

Ατομική Διπλωματική Εργασία

**Η ΕΠΙΔΡΑΣΗ ΤΟΥ ΚΟΙΝΩΝΙΚΟΥ ΔΙΚΤΥΟΥ TWITTER ΣΤΗ
ΒΕΛΤΙΩΣΗ ΤΗΣ ΕΠΙΔΟΣΗΣ ΤΩΝ ΔΙΚΤΥΩΝ ΠΑΡΑΔΟΣΗΣ
ΠΕΡΙΕΧΟΜΕΝΟΥ**

Δήμου Παναγιώτα

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2013

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Η Επίδραση του Κοινωνικού Δικτύου Twitter στη Βελτίωση της Επίδοσης των Δικτύων Παράδοσης Περιεχομένου

Δήμου Παναγιώτα

Επιβλέποντες Καθηγητές

Γεωργίου Χρύσης

Πάλλης Γιώργος

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

Μάιος 2013

Ευχαριστίες

Ευχαριστώ τους επιβλέποντες καθηγητές μου Δρ. Χρύση Γεωργίου και Δρ. Γιώργο Πάλλη για τη συνεχή καθοδήγηση και βοήθεια κατά τη διάρκεια της εκπόνησης της διπλωματικής μου εργασίας. Επίσης τους ευχαριστώ για την εμπιστοσύνη που έδειξαν προς το πρόσωπό μου, δίνοντας μου την ευκαιρία να εργαστώ σε αυτή τη διπλωματική εργασία.

Ευχαριστώ επίσης τους κύριο Κωνσταντίνο Στάμο, κύριο Ανδρέα Παπαδόπουλο και τον Γεώργιο Χριστοδούλου που μου πρόσφεραν τόσο τις γνώσεις τους όσο και πολύτιμη βοήθειά τους ανιδιοτελώς και απλόχερα.

Τέλος, θα ήθελα να πω ένα μεγάλο ευχαριστώ στους φίλους μου και ιδιαίτερα στον Αλέξανδρο, όπως επίσης και στην οικογένειά μου και ιδιαίτερα στους γονείς μου για την στήριξη, την αγάπη και την καθοδήγησή τους καθ' όλη τη διάρκεια των σπουδών μου.

Περίληψη

Το Διαδίκτυο είναι πλέον μέρος της καθημερινότητας ενός μεγάλου ποσοστού ανθρώπων παγκόσμια. Οι χρήστες του Διαδικτύου καθημερινά, δημιουργούν, διαμοιράζονται, αποστέλλουν και παραλαμβάνουν μεγάλη μάζα δεδομένων σε γοργούς ρυθμούς και έτσι απαιτούν την μέγιστη απόδοση του διαδικτύου και των υπηρεσιών που αυτό προσφέρει.

Η εξέλιξη λοιπόν της κοινωνίας και του διαδικτύου, φέρνει στην επιφάνεια την ανάγκη για περαιτέρω έρευνα με σκοπό τη βελτίωσή του. Σε αυτό λοιπόν στοχεύει αυτή η διπλωματική εργασία, δηλαδή στη βελτίωση των μηχανισμών που χρησιμοποιούνται σήμερα για την διανομή του περιεχομένου που αιτούνται οι χρήστες, δηλαδή τα δίκτυα παράδοσης περιεχομένου (Content Delivery Network - CDN).

Για τη βελτίωση της απόδοσης των δικτύων παράδοσης περιεχομένου, θα προσπαθήσουμε να αξιοποιήσουμε πληροφορίες που μας προσφέρουν τα κοινωνικά δίκτυα, μια υπηρεσία ευρέως διαδεδομένη σήμερα. Πιο συγκεκριμένα, το πώς οι χρήστες διαμοιράζονται και αξιοποιούν το περιεχόμενο στα κοινωνικά δίκτυα και οι σχέσεις που έχουν μεταξύ τους, μπορούν να αποτελέσουν μία εξαιρετική πηγή γνώσης. Τα πιο πάνω μπορούν επίσης να αποτελέσουν βοήθημα για την επιτυχή πρόβλεψη του τι θα γίνει δημοφιλές, πότε και πού έτσι ώστε το περιεχόμενο αυτό να τοποθετείτε έγκαιρα στους κοντινούς στους χρήστες εξυπηρετητές. Στόχος είναι να ελαχιστοποιηθούν η συμφόρηση, η καθυστέρηση και το μεγάλο κόστος στα δίκτυα παράδοσης περιεχομένου.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
	1.1 Κίνητρα	1
	1.2 Στόχοι	2
	1.3 Μεθοδολογία	3
	1.4 Οργάνωση Διπλωματικής Εργασίας	5
Κεφάλαιο 2	Υπόβαθρο και Σημαντικές Ορολογίες.....	7
	2.1 Δίκτυα Παράδοσης Περιεχομένου	7
	2.2 Κοινωνικά δίκτυα και Κοινωνικές διαδόσεις	9
	2.3 Σχετική Βιβλιογραφία	11
	2.4 Γεννήτρια Αιτήσεων	14
	2.5 Προσομοιωτής Δικτύων Παράδοσης Περιεχομένου CDNsim	16
Κεφάλαιο 3	Συλλογή και Επεξεργασία Δεδομένων.....	20
	3.1 Τοπολογία Διαδικτύου	20
	3.2 Τοπολογία Εξυπηρετητών και Χρηστών	20
	3.3 Χρήστες, Βίντεο και Αιτήσεις Χρηστών	24
Κεφάλαιο 4	Διατύπωση Προβλήματος.....	28
	4.1 Ορισμός Προβλήματος	28
	4.2 Περιγραφή Παραμέτρων	28
	4.3 Σκοπός	29
Κεφάλαιο 5	Πολιτικές Τοποθέτησης και Προ-ανάκλησης Περιεχομένου.....	31
	5.1 Στατικές Πολιτικές	31
	5.2 Δυναμική Πολιτική	35
Κεφάλαιο 6	Αξιολόγηση Στατικών Πολιτικών Τοποθέτησης Περιεχομένου	53
	6.1 Πειραματική Αξιολόγηση	53
	6.2 Αποτελέσματα	62
	6.3 Συμπεράσματα	70
Κεφάλαιο 7	Επίλογος	75
	7.1 Γενικά Συμπεράσματα	75
	7.2 Μελλοντική Εργασία	77

Βιβλιογραφία	79
---------------------------	-----------

Παράρτημα Α	A-1
--------------------------	------------

Παράρτημα Β	B-1
--------------------------	------------

Παράρτημα Γ	Γ-1
--------------------------	------------

Λίστα Σχημάτων

Σχήμα 2.1	Αναπαράσταση ενός Δικτύου Παράδοσης Περιεχομένου	8
Σχήμα 2.2	Παράδειγμα Κοινωνικής Διάδοσης – Εξάπλωση Αντικειμένου	11
Σχήμα 2.3	Κατανομή αιτήσεων στις ώρες της ημέρας [1]	15
Σχήμα 2.4	Κατανομή αιτήσεων στις μέρες της εβδομάδας [1]	16
Σχήμα 3.1	Διάγραμμα ροής για καθορισμό των εξυπηρετητών και συγκεκριμένη τοποθέτηση τους στην τοπολογία	23
Σχήμα 5.1	Συντελεστής διάδοσης σε σχέση με το μέγεθος των ζωνών ώρας σε λογαριθμική κλίμακα [1]	35
Σχήμα 5.2	Πιθανότητα ανά-δημοσίευσης βίντεο ανάλογα με το είδος της σχέσης μεταξύ των χρηστών [1]	36
Σχήμα 5.3	Ποσοστό κοινωνικής διάδοσης που έχει ολοκληρωθεί ανάλογα με την ώρα που έχει περάσει [1]	36
Σχήμα 5.4	Παράδειγμα για τη μετρική lobby-index	38
Σχήμα 5.5	Παράδειγμα για τη μετρική betweenness centrality [22]	40
Σχήμα 6.1	Παράδειγμα στατιστικής αναφοράς του προσομοιωτή δικτύου παράδοσης περιεχομένου CDNsim [2], η οποία δίδεται ως έξοδος μετά την προσομοίωση	54
Σχήμα 6.2	Στατιστική αναφορά του προσομοιωτή CDNsim σχετικά με την προσομοίωση της ακραίας πολιτικής τοποθέτησης όλου του περιεχομένου σε όλους τους υποκατάστατους εξυπηρετητές	59

Σχήμα 6.3	Στατιστική αναφορά του προσομοιωτή CDNsim σχετικά με την προσομοίωση της ακραίας πολιτικής κατά την οποία δεν γίνεται καμία τοποθέτηση περιεχομένου σε κανέναν από τους υποκατάστατους εξυπηρετητές	62
Σχήμα 6.4	Στατιστική αναφορά του προσομοιωτή CDNsim σχετικά με την προσομοίωση της πολιτικής τυχαίας τοποθέτησης αντικειμένων σε υποκατάστατους εξυπηρετητές, με πιθανότητα $p=10\%$	64
Σχήμα 6.5	Στατιστική αναφορά του προσομοιωτή CDNsim σχετικά με την προσομοίωση της πολιτικής τυχαίας τοποθέτησης αντικειμένων, με πιθανότητα $p=50\%$	65
Σχήμα 6.6	Γράφημα μέσου χρόνου απόκρισης σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές	67
Σχήμα 6.7	Γράφημα αριθμού μη-εξυπηρετούμενων αιτήσεων σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές	68
Σχήμα 6.8	Γράφημα μέσης χρησιμότητας υποκατάστατων εξυπηρετητών σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές	69
Σχήμα 6.9	Γράφημα του «hit ratio percentage» σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές	70

Λίστα Πινάκων

Πίνακας 3.1	Κατανομή υποκατάστατων εξυπηρετητών στη Γη για τους σκοπούς της πειραματικής αξιολόγησης	22
-------------	--	----

Κεφάλαιο 1

Εισαγωγή

1.1 Κίνητρα	1
1.2 Στόχοι	2
1.3 Μεθοδολογία	3
1.4 Οργάνωση Διπλωματικής Εργασίας	5

1.1 Κίνητρα

Με τη ραγδαία αύξηση των χρηστών του Διαδικτύου τα τελευταία χρόνια και τον τεράστιο όγκο δεδομένων και περιεχομένου που διακινείται καθημερινά στο Διαδίκτυο σε παγκόσμια κλίμακα, νέες μέθοδοι διαχείρισης των μηχανισμών του διαδικτύου είναι πλέον απαραίτητες.

Τα Δίκτυα Παράδοσης Περιεχομένου (Content Delivery Networks) [3] αποτελούν μηχανισμούς μεταφοράς και διαμοίρασης του περιεχομένου στο διαδίκτυο. Συγκεκριμένα, οι εξυπηρετητές προέλευσης (origin servers) κατανέμουν το περιεχόμενο σε υποκατάστατους εξυπηρετητές (surrogate servers) έτσι ώστε αυτό να φτάσει στους χρήστες. Λόγω των πιο πάνω εξελίξεων, τα Δίκτυα Παράδοσης Περιεχομένου δεν μπορούν πλέον να ελαχιστοποιήσουν πλήρως το κόστος και τον χρόνο απόκρισης του διαδικτύου με τον σημερινό τρόπο λειτουργίας τους. Αξίζει να αναφέρουμε ότι κάθε 1 λεπτό ανεβάζονται βίντεο διάρκειας 60 ωρών στο YouTube και καθημερινά πάνω από 4 δισεκατομμύρια βίντεο παρακολουθούνται από τους χρήστες του διαδικτύου [18]. Το κίνητρο λοιπόν για την συνεχή αναζήτηση τρόπων βελτίωσης των CDNs, έτσι ώστε η απόδοση του διαδικτύου να συμβαδίζει με την ταχεία ανάπτυξη του πληθυσμού του, είναι μεγάλο.

Σε αυτή τη έρευνα, σημαντικό ρόλο μπορούν να παίξουν τα διάφορα κοινωνικά δίκτυα (Social Networks), στα οποία συμμετέχει ένα μεγάλο μέρος του πληθυσμού της Γης.

Συγκεκριμένα στο Facebook, τον Οκτώβριο του 2012 υπήρχαν περισσότεροι από 1 δισεκατομμύριο ενεργοί χρήστες [19] ενώ στο Twitter υπάρχουν σήμερα περισσότεροι από 500 εκατομμύρια ενεργοί χρήστες [20].

Με αυτά τα στοιχεία βλέπουμε ότι τα κοινωνικά δίκτυα μπορούν να αποτελέσουν μια πολύτιμη πηγή πληροφόρησης για το πως διακινείται το περιεχόμενο στο διαδίκτυο. Η κίνηση του περιεχομένου στο διαδίκτυο και κατ' επέκταση η συμπεριφορά πλοήγησης των χρηστών επηρεάζονται από τα κοινωνικά δίκτυα και, έτσι υπάρχει η δυνατότητα πρόβλεψης του τί θα ζητήσουν οι χρήστες και πότε.

Χρησιμοποιώντας την πληροφορία και τα δεδομένα που τα κοινωνικά δίκτυα μας προσφέρουν, όπως τις προτιμήσεις των χρηστών, την γεωγραφική τοποθεσία τους, τις συνδέσεις μεταξύ φίλων και την κοινωνική τους δραστηριότητα, μπορούμε να προβλέψουμε ποιο περιεχόμενο θα γίνει δημοφιλές, πού και πότε. Μια τέτοια γνώση, θα μας επιτρέψει να προ-ανακαλέσουμε το περιεχόμενο στους κοντινούς υποκατάστατους εξυπηρετητές των συγκεκριμένων περιοχών, μειώνοντας έτσι τον χρόνο απόκρισης του διαδικτύου και αφήνοντας τους χρήστες ικανοποιημένους. Με αυτή τη μέθοδο, θα ελαχιστοποιηθεί επίσης το κόστος αντιγραφής περιεχομένου αφού πλέον θα γίνεται τοπικά και όχι καθολικά, επιτρέποντας περισσότερες και πιο εύστοχες αντιγραφές διαφορετικών αντικειμένων.

1.2 Στόχοι

Στόχος της ατομικής διπλωματικής εργασίας μου, είναι η δημιουργία πολιτικών προ-ανάκλησης (prefetching policies) οι οποίες θα έχουν ως σκοπό την βελτίωση της λειτουργίας των δικτύων παράδοσης περιεχομένου, χρησιμοποιώντας τη γνώση που μας προσφέρεται από τα κοινωνικά δίκτυα. Πιο συγκεκριμένα, αυτή η διπλωματική εργασία, αποσκοπεί στην κατασκευή μιας πολιτικής πρόβλεψης του περιεχομένου που πρόκειται να γίνει τοπικά δημοφιλές, σε ποια περιοχή και πότε με βάση την πλοήγηση των χρηστών στα κοινωνικά δίκτυα, έτσι ώστε το δίκτυο παράδοσης περιεχομένου να αντιγράψει έγκαιρα το περιεχόμενο στους κοντινούς, στην περιοχή αυτή, υποκατάστατους εξυπηρετητές (surrogate servers). Όπως αναφέρεται και πιο πάνω, με αυτήν την πολιτική ευελπιστούμε να βελτιώσουμε τον χρόνο απόκρισης (response time), τον αριθμό ικανοποιημένων αιτήσεων (success rate) και επίσης να

ελαχιστοποιήσουμε το κόστος αναπαραγωγής του περιεχομένου στους υποκατάστατους εξυπηρετητές.

Για αυτό τον σκοπό, αναλύονται, επεξηγούνται και χρησιμοποιούνται τα συμπεράσματα και τα δεδομένα που εξήλθαν από την Ατομική Διπλωματική Εργασία του Γιώργου Χριστοδούλου, απόφοιτου του τμήματος Πληροφορικής το 2012 [1]. Αναλύοντας τα δεδομένα του κοινωνικού δικτύου Twitter για περιεχόμενο του YouTube, εξήλθαν συγκεκριμένα συμπεράσματα για το πώς κινείται το περιεχόμενο στο διαδίκτυο μέσω των κοινωνικών δικτύων και άλλες σημαντικές πληροφορίες που μπορούμε να εκμεταλλευτούμε για να προβλέψουμε ποια αντικείμενα πρόκειται να γίνουν δημοφιλή. Επίσης χρησιμοποιείται και η Γεννήτρια Αιτήσεων η οποία μοντελοποιεί τις αιτήσεις στο YouTube χρησιμοποιώντας πραγματικές κατανομές και κοινωνικά δεδομένα. Αυτή η γεννήτρια είναι απαραίτητη για τη δοκιμή και τον έλεγχο της πολιτικής πρόβλεψης και προ-ανάκλησης περιεχομένου [1].

Επιπλέον, διαμορφώνεται το εργαλείο προσομοίωσης διαδικτύου, CDNsim [2], έτσι ώστε να ικανοποιεί πλήρως τις ανάγκες της πειραματικής αξιολόγησης των πολιτικών που κατασκευάσαμε και να υλοποιεί συγκεκριμένη τοποθέτηση των υποκατάστατων εξυπηρετητών στην τοπολογία του δικτύου.

Τέλος, κατασκευάζονται και αξιολογούνται πειραματικά, οι ακραίες πολιτικές τοποθέτησης περιεχομένου στους υποκατάστατους εξυπηρετητές, και προτείνεται σε θεωρητικό επίπεδο μία δυναμική πολιτική πρόβλεψης και προ-ανάκλησης περιεχομένου σύμφωνα με συμπεράσματα που εξήλθαν από το [1].

1.3 Μεθοδολογία

Αρχικά αναλύεται η σχετική βιβλιογραφία που αφορά τα δίκτυα παράδοσης περιεχομένου, τα κοινωνικά δίκτυα και προηγούμενες έρευνες που έχουν γίνει για το συγκεκριμένο σκοπό έτσι ώστε να εξαχθούν πληροφορίες, συμπεράσματα και στοιχεία που πρέπει να χαρακτηρίζουν την επιθυμητή πολιτική που πρόκειται να κατασκευάσουμε.

Προχωρώντας, γίνεται μια μελέτη και επεξεργασία των δεδομένων που έχουμε από προηγούμενες δουλειές, έτσι ώστε να μπορούν να χρησιμοποιηθούν στην κατασκευή των αλγορίθμων αλλά και να συμβαδίζουν με τις απαιτήσεις του προσομοιωτή ο οποίος θα αξιολογήσει τους αλγορίθμους αυτούς.

Ακολούθως γίνεται η διατύπωση του βασικού προβλήματος και του επιθυμητού αποτελέσματος.

Στη συνέχεια ανατέθηκε συγκεκριμένος αριθμός υποκατάστατων εξυπηρετητών στις 142 διαφορετικές ζώνες ώρας του κοινωνικού δικτύου Twitter – δηλαδή στις διάφορες περιοχές των χρηστών – ανάλογα με τον πληθυσμό που αντιστοιχεί στην κάθε περιοχή. Ακολούθως έγινε η αντιστοίχιση των εξυπηρετητών αυτών, σε συγκεκριμένους κόμβους της τοπολογίας του διαδικτύου έτσι ώστε η κατανομή των εξυπηρετητών στην τοπολογία να είναι πιο ρεαλιστική. Για να γίνει αυτό, υπολογιστήκαν οι συντεταγμένες των εξυπηρετητών όπως τοποθετήθηκαν σε κάθε περιοχή, και ακολούθως μετατράπηκαν σε συντεταγμένες του διαδικτύου.

Ακολούθως κατασκευάζονται δύο ακραίες πολιτικές:

1. Προ-ανάκληση όλου του περιεχομένου σε όλους του υποκατάστατους εξυπηρετητές.
2. Καμία προ-ανάκληση περιεχομένου σε κανέναν από τους υποκατάστατους εξυπηρετητές.

Επίσης κατασκευάζονται πολιτικές τυχαίας τοποθέτησης, κατά τις οποίες τοποθετούμε τα αντικείμενα στους υποκατάστατους εξυπηρετητές με μία πιθανότητα.

1. Πολιτική τυχαίας τοποθέτησης με πιθανότητα 10%.
2. Πολιτική τυχαίας τοποθέτησης με πιθανότητα 50%.

Οι ακραίες πολιτικές, μετά την πειραματική τους αξιολόγηση μας δίνουν τα όρια αποτελεσμάτων στα οποία θα πρέπει να κυμαίνεται η πολιτική προ-ανάκλησης περιεχομένου που θα κατασκευάσουμε. Οι πολιτικές τυχαίας τοποθέτησης και τα

αποτελέσματά τους θα μας δώσουν ένα άλλο μέτρο σύγκρισης το οποίο θα χρησιμοποιούμε για να δούμε κατά πόσον η πολιτική προ-ανάκλησης περιεχομένου όντως βελτιώνει την επίδοση του δικτύου.

Στη συνέχεια γίνεται μια εκτενής μελέτη της ανάλυσης των στοιχείων που έχουν συλλεχθεί από το [1] σχετικά με τα κοινωνικά δίκτυα και τη σχέση αυτών με την κίνηση του περιεχομένου μέσα στο διαδίκτυο. Επίσης μελετούνται οι κοινωνικές διαδόσεις και οι ιδιότητες τους οι οποίες θα αποτελούν στοιχεία για την κατασκευή της πολιτικής. Τα κύρια συμπεράσματα από την πιο πάνω μελέτη, θα χρησιμοποιηθούν στον καθορισμό της πολιτικής προ-ανάκλησης περιεχομένου στους υποκατάστατους εξυπηρετητές, έτσι ώστε να αξιολογήσουμε το κατά πόσον τα κοινωνικά δίκτυα μπορούν να δώσουν πληροφορία για την βελτίωση των δικτύων παράδοσης περιεχομένου.

Στο τέλος, παρουσιάζεται ο αλγόριθμος ο οποίος υλοποιεί τη δυναμική πολιτική πρόβλεψης και προ-ανάκλησης του περιεχομένου σύμφωνα με τα συμπεράσματα που έχουν εξαχθεί προηγουμένως. Επίσης προτείνονται μελλοντικές εργασίες που μπορούν να ακολουθήσουν την συγκεκριμένη διπλωματική εργασία.

1.4 Οργάνωση Διπλωματικής Εργασίας

Στο Κεφάλαιο 2 δίδεται το υπόβαθρο και επεξηγούνται σημαντικές έννοιες που αφορούν αυτή την διπλωματική εργασία. Επίσης περιγράφονται τα δύο εργαλεία που χρησιμοποιήθηκαν.

Στο Κεφάλαιο 3 γίνεται μία λεπτομερής ανάλυση για τη μεθοδολογία συλλογής και επεξεργασίας των δεδομένων που κρίθηκαν αναγκαία για την υλοποίηση των πολιτικών.

Στο Κεφάλαιο 4 διατυπώνεται το πρόβλημα και ορίζονται οι παράμετροι και τα κλειδιά που είναι σημαντικά για την κατασκευή της πολιτικής.

Στο Κεφάλαιο 5 παρουσιάζονται οι πολιτικές προ-ανάκλησης περιεχομένου στους υποκατάστατους εξυπηρετητές. Αρχικά επεξηγούνται οι στατικές πολιτικές (ακραίες

πολιτικές και πολιτικές τυχαίας τοποθέτησης) και ακολούθως παρουσιάζεται ο αλγόριθμος της δυναμικής πολιτικής. Η δυναμική πολιτική προσπαθεί με έναν «έξυπνο» τρόπο να τοποθετήσει τα αντικείμενα σε υποκατάστατους εξυπηρετητές κατά τη διάρκεια της έλευσης των αιτήσεων από τους χρήστες.

Στο Κεφάλαιο 6 παραθέτονται τα αποτελέσματα και τα συμπεράσματα μετά από την πειραματική αξιολόγηση των στατικών πολιτικών.

Τέλος, στο Κεφάλαιο 7 παρουσιάζονται τα γενικά συμπεράσματα και προτείνονται θέματα που θα μπορούσαν να αποτελέσουν μελλοντικές διπλωματικές εργασίες στο συγκεκριμένο πεδίο.

Κεφάλαιο 2

Υπόβαθρο και Σημαντικές Ορολογίες

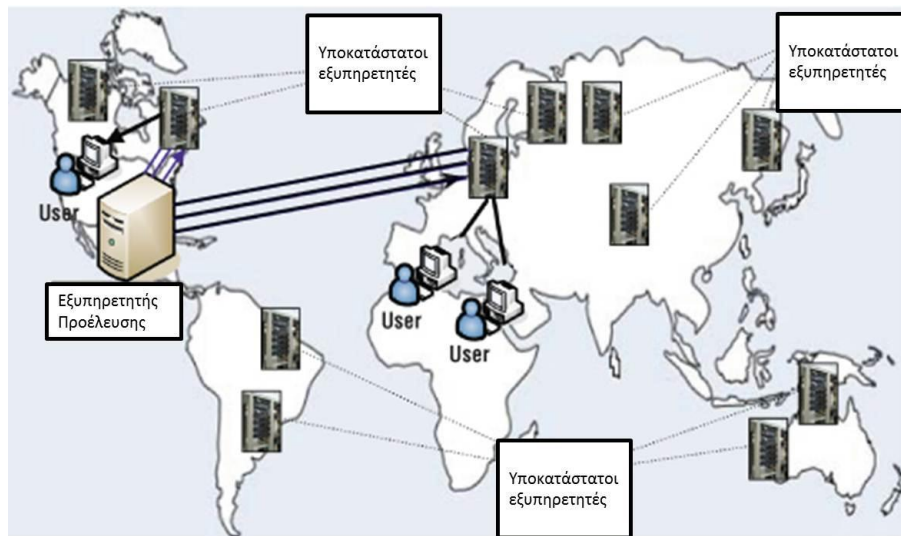
2.1 Δίκτυα Παράδοσης Περιεχομένου	7
2.2 Κοινωνικά δίκτυα και Κοινωνικές διαδόσεις	9
2.3 Σχετική Βιβλιογραφία	11
2.4 Γεννήτρια Αιτήσεων	14
2.5 Προσομοιωτής Δικτύων Παράδοσης Περιεχομένου CDNsim	16

2.1 Δίκτυα Παράδοσης Περιεχομένου

Τα Δίκτυα Παράδοσης Περιεχομένου (Content Delivery Networks) [3] παρουσιάζουν μεγάλο ερευνητικό ενδιαφέρον λόγω της υψηλής ποιότητας υπηρεσιών που προσφέρουν, μεταφέροντας το περιεχόμενο στους τελικούς χρήστες. Το περιεχόμενο διανέμεται σε υποκατάστατους εξυπηρετητές (surrogate servers) που βρίσκονται κοντά στους χρήστες που το έχουν ζητήσει, με γρήγορες και αξιόπιστες εφαρμογές και υπηρεσίες.

Ένα δίκτυο παράδοσης περιεχομένου, αποτελείται από τους εξυπηρετητές προέλευσης (origin servers) και τους υποκατάστατους εξυπηρετητές (surrogate servers) οι οποίοι είναι τοποθετημένοι κατανεμημένα σε διάφορα σημεία στη Γη και συνεργάζονται μεταξύ τους. Στο Σχήμα 2.1 δίνεται μία αναπαράσταση ενός Δικτύου Παράδοσης Περιεχομένου. Στους εξυπηρετητές προέλευσης υπάρχουν οι ιστοσελίδες και στους υποκατάστατους εξυπηρετητές μεταφέρονται αντίγραφα αυτών, από τους δρομολογητές του δικτύου, έτσι ώστε να είναι πιο κοντά στους χρήστες που τα έχουν αιτηθεί. Το αποτέλεσμα αυτής της τακτικής είναι η μείωση της συμφόρησης του διαδικτύου, αφού πλέον δεν επιβαρύνονται μόνο οι εξυπηρετητές προέλευσης από τον μεγάλο αριθμό αιτήσεων των χρηστών αλλά ο φόρτος κατανέμεται στους υποκατάστατους εξυπηρετητές που κατέχουν τα αντίγραφα. Επίσης, επιτυγχάνεται τόσο καλύτερη

εξυπηρέτηση των χρηστών όσο και ποιότητα, ταχύτητα και αξιοπιστία στην παράδοση του περιεχομένου [3].



Σχήμα 2.1: Αναπαράσταση ενός Δικτύου Παράδοσης Περιεχομένου[21]

Η επιτυχία των συγκεκριμένων δικτύων, έγκειται σε τρεις σημαντικούς παράγοντες [3]:

1. Η τοπολογία του δικτύου και η ορθή επιλογή της γεωγραφικής τοποθέτησης των υποκατάστατων εξυπηρετητών. Στόχος είναι η μέγιστη απόδοση και το μικρό κόστος υποδομής. Αυτό δεν σημαίνει κατ' ανάγκη, μεγάλος αριθμός εξυπηρετητών αλλά έξυπνη επιλογή τοποθέτησής τους έτσι ώστε να επιτυγχάνεται η εξυπηρέτηση των χρηστών ορθά, ανάλογα με την κατανομή των χρηστών στη Γη.
2. Η επιλογή του περιεχομένου που θα αντιγραφεί στους υποκατάστατους εξυπηρετητές. Η αντιγραφή περιεχομένου απαιτεί μνήμη, χρόνο και έχει υπολογιστικό κόστος. Έτσι πρέπει να γίνεται συγκεκριμένη και όχι τυχαία επιλογή του περιεχομένου που πρόκειται να αντιγραφεί. Επίσης θα ήταν καλύτερα να επιλέγεται για αντιγραφή το περιεχόμενο που ζητείται πιο συχνά από τους χρήστες. Αυτό μπορεί να επιτευχθεί, αναλύοντας την δραστηριότητα των χρηστών ή την τοπολογία των ιστοσελίδων οι οποίες στη δική μας εργασία αποτελούνται από τα κοινωνικά δίκτυα.
3. Η επιλογή της κατάλληλης πολιτικής για την αντιγραφή του περιεχομένου από τους εξυπηρετητές προέλευσης στους υποκατάστατους. Ανάλογα με τις

απαιτήσεις του δικτύου και τον τρόπο λειτουργίας του, είναι κρίσιμο να επιλεγεί η ορθή πολιτική αντιγραφής περιεχομένου – δηλαδή το πώς θα κινείται το περιεχόμενο στο δίκτυο – έτσι ώστε να γίνεται ορθή αξιοποίηση χώρου και χρόνου.

Υπάρχουν 3 πολιτικές για την αντιγραφή του περιεχομένου [3]

- a. Συνεργατική με ώθηση (cooperative push based) όπου το περιεχόμενο αντιγράφεται στον υποκατάστατο εξυπηρετητή πριν ζητηθεί και οι υπόλοιποι υποκατάστατοι δουλεύουν μαζί. Οι αιτήσεις των χρηστών εξυπηρετούνται από τον κοντινότερο εξυπηρετητή που έχει το περιεχόμενο, αλλιώς από τον εξυπηρετητή προέλευσης.
- b. Μη συνεργατική με έλξη (uncooperative pull-based) όπου το περιεχόμενο θα αντιγραφεί στον υποκατάστατο εξυπηρετητή από τον εξυπηρετητή προέλευσης, αφότου ζητηθεί από τους χρήστες. Δεν υπάρχει συνεργασία μεταξύ των υποκατάστατων εξυπηρετητών για ανταλλαγή περιεχομένου.
- c. Συνεργατική με έλξη (cooperative pull-based) όπου το περιεχόμενο θα αντιγραφεί στον υποκατάστατο εξυπηρετητή αφότου ζητηθεί από τους χρήστες. Σε αυτή την πολιτική υπάρχει συνεργασία μεταξύ των υποκατάστατων εξυπηρετητών σε περίπτωση όπου το επιθυμητό αντικείμενο για εξυπηρέτηση της αίτησης δεν υπάρχει στον υποκατάστατο εξυπηρετητή.

