



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Αλγόριθμοι για Διαχείριση και Τοποθέτηση Κρυφών Μνημών στο Διαδίκτυο

Ιωάννης Γ. Αικατερινίδης



Επιβλέπων:

Καθηγητής Τριανταφύλλου Παναγιώτης

Εξεταστική επιτροπή:

Καθηγητής Πατεράκης Μιχαήλ

Αναπλ. Καθηγητής Πνευματικάτος Διονύσιος

Διατριβή για το Μεταπτυχιακό Δίπλωμα Ειδίκευσης

Μάρτιος 2003

Περίληψη

Η αποδοτική παράδοση του περιεχομένου των εξυπηρετητών (servers) στους χρήστες μέσω του διαδικτύου, αποτελεί βασικό ζήτημα έρευνας τα τελευταία χρόνια. Οι αντιπρόσωποι διαδικτύου (web proxy servers), οι οποίοι βρίσκονται ανάμεσα στους χρήστες και στους εξυπηρετητές, διατηρούν προσωρινά αποθηκευμένα τα αντικείμενα των εξυπηρετητών και βοηθούν στην βελτίωση της απόδοσης εξυπηρετώντας τις αιτήσεις των χρηστών. Η στρατηγική τοποθέτηση των αντιπροσώπων στο δίκτυο των ISPs (Internet Service Providers) ή στα δίκτυα παροχής περιεχομένου (CDN - Content Delivery Networks) μπορεί να βελτιώσει περαιτέρω την απόδοση (από την άποψη της μείωσης των απαιτήσεων σε εύρος ζώνης, χρόνου απόκρισης κ.α.).

Ο στόχος της εργασίας αυτής είναι η ανάπτυξη ενός δυναμικού αλγορίθμου αντικατάστασης για αντιπροσώπους διαδικτύου καθώς και ενός εργαλείου που αποφασίζει σχετικά με τη θέση και τον αριθμό των αντιπροσώπων που απαιτούνται προκειμένου να επιτευχθούν οι επιθυμητές βελτιώσεις στην απόδοση του συστήματος. Πιστεύουμε ότι ένα τέτοιο εργαλείο είναι ιδιαίτερα χρήσιμο σε όλα τα εμπλεκόμενα μέρη (ISPs, CDNs, διαχειριστές εξυπηρετητών) του γενικότερου πλαισίου που έχει να κάνει με την παράδοση του περιεχομένου των εξυπηρετητών μέσω του διαδικτύου.

Στα πλαίσια αυτής της εργασίας μελετήθηκε εκτεταμένα η απόδοση των αλγορίθμων αντικατάστασης σε κρυφούς αντιπροσώπους. Τα αποτελέσματα, μας έδωσαν την δυνατότητα αφού συνοψίσουμε την απόδοση των αντιπροσώπων κάτω από διαφορετικά χαρακτηριστικά του περιβάλλοντος δράσης τους και μετρικές απόδοσης, να αναπτύξουμε ένα δυναμικό αλγόριθμο αντικατάστασης. Στη συνέχεια βασιζόμενοι στην ίδια μελέτη, προσαρμόσαμε τους αλγορίθμους τοποθέτησης αντιγράφων του περιεχομένου των εξυπηρετητών, που αναφέρονται σε σχετική βιβλιογραφία, στην τοποθέτηση κρυφών αντιπροσώπων και εξετάσαμε την απόδοσή τους. Τέλος επιλέγοντας τον αποδοτικότερο αλγόριθμο τοποθέτησης, αναπτύξαμε το σχετικό εργαλείο. Μελετήσαμε την αποτελεσματικότητα του εφαρμόζοντας το σε έξι ενδεικτικά σενάρια.

Ευχαριστίες

Στην οικογένεια μου.

Οι τιμές να επιβάλει η μάνα μου, ο πατέρας μου να με αγαπάει και η Παναγιώτα Τριανταφυλλίδη να με καλύπτει με την αγάπη της σε όλη τη διάρκεια. Η ευγνωμοσύνη μου είναι όλη της. Η ευγνωμοσύνη είναι η βάση της ζωής μου γιατί ήταν εκείνη που με έκανε να γίνω αυτό που είμαι σήμερα. Ευχαριστώ, μάνα, πατέρα, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει. Ευχαριστώ, Παναγιώτα, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει.

Ευχαριστώ, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει. Ευχαριστώ, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει. Ευχαριστώ, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει.

Ευχαριστώ, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει. Ευχαριστώ, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει. Ευχαριστώ, για όλη την αγάπη που μου έχετε δώσει και για όλη την υποστήριξη που μου έχετε προσφέρει.

Ευχαριστίες

Θα ήθελα να εκφράσω τις ειλικρινείς μου ευχαριστίες στον καθηγητή μου κ. Παναγιώτη Τριανταφύλλου για την πολύτιμη καθοδήγηση και βοήθεια του σε αυτή την διατριβή. Η εμπιστοσύνη που μου έδειξε όλο αυτό το διάστημα αλλά και το κουράγιο που μου παρείχε ήταν ιδιαίτερα σημαντικά για την ολοκλήρωση αυτής της μεταπτυχιακής εργασίας. Επιπλέον, μου έδωσε την δυνατότητα να αποκτήσω σημαντικές γνώσεις και αξιόλογες εμπειρίες.

Επίσης, θα ήθελα να ευχαριστήσω θερμά τους φίλους μου και ιδιαίτερα τη Λένα Χρυσανοπούλου, οι οποίοι με την συμπαράσταση και τις συμβουλές τους με βοήθησαν στην ολοκλήρωση αυτής της εργασίας και μου έδωσαν δύναμη για να συνεχίσω την προσπάθεια μου.

Μεγάλη ευγνωμοσύνη χρωστάω στους γονείς μου και στην αδερφή μου Μαρία, χωρίς την υλική και κυρίως ηθική υποστήριξη των οποίων, δε θα ήταν δυνατή η ολοκλήρωση και συνέχιση των σπουδών μου. Τους ευχαριστώ πολύ για την δύναμη που μου έδωσαν, να συνεχίσω τις σπουδές μου.

Περιεχόμενα

1	Εισαγωγή στους Αντιπροσώπους Διαδικτύου και στην Αντιγραφή και Διανομή Περιεχομένου.....	1
1.1	Κρυφός Αντιπρόσωπος	2
1.2	Αντιγραφή και Διανομή Περιεχομένου στο Διαδίκτυο	5
1.3	Στόχοι και Δομή της Διατριβής.....	6
2	Σχετικές Μελέτες Πάνω σε Κρυφούς Αντιπροσώπους και στη Διανομή Περιεχομένου στο Διαδίκτυο	9
2.1	Συνεργαζόμενοι Κρυφοί Αντιπρόσωποι.....	9
2.1.1	Ιεραρχικές Δομές	9
2.1.2	Μη Ιεραρχικές Δομές	10
2.2	Συνέπεια των Αντικειμένων και Αλγόριθμοι Αντικατάστασης	11
2.2.1	Συνέπεια των Αντικειμένων	11
2.2.2	Βασικοί Αλγόριθμοι Αντικατάστασης	12
2.2.3	Αλγόριθμοι Αντικατάστασης Σχεδιασμένοι για Κρυφούς Αντιπροσώπους.....	13
2.3	Πρώθηση Αντικειμένων στους Χρήστες.....	15
2.4	Αντιγραφή Περιεχομένου Εξυπηρετητή.....	16
2.5	Δίκτυα Παροχής Περιεχομένου.....	17
2.5.1	Διασπορά Αιτήσεων και Ανακατεύθυνση Αυτών	19
2.5.2	Συνέπεια των Αντιγράφων.....	20
2.6	Επιλογή Κατάλληλων Κόμβων για Φιλοξενία Αντικειμένων.....	20
2.6.1	Αλγόριθμοι Τοποθέτησης Αντιγράφων σε Δίκτυα Παροχής Περιεχομένου	22

3	Δομή Συστήματος για την Μελέτη Απόδοσης των Αντιπροσώπων Διαδικτύου	25
3.1	Στήσιμο Συστήματος για την Μελέτη της Απόδοσης	25
3.1.1	Μοντέλο Διαδικτύου	25
3.1.2	Δημιουργία Συνθετικών Ροών Αιτήσεων	27
3.1.3	Μοντελοποίηση Συνεργασίας Κρυφών Αντιπροσώπων.....	28
3.2	Αλγόριθμοι Αντικατάστασης και Σχετικοί Μηχανισμοί	28
3.3	Μετρικές Απόδοσης	30
4	Μοντέλο Συστήματος για την Τοποθέτηση Κρυφών Αντιπροσώπων στο Διαδίκτυο	33
4.1	Μοντελοποίηση Συστήματος.....	34
4.1.1	Μοντέλο Δικτύου	34
4.1.2	Εκτιμώμενη Απόδοση Κρυφών Αντιπροσώπων	37
4.1.3	Μοντέλο Απόδοσης Συστήματος.....	38
4.2	Αλγόριθμοι Τοποθέτησης Κρυφών Αντιπροσώπων.....	43
5	Αποτελέσματα Μελέτης Απόδοσης Αντιπροσώπων Διαδικτύου	45
5.1	Αποτελέσματα με Βάση τις Τρεις Μετρικές Απόδοσης	45
5.1.1	Αποτελέσματα Βασιζόμενοι στο Hit Ratio	45
5.1.2	Αποτελέσματα Βασιζόμενοι στο Μέσο Χρόνο Απόκρισης.....	46
5.1.3	Αποτελέσματα Βασιζόμενοι στις Απαιτήσεις Εύρους Ζώνης	47
5.1.4	Επιπρόσθετα Αποτελέσματα	47
5.2	Γενικότερα Συμπεράσματα Σχετικά με τα Αποτελέσματα	48
5.3	Παρουσίαση Δυναμικού Αλγόριθμου Αντικατάστασης	49
5.3.1	Εκτίμηση Απόδοσης Αλγορίθμων	50
5.3.2	Περιγραφή Υποσυστημάτων	50
5.3.3	Αποτελέσματα Μελέτης Απόδοσης Δυναμικού Αλγορίθμου	52
6	Μελέτη Απόδοσης Αλγορίθμων Τοποθέτησης Αντιπροσώπων Διαδικτύου	55
6.1	Τοπολογία Δικτύου.....	55
6.2	Υπολογισμός Κόστους Γράφου.....	57
6.3	Αποτελέσματα Απόδοσης	57

