τον εντοπισμό της τοποθεσίας ώστε να υποστηρίζονται οι κινητές συσκευές στη μέγιστη παροχή των υπηρεσιών της.
- *Αλληλεπιδράσεις σε πραγματικό χρόνο*: Σε αντίθεση με την παρτίδα επεξεργασίας που εκτελείται από διακομιστές Cloud, χρησιμοποιεί τα χαρακτηριστικά Fog για την υποστήριξη της ανάπτυξης και της επεξεργασίας των εφαρμογών σε πραγματικό χρόνο.
- *Επεκτασιμότητα και ευελιξία ομαδοποιημένων κόμβων Fog*: Το FC είναι προσαρμοστικό, δηλαδή μπορεί να σχηματίσει cluster-of-nodes ή cluster-of-cluster για την υποστήριξη των υπολογισμών, τη συγκέντρωση πόρων κ.λπ., υποστηρίζοντας εφαρμογές μεγάλης κλίμακας.
- *Πολλαπλές IoT εφαρμογές*: Οι συσκευές Fog χειρίζονται πολλές εφαρμογές IoT που ανταγωνίζονται για τους περιορισμένους τους πόρους
- *Υποστήριξη της εικονικοποίησης*: Εισάγει μία επέκταση λογισμικού μεταξύ του υλικού (hardware) και του λειτουργικού συστήματος και η εφαρμογή εκτελείται στο υλικό.

- *Διαλειτουργικότητα και ομοσπονδία*: Συνεργάζεται με διαφορετικούς παρόχους για την υποστήριξη "βαρέων" εφαρμογών, όπως ροή σε πραγματικό χρόνο.
- *Κυριαρχία της ασύρματης πρόσβασης*: Χρησιμοποιεί κάθε είδους συνδέσεις για την υποστήριξη της συνδεσιμότητας σε ένα ευρύ φάσμα τελικών συσκευών (οι οποίες ενδέχεται να μην υποστηρίζουν ενσύρματες συνδέσεις).

5.2.2 Τα συστατικά της προσομοίωσης FogComputingSim

Η αρχιτεκτονική του FC είναι μία ιεραρχική δομή από κόμβους Fog, διάσπαρτους στο δίκτυο, μεταξύ των αισθητήρων και του Cloud [40]. Συγκεκριμένα:

Οι *αισθητήρες IoT* τοποθετούνται στο χαμηλότερο επίπεδο της αρχιτεκτονικής, κατανέμονται σε διαφορετικές γεωγραφικές τοποθεσίες και έχουν τις εξής λειτουργίες:

- Ανιχνεύουν το περιβάλλον και εφαρμόζουν τα modules για να παρέχουν συμπληρωματική υπολογιστική ισχύ [40].
- Δρομολογούν τη ροή δεδομένων στα ανώτερα επίπεδα μέσω των πυλών και η επεξεργασμένη πληροφορία που μας ενδιαφέρει αποθηκεύεται στα data centers για περαιτέρω επεξεργασία ή/και φιλτάρισμα [40].
- Σχηματίζει μία αναφορά συμπεριφοράς και καθυστέρησης στην πύλη της συσκευής Fog, που ο αισθητήρας είναι συνδεδεμένος [39].
- Καθορίζει τα εξωτερικά χαρακτηριστικά του αισθητήρα και την κατανομή του χρόνου άφιξης του tuple στην πύλη [39].

Παρόμοια οι *IoT Actuators* λειτουργούν στο χαμηλότερο επίπεδο της αρχιτεκτονικής, είναι υπεύθυνοι για τον έλεγχο του συστήματος και αντιδρούν στις αλλαγές του περιβάλλοντος, ελεγχόμενοι από αισθητήρες.

Οι *συσκευές Fog* είναι κάθε στοιχείο στο δίκτυο που φιλοξενεί τα modules των εφαρμογών. Επίσης, οι συσκευές Fog ή πύλες (gateways) είναι αυτές που συνδέουν τους αισθητήρες στο διαδίκτυο [41].

5.2.3 Οι υπηρεσίες

Η αρχιτεκτονική Fog έχει τρεις κύριες υπηρεσίες και ένα μοντέλο εφαρμογών για το περιβάλλον IoT και Fog, όπως περιγράφεται παρακάτω:

- Τα στοιχεία Monitoring διατηρούν τον έλεγχο της συμμόρφωσης κατά τη διαχείριση των πόρων, των αισθητήρων, των actuators, των συσκευών Fog και των στοιχείων δικτύωσης. Ακόμη

παρακολουθούν τις εφαρμογές και τις υπηρεσίες που αναπτύσσονται στην υποδομή, ελέγχοντας την απόδοσή τους.

- Η διαχείριση των πόρων είναι στοιχείο που επιδιώκει την μέγιστη QoS.
- Ο έλεγχος ενέργειας είναι υπεύθυνος για τον έλεγχο και την εκτύπωση της κατανάλωσης ενέργειας των συσκευών Fog στη προσομοίωση.

Οι εφαρμογές στη αρχιτεκτονική Fog βασίζονται στο μοντέλο DDF (Distributed Data Flow). Κάθε εφαρμογή σχεδιάζεται ως μία συλλογή από modules, η οποία αποτελείται από επεξεργασμένα δεδομένα. Τα δεδομένα που θεωρούνται έξοδος από το module i , μπορούν να χρησιμοποιηθούν ως είσοδος από άλλο module j , δίνοντας αύξηση της εξάρτησης ανάμεσα στο module i και j . Αυτό το μοντέλο εφαρμογών επιτρέπει την γραφική αναπαράσταση των εφαρμογών και εμφανίζει τη ροή των δεδομένων μεταξύ των modules [39].

Με το module Sense-Process-Actuate οι πληροφορίες που συλλέγονται από τους αισθητήρες εκπέμπονται ως ροή δεδομένων, η οποία ενεργοποιείται από εφαρμογές που εκτελούνται στις συσκευές Fog και οι προκύπτουσες εντολές αποστέλλονται στους actuators (ενεργοποιητές) [41].

5.2.4 Σχεδιασμός και implementation

Σε αυτή την υποενότητα παρουσιάζουμε και αναλύουμε τον τρόπο λειτουργίας των συστατικών μερών της προσομοίωσης FogComputingSim.

Physical στοιχεία

Η τοπολογία της προσομοίωσης αποτελείται από οντότητες όπως χρήστες, κόμβους Fog και Cloud. Ανάλογα με το είδος των υπηρεσιών οι χρήστες προσαρμόζονται σε οντότητες όπως οι αισθητήρες και actuators. Σε κάθε αισθητήρα ορίζονται γνωρίσματα εξαρτόμενα από τη συνδεσιμότητα και τα εξωτερικά γνωρίσματα που ορίζονται από τον συγκεκριμένο κόμβο πύλης με τον οποίο είναι συνδεδεμένος.

Στην Εικόνα 5.1 βλέπουμε τους actors και την αλληλεπίδρασή τους, πως οι κόμβοι IoT συνδέονται με τους διακομιστές Fog, πως οι clients αναθέτουν σε τρίτους την εκχώρηση και την διαχείριση των πόρων.

Το χαμηλότερο επίπεδο των συσκευών Fog συνδέεται απευθείας με τους sensors και τους actuators. Οι συσκευές Fog δρουν σαν ένα datacenter που παρέχει μνήμη, δικτύωση και υπολογιστικούς πόρους. Κάθε συσκευή Fog αποτελείται από συγκεκριμένο εύρος επεξεργασίας και κατανάλωση ενέργειας, η οποία επιδρά στην ικανότητα, την ενεργειακή απόδοση, τα κόστη

καθυστερήσης, την κατανάλωση ενέργειας, την επεξεργασία, το εύρος ζώνης και τη μετεγκατάσταση.

Όταν απαιτείται να σταλεί μία πληροφορία σε μία άλλη οντότητα τότε δημιουργείται το tuple το οποίο φροντίζει για την αποστολή σε άλλη πύλη ή στον επόμενο κόμβο που φαίνεται στον πίνακα δρομολόγησης του πακέτου. Πρόκειται δηλαδή για ένα πακέτο που στέλνεται μέσω των δρομολογημένων συνδέσεων και χαρακτηρίζεται από την ταυτότητα της πηγής και τα modules προορισμού.

Τέλος, το FC χρειάζεται ένα επίπεδο ενορχήστρωσης (orchestration) όπου θα διαχειρίζεται το τρέχον περιεχόμενο και θα λαμβάνει αποφάσεις διαχείρισης με γνώμονα τις εφαρμογές και την τοποθέτηση των δεδομένων. Πρόκειται για μία υπηρεσία που μεσολαβεί ώστε να βρεθεί ο ιδανικός διακομιστής Fog, που θα παρέχει συγκεκριμένες δυνατότητες και απαιτήσεις. Όταν δηλαδή, λαμβάνει αιτήματα πελατών, με τα χαρακτηριστικά τους όπως πληροφορίες δικτύου, δυνατότητες υλικού και απόσταση, αναζητά και βρίσκει τον καλύτερο διαθέσιμο κόμβο εξυπηρέτησης των αιτημάτων του πελάτη [39].

Logical στοιχεία

Τα *Application Modules* και *Application edges* είναι τα λογικά συστατικά μέρη της προσομοίωσης [39]. Πρόκειται για εφαρμογή που παρέχει επεξεργασία στα εισερχόμενα δεδομένα ορίζοντας τις εξαρτήσεις δεδομένων μεταξύ των modules. Αυτές οι οντότητες χωρίζονται στις παρακάτω κλάσεις [41]:

- *AppModule*: Η κλάση *AppModule* εκπροσωπεί τα επεξεργάσιμα στοιχεία των εφαρμογών Fog. Αυτή η κλάση επεξεργάζεται κάθε εισερχόμενο tuple, είναι υπεύθυνη για την φιλοξενία του και αρχίζει την αποστολή του οριζόμενου tuple στο επόμενο module.
- *AppEdge*: Τα *AppEdges* προγραμματίζουν τα γεγονότα και την επόμενη δρομολόγηση που θα ακολουθήσει.
- *AppLoop*: Πρόκειται για μία επιπλέον κλάση που ορίζει τα loops που ενδιαφέρουν τον χρήστη [41].

Συστατικά Management

Τα συστατικά μέρη της διαχείρισης είναι ο Controller και τα Module Mapping αντικείμενα. Το Module Mapping αντικείμενο σύμφωνα με τις απαιτήσεις των AppModules αναζητάει διαθέσιμους πόρους στις συσκευές Fog και τους τοποθετεί ανάμεσά τους. Εάν μια συσκευή Fog δεν μπορεί ή δεν πληρεί τις απαιτήσεις ενός module, το module αποστέλλεται σε συσκευή Fog

σε ανώτερο επίπεδο. Ο Controller εκκινεί τα AppModule στις συσκευές Fog ακολουθώντας τις πληροφορίες τοποθέτησης που παρέχονται από το Module Mapping και περιοδικά διαχειρίζονται τους πόρους των συσκευών Fog. Όταν τερματιστεί η προσομοίωση, ο ελεγκτής (controller) συγκεντρώνει τα αποτελέσματα κόστους, χρήσης του δικτύου (network usage) και κατανάλωσης ενέργειας κατά την περίοδο προσομοίωσης από τις συσκευές Fog [39].

5.2.5 Δημιουργία προσομοίωσης στο FogComputingSim

Το Graphical User Interface (GUI) στην προσομοίωση επιτρέπει την φυσική διαμόρφωση τοπολογίας δικτύου. Δηλαδή δίνει τη δυνατότητα στο χρήστη προσθήκης, διαχείρισης και διαγραφής στοιχείων φυσικών οντοτήτων, όπως κέντρο δεδομένων Cloud, συσκευών Fog, αισθητήρων, actuators και συνδέσμων, ορίζοντας τα χαρακτηριστικά των οντοτήτων αυτών. Η σχεδιασμένη τοπολογία μπορεί να αποθηκευτεί και να επαναφορτώνεται διαμορφώνοντας την στο JSON (JavaScript Object Notation) αρχείο διαμόρφωσης [41].

Στην προσομοίωση FogComputingSim εκτελούνται 3 αλγόριθμοι βελτιστοποίησης:

1. Ο αλγόριθμος Genetic, μεταβιβάζει τα χαρακτηριστικά από γενιά σε γενιά, δημιουργώντας βέλτιστες λύσεις που οδηγούν σταδιακά στην εξαφάνιση των κακών λύσεων και στη βελτίωση των καλών λύσεων. Χρησιμοποιεί δύο συνθήκες διακοπής: το μέγιστο αριθμό γενεών και τον αριθμό των συγκλίσεων του αλγορίθμου. Στην αρχή δημιουργείται ένας νέος πληθυσμός με αυθαίρετο αριθμό ατόμων, καθένα από τα οποία τοποθετείται στον ψευδοτυχαία παραγόμενο πίνακα P. Στη συνέχεια, για κάθε επανάληψη κάθε άτομο λαμβάνει τους πίνακες R και V με τις βέλτιστες τιμές, βάση του πίνακα P. Ο χειριστής επιλογής δίνει προτεραιότητα στα άτομα με καλές βαθμολογίες φυσικής κατάστασης και τους επιτρέπει να περάσουν τα γονίδια τους στις διαδοχικές γενιές. Με αυτόν τον τρόπο, δημιουργούνται οι πίνακες P των καλύτερων ατόμων που περνάει τα νέα άτομα στη νέα γενιά. Τότε ο τελεστής crossover φροντίζει για το ζευγάρι μεταξύ των ατόμων, χρησιμοποιώντας το τελεστή επιλογής, επιλέγοντας δύο άτομα των τυχαίων επιλεγμένων θέσεων διασταύρωσης. Στη συνέχεια, τα γονίδια σε αυτές τις θέσεις διασταύρωσης ανταλλάσσονται δημιουργώντας έτσι ένα πλήρη νέο άτομο (απόγονο). Για να γίνει αυτό, για καθένα από τα υπόλοιπα νέα άτομα, επιλέγονται δύο τυχαία άτομα από το σύνολο των καλύτερων ατόμων και εκτελείται το μεταξύ τους ζευγάρι. Επιπλέον, εφαρμόζεται ο τελεστής μετάλλαξης, ο οποίος ορίζει μια τυχαία διαδρομή με βάση την εκκίνησή του και τον κόμβο προορισμού και ο οποίος εισάγει τα τυχαία γονίδια για να διατηρήσει την ποικιλομορφία στον πληθυσμό και να αποφευχθεί η πρόωρη σύγκλιση. Στη

συνέχεια, οι πίνακες R και V συμπληρώνονται με τους αντίστοιχους κόμβους αρχής και τον προορισμό τους που βασίζονται στον πίνακα P [39].