2.2 Κοινωνικά Δίκτυα και Κοινωνικές Διαδόσεις

Τα κοινωνικά δίκτυα είναι σελίδες, πλέον παγκόσμια διαδεδομένες, μέσα από τις οποίες οι χρήστες μπορούν να επικοινωνήσουν, να μοιράζονται πληροφορίες, γνώσεις και απόψεις. Αποσκοπούν στην ανταλλαγή, οργάνωση και εύρεση περιεχομένου από τους χρήστες αλλά και την έμμεση επαφή τους [4].

Σήμερα, τα δύο πιο δημοφιλή κοινωνικά δίκτυα είναι το Facebook και το Twitter τα οποία έχουν ως πρωταρχικό στόχο την επικοινωνία των χρηστών μεταξύ τους. Από αυτό όμως δεν εξυπακούεται η ομοιότητα των δύο κοινωνικών δικτύων ως προς τις λειτουργίες τους, οι οποίες στην πραγματικότητα διαφέρουν.

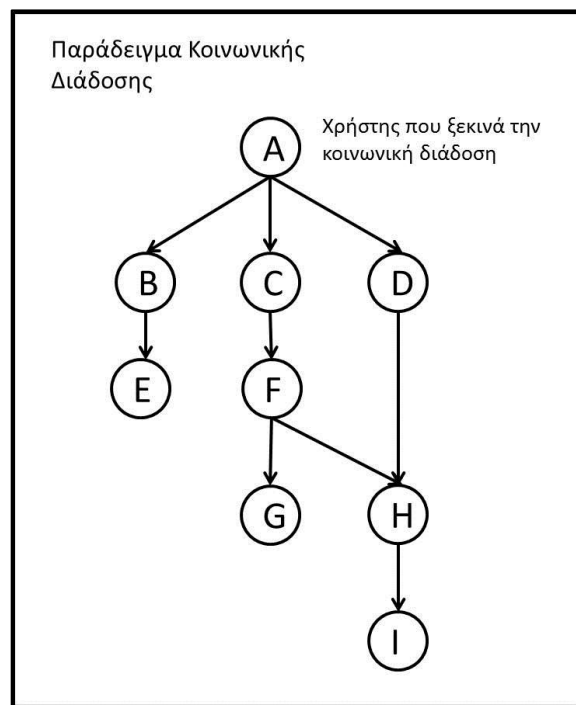
Ο κάθε χρήστης στο Facebook, έχει στην διάθεση του την κοινωνική ροή (social feed) στην οποία μπορεί να γράψει/δημοσιεύσει τις σκέψεις του και να διαβάσει δημοσιεύσεις φίλων του, να δει φωτογραφίες, βίντεο κτλ. Ο κάθε χρήστης μπορεί σε κάθε δημοσίευση φίλων του (αν αυτοί του επιτρέπουν) να αφήσει κάποιο σχόλιο, να δηλώσει ότι του αρέσει ή και να την αναδημοσιεύσει. Αντίθετα, στο Twitter οι χρήστες ίσως είναι κάπως πιο περιορισμένοι αφού μπορούν να δημοσιεύσουν κάτι μέσω ενός tweet (τιτίβισμα), το οποίο είναι ένα μήνυμα με μόνο 140 χαρακτήρες και μέσω αυτού μπορεί κάποιος να ενημερώσει τους ακολούθους του για το οτιδήποτε.

Επιπλέον γνωρίζουμε ότι στο Facebook υπάρχει η έννοια του «φίλου» και οι συνδέσεις μεταξύ αυτών στον κοινωνικό γράφο είναι αμφίδρομες (δηλαδή για να υπάρχει σύνδεσμος μεταξύ δύο χρηστών, πρέπει να το θέλουν και οι δύο). Αντίθετα, στο Twitter έχουμε την έννοια του «ακόλουθου». «Ακόλουθος» ονομάζεται κάποιος χρήστης στο Twitter ο οποίος δηλώνει ενδιαφέρον για κάποιον άλλο χρήστη του κοινωνικού δικτύου. Ως εκ τούτου, οι συνδέσεις στο Twitter είναι κατευθυνόμενες (δηλαδή δεν είναι αναγκαίο δύο χρήστες να έχουν αμφίδρομη σχέση για να υπάρχει σύνδεσμος μεταξύ τους – είναι αρκετό μόνο ο ένας από τους δύο να ακολουθεί τον δεύτερο).

Μία από τις πιο σημαντικές έννοιες στα κοινωνικά δίκτυα είναι αυτή της κοινωνικής διάδοσης (social cascade) [5]. Η κοινωνική διάδοση ενός περιεχομένου ξεκινά όταν κάποιος χρήστης ενός κοινωνικού δικτύου, μοιράζεται στην κοινωνική του ροή, κάποιο συμβάν ή κάποιο βίντεο ή κάτι που γενικά βρίσκει ενδιαφέρον (κόμβος Α στο Σχήμα 2.2), και ακολούθως κάποιος φίλος αυτού, το αναδημοσιεύει στη δική του κοινωνική ροή κοκ. Έτσι έχουμε την εξάπλωση αυτού του περιεχομένου, που μπορεί να παρομοιαστεί και με την εξάπλωση ενός ιού – επιδημία. Συγκεκριμένα, αυτό το φαινόμενο στα κοινωνικά δίκτυα, ονομάζεται κοινωνική διάδοση του αντικειμένου από τον πρώτο χρήστη στον δεύτερο. Οι κοινωνικές διαδόσεις και η μελέτη αυτών, μπορούν να δώσουν πληροφορίες για το ποια αντικείμενα θα γίνουν δημοφιλή και ποια αντικείμενα θα είναι αυτά τα οποία οι χρήστες θα τα αιτούνται πολύ συχνά.

Μπορούμε να αντιληφθούμε ότι πλέον τα κοινωνικά δίκτυα δεν αποτελούν μόνο μέσω σύνδεσης και ψυχαγωγίας των χρηστών αλλά και μια πολύτιμη πηγή πληροφοριών σχετικά με την κίνηση των δεδομένων στο διαδίκτυο λαμβάνοντας υπόψη τον μεγάλο

πληθυσμό ανθρώπων που συμμετέχουν σε αυτά. Οι διάφοροι μηχανισμοί παράδοσης περιεχομένου – όπως τα δίκτυα παράδοσης περιεχομένου – μπορούν να εκμεταλλευτούν τα αποτελέσματα και συμπεράσματα των ερευνών του συγκεκριμένου πεδίου έρευνας, με σκοπό την βελτίωση τους και την καλύτερη εξυπηρέτηση των χρηστών τους.



Σχήμα 2.2: Παράδειγμα Κοινωνικής Διάδοσης – Εξάπλωση Αντικειμένου

2.3 Σχετική Βιβλιογραφία

Με την πάροδο του χρόνου, έχουν γίνει προσπάθειες για την ένταξη της πληροφορίας που προσφέρουν τα κοινωνικά δίκτυα στον τρόπο που διαμοιράζεται το περιεχόμενο και τελικά φτάνει στον χρήστη. Πολλές από αυτές τις δουλειές χρησιμοποιούν άμεσα την πληροφορία από τα κοινωνικά δίκτυα, ενώ άλλες έμμεσα. Ο στόχος σε αυτό το ερευνητικό πεδίο είναι διάφοροι – η απόφαση για την αντιγραφή του περιεχομένου, η βελτιστοποίηση της πολιτικής προσωρινής αποθήκευσης κ.ά. Πιο κάτω περιγράφονται περιληπτικά τα πιο ενδεικτικά συστήματα.

2.3.1 Το μοντέλο Buzztraq

Αυτό το μοντέλο [6] προτείνει να χρησιμοποιείται η κοινωνική διάδοση για να αποφασίζεται το που θα τοποθετηθούν τα αντίγραφα. Πιο συγκεκριμένα, προτείνουν να χρησιμοποιείται η κοινωνική επιδημία (δηλαδή η εξάπλωση του αντικειμένου σε πολλούς χρήστες), και εκεί που είναι πυκνότερη σύμφωνα με τις προσβάσεις των φίλων του χρήστη που ζήτησε το αντικείμενο, να αντιγράφεται το αντικείμενο.

Πιο συγκεκριμένα, για κάθε αντικείμενο υπάρχει ένα βάρος σε κάθε περιοχή φίλων του χρήστη, και τελικά το αντικείμενο θα αντιγραφεί στις K περιοχές στις οποίες το αντικείμενο έχει το μεγαλύτερο βάρος. Έτσι χρησιμοποιείται όχι μόνο η γεωγραφική θέση του χρήστη που ζήτησε το αντικείμενο, αλλά και των χρηστών που το έχουν ζητήσει προηγουμένως. Στόχος αυτού του συστήματος είναι η ελαχιστοποίηση των απομακρυσμένων προσβάσεων.

Σε αυτό το σύστημα όμως, η έννοια της κοινωνικής διάδοσης γίνεται έμμεσα. Αυτό συμβαίνει γιατί στην πραγματικότητα δεν αναλύεται η κοινωνική διάδοση μέσω του κοινωνικού δικτύου απευθείας, αλλά απαιτείται μια τρίτη κεντρική σελίδα όπου οι χρήστες θα συνδέονται για να δουν βίντεο αλλά και να έχουν πρόσβαση στο κοινωνικό τους προφίλ.

2.3.2 Τοποθέτηση περιεχομένου με βάση την απόσταση μεταξύ των χρηστών και βελτίωση της πολιτικής προσωρινής αποθήκευσης

Το σύστημα του [7] αποφασίζει το πού θα αντιγραφεί το περιεχόμενο με βάση την απόσταση των χρηστών μεταξύ τους και χρησιμοποιώντας πληροφορίες και από τις κοινωνικές διαδόσεις. Επίσης στοχεύει στην βελτίωση της προσωρινής αποθήκευσης, δοκιμάζοντας το σύστημα πειραματικά με τρεις διαφορετικές προσεγγίσεις.

Αρχικά, για κάθε χρήστη-κόμβο, υπολογίζεται η τοπικότητα του (locality), που είναι μία μετρική η οποία ορίζει πόσο γεωγραφικά κοντά είναι ο συγκεκριμένος χρήστης από τους φίλους του, στο γράφο του κοινωνικού δικτύου. Ακολούθως, χρησιμοποιώντας μια δενδρική αναπαράσταση μιας κοινωνικής διάδοσης (ακμή από τον A στο B σημαίνει

κοινωνική διάδοση του περιεχομένου από τον A στον B – κάθε ακμή έχει βάρος την απόσταση μεταξύ των δύο χρηστών), υπολογίζονται οι μετρικές geo-diversity που είναι ο μέσος όρος των αποστάσεων μεταξύ όλων των χρηστών στο δέντρο και geo-range που είναι ο μέσος όρος των αποστάσεων μεταξύ όλων των χρηστών και του χρήστη που βρίσκεται στη ρίζα του δέντρου.

Αναλύοντας τις κοινωνικές διαδόσεις του κοινωνικού δικτύου Twitter και υπολογίζοντας τις πιο πάνω μετρικές, η ομάδα αυτή [7] έχει καταλήξει σε πολύ σπουδαία συμπεράσματα. Αρχικά βλέπουμε ότι η διάρκεια μιας κοινωνικής διάδοσης είναι πολύ μικρή, 30% των διαδόσεων τελειώνει σε 3 ώρες και το 80% των διαδόσεων τελειώνει σε 24 ώρες. Συμπέρασμα το οποίο έχει εξαχθεί και από το [1] και το οποίο εκμεταλλευόμαστε και εμείς για την κατασκευή της δυναμικής πολιτικής προ-ανάκλησης περιεχομένου. Επίσης δίδονται πληροφορίες σχετικά με τις καθυστερήσεις, δηλαδή στο πότε θα διαδώσουν οι χρήστες το αντικείμενο από την στιγμή που το έχουν δει στην κοινωνική τους ροή. Συγκεκριμένα 40% των διαδόσεων έχει καθυστέρηση 15 λεπτά ενώ 10% έχουν καθυστέρηση 2 λεπτά. Και τέλος, όσο μεγαλύτερη τοπικότητα έχει κάποιος χρήστης, τόσο πιο μικρή θα είναι η περιοχή εξάπλωσης μιας κοινωνικής ροής από τον συγκεκριμένο χρήστη.

Για να καθορίσουν λοιπόν ποιο βίντεο θα παραμείνει στον υποκατάστατο εξυπηρετητή, δίνουν σε αυτό μια τιμή με βάση την τοπικότητα των κόμβων που συμμετείχαν στην κοινωνική του διάδοση (Geo-cascade) ή με βάση την τοπικότητα όλων των κόμβων που μοιράστηκαν το βίντεο (Geo-social). Έτσι το περιεχόμενο με υψηλή τοπικότητα, θα πάρει ψηλή τιμή και θα παραμείνει στον υποκατάστατο εξυπηρετητή.

Συνδυάζοντας τις πιο πάνω μετρικές με 3 προσεγγίσεις προσωρινής αποθήκευσης (λιγότερο πρόσφατα χρησιμοποιημένο, πιο λίγο συχνά χρησιμοποιημένο και υβριδική προσέγγιση συνδυασμού και των δύο), η ομάδα έχει καταλήξει ότι το σύστημα δουλεύει καλύτερα σε σχέση με την απλή προσέγγιση όπου δεν δίδεται κάποιο βάρος στα βίντεο.

Το σύστημα αυτό βελτιώνει τις τεχνικές προσωρινής αποθήκευσης αλλά δεν χρησιμοποιεί την πληροφορία των κοινωνικών ροών για την πρόβλεψη των δημοφιλών αντικειμένων και την προ-ανάκληση τους στους υποκατάστατους εξυπηρετητές.

2.3.3 Το σύστημα TailGate

Το σύστημα αυτό [8], έχει ως στόχο τον προγραμματισμό της μεταφοράς του περιεχομένου στις διάφορες περιοχές παγκόσμια, κατά τις ώρες που δεν υπάρχει συμφόρηση στο δίκτυο.

Χρησιμοποιώντας το γεγονός ότι οι χρήστες επισκέπτονται το διαδίκτυο σε μεγαλύτερο ποσοστό συγκεκριμένες ώρες κατά τη διάρκεια της μέρας και ότι αυτές οι ώρες διαφέρουν στις διάφορες περιοχές στον κόσμο, η μεταφορά του περιεχομένου εκτός των ωρών αιχμής, περιορίζουν το κόστος. Έτσι, λαμβάνοντας υπόψη τις τοποθεσίες των φίλων του χρήστη και τις ώρες που γίνονται διαβάσματα σε κάθε τοποθεσία, το σύστημα βρίσκει την συντομότερη ώρα που πιθανό να υπάρξει αίτηση από κάποιον χρήστη για το συγκεκριμένο αντικείμενο. Ακολουθώντας βρίσκουν το πότε θα υπάρχει η λιγότερη συμφόρηση στο δίκτυο για να προγραμματιστεί η μεταφορά του αντικειμένου στον υποκατάστατο εξυπηρετητή. Επίσης, αναλύοντας κοινωνικές διαδόσεις του κοινωνικού δικτύου Twitter, κατέληξαν στο συμπέρασμα ότι υπάρχει ανάγκη χειρισμού του long tail περιεχομένου – το περιεχόμενο δηλαδή το οποίο θα ζητηθεί από λίγους χρήστες.

Το σύστημα αυτό καταφέρνει να αποδίδει καλά, να μειώνει το κόστος και να μειώνει τις άστοχες αιτήσεις (cache misses).

2.4 Γεννήτρια Αιτήσεων

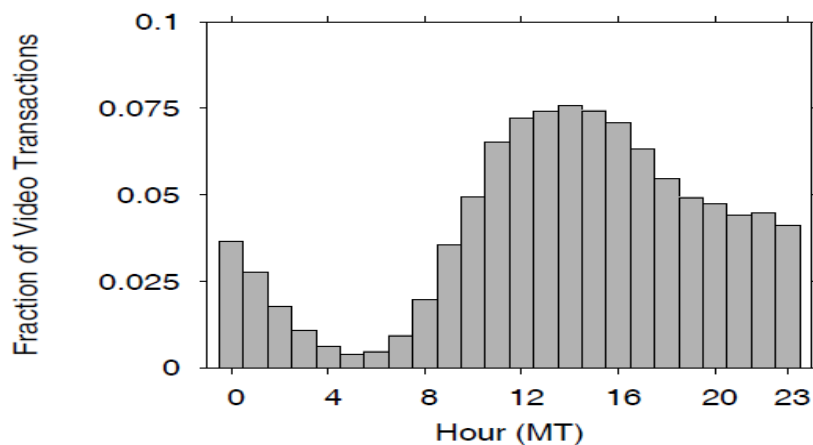
Στο [1] έγινε βαθιά μελέτη και έρευνα τόσο των πιο πάνω συστημάτων, όσο και της ροής του περιεχομένου στα κοινωνικά δίκτυα. Αποτέλεσμα αυτής της έρευνας ήταν η εξαγωγή σημαντικών συμπερασμάτων για το αν και πώς τα κοινωνικά δίκτυα επηρεάζουν την κίνηση στο διαδίκτυο. Στα πλαίσια του [1], έχει κατασκευαστεί και η Γεννήτρια Αιτήσεων – ένα πολύ σημαντικό εργαλείο – το οποίο αναλύεται πιο κάτω. Η παρούσα διπλωματική εργασία στηρίζεται στην έρευνα και τα ευρήματα της συγκεκριμένης δουλειάς, και επίσης αποτελεί την συνέχειά της.

Αρχικά λοιπόν, έγινε μια ιχνηλασία του κοινωνικού δικτύου Twitter για χρονικό διάστημα 5 μηνών που είχε στόχο την συλλογή πολύτιμων πληροφοριών τόσο για τους χρήστες του κοινωνικού δικτύου όσο και για τα tweets – μηνύματα που αυτοί

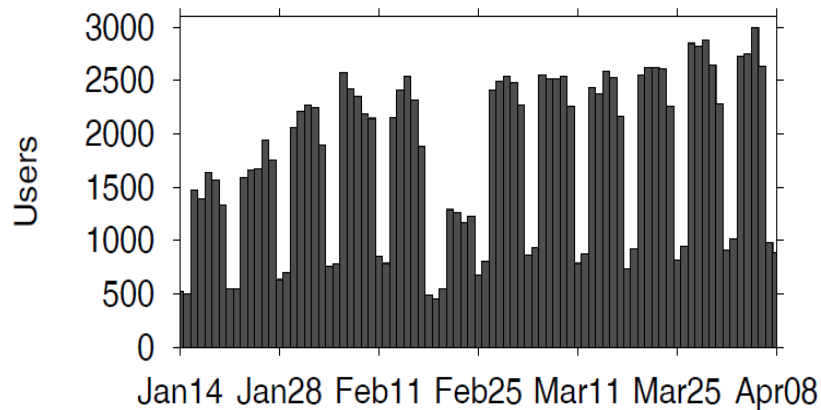
μοιραστήκαν με φίλους τους [1]. Ακολούθως έπρεπε να συλλεχθούν τα tweets που περιείχαν συνδέσμους σε βίντεο στο YouTube και από αυτά να εξαχθούν οι πραγματικοί σύνδεσμοι των βίντεο (αφού στα tweets υπήρχαν οι σε μορφή σμίκρυνσης). Τελικά συλλέχθηκαν και αναλύθηκαν δημοσιεύσεις για 1,384,758 χρήστες και πληροφορίες για 1,113,294 μοναδικά βίντεο που έχουν μοιραστεί οι χρήστες στο κοινωνικό δίκτυο Twitter.

Μετά την ανάλυση και διερμηνεία του μεγάλου όγκου δεδομένων που συλλέχθηκε, δημιουργήθηκε η Γεννήτρια Αιτήσεων η οποία αποσκοπεί στην δημιουργία αιτήσεων προς το διαδίκτυο οι οποίες ακολουθούν την κατανομή των πραγματικών αιτήσεων των χρηστών, κάτι που λείπει από τα προηγούμενα συστήματα που αναφέρθηκαν.

Κατά τη διάρκεια της κατασκευής της Γεννήτριας Αιτήσεων, εκτός των άλλων, λήφθηκαν υπόψη οι κατανομές των αιτήσεων στις ώρες της ημέρας και στις μέρες της εβδομάδας, όπως εμφανίζονται στο [9]. Εδώ φαίνεται ότι οι αιτήσεις αυξάνονται δραματικά γύρω στις 12 το μεσημέρι, παραμένουν ψηλά μέχρι τις βραδινές ώρες ενώ μειώνονται αισθητά μετά τα μεσάνυχτα (Σχήμα 2.3). Επίσης, υπάρχουν πολύ λιγότερες αιτήσεις κατά τη διάρκεια του Σαββατοκύριακου είτε κάποιας αργίας, σε σύγκριση με τις εργάσιμες μέρες τις εβδομάδας (Σχήμα 2.4).



Σχήμα 2.3: Κατανομή αιτήσεων στις ώρες της ημέρας [1]



Σχήμα 2.4: Κατανομή αιτήσεων στις μέρες της εβδομάδας [1]

Η Γεννήτρια Αιτήσεων [1] παίρνει σαν είσοδο τα εξής:

- Πραγματική ημερομηνία και ώρα έναρξης προσομοίωσης
- Χρονική διάρκεια προσομοίωσης
- Χρήστες και ζώνη ώρας του κάθε χρήστη
- Τα βίντεο με τον αριθμό θεάσεων και τη διάρκεια τους
- Κατανομή ανά ημέρα της εβδομάδας
- Κατανομή ανά ώρα της ημέρας
- Αθροιστική συνάρτηση κατανομής της διάρκειας συνόδων
- (Προαιρετικά) Εισερχόμενες διαδόσεις από το κοινωνικό διαδίκτυο με χρονοσφραγίδα για κάθε χρήστη
- (Προαιρετικά) Εξερχόμενες διαδόσεις από το κοινωνικό διαδίκτυο με χρονοσφραγίδα για κάθε χρήστη

Η έξοδος της Γεννήτριας Αιτήσεων είναι:

- Τα αιτήματα των χρηστών με τη μορφή:
 $\langle \text{Ταυτότητα Βίντεο} \rangle \langle \text{Ταυτότητα χρήστη} \rangle \langle \text{Χρονοσφραγίδα} \rangle$
 Όπου η χρονοσφραγίδα μετριέται σε δευτερόλεπτα από τη έναρξη της προσομοίωσης.

Η γεννήτρια Αιτήσεων δίνει ένα αρχείο εξόδου για κάθε μία από τις 142 ζώνες ώρας έτσι όπως ορίζονται στο Twitter.

Η έξοδος της Γεννήτριας Αιτήσεων, χρησιμοποιείται σε αυτή τη Διπλωματική Εργασία για την κατασκευή και πειραματική αξιολόγηση των πολιτικών προ-ανάκλησης περιεχομένου. Η Γεννήτρια Αιτήσεων είναι ένα πολύ σημαντικό εργαλείο αφού παρέχει την δυνατότητα εκτέλεσης πειραμάτων πάνω σε ρεαλιστικά δεδομένα και όχι πιθανοκρατικά. Η αξιολόγηση λοιπόν των πειραμάτων αυτών είναι αντικειμενική, χωρίς παραδοχές, αφού τα δεδομένα που χρησιμοποιούνται είναι ρεαλιστικά. Έτσι, μπορούμε πολύ πιο εύκολα να χαρακτηρίσουμε μια πιθανή πολιτική προ-ανάκλησης περιεχομένου ως επιτυχημένη ή αποτυχημένη.

2.5 Προσομοιωτής Δικτύων Παράδοσης Περιεχομένου CDNsim

Για την πειραματική αξιολόγηση τόσο των ακραίων και τυχαίων πολιτικών όσο και της πολιτικής προ-ανάκλησης περιεχομένου στου εξυπηρετητές, χρησιμοποιείται το CDNsim το οποίο είναι ένα εργαλείο όπου προσομοιώνει τη λειτουργία των Δικτύων Παράδοσης Περιεχομένου [2].

Δίνοντας στον προσομοιωτή κάποια συγκεκριμένα αρχεία τα οποία περιγράφουν το δίκτυο παράδοσης περιεχομένου που έχουμε στη διάθεση μας όπως και την κίνηση στο διαδίκτυο, μας παρέχει ως έξοδο ένα αρχείο με στατιστικές πληροφορίες σχετικά με τη λειτουργία και την απόδοση του δικτύου.

Η διαδικασία για την εξαγωγή της συγκεκριμένης στατιστικής αναφοράς αποτελείται από 6 βήματα. Τα πρώτα 4 βήματα αφορούν την εξαγωγή του «μπουκαλιού» (bottle), μια ορολογία που χρησιμοποιείται στον προσομοιωτή και αναφέρεται στην αναπαράσταση της συλλογής των δεδομένων περιγραφής του δικτύου, δηλαδή το σενάριο του δικτύου.

Τα βήματα είναι τα εξής:

- 1 Επιλογή πολιτικής με την οποία οι εξυπηρετητές θα διαχειρίζονται τις ιστοσελίδες. Για τους σκοπούς της παρούσας διπλωματικής εργασίας, χρησιμοποιείται η πολιτική «Συγκριτικό Περιβάλλον» - Comparative Environment. Η συγκεκριμένη πολιτική αναθέτει την αίτηση του χρήστη στον κοντινότερο σε αυτόν εξυπηρετητή μετρώντας τα hops της τοπολογίας του δικτύου. Επίσης, εάν ο κοντινότερος εξυπηρετητής δεν έχει το συγκεκριμένο

αντικείμενο στη μνήμη του, θα πάρει το αντικείμενο αυτό από τον κοντινότερο εξυπηρετητή που το διαθέτει.

- 2 Εισαγωγή του αρχείου που αναπαριστά την τοπολογία του δικτύου, δείχνοντας τους κόμβους αυτού και τις συνδέσεις – ακμές μεταξύ τους. Σε αυτό το βήμα καθορίζονται και κάποιες άλλες παράμετροι του συστήματος όπως ο αριθμός των χρηστών, υποκατάστατων εξυπηρετητών και εξυπηρετητών προέλευσης. Επίσης καθορίζεται και πόσες φορές ο εξυπηρετητής θα ξαναπροσπαθήσει να πάρει το αντικείμενο σε περίπτωση που η αίτηση αποτύχει (number of retries), όπως και το bandwidth με το οποίο τα δεδομένα θα κινούνται μέσα στην τοπολογία.
- 3 Εισαγωγή του αρχείου που περιέχει πληροφορίες των αντικειμένων τα οποία δύναται να διακινηθούν μέσα στο δίκτυο. Το συγκεκριμένο αρχείο είναι της μορφής <ταυτότητα αντικειμένου> <μέγεθος σε bytes>. Επίσης σε αυτό το βήμα εισάγουμε και το αρχείο το οποίο δείχνει την κίνηση του δικτύου – δηλαδή το σενάριο της προσομοίωσης για την οποία κάνουμε πειραματική αξιολόγηση. Ουσιαστικά σε αυτό το αρχείο βρίσκονται οι αιτήσεις των χρηστών (τις οποίες έχουμε στη διάθεσή μας από την Γεννήτρια Αιτήσεων [1]) σε μορφή <timestamp> <ταυτότητα χρήστη> <ταυτότητα αντικειμένου>.
- 4 Επιλέγουμε το αρχείο στο οποίο θέλουμε να τοποθετηθεί το bottle μετά την εκτέλεση του προσομοιωτή όπως και το όνομα του μπουκαλιού και αρχίζει η προσομοίωση.
- 5 Αφού τοποθετηθούν όλες οι πληροφορίες στο μπουκάλι, ο προσομοιωτής με κάποια βοηθητικά αρχεία, θα εκτελέσει το μπουκάλι, δηλαδή ουσιαστικά θα κινηθούν τα αντικείμενα στο δίκτυο παράδοσης περιεχομένου και θα εκτελεστούν ρεαλιστικά οι αιτήσεις των χρηστών. Όλες οι πληροφορίες από αυτό το βήμα θα αποθηκευτούν και πάλι στο μπουκάλι.
- 6 Τέλος, μπορούμε πολύ εύκολα, χρησιμοποιώντας μία άλλη λειτουργία του προσομοιωτή να πάρουμε μια αναφορά τύπου HTML η οποία μας παρέχει

στατιστικές πληροφορίες σχετικά με την προσομοίωση που μόλις έχει εκτελεσθεί.

Διαφοροποίηση του προσομοιωτή

Για τους σκοπούς της πειραματικής αξιολόγησης των πολιτικών που κατασκευάσαμε, χρειάστηκε να διαφοροποιηθεί μία λειτουργία του προσομοιωτή CDNsim. Πιο συγκεκριμένα, η λειτουργία του προσομοιωτή τοποθετεί τυχαία τους υποκατάστατους εξυπηρετητές και τους εξυπηρετητές προέλευσης στην τοπολογία που εμείς εισάγουμε. Επίσης τοποθετεί τυχαία και τους κόμβους με τους οποίους αναπαρίστανται οι χρήστες του δικτύου παράδοσης περιεχόμενου, οι οποίοι αιτούνται τα αντικείμενα.

Για τις δικές μας λειτουργίες, μια τυχαία τοποθέτηση στην τοπολογία του δικτύου ήταν ανεπιθύμητη αφού οι υποκατάστατοι εξυπηρετητές στην τοπολογία έπρεπε να τοποθετηθούν με τέτοιο τρόπο έτσι ώστε να αντιπροσωπεύουν την πραγματική τους θέση στη Γη. Επίσης οι χρήστες έπρεπε να τοποθετηθούν σε συγκεκριμένους κόμβους στην τοπολογία έτσι ώστε να απεικονίζουν την πραγματική περιοχή των ζωνών ώρας, όπως ορίζονται από το Twitter. Για τον σκοπό αυτό, διαφοροποιήθηκε ο προσομοιωτής CDNsim, έτσι ώστε να καθορίζει συγκεκριμένους κόμβους στην τοπολογία του δικτύου για κάθε εξυπηρετητή που ορίζεται (Παράρτημα Γ).

Το ποιοι είναι οι κόμβοι που αντιπροσωπεύουν τους πραγματικούς εξυπηρετητές και τις γεωγραφικές περιοχές των χρηστών, αποφασίσθηκε από εμάς (περισσότερες πληροφορίες στο κεφάλαιο 3) και δίνονται στον προσομοιωτή σε αρχεία με τη μορφή <ταυτότητα εξυπηρετητή> <ταυτότητα κόμβου στην τοπολογία>. Έτσι, σε συγκεκριμένο σημείο, ο προσομοιωτής ορίζει ότι ο x εξυπηρετητής θα αναπαρίσταται στην τοπολογία από τον y κόμβο.