7	Περιγραφή Εργαλείου Τοποθέτησης Αντιπροσώπων Διαδικτύου	61
7.1	Πλατφόρμα Ανάπτυξης.....	62
7.2	Περιγραφή του ProxyTeller.....	63
7.2.1	Είσοδος Χρηστών στο Σύστημα	63
7.2.2	Βασικά Σενάρια Δράσης του ProxyTeller.....	63
7.2.3	Είσοδος Παραμέτρων Συστήματος.....	67
7.2.4	Παρουσίαση Αποτελεσμάτων.....	71
7.2.5	Επιπρόσθετες Λειτουργίες του ProxyTeller.....	72
7.3	Παρουσίαση Ενδεικτικών Αποτελεσμάτων Χρήσης του ProxyTeller.....	75
7.3.1	Αποτελέσματα με Βάση το Χρόνο Απόκρισης.....	75
7.3.2	Αποτελέσματα με Βάση τις Απαιτήσεις σε Εύρος Ζώνης.....	79
7.3.3	Ενδεικτικό Παράδειγμα Εφαρμογής του Greedy Αλγορίθμου	80
7.4	Χρόνος Εκτέλεσης του ProxyTeller.....	81
8	Συμπεράσματα	83
	Βιβλιογραφία	87
	Ειδικό Λεξιλόγιο	91

Λίστα Γραφικών Παραστάσεων και Σχημάτων

Σχήμα 1-1. Λειτουργία ενός κρυφού αντιπροσώπου.	2
Σχήμα 1-2. Κρυφός αντιπρόσωπος διαδικτύου και εξερευνητή ιστοσελίδων.	3
Σχήμα 2-1. Δίκτυο Παροχής Περιεχομένου. Με κόκκινα βέλη φαίνεται η προώθηση των περιεχομένων του εξυπηρετητή στους κατάλληλους κόμβους του CDN.	18
Σχήμα 4-1. Η ραχοκοκαλιά του δικτύου της AT&T στη Βόρεια Αμερική.	34
Σχήμα 4-2. Ο γράφος που αντιπροσωπεύει την ραχοκοκαλιά του δικτύου της AT&T στη Βόρεια Αμερική.	35
Σχήμα 4-3. Γράφος με βάρη στις ακμές και στους κόμβους. Σε όλους τους κόμβους τα βάρη είναι ίσα με 10.	39
Σχήμα 4-4. Μονοπάτια με την μικρότερη απόσταση από κάθε κόμβο προς τον εξυπηρετητή. Στις άκρες των τόξων φαίνεται και η απόσταση.	39
Σχήμα 4-5. Μονοπάτια με την μικρότερη απόσταση από κάθε κόμβο προς τον εξυπηρετητή ή τον κρυφό αντιπρόσωπο (κόμβο 2).	40
Σχήμα 4-6. Μέρος του δικτύου με δύο κρυφούς αντιπροσώπους. Τα τόξα στο πάνω μέρος δείχνουν το ποσοστό πληροφoρίας που ζητείται, ενώ τα τόξα στο κάτω μέρος δείχνουν το κόστος σε κάθε βήμα. Το συνολικό κόστος προκύπτει αν προστεθούν όλοι οι όροι στο κάτω μέρος.	41
Σχήμα 5-1. Μοντέλο δυναμικού αλγόριθμου αντικατάστασης.	51
Σχήμα 5-2. Hit Ratio μεταβάλλοντας το μέγεθος του δίσκου.	52
Σχήμα 5-3. Χρόνος απόκρισης με μεταβλητό μέγεθος δίσκου.	52
Σχήμα 5-4. Χρήση εύρους ζώνης με μεταβλητό μέγεθος δίσκου.	53
Σχήμα 6-1. Ο γράφος που αντιπροσωπεύει την ραχοκοκαλιά του δικτύου της C&W στη Βόρεια Αμερική.	56
Σχήμα 6-2. Απόδοση των αλγορίθμων μετά από εφαρμογή τους στην τοπολογία δικτύου της AT&T.	58
Σχήμα 6-3. Απόδοση των αλγορίθμων μετά από εφαρμογή τους στην τοπολογία δικτύου της C&W.	59
Σχήμα 6-4. Απόδοση του αλγορίθμου Greedy μετά από εφαρμογή τους στην τοπολογία δικτύου της C&W, με μεταβλητό ποσοστό επικάλυψης.	59
Σχήμα 7-1. Φόρμα εισαγωγής στοιχείων χρήστη με σκοπό την εγγραφή του στο σύστημα.	62

Σχήμα 7-2. Φόρμα εισαγωγής στοιχείων χρήστη με σκοπό την είσοδο του στο σύστημα.....	63
Σχήμα 7-3. Σελίδα επιλογής σεναρίου δράσης.....	65
Σχήμα 7-4. Φόρμα εισαγωγής παραμέτρων συστήματος.....	66
Σχήμα 7-5 Παράδειγμα Τοπολογίας Δικτύου.	67
Σχήμα 7-6 Περιγραφή της τοπολογίας του Σχήμα 7-5.	67
Σχήμα 7-7. Δημιουργία νέας τοπολογίας με την βοήθεια του γραφικού εργαλείου.....	68
Σχήμα 7-8. Διαδικασία εισαγωγής αρχείων ροής αιτήσεων.	69
Σχήμα 7-9. Παρουσίαση αποτελεσμάτων.....	72
Σχήμα 7-10. Σελίδα ανάκτησης αποτελεσμάτων.....	73
Σχήμα 7-11. Παρουσίαση τοπολογιών που έχει εισάγει / δημιουργήσει ο χρήστης.....	74
Σχήμα 7-12. Παρουσίαση φόρτου εργασίας που έχει χρησιμοποιήσει ο χρήστης.	75
Σχήμα 7-13. Λύση του προβλήματος κατά το σενάριο 1.....	76
Σχήμα 7-14. Λύση του προβλήματος κατά το σενάριο 3.....	78
Σχήμα 7-15. Δίκτυο με 8 κόμβους μετά την τοποθέτηση 6 κρυφών αντιπροσώπων.	80

Κεφάλαιο 1

Εισαγωγή στους Αντιπροσώπους Διαδικτύου και στην Αντιγραφή και Διανομή Περιεχομένου

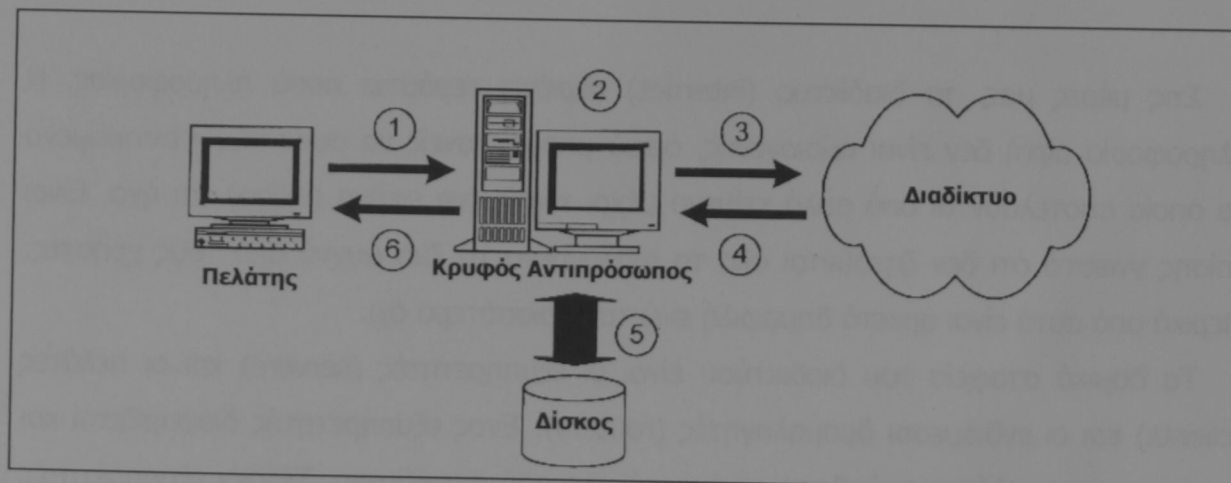
Στις μέρες μας, το διαδίκτυο (internet) περιέχει τεράστια ποσά πληροφορίας. Η πληροφορία αυτή δεν είναι ομοιογενής, αφού μπορεί κανείς να συναντήσει αντικείμενα τα οποία αποτελούνται από απλό κείμενο μέχρι κινούμενη εικόνα (video) και ήχο. Είναι επίσης γνωστό ότι δεν ζητούνται όλα τα αντικείμενα το ίδιο συχνά από τους χρήστες. Μερικά από αυτά είναι αρκετά δημοφιλή ενώ τα περισσότερα όχι.