2. Ο Random αλγόριθμος συνίσταται με τη δημιουργία ψευδοτυχαίων λύσεων, χωρίς τη γνώση των λύσεων που δημιουργήθηκαν προηγουμένως, ενώ χρησιμοποιούν δύο περιοριστικές συνθήκες: Ορίζεται ο μέγιστος επιτρεπόμενος αριθμός επαναλήψεων που μπορεί να εκτελέσει ο αλγόριθμος και προστίθεται ο αριθμός των επαναλήψεων στις οποίες θεωρείται ότι έχει κάνει ο αλγόριθμος. Το δεύτερο πραγματοποιείται με τη χρήση ενός μετρητή ο οποίος αυξάνεται όταν μεταξύ των επαναλήψεων, δεν παρουσιάζεται καλύτερη λύση, αντίθετα, όποτε συμβαίνει, αυτό ο μετρητής ορίζεται στο 0. Για κάθε επανάληψη δημιουργεί έναν δεδομένο ψευδοτυχαίο πίνακα P. Στη συνέχεια, με βάση τον παραγόμενο πίνακα P, δημιουργούνται οι δύο πίνακες δρομολόγησης R και V, με τις βέλτιστες λύσεις. Ελέγχονται όλοι οι κόμβοι πηγής και προορισμού και επαληθεύει εάν ο προορισμός έχει ήδη βρεθεί ώστε να γεμίσουν οι πίνακες. Έτσι κάθε στήλη γεμίζει με έναν κόμβο προορισμού και δημιουργείται μία λίστα με τους πιθανούς κόμβους. Στη συνέχεια επιλέγεται ένας τυχαίος κόμβος από το σύνολο των πιθανών κόμβων [39].
3. Ο Brute Force είναι αλγόριθμος που αναζητά τη βέλτιστη λύση από ένα σύνολο τιμών και υλοποιείται χρησιμοποιώντας τη τυπική αναδρομική προσέγγιση brute force. Ο αλγόριθμος, αρχίζει αναδρομικά να γεμίζει το πίνακα τοποθέτησης, P. Ενώ το κάνει αυτό, επαληθεύει εάν ο επιλεγμένος κόμβος n μπορεί να φιλοξενήσει το επιλεγμένο module m επιβεβαιώνοντας εάν θα τοποθετηθεί στον πίνακα D ή στον P. Εάν δεν πληροί τους περιορισμούς για το D, συνεχίζει να γεμίζει τον P και επιλέγει τον επόμενο κόμβο. Όταν το P έχει τελικά γεμίσει (με τις πιθανές λύσεις), και έχουν εκλεχθεί όλοι οι κόμβοι η διαδικασία επαναλαμβάνεται για να γεμίσει ο πίνακας V. Ο στόχος είναι να βρεθούν οι καλύτερες διαδρομές δεδομένου κάθε αρχικού και τελικού κόμβου. Οι περιορισμοί εκτελούν τον κανόνα του κόστους υψηλότερης προτεραιότητας χωρίς σχετική ανοχή. Στη συνέχεια, για κάθε στόχο, γίνεται σύγκριση των τιμών κόστους των δύο πινάκων με φθίνουσα σειρά προτεραιότητας χρησιμοποιώντας μια δεδομένη σχέση ανοχής [39].

Εκτελώντας τους αλγορίθμους αξιολογούνται τα διαφορετικά στατικά ή μεταβλητά σενάρια και εμφανίζουν τιμές για τις συναρτήσεις κόστους:

- καθυστέρησης
- ενέργειας
- επεξεργασίας
- εύρους ζώνης και

- μετεγκατάστασης (Εικόνα 5.2).

```

30
31 public static void analyzeSolution(Algorithm algorithm, Solution solution) {
32     double constraint = Constraints.checkConstraints(algorithm, solution);
33     solution.setDetailedCost(Config.LATENCY_COST, computeLatencyCost(algorithm, solution));
34     solution.setDetailedCost(Config.POWER_COST, computePowerCost(algorithm, solution));
35     solution.setDetailedCost(Config.PROCESSING_COST, computeProcessingCost(algorithm, solution));
36     solution.setDetailedCost(Config.BANDWIDTH_COST, computeBandwidthCost(algorithm, solution));
37     solution.setDetailedCost(Config.MIGRATION_COST, computeMigrationCost(algorithm, solution));
38     solution.setConstraint(constraint);
39 }

```

Εικόνα 5.2: Συναρτήσεις κόστους

Η αρχική εγκατάσταση του προγράμματος εκτελεί τη συνάρτηση κόστους QoS (ποιότητας υπηρεσιών), την οποία αντικαταστήσαμε με την συνάρτηση *κόστους καθυστέρησης* (Εικόνα 5.3). Δημιουργήσαμε τον κώδικά της όπου ενημερώνει τις τιμές κατά την δρομολόγηση των tuples στα modules της προσομοίωσης που εκτελείται, στα στατικά ή στα μεταβλητά σενάρια της εφαρμογής.

```

30
31 private static double computeLatencyCost(Algorithm algorithm, Solution solution) {
32
33
34     int[][] tupleRoutingMap = solution.getTupleRoutingMap();
35
36     double latency = 0;
37
38     // For each Link, in the tuple routing map sum the total latency
39     for(int j = 0; j < algorithm.getNumberOfModules(); j++) {
40         for (int i = 0; i < algorithm.getNumberOfNodes() - 1; i++) {
41             int start = tupleRoutingMap[j][i];
42             int end = tupleRoutingMap[j][i+1];
43
44             if(start == end) continue;
45
46             latency += algorithm.getfLatencyMap()[start][end]; // Link latency
47
48         }
49     }
50     return latency;
51 }
52
53
54

```

Εικόνα 5.3: Συνάρτηση κόστους καθυστέρησης

5.3 Εφαρμογές σε περιβάλλον Fog

Σε αυτή την ενότητα παρουσιάζουμε δυο μελέτες προσομοιώσεων που εφαρμόζονται σε φυσικές προσομοιώσεις:

1. Online παιχνίδι που είναι ευάλωτο στην καθυστέρηση.
2. Έξυπνη επιτήρηση μιας περιοχής μέσω κατανεμημένου δικτύου κάμερας.

5.3.1 Το Electroencephalogram (EEG) Tractor Beam Game

Το Electroencephalogram (EEG) Tractor Beam Game τρέχει σε Android λογισμικό και αφορά επαυξημένη αλληλεπίδραση εγκεφάλου με υπολογιστή. Η εφαρμογή παρέχει επεξεργασία σε πραγματικό χρόνο στα EEG σήματα που δρομολογούνται από το EEG Tractor Beam Game και υπολογίζει την εγκεφαλική κατάσταση του χρήστη[41].

Η εφαρμογή EEG έχει τρία modules τα:

1. Client
2. Concentration Calculator and
- 3.Coordinator.

Τα modules της εφαρμογής σχεδιάζονται στην κλάση AppModule και στην κλάση AppEdge σχεδιάζονται οι εξαρτήσεις των δεδομένων μεταξύ των modules. Τέλος ο έλεγχος loop σχεδιάζεται στην κλάση AppLoop. Η εφαρμογή τροφοδοτείται με σήματα EEG από τον αισθητήρα EEG και έναν actuator DISPLAY που στέλνει την τρέχουσα εφαρμογή στον χρήστη.

Η λειτουργία των modules είναι:

1. Client: Το Client module αλληλεπιδρά με τον αισθητήρα και λαμβάνει τα σήματα EEG. Ελέγχει τις τιμές των ληφθέντων σημάτων για τυχόν ασυμφωνίες και απορρίπτει κάθε τιμή που είναι φαινομενικά ασυνεπείς. Εάν η τιμή του σήματος από τον αισθητήρα είναι σταθερή, στέλνεται στο module Concentration Calculator λαμβάνοντας το επίπεδο συγκέντρωσης του χρήστη από το σήμα. Στη συνέχεια εμφανίζει το επίπεδο συγκέντρωσης στέλνοντας την τιμή στον actuator DISPLAY.
1. Concentration Calculator: Το module Concentration Calculator είναι υπεύθυνο για τον έλεγχο και τον υπολογισμό των τιμών του σήματος EEG και στη συνέχεια πληροφορεί το module Client σχετικά με το υπολογισμένο επίπεδο συγκέντρωσης, ώστε να αναβαθμιστεί η αιτούμενη εφαρμογή από τον χρήστη.
2. Coordinator: Το Coordinator λειτουργεί σε σφαιρικό επίπεδο, συντονίζοντας τους παίκτες μεταξύ τους, που πιθανόν να βρίσκονται σε διαφορετικά κατανεμημένες περιοχές. Ακόμη στέλνει τη τρέχουσα κατάσταση του παιχνιδιού στο module των χρηστών που είναι συνδεδεμένοι.

5.3.2 Κατανεμημένο σύστημα από κάμερες

Η συγκεκριμένη εφαρμογή αφορά ένα κατανεμημένο σύστημα από κάμερες που παρακολουθεί μία περιοχή, έχοντας υψηλές απαιτήσεις συστήματος καθώς χρειαζόμαστε εργαλεία τα οποία

αναλύουν αυτόματα τα δεδομένα, που προέρχονται από αυτές και συνοψίζουν τα αποτελέσματα με τρόπο ωφέλιμο για τον χρήστη[41].

Το μοντέλο αυτό αποτελείται από πέντε modules που κάνουν την επεξεργασία:

1. Motion Detector
2. Object Detector
3. Object Tracker
4. PTZ Control
5. User Interface

Η εφαρμογή τροφοδοτείται με ζωντανές ροές βίντεο από τις κάμερες και τον ελεγκτή PTZ σε κάθε κάμερα όπου προσαρμόζει τις PTZ παραμέτρους.

Αναλυτικά η λειτουργία των modules είναι:

1. Motion Detection: Πρόκειται για module που είναι ενσωματωμένο σε κάμερα και όταν ανιχνεύσει κίνηση από κάποιο αντικείμενο, προωθεί το βίντεο στο module Object Detection για παραπέρα επεξεργασία.
2. Object Detection: Συγκρίνει τα κινούμενα αντικείμενα στην κάμερα με τα προηγούμενα που είχαν καταγραφεί και κάνει τις ανάλογες μετρήσεις.
3. Object Tracker: Λαμβάνει τις μετρήσεις από το Object detection και υπολογίζει την PTZ διαμόρφωση όλων των καμερών που είναι στην περιοχή, ώστε τα αντικείμενα να παρακολουθούνται και καταγράφονται με τον πιο αποτελεσματικό και ωφέλιμο για τον χρήστη τρόπο.
4. PTZ Control (Pan-Tilt_Zoom): Πρόκειται για module που είναι σε κάθε κάμερα και λειτουργεί ως ενεργοποιητής (actuator) του συστήματος.
5. User Interface: Στέλνει την ροή βίντεο με τα αντικείμενα που παρακολουθούνται, στη συσκευή του χρήστη.

5.4 Σενάρια προσομοίωσης

Με τον προσομοιωτή FogComputingSim στο γραφικό περιβάλλον του GUI σχεδιάσαμε δεκατρία σενάρια προσομοιώσεων συστήματος Fog και μελετήσαμε το συνολικό κόστος των συναρτήσεων του συστήματος. Στα σενάρια (Πίνακας 5.1) αλλάξαμε τον αριθμό των παραμέτρων όπως το Cloud, τις συσκευές Fog, τα apps, το κόστος υπολογισμού (RAM, CPU), κάναμε τις μεταξύ τους συνδέσεις και

με τις μετρικές του κόστους δικτύου (bandwidth, καθυστέρηση Fog) υπολογίζεται το κόστος καθυστέρησης, το κόστος της ενέργειας, το κόστος επεξεργασίας, το κόστος εύρους ζώνης και το κόστος migration.

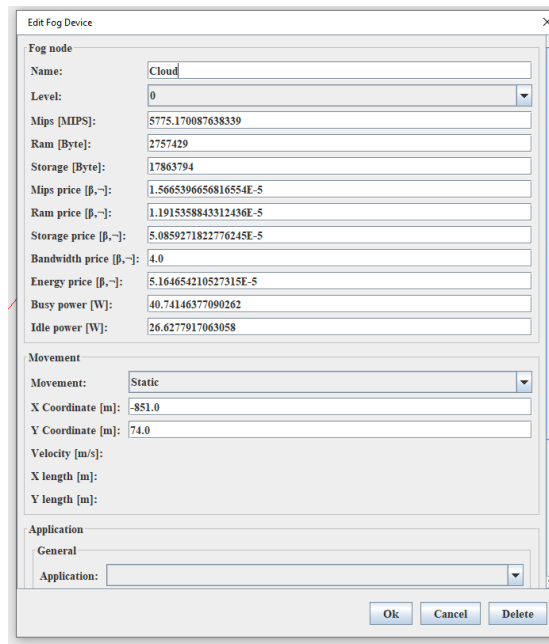
Πίνακας 5.1: Οι παράμετροι των σεναρίων.