Κεφάλαιο 3

Συλλογή και Επεξεργασία Δεδομένων

3.1 Τοπολογία Διαδικτύου	20
3.2 Τοπολογία Εξυπηρετητών και Χρηστών	20
3.3 Χρήστες, Βίντεο και Αιτήσεις Χρηστών	24

3.1 Τοπολογία Διαδικτύου

Για την πειραματική αξιολόγηση των πολιτικών που κατασκευάστηκαν, ήταν απαραίτητη μία τοπολογία δικτύου έτσι ώστε ο προσομοιωτής CDNSim να μπορέσει να εκτελέσει το σενάριο της κάθε πολιτικής στο δίκτυο παράδοσης περιεχομένου.

Η Γεννήτρια INET [10] δίνει τη δυνατότητα δημιουργίας μίας AS (autonomous system)-επιπέδου, αναπαράσταση της τοπολογίας του διαδικτύου. Δίνοντας σε αυτή τη γεννήτρια τον αριθμό των κόμβων που θέλουμε να υπάρχουν στην τοπολογία που θα χρησιμοποιηθεί για την προσομοίωση, μας παράγει ένα αρχείο το οποίο περιέχει την τοπολογία. Συγκεκριμένα, στο αρχείο αυτό εμφανίζονται οι ταυτότητες του κάθε κόμβου και οι συντεταγμένες του στην τοπολογία όπως και οι ακμές μεταξύ των κόμβων, δίνοντας έτσι μια αναπαράσταση της τοπολογίας στο διαδίκτυο. Στα σενάρια που θέλαμε να προσομοιώσουμε στο CDNSim, χρησιμοποιήσαμε τοπολογία διαδικτύου με 3500 κόμβους.

3.2 Τοπολογία Εξυπηρετητών και Χρηστών

Το επόμενο βήμα ήταν η τοποθέτηση του εξυπηρετητή προέλευσης, των υποκατάστατων εξυπηρετητών αλλά και των χρηστών σε συγκεκριμένους κόμβους στην τοπολογία. Η τοπολογία του διαδικτύου που μας έδωσε η Γεννήτρια INET, θέλουμε να αντανakλά ρεαλιστική κατανομή των εξυπηρετητών αλλά και χρηστών όπως είναι και στην πραγματικότητα.

Καθορισμός τοποθεσίας εξυπηρετητών στην τοπολογία του διαδικτύου

Όπως προαναφέρθηκε, η γεννήτρια τοπολογίας δίνει στην έξοδο τους κόμβους της τοπολογίας, τον κάθε ένα με μια ταυτότητα και τις συντεταγμένες του σε αυτή. Έπρεπε λοιπόν, να καθορίσουμε ποιοι κόμβοι θα αναπαριστούσαν τους υποκατάστατους εξυπηρετητές, ποιοι τους εξυπηρετητές προέλευσης και ποιοι τους χρήστες που θα έκαναν αιτήσεις. Η διαδικασία η οποία αναλύεται πιο κάτω, περιγράφεται και στο διάγραμμα ροής στο Σχήμα 3.1.

Αρχικά, έπρεπε να καθορίσουμε πόσοι εξυπηρετητές θα προσομοίωναν τις πολιτικές μας και πού θα ήταν τοποθετημένοι σε πραγματική γεωγραφική θέση. Για τον σκοπό αυτό, χρησιμοποιήσαμε τη γεωγραφική κατανομή των εξυπηρετητών του δικτύου Limelight [11] όσον αφορά τις γεωγραφικές περιοχές όπου θα τοποθετούνταν οι εξυπηρετητές. Συνολικά τοποθετήσαμε 423 υποκατάστατους εξυπηρετητές όπως φαίνονται στον Πίνακα 3.1.

Πόλη	#Υποκατάστατων Εξυπηρετητών	Πόλη	#Υποκατάστατων Εξυπηρετητών
Washington CD	55	Toronto	12
New York	43	Amsterdam	20
Atlanta	11	London	30
Miami	11	Frankfurt	31
Chicago	37	Paris	12
Dallas	19	Moscow	10
Los Angeles	52	Hong Kong	8
San Jose	37	Tokyo	12
Seattle	15	Changi	5
Phoenix	3	Sydney	1

Πίνακας 3.1: Κατανομή υποκατάστατων εξυπηρετητών στη Γη για τους σκοπούς της πειραματικής αξιολόγησης

Ο καθορισμός του αριθμού των υποκατάστατων εξυπηρετητών που μπήκε σε κάθε περιοχή, εξήλθε με βάση την κατανομή των εξυπηρετητών του δικτύου Limelight [11], με μειωμένους αριθμούς στο 10% έτσι ώστε να υπάρχει ισορροπία με το συνολικό αριθμό κόμβων στην δική μας τοπολογία διαδικτύου.

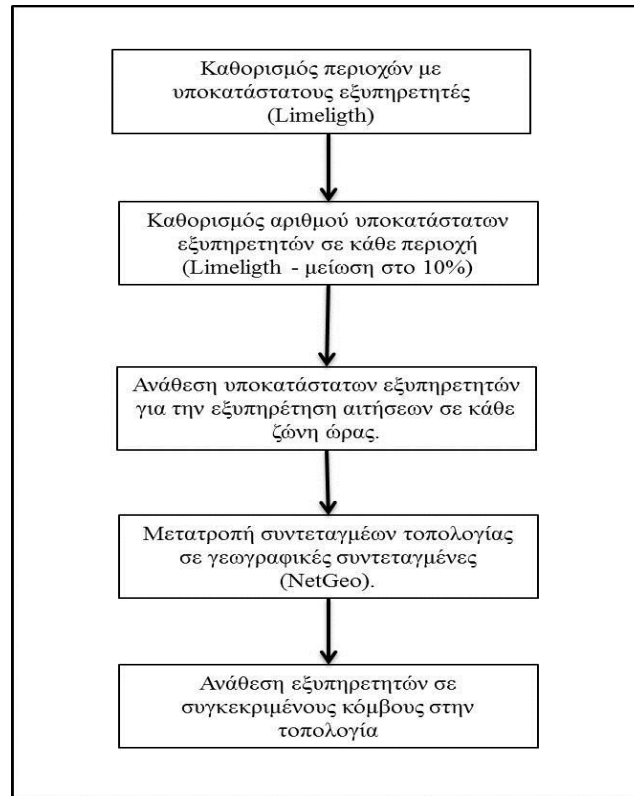
Ακολούθως, σύμφωνα με αυτή την τοποθέτηση, καθορίσαμε συγκεκριμένους υποκατάστατους εξυπηρετητές οι οποίοι θα λαμβάνουν αρχικά τις αιτήσεις των χρηστών από την κάθε ζώνη ώρας. Υπολογίζεται η απόσταση μεταξύ των συντεταγμένων των 142 ζωνών ώρας του Twitter και των 20 περιοχών στις οποίες σύμφωνα με το Limelight δίκτυο τοποθετούνται οι υποκατάστατοι εξυπηρετητές. Ανάλογα με το ποια περιοχή εξυπηρετητών βρίσκεται πιο κοντά σε κάθε ζώνη ώρας, αποφασίστηκε το που θα ανακατευθύνονται οι αιτήσεις από την συγκεκριμένη ζώνη ώρας. Για την απόφαση αυτή, λήφθηκε υπόψη και ο πληθυσμός σε κάθε ζώνη ώρας έτσι ώστε να μην υπάρχει μεγάλος φόρτος αιτήσεων σε κάποιες περιοχές υποκατάστατων εξυπηρετητών ενώ σε κάποιες άλλες όχι.

Το επόμενο βήμα όριζε την συγκεκριμένη τοποθέτηση των εξυπηρετητών, στην τοπολογία του διαδικτύου. Το πρόβλημα εδώ έγκειται στο ότι οι γεωγραφικές συντεταγμένες δε συνάδουν με τις συντεταγμένες της τοπολογίας του διαδικτύου, έτσι δεν μπορούσαμε άμεσα να βρούμε τις τοποθεσίες του κάθε εξυπηρετητή στην τοπολογία.

Η CAIDA παρέχει ένα εργαλείο, το NetGeo το οποίο καταφέρνει να χαρτογραφεί IP διευθύνσεις και Autonomous System – AS συντεταγμένες, σε γεωγραφικές συντεταγμένες [12, 13]. Το εργαλείο αυτό, για να βρει το γεωγραφικό πλάτος και μήκος μίας IP διεύθυνσης ή ενός AS, αρχικά ψάχνει για αυτό το στοιχείο σε ιδιόκτητη βάση δεδομένων που κατέχει και αν δεν το βρει εκεί, θα ψάξει σε ARIN/APNIC/RIPE εξυπηρετητές έτσι ώστε να βρει την πληροφορία που ψάχνει [12, 13]. Η έξοδος αυτού του εργαλείου είναι οι γεωγραφικές συντεταγμένες των στοιχείων που ζητήθηκαν.

Μετά την εξαγωγή των γεωγραφικών θέσεων των κόμβων της τοπολογίας του διαδικτύου, μπορούσαμε εύκολα να συσχετίσουμε τους εξυπηρετητές στην τοπολογία, υπολογίζοντας την απόσταση τους. Για παράδειγμα, για να τοποθετήσουμε τους 55

υποκατάστατους εξυπηρετητές που ορίσαμε πως υπάρχουν στην πόλη Washington DC, βρήκαμε τους 55 πιο κοντινούς κόμβους στην τοπολογία συγκρίνοντας τις γεωγραφικές συντεταγμένες τους με αυτές της πόλης Washington DC. Έτσι καταφέραμε να αναπαραστήσουμε τους εξυπηρετητές στην τοπολογία, με ρεαλιστική κατανομή.



Σχήμα 3.1: Διάγραμμα ροής για καθορισμό των εξυπηρετητών και συγκεκριμένη τοποθέτησή τους στην τοπολογία.

Καθορισμός τοποθεσίας εξυπηρετητών στην τοπολογία του διαδικτύου

Έπρεπε επίσης να καθοριστεί και το ποιοι κόμβοι θα αναπαριστούν τους χρήστες που θα κάνουν αιτήσεις προς το διαδίκτυο. Αρχικά παρατηρήσαμε ότι θα ήταν ακατόρθωτο να κάνουμε αυτόν τον καθορισμό για τον κάθε χρήστη που έτυχε ιχνηλασίας από το [1]. Επίσης αυτό δεν θα ήταν κάτι σταθερό, αφού κάθε φορά που χρησιμοποιούμε την Γεννήτρια Αιτήσεων, μπορεί να δημιουργηθούν αιτήσεις από διαφορετικούς χρήστες. Επιπρόσθετα, το σύνολο των χρηστών είναι πολύ μεγάλο και η απόφαση για την τοποθεσία του κάθε χρήστη ξεχωριστά απαιτεί πολύ χρόνο.

Για να πετύχουμε λοιπόν μία σταθερή τοποθέτηση των χρηστών στην τοπολογία, έπρεπε να συμπίεσουμε τους χρήστες στην κάθε ζώνη ώρας, έτσι ώστε να φαίνονται σαν ένας χρήστης. Οπότε, όλοι οι χρήστες από την ζώνη ώρας του Λονδίνου για παράδειγμα, θα έχουν μια νέα μοναδική ταυτότητα και έτσι θα ξεχωρίζουν από τους υπόλοιπους. Η τοποθέτηση λοιπόν στην τοπολογία θα γίνει μία φορά και θα είναι σταθερή.

Αφού λοιπόν έγινε η συμπίεση των χρηστών ανά ζώνη ώρας, έπρεπε η κάθε ομάδα χρηστών να τοποθετηθεί σε κάποιο κόμβο της τοπολογίας. Ο συγκεκριμένος κόμβος έπρεπε να είναι κοντά σε κάποιον από τους υποκατάστατους εξυπηρετητές οι οποίοι, όπως έχει οριστεί προηγουμένως, ανήκουν στην περιοχή εξυπηρετητών η οποία θα εξυπηρετεί τις αιτήσεις των χρηστών της συγκεκριμένης ζώνης ώρας. «Κοντινός κόμβος» καθορίζεται αυτός που απέχει τα λιγότερα hops στην τοπολογία του διαδικτύου.

Η αναγκαιότητα της συγκεκριμένης διαδικασίας οφείλεται στο ότι για την προσομοίωση του δικτύου παράδοσης περιεχομένου και την εκτέλεση των πολιτικών, χρησιμοποιούμε την πολιτική «Συγκριτικό Περιβάλλον» (Comparative Environment) για την διαχείριση των ιστοσελίδων από τους εξυπηρετητές. Όπως προαναφέρθηκε, η συγκεκριμένη πολιτική ανακατευθύνει την αίτηση του χρήστη στον κοντινότερο σε αυτόν εξυπηρετητή μετρώντας τα hops της τοπολογίας του διαδικτύου. Έτσι έπρεπε οι χρήστες να τοποθετηθούν στους πιο κοντινούς κόμβους από αυτούς που αποτελούν τους εξυπηρετητές τους οποίους θέλουμε να αναλαμβάνουν τα αιτήματα των ζωνών ώρας και με αυτόν τον τρόπο να πετύχουμε ρεαλιστική κίνηση στο διαδίκτυο.

Αν όμως οι χρήστες τοποθετούνταν τυχαία στην τοπολογία θα ήταν πολύ πιθανόν, κάποιος χρήστης που βρίσκεται στην ζώνη ώρας του Λονδίνου να εξυπηρετείται από υποκατάστατο εξυπηρετητή του Σύδνεϋ, κάτι που είναι ανεπιθύμητο.

3.3 Χρήστες, Βίντεο και Αιτήσεις Χρηστών

Αρχικά, χρησιμοποιούνται όλα τα δεδομένα τα οποία έχουν εξαχθεί στο [1] και αφορούσαν την ιχνηλασία που πραγματοποιήθηκε στο κοινωνικό δίκτυο Twitter. Η

ιχνηλασία αυτή έγινε με βάση συγκεκριμένους χρήστες οι οποίοι ήταν γνωστό ότι υπήρχαν στο κοινωνικό δίκτυο Twitter και αφορούσε την χρονική περίοδο του 2009.

Είχαμε λοιπόν στη διάθεση μας σημαντικές πληροφορίες τόσο για τους χρήστες του κοινωνικού δικτύου, όσο και για τα βίντεο του YouTube τα οποία έλαβαν μέρος σε κοινωνική διάδοση. Επίσης, έχουμε τον κοινωνικό γράφο του δικτύου όπου αποτελείται από 37,343,273 χρήστες και 6,820,494,777 ακμές [1].

Χρήστες

Οι πληροφορίες που είχαν συλλεχθεί για κάθε χρήστη και μας ενδιέφεραν στην κατασκευή και προσομοίωση των πολιτικών είναι οι εξής [1]:

Id	Ταυτότητα του χρήστη
followers_count	Αριθμός ακολούθων του χρήστη
friends_count	Αριθμός χρηστών που ακολουθεί ο χρήστης
Location	Δηλωμένη τοποθεσία του χρήστη
Time_zone	Ζώνη ώρας στην οποία βρίσκεται ο χρήστης

Βίντεο

Οι πληροφορίες που αφορούν τα βίντεο είναι οι εξής [1]:

Id	Ταυτότητα του βίντεο
Duration	Διάρκεια του βίντεο σε δευτερόλεπτα
Views	Αριθμός θεάσεων του βίντεο

Εύρεση μεγέθους των βίντεο

Μέσα από την ιχνηλασία στο [1], έχουν εξαχθεί συγκεκριμένα βίντεο στο YouTube στα οποία ακολούθως έχει ανατεθεί μοναδικός αριθμός – ταυτότητα και με βάση αυτή μπορούμε να βρούμε διάφορες πληροφορίες για το κάθε βίντεο, όπως είναι η διάρκειά του, πόσες θεάσεις έχει κ.ά. Κάτι όμως που χρειάζεται στην προσομοίωση των πολιτικών και δεν περιέχεται στις πληροφορίες που έχουν συλλεχθεί, είναι το μέγεθος του κάθε βίντεο. Στόχος λοιπόν είναι με βάση την διάρκεια του βίντεο, να υπολογισθεί το μέγεθος του σε bytes.

Μελετώντας διάφορες τακτικές και μεθόδους, καταλήξαμε στην επιλογή μίας μεθόδου η οποία φαίνεται να είναι καθολικά αποδεκτή και περιγράφεται στο [14]. Για να υπολογίσουμε το μέγεθος των βίντεο, χρειάστηκε να κάνουμε την παραδοχή ότι όλα αυτά τα βίντεο έχουν ανάλυση 480p που είναι και η πιο συνηθισμένη ανάλυση στο YouTube.

Τα βήματα για την εύρεση του μεγέθους ενός βίντεο χρησιμοποιώντας την διάρκεια του είναι τα εξής:

1. Μέγεθος πλαισίου (Frame size)

$$= (\text{πλάτος των pixel} * \text{ύψος των pixel} * \text{βάθος του bit}) / 8$$

$$= (640 * 480 * 24) / 8$$

Η διαίρεση με το 8 πραγματοποιείται για να πάρουμε την τιμή σε bytes.

2. Πλαίσια ανά δευτερόλεπτο (Frames per second) = 30

(σύμφωνα με την παραδοχή που έχουμε κάνει)

3. Μη συμπιεσμένο μέγεθος του βίντεο (Uncompressed Video Size) (σε bytes)

$$= \text{Μέγεθος πλαισίου} * \text{Πλαίσια ανά δευτερόλεπτο} * \text{Διάρκεια του βίντεο (duration)}$$

Η διάρκεια του βίντεο σε δευτερόλεπτα

4. Συμπιεσμένο μέγεθος του βίντεο (Compressed Video Size)

$$= \text{Μη συμπιεσμένο μέγεθος του βίντεο} / \text{Αναλογία συμπίεσης (Compression Ratio)}$$

Αναλογία συμπίεσης = 40 σύμφωνα με την παραδοχή

5. Μέγεθος ήχου (Audio Size)

$$= \text{Ρυθμός ηχητικών δεδομένων (Audio Data Rate)} * \text{Διάρκεια του βίντεο}$$

$$= [(128 * 1024) / 8] * \text{Διάρκεια του βίντεο}$$

όπου το Audio Data Rate έχει αυτή την τιμή λόγω της παραδοχής για την ανάλυση του βίντεο

6. Total Video Size = Compressed Video Size + Audio Size

Αιτήσεις Χρηστών

Επίσης, χρησιμοποιώντας την Γεννήτρια Αιτήσεων [1] συλλέξαμε ένα σύνολο από αιτήσεις χρηστών οι οποίες αποτελούνται από τις ταυτότητες των πραγματικών χρηστών του κοινωνικού δικτύου και τις πραγματικές ταυτότητες των βίντεο του YouTube όπως εξήχθηκαν μετά από την ιχνηλασία. Οι αιτήσεις αυτές, όπως έχει αναφερθεί ξανά, ανταποκρίνονται σε ρεαλιστικές κατανομές αιτήσεων του διαδικτύου κάτι πολύ σημαντικό για την προσομοίωση των πολιτικών αφού τα αποτελέσματα της πειραματικής αξιολόγησης βασίζονται πάνω σε ρεαλιστικά δεδομένα.

Κεφάλαιο 4

Διατύπωση Προβλήματος

4.1 Ορισμός Προβλήματος	28
4.2 Περιγραφή Παραμέτρων	28
4.3 Σκοπός	29

4.1 Ορισμός Προβλήματος

Ζητούμενο είναι η εύρεση κάποιων καλών πολιτικών προ-ανάκλησης περιεχομένου στους υποκατάστατους εξυπηρετητές, λαμβάνοντας υπόψη τα δεδομένα που μας προσφέρουν τα κοινωνικά δίκτυα και οι ενέργειες των χρηστών σε αυτά, έτσι ώστε να εντοπίσουμε αντικείμενα που θα γίνουν δημοφιλή και να τα αντιγράψουμε στους υποκατάστατους εξυπηρετητές. Στόχος είναι η ελαχιστοποίηση του χρόνου απόκρισης και η μεγιστοποίηση του αριθμού ικανοποιημένων αιτήσεων (success rate), δηλαδή η βελτίωση της απόδοσης του δικτύου παράδοσης περιεχομένων. Παράλληλα επιθυμούμε την ελαχιστοποίηση του κόστους αναπαραγωγής του περιεχομένου από τους εξυπηρετητές προέλευσης (origin) στους υποκατάστατους.

4.2 Περιγραφή Παραμέτρων

Για να το πετύχουμε αυτό θα χρησιμοποιήσουμε τα εξής δεδομένα:

- $G(V,E)$: Γράφος που αναπαριστά το κοινωνικό δίκτυο
 - $V = \{V_1, V_2, \dots, V_n\}$: Το σύνολο των κόμβων, δηλ. των χρηστών του κοινωνικού δικτύου.
 - $E \in \{E_{00}, E_{01}, E_{02}, \dots, E_{nn}\}$: Το σύνολο των ακμών που αναπαριστούν τις συνδέσεις των χρηστών στο κοινωνικό δίκτυο. Η ακμή E_{ij} αναπαριστά ότι ο χρήστης i είναι φίλος με τον χρήστη j .

- $R = (r_1, r_2, \dots, r_t)$: Σύνολο περιοχών.
- $N = (n_1, n_2, \dots, n_u)$: Σύνολο των υποκατάστατων εξυπηρετητών.
- C : Χωρητικότητα του κάθε υποκατάστατου εξυπηρετητή.
- $O = (o_1, o_2, \dots, o_w)$: Σύνολο αντικειμένων (στην περίπτωση μας τα αντικείμενα είναι βίντεο).
- S_i : Μέγεθος του αντικειμένου i σε bytes όπου $i \in O$.
- P_i : Δημοτικότητα (popularity) του αντικειμένου i , $i \in O$.
- $Q = \{q_1, q_2, \dots, q_m\}$: Σύνολο αιτήσεων χρηστών (χρησιμοποιώντας τον Request Generator)
 - $q_i = \{o, v, \text{timestamp}\}$, η αίτηση i αποτελείται από την ταυτότητα του αντικειμένου, την ταυτότητα του χρήστη που το ζήτησε και την χρονοσφραγίδα.

4.3 Σκοπός

Χρησιμοποιώντας τα πιο πάνω δεδομένα μπορούμε να πετύχουμε την πρόβλεψη για το ποια αντικείμενα θα γίνουν δημοφιλή, σε ποιες περιοχές και πότε.

Έστω ότι αυτή η συνάρτηση είναι η **Predict(G, Q, R, O)**, η οποία δέχεται σαν παραμέτρους τον κοινωνικό γράφο $G(V, E)$, το σύνολο των αιτήσεων Q , το σύνολο των περιοχών R όπως και το σύνολο των αντικειμένων O . Αυτή η συνάρτηση προβλέπει τα m αντικείμενα $\{o_1, o_2, \dots, o_m\} \in O$ τα οποία θα γίνουν παγκόσμια δημοφιλή και τα t αντικείμενα $\{o_1, o_2, \dots, o_t\} \in O$ τα οποία θα γίνουν τοπικά δημοφιλή. Το δικό μας ενδιαφέρον εστιάζεται στα t αντικείμενα που θα γίνουν τοπικά δημοφιλή.

Με τα αποτελέσματα της $\text{Predict}(G, Q, R, O)$ θα μπορέσουμε να προχωρήσουμε στην εύρεση μιας καλής πολιτικής προ-ανάκλησης περιεχομένων/αντικειμένων, από τους εξυπηρετητές προέλευσης στους υποκατάστατους εξυπηρετητές. Την συγκεκριμένη πολιτική αναπαριστά η συνάρτηση **Put(n_i , Predict(G, Q, R, O))**, η οποία παίρνει σαν δεδομένα εισόδου έναν συγκεκριμένο υποκατάστατο εξυπηρετητή $n_i \in N$, και τα αποτελέσματα της προηγούμενης συνάρτησης, δηλαδή το σύνολο των m αντικειμένων

που θα γίνουν παγκόσμια δημοφιλή και το σύνολο των t αντικειμένων που θα γίνουν τοπικά δημοφιλή. Αυτή η συνάρτηση μας δίνει ως αποτέλεσμα, ποια αντικείμενα $o \in O$ θα πρέπει να τοποθετηθούν στον συγκεκριμένο υποκατάστατο εξυπηρετητή $n_i \in N$, με βάση την πρόβλεψη που γίνεται. Αυτό που επίσης πρέπει να λαμβάνεται υπόψη στην απόφαση αυτή, είναι η χωρητικότητα C_{ni} του επεξεργαστή n_i , η οποία είναι συγκεκριμένη για τον κάθε επεξεργαστή και για τα x αντικείμενα που θα αντιγραφούν στον υποκατάστατο εξυπηρετητή πρέπει να ισχύει η σχέση: $S_{o1}+S_{o2}+S_{o3}+\dots+S_{ox} \leq C_{ni}$.

Η συγκεκριμένη πολιτική θα στοχεύει στη βελτίωση του δικτύου παράδοσης περιεχομένων (CDN) ως προς δύο παράγοντες:

- **Ελαχιστοποίηση του χρόνου απόκρισης και μεγιστοποίηση του αριθμού ικανοποιημένων αιτήσεων**, δηλαδή μεγιστοποίηση του $\text{success rate} = \frac{Q_{hit}}{Q_{total}}$.
- **Ελαχιστοποίηση του κόστους αναπαραγωγής** περιεχομένου από τον εξυπηρετητή προέλευσης στους υποκατάστατους εξυπηρετητές.

Μία καλή πολιτική προ-ανάκλησης περιεχομένου πρέπει να βρίσκεται μεταξύ των δύο ακραίων πολιτικών οι οποίες βελτιώνουν το δίκτυο ως προς τον έναν παράγοντα μόνον.

Αυτές οι δύο **ακραίες πολιτικές** είναι οι εξής:

1. Τοποθέτηση όλου του περιεχομένου σε όλους του υποκατάστατους εξυπηρετητές. Έτσι βελτιώνεται στο μέγιστο ο χρόνος απόκρισης και μεγιστοποιείται το success rate (αφού κάθε αίτηση θα ικανοποιείται αυτόματα) αλλά μεγιστοποιείται το κόστος αναπαραγωγής περιεχομένου.
2. Καμία τοποθέτηση περιεχομένου σε κανέναν από τους υποκατάστατους εξυπηρετητές. Αφού το περιεχόμενο παραδίδεται στον υποκατάστατο εξυπηρετητή την στιγμή που το αιτείται ο χρήστης αυτόματα ελαχιστοποιείται το κόστος αναπαραγωγής του περιεχομένου. Αλλά μεγιστοποιείται ο χρόνος απόκρισης.

Επίσης κατασκευάζονται και δύο **πολιτικές τυχαίας τοποθέτησης** αντικειμένων στους υποκατάστατους εξυπηρετητές – με πιθανότητα τοποθέτησης 10% και 50% - έτσι ώστε να υπάρχει ακόμα ένα μέτρο σύγκρισης στην πολιτική πρόβλεψης και προ-ανάκλησης περιεχομένου.

Κεφάλαιο 5

Πολιτικές Τοποθέτησης και Προ-ανάκλησης Περιεχομένου

5.1 Στατικές Πολιτικές	31
5.2 Δυναμική Πολιτική	35

5.1 Στατικές Πολιτικές

Όπως αναφέρθηκε σε προηγούμενο κεφάλαιο, για την αξιολόγηση της πολιτικής προ-ανάκλησης περιεχομένου, χρειάστηκε να κατασκευαστούν κάποιες στατικές πολιτικές τοποθέτησης αντικειμένων στους υποκατάστατους εξυπηρετητές, έτσι ώστε να αποτελέσουν το μέτρο σύγκρισης για την δυναμική πολιτική.

Οι στατικές πολιτικές αποτελούν τις πολιτικές στις οποίες τα αντικείμενα τοποθετούνται στους υποκατάστατους εξυπηρετητές προτού προσομοιώσουμε την κίνηση των αιτήσεων στον προσομοιωτή CDNSim. Ο προσομοιωτής είναι αυτός που ακολούθως θα αποφασίσει πως θα κινηθούν τα δεδομένα στο δίκτυο και στους εξυπηρετητές. Δεν υπάρχει καμία επιπλέον δυναμική απόφαση για το πως θα κινηθούν τα δεδομένα κατά τη διάρκεια άφιξης αιτήσεων.

Οι τέσσερις πιο κάτω πολιτικές τοποθέτησης περιεχομένου όπου πρόκειται να επεξηγηθούν, όπως και τα στατιστικά αποτελέσματα που μας έχουν δώσει, αφότου έγινε η προσομοίωση τους στο εργαλείο CDNSim (τα οποία αναλύονται στο επόμενο κεφάλαιο), αποτελούν μέτρο σύγκρισης ως προς τα αποτελέσματα της δυναμική πολιτική πρόβλεψης και προ-ανάκλησης περιεχομένου στους υποκατάστατους εξυπηρετητές που έχει προταθεί. Τα συμπεράσματα της σύγκρισης αυτής θα καθορίσουν το κατά πόσον η έξυπνη πολιτική, αποτελεί μία αξιόλογη πολιτική για πρακτική εφαρμογή.

Οι δύο ακραίες πολιτικές δίνουν τα όρια στα οποία πρέπει τα αποτελέσματα της «έξυπνης» πολιτικής να κυμαίνονται. Οι δύο πολιτικές τυχαίας τοποθέτησης αντικειμένων, δίνουν πληροφορία για το κατά πόσον η «έξυπνη» πολιτική είναι κατάλληλη για πρακτική εφαρμογή. Αν τα αποτελέσματα της «έξυπνης» πολιτικής που έχει προταθεί είναι χειρότερα από αυτά των πολιτικών τυχαίας τοποθέτησης, τότε δεν αξίζει να εφαρμοσθεί πρακτικά.

Πιο κάτω, αναφέρονται και επεξηγούνται οι τέσσερις αυτές πολιτικές. Οι δύο πρώτες είναι πολιτικές συγκεκριμένης τοποθέτησης περιεχομένου στους υποκατάστατους εξυπηρετητές και οι δύο επόμενες αφορούν τυχαία τοποθέτηση της πληροφορίας στους υποκατάστατους εξυπηρετητές.

Ακραία Πολιτική 1 - Τοποθέτηση όλου του περιεχομένου σε όλους τους υποκατάστατους εξυπηρετητές

Η συγκεκριμένη πολιτική αναφέρεται στην περίπτωση όπου όλο το περιεχόμενο, δηλαδή όλα τα αντικείμενα του συνόλου αιτήσεων που έχουμε στη διάθεσή μας, τοποθετούνται σε όλους τους υποκατάστατους εξυπηρετητές.

Πιο συγκεκριμένα, για την υλοποίηση της πολιτικής αυτής, τοποθετήσαμε όλα τα αντικείμενα που παρουσιάζονταν σε όλες τις αιτήσεις, σε όλους τους υποκατάστατους εξυπηρετητές. Ακολούθως, ο προσομοιωτής αναλαμβάνει να εκτελέσει τις αιτήσεις που του δίνουμε, χωρίς να υπάρχει καμία παρέμβαση στον τρόπο λειτουργίας του.