Τα δομικά στοιχεία του διαδικτύου είναι οι εξυπηρετητές (servers) και οι πελάτες (clients) και οι ενδιάμεσοι δρομολογητές (routers). Ένας εξυπηρετητής διαχειρίζεται και παρέχει στους πελάτες πρόσβαση σε ένα σύνολο από αντικείμενα. Τέτοιοι εξυπηρετητές είναι για παράδειγμα ο Apache [28], ενώ και εταιρίες όπως η Microsoft και Netscape έχουν αναπτύξει εξυπηρετητές. Οι πελάτες στέλνουν μία αίτηση για ένα αντικείμενο στο εξυπηρετητή και ο τελευταίος αναλαμβάνει να τους το παραδώσει. Εξαιτίας του ότι ένας δικτυακός τόπος (web site) είναι πιθανό να δέχεται εκατοντάδες χιλιάδες αιτήσεις κάθε μέρα με αρκετά υψηλούς ρυθμούς αφίξεων, οι εξυπηρετητές πρέπει να είναι προσεκτικά σχεδιασμένοι για να αντέξουν όλο αυτό το φόρτο εργασίας.

Ο πιο κοινός πελάτης στο διαδίκτυο σήμερα, είναι ο εξερευνητής ιστοσελίδων (browser). Είναι εφαρμογές όπως ο Internet Explorer, Netscape Navigator κ.α. που σκοπό έχουν να παρουσιάζουν στον χειριστή του ηλεκτρονικού υπολογιστή την πληροφορία που ζητάει.

1.1 Κρυφός Αντιπρόσωπος

Τα τελευταία χρόνια, πολύς λόγος έχει γίνει για τους κρυφούς αντιπροσώπους (proxy cache ή κρυφή μνήμη). Ένας κρυφός αντιπρόσωπος βρίσκεται ανάμεσα στο εξυπηρετητή και στον πελάτη. Συμπεριφέρεται σαν εξυπηρετητής για τους πελάτες και σαν πελάτης για τους εξυπηρετητές. Μπορούν να χρησιμοποιηθούν σαν συστήματα ελέγχου πρόσβασης και προστασίας τοπικού δικτύου (firewalls) κ.α. Η πιο διαδεδομένη χρήση τους είναι η προσωρινή αποθήκευση αντικειμένων (caching) με σκοπό την μείωση της κίνησης στο διαδίκτυο και την γρηγορότερη απόκριση στις αιτήσεις των χρηστών. Η λειτουργία τους σε αυτή την περίπτωση περιγράφεται στο Σχήμα 1-1 ως εξής:

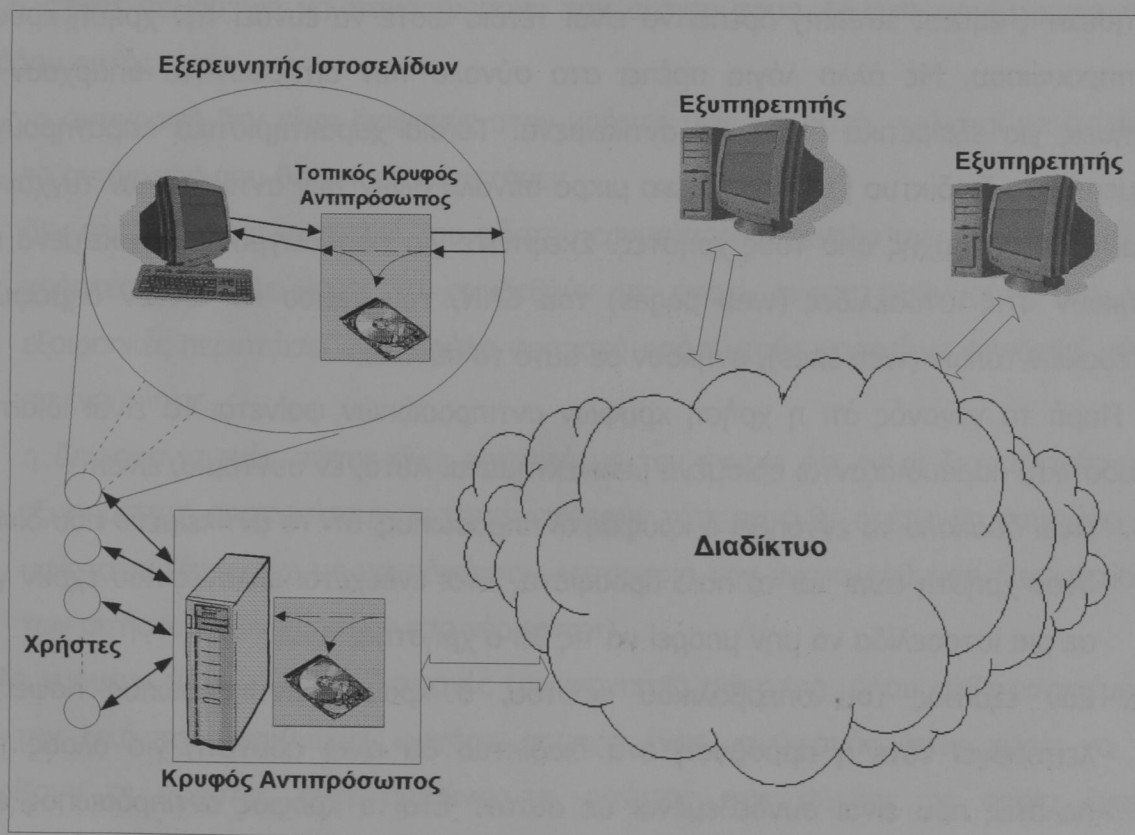


Σχήμα 1-1. Λειτουργία ενός κρυφού αντιπροσώπου.

όταν ένας πελάτης στείλει μία αίτηση για ένα αντικείμενο ① στον εξυπηρετητή, η αίτηση πρώτα επεξεργάζεται από τον κρυφό αντιπρόσωπο. Ο τελευταίος αναζητά το αντικείμενο στον αποθηκευτικό του χώρο (που μπορεί να είναι ο σκληρός δίσκος ή μνήμη RAM ή και τα δύο) ②. Εάν βρεθεί τότε στέλνεται πίσω στον πελάτη ⑥ χωρίς να χρειαστεί να επικοινωνήσει με τον απομακρυσμένο εξυπηρετητή. Σε αυτή την περίπτωση

έχουμε τοπικό κτύπημα (hit) στα αποθηκευμένα αντικείμενα του αντιπροσώπου. Έτσι από τι μία ο πελάτης λαμβάνει πιο γρήγορα το αντικείμενο, ενώ από την άλλη αποφεύγεται η επιπλέον κίνηση στο διαδίκτυο και η επιβάρυνση του εξυπηρετητή που διαθέτει το αντικείμενο. Εάν δεν βρεθεί το αντικείμενο, τότε ο αντιπρόσωπος επικοινωνεί με τον εξυπηρετητή ③, λαμβάνει το αντικείμενο ④, και το προωθεί στον πελάτη ⑥ ενώ παράλληλα το αποθηκεύει στον αποθηκευτικό του χώρο για εξυπηρέτηση μελλοντικών αιτήσεων ⑤.

Οι κρυφοί αντιπρόσωποι χωρίζονται σε δύο κατηγορίες: *forward* και *reverse*. Οι πρώτοι παρουσιάζονται σε κομβικά σημεία της ραχοκοκαλιάς του δικτύου των ISPs βοηθώντας στη μείωση της κίνησης του διαδικτύου. Οι τελευταίοι υπάρχουν στα δίκτυα παροχής περιεχομένου (CDN – Content Delivery Networks) και σκοπό έχουν να διαχειρίζονται το περιεχόμενο των εξυπηρετητών που φιλοξενούνται στο δίκτυο των CDNs. Και οι δύο τύποι κρυφών αντιπροσώπων ενεργούν κατά τον ίδιο τρόπο βοηθώντας στην βελτίωση της απόδοσης τους συστήματος στο οποίο δραστηριοποιούνται.



Σχήμα 1-2. Κρυφός αντιπρόσωπος διαδικτύου και εξερευνητή ιστοσελίδων.

Παράλληλα με τους κρυφούς αντιπροσώπους στο διαδίκτυο, υπάρχουν και κρυφοί αντιπρόσωποι υλοποιημένοι μαζί με τους εξερευνητές ιστοσελίδων (browser cache) που σκοπό έχουν να αποθηκεύουν προσωρινά, τα αντικείμενα τα οποία ζητάει ένας μόνο χρήστης. Οι διαφορές τους από τους αντιπροσώπους του διαδικτύου είναι ότι οι τελευταίοι καλύπτουν ένα μεγαλύτερο αριθμό χρηστών όπως φαίνεται στο Σχήμα 1-2.