Σενάριο	Αριθμός Cloud	Αριθμός Fog	Αριθμός Apps
1°	1	2	3
2°	1	3	3
3°	1	4	3
4°	1	5	3
5°	1	2	4
6°	1	3	4
7°	1	4	4
8°	1	5	4
9°	1	2	5
10°	1	3	5
11°	1	4	5
12°	1	5	5
13°	1	2	6

5.4.1 Τιμές παραμέτρων

Κατά την προσθήκη των συσκευών Fog για τη δημιουργία της προσομοίωσης ενημερώνουμε τις τιμές των παραμέτρων (Πίνακας 5.2) των συσκευών Fog (Εικόνα 5.4) και του Cloud (Εικόνα 5.5).

Εικόνα 5.4: Παράμετροι συσκευών Fog.



Εικόνα 5.5: Παράμετροι συσκευής Cloud.

Πίνακας 5.2: Παράμετροι προσομοίωσης.

Παράμετροι	Cloud	Fog node	Apps
CPU length (MIPS)	5775	1000	1000
Bandwidth	4	3000	-----
Bandwidth Connection	52428	52428	52428
Level	0	1	2
Fog Latency(ms)	0.2	0.2	0.2
Cloud Latency(ms)	0.2	0.2	0.2

5.4.2 Αξιολόγηση της απόδοσης

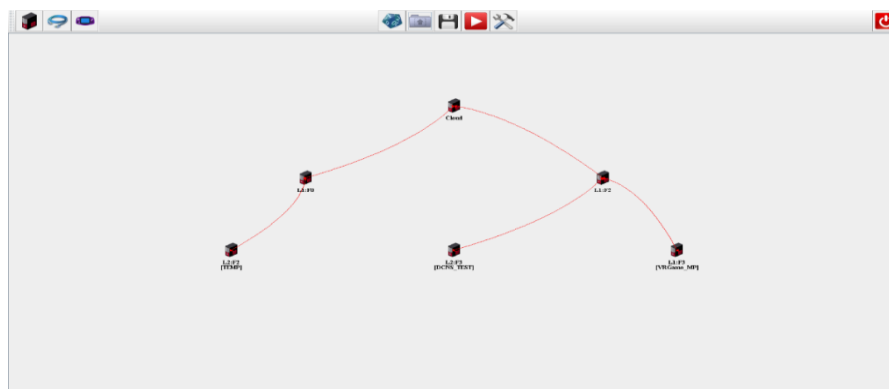
Σε αυτή την υποενότητα παρουσιάζουμε τις δεκατρείς προσομοιώσεις FC και για κάθε μία από τις τρεις αλγοριθμικές περιπτώσεις, που εκτελούνται στην προσομοίωση μας, δείχνουμε τα κόστη των συναρτήσεων:

1. Καθυστέρησης (Latency)
2. Ενέργειας (Power)
3. Επεξεργασίας (Processing)
4. Εύρους ζώνης (Bandwidth)
5. Μετεγκατάστασης (Migration)

5.4.2.1 Σενάριο 1

Το σενάριο 1 αποτελείται από 6 κόμβους, συγκεκριμένα (Σχήμα 1):

- 1 κόμβο Cloud
- 2 κόμβους Fog και
- 3 τυχαίες εφαρμογές (apps)



Σχήμα 1: Σενάριο 1

Τα κόστη (Πίνακας 5.3) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

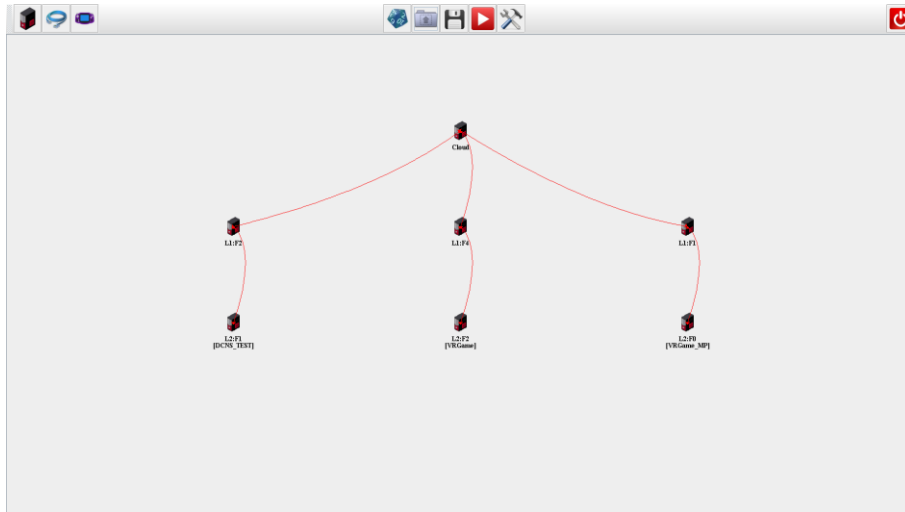
Πίνακας 5.3: Πίνακας τιμών κόστους σεναρίου 1

	Brute Force	Random	Genetic
Latency	7.0	8.0	7.0
Power	43.56	43.75	43.56
Processing	0.06	0.55	0.06
Bandwidth	0.0021	8.22	0.0021
Migration	0.0	0.0	0.0
Συνολικό κόστος	50.6241	60.52	50.6241

5.4.2.2 Σενάριο 2

Το σενάριο 2 αποτελείται από 7 κόμβους, συγκεκριμένα (Σχήμα 2):

- 1 κόμβο Cloud
- 3 κόμβους Fog και
- 3 τυχαίες εφαρμογές (apps)



Σχήμα 2: Σενάριο 2

Τα κόστη (Πίνακας 5.4) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

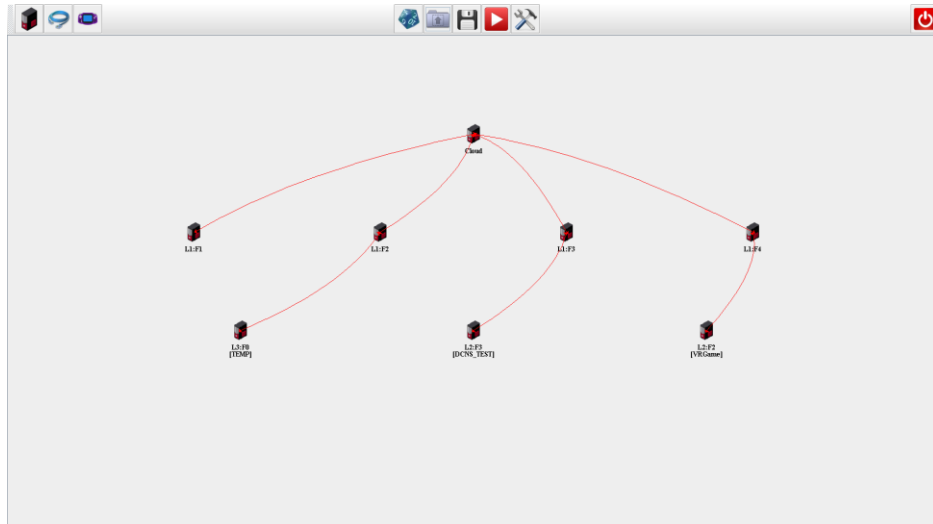
Πίνακας 5.4: Πίνακας τιμών κόστους σεναρίου 2.

	Brute Force	Random	Genetic
Latency	2	8	2
Power	0.023	1.2718	0.023
Processing	0.003	0.1825	0.003
Bandwidth	2.384	0.025	2.384
Συνολικό κόστος	4.410	9.4793	4.410

5.4.2.3 Σενάριο 3

Το σενάριο 3 αποτελείται από 8 κόμβους, συγκεκριμένα (Σχήμα 3):

- 1 κόμβο Cloud
- 4 κόμβους Fog και
- 3 τυχαίες εφαρμογές (apps)



Σχήμα 3: Σενάριο3

Τα κόστη (Πίνακας 5.5) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

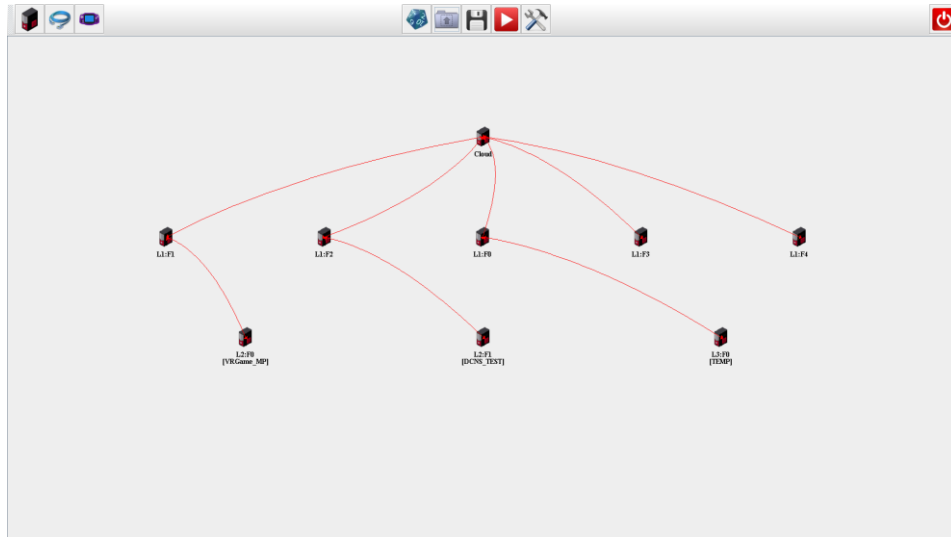
Πίνακας 5.5: Πίνακας τιμών κόστους σεναρίου 3.

	Brute Force	Random	Genetic
Latency	0	8	0
Power	0	0.4	0
Processing	0	0.003	0
Bandwidth	0	8.9	0
Migration	0	0	0
Συνολικό κόστος	0	16.943	0

5.4.2.4 Σενάριο 4

Το σενάριο 4 αποτελείται από 9 κόμβους, συγκεκριμένα (Σχήμα 4):

- 1 κόμβο Cloud
- 5 κόμβους Fog και
- 3 τυχαίες εφαρμογές (apps)



Σχήμα 4: Σενάριο 4

Τα κόστη (Πίνακας 5.6) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

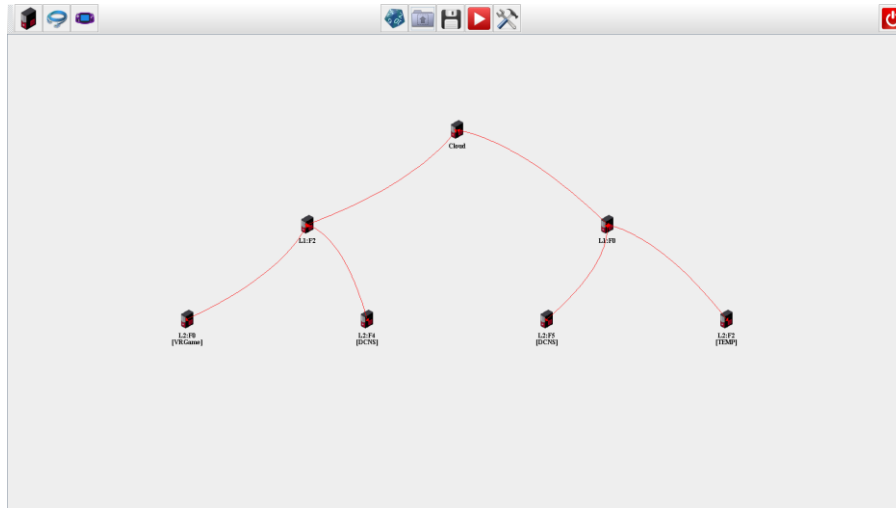
Πίνακας 5.6: Πίνακας τιμών κόστους σεναρίου 4.

	Brute Force	Random	Genetic
Latency	2	8	2
Power	0.023	0.067	0.023
Processing	0.003	0.442	0.003
Bandwidth	2.384	6.908	2.384
Συνολικό κόστος	4.416	15.417	4.416

5.4.2.5 Σενάριο 5

Το σενάριο 5 αποτελείται από 7 κόμβους, συγκεκριμένα (Σχήμα 5):

- 1 κόμβο Cloud
- 2 κόμβους Fog και
- 4 τυχαίες εφαρμογές (apps)



Σχήμα 5: Σενάριο 5

Τα κόστη (Πίνακας 5.7) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

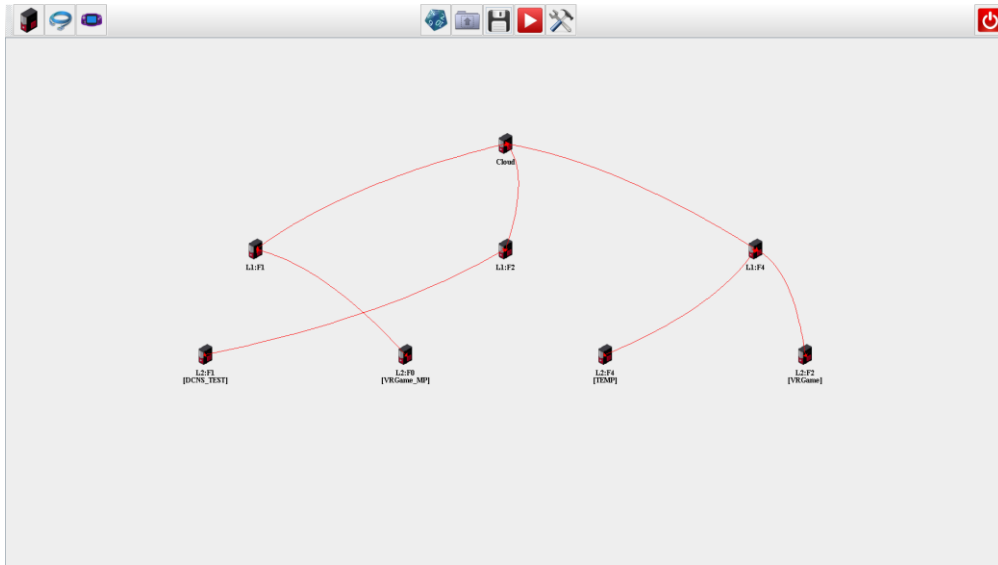
Πίνακας 5.7: Πίνακας τιμών κόστους σεναρίου 5.