Αναμένουμε από την συγκεκριμένη πολιτική να έχει τα καλύτερα αποτελέσματα σε θέματα χρόνου απόκρισης και αριθμού ικανοποιημένων αιτήσεων. Ο λόγος έγκειται στο γεγονός ότι οι αιτήσεις των χρηστών θα μπορούν να εξυπηρετηθούν αυτόματα από τον πρώτο υποκατάστατο εξυπηρετητή στον οποίο θα αποσταλεί η αίτηση από τον προσομοιωτή, χωρίς να χρειάζεται η οποιαδήποτε ανακατεύθυνση. Είναι εμφανές όμως από μια απλή περιγραφή της πολιτικής αυτής, ότι η πρακτική υλοποίησή της είναι απαγορευτική, αφού έχει τεράστιο κόστος αναπαραγωγής περιεχομένου – απαιτεί τεράστια κρυφή μνήμη (cache) στους υποκατάστατους εξυπηρετητές για να μπορούν να τοποθετηθούν όλα τα αντικείμενα σε αυτούς.

Ακραία Πολιτική 2 - Καμία τοποθέτηση περιεχομένου σε κανέναν από τους υποκατάστατους εξυπηρετητές

Στην συγκεκριμένη πολιτική, η οποία είναι η εντελώς αντίθετη από την προηγούμενη, δεν τοποθετείται κανένα από τα αντικείμενα στους υποκατάστατους εξυπηρετητές πριν να εκτελέσουμε την προσομοίωση. Όλοι οι υποκατάστατοι εξυπηρετητές δηλαδή, είναι εντελώς άδειοι στην αρχή της προσομοίωσης.

Για την υλοποίηση της συγκεκριμένης ακραίας πολιτικής, τοποθετήσαμε όλα τα αντικείμενα μόνο στον εξυπηρετητή προέλευσης και αφήσαμε όλους τους υποκατάστατους εξυπηρετητές κενούς. Και πάλι αφήσαμε τον προσομοιωτή να εκτελέσει τις αιτήσεις χωρίς καμία περεταίρω παρέμβαση στην λειτουργία του και στον τρόπο κίνησης του περιεχομένου.

Σε αντίθεση με την προηγούμενη πολιτική, αυτή η πολιτική έχει μηδαμινό κόστος αναπαραγωγής του περιεχομένου. Αναμένουμε όμως ότι θα έχει πολύ πιο μεγάλο χρόνο απόκρισης από την προηγούμενη όπως και πιο μικρό αριθμό εξυπηρετούμενων αιτήσεων, αφού πλέον οι υποκατάστατοι εξυπηρετητές θα πρέπει να αιτηθούν τα αντικείμενα των αιτήσεων από τον εξυπηρετητή προέλευσης για να εξυπηρετήσουν την αίτηση του χρήστη. Άρα, ούτε και αυτή η πολιτική μπορεί να υλοποιηθεί στην πράξη.

Πολιτική τυχαίας τοποθέτησης 1 – Πιθανότητα 10%

Όπως και οι δύο προηγούμενες, έτσι και οι επόμενες δύο πολιτικές είναι στατικές. Ο κώδικας της υλοποίησής τους περιέχεται στο Παράρτημα Α. Εδώ, αντί για μια συγκεκριμένη τοποθέτηση αντικειμένων, παίρνουμε την απόφαση τοποθέτησης κάθε αντικειμένου με μία πιθανότητα p .

Για την υλοποίηση της συγκεκριμένης πολιτικής, τοποθετούμε κάθε αντικείμενο σε κάθε υποκατάστατο εξυπηρετητή με πιθανότητα 10%. Αναμένεται ότι ο κάθε υποκατάστατος εξυπηρετητής, μετά την τοποθέτηση του περιεχομένου, θα έχει στην κρυφή μνήμη του το 10% του συνόλου των αντικειμένων. Ακολουθώντας, όπως και προηγουμένως, προσομοιώνουμε το σενάριο στο CDNSim χωρίς καμία επιπλέον παρέμβαση στην λειτουργία του.

Δεν μπορεί να γίνει κάποια πρόβλεψη για τα αποτελέσματα που πρόκειται να δώσει ο προσομοιωτής μετά την εκτέλεση των αιτήσεων, αφού η τοποθέτηση του περιεχομένου αρχικά γίνεται τυχαία έτσι δεν ξέρουμε πως θα αντιδράσει το δίκτυο.

Πολιτική τυχαίας τοποθέτησης 2 – Πιθανότητα 50%

Η υλοποίηση της δεύτερης πολιτικής τυχαίας τοποθέτησης περιεχομένου, γίνεται με τον ίδιο τρόπο όπως και με την πρώτη. Η διαφορά εδώ είναι ότι χρησιμοποιούμε πιθανότητα $p=50\%$, δηλαδή τοποθετούμε τα αντικείμενα στους υποκατάστατους εξυπηρετητές χρησιμοποιώντας μεγαλύτερη πιθανότητα. Αυτό δείχνει ότι σε αυτή την περίπτωση, με την έναρξη της προσομοίωσης, ανά υποκατάστατο εξυπηρετητή θα βρίσκεται το μισό σύνολο περίπου των αντικειμένων που εμφανίζονται στις αιτήσεις.

Δεν μπορούμε να προβλέψουμε τα αποτελέσματα του προσομοιωτή μετά την εκτέλεση του σεναρίου αυτού. Όμως μπορούμε να υποθέσουμε ότι το πιο πιθανόν είναι να έχει καλύτερα αποτελέσματα από την προηγούμενη πολιτική τυχαίας τοποθέτησης, αφού πλέον σε κάθε εξυπηρετητή βρίσκεται μεγαλύτερος αριθμός αντικειμένων έτσι λογικά θα εξυπηρετούνται πιο εύκολα, περισσότερες αιτήσεις χρηστών.

5.2 Δυναμική Πολιτική

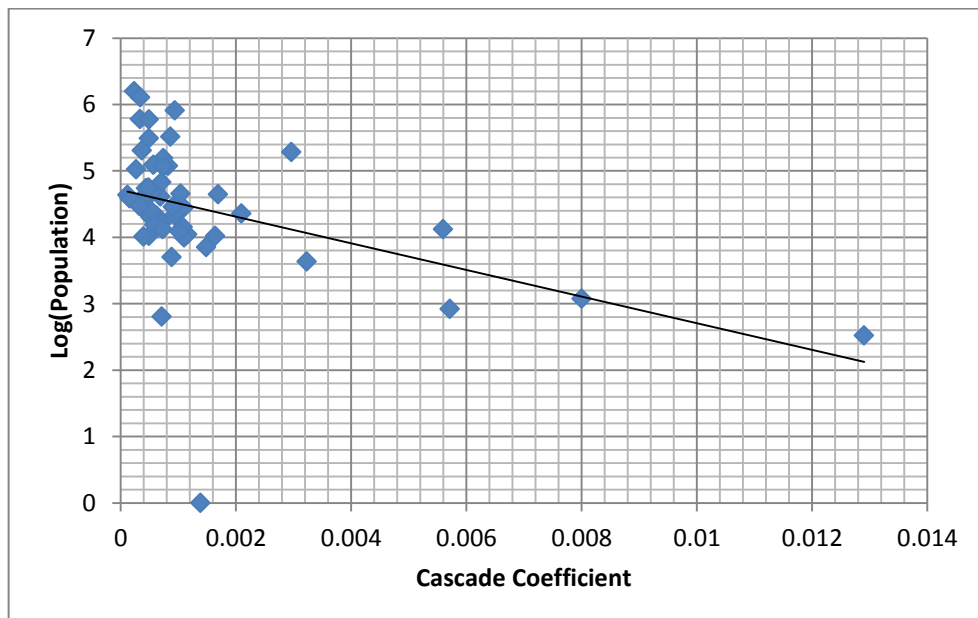
Μία δυναμική πολιτική, στοχεύει στην προ-ανάκληση και αντιγραφή του περιεχομένου που πρόκειται να ζητηθεί από τους χρήστες, πριν ακόμα αυτό ζητηθεί. Η διαφορά μεταξύ της συγκεκριμένης πολιτικής και των προηγούμενων, είναι ότι σε μία δυναμική πολιτική απαιτείται η λήψη αποφάσεων κατά τη διάρκεια της προσομοίωσης και όχι προηγουμένως. Με την έλευση λοιπόν μίας καινούριας αίτησης από κάποιον χρήστη, πρέπει η πολιτική αυτή να είναι σε θέση να αποφασίσει πού θα αντιγραφεί το περιεχόμενο αυτό – τοπικά ή καθολικά;

Μία τέτοια δυναμική πολιτική μπορεί να συμβάλει δραματικά στη βελτίωση των Δικτύων Παράδοσης Περιεχομένου, ελαχιστοποιώντας το χρόνο απόκρισης του δικτύου, χωρίς να απαιτεί τεράστιο κόστος αναπαραγωγής του περιεχομένου.

5.2.1 Περιγραφή Συμπερασμάτων από το [1]

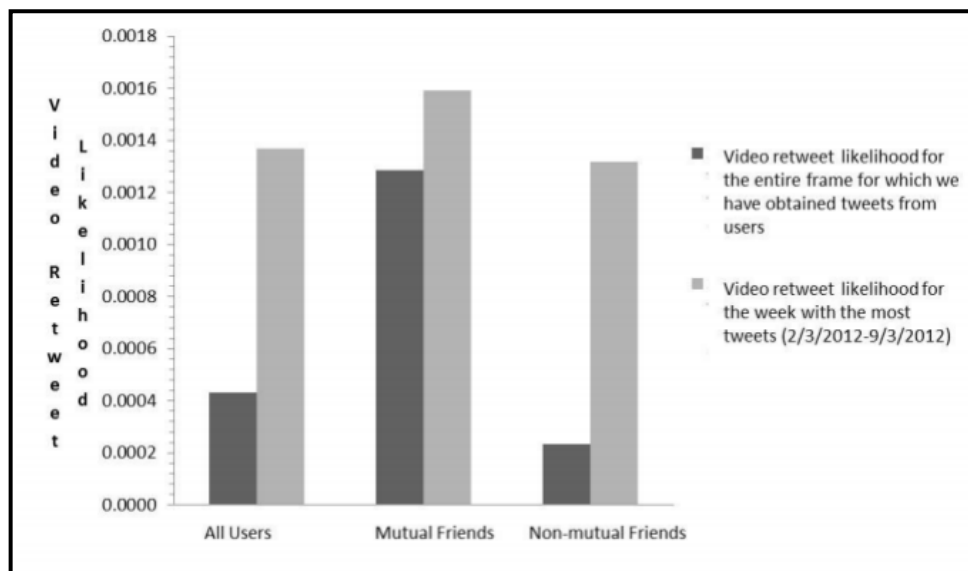
Για την κατασκευή και τον ορισμό της δυναμικής πολιτικής που παρουσιάζεται πιο κάτω, χρησιμοποιήθηκαν συμπεράσματα τα οποία εξήλθαν από την έρευνα και την μελέτη των πραγματικών δεδομένων μετά την ιχνηλασία στο [1]. Φαίνεται ξεκάθαρα ότι οι χρήστες και οι αιτήσεις τους στο διαδίκτυο, επηρεάζονται από τα κοινωνικά δίκτυα με βάση κάποιους παράγοντες [1]. Οι παράγοντες αυτοί που αναφέρονται και επεξηγούνται πιο κάτω, έχουν χρησιμοποιηθεί για την κατασκευή της δυναμικής πολιτικής στην παρούσα Διπλωματική Εργασία.

- **Οι χρήστες επηρεάζονται περισσότερο από τους γεωγραφικά κοντινούς τους φίλους.** Χρησιμοποιώντας τις ζώνες ώρας που ορίζονται στο κοινωνικό δίκτυο Twitter έχει αποδειχθεί ότι οι χρήστες επηρεάζονται πολύ περισσότερο από τους χρήστες που είναι πιο κοντά τους [1]. Ο συντελεστής διάδοσης είναι πολύ πιο ψηλός σε ζώνες ώρας με μικρό πληθυσμό ενώ χαμηλότερος σε ζώνες ώρας με πολύ μεγάλο πληθυσμό (Σχήμα 5.1). Αυτό σημαίνει ότι η κοινωνική διάδοση έχει μεγάλη επιρροή σε πιο εστιασμένες γεωγραφικές περιοχές και ως εκ τούτου, οι χρήστες επηρεάζονται περισσότερο από τις δημοσιεύσεις φίλων τους που βρίσκονται γεωγραφικά κοντά [1].



Σχήμα 5.1: Συντελεστής διάδοσης σε σχέση με το μέγεθος των ζωνών ώρας σε λογαριθμική κλίμακα [1]

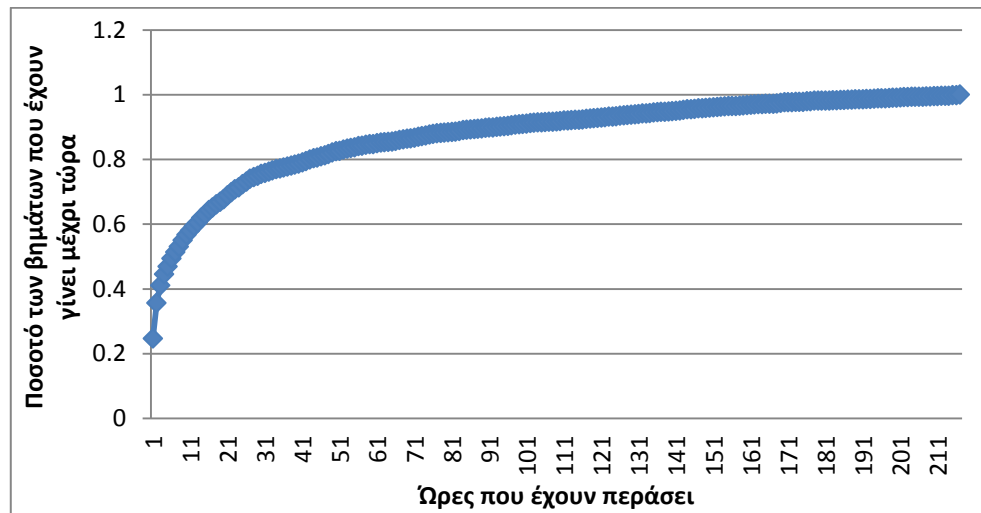
- Οι χρήστες επηρεάζονται περισσότερο από φίλους με τους οποίους έχουν αμφίδρομο σύνδεσμο στο κοινωνικό δίκτυο Twitter. Υπενθυμίζουμε ότι με την έννοια «αμφίδρομος σύνδεσμος» στον κοινωνικό γράφο του Twitter μεταξύ των χρηστών A και B, εξυπακούεται ότι και ο A είναι «ακόλουθος» του B αλλά και το αντίστροφο. Όπως φαίνεται και στο Σχήμα 5.2, η πιθανότητα κάποιος χρήστης να αναδημοσιεύσει ένα βίντεο το οποίο έχει δημοσιευθεί από έναν δεύτερο χρήστη στο κοινωνικό δίκτυο Twitter, είναι κατά 6 φορές μεγαλύτερη όταν η σχέση των δύο χρηστών είναι αμοιβαία [1]. Η πιθανότητα διαφοροποιείται όταν πρόκειται για την εβδομάδα με τις περισσότερες δημοσιεύσεις, όπου εκεί βλέπουμε υψηλές πιθανότητες και στους μη-αμοιβαίους φίλους αλλά στο εύρος της χρονικής διάρκειας των κοινωνικών διαδόσεων, οι κοινοί φίλοι κάνουν την διαφορά [1].



Σχήμα 5.2: Πιθανότητα ανά-δημοσίευσης βίντεο ανάλογα με το είδος της σχέσης μεταξύ των χρηστών [1]

- Το μεγαλύτερο ποσοστό των κοινωνικών διαδόσεων ολοκληρώνεται μέσα σε 24 ώρες. Συγκεκριμένα, το 25% των διαδόσεων ολοκληρώνεται εντός 1 ώρας και μέχρι το τέλος της πρώτης μέρας ολοκληρώνεται το 75-80% των διαδόσεων (Σχήμα 5.3) [1]. Επίσης, για το ποσοστό της διάδοσης που συνεχίζεται τις επόμενες μέρες ίσως να μην οφείλεται στο κοινωνικό δίκτυο

(αφού το βίντεο δεν εμφανίζεται στην κοινωνική ροή του χρήστη επ' αόριστον) ή ίσως να έχει ξεκινήσει νέα κοινωνική διάδοση για το συγκεκριμένο βίντεο [1].



Σχήμα 5.3: Ποσοστό κοινωνικής διάδοσης που έχει ολοκληρωθεί ανάλογα με την ώρα που έχει περάσει [1]

- **Οι κοινωνικές διαδόσεις επηρεάζονται από τον αριθμό θεάσεων των βίντεο στο YouTube.** Όσες πιο πολλές είναι οι θεάσεις ενός βίντεο στο YouTube, τόσες περισσότερες είναι και οι κοινωνικές διαδόσεις που αφορούν αυτό το βίντεο [1]. Ο αριθμός θεάσεων ενός βίντεο στο YouTube, έμμεσα υποδηλώνει την δημοτικότητα (popularity) του βίντεο, άρα όσο πιο δημοφιλές είναι ένα βίντεο, όλο και περισσότεροι χρήστες τείνουν να το δημοσιεύουν στην κοινωνική τους ροή. Επίσης, φαίνεται ότι σε κάποιο σημείο υπάρχει μεγάλη αύξηση του αριθμού των διαδόσεων ενώ οι θεάσεις παραμένουν σταθερές. Από το φαινόμενο αυτό μπορεί να συμπεράνουμε ότι οι κοινωνικές διαδόσεις ίσως να συμβάλλουν στην ανάπτυξη των θεάσεων των βίντεο στο YouTube [1].

5.2.2 Σημαντικοί Αλγόριθμοι και Ορολογίες

Αλγόριθμος HITS Hyperlink-Induced Topic Search

Ο αλγόριθμος HITS γνωστός και ως «Hubs and Authorities» αλγόριθμος, είναι ένας αλγόριθμος ανάλυσης συνδέσμων σελίδων, ο οποίος βαθμολογεί σελίδες διαδικτύου, και αναπτύχθηκε από τον Jon Kleinberg [15]. Στο συγκεκριμένο αλγόριθμο, η κάθε ιστοσελίδα έχει μία τιμή η οποία την χαρακτηρίζει ως «hub» και μία τιμή η οποία την χαρακτηρίζει ως «authority».

- «Hub», δηλαδή κομβικό σημείο, είναι μία σελίδα η οποία έχει συνδέσμους σε authority σελίδες, δηλαδή σελίδες ειδικές για κάποιο θέμα. Ουσιαστικά, τα hubs λειτουργούν ως κατάλογοι για σελίδες με συγκεκριμένα θέματα (λίστες εφημερίδων, κατάλογοι μαθημάτων κτλ). Για να βρεθεί η τιμή του hub, αθροίζονται οι τιμές authority των σελίδων προς τις οποίες έχει συνδέσμους.
- «Authority», δηλαδή αυθεντία, είναι μία σελίδα η οποία περιέχει περιεχόμενο για συγκεκριμένο θέμα (για παράδειγμα η πραγματική σελίδα της εφημερίδας ή του μαθήματος). Για να βρεθεί η τιμή του authority, αθροίζονται οι τιμές hub των σελίδων από τις οποίες δέχεται συνδέσμους.

Ένα καλό hub έχει συνδέσμους προς καλά authorities ενώ ένα καλό authority δέχεται συνδέσμους από καλά hubs. Συμπεραίνουμε λοιπόν ότι αν μία σελίδα είναι καλό hub δεν εξυπακούεται ότι θα αποτελεί και καλό authority και αντίστροφα.

Ο αλγόριθμος HITS παίρνει σαν είσοδο έναν κατευθυνόμενο γράφο στον οποίο οι κόμβοι αντιπροσωπεύουν τις ιστοσελίδες και οι ακμές αντιπροσωπεύουν συνδέσμους από μία ιστοσελίδα σε άλλη. Ο αλγόριθμος λειτουργεί ως εξής:

- Αρχικοποίηση των τιμών hub και authority της κάθε σελίδας στο γράφο με 1.
- Με αναδρομή κάνε τα εξής για κάθε κόμβο X στον γράφο:
 - Ενημέρωση στην τιμή authority: άθροισμα των τιμών hub, των κόμβων οι οποίοι έχουν συνδέσμους προς τον κόμβο X.
 - Ενημέρωση στην τιμή hub: άθροισμα των τιμών authority, των κόμβων εκείνων προς τους οποίους ο κόμβος X έχει συνδέσμους.
 - Κανονικοποίηση των τιμών hub και authority διαιρώντας με την τετραγωνική ρίζα του αθροίσματος των τετραγώνων όλων των τιμών hub και authority αντίστοιχα.

Η διαδικασία συνεχίζεται μέχρι οι τιμές των δύο χαρακτηριστικών να συγκλίνουν και να μην αλλάζουν.

Στην δυναμική πολιτική που προτείνουμε, εφαρμόζουμε τον αλγόριθμο HITS στον κοινωνικό γράφο στον οποίο φαίνονται οι σχέσεις και οι σύνδεσμοι μεταξύ των

χρηστών του κοινωνικού δικτύου Twitter. Στόχος είναι ο εντοπισμός των χρηστών με ψηλό σκορ authority. Πιστεύουμε ότι οι συγκεκριμένοι χρήστες είναι διάσημοι, με πολλούς ακόλουθους. Ένα βίντεο μίας δημοσίευσης από κάποιον «authority» χρήστη στο κοινωνικό δίκτυο, μπορεί να ζητηθεί από πολλούς χρήστες οι οποίοι είναι ακόλουθοι του συγκεκριμένου χρήστη «authority». Κάτι τέτοιο μας οδηγεί στην πεποίθηση ότι το περιεχόμενο των δημοσιεύσεων από χρήστες «αυθεντίες» σύμφωνα με τον αλγόριθμο HITS, πρέπει να προ-ανακαλείται σε καθολικό επίπεδο αφού ενδέχεται να ζητηθεί από πολλούς χρήστες σε μεγάλη κλίμακα.

Lobby-Index για δίκτυα

Το lobby-index είναι μία μετρική κεντρικότητας των κόμβων σε δίκτυα η οποία έχει προταθεί το 2008 από τους A. Korn, A. Schubert και A. Telcs [16].

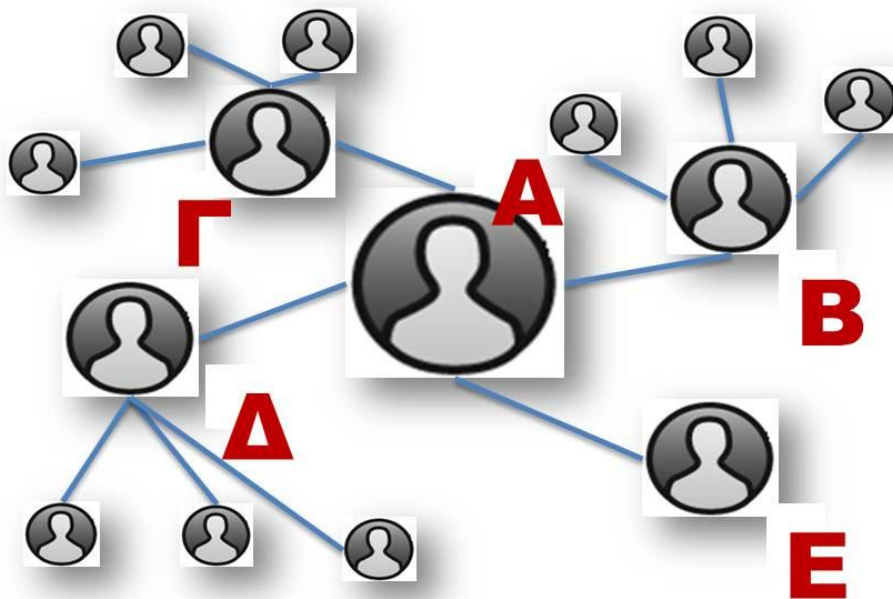
Το lobby-index ή αλλιώς l-index ενός κόμβου u σε κάποιο γράφο, συμβολισμένο ως $l_u(t)$, είναι ο μεγαλύτερος ακέραιος αριθμός k τέτοιος ώστε ο αριθμός των γειτονικών κόμβων του u οι οποίοι έχουν βαθμό ίσο με k , να είναι k , τη χρονική στιγμή t [16, 23]. Δηλαδή ο κόμβος u να έχει k γείτονες οι οποίοι έχουν k γείτονες.

Έστω ότι $G=(V,E)$ ο κοινωνικός γράφος του Twitter. Ορίζουμε ως N_u το σύνολο των γειτόνων του κόμβου $u \in V$ και $\deg(u)$ είναι ο βαθμός του κόμβου στο γράφο, δηλαδή ο αριθμός των γειτόνων του.

$\forall$ κόμβο $y_i \in N_u$, τέτοιο ώστε $\deg(y_1) \geq \deg(y_2) \dots$ τότε το l-index του u τη χρονική στιγμή t δίδεται από την εξής σχέση:

$$l_u(t) = \max \{k : \deg(y_k) \geq k\} \quad [16]$$

Παράδειγμα της μετρικής αυτής έτσι όπως εφαρμόζεται στα κοινωνικά δίκτυα, φαίνεται στο Σχήμα 5.4. Όπως φαίνεται, ο χρήστης A είναι φίλος με τους χρήστες B, Γ, Δ, E. Οι χρήστες B, Γ και Δ έχουν από 4 φίλους ο κάθε ένας (τρεις διαφορετικούς και τον A). Ο χρήστης E όμως, είναι φίλος μόνον με το χρήστη A. Το $l(A)$ λοιπόν, είναι ο μέγιστος βαθμός k , έτσι ώστε να έχει k φίλους οι οποίοι έχουν k φίλους. Ο αριθμός αυτός λοιπόν είναι το 3, στην περίπτωση του A. Αν και έχει 4 φίλους, μόνο οι τρεις από αυτούς έχουν και αυτοί από 3 φίλους.



Σχήμα 5.4: Παράδειγμα για τη μετρική lobby-index.
 $l_A(t) = \max \{k : \deg(yk) \geq k\} = 3$

Betweenness Κεντρικότητα (Centrality) για δίκτυα

Όπως και το lobby-index έτσι και η συγκεκριμένη μετρική, υπολογίζει την κεντρικότητα ενός κόμβου σε έναν κατευθυνόμενο γράφο και χρησιμοποιείται ευρέως για γράφους του διαδικτύου, κοινωνικών δικτύων κ.ά. Με τη συγκεκριμένη μετρική μπορεί κανείς να υπολογίσει το φορτίο αλλά και την σημασία του κάθε κόμβου σε σχέση με τους υπόλοιπους. Το Betweenness Centrality αποδίδεται κυρίως στον κοινωνιολόγο Linton Freeman ο οποίος είχε επίσης αναπτύξει μία σειρά από άλλα μέτρα κεντρικότητας [17], αν και η ιδέα αυτής της μετρικής είχε προταθεί νωρίτερα από τον μαθηματικό J. Anthonisse, αλλά το έργο του δε δημοσιεύθηκε ποτέ.

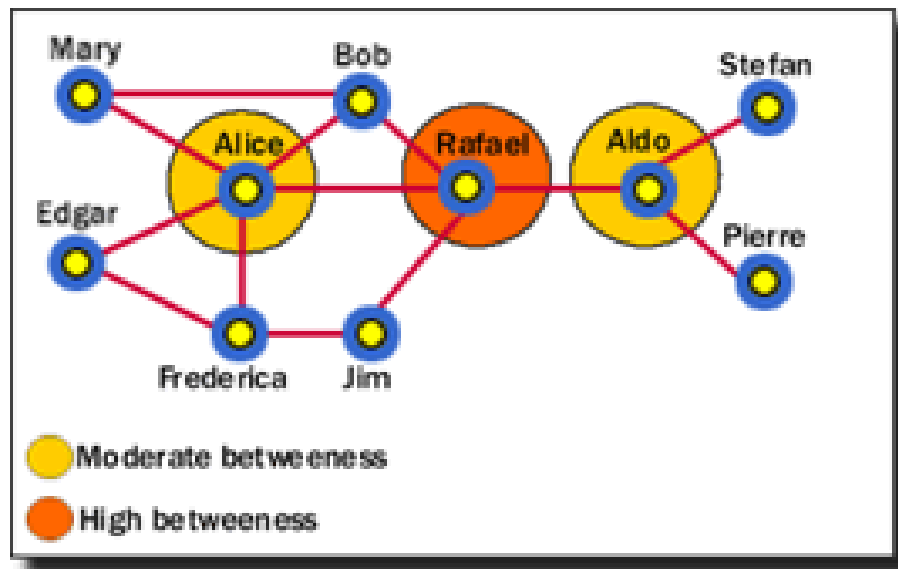
Η κεντρικότητα Betweenness ενός κόμβου x , ισούται με τον αριθμό των συντομότερων μονοπατιών από όλους τους κόμβους σε όλους τους άλλους που περνούν μέσα από αυτόν τον κόμβο, σε κάποια χρονική στιγμή t .

Πιο συγκεκριμένα, έστω ότι $G=(V,E)$ ο κοινωνικός γράφος του Twitter. Η κεντρικότητα Betweenness ενός κόμβου $u \in V$ δίδεται από την εξής μαθηματική έκφραση [17]:

$$g_u(t) = \sum_{j \neq u \neq k} \frac{\sigma_{jk}(u, t)}{\sigma_{jk}(t)}$$

Όπου το $\sigma_{jk}(t)$ είναι ο συνολικός αριθμός των συντομότερων μονοπατιών από τον κόμβο j στον κόμβο k τη χρονική στιγμή t , και $\sigma_{jk}(u, t)$ είναι ο αριθμός των συντομότερων μονοπατιών από τον κόμβο j στον κόμβο k τα οποία περνούν μέσω του u τη χρονική στιγμή t [17, 23].

Στο Σχήμα 5.5 φαίνεται ένα παράδειγμα υπολογισμού της betweenness κεντρικότητας [22]. Το χρώμα πορτοκαλί φανερώνει υψηλή κεντρικότητα ενώ το χρώμα κίτρινο δείχνει μέτρια κεντρικότητα. Βλέπουμε λοιπόν ότι ο Rafael έχει την υψηλότερη κεντρικότητα αφού βρίσκεται ανάμεσα στην Alice και στον Aldo οι οποίοι έχουν μέτρια κεντρικότητα. Υπάρχουν πολλά μονοπάτια που περνούν από τον Rafael, ο οποίος βρίσκεται ενδιάμεσα στο γράφο, και αυτό του δίνει υψηλή betweenness κεντρικότητα. Αν και η Alice έχει τους περισσότερους συνδέσμους, είναι κεντρική μόνο σε δική της «ομάδα» και όχι σε όλο το εύρος του γράφου [22].