Εν κατακλείδι, η χρήση των κρυφών αντιπροσώπων αποβλέπει στο να μειώσει το χρόνο απόκριση (latency) των αιτήσεων των χρηστών, να μειώσει την κατανάλωση του πολύτιμου εύρους ζώνης (bandwidth), μιας και η επικοινωνία με τον εξυπηρετητή γίνεται λιγότερο συχνά, και τέλος να μειώσει τον φόρτο εργασίας στον εξυπηρετητή (server load) [11],[24]. Για να είναι όμως αποτελεσματικό το εγχείρημα αυτό πρέπει από τη μια το κόστος παράδοσης του αντικειμένου μέσω του κρυφού αντιπροσώπου να είναι μικρότερο από το κόστος παράδοσης του αντικειμένου κατευθείαν από τον εξυπηρετητή (το κόστος μπορεί να εκφραστεί σαν την αποτελεσματικότητα του software και hardware που χρησιμοποιείται στον κρυφό αντιπρόσωπο σε συνδυασμό με τον φόρτο εργασίας του σε μια δεδομένη στιγμή). Από την άλλη τα χαρακτηριστικά της ροής αιτήσεων (request stream) πρέπει να είναι τέτοια ώστε να ευνοεί την χρήση κρυφού αντιπροσώπου. Με άλλα λόγια πρέπει στο σύνολο των αιτήσεων να υπάρχουν και αιτήσεις για εξαιρετικά δημοφιλή αντικείμενα. Τέτοια χαρακτηριστικά παρατηρούνται σήμερα στο διαδίκτυο [14], αφού ένα μικρό σύνολο όλων των αντικειμένων τυγχάνουν ιδιαίτερης προσοχής από τους χρήστες. Σκεφτείτε για παράδειγμα τα αντικείμενα που ανήκουν στις ιστοσελίδες (web pages) του CNN, του Yahoo και άλλων δημοφιλών δικτυακών τόπων (web sites), ανήκουν σε αυτό το σύνολο.

Παρά το γεγονός ότι η χρήση κρυφών αντιπροσώπων φαίνεται να είναι ιδιαίτερα αποδοτική, παρουσιάζονται ορισμένα μειονεκτήματα. Αυτά, εν συντομία, είναι:

- i. είναι δύσκολο να εγγυηθεί ο κρυφός αντιπρόσωπος, ότι το αντικείμενο που δίνεται στον χρήστη είναι και το πιο πρόσφατο. Έτσι ενδέχεται αλλαγές που έχουν γίνει σε μια ιστοσελίδα να μην μπορεί να τις δει ο χρήστης.
- ii. Εάν εξαιτίας του υπερβολικού φόρτου, ο κρυφός αντιπρόσωπος πάψει να λειτουργεί τότε η πρόσβαση στο διαδίκτυο θα είναι αδύνατη για όλους τους πελάτες που είναι συνδεδεμένοι σε αυτόν. Έτσι ο κρυφός αντιπρόσωπος είναι δυνατό να αποτελέσει σημείο κατάρρευσης.

- iii. Οι διαχειριστές (administrators) των εξυπηρετητών πολλές φορές θέλουν να γνωρίζουν την καταγωγή των χρηστών που προσπελαίνουν τα αντικείμενα τους, ποια αντικείμενα ζητούνται περισσότερο και άλλα στατιστικά, τα οποία τείνουν να θολώνουν οι κρυφοί αντιπρόσωποι αφού όλοι οι χρήστες έχουν το ίδιο IP.

1.2 Αντιγραφή και Διανομή Περιεχομένου στο Διαδίκτυο

Αντίθετα με τους κρυφούς αντιπροσώπους των οποίων το περιεχόμενο καθορίζεται από τις αιτήσεις των χρηστών, η αντιγραφή (replication) και η διανομή περιεχομένου είναι ένα ζήτημα για το οποίο υπεύθυνοι είναι οι διαχειριστές των εξυπηρετητών.

Μία ευρέως διαδεδομένη μέθοδος αντιγραφής περιεχομένου είναι το *mirroring*. Ιδιαίτερα φορτωμένοι εξυπηρετητές αναλαμβάνουν να δημιουργήσουν αντίγραφα του περιεχομένου τους (replicas) σε διαφορετικούς δικτυακούς τόπους (*mirrors*) οι οποίοι βρίσκονται σε χώρες σε όλο τον κόσμο. Οι χρήστες διαλέγουν ένα από τα mirrors από μία κεντρική σελίδα για να ικανοποιήσουν την αίτηση τους. Μερικά προβλήματα της μεθόδου αυτής είναι:

- i. η αντιγραφή δεν είναι διαφανής στον χρήστη μιας και αυτός καλείται να επιλέξει το αντίγραφο που θα τον εξυπηρετήσει.
- ii. δεν επιτυγχάνεται κατανομή του φόρτου εργασίας στο σύνολο των mirrors, γιατί οι χρήστες είναι ελεύθεροι να συνδεθούν σε όποιο mirror επιθυμούν. Έτσι σε εξαιρετικές περιπτώσεις ένα μεγάλο ποσοστό από αυτούς μπορεί να συνδεθεί μόνο σε ένα mirror.
- iii. η δημιουργία ενός mirror είναι οριστική με την έννοια ότι αν οι διαχειριστές του εξυπηρετητή αποφασίσουν να το μεταφέρουν τότε αυτό θα πρέπει να συνεχίσει να υφίσταται (έστω και με μοναδική του λειτουργία την ανακατεύθυνση (redirection) των αιτήσεων των χρηστών στο νέο mirror).

Το επόμενο βήμα είναι το διαφανές (transparent) mirroring, όπου κάθε αντικείμενο έχει την δική του διεύθυνση, ωστόσο υπάρχει ένας μηχανισμός όπου χωρίς να το γνωρίζουν οι χρήστες ανακατευθύνει τις αιτήσεις στα mirrors τα οποία έχουν δημιουργήσει οι διαχειριστές. Αυτή η προσέγγιση αν και ποιο ευέλικτη από την προηγούμενη δεν είναι εφικτό να εφαρμοστεί για εξυπηρετητές των οποίων οι

διαχειριστές δεν διαθέτουν τα χρήματα και την δυνατότητα να αντιγράψουν με αυτό τον τρόπο το περιεχόμενο τους σε ολόκληρο τον κόσμο.

Έτσι, αναπτύχθηκαν εταιρίες οι οποίες παρέχουν υπηρεσίες φιλοξενίας, διαχείρισης και παροχής του περιεχομένου των εξυπηρετητών. Τα δίκτυα παροχής περιεχομένου όπως είναι γνωστά (CDN – Content Delivery Networks), είναι εξοπλισμένα με δίκτυα υψηλής ταχύτητας ανεπτυγμένα σε ολόκληρο τον κόσμο και αναλαμβάνουν τόσο να φιλοξενήσουν όσο και να καταλείψουν τα αντικείμενα των εξυπηρετητών που τους έχουν εμπιστευθεί το περιεχόμενό τους, προωθώντας το με διαφανή τρόπο στους χρήστες. Η τοποθέτηση των αντιγράφων σε αυτά τα δίκτυα γίνεται με τέτοιο τρόπο ώστε να ανταποκρίνεται στις απαιτήσεις των χρηστών επιτυγχάνοντας κατανομή του φόρτου εργασίας του εξυπηρετητή, μείωση του χρόνου απόκρισης των χρηστών και της κίνησης του δικτύου. Τα οφέλη είναι ανάλογα με αυτά της ύπαρξης των κρυφών αντιπροσώπων, όμως έχοντας αφήσει τον πλήρη έλεγχο του περιεχομένου στους CDNs, ορισμένα προβλήματα που παρουσιάζονται στα συστήματα κρυφών αντιπροσώπων (όπως η συνέπεια των αντικειμένων - consistency) αντιμετωπίζονται με ποιο αποδοτικό τρόπο.

1.3 Στόχοι και Δομή της Διατριβής

Στα πλαίσια αυτής της διατριβής παρουσιάζεται μια εκτεταμένη μελέτη της απόδοσης των αλγορίθμων αντικατάστασης σε κρυφούς αντιπροσώπους. Τα αποτελέσματα, μας δίνουν την δυνατότητα να συνοψίσουμε την απόδοση των αντιπροσώπων κάτω από διαφορετικά χαρακτηριστικά του περιβάλλοντος δράσης του και μετρικές απόδοσης. Έτσι γνωρίζοντας ποιος είναι ο καταλληλότερος αλγόριθμος κάτω από ποιες προϋποθέσεις αναπτύσσουμε έναν δυναμικό αλγόριθμο αντικατάστασης. Η μελέτη μας επεκτείνεται στο πεδίο της αποδοτικής τοποθέτησης κρυφών αντιπροσώπων ή αντιγράφων στο διαδίκτυο, αποσκοπώντας στην ανάπτυξη ενός χρήσιμου εργαλείου, του *ProxyTeller*. Πιστεύουμε ότι το συγκεκριμένο εργαλείο είναι ιδιαίτερα χρήσιμο σε όλα τα εμπλεκόμενα μέρη του γενικότερου πλαισίου που έχει να κάνει με την παράδοση του περιεχομένου των εξυπηρετητών μέσω του διαδικτύου.