	Brute Force	Random	Genetic
Latency	0	4.0	0
Power	0	0.84	0
Processing	0	0.06	0
Bandwidth	0	0.19	0
Migration	0	0.0	0
Συνολικό κόστος	0	5.09	0

5.4.2.6 Σενάριο 6

Το σενάριο 6 αποτελείται από 8 κόμβους, συγκεκριμένα (Σχήμα 6):

- 1 κόμβο Cloud
- 3 κόμβους Fog και
- 4 τυχαίες εφαρμογές (apps)



Σχήμα 6: Σενάριο 6

Τα κόστη (Πίνακας 5.8) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

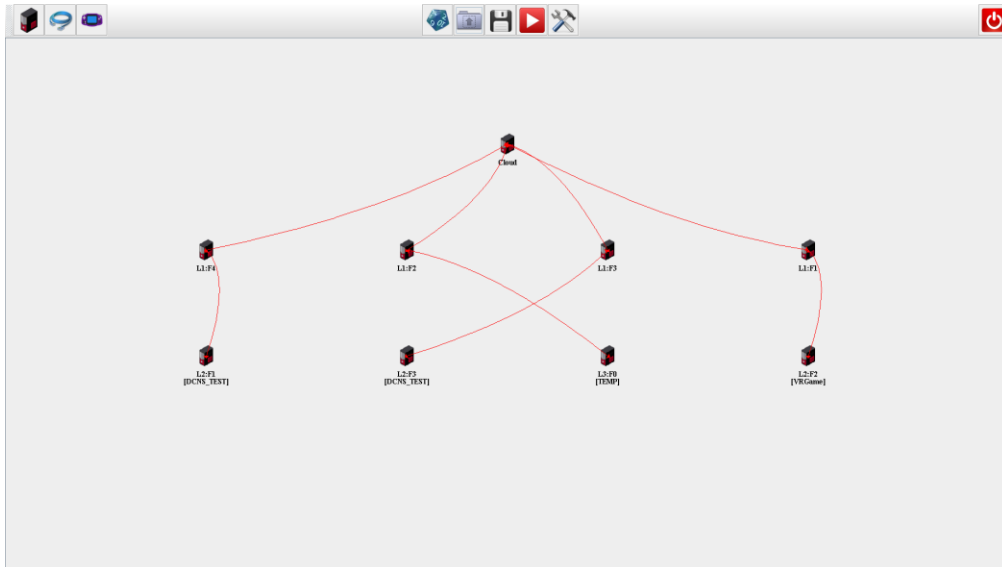
Πίνακας 5.8: Πίνακας τιμών κόστους σεναρίου 6.

	Brute Force	Random	Genetic
Latency	2	14	2
Power	0.023	0.023	0.023
Processing	0.003	0.005	0.0033
Bandwidth	2.384	0.506	2.384
Συνολικό κόστος	4.41	14.534	4.41

5.4.2.7 Σενάριο 7

Το σενάριο 7 αποτελείται από 9 κόμβους, συγκεκριμένα (Σχήμα 7):

- 1 κόμβο Cloud
- 4 κόμβους Fog και
- 4 τυχαίες εφαρμογές (apps)



Σχήμα 7: Σενάριο 7

Τα κόστη (Πίνακας 5.9) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

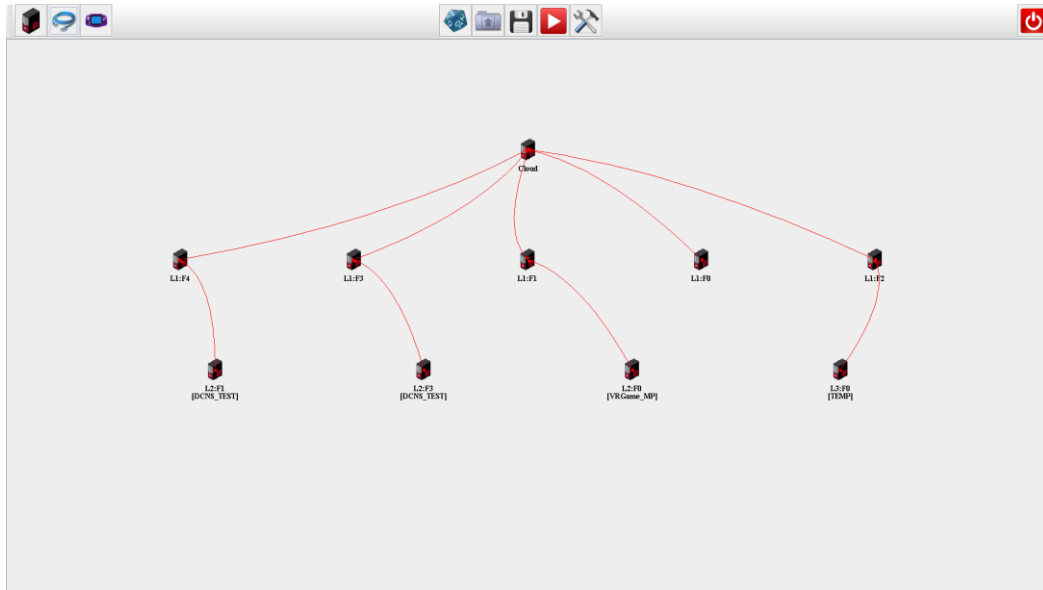
Πίνακας 5.9: Πίνακας τιμών κόστους σεναρίου 7.

	Brute Force	Random	Genetic
Latency	0	25.69	0
Power	0	0.17	0
Processing	0	0.015	0
Bandwidth	0	0.24	0
Συνολικό κόστος	0	26.115	0

5.4.2.8 Σενάριο 8

Το σενάριο 8 αποτελείται από 10 κόμβους, συγκεκριμένα (Σχήμα 8):

- 1 κόμβο Cloud
- 5 κόμβους Fog και
- 4 τυχαίες εφαρμογές (apps)



Σχήμα 8: Σενάριο 1

Τα κόστη (Πίνακας 5.10) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

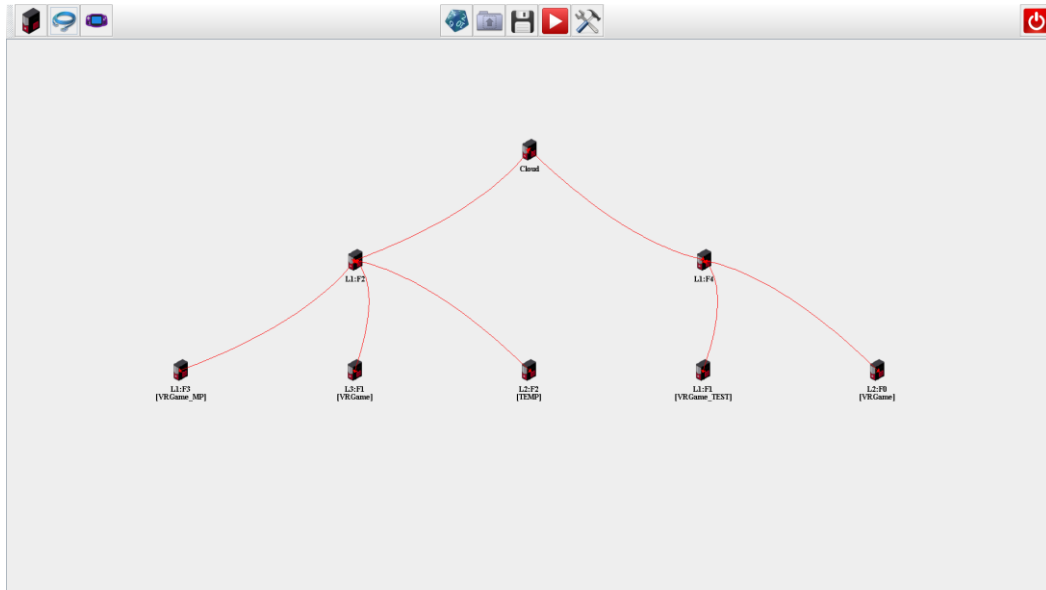
Πίνακας 5.10: Πίνακας τιμών κόστους σεναρίου 8.

	Brute Force	Random	Genetic
Latency	2	45.393	2
Power	0.023	0.0256	0.023
Processing	0.0033	0.0638	0.0033
Bandwidth	2.384	0.458	2.384
Συνολικό κόστος	4.4103	45.9404	4.4103

5.4.2.9 Σενάριο 9

Το σενάριο 9 αποτελείται από 8 κόμβους, συγκεκριμένα (Σχήμα 9):

- 1 κόμβο Cloud
- 2 κόμβους Fog και
- 5 τυχαίες εφαρμογές (apps)



Σχήμα 9: Σενάριο 2

Τα κόστη (Πίνακας 5.11) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

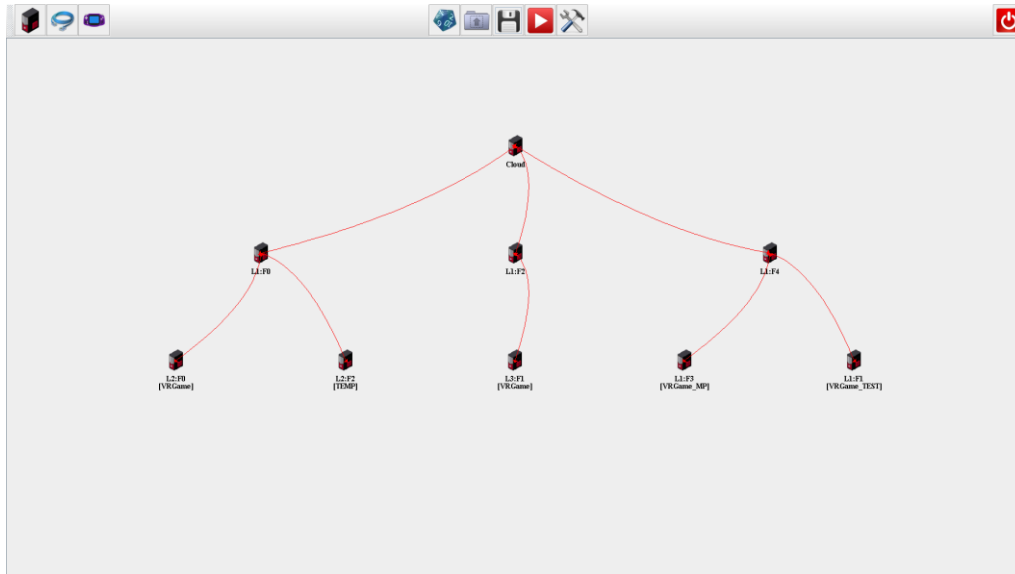
Πίνακας 5.11: Πίνακας τιμών κόστους σεναρίου 9.

	Brute Force	Random	Genetic
Latency	4	15.904	4
Power	43.56	44.83	43.56
Processing	0.067	0.068	0.067
Bandwidth	6.28	0.004	6.28
Συνολικό κόστος	53.907	60.806	53.907

5.4.2.10 Σενάριο 10

Το σενάριο 10 αποτελείται από 9 κόμβους, συγκεκριμένα (Σχήμα 10):

- 1 κόμβο Cloud
- 3 κόμβους Fog και
- 5 τυχαίες εφαρμογές (apps)



Σχήμα 10: Σενάριο 3

Τα κόστη (Πίνακας 5.12: Πίνακας τιμών κόστους σεναρίου 10. Πίνακας 5.12) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

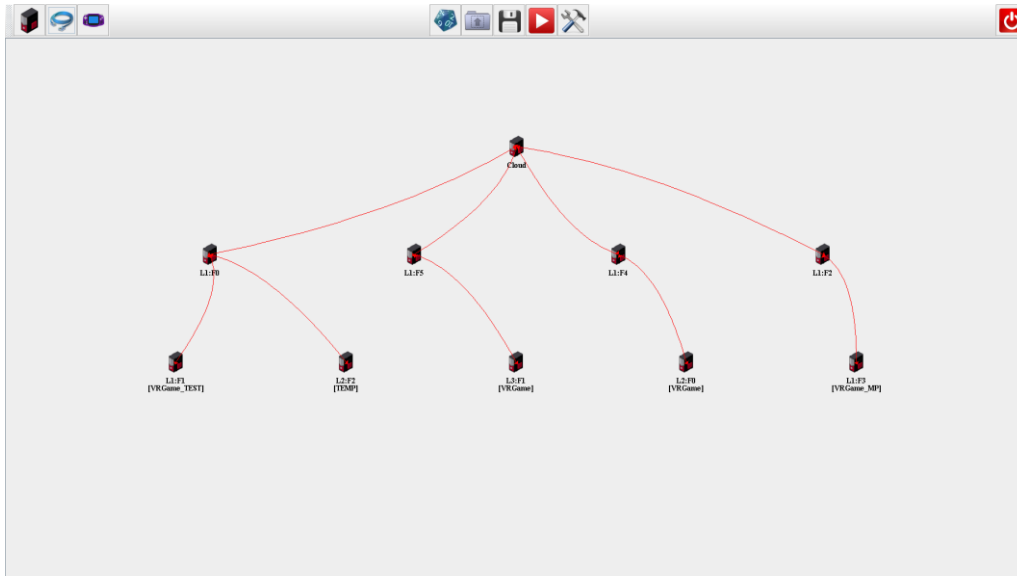
Πίνακας 5.12: Πίνακας τιμών κόστους σεναρίου 10.