Σχήμα 5.5: Παράδειγμα για τη μετρική betweenness centrality. Ο Rafael έχει την υψηλότερη κεντρικότητα αφού βρίσκεται ανάμεσα στην Alice και στον Aldo οι οποίοι έχουν μέτρια κεντρικότητα. [22]

5.2.3 Γενική Ιδέα Δυναμικής Πολιτικής

Η ιδέα της συγκεκριμένης πολιτικής είναι η τοποθέτηση των αντικειμένων στους υποκατάστατους εξυπηρετητές είτε τοπικά είτε καθολικά, ανάλογα με τα χαρακτηριστικά των περιεχομένων μίας συγκεκριμένης αίτησης για κάποιο αντικείμενο/βίντεο από κάποιο χρήστη.

Γενικές Πληροφορίες

Τα αντικείμενα χαρακτηρίζονται από κάποια χρονική μεταβλητή (timer) η οποία δείχνει τον χρόνο που πέρασε από την πρώτη εμφάνιση του αντικειμένου σε κάποια αίτηση κατά τη διάρκεια της προσομοίωσης. Η μεταβλητή αυτή χρησιμεύει στο να υπολογίζεται ο χρόνος που έχει περάσει από την έναρξη της κοινωνικής διάδοσης για το συγκεκριμένο αντικείμενο, και ανάλογα να αποφασίζεται αν θα αντιγραφεί ή όχι σε κάποιον εξυπηρετητή.

Για τον κάθε χρήστη έχουμε την πληροφορία της περιοχής του, δηλαδή τη ζώνη ώρας που έχει καθορίσει ο ίδιος στο κοινωνικό δίκτυο Twitter. Επίσης, έχουμε τον κοινωνικό γράφο του Twitter ο οποίος μας δίνει την πληροφορία για τους συνδέσμους μεταξύ των χρηστών και τους ακολούθους του κάθε χρήστη. Τέλος, χρησιμοποιώντας την ιδέα του αλγορίθμου HITS χαρακτηρίζουμε τον κάθε χρήστη ως authority ή όχι, κάτι που θα καθορίσει κατά πόσον το βίντεο θα αντιγραφεί καθολικά ή τοπικά.

Κεντρική Ιδέα για κάθε νέα αίτηση

Αρχικά, ερχόμενης μίας αίτησης, γίνεται ο έλεγχος κατά πόσον η κοινωνική διάδοση του αντικειμένου που εμφανίζεται στην αίτηση έχει ξεπεράσει τις 24 ώρες ή όχι.

Σε περίπτωση που η κοινωνική διάδοση ξεπέρασε το φράγμα των 24 ωρών τότε δεν θα εφαρμοσθεί καμία προ-ανάκληση στο συγκεκριμένο βίντεο σε οποιονδήποτε άλλο υποκατάστατο εξυπηρετητή εκτός αυτού που θα εξυπηρετήσει την αίτηση. Η απόφαση αυτή στηρίζεται στο συμπέρασμα ότι το μεγαλύτερο ποσοστό μίας κοινωνικής διάδοσης ολοκληρώνεται εντός των 24 ωρών [1].

Σε περίπτωση που η κοινωνική διάδοση δεν έχει ξεπεράσει τις 24 ώρες από την στιγμή έναρξης της τότε θα πρέπει να καθορισθεί το είδος της προ-ανάκλησης του αντικειμένου – καθολικά ή τοπικά;

Καθολική αντιγραφή γίνεται εάν ο χρήστης που έκανε την αίτηση είναι authority σύμφωνα με τα αποτελέσματα του αλγορίθμου HITS. Εάν ο χρήστης είναι πράγματι authority τότε ενδέχεται το βίντεο το οποίο ο ίδιος έχει δημοσιεύσει να γίνει δημοφιλές καθολικά λόγω του ότι ο ίδιος έχει υψηλή δημοτικότητα.

Τοπική αντιγραφή γίνεται εάν ο χρήστης δεν είναι authority. Έτσι το αντικείμενο, εκτός από τον εξυπηρετητή ο οποίος παρέλαβε την αίτηση, θα τοποθετηθεί και σε κάποιους άλλους υποκατάστατους εξυπηρετητές. Η επιλογή αυτών των εξυπηρετητών μπορεί να γίνει με διάφορες προσεγγίσεις οι οποίες περιγράφονται και επεξηγούνται αναλυτικά σε μετέπειτα υπό-κεφάλαιο. Ουσιαστικά, στο σημείο αυτό χρησιμοποιείται το συμπέρασμα ότι οι χρήστες επηρεάζονται από τους κοντινούς τους φίλους και αυτούς με τους οποίους η σχέση τους στο κοινωνικό δίκτυο είναι αμοιβαία.

5.2.4 Προετοιμασία Δεδομένων Εισόδου

Προτού ξεκινήσει η προσομοίωση και συνεπώς η λειτουργία της δυναμικής πολιτικής, είναι απαραίτητο να γίνει κάποια προετοιμασία στα υπάρχοντα δεδομένα έτσι ώστε να ετοιμαστούν οι είσοδοι για την λειτουργία της δυναμικής πολιτικής. Πιο κάτω, περιγράφονται οι απαραίτητες προκαταρκτικές λειτουργίες.

1. Υπολογισμός των γεωγραφικών αποστάσεων μεταξύ των ζωνών ώρας έτσι ώστε κατά την τοπική αντιγραφή στη δυναμική πολιτική να επιλέγονται εύκολα οι X κοντινότερες ζώνες ώρας στις οποίες ο κάθε χρήστης έχει αμοιβαίους φίλους.
2. Σε αυτό το βήμα θα γίνει είτε το A είτε το B, ανάλογα με την μετρική κεντρικότητας που θέλουμε να χρησιμοποιήσουμε στον αλγόριθμο.

A - Υπολογισμός του lobby-index του κάθε χρήστη για κάθε ζώνη ώρας στην οποία έχει αμοιβαίους φίλους. Η πληροφορία αυτή σε συνδυασμό με άλλες πληροφορίες αποτελεί κριτήριο κατά την τοπική αντιγραφή στη δυναμική πολιτική.

B - Υπολογισμός του betweenness centrality του κάθε χρήστη για κάθε ζώνη ώρας στην οποία έχει αμοιβαίους φίλους. Η πληροφορία αυτή αποτελεί εναλλακτικό κριτήριο για την επιλογή της τοπικής αντιγραφής στη δυναμική πολιτική.

3. Υπολογισμός των διαφορών ώρας μεταξύ όλων των ζωνών ώρας. Ο συγκεκριμένος προ-υπολογισμός γίνεται έτσι ώστε, σε μετέπειτα στάδιο, η προ-ανάκληση του περιεχομένου σε κάθε ζώνη ώρας να προγραμματίζεται σύμφωνα με τη διαφορά ώρας. Η λειτουργία αυτή αποτελεί μία από τις βελτιώσεις της δυναμικής πολιτικής και επεξηγείτε σε μετέπειτα υπό-κεφάλαιο.
4. Τρέχουμε τον αλγόριθμο HITS στο κατευθυνόμενο γράφο του κοινωνικού δικτύου Twitter έτσι ώστε σε κάθε χρήστη να ανατεθεί μία τιμή που θα τον χαρακτηρίζει ως hub και μία τιμή που θα τον χαρακτηρίζει ως authority. Όπως έχει προαναφερθεί, ο στόχος εδώ είναι η εύρεση των k χρηστών με το πιο ψηλό «authority» σκορ, οι οποίοι θα θεωρούνται διάσημοι και τα βίντεο που αιτούνται οι ίδιοι, θα τυχαίνουν καθολικής αντιγραφής.

5.2.5 Αλγόριθμος Δυναμικής Πολιτικής

Πιο κάτω παραθέτω τον αλγόριθμο της δυναμικής πολιτικής σε μορφή ψευδοκώδικα, ο οποίος θα εκτελείται δυναμικά κατά τη διάρκεια της προσομοίωσης.

Ο αλγόριθμος χωρίζεται σε δύο μέρη: τις λειτουργίες του δικτύου παράδοσης περιεχομένου για κάθε νέα αίτηση και τις λειτουργίες του κάθε υποκατάστατου εξυπηρετητή με την έλευση κάθε νέου αντικειμένου στην κρυφή μνήμη του.

Αλγόριθμος για κάθε καινούρια αίτηση <timestamp, U, O> στο Δίκτυο Παράδοσης

Περιεχόμενου:

```
//Έλεγχος αν είναι η πρώτη εμφάνιση του αντικειμένου
//Η μεταβλητή o.timestamp δείχνει την χρονοσφραγίδα της τελευταίας εμφάνισης του αντικειμένου σε μία
//αίτηση και βοηθά στον υπολογισμό του timer το οποίο καθορίζει την διάρκεια του cascade
IF (o.timestamp == 0)    //Το συγκεκριμένο αντικείμενο δεν έχει ξανά-εμφανιστεί σε αίτηση
    o.timer = 0          //Χρονόμετρο για την κοινωνική διάδοση του αντικειμένου
    o.timestamp = request_timestamp    //Ο χρόνος έναρξης της κοινωνικής διάδοσης του
                                        //αντικειμένου ισούται με την χρονοσφραγίδα της
                                        //αίτησης

ELSE IF (o.timestamp!=0)    //Το συγκεκριμένο αντικείμενο συμπεριλήφθηκε ξανά σε αίτηση
    o.timer = o.timer + (request_timestamp – o.timestamp)    //Αύξηση του χρονομέτρου
    o.timestamp = request_timestamp

//Έλεγχος αν το χρονόμετρο της κοινωνικής διάδοσης του αντικειμένου έχει ξεπεράσει ένα κατώφλι
//χρονικής διάρκειας t.

//Η κοινωνική διάδοση ολοκληρώθηκε - καμία επιπλέον αντιγραφή - αρχικοποίηση μεταβλητών
IF (timer > t)
    o.timer = 0
    o.timestamp = 0

//Η κοινωνική διάδοση συνεχίζεται και ο χρήστης της αίτησης είναι authority – ΚΑΘΟΛΙΚΗ ΑΝΤΙΓΡΑΦΗ
ELSE IF (o.timer < t && user.authorityScore > authority_threshold)
    Copy object O to surrogate that servers user U time zone
    //Καθολική αντιγραφή αντικειμένου στους υποκατάστατους που εξυπηρετούν τις
    //ζώνες ώρας των ακολούθων του χρήστη της αίτησης.
    For each user y that follows user U
        Find surrogate server S that serves y's time zone
        Copy object O to S

//Η κοινωνική διάδοση συνεχίζεται και ο χρήστης της αίτησης δεν είναι authority – ΤΟΠΙΚΗ ΑΝΤΙΓΡΑΦΗ
ELSE IF (o.timer < t)
    Copy object O to surrogate that serves user U time zone
    //Τοπική αντιγραφή αντικειμένου σύμφωνα με μία από τις τρεις υπό-πολιτικές στους
    //υποκατάστατους //που εξυπηρετούν τις ζώνες ώρας των φίλων του χρήστη με αμοιβαία σχέση
    Copy object O to surrogates S that sub-policy A OR sub-policy B
    OR sub-policy C decides
```

Υπό-πολιτική A για τοπική αντιγραφή

```
//Εύρεση X κοντινότερων ζωνών ώρας  
//και γίνεται αντιγραφή του αντικειμένου στους εξυπηρετητές που εξυπηρετούν τις αιτήσεις  
//αυτών των ζωνών ώρας  
Find the X time zones where (user U has mutual friends && their distance is the closest  
to user's U time zone)  
For each time zone t that belongs to X  
    Find surrogate server S that serves time zone t  
    Copy object O to S
```

Η συγκεκριμένη υπό-πολιτική στηρίζεται στο συμπέρασμα ότι οι χρήστες επηρεάζονται περισσότερο από κοντινούς τους φίλους αλλά και φίλους με τους οποίους η σχέση τους στο κοινωνικό δίκτυο είναι αμοιβαία [1]. Η υπό-πολιτική αυτή, αποτελεί και την πιο απλή υπό-πολιτική για την τοπική αντιγραφή.

Υπό-πολιτική B για τοπική αντιγραφή

```
//Εύρεση X κοντινότερων ζωνών ώρας στις οποίες ο χρήστης έχει αμοιβαίους φίλους  
//και από αυτές, εύρεση των L ζωνών ώρας με το ψηλότερο lobby-index.  
//Ακολούθως γίνεται αντιγραφή του αντικειμένου στους εξυπηρετητές που εξυπηρετούν τις αιτήσεις  
//αυτών των ζωνών ώρας  
Find the X time zones where (user U has mutual friends && their distance is the closest  
to user's U time zone)  
Find the  $L \subseteq X$  time zones that (belong to X && have the highest lobby-index score)  
For each time zone t that belongs to L  
    Find surrogate server S that serves time zone t  
    Copy object O to S
```

Το πρώτο μέρος της B υπό-πολιτικής υλοποιεί ουσιαστικά την A υπό-πολιτική. Η συγκεκριμένη, στηρίζεται στα ίδια συμπεράσματα όπως και η προηγούμενη αλλά σε αυτή την υπό-πολιτική εισάγουμε και την έννοια του lobby-index. Βρίσκοντας τις κοντινές ζώνες ώρες στις οποίες ο χρήστης έχει ψηλό lobby-index σημαίνει ότι οι φίλοι του χρήστη σε εκείνες τις ζώνες ώρας έχουν πολλούς φίλους και είναι πιθανόν το αντικείμενο να ζητηθεί περισσότερες φορές.

Υπό-πολιτική Γ για τοπική αντιγραφή

```
//Εύρεση X κοντινότερων ζωνών ώρας στις οποίες ο χρήστης έχει αμοιβαίους φίλους  
//και από αυτές, εύρεση των B ζωνών ώρας με το ψηλότερο betweenness centrality.  
//Ακολουθως γίνεται αντιγραφή του αντικειμένου στους εξυπηρετητές που εξυπηρετούν τις αιτήσεις  
//αυτών των ζωνών ώρας  
Find the X time zones where (user U has mutual friends && their distance is the closest  
to user's U time zone)  
Find  $Y \subseteq X$  time zones that (belong to X && have the highest betweenness centrality  
score)  
For each time zone t that belongs to B  
    Find surrogate server S that serves time zone t  
    Copy object O to S
```

Η Γ υπό-πολιτική λειτουργεί ακριβώς όπως και η Β με τη διαφορά ότι σε αυτή τη περίπτωση χρησιμοποιούμε την «betweenness centrality» μετρική για τη συγκεκριμενοποίηση των ζωνών ώρας στις οποίες θα αντιγραφεί το αντικείμενο. Και πάλι, στηρίζουμε την υπό-πολιτική αυτή στο συμπέρασμα ότι οι χρήστες επηρεάζονται περισσότερο από κοντινούς τους φίλους αλλά και φίλους με τους οποίους η σχέση τους στο κοινωνικό δίκτυο είναι αμοιβαία [1]. Εισάγουμε όμως το betweenness centrality γιατί οι χρήστες με υψηλό σκορ σε αυτή τη μετρική για συγκεκριμένη ζώνη ώρας, θεωρούνται κεντρικοί και άρα η αντιγραφή του αντικειμένου σε αυτή τη ζώνη ώρας θα είναι μεγαλύτερης σημασίας. Πιθανών οι φίλοι του χρήστη που βρίσκονται σε ζώνη ώρας στην οποία ο χρήστης της αίτησης είναι κεντρικός, να ζητήσουν περισσότερες φορές το αντικείμενο, από αυτούς για τους οποίους ο χρήστης δεν είναι κεντρικός (δηλαδή χαμηλό betweenness centrality).

Αλγόριθμος για κάθε έλευση καινούριου αντικειμένου O στον υποκατάστατο εξυπηρετητή S:

```
//Το αντικείμενο χωράει στην cache του εξυπηρετητή
IF (o.size + current_cache_size <= total_cache_size)
    Put object O to surrogate S's cache

//Το αντικείμενο δε χωράει στην cache του εξυπηρετητή
ELSE IF ( object_size + current_cache_size > total_cache_size )

    //Όσο το αντικείμενο δε χωράει στην cache του εξυπηρετητή
    WHILE (o.size + current_cache_size > total_cache_size )

        //Εύρεση αντικειμένων των οποίων το χρονόμετρο τους έχει ξεπεράσει
        //ένα όριο χρονικής διάρκειας η οποία κρατά τα αντικείμενα στη cache, και
        //τοποθέτησή τους σε μία συνδεδεμένη λίστα
        FOR each object q in cache
            IF ( (current_timestamp - q.timestamp)+q.timer >time_threshold)
                put object q in CandidateList

        //Εάν βρέθηκαν αντικείμενα με χρονόμετρο μεγαλύτερο από το threshold και η λίστα δεν
        //περιέχει όλα τα αντικείμενα που υπάρχουν στη cache (δεν είναι όλα τα αντικείμενα παλιά),
        //διαγράφεται αυτό με το μεγαλύτερο χρονόμετρο (δηλ. αυτό με τη διάρκεια κοιν. διάδοσης)
        IF (CandidateList.size>0 && CandidateList.size != total_cache_size)
            find object q that q.timestamp is maximum and delete it

        //Εάν κανένα από τα αντικείμενα δεν είναι παλιά ή εάν όλα τα αντικείμενα είναι παλιά
        //αφήνουμε την αρχική πολιτική αντικατάστασης αντικειμένων η οποία είναι η LRU
        ELSE IF (CandidateList.size==0 ||
CandidateList.size==total_cache_size)
            Use LRU (Least Recently Used) policy to delete any object

//Τελικά τοποθετούμε το αντικείμενο O στην κρυφή μνήμη του υποκατάστατου εξυπηρετητή
Put object O to surrogate S's cache
```

Το πιο πάνω κομμάτι του αλγορίθμου, καθορίζει τη συμπεριφορά των εξυπηρετητών με την έλευση νέων αντικειμένων στη κρυφή τους μνήμη, για εξυπηρέτηση αιτήσεων. Η απόφαση που πρέπει να παρθεί σε αυτό το σημείο είναι το ποιο αντικείμενο να αφαιρεθεί από την κρυφή μνήμη του εξυπηρετητή έτσι ώστε να μπει το καινούριο, στην περίπτωση όπου αυτό δεν χωράει απευθείας.

Ορίζουμε το time_threshold ως η παράμετρος η οποία θα ορίζει την λήξη του χρόνου διαμονής ενός αντικειμένου στην κρυφή μνήμη. Βρίσκουμε λοιπόν τα παλαιότερα αντικείμενα και τα διαγράφουμε. Στην περίπτωση που δεν υπάρχουν τέτοια

αντικείμενα, διαγράφονται αυτά με την μεγαλύτερη κοινωνική διάδοση, δηλαδή με την μεγαλύτερη χρονοσφραγίδα. Τέλος, αν ούτε και αυτό είναι εφικτό, τότε χρησιμοποιείται η προεπιλεγμένη πολιτική, η LRU (Least Recently Used). Η πολιτική αυτή αφαιρεί από την κρυφή μνήμη του εξυπηρετητή, το λιγότερο πρόσφατα χρησιμοποιημένο αντικείμενο.

5.2.6 Παράμετροι Δυναμικής Πολιτικής

Κατά τη διάρκεια της δυναμικής πολιτικής, απαιτείται να αποφασιστεί το πού και το πώς θα αντιγραφεί το αντικείμενο αλλά και η επιλογή των αντικειμένων που τυχόν να πρέπει να εγκαταλείψουν την κρυφή μνήμη ενός υποκατάστατου εξυπηρετητή, με την έλευση καινούριου αντικειμένου. Οι αποφάσεις αυτές παίρνονται με βάση κάποια κριτήρια τα οποία χαρακτηρίζονται ως παράμετροι. Η ύπαρξη παραμέτρων σε έναν αλγόριθμο που δίδεται προς υλοποίηση, αφήνουν περιθώριο βελτίωσης των αποτελεσμάτων αυτού του αλγορίθμου κατά την πειραματική του αξιολόγηση. Αλλαγές σε κάθε παράμετρο ξεχωριστά αλλά και με το συνδυασμό αυτών ίσως να δώσουν τα επιθυμητά αποτελέσματα ή έστω να τα βελτιώσουν.

Πιο κάτω αναγράφονται οι παράμετροι οι οποίες θα πρέπει να δίδονται στη δυναμική πολιτική πριν την εκκίνηση της προσομοίωσης.

- **Χρήστης authority:** κάθε χρήστης έχει ένα σκορ για το «authority» χαρακτηριστικό του, μετά την εκτέλεση του αλγορίθμου HITS. Το ποιος χρήστης θα θεωρείται «καλό» authority και κατ' επέκταση δημοφιλές χρήστης στον κοινωνικό γράφο του Twitter, εξαρτάται από το όριο της τιμής του χαρακτηριστικού αυτού το οποίο εμείς θα θέτουμε προτού ξεκινήσει η προσομοίωση. Επίσης, μπορούμε να θεωρούμε ως χρήστες «authority» τους K χρήστες με την ψηλότερη τιμή. Για παράδειγμα, μπορούμε να θέσουμε ότι χρήστης authority θα θεωρείται κάποιος όπου η τιμή του χαρακτηριστικού «authority» να είναι μεγαλύτερη από 1000 ή τους 1000 χρήστες με την ψηλότερη τιμή στο χαρακτηριστικό αυτό.
- **Κατώφλι t για έλεγχο διάρκειας της κοινωνικής διάδοσης:** στο πρώτο μέρος της πολιτικής, παίρνονται αποφάσεις για το αντικείμενα των οποίων η κοινωνική τους

διάδοση δεν έχει ξεπεράσει το χρονικό διάστημα το οποίο ορίζεται με το κατώφλι t . Το κατώφλι αυτό αναπαριστά χρονικό διάστημα μεταξύ μίας ώρας και 24 ωρών πολλαπλασιασμένο όμως με το 3600 για την αναπαράσταση του σε δευτερόλεπτα.

- **X πιο κοντινές ζώνες ώρας για τοπική αντιγραφή:** κάθε χρήστης έχει διαφόρων ειδών σχέσεις με άλλους χρήστες στο κοινωνικό δίκτυο, σε διάφορες ζώνες ώρας. Αν η αντιγραφή του αντικειμένου μίας αίτησης θα γίνει τοπικά – με τη χρήση με της οποιαδήποτε υπό-πολιτικής που έχουμε ορίσει – θα πρέπει να γίνει σε κοντινές ζώνες ώρας στις οποίες ο χρήστης έχει φίλους με αμοιβαία σχέση. Ο αριθμός των κοντινότερων ζωνών ώρας είναι μία παράμετρος η οποία μπορεί να χαρακτηρίζεται είτε με σταθερό ακέραιο αριθμό, είτε με ένα σταθερό ποσοστό του πλήθους των ζωνών ώρας στις οποίες ο χρήστης έχει αμοιβαίους φίλους. Για παράδειγμα, μπορεί να ορίσουμε ότι θα επιλέγονται οι 3 πλησιέστερες ζώνες ώρας ή ότι θα επιλέγεται αριθμός ζωνών ώρας ίσος με το 15% του πλήθους των ζωνών ώρας στις οποίες ο χρήστης έχει αμοιβαίους φίλους. Είναι διαφορετικό να επιλεγθούν οι 3 πλησιέστερες περιοχές για κάποιον που έχει φίλους σε 50 ζώνες ώρας σε σχέση με κάποιον που έχει φίλους σε 5 ζώνες ώρας. Οπότε ίσως το ποσοστό να είναι καλύτερη επιλογή για αυτή τη παράμετρο.
- **L ζώνες ώρας με το υψηλότερο lobby-index:** Η συγκεκριμένη παράμετρος χρησιμοποιείται στην υπό-πολιτική Β για την τοπική αντιγραφή. Συγκεκριμένα, από τις X πιο κοντινές ζώνες ώρας (όπως ορίστηκαν πιο πάνω), επιλέγονται οι L ζώνες ώρας με το υψηλότερο lobby-index για το χρήστη ο οποίος έκανε τη συγκεκριμένη αίτηση. Η παράμετρος αυτή λοιπόν, αναφέρεται στο πλήθος των ζωνών ώρας που θα επιλεγθούν μετά την επιλογή των X κοντινότερων, με βάση το lobby-index του χρήστη σε κάθε μία από τις X ζώνες ώρας. Όπως και η προηγούμενη, έτσι και αυτή η παράμετρος, μπορεί να παρουσιάζεται ως σταθερός αριθμός για κάθε περίπτωση ή ως ποσοστό πάνω στις X κοντινότερες ζώνες ώρας.
- **Y ζώνες ώρας με το υψηλότερο betweenness centrality:** Αυτή η παράμετρος χρησιμοποιείται στη περίπτωση της επιλογής της Γ υπό-πολιτικής για τοπική αντιγραφή. Από τις X πιο κοντινές ζώνες ώρας (όπως ορίστηκαν πιο πάνω), επιλέγονται οι Y ζώνες ώρας με το υψηλότερο betweenness centrality για το χρήστη ο οποίος έκανε τη συγκεκριμένη αίτηση. Όπως και οι άλλες δύο παράμετροι για την τοπική αντιγραφή στη δυναμική πολιτική, έτσι και αυτή μπορεί να δίδεται ως

σταθερός ακέραιος αριθμός ή ως ποσοστό των X ζωνών ώρας που επιλέχθηκαν ως οι κοντινότερες στο χρήστη.

- **Όριο χρονομέτρου για διαγραφή από τη cache – time_threshold:** Στο δεύτερο μέρος της δυναμικής πολιτικής δίδεται αλγόριθμος για την επιλογή των αντικειμένων που πρόκειται να διαγραφούν από την κρυφή μνήμη ενός υποκατάστατου εξυπηρετητή, για την αποθήκευση ενός καινούριου αντικειμένου. Για την επιλογή του αντικειμένου που θα αφαιρεθεί, συγκρίνουμε το χρονόμετρο του αντικειμένου με ένα κατώφλι το οποίο αποτελεί τη συγκεκριμένη παράμετρο. Η τιμή αυτή καθορίζει κατά πόσον το αντικείμενο είναι παλιό ή όχι. Η παράμετρος αυτή μπορεί να κυμαίνεται μεταξύ της μίας ώρας και των 24 ωρών, πολλαπλασιασμένο όμως με το 3600 για την αναπαράσταση του σε δευτερόλεπτα, όπως δηλαδή υπολογίζεται και το κατώφλι t.

Πιο συγκεκριμένα, κάθε αντικείμενο διαθέτει μία χρονοσφραγίδα που δείχνει την στιγμή που τοποθετήθηκε στον υποκατάστατο εξυπηρετητή. Αφαιρώντας τη συγκεκριμένη χρονοσφραγίδα από τη χρονοσφραγίδα της δεδομένης στιγμής κατά τη προσομοίωση και προσθέτοντας το χρονόμετρο του αντικειμένου, έχουμε το χρονικό διάστημα της κοινωνικής διάδοσης του αντικειμένου μέχρι τη συγκεκριμένη χρονική στιγμή. Αν η τιμή αυτή ξεπεράσει τη συγκεκριμένη παράμετρο τότε το αντικείμενο θεωρείται «παλιό» και ίσως να αφαιρεθεί, αφού η κοινωνική του διάδοση τείνει να ολοκληρωθεί.

5.2.7 Προτεινόμενες Βελτιώσεις Αλγορίθμου

Πιο κάτω παρουσιάζονται κάποιες βελτιώσεις του αλγορίθμου που μπορούν να υλοποιηθούν σε μετέπειτα στάδιο και να συγκριθούν με τα αποτελέσματα της αρχικής μορφής της δυναμικής πολιτικής. Οι πιο κάτω βελτιώσεις αποτελούνται από σύνθετες λειτουργίες οι οποίες δεν κρίθηκαν αναγκαίες για την αρχική μορφή της δυναμικής πολιτικής.

1. Τοποθέτηση των αντικειμένων στους υποκατάστατους εξυπηρετητές σε ώρα ανάλογη με την ζώνη ώρας για την οποία προορίζεται το αντικείμενο και όχι αυτόματη αντιγραφή. Λόγω της διαφοράς ώρας μεταξύ των ζωνών ώρας, η αυτόματη αντιγραφή των αντικειμένων σε υποκατάστατους εξυπηρετητές

διαφορετικών ζωνών ώρας, πιθανών να μην δίνει τα βέλτιστα αποτελέσματα. Ο λόγος έγκειται στο γεγονός ότι λόγω της διαφοράς ώρας, κάποιο αντικείμενο μπορεί να χρειαστεί από τους χρήστες σε διαφορετική στιγμή σε διαφορετικές ζώνες ώρας. Έστω ότι κάποιος χρήστης δημοσιεύει ένα βίντεο στη ζώνη ώρας του Λονδίνου στις 10 το πρωί και έστω ότι έχει αποφασισθεί ότι το αντικείμενο πρέπει να αντιγραφεί σε υποκατάστατους εξυπηρετητές στη Νέα Υόρκη λόγω των φίλων του χρήστη. Όταν όμως στο Λονδίνο είναι 10 το πρωί, στην Νέα Υόρκη είναι 5 το πρωί, υπάρχει δηλαδή διαφορά ώρας 5 ωρών. Πόσο πιθανόν είναι κάποιος φίλος του χρήστη να ζητήσει το βίντεο στις 5 το πρωί στη Νέα Υόρκη;

Αντιγράφοντας λοιπόν το αντικείμενο την ίδια στιγμή και στις δύο περιοχές, υπάρχει κίνδυνος το αντικείμενο να αφαιρεθεί από την κρυφή μνήμη του υποκατάστατου εξυπηρετητή για παράδειγμα στη Νέα Υόρκη, προτού αιτηθεί από τους χρήστες, λόγω της έλευσης νέων αντικειμένων στον εξυπηρετητή. Αυτό δύναται να αυξήσει τόσο το κόστος αναπαραγωγής (αφού αν το αντικείμενο ζητηθεί μετά τη διαγραφή του θα πρέπει να αντιγραφεί ξανά στον υποκατάστατο εξυπηρετητή), όσο και το χρόνο απόκρισης του δικτύου παράδοσης περιεχομένου. Αν όμως το αντικείμενο αντιγραφεί σε υποκατάστατο εξυπηρετητή σύμφωνα με τη ζώνη ώρας του και τη διαφορά ώρας μεταξύ της ζώνης ώρας που έλαβε την αίτηση, τότε ίσως το πιο πάνω πρόβλημα να περιοριστεί αφού πλέον τα αντικείμενα θα τοποθετούνται πιο κοντά στη χρονική στιγμή στην οποία ενδέχεται οι χρήστες να τα ζητήσουν.

2. Εισαγωγή της έννοιας του βίντεο με ψηλή δημοτικότητα. Το κατά πόσον ένα βίντεο θεωρείται δημοφιλές ή όχι καθορίζεται από τον αριθμό θεάσεων του στο YouTube. Αν λοιπόν ένα βίντεο έχει αριθμό θεάσεων πάνω από ένα κατώφλι τότε θεωρείται βίντεο με ψηλή δημοτικότητα και ίσως να είναι καλύτερα να τυχαίνει καθολικής αντιγραφής. Έχουμε ήδη αναφέρει ότι οι κοινωνικές διαδόσεις επηρεάζονται από τον αριθμό θεάσεων των βίντεο στο YouTube [1]. Ίσως λοιπόν μία τέτοια λειτουργία να βελτιώσει τα αποτελέσματα της δυναμικής πολιτικής.