Η υπόλοιπη διατριβή οργανώνεται ως εξής. Στο κεφάλαιο 2 παρουσιάζεται η επισκόπηση της σχετικής βιβλιογραφίας πάνω σε κρυφούς αντιπροσώπους διαδικτύου με ιδιαίτερη έμφαση στις δομές λειτουργίας τους (ιεραρχικές και μη), στους αλγορίθμους

αντικατάστασης, στη συνέπεια των αντικειμένων και στην προώθηση αυτών στους χρήστες. Στο ίδιο κεφάλαιο συνεχίζουμε με θέματα τα οποία έχουν να κάνουν με ζητήματα που αφορούν την διανομή περιεχομένου στο διαδίκτυο. Παρουσιάζεται η λειτουργία και ο στόχος των δικτύων παροχής περιεχομένου δίνοντας ιδιαίτερο βάρος στο πρόβλημα της κατάλληλης τοποθέτησης των πλήρων αντιγράφων του περιεχομένου των εξυπηρετητών στο δίκτυο. Στο κεφάλαιο 3 παρουσιάζουμε το μοντέλο που ακολουθήσαμε με σκοπό να μελετήσουμε την απόδοση των κρυφών αντιπροσώπων στο διαδίκτυο. Το αντίστοιχο μοντέλο για την μελέτη της απόδοσης των αλγορίθμων τοποθέτησης αντιπροσώπων και αντιγράφων στο δίκτυο αναπτύσσεται στο κεφάλαιο 4. Στο κεφάλαιο 5 γίνεται μία σύντομη αναφορά των αποτελεσμάτων πάνω στη μελέτη απόδοσης των κρυφών αντιπροσώπων ενώ παράλληλα παρουσιάζεται ένας δυναμικός αλγόριθμος αντικατάστασης αντικειμένων συνοδευόμενος με τα αποτελέσματα απόδοσης του και τα γενικά συμπεράσματα μας. Στο κεφάλαιο 6 παρουσιάζονται τα συμπεράσματα μας πάνω στην απόδοση των σχετικών με την τοποθέτηση κρυφών αντιπροσώπων αλγορίθμων, και στο κεφάλαιο 7 παρουσιάζεται η βασική λειτουργία του εργαλείου που αναπτύχθηκε και έχει σαν στόχο του την αποδοτική τοποθέτηση κρυφών αντιπροσώπων ή πλήρων αντιγράφων του περιεχομένου των εξυπηρετητών στο δίκτυο. Η διατριβή τελειώνει με το κεφάλαιο 8 όπου παραθέτουμε τα συμπεράσματα αυτής της μελέτης καθώς και μελλοντικές επεκτάσεις της.

Κεφάλαιο 2

Σχετικές Μελέτες Πάνω σε Κρυφούς Αντιπροσώπους και στη Διανομή Περιεχομένου στο Διαδίκτυο

2.1 Συνεργαζόμενοι Κρυφοί Αντιπρόσωποι

2.1.1 Ιεραρχικές Δομές

Η βασική ιδέα είναι ότι οι αιτήσεις σε αντικείμενα τα οποία δεν βρίσκονται σε ένα κρυφό αντιπρόσωπο, μπορούν να εξυπηρετηθούν από ένα γειτονικό (κοντινό) αντιπρόσωπο ο οποίος ενδέχεται να έχει το αντικείμενο. Έτσι πολλοί κρυφοί αντιπρόσωποι συνεργάζονται, σχηματίζοντας μια ιεραρχική δομή (ένα δέντρο) [7]. Αιτήσεις που δεν μπορούν να εξυπηρετηθούν από τον δίσκο ενός κρυφού αντιπροσώπου, διαβιβάζονται είτε στον πατέρα του είτε στα αδέρφια του. Η διαδικασία αυτή συνεχίζεται μέχρι η αίτηση να φτάσει στην ρίζα του δέντρου όπου τελικά αν δεν βρεθεί ούτε εκεί το αντικείμενο, τότε διοχετεύεται στον απομακρυσμένο εξυπηρετητή.

Πολλές μελέτες έχουν γίνει πάνω σε αυτό το μοντέλο και κατά καιρούς έχουν παρουσιαστεί ορισμένες παραλλαγές, όπως για παράδειγμα να προωθείται η αίτηση μέχρι το σημείο στο οποίο θεωρείται ότι είναι πιο αποδοτικότερο, σε σύγκριση με την

απευθείας ανάκτηση του αντικειμένου από τον εξυπηρετητή. Μερικά από τα προϊόντα τα οποία εκμεταλλεύονται την ιεραρχική δομή είναι τα Harvest [6],[7], Squid [27], Apache[28].

2.1.2 Μη Ιεραρχικές Δομές

Οι ιεραρχικές δομές παρουσιάζουν ορισμένα προβλήματα πράγμα το οποίο οδήγησε στην κατανομημένη προσέγγιση του προβλήματος. Ένα από τα προβλήματα είναι ότι παρά την καλή απόδοση του συστήματος όσον αφορά το hit ratio (το hit ratio ορίζεται σαν τον λόγο των αιτήσεων που εξυπηρετήθηκαν από τον κρυφό αντιπρόσωπο προς το σύνολο των αιτήσεων) ο χρόνος απόκρισης που παρατηρούν οι χρήστες είναι αρκετά μεγάλος εξαιτίας των πολλών επιπέδων στο δέντρο, του υπερβολικού φόρτου που παρουσιάζεται στους κρυφούς αντιπροσώπους και της απόστασης αυτών από τους χρήστες.

Υπάρχουν τρεις βασικές σχεδιαστικές επιλογές οι οποίες πρέπει να ακολουθηθούν σε μία κατανομημένη προσέγγιση [10]: i) τα αντικείμενα πρέπει να εντοπίζονται και να παραδίνονται στους χρήστες με βάση την ελάχιστη απόσταση από αυτούς, ii) τα αντικείμενα πρέπει να διαμοιράζονται ανάμεσα σε πολλούς αντιπροσώπους και σε πολλούς χρήστες και iii) οι αντιπρόσωποι πρέπει να είναι όσο το δυνατόν πιο κοντά στους χρήστες.

Μια πρώτη απόπειρα κατανομημένου συστήματος κρυφών αντιπροσώπων έχει γίνει από το CRISP (**C**aching and **R**eplication for **I**nternet **S**ervice **P**erformance) [8],[9]. Η βασική ιδέα, κρύβεται πίσω από την έννοια του συμβουλευτικού εξυπηρετητή (directory server ή mapping server) ο οποίος διατηρεί πληροφορία με το τι είναι αποθηκευμένου και που. Στην περίπτωση που μία αίτηση δεν μπορεί να εξυπηρετηθεί από ένα αντιπρόσωπο, ρωτάται ο συμβουλευτικός εξυπηρετητής ο οποίος γνωρίζει τι είναι αποθηκευμένο σε όλους τους αντιπροσώπους που απαρτίζουν το κατανομημένο σύστημα. Σε περίπτωση που το αντικείμενο είναι κάπου αποθηκευμένο στο σύστημα, η αίτηση κατευθύνεται στον αντιπρόσωπο που το έχει, διαφορετικά η αίτηση κατευθύνεται στον απομακρυσμένο εξυπηρετητή. Ένα παρόμοιο με το CRISP σύστημα το οποίο όμως υποστηρίζει και δυναμικού τύπου αντικείμενα όπως βίντεο και μουσική είναι το MiddleMan και παρουσιάζεται στο [16].

Το ότι υπάρχει μία κεντρική υπηρεσία που αναλαμβάνει να απαντήσει στις ερωτήσεις των αντιπροσώπων μπορεί να οδηγήσει στην κατάρρευση του συστήματος όταν ο συμβουλευτικός εξυπηρετητής πάψει να λειτουργεί. Μία εναλλακτική προσέγγιση είναι ο κατακερματισμός του καταλόγου των αντικειμένων που είναι αποθηκευμένα στους αντιπροσώπους σε N υποκαταλόγους τους οποίους διαχειρίζονται N συμβουλευτικοί εξυπηρετητές [29]. Έτσι ο φόρτος στον συμβουλευτικό εξυπηρετητή τώρα κατανέμεται. Μία πιθανή βλάβη σε ένα από τους συμβουλευτές επηρεάζει μόνο τον υποκατάλογο που έχει αναλάβει να διαχειρίζεται. Αυτό μειώνει την πιθανότητα καθολικής βλάβης του συστήματος. Ένα βήμα παραπέρα μας οδηγεί στην απουσία συμβουλευτικών εξυπηρετητών. Σε αυτή την περίπτωση ο κατάλογος των αντικειμένων που είναι αποθηκευμένα στο σύστημα αντιγράφεται σε όλους τους κρυφούς αντιπροσώπους του συστήματος. Έτσι ανά πάσα στιγμή, οι κρυφοί αντιπρόσωποι γνωρίζουν που να προωθήσουν την αίτηση για αντικείμενο το οποίο δεν έχουν οι ίδιοι. Εξαιτίας του ότι με την πάροδο του χρόνου ο κατάλογος των αντικειμένων μπορεί να γίνει ιδιαίτερα μεγάλος, πράγμα το οποίο οδηγεί σε αυξημένες ανάγκες αποθηκευτικού χώρου, αλλά και διαχείρισης του καταλόγου, οδηγούμαστε στην χρήση τμήματος του καταλόγου. Έτσι αντιγράφεται στους αντιπροσώπους το τμήμα του καταλόγου που προβλέπεται να παράγει τα περισσότερα χτυπήματα στο σύστημα (hits, εξυπηρέτηση των αιτήσεων από τον δίσκο του κρυφού αντιπροσώπου).