	Brute Force	Random	Genetic
Latency	8	38.80	8
Power	43.56	44.93	43.56
Processing	0.067	0.263	0.067
Bandwidth	0.0021	0.023	0.0021
Συνολικό κόστος	51.6241	84.016	51.6221

5.4.2.11 Σενάριο 11

Το σενάριο 11 αποτελείται από 10 κόμβους, συγκεκριμένα (Σχήμα 11):

- 1 κόμβο Cloud
- 4 κόμβους Fog και
- 5 τυχαίες εφαρμογές (apps)



Σχήμα 11: Σενάριο 4

Τα κόστη (Πίνακας 5.13) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

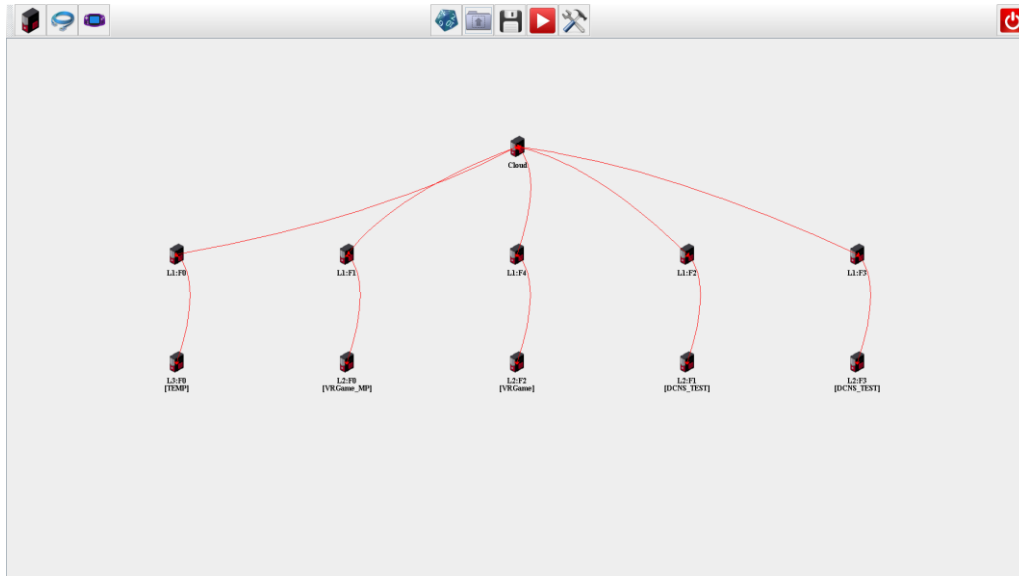
Πίνακας 5.13: Πίνακας τιμών κόστους σεναρίου 11.

	Brute Force	Random	Genetic
Latency	6	43.3	7
Power	43.56	45.25	43.56
Processing	0.062	0.569	0.06
Bandwidth	0.0021	0.025	0.0021
Συνολικό κόστος	49.6241	89.144	50.6221

5.4.2.12 Σενάριο 12

Το σενάριο 12 αποτελείται από 11 κόμβους (Σχήμα 12), συγκεκριμένα:

- 1 κόμβο Cloud
- 5 κόμβους Fog και
- 5 τυχαίες εφαρμογές (apps)



Σχήμα 12: Σενάριο 5

Τα κόστη (Πίνακας 5.14) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

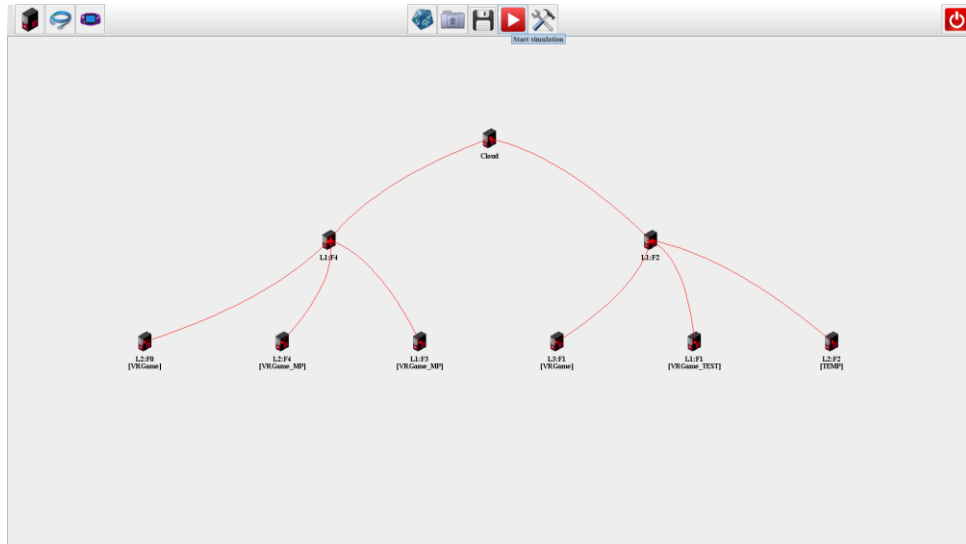
Πίνακας 5.14: Πίνακας τιμών κόστους σεναρίου 12.

	Brute Force	Random	Genetic
Latency	0	57.39	2
Power	0	0.17	0.023
Processing	0	0.02	0.003
Bandwidth	0	0.49	2.384
Συνολικό κόστος	0	58.07	4.406

5.4.2.13 Σενάριο 13

Το σενάριο 13 αποτελείται από 9 κόμβους (Σχήμα 13), συγκεκριμένα:

- 1 κόμβο Cloud
- 2 κόμβους Fog και
- 6 τυχαίες εφαρμογές (apps)



Σχήμα 13: Σενάριο 6

Τα κόστη (Πίνακας 5.15) που προκύπτουν με την εκτέλεση των τριών αλγορίθμων για το συγκεκριμένο σενάριο είναι:

Πίνακας 5.15: Πίνακας τιμών κόστους σεναρίου 13.

	Brute Force	Random	Genetic
Latency	10.9	34.26	6
Power	43.58	45.11	43.58
Processing	0.072	0.37	0.072
Bandwidth	0.001	0.03	8.66
Συνολικό κόστος	54.553	79.77	50.312

5.5 Ανάλυση αποδόσεων

Για να αναλύσουμε το συνολικό κόστος της προσομοίωσης FC, όπου συνολικό κόστος είναι το άθροισμα τιμών του κόστους καθυστέρησης, κόστους ενέργειας, κόστους επεξεργασίας και κόστους εύρους ζώνης, διαχωρίσαμε τα σενάρια με μεταβλητό αριθμό κόμβων Fog και έχοντας τον ίδιο αριθμό apps, που συνδέονται σε αυτές.

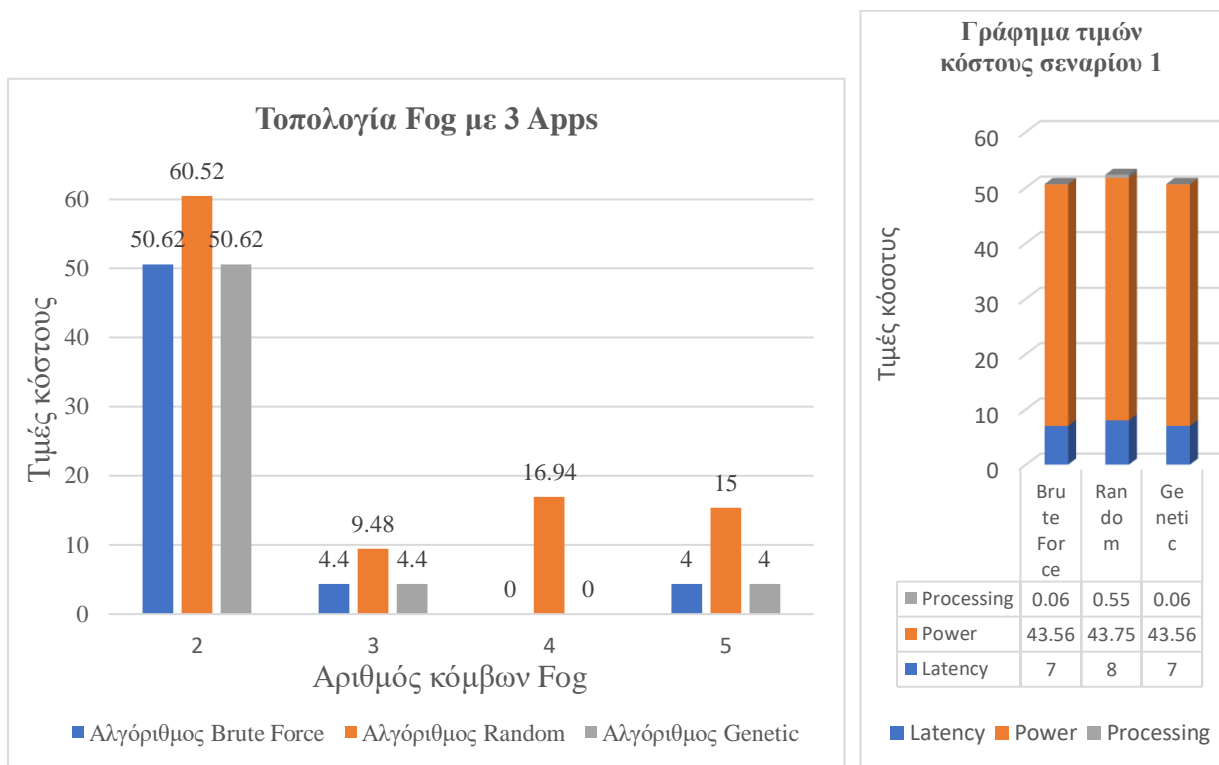
Ακολουθούν οι πίνακες τιμών κόστους των σεναρίων βάση της κατάταξης αυτής.

- Το συνολικό κόστος (Πίνακας 5.16) των προσομοιώσεων με τρεις συσκευές apps είναι:

Πίνακας 5.16: Συνολικό κόστος προσομοίωσης με τρία Apps.

Σενάριο	Αριθμός συσκευών Fog	Αλγόριθμος Brute Force	Αλγόριθμος Random	Αλγόριθμος Genetic
1	2	50.6241	60.52	50.6241
2	3	4.410	9.48	4.410
3	4	0	16.943	0
4	5	4.416	15.417	4.416

Η γραφική αναπαράσταση (Εικόνα 5.6) του πίνακα τιμών για μεταβλητό αριθμό συσκευών Fog και τρεις συσκευές apps είναι:



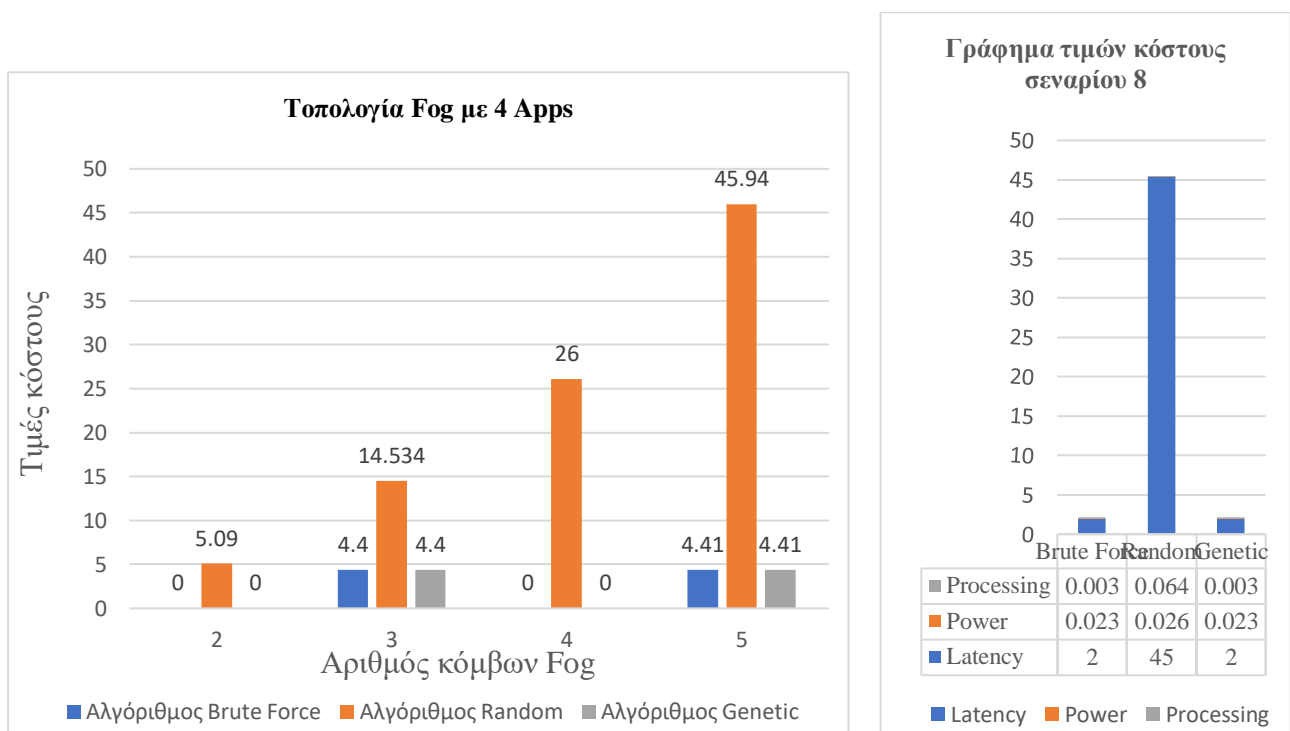
Εικόνα 5.6 (α) Τιμές κόστους με 3 Apps και μεταβλητό αριθμό συσκευών Fog (β) Γράφημα τιμών κόστους σεναρίου 1.