Κεφάλαιο 6

Αξιολόγηση Στατικών Πολιτικών Τοποθέτησης Περιεχομένου

6.1 Πειραματική Αξιολόγηση	53
6.2 Αποτελέσματα	62
6.3 Συμπεράσματα	70

6.1 Πειραματική Αξιολόγηση

Για τους σκοπούς της πειραματικής αξιολόγησης των στατικών πολιτικών, χρησιμοποιήσαμε τον προσομοιωτή δικτύου παράδοσης περιεχομένου CDNsim. Για να χρησιμοποιήσουμε όμως τον προσομοιωτή αυτό, χρειάστηκε να έχουμε συγκεκριμένη τοπολογία διαδικτύου και επίσης να γίνουν κάποιες παραδοχές ως προς τα δεδομένα, έτσι ώστε να έχουμε αξιόπιστα αποτελέσματα από τον προσομοιωτή.

6.1.1 Παραδοχές για χρήση προσομοιωτή CDNsim

- **Αριθμός κόμβων στην τοπολογία και αριθμός επεξεργαστών:**

Ξεκινήσαμε να τρέχουμε πειράματα στο εργαλείο με μία αρκετά μεγάλη τοπολογία διαδικτύου. Ακολουθώντας, λόγω του ότι το εργαλείο δεν μπορούσε να ανταποκριθεί σε μία τόσο μεγάλη τοπολογία, αναγκαστήκαμε να μειώσουμε τον αριθμό των κόμβων της τοπολογίας στο 10% και χρησιμοποιήσαμε τελικά μία τοπολογία διαδικτύου με 3500 κόμβους.

Περίπου στο ίδιο ποσοστό μειώσαμε και τους υποκατάστατους εξυπηρετητές οι οποίοι από 4879 μειώθηκαν σε 423. Επίσης υπάρχει ένας εξυπηρετητής προέλευσης.

Έτσι η τοπολογία του διαδικτύου που χρησιμοποιούμε, αποτελείται από 3500 κόμβους εκ των οποίων οι 423 είναι υποκατάστατοι εξυπηρετητές και ένας από αυτούς είναι και εξυπηρετητής προέλευσης.

- **Αριθμός Αιτήσεων:**

Για τους ίδιους λόγους όπως και πιο πάνω, έπρεπε να μειώσουμε των αριθμό αιτήσεων με τις οποίες εκτελούμε τα πειράματα στον προσομοιωτή. Ενώ ξεκινήσαμε με περίπου 1 εκατομμύριο αιτήσεις, χρειάστηκε να μειώσουμε τις αιτήσεις σε 302630.

Το πρόβλημα εδώ είναι ότι όσες περισσότερες αιτήσεις δίνουμε στον προσομοιωτή, η ανάγκη του για μνήμη κατά τη διάρκεια εκτέλεσης των πειραμάτων αυξάνεται εκθετικά [2] και στο τέλος τερματίζει την προσομοίωση ανεπιτυχώς. Για αυτό το λόγο μειώσαμε τον όγκο των αιτήσεων, οι οποίες αποτελούν το 5% των αιτήσεων που μας έδωσε η Γεννήτρια Αιτήσεων[1], έτσι το σύνολο των αιτήσεων που χρησιμοποιείται, αν και είναι μικρό, εξακολουθεί να είναι ρεαλιστικό.

- **Μέγεθος βίντεο και Εύρος Ζώνης (Bandwidth):**

Σε προηγούμενο κεφάλαιο έχει γίνει ανάλυση της μεθοδολογίας υπολογισμού του μεγέθους (size) των βίντεο χρησιμοποιώντας τη διάρκεια των βίντεο. Αφού λοιπόν υπολογίσθηκαν τα μεγέθη των βίντεο τα οποία αποτελούν και το περιεχόμενο που διακινείται στο δίκτυο παράδοσης περιεχομένου, αντιληφθήκαμε ότι κάποια από αυτά ήταν πάρα πολύ μεγάλα.

Πιο συγκεκριμένα, τα αντικείμενα δίνονται στον προσομοιωτή μαζί με το μέγεθός τους το οποίο πρέπει να αναπαρίσταται σε bytes. Το εργαλείο όμως δεν μπορεί να διαχειριστεί αντικείμενα τα οποία είναι μεγαλύτερα από 1 δισεκατομμύριο bytes, δηλαδή μεγαλύτερα από 1 GB. Έτσι έπρεπε να κάνουμε κλιμακωτή απομείωση τα μεγέθη όλων των αντικειμένων.

Για να το πετύχουμε αυτό, διαιρέσαμε τα μεγέθη των βίντεο με το 1024 για να ελαττωθούν κατά μία τάξη και ως εκ τούτου να μειωθεί το μέγεθος τους. Για να έχουμε συνοχή όμως στα δεδομένα μας, έπρεπε να εφαρμόσουμε την ίδια διαίρεση στην κρυφή μνήμη των εξυπηρετητών. Επιπλέον, έπρεπε να μειώσουμε το εύρος ζώνης επίσης κατά μία τάξη, όπως έχει γίνει και με τα αντικείμενα. Αφού λοιπόν με τα αρχικά μεγέθη των αντικειμένων, οι σύνδεσμοι (links) των κόμβων της τοπολογίας λειτουργούσαν με 1000Mbits/sec εύρος ζώνης, με τη διαίρεση μειώνεται σε 1Mbit/sec για να υπάρξει

συνοχή και το δίκτυο να συμπεριφέρεται ορθά με τα νέα μεγέθη των αντικειμένων. Αν δεν μειώναμε το εύρος ζώνης, θα ήταν πολύ εύκολο στο δίκτυο να διακινήσει το περιεχόμενο αφού πλέον τα μεγέθη των βίντεο έχουν ελαττωθεί, έτσι δεν θα μπορούσαμε να ανιχνεύσουμε τυχόν προβλήματα και συμφόρηση στο δίκτυο.

- **Ομαδοποίηση Χρηστών**

Για να λειτουργήσει το εργαλείο, χρειαζόταν οι ταυτότητες των χρηστών που αιτούνται περιεχόμενο από το διαδίκτυο να είναι συνεχή, δηλαδή να ξεκινούν από το 0 και να φθάνουν μέχρι το πλήθος των χρηστών. Για παράδειγμα αν το πλήθος των χρηστών ήταν 50, δεν επιτρέπεται να υπάρχει χρήστης με ταυτότητα 51.

Λόγω του ότι χρησιμοποιούμε την Γεννήτρια Αιτήσεων για να μας δώσει τις αιτήσεις στις οποίες υπάρχουν διαφορετικοί χρήστες κάθε φορά, αποφασίσαμε να ομαδοποιήσουμε τους χρήστες ανά time zone, ούτως ώστε οι ταυτότητες των χρηστών να είναι σταθερές και να μην αλλάζουν ανάλογα με τις αιτήσεις της Γεννήτριας Αιτήσεων. Για παράδειγμα αν στο time zone του Λονδίνου υπάρχουν οι χρήστες με ταυτότητες 25633 και 102, τότε αν ο κωδικός που έχουμε δώσει στο time zone του Λονδίνου είναι το 5, αυτός ο κωδικός θα δοθεί και σαν ταυτότητα στους συγκεκριμένους χρήστες. Συγκεκριμένα έχουμε 162 κωδικούς για κάθε time zone – αν και το twitter ορίζει 142 time zones κάποια από αυτά έπρεπε να διασπαστούν και να κατανεμηθούν σε όλους τους εξυπηρετητές γιατί αποτελούνται από γενικές τοποθεσίες, για παράδειγμα οι χρήστες που δεν έχουν ορίσει κάποια περιοχή στον λογαριασμό τους στο twitter.

Η συγκεκριμένη ομαδοποίηση των χρηστών μας βοήθησε και όταν τοποθετήσαμε τις ομάδες αυτές σε συγκεκριμένους κόμβους της τοπολογίας, έτσι ώστε η θέση τους στην τοπολογία να αντανακλά την γεωγραφική τους θέση αλλά και να βρίσκονται κοντά στους υποκατάστατους εξυπηρετητές από τους οποίους εμείς ορίσαμε ότι θα εξυπηρετούνται – όπως επεξηγείται στο Κεφάλαιο 3.

- **Αύξηση του χρονικού παραθύρου μεταξύ των αιτήσεων**

Λόγω του ότι η Γεννήτρια Αιτήσεων παράγει αιτήσεις με πραγματικές χρονοσφραγίδες (timestamp), δηλαδή οι αιτήσεις είναι σε πραγματικό χρόνο, παρατηρούσαμε μεγάλο

αριθμό αποτυχημένων αιτήσεων κατά τη διάρκεια της πειραματικής αξιολόγησης των πολιτικών που έχουν κατασκευαστεί. Ο προσομοιωτής δεν μπορούσε να προσομοιώσει στο βέλτιστο την συμπεριφορά του δικτύου λόγω του μεγάλου όγκου αιτήσεων που δεχόταν σε πολύ μικρά χρονικά διαστήματα και λόγω του χαμηλού bandwidth. Αποτέλεσμα αυτής της συμπεριφοράς είναι ο μεγάλος αριθμός αποτυχημένων μη-εξυπηρετούμενων αιτήσεων.

Για να βελτιωθεί λοιπόν η απόδοση του προσομοιωτή, αποφασίσαμε να αυξήσουμε το χρονικό διάστημα που μεσολαβούσε μεταξύ των αιτήσεων. Για το σκοπό αυτό έπρεπε αρχικά να πολλαπλασιαστούν όλες τις χρονοσφραγίδες του συνόλου των αιτήσεων με έναν σταθερό αριθμό έτσι ώστε να μεγαλώσει γενικά το εύρος του χρόνου. Δοκιμάζοντας διάφορες τιμές για την σταθερή τιμή με την οποία θα πολλαπλασιάζονταν οι χρονοσφραγίδες, επιλέχθηκε ο αριθμός 5000 γιατί αυτή η τιμή οδηγούσε σε μία καλή κατανομή των αιτήσεων και βελτίωνε τα αποτελέσματα του προσομοιωτή σε μεγάλο βαθμό.

Μετά από αυτό το βήμα, δηλαδή τον πολλαπλασιασμό με την σταθερά τιμή, έπρεπε να διαμοιραστούν οι αιτήσεις της συγκεκριμένης χρονοσφραγίδας στην χρονική περίοδο που δημιουργείται μετά τον πολλαπλασιασμό.

Για χάριν παραδείγματος υποθέστε ότι υπάρχουν οι χρονοσφραγίδες 1 και 2 στο σύνολο των αιτήσεων που χρησιμοποιούμε στα πειράματα. Μετά τον πολλαπλασιασμό, η χρονοσφραγίδα 1 θα αντικατασταθεί με την χρονοσφραγίδα 5000, ενώ η 2 θα αντικατασταθεί με την 10000. Μεταξύ των δύο χρονοσφραγίδων λοιπόν, έχουν δημιουργηθεί 5000 καινούριες χρονικές στιγμές. Μέσα σε αυτές τις χρονικές στιγμές, διαμοιράζονται οι αιτήσεις που υπήρχαν στην χρονοσφραγίδα 1 αρχικά. Αν προηγουμένως υπήρχαν 10 αιτήσεις στην χρονοσφραγίδα «1», τότε αυτές οι αιτήσεις θα διαμοιραστούν πλέον στην περίοδο χρόνου [5000..9999]. Ουσιαστικά θα μπει μία αίτηση στην χρονοσφραγίδα 5000, η επόμενη στη χρονοσφραγίδα 5500 κοκ. Οι αιτήσεις που υπήρχαν στην χρονοσφραγίδα 2, θα τοποθετηθούν στο εύρος χρόνου [10000..14999].

Αυξάνοντας με τέτοιο τρόπο το χρονικό παράθυρο μεταξύ των αιτήσεων, πετυχαίνουμε καλύτερα και πιο αξιόπιστα αποτελέσματα από τον προσομοιωτή CDNsim, χωρίς να επηρεάζεται η πραγματική κατανομή των αιτήσεων όπως αυτές δίνονται από την Γεννήτρια Αιτήσεων. Τα χαρακτηριστικά των αιτήσεων ως προς την κατανομή τους στη διάρκεια της ημέρας και της εβδομάδας παραμένουν αναλλοίωτα, αφού οι ίδιες αιτήσεις γίνονται σε συγκεκριμένο χρονικό διάστημα χωρίς να μπερδεύονται μεταξύ τους. Ο κώδικας για την αύξηση του χρονικού παραθύρου μεταξύ των αιτήσεων βρίσκεται στο Παράρτημα Β.

6.1.2 Επεξήγηση εξόδου προσομοιωτή

Όπως έχει αναφερθεί σε προηγούμενο κεφάλαιο, ο προσομοιωτής δικτύων παράδοσης περιχομένου CDNsim, μετά την προσομοίωση, κατασκευάζει μια στατιστική αναφορά η οποία αποτελεί και την κύρια έξοδό του. Η αναφορά αυτή αποτελεί και το μέτρο αξιολόγησης για τις πολιτικές που έχουν κατασκευαστεί και δίνει πληροφορίες για την συμπεριφορά των χρηστών, των υποκατάστατων εξυπηρετητών και των εξυπηρετητών προέλευσης κατά τη διάρκεια της προσομοίωσης. Με βάση αυτή την στατιστική αναφορά μπορούμε να αξιολογούμε, να κρίνουμε και να συγκρίνουμε στατικές και δυναμικές πολιτικές που είτε έχουν κατασκευαστεί, είτε πρόκειται να κατασκευαστούν μελλοντικά.

Πιο κάτω παρουσιάζεται μία επεξήγηση των αποτελεσμάτων που μας δίνει ο προσομοιωτής έτσι όπως περιγράφονται στο [2]. Για την επεξήγηση χρησιμοποιείται ένα παράδειγμα εξόδου του προσομοιωτή μετά την εκτέλεση της ακραίας πολιτικής στην οποία υπάρχει καθολική τοποθέτηση όλων των αντικειμένων στους υποκατάστατους εξυπηρετητές.

Όπως φαίνεται στο Σχήμα 6.1, η στατιστική αναφορά η οποία αποτελεί και την έξοδο του προσομοιωτή χωρίζεται σε 4 μέρη.

```
Bottle: full_alloc_5000
CLIENT SIDE MEASUREMENTS
Completed requests: 302456
Aborted requests: 174
Retries: 744
Not enough outgoing connections: 0
Mean response time: 13.3991089974
SURROGATE SIDE MEASUREMENTS
Hit ratio percentage: 100.0
Byte hit ratio percentage: 100.0
Not enough outgoing connections: 0
Not enough incoming connections: 0
Failed pulls: 0
ORIGIN SIDE MEASUREMENTS
Not enough incoming connections: 0
SURROGATE SERVERS UTILITIES
Active surrogate servers: 78
Mean surrogate servers utility 0.999999998621
```

Σχήμα 6.1: Παράδειγμα στατιστικής αναφοράς του προσομοιωτή δικτύου παράδοσης περιεχομένου CDNSim [2], η οποία δίδεται ως έξοδος μετά την προσομοίωση

1^ο Μέρος: CLIENT SIDE MEASUREMENTS

Αυτό το μέρος της στατιστικής αναφοράς, μας δίνει τα αποτελέσματα για την λειτουργία και την εξυπηρέτηση των χρηστών και συγκεκριμένα των αιτήσεών τους.

- **Completed requests:** η μετρική αυτή αναφέρεται στον αριθμό των ολοκληρωμένων αιτήσεων που έχουν γίνει στο δίκτυο, αιτήσεις δηλαδή που έχουν εξυπηρετηθεί επιτυχώς και επομένως το περιεχόμενο έχει παραδοθεί στον χρήστη.
- **Aborted requests:** εδώ αναγράφεται ο αριθμός των αιτήσεων οι οποίες δεν έχουν ολοκληρωθεί, δηλαδή ήταν αποτυχημένες ως προς την εξυπηρέτησή τους.
- **Retries:** με τον όρο αυτό εννοούμε τις επαναλήψεις που χρειάστηκαν να γίνουν για κάποιες από τις αιτήσεις. Για παράδειγμα αν κάποια αίτηση δεν εξυπηρετηθεί, μπορεί να γίνει επανάληψη της συγκεκριμένης αίτησης στο δίκτυο για μετέπειτα εξυπηρέτηση. Ο αριθμός των μέγιστων επαναλήψεων που

μπορούν να γίνουν αποτελεί παράμετρο του προσομοιωτή η οποία ορίζεται στην αρχή – προτού ξεκινήσει η εκτέλεση της προσομοίωσης – και στην περίπτωση μας έχει οριστεί στις 10 επαναλήψεις. Επίσης, μπορούμε να καθορίσουμε τον χρόνο ο οποίος θα μεσολαβεί μεταξύ των επαναλήψεων τον οποίο εμείς τον έχουμε καθορίσει στα 5 δευτερόλεπτα.

- **Not enough outgoing connection:** κάθε χρήστης αντιπροσωπεύεται από κάποιον εξυπηρετητή στην τοπολογία του διαδικτύου. Κάθε εξυπηρετητής-χρήστης έχει κάποιους συνδέσμους εξόδου από τους οποίους αποστέλλεται η κάθε αίτηση. Η συγκεκριμένη τιμή δείχνει τις φορές όπου οι εξερχόμενοι σύνδεσμοι δεν ήταν αρκετοί για να αποστείλουν όλες τις αιτήσεις στην δεδομένη στιγμή. Για παράδειγμα αν υπάρχουν 5 εξερχόμενοι σύνδεσμοι αλλά σε κάποια συγκεκριμένη χρονοσφραγίδα εμφανισθούν 6 αιτήσεις, τότε κάποια από τις αιτήσεις δεν θα μπορέσει να αποσταλεί και έτσι η μετρική αυτή θα αυξηθεί κατά ένα. Ο αριθμός των εξερχόμενων συνδέσμων στους χρήστες είναι παράμετρος του προσομοιωτή και για την εκτέλεση των πολιτικών μας θέσαμε την παράμετρο αυτή στα 1000.
- **Mean response time:** ο όρος αυτός αναφέρεται στο μέσο χρόνο απόκρισης – ίσως η πιο σημαντική μετρική του πρώτου μέρους της στατιστικής αναφοράς. Αυτή η τιμή μετριέται σε δευτερόλεπτα και μας δείχνει τον μέσο όρο χρόνου στον οποίο εξυπηρετήθηκαν οι αιτήσεις, δηλαδή ο μέσος χρόνος που χρειάστηκε για να φτάσει το κάθε αντικείμενο στον χρήστη που το είχε αιτηθεί. Είναι μία πολύ σημαντική μετρική για την αξιολόγηση των διάφορων πολιτικών προ-ανάκλησης περιεχομένου. Ένας μεγάλος μέσος χρόνος απόκρισης δείχνει προβλήματα στο δίκτυο, όπως είναι η συμφόρηση.

2^ο Μέρος: SURROGATES SIDE MEASUREMENTS

Το δεύτερο μέρος της αναφοράς αφορά την λειτουργία των υποκατάστατων εξυπηρετητών σε σχέση με την εξυπηρέτηση των αιτήσεων που λαμβάνουν κατά τη διάρκεια της προσομοίωσης.

- **Hit ratio percentage:** η μετρική αυτή δείχνει το ποσοστό των αιτήσεων που πέτυχαν «επιτυχία» στην κρυφή μνήμη του υποκατάστατου εξυπηρετητή στον οποίο στάλθηκε η αίτηση εξαρχής. Δηλαδή αν η αίτηση αφορά το αντικείμενο με ταυτότητα 25 και στέλνεται στον Α εξυπηρετητή, αν το αντικείμενο βρίσκεται στην μνήμη του εξυπηρετητή τότε έχουμε «cache hit» και το ποσοστό της μετρικής αυξάνεται. Υψηλά ποσοστά στα αποτελέσματα φανερώνουν υψηλή ποιότητα στην τοποθέτηση και κατανομή του περιεχομένου στους υποκατάστατους εξυπηρετητές.
- **Byte hit ratio percentage:** η μετρική αυτή είναι η ίδια με την προηγούμενη, με τη διαφορά ότι αντί για το ποσοστό αιτήσεων, μετρούμε το ποσοστό των bytes των αντικειμένων που αφορούσαν τις αντίστοιχες αιτήσεις οι οποίες πέτυχαν «επιτυχία» στην κρυφή μνήμη του υποκατάστατου εξυπηρετητή. Η τιμή αυτή μετριέται σε ποσοστό. Οι ψηλές τιμές δείχνουν βελτιστοποιημένη χρήση του χώρου των υποκατάστατων εξυπηρετητών και χαμηλή συμφόρηση δικτύου.
- **Not enough outgoing connections:** οι υποκατάστατοι εξυπηρετητές έχουν εξερχόμενους συνδέσμους για την αποστολή των αιτημάτων σε άλλους εξυπηρετητές έτσι ώστε να ζητήσουν αντικείμενα και περιεχόμενο που δεν έχουν στην μνήμη τους και πιθανών να χρειάζονται. Η μετρική αυτή παρουσιάζει τις φορές όπου ο αριθμός των εξερχόμενων συνδέσμων στους υποκατάστατους εξυπηρετητές δεν ήταν αρκετός για την αποστολή όλων των αιτήσεων της δεδομένης χρονοσφραγίδας. Στα δικά μας σενάρια, ο αριθμός των εξερχόμενων συνδέσμων ο οποίος είναι και παράμετρος του προσομοιωτή, ορίζεται στα 1000.
- **Not enough incoming connections:** στους υποκατάστατους εξυπηρετητές υπάρχουν και εισερχόμενοι σύνδεσμοι οι οποίοι παρουσιάζουν ενδιαφέρον στην απόδοσή τους. Δηλαδή μετριέται το κατά πόσον κάποιος υποκατάστατος εξυπηρετητής κατάφερε να αποδεχθεί όλα τα αιτήματα που του έχουν σταλεί σε κάθε συγκεκριμένη χρονοσφραγίδα. Η μετρική αυτή δεν διαφέρει ιδεολογικά από την προηγούμενη αλλά αφορά τους εισερχόμενους συνδέσμους των

υποκατάστατων εξυπηρετητών. Στα σενάρια των πολιτικών μας ορίζουμε την παράμετρο των εισερχόμενων συνδέσμων στους υποκατάστατους εξυπηρετητές στα 1000.

- **Failed Pulls:** είναι ο αριθμός των αποτυχημένων προσπαθειών ενός υποκατάστατου εξυπηρετητή για έλξη ενός αντικειμένου, είτε από κάποιον άλλον υποκατάστατο εξυπηρετητή είτε από τον εξυπηρετητή προέλευσης.

3^ο Μέρος: ORIGIN SIDE MEASUREMENTS

Το τρίτο μέρος της αναφορά αναφέρεται στην λειτουργία του εξυπηρετητή προέλευσης. Για τα πειράματα σε αυτή τη διπλωματική εργασία, χρησιμοποιείται ένας εξυπηρετητής προέλευσης.

- **Not enough incoming connections:** όπως και οι υπόλοιποι εξυπηρετητές, έτσι και ο εξυπηρετητής προέλευσης έχει έναν αριθμό από εισερχόμενους συνδέσμους – τους οποίους ορίσαμε στα 1000 στις προσομοιώσεις των πολιτικών. Η μετρική αυτή δείχνει τις φορές στις οποίες ο αριθμός αυτός δεν ήταν αρκετός για την έλευση όλων των αιτήσεων προς τον εξυπηρετητή προέλευσης σε συγκεκριμένη χρονοσφραγίδα.

4^ο Μέρος: SURROGATE SERVERS UTILITIES

Το τέταρτο και τελευταίο μέρος της στατιστικής αναφοράς του προσομοιωτή αφορά την απόδοση ως προς την λειτουργία των υποκατάστατων εξυπηρετητών.

- **Active surrogate servers:** ο αριθμός των ενεργών υποκατάστατων εξυπηρετητών κατά τη διάρκεια της προσομοίωσης. Αν κάποιος υποκατάστατος εξυπηρετητής δε χρειάστηκε να λειτουργήσει, δηλαδή ούτε παρέλαβε αίτηση αλλά ούτε και απέστειλε περιεχόμενο, τότε δεν περιλαμβάνεται στο σύνολο των ενεργών υποκατάστατων εξυπηρετητών.
- **Mean surrogate servers utility:** αναφέρεται στην μέση χρησιμότητα των υποκατάστατων εξυπηρετητών και μετριέται με τιμή η οποία κυμαίνεται από το 0 έως το 1. Αυτή η μετρική εκφράζει τη σχέση μεταξύ του αριθμού των bytes

του περιεχομένου που εξυπηρετούνται από τον υποκατάστατο εξυπηρετητή, προς τον αριθμό των bytes που ο υποκατάστατος εξυπηρετητής χρειάστηκε να ζητήσει και να λάβει από κάποιον άλλον εξυπηρετητή για την ικανοποίηση μίας αίτησης. Η συγκεκριμένη μετρική παρέχει μία ένδειξη σχετικά με την απόδοση του δικτύου παράδοσης περιεχομένου.

6.2 Αποτελέσματα

Τα πιο κάτω αποτελέσματα αφορούν προσομοιώσεις με τα εξής χαρακτηριστικά:

Αριθμός κόμβων στην τοπολογία	3500	Αριθμός χρηστών	149
Αριθμός εξυπηρετητών προέλευσης	1	Αριθμός αιτήσεων	302630
Αριθμός υποκατάστατων εξυπ.	423	Εύρος ζώνης	1Mbit/sec

6.2.1 Ακραία Πολιτική 1 - Τοποθέτηση όλου του περιεχομένου σε όλους του υποκατάστατους εξυπηρετητές

Πιο κάτω παραθέτονται τα στατιστικά αποτελέσματα που έδωσε ο προσομοιωτής CDNsim μετά την προσομοίωση της πρώτης ακραίας πολιτικής η οποία αφορά την πλήρη τοποθέτηση του περιεχομένου σε όλους τους υποκατάστατους εξυπηρετητές πριν την έναρξη της προσομοίωσης.

```

Bottle: full_alloc_5000
CLIENT SIDE MEASUREMENTS
Completed requests: 302456
Aborted requests: 174
Retries: 744
Not enough outgoing connections: 0
Mean response time: 13.3991089974
SURROGATE SIDE MEASUREMENTS
Hit ratio percentage: 100.0
Byte hit ratio percentage: 100.0
Not enough outgoing connections: 0
Not enough incoming connections: 0
Failed pulls: 0
ORIGIN SIDE MEASUREMENTS
Not enough incoming connections: 0
SURROGATE SERVERS UTILITIES
Active surrogate servers: 78
Mean surrogate servers utility 0.999999998621

```

Σχήμα 6.2: Στατιστική αναφορά του προσομοιωτή CDNsim σχετικά με την προσομοίωση της ακραίας πολιτικής τοποθέτησης όλου του περιεχομένου σε όλους τους υποκατάστατους εξυπηρετητές

Τα αποτελέσματα (Σχήμα 6.2) μας δείχνουν ότι με την συγκεκριμένη πολιτική επιτυγχάνεται η πλήρης σχεδόν ικανοποίηση των αιτήσεων – μόνο 174 αιτήσεις δεν καταφέρνουν να εξυπηρετηθούν από τους υποκατάστατους εξυπηρετητές (aborted requests) ενώ οι υπόλοιπες 302456 αιτήσεις ικανοποιούνται επιτυχώς (completed requests). Αυτό είναι κάτι που αναμενόταν με την συγκεκριμένη πολιτική αφού σε κάθε υποκατάστατο εξυπηρετητή υπάρχουν όλα τα αντικείμενα τα οποία δύναται να ζητηθούν από τους χρήστες. Όσον αφορά τον αριθμό των αιτήσεων που δεν κατάφεραν να εξυπηρετηθούν, αυτό ίσως να συνέβη λόγω κάποιας καθυστέρησης στο δίκτυο ή λόγω του μειωμένου bandwidth. Επίσης παρατηρούμε ότι κατά τη διάρκεια της προσομοίωσης χρειάστηκε να γίνουν 744 επιπλέον προσπάθειες για κάποια αίτηση (retries) που και αυτό το πιο πιθανόν να οφείλεται στην κατάσταση του δικτύου την δεδομένη στιγμή.

Αυτό που είναι σημαντικό να προσέξουμε σε αυτά τα αποτελέσματα είναι ότι ο μέσος χρόνος απόκρισης (mean response time) είναι μόλις 13.39 δευτερόλεπτα, ο οποίος είναι αρκετά μικρός. Συγκρίνοντας τον χρόνο της συγκεκριμένης πολιτικής με τους χρόνους των υπόλοιπων στατικών πολιτικών που παραθέτονται πιο κάτω, παρατηρούμε ότι αυτή η πολιτική πετυχαίνει τον καλύτερο χρόνο απόκρισης, όπως άλλωστε είχαμε προβλέψει.

Προχωρώντας στα στατιστικά στοιχεία που αφορούν τους υποκατάστατους εξυπηρετητές, βλέπουμε ότι το hit ratio percentage που αφορά το ποσοστό με το οποίο το αντικείμενο μίας αίτησης βρίσκεται στον υποκατάστατο εξυπηρετητή με την πρώτη εμφάνιση της αίτησης, είναι 100%. Επίσης το byte hit ratio percentage που αφορά το ίδιο ποσοστό με πιο πάνω αλλά σε μορφή bytes, είναι 100%. Αυτό συμβαίνει αφού όλα τα αντικείμενα βρίσκονται σε όλους τους υποκατάστατους εξυπηρετητές και ως εκ τούτου δεν υπάρχει περίπτωση με την εμφάνιση μίας καινούριας αίτησης σε κάποιον εξυπηρετητή, το αντικείμενο αυτής να απουσιάζει από τον εξυπηρετητή. Επιπλέον, βλέπουμε ότι τόσο οι εισερχόμενοι όσο και οι εξερχόμενοι σύνδεσμοι στους υποκατάστατους εξυπηρετητές, είναι αρκετοί σύμφωνα με την κίνηση των αιτήσεων. Το ίδιο ισχύει και για τους εισερχόμενους συνδέσμους του εξυπηρετητή προέλευσης.