Το σύστημα το οποίο παρουσιάζεται στο [10], υποστηρίζει μια υπηρεσία βαθμωτής εύρεσης αντικειμένων η οποία ονομάζεται ιεραρχία υποδείξεων (hint hierarchy) η οποία επιτρέπει σε κάθε κρυφό αντιπρόσωπο να εντοπίσει ποιος (κρυφός αντιπρόσωπος ή εξυπηρετητής) κατέχει το κάθε αντικείμενο που ζητείται από τους χρήστες και βρίσκεται στην κοντινότερη απόσταση (αριθμός hops). Στο [20] παρουσιάζεται ένα σύστημα "Summary Cache" το οποίο επιτρέπει τον διαμοιρασμό αντικειμένων μεταξύ αρκετών κρυφών αντιπροσώπων.

2.2 Συνέπεια των Αντικειμένων και Αλγόριθμοι Αντικατάστασης

2.2.1 Συνέπεια των Αντικειμένων

Όπως αναφέρθηκε στην εισαγωγή, είναι δυνατό τα αντικείμενα που είναι

αποθηκευμένα στους αντιπροσώπους να μην είναι συνεπή με τα αντίστοιχα αντικείμενα που βρίσκονται στους εξυπηρετητές. Ένας κρυφός αντιπρόσωπος μπορεί να παρέχει στους χρήστες *ασθενή συνέπεια* (weak consistency), δηλαδή να είναι επιτρεπτή η παράδοση ασυνεπή αντικειμένου ή *ισχυρή συνέπεια* (strong consistency), δηλαδή να παραδίδεται πάντα το πιο φρέσκο (up to date) αντικείμενο στον χρήστη. Οι περισσότεροι κρυφοί αντιπρόσωποι παρέχουν ασθενή συνέπεια γιατί το κόστος παροχής ισχυρής συνέπειας είναι ιδιαίτερα υψηλό [30].

Ένας τρόπος εξασφάλισης ασθενούς συνέπειας, είναι η χρήση του χρόνου ζωής του αντικειμένου (TTL, Time To Live) ο οποίος παρέχεται από τους εξυπηρετητές. Ο χρόνος ζωής του αντικειμένου είναι μια εκτίμηση του χρόνου κατά τον οποίο πιστεύεται ότι το αντικείμενο θα παραμείνει αμετάβλητο. Οι κρυφοί αντιπρόσωποι εξυπηρετούν ένα αντικείμενο από τον δίσκο τους μέχρι να παρέλθει ο χρόνος ζωής του. Από εκεί και πέρα ζητούν από τον εξυπηρετητή την νέα έκδοση του αντικειμένου. Ένα από τα προβλήματα αυτής της προσέγγισης είναι η δυσκολία πρόβλεψης του πραγματικού χρόνου ζωής του αντικειμένου.

Ισχυρή συνέπεια μπορεί να εξασφαλισθεί με το να ρωτάται ο εξυπηρετητής αν το αντικείμενο έχει αλλάξει, κάθε φορά που ο κρυφός αντιπρόσωπος δέχεται αίτηση για το συγκεκριμένο αντικείμενο. Είναι προφανές ότι ο χρόνος απόκρισης των αιτήσεων επιβαρύνεται με τον χρόνο που απαιτείται για την επικοινωνία του κρυφού αντιπροσώπου με τον εξυπηρετητή. Όμως ο χρήστης σε κάθε περίπτωση δεν πρόκειται να λάβει αντικείμενο το οποίο έχει αλλάξει χωρίς να το γνωρίζει.

2.2.2 Βασικοί Αλγόριθμοι Αντικατάστασης

Ο αποθηκευτικός χώρος (συνήθως σκληρός δίσκος) των κρυφών αντιπροσώπων δεν είναι απεριόριστος. Έτσι ο αλγόριθμος αντικατάστασης αναλαμβάνει να διώξει από τον δίσκο παλιότερα αντικείμενα έτσι ώστε να ελευθερωθεί χώρος για νέα. Οι πιο γνωστοί αλγόριθμοι οι οποίοι έχουν χρησιμοποιηθεί αρκετά στο παρελθόν σε διάφορους τομείς είναι:

- i. **LRU.** Ο αλγόριθμος LRU (Least Recently Used) διαγράφει τα αντικείμενα αυτά τα οποία δεν έχουν χρησιμοποιηθεί για το μεγαλύτερο χρονικό διάστημα. Υλοποιείται με την βοήθεια μίας λίστας. Αντικείμενα τα οποία ζητούνται τοποθετούνται στην

κορυφή της λίστας και αντικείμενα προς διαγραφή επιλέγονται από το τέλος της λίστας.

ii. **SIZE** [3]. Το κριτήριο διαγραφής σε αυτό τον αλγόριθμο είναι το μέγεθος του αντικειμένου. Τα μεγαλύτερα αντικείμενα διαγράφονται πρώτα έτσι ώστε να ελευθερωθεί αρκετός χώρος για την αποθήκευση περισσότερων μικρότερων αντικειμένων.

iii. **LRUMIN** [3]. Αυτός ο αλγόριθμος είναι ένας συνδυασμός του LRU και SIZE. Αρχικά εντοπίζονται αντικείμενα με μέγεθος ίσο ή μεγαλύτερο από το μέγεθος του αντικειμένου που πρόκειται να αποθηκευτεί και στη συνέχεια εφαρμόζεται ο αλγόριθμος LRU σε αυτά τα αντικείμενα.

iv. **LFU**. Είναι παρόμοιος με τον LRU μόνο που το κριτήριο δεν είναι ο χρόνος από την τελευταία αίτηση για το αντικείμενο, αλλά ο αριθμός των αιτήσεων σε αυτό. Έτσι αντικείμενα με λίγες αιτήσεις είναι υποψήφια για διαγραφή, αφού δεν είναι αρκετά δημοφιλή.

v. **LRU-K** [5]. Είναι μία επέκταση του LRU, κατά την οποία δεν λαμβάνεται υπόψη ο χρόνος από την τελευταία αίτηση, αλλά ο χρόνος από την τελευταία $K^{οστη}$ αίτηση στο αντικείμενο. Είναι προφανές ότι για $K = 1$ προκύπτει ο αλγόριθμος LRU.

2.2.3 Αλγόριθμοι Αντικατάστασης Σχεδιασμένοι για Κρυφούς Αντιπροσώπους

Οι αλγόριθμοι αντικατάστασης που παρουσιάστηκαν πιο πάνω βρίσκουν εφαρμογή σε διάφορους τομείς. Ένας από αυτούς είναι και το διαδίκτυο, όπου όμως δεν έχουν ικανοποιητική απόδοση διότι δεν αξιοποιούν αποτελεσματικά τα ιδιαίτερα χαρακτηριστικά της ροής αιτήσεων των χρηστών. Αλγόριθμοι ειδικά σχεδιασμένοι για κρυφούς αντιπροσώπους περιγράφονται παρακάτω. Χωρίζονται στις εξής τρεις κατηγορίες

i. **Επεκτάσεις των παραδοσιακών αλγόριθμων αντικατάστασης**. Τέτοιοι είναι ο LRU, LFU, FIFO, LRU-THOLD [22], LRU-K [5] οι οποίοι δεν δείχνουν να έχουν καλή απόδοση, παρά μόνο κάτω από συγκεκριμένες προϋποθέσεις.

ii. **Αλγόριθμοι βασισμένοι σε κλειδιά**. Η βασική ιδέα είναι η ταξινόμηση των αντικειμένων με βάση ένα πρωτεύον κλειδί. Στη συνέχεια γίνεται χρήση ενός δευτερεύοντος κλειδιού κ.ο.κ. Σε αυτή την κατηγορία βρίσκεται ο LOG2-SIZE [3] όπου σβήνονται πρώτα τα αντικείμενα με την μεγαλύτερη τιμή $\log_2(\text{size})$. Ο

HYPER-G [3] όπου αρχικά εφαρμόζεται ο LFU, στη συνέχεια για αντικείμενα που έχουν ίδιο αριθμό αναφορών, εφαρμόζεται ο LRU και τέλος αν υπάρχει ανάγκη χρησιμοποιείται το κριτήριο του μεγέθους όπου τα μεγαλύτερα αντικείμενα σβήνονται πρώτα.

- iii. **Αλγόριθμοι βασισμένοι στην τιμή μιας συνάρτησης.** Αλγόριθμοι αυτής της κατηγορίας επιλέγουν τα αντικείμενα προς διαγραφή με βάση μία τιμή που δείχνει πόσο πολύτιμο είναι το κάθε αντικείμενο. Η τιμή αυτή προκύπτει από την εφαρμογή μίας συνάρτησης η οποία εμπλέκει διάφορους παράγοντες που χαρακτηρίζουν ένα αντικείμενο. Αυτοί είναι για παράδειγμα ο χρόνος που μεσολάβησε από την τελευταία αίτηση προς το ίδιο αντικείμενο, η χρονική στιγμή της πρώτης αίτησης προς το αντικείμενο, το κόστος ανάκτησης του αντικειμένου από τον εξυπηρετητή κ.α. Αλγόριθμοι που ανήκουν σε αυτή την κατηγορία είναι οι SLRU [1], PPS [1], GDS [19], LNC-R-W3-U [2] και LRV [4] κ.α. Δύο από αυτούς τους αλγόριθμους περιγράφονται παρακάτω.