➤ Το συνολικό κόστος (Πίνακας 5.17) των προσομοιώσεων με τέσσερις συσκευές apps είναι:

Πίνακας 5.17: Συνολικό κόστος προσομοίωσης με τέσσερα Apps.

Σενάριο	Αριθμός συσκευών Fog	Αλγόριθμος Brute Force	Αλγόριθμος Random	Αλγόριθμος Genetic
5	2	0	5.09	0
6	3	4.41	14.534	4.41
7	4	0	26.115	0
8	5	4.4103	45.9404	4.4103

Η γραφική αναπαράσταση (Εικόνα 5.7) του πίνακα τιμών για μεταβλητό αριθμό συσκευών Fog και τέσσερις συσκευές apps είναι:



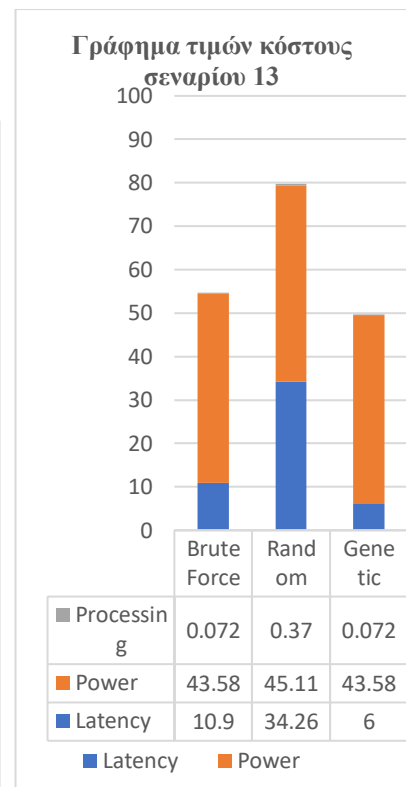
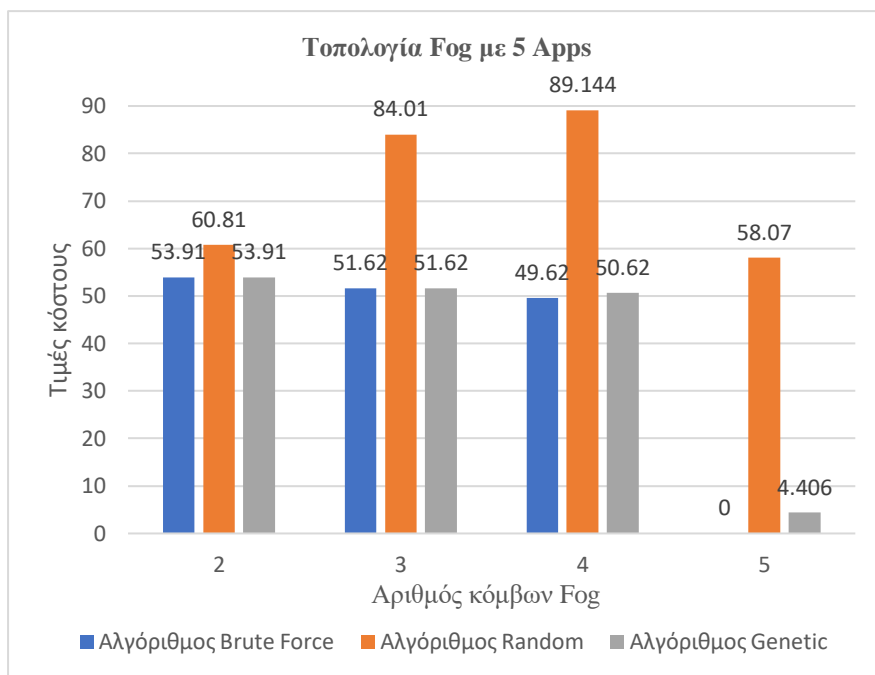
Εικόνα 5.7: (α) Τιμές κόστους με 4 Apps και μεταβλητό αριθμό συσκευών Fog. (β) Γράφημα τιμών κόστους σεναρίου 8.

➤ Το συνολικό κόστος (Πίνακας 5.18) των προσομοιώσεων με πέντε συσκευές apps είναι:

Πίνακας 5.18: Συνολικό κόστος προσομοίωσης με πέντε Apps.

Σενάριο	Αριθμός συσκευών Fog	Αλγόριθμος Brute Force	Αλγόριθμος Random	Αλγόριθμος Genetic
9	2	53.91	60.81	53.91
10	3	51.62	84.02	51.62
11	4	49.6241	89.144	50.6221
12	5	0	58.07	4.406

Η γραφική αναπαράσταση (Εικόνα 5.8) του πίνακα τιμών για μεταβλητό αριθμό συσκευών Fog και πέντε συσκευές apps είναι:



Εικόνα 5.8: (α) Τιμές κόστους με 4 Apps και μεταβλητό αριθμό συσκευών Fog. (β) Γράφημα τιμών κόστους σεναρίου 13.

Στην Εικόνα 5.6 (α) παρατηρούμε ότι το κόστος στους αλγορίθμους brute force και genetic μηδενίζεται, ενώ στον αλγόριθμο random το κόστος αυξάνεται όταν ο αριθμός των συσκευών Fog είναι κατά μία περισσότερες από τον αριθμό των apps και η σύνδεσή τους είναι ένα app προς μία συσκευή Fog. Ακόμη το κόστος αυξάνεται σταδιακά και στους τρεις αλγορίθμους όσο αυξάνεται ο αριθμός των συσκευών Fog που δε συνδέονται με κάποιο app. Το κόστος αυξάνεται πάρα πολύ όταν σε μία συσκευή Fog συνδέονται δύο apps ενώ στην άλλη ένα app, παρατηρούμε δηλαδή ότι η ανομοιομορφία των συνδέσεων προκαλεί αυξημένο κόστος και στους τρεις αλγορίθμους.

Στο γράφημα τιμών κόστους του σεναρίου 1 (Εικόνα 5.6 (β)) βλέπουμε ότι στις τρεις αλγοριθμικές περιπτώσεις το κόστος της ενέργειας είναι αρκετά υψηλό, το κόστος latency παρουσιάζει μία σχετική αύξηση, ενώ το κόστος processing είναι μηδαμινό εκτός από τον αλγόριθμο random που έχει μία σχετική αύξηση.

Στην Εικόνα 5.7 (α) παρατηρούμε στους αλγορίθμους brute force και genetic το κόστος να μηδενίζεται όταν έχουμε τη σύνδεση δύο app σε μία συσκευή Fog ή όταν είναι ένα apps σε μία συσκευή Fog. Ακόμη στον αλγόριθμο random όταν συνδέονται δύο app σε μία συσκευή Fog το κόστος βελτιστοποιείται. Ενώ το κόστος και στους τρεις αλγορίθμους αυξάνεται όταν σε μία συσκευή Fog συνδέονται δύο apps ενώ στην άλλη ένα app. Πολύ μεγάλη αύξηση έχουμε όταν τρία apps συνδέονται

σε μία συσκευή Fog και ειδικά στον αλγόριθμο random όταν αυξάνεται ο αριθμός των συσκευών και μία συσκευή Fog παραμένει χωρίς να έχει κάποια σύνδεση.

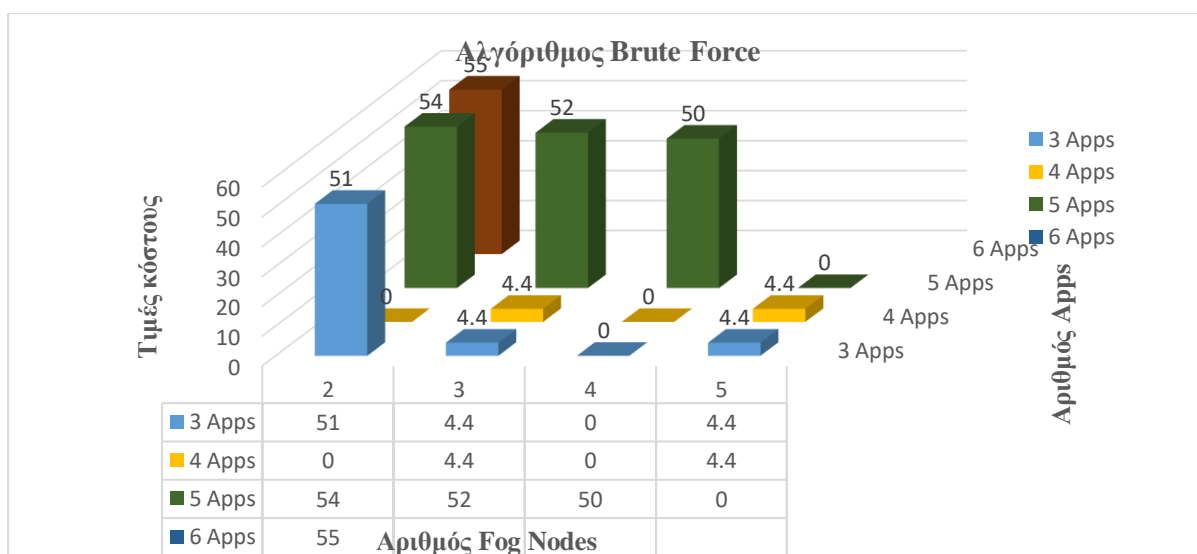
Στην περίπτωση του γραφήματος τιμών κόστους του σεναρίου 8 (Εικόνα 5.7 (β)) παρατηρούμε ότι στους αλγόριθμους brute force και genetic τα κόστη latency, power και processing είναι σχετικά χαμηλά. Στον αλγόριθμο random το κόστος latency είναι πολύ υψηλό και τα κόστη power και processing είναι ελάχιστα. Συμπεραίνουμε ότι είναι η πιο ευνοϊκή περίπτωση σεναρίου όταν δηλαδή, ένα app συνδέεται σε ένα Fog και ειδικά, στις τιμές κόστους των αλγορίθμων brute force και genetic.

Παρατηρώντας στη Εικόνα 5.8 (α) συμπεραίνουμε ότι το κόστος αυξάνεται αρκετά όταν υπάρχει διαφορετικός αριθμός apps που συνδέονται σε κάθε συσκευή Fog και όταν τρία app συνδέονται σε μία συσκευή Fog. Ενώ στον αλγόριθμο brute force το κόστος παραμένει μηδέν όταν συνδέεται ένα app σε μία συσκευή Fog.

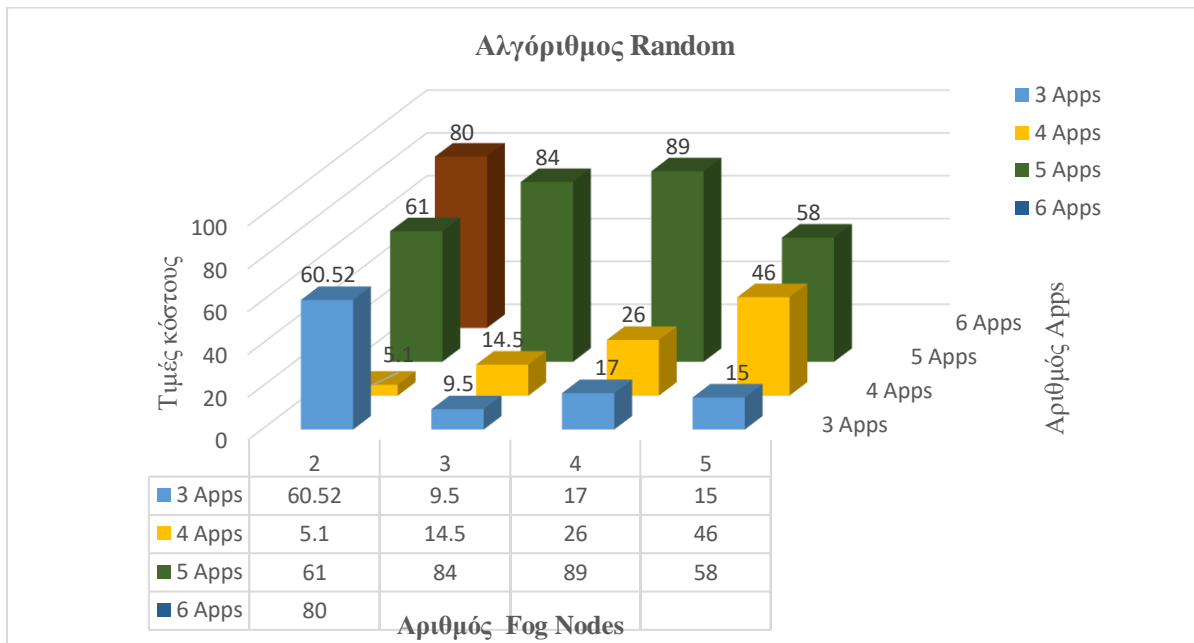
Στο γράφημα τιμών κόστους του σεναρίου 13 (Εικόνα 5.8) παρατηρούμε ότι και στους τρεις αλγόριθμους το κόστος power και latency είναι αρκετά υψηλό, ιδιαίτερα μεγάλη αύξηση στον αλγόριθμο random. Ενώ το κόστος processing είναι ελάχιστο με μία μικρή άνοδο στον αλγόριθμο random.