Αναλύοντας τα αποτελέσματα για την λειτουργία των υποκατάστατων εξυπηρετητών, παρατηρούμε ότι οι 78 από τους 423 χρησιμοποιούνται σε αυτή την πολιτική και αυτή η πληροφορία μας δίνει μία εικόνα για την κατανομή των αιτήσεων στην τοπολογία. Ο μικρός αριθμός ενεργών υποκατάστατων εξυπηρετητών, οφείλεται στο γεγονός ότι όλο το περιεχόμενο τοποθετείται σε όλους τους υποκατάστατους εξυπηρετητές. Έτσι, κανένας από αυτούς δε χρειάζεται να ζητήσει αντικείμενο από κάποιον άλλον υποκατάστατο εξυπηρετητή. Αν και φαινομενικά φαίνεται ότι οι αιτήσεις περιορίζονται μόνο σε 78 περιοχές ενώ στην πραγματικότητα έπρεπε να είναι κατανεμημένες σε μεγαλύτερο εύρος περιοχών, αξίζει να αναφέρουμε ότι η κατανομή των αιτήσεων καθορίζεται με βάση τις ζώνες ώρες των χρηστών που συμμετέχουν στο συγκεκριμένο σύνολο των αιτήσεων. Υπάρχει δηλαδή περίπτωση να μην αιτούνται περιεχόμενο όλες οι ζώνες ώρας, και αυτό είναι κάτι το οποίο καθορίζεται από τη Γεννήτρια Αιτήσεων. Επίσης, κάποιες από τις ζώνες ώρες μοιράζονται τον ίδιο κόμβο στην τοπολογία – λόγω της πραγματικής τους γεωγραφικής θέσης - έτσι μπορεί οι αιτήσεις αυτών να ανακατευθύνονται στον ίδιο υποκατάστατο εξυπηρετητή λόγω του ότι είναι ο πλησιέστερος.

Τέλος, παρατηρούμε ότι η μέση χρησιμότητα των υποκατάστατων εξυπηρετητών είναι 0,99, δηλαδή σχεδόν πλήρης, και αυτό μας δείχνει ότι οι εξυπηρετητές ως επί το πλείστον αποστέλλουν περιεχόμενο στο δίκτυο (δηλαδή στέλνουν περιεχόμενο είτε σε άλλους εξυπηρετητές είτε σε χρήστες) παρά να λαμβάνουν περιεχόμενο από τυχόν άλλους εξυπηρετητές.

6.2.2 Ακραία Πολιτική 2 - Καμία τοποθέτηση περιεχομένου σε κανέναν από τους υποκατάστατους εξυπηρετητές

Όπως έχει αναφερθεί και προηγουμένως, στην δεύτερη ακραία πολιτική όλοι οι υποκατάστατοι εξυπηρετητές είναι άδεια πριν την προσομοίωση. Όλο το περιεχόμενο που πρόκειται να ζητηθεί, βρίσκεται μόνον στον εξυπηρετητή προέλευσης.

Είναι εμφανές ότι η συγκεκριμένη ακραία πολιτική έχει πολύ διαφορετικά αποτελέσματα (Σχήμα 6.3) από την πρώτη ακραία πολιτική, κάτι το οποίο ήταν αναμενόμενο από τη στιγμή που η ιδέα της δεύτερης πολιτικής είναι ακριβώς αντίθετη από αυτή της πρώτης.

Στα αποτελέσματα για τους χρήστες του προσομοιωτή φαίνεται ότι έχουν ολοκληρωθεί 300778 αιτήσεις, αρκετά πιο λίγες από αυτές που εξυπηρετήθηκαν με την προσομοίωση της προηγούμενης πολιτικής. Συγκεκριμένα, δεν εξυπηρετούνται 1852 αιτήσεις και γίνονται 798 επιπλέον προσπάθειες (retries) για την εξυπηρέτηση κάποιων από τις αιτήσεις. Ο μέσος χρόνος απόκρισης είναι 28,6 δευτερόλεπτα, ακριβώς διπλάσιος από τον χρόνο απόκρισης της προηγούμενης πολιτικής.

```
Bottle: no_alloc_5000
CLIENT SIDE MEASUREMENTS
Completed requests: 300778
Aborted requests: 1852
Retries: 798
Not enough outgoing connections: 0
Mean response time: 28.6455194693
SURROGATE SIDE MEASUREMENTS
Hit ratio percentage: 0.0258361985015
Byte hit ratio percentage: 0.00837517387218
Not enough outgoing connections: 0
Not enough incoming connections: 0
Failed pulls: 42
ORIGIN SIDE MEASUREMENTS
Not enough incoming connections: 0
SURROGATE SERVERS UTILITIES
Active surrogate servers: 78
Mean surrogate servers utility 0.502519856765
```

***Σχήμα 6.3:** Στατιστική αναφορά του προσομοιωτή CDNSim σχετικά με την προσομοίωση της ακραίας πολιτικής κατά την οποία δεν γίνεται καμία τοποθέτηση περιεχομένου σε κανέναν από τους υποκατάστατους εξυπηρετητές*

Προχωρώντας στα αποτελέσματα για τους υποκατάστατους εξυπηρετητές, βλέπουμε ότι το ποσοστό επιτυχίας στην εύρεση του αντικειμένου από την αρχή της εμφάνισης της αίτησης είναι 0,02% και το ποσοστό αυτό που αφορά την εύρεση των αντικειμένων εκφρασμένα σε bytes είναι 0,008%. Επίσης παρατηρούμε ότι σε αυτή την ομάδα αποτελεσμάτων, αν και όπως και προηγουμένως δεν υπάρχει κάποιο πρόβλημα για τις εισερχόμενες και εξερχόμενες συνδέσεις, υπάρχουν 42 αποτυχημένες ανακλήσεις

περιεχομένου. Δηλαδή 42 φορές, οι επεξεργαστές απέτυχαν να πάρουν περιεχόμενο είτε από κάποιον άλλον υποκατάστατο εξυπηρετητή είτε από τον εξυπηρετητή προέλευσης.

Παρόλο που δεν υπάρχει περιεχόμενο τοποθετημένο στους υποκατάστατους εξυπηρετητές, και σε αυτή την προσομοίωση έχουν χρησιμοποιηθεί 78 επεξεργαστές από το σύνολο των 423. Αυτό συμβαίνει διότι κάθε εξυπηρετητής, αφότου λάβει κάποια αίτηση θα ζητήσει αυτόματα το αντικείμενο αυτό από τον εξυπηρετητή προέλευσης διότι «γνωρίζει» ότι μόνο ο εξυπηρετητής προέλευσης περιέχει όλα τα αντικείμενα. Έτσι, κανένας υποκατάστατος εξυπηρετητής δεν αιτείται περιεχομένου από άλλους υποκατάστατους, παρά μόνον από τον εξυπηρετητή προέλευσης. Επιπρόσθετα, η μέση χρησιμότητα των υποκατάστατων εξυπηρετητών είναι 50%. Η χρησιμότητα αυτή μας δείχνει ότι οι υποκατάστατοι εξυπηρετητές αποστέλλουν και λαμβάνουν περιεχόμενο με το ίδιο ποσοστό, δηλαδή όσο όγκο περιεχομένου έχουν λάβει, τόσο όγκο περιεχομένου έχουν αποστέλλει στους χρήστες.

6.2.3 Πολιτική τυχαίας τοποθέτησης 1 – Πιθανότητα 10%

Η πολιτική αυτή αποτελεί ακόμα μία στατική πολιτική τοποθέτησης περιεχομένου, η οποία όμως, σε αντίθεση με τις δύο προηγούμενες, χρησιμοποιεί μία πιθανότητα 10% για την τοποθέτηση των αντικειμένων σε κάθε υποκατάστατο εξυπηρετητή.

Αφού λοιπόν τοποθετηθούν τα δεδομένα με την πιθανότητα αυτή στους εξυπηρετητές και εκτελέσουμε την προσομοίωση, εξάγουμε τα πιο κάτω αποτελέσματα από τον προσομοιωτή CDNsim.

Τα αποτελέσματα αυτά (Σχήμα 6.4) βρίσκονται μεταξύ των δύο ακραίων πολιτικών. Σε αυτή την περίπτωση έχουν ολοκληρωθεί 302350 αιτήσεις και 280 αιτήσεις δεν κατάφεραν να εξυπηρετηθούν από τους υποκατάστατους εξυπηρετητές. Επίσης 766 αιτήσεις έχουν επαναληφθεί έτσι ώστε το δίκτυο παράδοσης περιεχομένου του προσομοιωτή να μπορέσει να τις εξυπηρετήσει. Τέλος, από το πρώτο μέρος των αποτελεσμάτων παρατηρούμε ότι ο μέσος χρόνος απόκρισης της πολιτικής αυτής είναι 21.86 δευτερόλεπτα, αρκετά πιο μεγάλο από την πρώτη πολιτική της οποίας ο χρόνος απόκρισης είναι ο καλύτερος μέχρι στιγμής.

```
Bottle: random10_5000
CLIENT SIDE MEASUREMENTS
Completed requests: 302350
Aborted requests: 280
Retries: 766
Not enough outgoing connections: 0
Mean response time: 21.8687768826
SURROGATE SIDE MEASUREMENTS
Hit ratio percentage: 37.3426149323
Byte hit ratio percentage: 37.4436664427
Not enough outgoing connections: 0
Not enough incoming connections: 0
Failed pulls: 8
ORIGIN SIDE MEASUREMENTS
Not enough incoming connections: 0
SURROGATE SERVERS UTILITIES
Active surrogate servers: 371
Mean surrogate servers utility 0.927764172962
```

Σχήμα 6.4: Στατιστική αναφορά του προσομοιωτή CDNSim σχετικά με την προσομοίωση της πολιτικής τυχαίας τοποθέτησης αντικειμένων σε υποκατάστατους εξυπηρετητές, με πιθανότητα $p=10\%$.

Τα αποτελέσματα που αφορούν την λειτουργία των υποκατάστατων εξυπηρετητών, αναφέρουν ότι 37% των αντικειμένων που έχουν ζητηθεί σε αιτήσεις, υπήρχαν ήδη στην κρυφή μνήμη του υποκατάστατου εξυπηρετητή όπου αρχικά ανακατευθύνθηκε η αίτηση (hit ratio percentage). Επίσης και πάλι δεν υπήρχε κάποιο πρόβλημα με τους συνδέσμους των υποκατάστατων εξυπηρετητών ή του εξυπηρετητή προέλευσης, αλλά έχουν πραγματοποιηθεί 8 αποτυχημένες λήψεις από εξυπηρετητές (failed pulls), τιμή αρκετά πιο μικρή από την αντίστοιχη τιμή στα αποτελέσματα της δεύτερης ακραίας πολιτικής.

Τέλος, στο τελευταίο μέρος της στατιστικής αναφοράς, βλέπουμε ότι στην συγκεκριμένη προσομοίωση έχουν χρησιμοποιηθεί 371 υποκατάστατοι εξυπηρετητές από τους 423. Ένας αριθμός πολύ πιο μεγάλος από αυτούς που συναντήσαμε στις δύο προηγούμενες στατιστικές αναφορές για τις δύο ακραίες πολιτικές. Αυτό μας δείχνει ότι πλέον οι επεξεργαστές έχουν αρχίσει να κινούν και να διαμοιράζονται περιεχόμενο

μεταξύ τους αφού ο καθένας περιέχει διαφορετικό περιεχόμενο και έτσι θέτονται σε λειτουργία πολύ περισσότεροι, μοιράζοντας τον φόρτο εξυπηρέτησης των αιτήσεων. Αυτό φαίνεται και από την μέση χρησιμότητα των επεξεργαστών η οποία φτάνει περίπου το 93%. Βλέπουμε λοιπόν ότι οι επεξεργαστές αποστέλλουν περιεχόμενο είτε σε χρήστες είτε σε άλλους εξυπηρετητές με υψηλότερο ποσοστό από αυτό που κάνουν λήψη περιεχομένου.

6.2.4 Πολιτική τυχαίας τοποθέτησης 2 – Πιθανότητα 50%

Πιο κάτω, παραθέτονται τα αποτελέσματα της δεύτερης πολιτικής τυχαίας τοποθέτησης αντικειμένων στους υποκατάστατους εξυπηρετητές με τη διαφορά ότι σε αυτή την πολιτική χρησιμοποιούμε πιθανότητα 50%. Υπενθυμίζουμε σε αυτό το σημείο ότι η πολιτική αυτή είναι επίσης στατική και τα αντικείμενα τοποθετούνται στους υποκατάστατους εξυπηρετητές στην αρχή της προσομοίωσης και ακολούθως δεν γίνεται καμία παρεμβολή στον τρόπο λειτουργίας του προσομοιωτή.

```
Bottle: random50_5000
CLIENT SIDE MEASUREMENTS
Completed requests: 302375
Aborted requests: 255
Retries: 776
Not enough outgoing connections: 0
Mean response time: 20.1312222344
SURROGATE SIDE MEASUREMENTS
Hit ratio percentage: 50.6548980574
Byte hit ratio percentage: 50.5008064935
Not enough outgoing connections: 0
Not enough incoming connections: 0
Failed pulls: 19
ORIGIN SIDE MEASUREMENTS
Not enough incoming connections: 0
SURROGATE SERVERS UTILITIES
Active surrogate servers: 312
Mean surrogate servers utility 0.924246252968
```

Σχήμα 6.5: Στατιστική αναφορά του προσομοιωτή CDNSim σχετικά με την προσομοίωση της πολιτικής τυχαίας τοποθέτησης αντικειμένων, με πιθανότητα $p=50\%$.

Τα αποτελέσματα (Σχήμα 6.5) της πολιτικής αυτής, όπως και της προηγούμενης, κυμαίνονται στα όρια των δύο ακραίων πολιτικών. Και πάλι όμως προσεγγίζουν τα αποτελέσματα της δεύτερης ακραίας πολιτικής η οποία, όπως περιμέναμε, δεν είχε την καλύτερη απόδοση. Παρατηρούμε λοιπόν ότι ολοκληρώνονται 302375 αιτήσεις ενώ 255 αιτήσεις δεν εξυπηρετούνται. Επίσης γίνονται 776 επαναλήψεις αιτήσεων και ο μέσος χρόνος απόκρισης είναι περίπου 20 δευτερόλεπτα. Είναι εύκολο λοιπόν να συμπεράνουμε ότι χρησιμοποιώντας πιθανότητα 50%, έχουμε καλύτερα αποτελέσματα στην προσομοίωση σε αντίθεση με την πιθανότητα 10%. Αυξάνονται οι επιτυχημένες αιτήσεις και ελαττώνεται ο χρόνος απόκρισης του δικτύου, χωρίς όμως να πλησιάζει τα βέλτιστα επίπεδα της πρώτης ακραίας πολιτικής.

Προχωρώντας στο επόμενο μέρος της στατιστικής αναφοράς, είναι λογικό ότι το ποσοστό εύρεσης αντικειμένων στους υποκατάστατους εξυπηρετητές από την πρώτη εμφάνιση των αιτήσεων είναι 50%. Αυτό συμβαίνει αφού το μισό πλήθος των αντικειμένων που δύναται να εμφανιστούν σε αιτήσεις χρηστών, τοποθετείται τυχαία στο σύνολο των υποκατάστατων εξυπηρετητών. Άρα το να υπάρχει το αντικείμενο μίας αίτησης σε έναν εξυπηρετητή ή όχι καθορίζεται από την πιθανότητα 50%. Επιπλέον βλέπουμε ότι για ακόμα μια φορά δεν υπάρχει η οποιαδήποτε έλλειψη εισερχόμενων ή εξερχόμενων συνδέσμων και ότι σε αυτή την προσομοίωση έχουμε μόνο 19 αποτυχημένες λήψεις περιεχομένου.

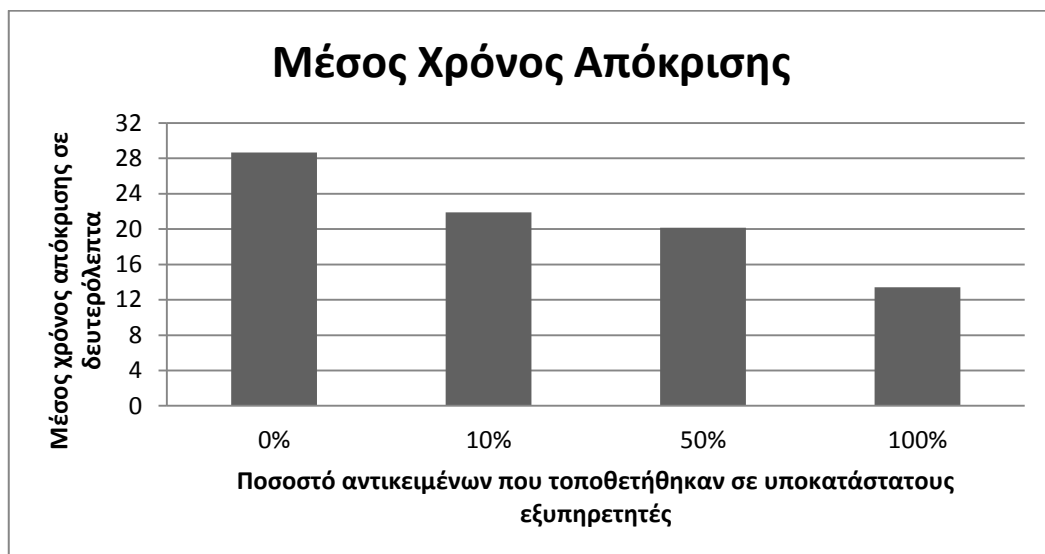
Τέλος, σε αυτή την προσομοίωση έχουν λειτουργήσει 312 υποκατάστατοι εξυπηρετητές, πιο μικρός αριθμός από τους 371 υποκατάστατους εξυπηρετητές που χρειάστηκαν στην προηγούμενη πολιτικής τυχαίας τοποθέτησης περιεχομένου. Αυτό συμβαίνει για τους ίδιους λόγους που χρησιμοποιείται μικρός αριθμός επεξεργαστών στην πρώτη ακραία πολιτική. Λόγω του ότι οι επεξεργαστές περιέχουν ένα αρκετά μεγάλο όγκο περιεχομένου, δεν χρειάζεται να ζητούν περιεχόμενο από άλλους υποκατάστατους εξυπηρετητές ή από τον εξυπηρετητή προέλευσης. Αυτό έρχεται σε αντίθεση με την περίπτωση όπου χρησιμοποιούμε πιθανότητα 10% για την τοποθέτηση των αντικειμένων αλλά και με την περίπτωση όπου οι υποκατάστατοι εξυπηρετητές ξεκινούν την προσομοίωση άδειοι. Βλέπουμε επίσης ότι η μέση χρησιμότητα των υποκατάστατων εξυπηρετητών φτάνει το 92% και συμπεραίνουμε ότι στην

συγκεκριμένη προσομοίωση το περιεχόμενο που αποστέλλουν οι υποκατάστατοι εξυπηρετητές είναι περισσότερο από αυτό που λαμβάνουν.

6.3 Συμπεράσματα

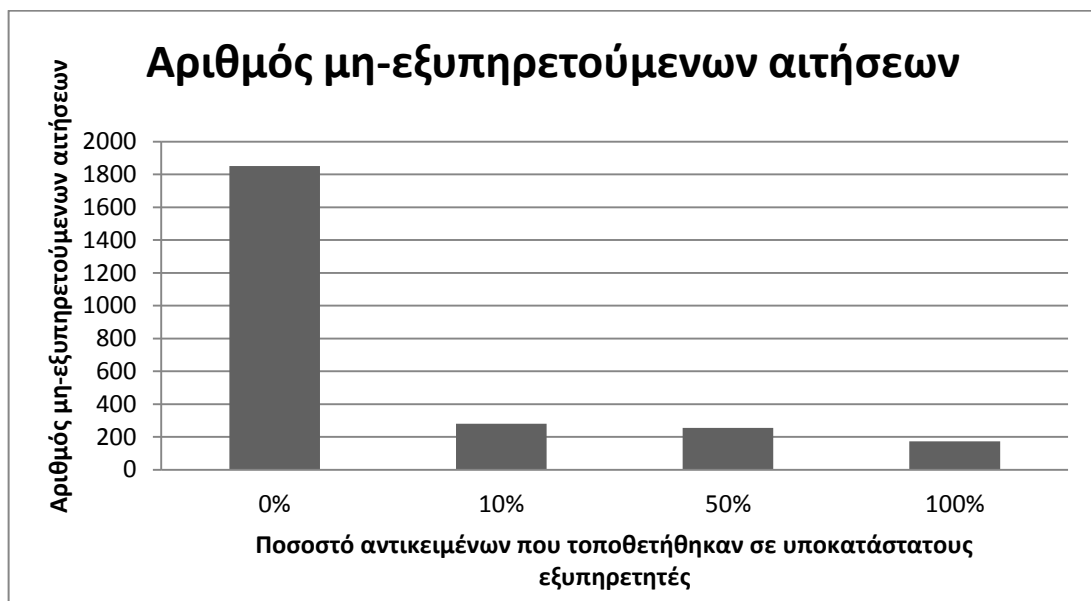
Συγκρίνοντας τα αποτελέσματα των πιο πάνω στατικών πολιτικών τοποθέτησης περιεχομένου, εξάγονται κάποια συμπεράσματα για τη συμπεριφορά του δικτύου παράδοσης περιεχομένου, ανάλογα με το ποσοστό των αντικειμένων που τοποθετούνται στους υποκατάστατους εξυπηρετητές. Πιο κάτω παραθέτονται συγκεντρωτικά τα αποτελέσματα υπό μορφή γραφημάτων και τέλος αναφέρονται τα συμπεράσματα σχετικά με τις στατικές πολιτικές.

Στο Σχήμα 6.6 φαίνεται το γράφημα του μέσου χρόνου απόκρισης ανάλογα με το ποσοστό των αντικειμένων που τοποθετούνται στους υποκατάστατους εξυπηρετητές, προτού ξεκινήσει η προσομοίωση. Φαίνεται ξεκάθαρα λοιπόν ότι τα καλύτερα αποτελέσματα δίδονται από την ακραία πολιτική στην οποία τοποθετούμε όλα τα αντικείμενα σε όλους τους υποκατάστατους εξυπηρετητές πριν την προσομοίωση. Αντίθετα, η ακραία πολιτική κατά την οποία δεν τοποθετείται κανένα αντικείμενο, έχει το μεγαλύτερο χρόνο απόκρισης, διπλάσιο από αυτόν στην περίπτωση που έχουμε 100% τοποθέτηση του περιεχομένου. Ο χρόνος απόκρισης που δίνουν οι δύο πολιτικές τυχαίας τοποθέτησης, κυμαίνεται μεταξύ των δύο ακραίων με την πολιτική της πιθανότητας $p=50\%$ να έχει ελαφρώς καλύτερα αποτελέσματα.



Σχήμα 6.6: Γράφημα μέσου χρόνου απόκρισης σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές

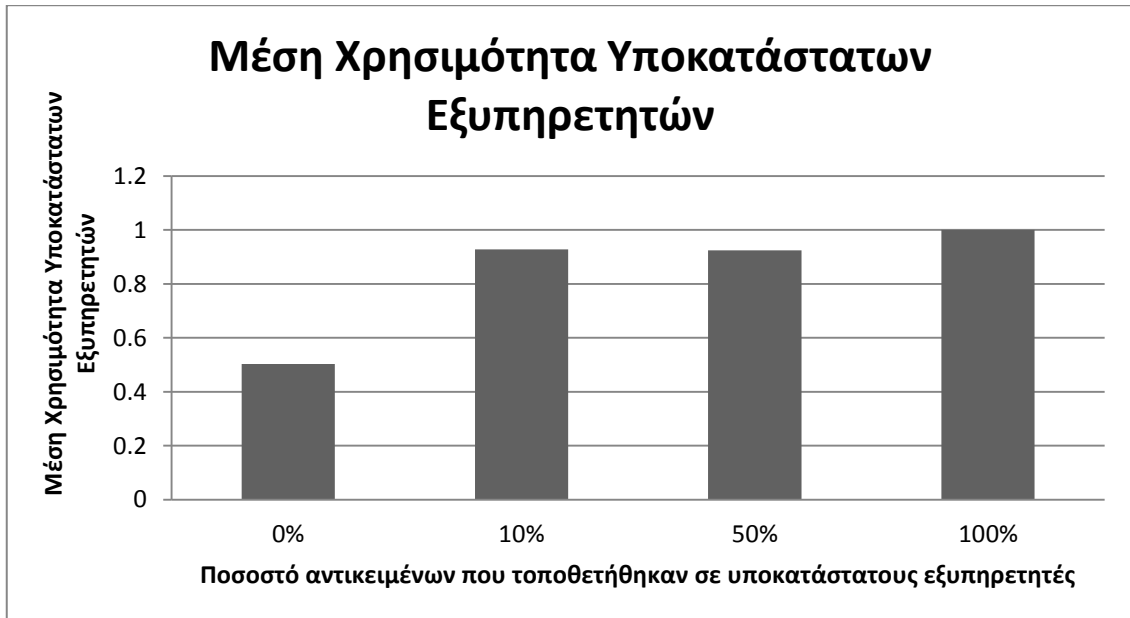
Στο Σχήμα 6.7 φαίνεται η σχέση μεταξύ των μη εξυπηρετούμενων αιτήσεων ανάλογα με το ποσοστό των αντικειμένων που τοποθετείται στους υποκατάστατους εξυπηρετητές. Και πάλι, η ακραία πολιτική στην οποία τοποθετούνται όλα τα αντικείμενα, έχει τα καλύτερα αποτελέσματα (λιγότερες μη-εξυπηρετούμενες αιτήσεις), ενώ η πολιτική στην οποία δεν τοποθετείται κανένα αντικείμενο έχει τα χειρότερα (περισσότερες μη-εξυπηρετούμενες αιτήσεις). Ο αριθμός των μη-εξυπηρετούμενων αιτήσεων αυξάνεται κατακόρυφα στην περίπτωση όπου στους υποκατάστατους εξυπηρετητές τοποθετείται το 0% του περιεχομένου.



Σχήμα 6.7: Γράφημα αριθμού μη-εξυπηρετούμενων αιτήσεων σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές

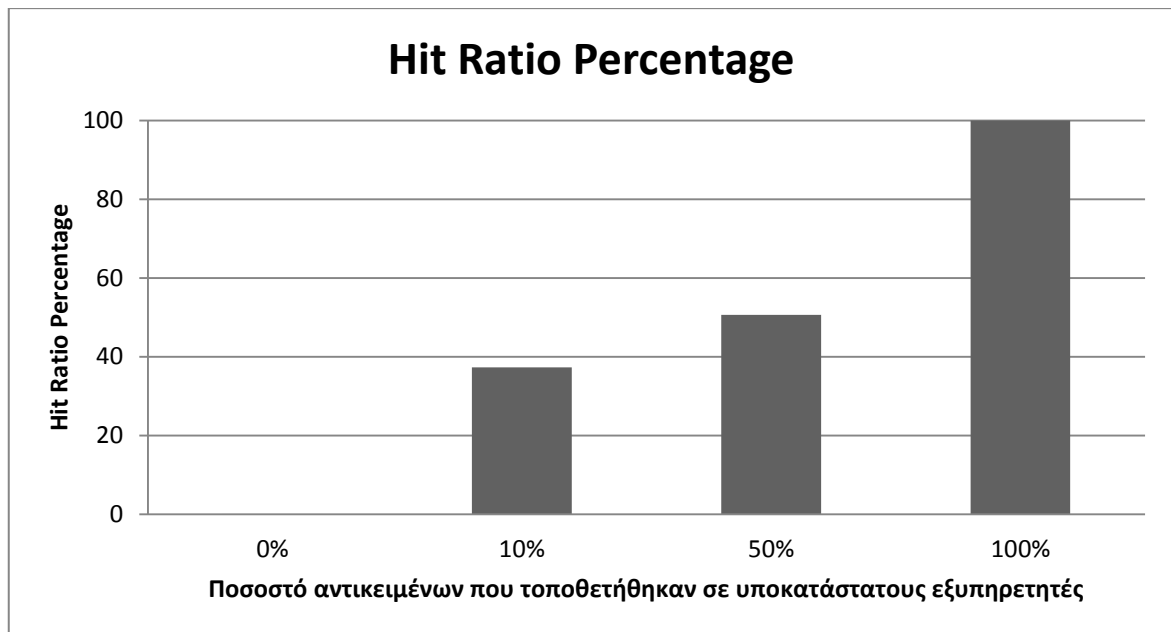
Στο Σχήμα 6.8 παρουσιάζεται η μέση χρησιμότητα των υποκατάστατων εξυπηρετητών σε σχέση και πάλι με το ποσοστό των αντικειμένων που τοποθετούνται στους υποκατάστατους εξυπηρετητές. Όπως έχει επεξηγηθεί ξανά, η τιμή της χρησιμότητας ενός υποκατάστατου εξυπηρετητή, είναι το ποσοστό περιεχομένου που ο εξυπηρετητής αποστέλλει, προς το ποσοστό περιεχομένου που ο εξυπηρετητής λαμβάνει. Επίσης η συγκεκριμένη μετρική παρέχει μία ένδειξη σχετικά με την απόδοση του δικτύου παράδοσης περιεχομένου. Φαίνεται λοιπόν ότι στην περίπτωση της ακραίας πολιτικής στην οποία τοποθετείται όλο το περιεχόμενο σε όλους τους εξυπηρετητές η απόδοση του δικτύου είναι η βέλτιστη. Την συγκεκριμένη τιμή πλησιάζουν και οι πολιτικές τυχαίας τοποθέτησης περιεχομένου. Πολύ χαμηλά αποτελέσματα όμως παρουσιάζει η

ακραία πολιτική κατά την οποία δεν τοποθετείται κανένα αντικείμενο στους υποκατάστατους εξυπηρετητές. Αποδεικνύεται λοιπόν ότι όσο πιο λίγο περιεχόμενο υπάρχει στους υποκατάστατους εξυπηρετητές, τόσο περισσότερο αναγκάζονται αυτοί να ζητούν περιεχόμενο, μειώνοντας με αυτό τον τρόπο την χρησιμότητά τους.



Σχήμα 6.8: Γράφημα μέσης χρησιμότητας υποκατάστατων εξυπηρετητών σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές

Τέλος, στο Σχήμα 6.9 παρουσιάζονται τα αποτελέσματα όσον αφορά το ποσοστό με το οποίο κατά την έλευση μίας νέας αίτησης το αντικείμενο υπήρχε ήδη στον υποκατάστατο εξυπηρετητή, δηλαδή το hit ratio percentage. Υπενθυμίζουμε ότι υψηλά ποσοστά στα αποτελέσματα της μετρικής αυτής, φανερώνουν υψηλή ποιότητα στην τοποθέτηση και κατανομή του περιεχομένου στους υποκατάστατους εξυπηρετητές. Όπως έχουμε προαναφέρει στην περιγραφή των αποτελεσμάτων, η πρώτη ακραία πολιτική, λόγω της πλήρους τοποθέτησης των αντικειμένων, επιτυγχάνει 100% hit ratio αφού όλες οι αιτήσεις θα βρουν τα αντικείμενα στην κρυφή μνήμη των υποκατάστατων εξυπηρετητών. Αντίθετα, η δεύτερη ακραία πολιτική έχει μικρό ποσοστό, κατ' ακρίβεια το μικρότερο από όλες τις πολιτικές. Και τέλος, για ακόμα μία φορά, φαίνεται ότι οι δύο πολιτικές τυχαίας τοποθέτησης κυμαίνονται μεταξύ των δύο ακραίων πολιτικών. Η τυχαία πολιτική όμως με πιθανότητα $p=50\%$ φαίνεται να έχει ελαφρώς καλύτερα αποτελέσματα.