SLRU (Size-Adjusted LRU) [1]

Σε αυτό τον αλγόριθμο, κάθε αντικείμενο περιγράφεται από μία τιμή η οποία δίνεται από την σχέση $1/(s_i \Delta T_i)$ όπου s_i είναι το μέγεθος του αντικειμένου i και ΔT_i είναι ο αριθμός των αιτήσεων (προς άλλα αντικείμενα) που μεσολάβησαν από την τελευταία αίτηση στο αντικείμενο i . Όταν λοιπόν απαιτείται ελεύθερος χώρος, σβήνονται πρώτα τα αντικείμενα με την μικρότερη τιμή. Είναι προφανές ότι ο SLRU διατηρεί στον δίσκο δημοφιλή αντικείμενα με μικρό μέγεθος.

GDS (Greedy Dual Size) [19]

Η τιμή (H) που περιγράφει ένα αντικείμενο δίνεται από την σχέση $H = c_i/\text{size}_i$. Όπου c_i είναι το κόστος ανάκτησης του αντικειμένου από τον εξυπηρετητή (το οποίο μπορεί να εκφραστεί σε χρόνο, hops ή ακόμα μπορεί να θεωρηθεί ίσο με την μονάδα όταν στόχος είναι η αύξηση του hit ratio) και size_i είναι το μέγεθος του αντικειμένου. Κατά την διαγραφή αντικειμένων επιλέγονται πρώτα αυτά με την μικρότερη τιμή. Στη συνέχεια η τιμή H όλων των αντικειμένων που βρίσκονται στον δίσκο, μειώνεται κατά την τιμή H του αντικειμένου που διαγράφηκε. Όταν προκύψει αίτηση για το αντικείμενο i που βρίσκεται στον δίσκο, η τιμή του (H_i) ανανεώνεται στην αρχική του τιμή (δηλαδή την

τιμή που είχε την πρώτη φορά που αποθηκεύτηκε στον δίσκο). Έτσι δημοφιλή αντικείμενα τείνουν να έχουν υψηλές τιμές σε σύγκριση με όχι και τόσο δημοφιλή αντικείμενα. Μία επέκταση του GDS αλγόριθμου είναι ο GreedyDual* [21].

2.3 Προώθηση Αντικειμένων στους Χρήστες

Πέρα από την χρήση κρυφών αντιπροσώπων με σκοπό την μείωση του χρόνου απόκρισης των αιτήσεων των χρηστών, ένας άλλος μηχανισμός που αποβλέπει στο ίδιο αποτέλεσμα, είναι η προώθηση των αντικειμένων στους χρήστες πριν ακόμα αυτοί τα ζητήσουν (prefetching). Η παράδοση των αντικειμένων γίνεται κατά το διάστημα όπου ο χρήστης βλέπει τα περιεχόμενα μίας ιστοσελίδας. Το ποια αντικείμενα πρόκειται να παραδοθούν στους χρήστες, καθορίζεται από τον τρόπο με τον οποίο οι χρήστες ακολουθούν τους συνδέσμους (hyperlinks) μιας ιστοσελίδας σε συνδυασμό με την παρακολούθηση της ροής αιτήσεων από τους εξυπηρετητές που έχει σαν αποτέλεσμα τον εντοπισμό των δημοφιλών αντικειμένων.

Το σύστημα λοιπόν που αναλαμβάνει να υλοποιήσει τον παραπάνω μηχανισμό [31], αποτελείται από ένα σύνολο από διεργασίες (processes) που εκτελούνται τόσο στην πλευρά του εξυπηρετητή, όσο και στην πλευρά του χρήστη. Οι διεργασίες στην πλευρά του εξυπηρετητή αναλαμβάνουν να προβλέψουν την επόμενη αίτηση του χρήστη και να προωθήσουν το επόμενο αντικείμενο σε αυτόν. Οι διεργασίες στην πλευρά του χρήστη αναλαμβάνουν να παραδώσουν τα αντικείμενα στον χρήστη αλλά και να ενημερώσουν τις διεργασίες στον εξυπηρετητή για την συμπεριφορά του χρήστη.

Η χρήση κρυφών αντιπροσώπων σε συνδυασμό με την προώθηση αντικειμένων μπορεί να μειώσει ακόμα περισσότερο τον χρόνο που απαιτείται για να παραδοθεί ένα αντικείμενο από την στιγμή που αυτό θα ζητηθεί [26]. Οι περισσότεροι κρυφοί αντιπρόσωποι είναι παθητικοί με την έννοια ότι αναλαμβάνουν δράση μόνο όταν γίνει η απαραίτητη αίτηση (passive caching). Ενεργητικοί κρυφοί αντιπρόσωποι είναι αυτοί, που παράλληλα χρησιμοποιούν και ένα μηχανισμό προώθησης αντικειμένων (παρόμοιος με αυτόν που περιγράφηκε πιο πριν), αποθηκεύοντας αντικείμενα στον δίσκο που προβλέπεται ότι θα ζητηθούν στο μέλλον από τους χρήστες (active caching). Η προώθηση αντικειμένων μπορεί να χωριστεί σε δύο κατηγορίες με βάση τον "τόπο" όπου

γίνεται η πρόβλεψη. Έτσι η πρώτη κατηγορία είναι τοπική προώθηση (local prefetching), όπου χρησιμοποιείται τοπική πληροφορία για την πρόβλεψη (δηλαδή μελετάται η ροή αιτήσεων του συνόλου των χρηστών που καλύπτει ο κρυφός αντιπρόσωπος). Η δεύτερη κατηγορία είναι η προώθηση με βάση τις υποδείξεις του εξυπηρετητή (server hint prefetching) όπου η πρόβλεψη γίνεται με βάση τις παρατηρήσεις από την πλευρά του εξυπηρετητή σχετικά με τα αντικείμενα και την ροή αιτήσεων προς αυτά.

2.4 Αντιγραφή Περιεχομένου Εξυπηρετητή

Έχοντας αναφέρει το πλαίσιο εργασίας των κρυφών αντιπροσώπων και τα ζητήματα που σχετίζονται με αυτούς, θα παρουσιάσουμε στη συνέχεια τα δίκτυα παροχής περιεχομένου δίνοντας ιδιαίτερο βάρος στο πρόβλημα της αποδοτικής τοποθέτησης αντιγράφων στο δίκτυο, με το οποίο και θα ασχοληθούμε στη συνέχεια.

Με τον όρο αντιγραφή του περιεχομένων του εξυπηρετητή αναφερόμαστε στην δημιουργία ενός συστήματος που σκοπό έχει να δημιουργήσει και να τοποθετήσει σε κομβικά σημεία του διαδικτύου ολοκληρωμένα αντίγραφα των περιεχομένων του εξυπηρετητή. Απώτερος στόχος του συστήματος είναι η εξυπηρέτηση των αιτήσεων των χρηστών, να γίνεται στο εξής από τους εξυπηρετητές αντιγράφων (replica servers) και τον ίδιο τον εξυπηρετητή, με τέτοιο τρόπο ώστε να διασπείρονται αποδοτικά οι αιτήσεις σε όλους τους παραπάνω. Τέτοιες λύσεις βρίσκουν εφαρμογή σε ιδιαίτερα φορτωμένους εξυπηρετητές που προσπαθούν να αποφύγουν τον επιπλέον φόρτο εργασίας. Τελευταία έχουν γίνει μελέτες πάνω σε αυτό το θέμα οι οποίες προσπαθούν να μειώσουν τον αριθμό των hops με σκοπό την πρόσβαση σε ένα αντικείμενο [35], ή να τοποθετήσουν κατάλληλα τα αντικείμενα στο δίκτυο λαμβάνοντας τους περιορισμούς σε εύρος ζώνης [33].

Οι αιτήσεις είναι αναγκαίο να διοχετεύονται στα αντίγραφα με διαφανή τρόπο. Δηλαδή ο χρήστης δεν πρέπει να γνωρίζει ότι η αίτηση που έκανε προς ένα αντικείμενο του εξυπηρετητή δεν εξυπηρετείται από αυτόν, αλλά από ένα αντίγραφο του.

Ζητήματα τα οποία προκύπτουν σε τέτοια συστήματα, είναι:

- i. το πως επιλέγεται το κατάλληλο αντίγραφο για κάθε χρήστη.
- ii. πως εγγυάται το σύστημα ότι τα αντίγραφα θα είναι συνεπή με τα περιεχόμενα του εξυπηρετητή.

iii. πόσα αντίγραφα είναι απαραίτητο να δημιουργηθούν και που να τοποθετηθούν (γνωστό και ως facility location problem).