5.6 Τιμές κόστους ανά αλγόριθμο

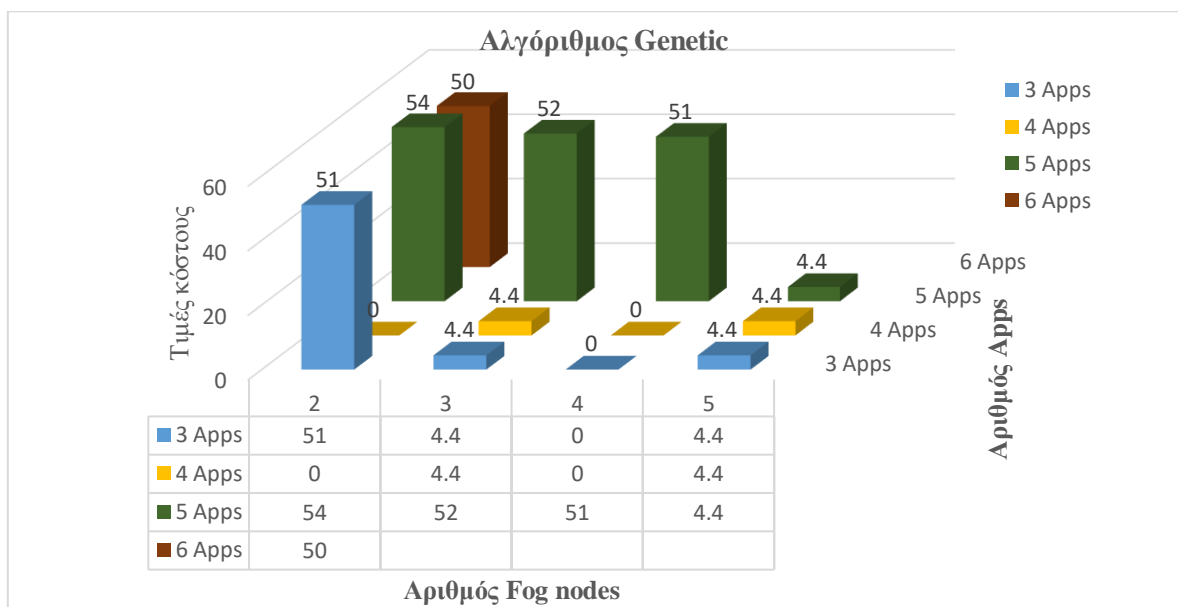
Στην ενότητα αυτή παρουσιάζονται οι γραφικές αναπαραστάσεις των τιμών κόστους ανά αλγόριθμο (Εικόνα 5.9 -Εικόνα 5.10 Εικόνα 5.11):



Εικόνα 5.9:Γράφημα τιμών κόστους αλγορίθμου Brute Force.



Εικόνα 5.10: Γράφημα τιμών κόστους αλγορίθμου Random.



Εικόνα 5.11: Γράφημα τιμών κόστους αλγορίθμου Genetic.

Όπως αποτυπώνεται στα γραφήματα 5.9 - 5.11, παρατηρούμε ότι:

1. Όταν ο αριθμός των κόμβων Fog είναι ο ίδιος ή πλησιάζει τον αριθμό των apps το κόστος είναι κοντά στη τιμή μηδέν.
2. Όταν ο αριθμός των συσκευών Fog είναι μικρός το κόστος αυξάνεται ενώ αντίθετα βελτιώνεται και στους τρεις αλγορίθμους όταν αυξάνεται ο αριθμός των συσκευών Fog.
3. Οι καλύτερες περιπτώσεις στους αλγορίθμους Brute force και Genetic είναι όταν αναλογεί ένα app σε ένα Fog αλλά και όταν συνδέονται 2 apps σε ένα Fog
4. Η τιμή του κόστους γίνεται υψηλή όταν ο αριθμός των apps που συνδέονται σε ένα κόμβο Fog είναι μεγαλύτερος από δύο.
5. Στον αλγόριθμο random εάν υπάρχει μία συσκευή Fog χωρίς apps τότε έχουμε υψηλό κόστος.
6. Ο αλγόριθμος random αποτελεί τη χειρότερη περίπτωση αλγορίθμου ενδεχομένως, γιατί τοποθετεί τυχαία τις εφαρμογές πάνω στις συσκευές Fog.
7. Εάν συνδέεται διαφορετικός αριθμός apps σε κάθε συσκευή Fog αυξάνεται το κόστος.

5.7 Συμπεράσματα

Το FC αναδεικνύεται ως ελκυστική λύση σε προβλήματα επεξεργασίας δεδομένων με υψηλές απαιτήσεις στο IoT. Χρησιμοποιούν συσκευές με μεγάλη υπολογιστική ισχύ στην άκρη του δικτύου και κοντά στο IoT, αντί να αναθέτουν τις λειτουργίες στο Cloud, μειώνοντας την καθυστέρηση, τη συμφόρηση στο δίκτυο και βελτιώνοντας την ποιότητα των υπηρεσιών. Σε αυτή την ενότητα παρουσιάζουμε το προσομοιωτή FogComputingSim και δημιουργούμε περιβάλλοντα με συσκευές Cloud, συσκευές Fog και apps.

Συγκεκριμένα, ο προσομοιωτής FogComputingSim επιτρέπει την έρευνα και τη σύγκριση των τεχνικών διαχείρισης πόρων, βάση των QoS κριτηρίων όπως καθυστέρηση, bandwidth, RAM και CPU. Με τη προσομοίωση σχεδιάσαμε και περιγράψαμε δεκατρία σενάρια και ο προσομοιωτής απέδειξε την αποτελεσματικότητα του στην διαχείριση των πόρων και στον υπολογισμό του συνολικού κόστους, ώστε σε εφαρμογές του συστήματος Fog, να επιτύχουμε την ελαχιστοποίηση του κόστους και την αποδοτικότερη συνδεσιμότητα των συσκευών apps και Fog.

Οι αλγόριθμοι Brute force και Genetic είναι οι πιο αποτελεσματικοί, δίνοντας τη βέλτιστη και αποδοτική λύση στην εφαρμογή των σεναρίων προσομοίωσης, στον υπολογισμό των μονάδων κόστους και στο συνολικό κόστος του συστήματος γιατί, πραγματοποιούν εκτενή ανανέωση και έλεγχο των βέλτιστων και χαμηλότερων τιμών κόστους. Οι δύο προηγούμενοι αλγόριθμοι φαίνεται να έχουν το ελάχιστο κόστος σε περιπτώσεις ομοιόμορφης κατανομής στη σύνδεση των συσκευών apps με τις συσκευές Fog και αναμενόμενη αύξηση κατά την προσθήκη επιπλέον συσκευών στο σύστημα Fog. Το κόστος αυξάνεται αρκετά εάν κάποιοι κόμβοι Fog παραμένουν σε αδράνεια, δηλαδή όταν δε συνδέεται σε αυτούς κάποιο app.

Ο αλγόριθμος random επιφέρει μεγαλύτερο κόστος γιατί πιθανόν να μην ανταποκρίνεται σωστά στη λειτουργία των προσομοιώσεων για την ελαχιστοποίηση του κόστους του συστήματος, καθώς κάνει τυχαία επιλογή τιμών και δεν μπορούν να είναι οι βέλτιστες τιμές κόστους.

Στις περιπτώσεις των γραφημάτων των τιμών κόστους των σεναρίων 1, 8 και 13 παρατηρούμε ότι οι δύο αλγόριθμοι δίνουν τις βέλτιστες τιμές κόστους για τις περιπτώσεις των σεναρίων που ένα app συνδέεται σε ένα Fog, όπως στην περίπτωση του σεναρίου 8, με εξαίρεση τον αλγόριθμο random που δίνει πολλή υψηλή τιμή κόστους latency, καθώς η επιλογή τιμών δε βασίζεται στις βέλτιστες τιμές κόστους, αλλά σε τυχαία επιλογή τους.

6 Συμπεράσματα

Το FC βρίσκεται ενδιάμεσα του Cloud και των συσκευών IoT για να παρέχει υπηρεσίες στους χρήστες όπως υπολογισμό, δικτύωση, αποθήκευση, επεξεργασία, συλλογή και ανταλλαγή δεδομένων, καθώς οι απαιτήσεις των χρηστών για επεξεργασία δεδομένων αυξάνονται καθημερινά. Το FC βρίσκει εφαρμογές όπως στα έξυπνα δίκτυα και έξυπνες πόλεις, συνδεδεμένα οχήματα και αναλύσεις σε πραγματικό χρόνο, με αποτέλεσμα οι συσκευές IoT να αλληλοεπιδρούν με εφαρμογές που είναι ευαίσθητες στην καθυστέρηση και να απαιτούν γρήγορη επεξεργασία δεδομένων και χαμηλή κατανάλωση ενέργειας.

Το FC αποτελείται από διάσπαρτους κόμβους Fog που αναμένουν να εξυπηρετήσουν τα αιτήματα των χρηστών κάνοντας χρήση των τοπικών υπολογιστικών πόρων αντί του Cloud, μειώνοντας την καθυστέρηση, εξοικονομώντας εύρος ζώνης, υποστηρίζοντας τη διαλειτουργικότητα, την ασφάλεια των δεδομένων και αυξάνοντας την απόδοση του συστήματος. Η γεωγραφική εγγύτητα του FC μειώνει τον λανθάνοντα χρόνο μετάδοσης, κάνοντας κατάλληλη αξιοποίηση των τοπικών πόρων των υπολογιστικών συστημάτων για την επεξεργασία των εργασιών που αιτείται.

Όμως οι αυξανόμενες IoT απαιτήσεις κάνουν την αποτελεσματική εκχώρηση των πόρων, την επίτευξη της μέγιστης απόδοσης και την ικανοποίηση των αιτημάτων των χρηστών να είναι πολύπλοκο πρόβλημα με αυξημένο κόστος υποδομής, ενέργειας, επικοινωνίας, επεξεργασίας και υπολογισμού στα περιβάλλοντα Fog.

Στην εργασία αυτή αναλύονται διάφορες πτυχές του κόστους στο FC και των υπολογιστικών τεχνολογιών, προσεγγίζοντας διάφορα μοντέλα, με τις παραμέτρους που τα ορίζουν (Πίνακας 6.1). Συγκεκριμένα παρουσιάσαμε το κόστος υποδομής του δικτύου διαχωρίζοντάς το στο κεφαλαιουχικό και στα λειτουργικά έξοδα με τις παραμέτρους υπολογισμού των μοντέλων και τη μείωση της κατανάλωσης ενέργειας και διερευνάμε το πρόβλημα της βελτιστοποίησης σε συνδυασμό με την απόδοση του συστήματος.

Ακόμη αναφερθήκαμε στην έννοια του κόστους επικοινωνίας, τις τεχνολογίες διασύνδεσης και δικτύωσης στο σύστημα FC. Εξετάσαμε το κόστος επικοινωνίας στο σύστημα Fog, αναλύοντας τον τρόπο δόμησης του συστήματος και τις παραμέτρους που το επηρεάζουν. Αναλύσαμε διάφορα μοντέλα που βασίζονται στην αρχιτεκτονική αυτών των συστημάτων, λαμβάνοντας υπόψη τα στοιχεία που τα συνθέτουν και τις παραμέτρους υπολογισμού του κόστους.

Ο υπολογισμός πραγματοποιείται σε συσκευές αιχμής, ενδιάμεσους κόμβους ή σε κέντρα δεδομένων Cloud, ανάλογα με τις τρέχουσες ανάγκες και τη διαθεσιμότητα των πόρων και δημιουργείται το απαιτητικό πρόβλημα της διαχείρισης των πόρων, επεξεργασίας, αποθήκευσης και εκτέλεσης των εργασιών μέσα στα επιθυμητά πλαίσια τιμών και με το βέλτιστο κόστος.

Στη συνέχεια εξετάσαμε το συνολικό κόστος δικτύου, δημιουργώντας διάφορες τοπολογίες δικτύου, με τον προσομοιωτή FogComputingSim, έχοντας διαφορετικό αριθμό εφαρμογών, κόμβων Fog και τρόπο σύνδεσης των συσκευών. Κάναμε σύγκριση, των τιμών κόστους και την απόδοσή τους στις βασικότερες τοπολογίες και αναλύσαμε συμπεριφορά του κόστους στους τρεις αλγορίθμους.

Η εργασία αυτή ασχολείται με το κόστος και εξετάζει ανοιχτές προοπτικές και μελλοντικές κατευθύνσεις έρευνας, ορίζοντας διαφορετική αξιολόγηση με παράγοντες που συνήθως, χρησιμοποιούν προσομοιωτές ως εργαλεία αξιολόγησης. Προσπαθώντας να διακρίνουμε τις βέλτιστες παραμέτρους ελαχιστοποίησης του κόστους, που βελτιώνουν τους μηχανισμούς διαχείρισης πόρων και επικεντρώνονται στην ελαχιστοποίηση των βασικών πτυχών του κόστους.

Από το σύνολο των μοντέλων, συμπεραίνουμε ότι το FC είναι μία νέα τεχνολογία, με διαφορετικές προσεγγίσεις και παραμέτρους που λήφθηκαν υπόψη και αποτέλεσαν πρόκληση και αναγκαιότητα ανάλυσης τους, για τον υπολογισμό του κόστους, ώστε σε συνδυασμό με τους πολλούς περιορισμούς του κάθε μοντέλου, να διακρίνουμε εκείνα τα κόστη που πραγματικά επηρεάζουν την αρχιτεκτονική Fog, όπου βελτιώνοντάς τα θα μπορούσαμε να επιτύχουμε την ελαχιστοποίηση του κόστους και την απόδοση του FC.

Πίνακας 6.1: Προσεγγίσεις μοντέλων κόστους στο FC.