Σχήμα 6.9: Γράφημα του «hit ratio percentage» σε σχέση με το ποσοστό αντικειμένων που τοποθετήθηκαν στους υποκατάστατους εξυπηρετητές

Παρατηρώντας τα πιο πάνω γραφήματα, δεν είναι δύσκολο να συμπεράνουμε ότι η καλύτερη πολιτική είναι η πρώτη ακραία πολιτική η οποία τοποθετεί όλο το περιεχόμενο σε όλους τους υποκατάστατους εξυπηρετητές πριν την εκκίνηση της προσομοίωσης. Αυτό άλλωστε ήταν προβλέψιμο. Έχουμε όμως ήδη αναφέρει ότι η συγκεκριμένη πολιτική είναι απαγορευτική για πρακτική εφαρμογή λόγω του υψηλού κόστους αναπαραγωγής περιεχομένου που απαιτεί. Η κρυφή μνήμη του κάθε υποκατάστατου εξυπηρετητή πρέπει να είναι πολύ μεγάλης χωρητικότητας για να εξυπηρετήσει τις ανάγκες της πολιτικής αυτής. Φανταστείτε όμως πόσο μεγάλη μνήμη απαιτείται για να χωράει όλα τα πιθανά αντικείμενα που μπορεί να αιτηθεί ο οποιοσδήποτε χρήστης. Κάτι τέτοιο λοιπόν είναι ανέφικτο.

Αντίθετα, η ακραία πολιτική που δεν τοποθετεί κανένα αντικείμενο σε κανέναν υποκατάστατο εξυπηρετητή έχει τα χειρότερα αποτελέσματα σε όλες τις πτυχές. Αν και το κόστος αναπαραγωγής του περιεχομένου είναι το ελάχιστο στη συγκεκριμένη πολιτική, αποδεικνύεται ότι τα αποτελέσματα της είναι τόσο κακά που την βγάζουν εκτός ανταγωνισμού.

Το χάσμα λοιπόν μεταξύ των δύο ακραίων πολιτικών είναι μεγάλο και για αυτόν ακριβώς το λόγο τα αποτελέσματά τους θέτουν τα όρια για την αξιολόγηση των αποτελεσμάτων μίας έξυπνης δυναμικής πολιτικής.

Το ενδιαφέρον όμως είναι ότι οι πολιτικές τυχαίας τοποθέτησης έχουν καλά αποτελέσματα τα οποία πολλές φορές πλησιάζουν αυτά της πρώτης ακραίας πολιτικής. Σίγουρα η τυχαία πολιτικής με πιθανότητα $p=50\%$ έχει καλύτερα αποτελέσματα με αυτήν με πιθανότητα $p=10\%$, η διαφορά τους όμως είναι πολύ μικρή και ίσως η δεύτερη τυχαία πολιτική είναι πιο συμφέρουσα αν αναλογιστεί κανείς ότι το κόστος αναπαραγωγής στην πρώτη είναι μεγαλύτερο. Η πολιτικής τυχαίας τοποθέτησης με πιθανότητα 50%, απαιτεί την τοποθέτηση πενταπλάσιου όγκου δεδομένων στην κρυφή μνήμη των υποκατάστατων εξυπηρετητών, σε σχέση με την τυχαία τοποθέτηση με πιθανότητα 10%.

Θέλουμε και αναμένουμε τα στατιστικά αποτελέσματα της δυναμικής πολιτικής να κυμαίνονται σίγουρα μεταξύ των δύο ακραίων πολιτικών. Αναμένουμε όμως με αγωνία να συγκρίνουμε τα αποτελέσματα της δυναμικής πολιτικής με τις δύο πολιτικές τυχαίας τοποθέτησης.

Αν τα αποτελέσματα της δυναμικής πολιτικής είναι χειρότερα από αυτά των δύο τυχαίων πολιτικών τότε δεν θα υπάρχει λόγος πρακτικής εφαρμογής της δυναμικής πολιτικής. Αν είναι μεταξύ των δύο τυχαίων πολιτικών, τότε θα πρέπει να υπολογιστεί το κόστος αναπαραγωγής του περιεχομένου και ανάλογα να καθοριστεί είναι η επιτυχία είτε η αποτυχία της δυναμικής πολιτικής. Αν όμως τα αποτελέσματα της δυναμικής πολιτικής είναι καλύτερα, τότε αποδεικνύεται πρακτικά και ξεκάθαρα ότι η εκμετάλλευση της πληροφορίας των κοινωνικών δικτύων μπορεί πράγματι να βελτιώσει την λειτουργία των δικτύων παράδοσης περιεχομένου.

Κεφάλαιο 7

Επίλογος

7.1 Γενικά Συμπεράσματα	75
7.2 Μελλοντική Εργασία	77

7.1 Γενικά Συμπεράσματα

Αδιαμφισβήτητα τα Δίκτυα Παράδοσης Περιεχομένου είναι μία από τις πιο σημαντικές υποδομές για τη διαμοίραση και τη μεταφορά του περιεχομένου στον Παγκόσμιο Ιστό. Συνεχώς γίνονται προσπάθειες για τη βελτίωση και ανάπτυξη των δικτύων παράδοσης περιεχομένου, με απώτερο σκοπό την βέλτιστη εξυπηρέτηση των χρηστών του διαδικτύου. Η σημαντικότητα των δικτύων παράδοσης περιεχομένου αποδεικνύεται με το εύρος της έρευνας που ασχολείται με το πεδίο αυτό.

Με την ένταξη των κοινωνικών δικτύων στη ζωή μας και τη συνεχή ανάπτυξή τους, έχει γίνει αντιληπτό ότι η πληροφορία που παρέχουν μπορεί να παίζει καθοριστικό ρόλο στην απόδοση των δικτύων παράδοσης περιεχομένου. Στο [1] έχει γίνει μία ενδελεχής μελέτη της πληροφορίας που μπορεί να εξαχθεί από ένα κοινωνικό δίκτυο. Όπως και στο [1] αλλά και σε άλλες δουλειές έχουν παραχθεί πολλά σημαντικά συμπεράσματα, η εύρεση των οποίων μπορεί να καθορίσει τον δρόμο της έρευνας για την ανάπτυξη και τη βελτίωση των δικτύων παράδοσης περιεχομένου.

Παρατηρώντας λοιπόν ότι πράγματι οι χρήστες μπορούν να επηρεαστούν από το κοινωνικό δίκτυο στο οποίο συμμετέχουν και μελετώντας το πώς αυτό μπορεί να γίνει [1], προσπαθήσαμε να προτείνουμε μία δυναμική πολιτική η οποία ευελπιστούμε με μελλοντική υλοποίησή της, να αποδειχθεί κατάλληλη για εφαρμογή στα δίκτυα παράδοσης περιεχομένου. Η χρήση των κοινωνικών δικτύων από μεγάλο ποσοστό του πληθυσμού, μπορεί να επηρεάσει την κίνηση του περιεχομένου στο διαδίκτυο και η

δυναμική πολιτική που έχει προταθεί προσπαθεί να προβλέψει την κίνηση αυτή και να τοποθετήσει το περιεχόμενο κοντά στους χρήστες, πριν αυτοί το ζητήσουν.

Παρατηρώντας τα αποτελέσματα των ακραίων και τυχαίων πολιτικών, φαίνεται η συμπεριφορά του δικτύου παράδοσης περιεχομένου με βάση το ποσοστό των αντικειμένων που τοποθετούνται στους εξυπηρετητές. Επίσης τα συγκεκριμένα αποτελέσματα θα αποτελέσουν και το μέτρο αξιολόγησης της δυναμικής πολιτικής στη μελλοντική της υλοποίησης. Τα επιθυμητά αποτελέσματα που θα υποδηλώνουν επιτυχία πρέπει να κυμαίνονται μεταξύ των δύο ακραίων πολιτικών και να είναι καλύτερα από τις δύο πολιτικές τυχαίας τοποθέτησης.

Τόσο τα συμπεράσματα στο [1] όσο και τα αποτελέσματα που εξήχθηκαν με την πειραματική προσομοίωση των ακραίων πολιτικών και των στατικών πολιτικών, χάραξαν το δρόμο για την κατασκευή της δυναμικής πολιτικής. Εκμεταλλευτήκαμε όλη αυτή την πολύτιμη γνώση για να χτίσουμε μία δυναμική πολιτική η οποία χρησιμοποιεί τα κατάλληλα κριτήρια για την «έξυπνη» επιλογή τοποθέτησης του περιεχομένου στους υποκατάστατους εξυπηρετητές.

Μελετώντας σχετική βιβλιογραφία, αντιλήφθηκα ότι πολλές από τις δουλειές αυτές υπολείπονταν ρεαλιστικά δεδομένα για τον έλεγχο και την αξιολόγηση των συστημάτων που πρότειναν. Έχω φτάσει στο συμπέρασμα ότι έχοντας στην κατοχή μας τη Γεννήτρια Αιτήσεων [1], αποδεικνύουμε την εγκυρότητα και την αξιοπιστία των αποτελεσμάτων των πειραμάτων που έχουν εκτελεστεί στην παρούσα διπλωματική εργασία. Η αξιοπιστία αυτή, θα χαρακτηρίζει και τα αποτελέσματα μελλοντικής πειραματικής αξιολόγησης της δυναμικής πολιτικής που έχει προταθεί.

Σε αυτό το σημείο θα ήθελα επίσης να σημειώσω την σημαντικότητα του προσομοιωτή δικτύου παράδοσης περιεχομένου CDNsim. Η προσομοίωση επιτυγχάνεται με εύκολες και ευέλικτες λειτουργίες. Η ύπαρξη ενός τέτοιου εργαλείου σε μία έρευνα σχετική με δίκτυα παράδοσης περιεχομένου, είναι απαραίτητη γιατί η κατασκευή του είναι πολύπλοκη και χρονοβόρα. Παρόλο που ο πηγαίος κώδικας του εργαλείου είναι ανοικτής φύσεως προς το κοινό, χρειάζεται χρόνος για την εξοικείωση με το εργαλείο

και την προσαρμογή τόσο του ιδίου όσο και των δεδομένων για να είναι συμβατά με τις λειτουργίες της διπλωματικής αυτής.

7.2 Μελλοντική Εργασία

Πλέον υπάρχουν στη διάθεση μας συμπεράσματα τα οποία δείχνουν πώς τα κοινωνικά δίκτυα επηρεάζουν τα δίκτυα παράδοσης περιεχομένου. Επίσης, κατά την εκπόνηση αυτής της διπλωματικής εργασίας έχουν κατασκευαστεί, υλοποιηθεί και αξιολογηθεί πειραματικά οι ακραίες και τυχαίες πολιτικές οι οποίες αποτελούν τα μέτρα σύγκρισης για την δυναμική πολιτική προ-ανάκλησης περιεχομένου.

Η δυναμική πολιτική έχει κατασκευαστεί και περιγράφει αναλυτικά σε αυτή την διπλωματική εργασία. Η υλοποίησή της όμως απαιτεί βαθιά μελέτη και κατανόηση της λειτουργίας του προσομοιωτή δικτύων παράδοσης περιεχομένου CDNsim. Επίσης, για την υλοποίηση της δυναμικής πολιτικής απαιτείται η αλλαγή του προσομοιωτή για να εκτελεί και να παίρνει αποφάσεις σύμφωνα με την πολιτική μας, τόσο για τον χειρισμό των αιτήσεων και την τοποθέτηση των αντικειμένων, όσο και την αντικατάσταση των αντικειμένων στις cache των υποκατάστατων εξυπηρετητών με την έλευση νέων αντικειμένων.

Ορίζεται λοιπόν ως μελλοντική εργασία η υλοποίηση της προτεινόμενης δυναμικής πολιτικής προ-ανάκλησης και τοποθέτησης περιεχομένου, έτσι όπως περιγράφεται σε αυτή την διπλωματική εργασία. Θα χρειαστεί να υλοποιηθούν τόσο οι προκαταρκτικές λειτουργίες όσο και τα δύο μέρη της πολιτικής. Υπενθυμίζουμε εδώ ότι όσον αφορά την τοπική αντιγραφή, προτείνονται 3 υπό-πολιτικές αντιγραφής του περιεχομένου.

Μετά την υλοποίηση και τον έλεγχο της κύριας δυναμικής πολιτικής θα πρέπει να γίνει πειραματική αξιολόγηση και ανάλογα με τα αποτελέσματα να επιλεγούν οι κατάλληλες τιμές για τις παραμέτρους, οι οποίες έχουν επεξηγηθεί σε προηγούμενο κεφάλαιο. Ο συνδυασμός διαφορετικών τιμών για τις παραμέτρους, πιθανών να δώσει τα επιθυμητά αποτελέσματα τα οποία θα αποδείξουν ότι η συγκεκριμένη δυναμική πολιτική, εκμεταλλευόμενη την πληροφορία των δικτύων παράδοσης περιεχομένου, μπορεί να βελτίωση τη λειτουργία των δικτύων παράδοσης περιεχομένου. Φυσικά η απόφαση αυτή θα παρθεί αφότου τα αποτελέσματα αυτά συγκριθούν με εκείνα των ακραίων

πολιτικών και των πολιτικών τυχαίας τοποθέτησης, όπως παρουσιάζονται στην συγκεκριμένη διπλωματική εργασία.

Επίσης σε μεταγενέστερο στάδιο θα ήταν καλό να υλοποιηθούν και οι παραλλαγές της δυναμικής πολιτικής που αποτελούν τις βελτιώσεις της, εισάγοντας πιο σύνθετες λειτουργίες. Ακολούθως θα πρέπει να γίνει πειραματική αξιολόγηση και σύγκριση των αποτελεσμάτων των παραλλαγών της δυναμικής πολιτικής για να καταλήξουμε στην καλύτερη περίπτωση.

Τέλος, θα ήθελα να προσθέσω ότι η ανάπτυξη του Διαδικτύου και συγκεκριμένα των δικτύων κοινωνικής δικτύωσης είναι συνεχής και αδιάκοπη. Επιπλέον, οι χρήστες του διαδικτύου αυξάνονται δραματικά κάνοντας αναγκαία την εύρεση νέων τρόπων εξυπηρέτησής τους και επεξεργασίας του μεγάλου όγκου δεδομένων που αυτοί παράγουν καθημερινά. Εμφανίζονται συνεχώς νέα δεδομένα τα οποία μπορούν να συμβάλουν στη βελτίωση των δικτύων παράδοσης περιεχομένου. Η δυναμική πολιτική λοιπόν που προτείνεται αλγοριθμικά στην παρούσα διπλωματική εργασία μπορεί να εμπλουτίζεται ή να διαφοροποιείται έτσι ώστε να επιτυγχάνει τα καλύτερα αποτελέσματα και τον γενικό σκοπό που είναι η μεγιστοποίηση της απόδοσης των δικτύων παράδοσης περιεχομένου και η ελαχιστοποίηση του κόστους αναπαραγωγής.

Βιβλιογραφία

- [1] Christodoulou, G., Georgiou, C., & Pallis, G. (2012). The role of twitter in YouTube videos diffusion. In *Web Information Systems Engineering-WISE 2012* (pp. 426-439). Springer Berlin Heidelberg.
- [2] Stamos, K., Pallis, G., Vakali, A., Katsaros, D., Sidiropoulos, A., & Manolopoulos, Y. (2010). CDNsim: A simulation tool for content distribution networks. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 20(2), 10.
- [3] Pallis, G., & Vakali, A. (2006). Insight and perspectives for content delivery networks. *Communications of the ACM*, 49(1), 101-106.
- [4] Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P., & Bhattacharjee, B. (2007, October). Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (pp. 29-42). ACM.
- [5] Cha, M., Mislove, A., Adams, B., & Gummadi, K. P. (2008, August). Characterizing social cascades in flickr. In *Proceedings of the first workshop on Online social networks* (pp. 13-18). ACM.
- [6] Sastry, N., Yoneki, E., & Crowcroft, J. (2009, March). Buzztraq: predicting geographical access patterns of social cascades using social networks. In *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems* (pp. 39-45). ACM.
- [7] Scellato, S., Mascolo, C., Musolesi, M., & Crowcroft, J. (2011, March). Track globally, deliver locally: improving content delivery networks by tracking geographic social cascades. In *Proceedings of the 20th international conference on World Wide Web* (pp. 457-466). ACM.

- [8] Traverso, Stefano, et al. "Tailgate: handling long-tail content with a little help from friends." *Proceedings of the 21st international conference on World Wide Web*. ACM, 2012.
- [9] Gill, P., Arlitt, M., Li, Z., & Mahanti, A. (2007, October). Youtube traffic characterization: a view from the edge. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (pp. 15-28). ACM.
- [10] Winick, J., & Jamin, S. (2002). *Inet-3.0: Internet topology generator*. Technical Report CSE-TR-456-02, University of Michigan.
- [11] Huang, C., Wang, A., Li, J., & Ross, K. W. (2008, August). Measuring and evaluating large-scale CDNs. In *Internet Measurement Conference (IMC)*.
- [12] David Moore, Ram Periakaruppan, Jim Donohoe INET 2000 Abstract: Where in the World is netgeo.caida.org? (CAIDA, SDSC, UCSD)
- [13] <http://www.caida.org/tools/utilities/netgeo/>
- [14] <http://neuron2.net/LVG/ratesandsizes.html>
- [15] Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5), 604-632.
- [16] Korn, A., Schubert, A., & Telcs, A. (2009). Lobby index in networks. *Physica A: Statistical Mechanics and its Applications*, 388(11), 2221-2226.
- [17] Freeman, L. C. (1979). Centrality in social networks conceptual clarification. *Social networks*, 1(3), 215-239.

- [18] <http://www.jeffbullas.com/2012/05/23/35-mind-numbing-youtube-facts-figures-and-statistics-infographic>
- [19] <http://en.wikipedia.org/wiki/Facebook>
- [20] <http://en.wikipedia.org/wiki/Twitter>
- [21] Buyya, R., Pathan, A. M. K., Broberg, J., & Tari, Z. (2006). A case for peering of content delivery networks. *Distributed Systems Online, IEEE*, 7(10), 3-3
- [22] <http://www.fmsasg.com/SocialNetworkAnalysis/>
- [23] Pallis, G., Katsaros, D., Dikaiakos, M. D., Loulloudes, N., & Tassiulas, L. (2009, September). On the structure and evolution of vehicular networks. In *Modeling, Analysis & Simulation of Computer and Telecommunication Systems, 2009. MASCOTS'09. IEEE International Symposium on* (pp. 1-10). IEEE.

Παράρτημα Α

Στο παράρτημα αυτό παρουσιάζεται ο κώδικας ο οποίος κατασκευάζει τα απαραίτητα αρχεία για την πολιτική τυχαίας τοποθέτησης. Τα συγκεκριμένα αρχεία θα δοθούν στον προσομοιωτή CDNsim, για την εκτέλεση της πολιτικής αυτής. Χρησιμοποιείται η γλώσσα προγραμματισμού Java.

```
package random_politic;
```

```
import java.io.BufferedWriter;  
import java.io.File;  
import java.io.FileWriter;  
import java.io.BufferedReader;  
import java.io.FileReader;  
import java.util.ArrayList;
```

```
public class Random {
```

```
    @SuppressWarnings("resource")
```

```
    public static void main(String[] args) throws Exception{
```

```
        /*Allocate the objects in each of the surrogate servers  
        * with a specific probability set by the user.  
        * This program is for the RANDOM POLITICS*/
```

```
        /*The input of this program are two files - the one with the  
        * objects and their sizes  
        * and the one with the surrogates ids both real and in  
        * topology.*/
```

```
        /*The output of this program are two files - cache file of all  
        * the surrogates and  
        * cache files for each of the surrogates. Those files are needed  
        * by the simulator CDNsim.*/
```

```
        BufferedReader br = new BufferedReader(new  
FileReader("new_website_small_shrunked.txt"));  
        BufferedReader sur_alloc = new BufferedReader(new  
FileReader("surrogate_allocation.txt"));  
        String line = null;  
        String sCurrentLine[];  
        long size;
```

```

//Read the objects and their sizes
ArrayList<Long> object_sizes = new ArrayList<Long>();
while((line = br.readLine()) != null){
    sCurrentLine = line.split(" ");
    Integer.parseInt(sCurrentLine[0]);
    size =Long.parseLong(sCurrentLine[1]);
    object_sizes.add(size);
}
int real=0;

//Read the ids of surrogate servers
ArrayList<Integer> surr_allocation = new ArrayList<Integer>();
while((line = sur_alloc.readLine()) != null){
    sCurrentLine = line.split(" ");
    Integer.parseInt(sCurrentLine[0]);
    real =Integer.parseInt(sCurrentLine[1]);
    surr_allocation.add(real);
}

br.close();

int counter = 0;

//Set the probability of the random policy
int probability = 10;

//The counter is set to 10 to make 10 different random allocations for
//testing purposes
while(counter<10){

    new File("Folder"+counter).mkdir();
    BufferedWriter out = new BufferedWriter(new
FileWriter("Folder"+counter+"/"+servers_cache_withAllocation_small
+counter+".txt"));
    BufferedReader file_names = new BufferedReader(new
FileReader("file_names.txt"));
    String file_name=null;
    String server_id = null;
    long server_cache_size = 0;

    while((file_name = file_names.readLine()) != null){
        BufferedReader server_file = new BufferedReader(new
FileReader(file_name));
        server_id =server_file.readLine();
        while((server_id = server_file.readLine()) != null){

```

```

        //Create the allocation files for each surrogate
        //server - Needed to use the CDNsim
        BufferedWriter server_out = new
        BufferedWriter(new
        FileWriter("Folder"+counter+"/"+server_id+".txt"));

        for(int i=0; i<object_sizes.size(); i++){
            //Create random number starting from zero
            //until the probability number.
            //If that number is zero, then the object is
            //chosen to get into the server's cache.
            if(new
            java.util.Random().nextInt(probability)==0){
                server_out.write(i+"\n");
                server_cache_size =
                server_cache_size + object_sizes.get(i);
            }
        }

        //Write into the cache file for CDNsim
        server_out.close();
        out.write(server_id+"\n");
        out.write("1\n");
        out.write(server_id+".txt\n");
        out.write("LRU\n");
        out.write(server_cache_size+"\n");
        server_cache_size=0;
    }
    server_file.close();

}
out.close();

    BufferedWriter out_sorted = new BufferedWriter(new
    FileWriter("Folder"+counter+"/"+servers_cache_withAllocation_small"
    +counter+"_sorted.txt"));
    ArrayList<Integer> server_id2 = new ArrayList<Integer>();
    ArrayList<String> file_id2 = new ArrayList<String>();
    ArrayList<Long> size2 = new ArrayList<Long>();
    BufferedReader in = new BufferedReader(new
    FileReader("Folder"+counter+"/"+servers_cache_withAllocation_small
    "+counter+".txt"));
    String this_Line = null;
    int j=1;

```

```

//Read again the cache file so that it can be sorted according to
//the servers ids.
while((this_Line = in.readLine()) != null){
    if(j==1){
        server_id2.add(Integer.parseInt(this_Line));
        j++;
    }
    else if(j==2 || j==4){
        j++;
        continue;
    }
    else if(j==3){
        file_id2.add(this_Line);
        j++;
    }
    else if(j==5){
        size2.add(Long.parseLong(this_Line));
        j=1;
    }
}

int g=0;
int index=0;
int num=0;
System.out.println(server_id2.size()+" "+surr_allocation.size());

//Re-write the cache file with sorted surrogate ids.
for(int i=0; i<surr_allocation.size(); i++){

    System.out.println(i);
    g = surr_allocation.get(i);
    num=i+1;
    index = server_id2.indexOf(g);
    out_sorted.write(num-1+"\n");
    out_sorted.write("1\n");
    out_sorted.write(file_id2.get(index)+"\n");
    out_sorted.write("LRU\n");
    out_sorted.write(size2.get(index)+"\n");
}

out_sorted.close();
in.close();
server_id2.clear();
file_id2.clear();
size2.clear();
file_names.close();
counter++;
}}}

```

Παράρτημα Β

Στο παράρτημα αυτό, παρουσιάζεται ο κώδικας ο οποίος διαχειρίζεται τις χρονοσφραγίδες των αιτήσεων, έτσι ώστε να κατανεμηθούν αραιά στο χρόνο, χωρίς να επηρεάζονται οι ιδιότητές τους, όπως καθορίστηκαν από την Γεννήτρια Αιτήσεων. Σκοπός είναι η καλύτερη προσομοίωση των πολιτικών στον προσομοιωτή CDNsim. Χρησιμοποιείται η γλώσσα προγραμματισμού Java.

```
package Change_requests;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.ArrayList;

public class Requests {

    public static void main(String[] args) throws IOException {

        /*This program changes the timestamps of the requests. We multiply the
        * timestamp with a constant number (5000) and then split the requests of
        * each timestamp through the new time that is created after the
        * multiplication.*/

        /*The input of this program is the initial file of the requests.*/

        /*The input of this program is the new file of the requests
        * with their timestamps sparser.*/

        BufferedReader traffic = new BufferedReader(new FileReader(
"new_traffic_mergedIDs_small1.txt"));
        BufferedWriter traffic_new = new BufferedWriter(new FileWriter(
"new_traffic_mergedIDs_small3.txt"));

        ArrayList<Long> timestamp = new ArrayList<Long>();

        String line;
        String[] temp;
        int counter=0;
        int j=1;
        long time=0;
```

```

//Step 1 is multiplying each timestamp with a specific integer and
//store the new requests to another file.
//The integer is set to be 5000. But it can change to what the
//user wants.
System.out.println("Step1");
while((line = traffic.readLine()) != null){
    System.out.println(j);
    temp = line.split("\\s+");
    time = Long.parseLong(temp[0])*5000;
    traffic_new.write(time+" "+temp[1]+" "+temp[2]+"\\n");
    if(!timestamp.contains(time)){
        timestamp.add(time);
    }

    j++;
}

traffic_new.close();
traffic.close();

//This step calculates the number of requests that were
//assigned to the same timestamp.
//This step will help us later on to divide the requests
//through the new time that is created.
System.out.println("Step2");
traffic = new BufferedReader(new
FileReader("new_traffic_mergedIDs_small3.txt"));

int[] count = new int[timestamp.size()];
long[] new_timestamps = new long[302630];

for(int i=0; i<timestamp.size(); i++){
    count[i]=0;
}

while((line = traffic.readLine()) != null){
    temp = line.split("\\s+");
    count[timestamp.indexOf(Long.parseLong(temp[0]))]++;
}
traffic.close();

BufferedWriter counts = new BufferedWriter(new
FileWriter("counts.txt"));

```

```

        double sum=0;
        for(int i=0; i<timestamp.size(); i++){
            counts.write(timestamp.get(i)+" "+count[i]+"\\n");
            sum = sum + (double) count[i];
        }

        counts.close();

        int f=0;
        traffic = new BufferedReader(new FileReader(
"new_traffic_mergedIDs_small3.txt"));
        BufferedWriter new_traffic = new BufferedWriter(new FileWriter(
"NEW_traffic_mergedIDs_small5.txt"));

        //This step assigns the requests sparser in time and
        //save the to the final file
        System.out.println("Step3");
        for(int i=0; i<timestamp.size(); i++){
            //Devide 5000 with the number of requests that
            //were assigned to the specific timestamp.
            //This is done so that we know the time between two requests.
            int number = 5000/count[i];
            long new_time = timestamp.get(i);
            for(j=0; j<count[i]; j++){
                new_timestamps[f] = new_time;
                new_time = new_time + number;
                f++;
                System.out.println(f);
            }
        }

        f=0;

        //Write the new requests with the new timestamps to the final file.
        while((line = traffic.readLine()) != null){
            temp = line.split("\\s+");
            new_traffic.write(new_timestamps[f]+" "+temp[1]+
" "+temp[2]+"\\n");
            f++;
        }
        traffic.close();
        new_traffic.close();
    }
}

```

Παράρτημα Γ

Σε αυτό το παράρτημα, παρουσιάζεται το κομμάτι κώδικα (γραμμένο σε γλώσσα προγραμματισμού python) το οποίο έχει προσαρμοστεί για τα δικά μας δεδομένα, στον προσομοιωτή CDNSim. Η αλλαγή αφορά το αρχείο BottleBuilder.py στο σημείο όπου γίνεται η ανάθεση των υποκατάστατων εξυπηρετητών, του εξυπηρετητή προέλευσης και των χρηστών στην τοπολογία του διαδικτύου που δίνεται στον προσομοιωτή. Το αρχείο αυτό γενικά, είναι υπεύθυνο για την κατασκευή του «μπουκαλιού» στο οποίο περιέχονται όλες οι πληροφορίες τις οποίες αργότερα ο προσομοιωτής χρησιμοποιεί για την εκτέλεση του σεναρίου.

Συγκεκριμένα, το πρόγραμμα διαβάζει τρία αρχεία τα οποία περιέχουν τη συγκεκριμένη ανάθεση και κάνει την αντιστοίχιση. Σε αυτό το κομμάτι κώδικα, δεν φαίνεται ο τρόπος ο οποίος επιλέγεται ο αντιπροσωπευτικός κόμβος στην τοπολογία για τον κάθε εξυπηρετητή, αφού για αυτή την απόφαση χρειάστηκε η κατασκευή πολλών μικρών εργαλείων όπως και προσωπικές αποφάσεις.

. (Προηγείται κώδικας...)

.

.

.

```
f = open("clients_allocation.txt", "r")
    bottleAttributes.clientIds = range(nRouters, nRouters+options.clients)
    clientsReference = {}
    myline = f.readline()
    for i in bottleAttributes.clientIds:
        firstvalue, secondvalue = myline.split()
        number1 = (int(firstvalue))
        number2 = (int(secondvalue))
        print 'PANIA'
        print number1
        print number2
        clientsReference[i]=number2
        routersReference[clientsReference[i]] += [i]
        myline = f.readline()
    f.close
    f = open("surrogate_allocation.txt", "r")
```

```

        bottleAttributes.surrogatesIds = range(nRouters+options.clients,
nRouters+options.clients+options.surrogates)
        surrogatesReference = {}
        myline = f.readline()
        for i in bottleAttributes.surrogatesIds:
            firstvalue, secondvalue = myline.split()
            number1 = (int(firstvalue))
            number2 = (int(secondvalue))
            print 'PANIA'
            print number1
            print number2
            surrogatesReference[i]=number2
            routersReference[surrogatesReference[i]] += [i]
            myline = f.readline()
        f.close

        f = open("origin_allocation.txt", "r")
        bottleAttributes.originsIds = range(nRouters+options.clients+options.surrogates,
nRouters+options.clients+options.surrogates+options.origins)
        originsReference = {}
        myline = f.readline()
        for i in bottleAttributes.originsIds:
            firstvalue, secondvalue = myline.split()
            number1 = (int(firstvalue))
            number2 = (int(secondvalue))
            print 'PANIA'
            print number1
            print number2
            originsReference[i]=number2
            routersReference[originsReference[i]] += [i]
            myline = f.readline()
        f.close

.
.
.
. (...συνέχεια)

```