	Infrastructure		Network		Computational					
	Operational	Power/ Energy	Network usage	Latency	Resource allocation	Task scheduling	Migration	Offloading	Security	Mobile computing
Ahvar 2019, [19]		X	X							
Ali 2020, [15]		X	X	X						
Battula 2019, [8]	X	X	X	X	X		X			
Bellendorf 2020, [18]	X	X		X						
Brogi 2018, [13]	X									
He 2012, [10]			X							
Karakus 2018), [9]	X									
Lin 2018, [12]	X		X	X						
Liu 2017, [29]				X		X				
Liu 2019, [16]			X	X						
Loffi.2021,[37]									X	
Mao 2021, [26]		X	X							X
Motiwala 2019, [28]		X	X							
Pham 2017, [31]			X			X				
Rah 2017, [33]			X			X				
Rejiba 2019, [3]				X			X			
Ren 2019, [20]				X						
Suryadevara 2021, [17]		X	X							
Wang 2021, [23]			X					X		
Yousefpour 2018, [22]	X									
Yu 2017, [32]		X		X	X					
Zhang 2017, [11]	X		X							

Μελλοντικές προσεγγίσεις

Δεδομένου ότι έχουν εντοπισθεί και επισημανθεί οι βασικοί παράγοντες κόστους σε μια ανάπτυξη συστημάτων FC, μελλοντικά θα μπορούσε να υλοποιηθεί ένα ενιαίο μοντέλο υπολογισμού κόστους.

Συγκεκριμένα θα μπορούσε να υλοποιηθεί ένα μοντέλο κόστους:

- Που θα συμπεριλαμβάνει όλες τις βασικές πηγές κόστους: δίκτυο, υποδομή, υπολογισμούς.
- Θα επιτρέπει στο χρήστη το επίπεδο της λεπτομέρειας ανάλυσης κόστους για κάθε βασική πηγή.
- Θα εξετάσει την εφαρμογή αλγορίθμων βελτιστοποίησης, για τη μείωση του κόστους.
- Θα εξετάσει την επιμέρους μείωση του κόστους, για παράδειγμα το κόστος που αφορά την ενέργεια.
- Η συνολική βελτιστοποίηση θα μπορούσε να εφαρμοστεί σε μια κατανομημένη προσέγγιση προκειμένου να αυξηθεί η επεκτασιμότητα και η αποδοτικότητα των πόρων.

BIBΛΙΟΓΡΑΦΙΑ

- [1]. Abouaoumar, A., Cherkaoui, S., Kobbane, A., & Dambri, O. A. (2019, December). A resources representation for resource allocation in Fog computing networks. In *2019 IEEE Global Communications Conference (GLOBECOM)* (pp. 1-6). IEEE.
- [2]. Margariti, S. V., Dimakopoulos, V. V., & Tsoumanis, G. (2020). Modeling and Simulation Tools for Fog Computing—A Comprehensive Survey from a Cost Perspective. *Future Internet*, *12*(5), 89.
- [3]. Rejiba, Z., Masip-Bruin, X., & Marín-Tordera, E. (2019). A survey on mobility-induced service migration in the fog, edge, and related computing paradigms. *ACM Computing Surveys (CSUR)*, *52*(5), 1-33.
- [4]. Khan, S., Parkinson, S., & Qin, Y. (2017). Fog computing security: a review of current applications and security solutions. *Journal of Cloud Computing*, *6*(1), 1-22.
- [5]. Chiang, M., Ha, S., Risso, F., Zhang, T., & Chih-Lin, I. (2017). Clarifying Fog computing and networking: 10 questions and answers. *IEEE Communications Magazine*, *55*(4), 18-20.
- [6]. Xu, X., Fu, S., Cai, Q., Tian, W., Liu, W., Dou, W., ... & Liu, A. X. (2018). Dynamic resource allocation for load balancing in Fog environment. *Wireless Communications and Mobile Computing*, 2018.
- [7]. Liu, Y., Fieldsend, J. E., & Min, G. (2017). A framework of Fog computing: Architecture, challenges, and optimization. *IEEE Access*, *5*, 25445-25454.
- [8]. Battula, S. K., Garg, S., Naha, R. K., Thulasiraman, P., & Thulasiram, R. (2019). A micro-level compensation-based cost model for resource allocation in a Fog environment. *Sensors*, *19*(13), 2954.
- [9]. Karakus, M., & Duresi, A. (2017, March). Service cost in software defined networking (SDN). In *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)* (pp. 468-475). IEEE.
- [10]. He, Z., Zhang, Y., Tak, B. and Peng, L., 2019. Green fog planning for optimal internet-of-thing task scheduling. *IEEE Access*, *8*, pp.1224-1234.
- [11]. Zhang, W., Lin, B., Yin, Q., & Zhao, T. (2017). Infrastructure deployment and optimization of Fog network based on MicroDC and LRPN integration. *Peer-to-Peer Networking and Applications*, *10*(3), 579-591.
- [12]. Lin, C. C., & Yang, J. W. (2018). Cost-efficient deployment of Fog computing systems at logistics centers in industry 4.0. *IEEE Transactions on Industrial Informatics*, *14*(10), 4603-4611.
- [13]. Brogi, A., Forti, S., & Ibrahim, A. (2018). Deploying Fog applications: How much does it cost, by the way?. In *CLOSER* (pp. 68-77).
- [14]. Mahmud, R., Ramamohanarao, K., & Buyya, R. (2018). Latency-aware application module management for Fog computing environments. *ACM Transactions on Internet Technology (TOIT)*, *19*(1).
- [15]. Ali, B., Pasha, M. A., ul Islam, S., Song, H., & Buyya, R. (2020). A Volunteer-Supported Fog Computing Environment for Delay-Sensitive IoT Applications. *IEEE Internet of Things Journal*, *8*(5), 3822-3830.
- [16]. Liu, L., Chang, Z., Guo, X., Mao, S., & Ristaniemi, T. (2017). Multiobjective optimization for computation offloading in Fog computing. *IEEE Internet of Things Journal*, *5*(1), 283-294.
- [17]. Suryadevara, N. K. (2021). Energy and Latency reductions at the Fog gateway using a Machine Learning classifier. *Sustainable Computing: Informatics and Systems*, 100582.
- [18]. Bellendorf, J., & Mann, Z. Á. (2020). Classification of optimization problems in fog computing. *Future Generation Computer Systems*, *107*, 158-176.
- [19]. Ahvar, E., Orgerie, A. C., & Lebre, A. (2019). Estimating energy consumption of cloud, fog and edge computing infrastructures. *IEEE Transactions on Sustainable Computing*.
- [20]. Ren, J., Zhang, D., He, S., Zhang, Y., & Li, T. (2019). A survey on end-edge-cloud orchestrated network computing paradigms: Transparent computing, mobile edge computing, fog computing, and cloudlet. *ACM Computing Surveys (CSUR)*, *52*(6), 1-36.
- [21]. Chiang, M., Ha, S., Risso, F., Zhang, T., & Chih-Lin, I. (2017). Clarifying Fog computing and networking: 10 questions and answers. *IEEE Communications Magazine*, *55*(4), 18-20.

- [22]. Yousefpour, A., Ishigaki, G., Gour, R., & Jue, J. P. (2018). On reducing IoT service delay via Fog offloading. *IEEE Internet of Things Journal*, 5(2), 998-1010.
- [23]. Wang, S., Ruan, Y., Tu, Y., Wagle, S., Brinton, C. G., & Joe-Wong, C. (2021). Network-aware optimization of distributed learning for Fog computing. *IEEE/ACM Transactions on Networking*.
- [24]. Luan, T. H., Gao, L., Li, Z., Xiang, Y., Wei, G., & Sun, L. (2015). Fog computing: Focusing on mobile users at the edge. *arXiv preprint arXiv:1502.01815*.
- [25]. Neware, R., & Shrawankar, U. (2020). Fog computing architecture, applications and security issues. *International Journal of Fog Computing (IJFC)*, 3(1), 75-105.
- [26]. Mao, Y., You, C., Zhang, J., Huang, K. and Letaief, K.B., 2017. A survey on mobile edge computing: The communication perspective. *IEEE Communications Surveys & Tutorials*, 19(4), pp.2322-2358.
- [27]. Parvez, I., Rahmati, A., Guvenc, I., Sarwat, A. I., & Dai, H. (2018). A survey on low latency towards 5G: RAN, core network and caching solutions. *IEEE Communications Surveys & Tutorials*, 20(4), 3098-3130.
- [28]. Motiwala, M., Dhamdhare, A., Feamster, N., & Lakhina, A. (2012). Towards a cost model for network traffic. *ACM SIGCOMM Computer Communication Review*, 42(1), 54-60.
- [29]. Liu, Z., Yang, Y., Chen, Y., Li, K., Li, Z., & Luo, X. (2019, April). A multi-tier cost model for effective user scheduling in Fog computing networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs)* (pp. 1-6). IEEE.
- [30]. He, J., Wei, J., Chen, K., Tang, Z., Zhou, Y., & Zhang, Y. (2017). Multitier Fog computing with large-scale iot data analytics for smart cities. *IEEE Internet of Things Journal*, 5(2), 677-686.
- [31]. Pham, X. Q., Man, N. D., Tri, N. D. T., Thai, N. Q., & Huh, E. N. (2017). A cost-and performance-effective approach for task scheduling based on collaboration between cloud and Fog computing. *International Journal of Distributed Sensor Networks*, 13(11), 1550147717742073.
- [32]. Yu, L., Jiang, T., & Zou, Y. (2017). Fog-assisted operational cost reduction for cloud data centers. *IEEE Access*, 5, 13578-13586.
- [33]. Rahbari, D., & Nickray, M. (2017, November). Scheduling of Fog networks with optimized knapsack by symbiotic organisms search. In *2017 21st Conference of Open Innovations Association (FRUCT)* (pp. 278-283). IEEE.
- [34]. Naha, R. K., Garg, S., Georgakopoulos, D., Jayaraman, P. P., Gao, L., Xiang, Y., & Ranjan, R. (2018). Fog computing: Survey of trends, architectures, requirements, and research directions. *IEEE access*, 6, 47980-48009.
- [35]. Rejiba, Z., Masip-Bruin, X., & Marín-Tordera, E. (2019). A survey on mobility-induced service migration in the fog, edge, and related computing paradigms. *ACM Computing Surveys (CSUR)*, 52(5), 1-33.
- [36]. Khan, S., Parkinson, S., & Qin, Y. (2017). Fog computing security: a review of current applications and security solutions. *Journal of Cloud Computing*, 6(1), 1-22.
- [37]. Loffi, L., Westphall, C. M., Grütner, L. D., & Westphall, C. B. (2021). Mutual authentication with multi-factor in IoT-Fog-Cloud environment. *Journal of Network and Computer Applications*, 176, p.102932
- [38]. Mahmud, R., & Buyya, R. (2019). Modelling and simulation of fog and edge computing environments using iFogSim toolkit. *Fog and edge computing: Principles and paradigms*, 1-35.
- [39]. LoseCviera, FogComputingSim, <https://github.com/JoseCVieira/FogComputingSim>, 20/06/2021.
- [40]. Balasubramaniyam, H., 2019. *The architecture of fog network - A bridge between Cloud and IoT (Part 2)*. [online] Medium. Available at: <<https://hindujab.medium.com/the-architecture-of-fog-network-a-bridge-between-cloud-and-iot-part-2-a45612145a0b>> [Accessed 27 September 2021].
- [41]. Gupta, H., Vahid Dastjerdi, A., Ghosh, S. K., & Buyya, R. (2017). iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments. *Software: Practice and Experience*, 47(9), 1275-1296.

ΠΑΡΑΡΤΗΜΑ Α

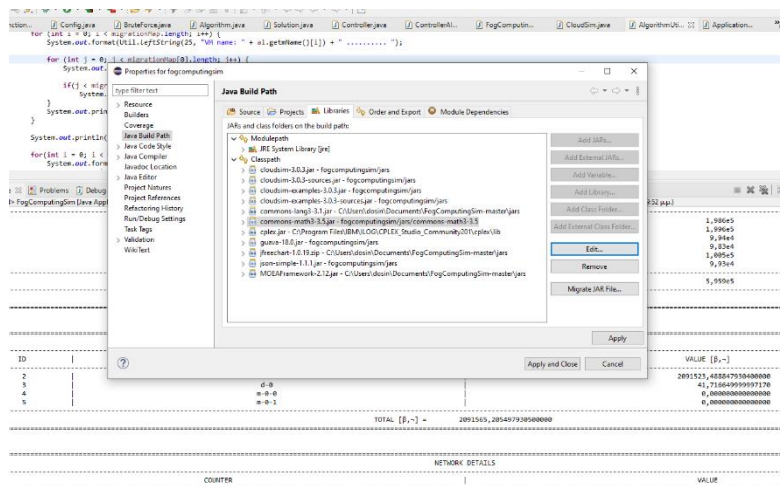
Εγκατάσταση του προσομοιωτή FogComputingSim

Τα βήματα εγκατάστασης της προσομοίωσης είναι:

3. Ο σύνδεσμος εγκατάστασης του κώδικα FogComputingSim είναι:

<https://github.com/JoseCVieira/FogComputingSim>

1. Εξάγουμε το zip αρχείο με όνομα *FogComputingSim-master*.
2. Εγκαθιστούμε το Java Standard Edition Development Kit (jdk) και το Eclipse στον υπολογιστή.
3. Ορίζουμε workspace για το Eclipse.
4. Δημιουργούμε ένα φάκελο στο workspace.
5. Αντιγραφή και Επικόλληση όλων των αρχείων από το φάκελο *FogComputingSim-master* στον καινούριο φάκελο στο workspace.
6. Ανοίγουμε την eclipse εφαρμογή και δημιουργούμε ένα Νέο Java Project δίνοντας το ίδιο όνομα με το καινούριο δημιουργημένο φάκελο.
7. Κάνω δεξί κλικ στο project και επιλέγω configure Build Path (Εικόνα 0.1), στο παράθυρο Libraries κάνω απεγκατάσταση των βιβλιοθηκών που δε λειτουργούν και προσθέτω την βιβλιοθήκη crplex.jar



Εικόνα 0.1: Παράθυρο Build path

8. Μόλις εγκατασταθεί η βιβλιοθήκη η δομή του καταλόγου του *FogComputingSim-master* φαίνεται στο Eclipse IDE στο *Project Name->src*. Υπάρχουν πολλαπλά πακέτα με τον κώδικα Java για κάθε εκτέλεση του FC, IoT και Edge Computing.