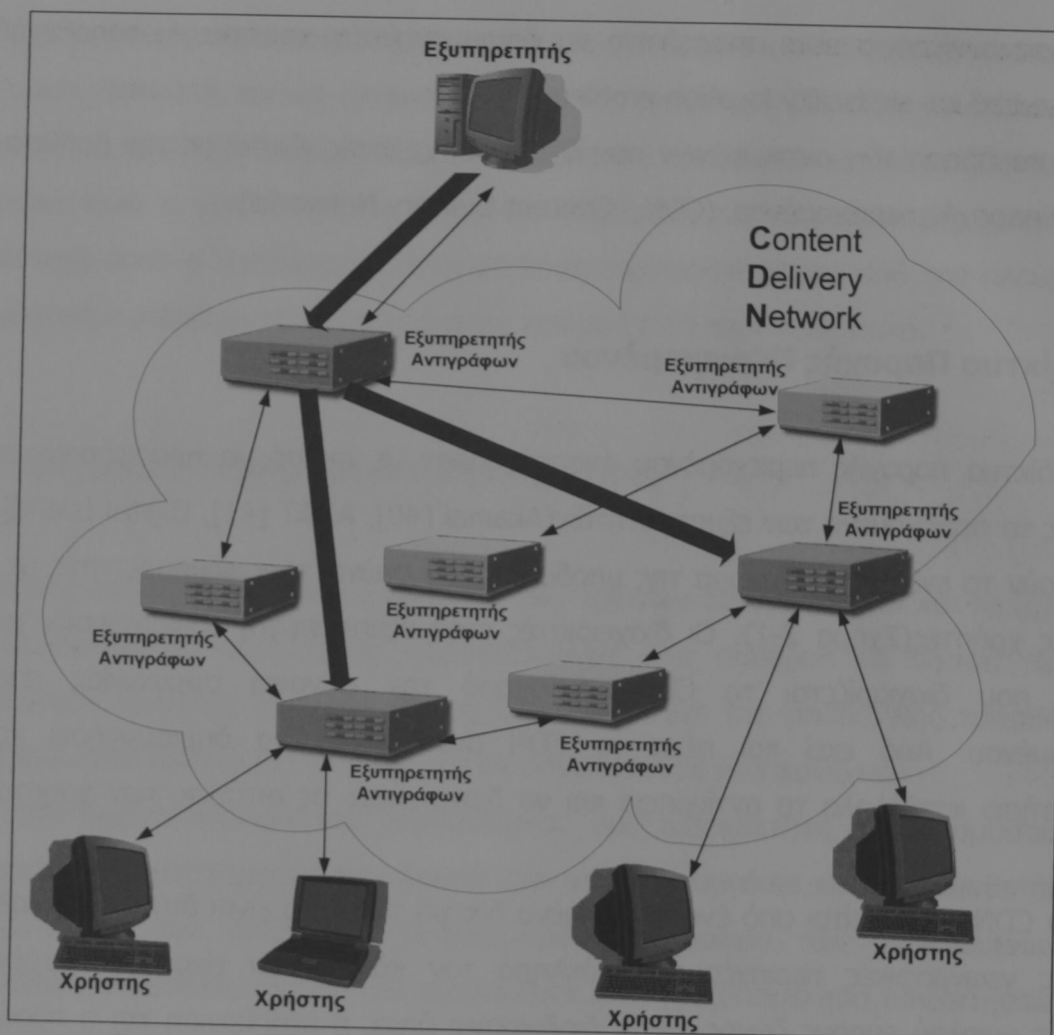
Η παράδοση των αντικειμένων που ζητούν οι χρήστες γίνεται με την βοήθεια των δικτύων παροχής περιεχομένου. (CDN - Content Delivery Networks).

2.5 Δίκτυα Παροχής Περιεχομένου

Τα δίκτυα παροχής περιεχομένου αναπτύχθηκαν με σκοπό να προσφέρουν στους χρήστες τα περιεχόμενα των εξυπηρετητών (Akamai [40], AT&T [41], Digital Island [42]). Αποτελούν το ενδιάμεσο στρώμα της υποδομής που ενώνει τους εξυπηρετητές με τους τελικούς χρήστες (Σχήμα 2-1). Οι διαχειριστές ενός εξυπηρετητή επικοινωνούν με την εταιρία που διαχειρίζεται το CDN με σκοπό την σύναψη συμφωνίας παροχής περιεχομένου. Από εκεί και πέρα το CDN αναλαμβάνει να δημιουργήσει και να τοποθετήσει κατάλληλα τα αντίγραφα και να διοχετεύσει τις αιτήσεις των χρηστών σε αυτά.

Ένα CDN αποτελείται από ένα εκτεταμένο δίκτυο το οποίο είναι δυνατό να καλύπτει μεγάλες γεωγραφικές περιοχές σε ολόκληρο τον κόσμο, και είναι εξοπλισμένο με γραμμές υψηλού εύρους ζώνης, ώστε διαδικασίες όπως η ενημέρωση και η προώθηση του περιεχομένου να γίνεται αποδοτικά (Σχήμα 2-1). Έτσι ένα από τα πλεονεκτήματα των εξυπηρετητών που αποφασίζουν να εμπιστευθούν τα περιεχόμενα τους σε CDNs, είναι ότι οι χρήστες που ζητούν αντικείμενα από αυτούς μπορούν να τα ανακτήσουν από κοντινούς σε αυτούς εξυπηρετητές (πράγμα το οποίο οδηγεί σε μικρό χρόνο απόκρισης). Παράλληλα μπορεί να αντιμετωπισθεί αποδοτικά η περίπτωση κατά την οποία το ενδιαφέρον των χρηστών είναι αυξημένο για ένα υποσύνολο των αντικειμένων (μιας και οι αιτήσεις προς αυτά τα αντικείμενα διασπείρονται σε πολλούς εξυπηρετητές σε όλο τον κόσμο), πράγμα το οποίο σε αντίθετη περίπτωση θα οδηγούσε σε υπερφόρτωση του εξυπηρετητή και πιθανή παύση της λειτουργίας του.

Τα δύο λοιπόν οφέλη που πρόκειται να αποκομίσουν οι πελάτες των CDNs (δηλαδή οι διαχειριστές των εξυπηρετητών) είναι i) η διασπορά του περιεχομένου του εξυπηρετητή σε όλο τον κόσμο και ii) η προστασία του εξυπηρετητή από απροσδόκητη αύξηση του φόρτου εργασίας του.



Σχήμα 2-1. Δίκτυο Παροχής Περιεχομένου. Με κόκκινα βέλη φαίνεται η προώθηση των περιεχομένων του εξυπηρετητή στους κατάλληλους κόμβους του CDN.

Θα μπορούσε κανείς να πει ότι τα δίκτυα παροχής περιεχομένου θα μπορούσαν να αντικατασταθούν από συνεργαζόμενους κρυφούς αντιπροσώπους αφού τέτοια συστήματα παρέχουν τα δύο παραπάνω οφέλη, επιτρέποντας στους χρήστες να ικανοποιήσουν τις αιτήσεις τους από κοντινότερους σε αυτούς εξυπηρετητές (τους κρυφούς αντιπροσώπους). Οι λόγοι όμως για τους οποίους αυτό δεν μπορεί να γίνει είναι:

- i. στο περιβάλλον ενός CDN οι εξυπηρετητές μπορούν και ελέγχουν τα αντίγραφα των περιεχομένων τους σχετικά με την συνέπεια αυτών ενώ παράλληλα μπορούν να συλλέγουν και στατιστικά ζήτησης του περιεχομένου τους.
- ii. οι CDNs μπορούν να διαχειριστούν δυναμικό περιεχόμενο, περιεχόμενο συνεχούς ροής (π.χ. video) και περιεχόμενο που υπόκεινται σε μηχανισμούς και κανόνες ασφάλειας.

iii. η προώθηση περιεχομένου στους χρήστες είναι περισσότερο αποδοτική στο περιβάλλον ενός CDN εξαιτίας των επιχειρηματικών του σχέσεων με τους εξυπηρετητές.

Τα δίκτυα παροχής περιεχομένου κατηγοριοποιούνται α) με βάση την σχέση τους με τους ISPs (Internet Service Providers) των οποίων το δίκτυο χρησιμοποιούν, και β) με βάση τους μηχανισμούς που χρησιμοποιούν για να εξυπηρετήσουν τις αιτήσεις των χρηστών.

Όσον αφορά την σχέση που αναπτύσσεται μεταξύ του δικτύου παροχής περιεχομένου και του ISP, οι CDNs μπορούν να χωριστούν σε δύο κατηγορίες: multi-ISP CDN και single-ISP CDN. Οι multi-ISP CDN χρησιμοποιούν τα δίκτυα πολλών ISPs σε ολόκληρο τον κόσμο, με σκοπό να κτίσουν το δίκτυο τους και να προσφέρουν τις υπηρεσίες τους. Το πλεονέκτημα αυτής της προσέγγισης είναι ότι μπορούν να επεκταθούν εύκολα σε όλο τον κόσμο φτάνοντας πιο κοντά στους χρήστες. Οι single-ISP CDNs αναπτύσσουν το δικό τους ιδιωτικό δίκτυο (όπως για παράδειγμα η AT&T) σε ολόκληρο τον κόσμο χωρίς να βασίζονται στην αγορά και εκμετάλλευση εύρους ζώνης από άλλους ISPs. Από την άλλη όμως δεν μπορούν να επεκταθούν σε ολόκληρο τον κόσμο, τόσο εύκολα όσο οι multi-ISP CDNs.

Όσον αφορά τον τρόπο με τον οποίο διαχειρίζονται τα περιεχόμενα του εξυπηρετητή, μπορούν να χωριστούν σε διάφορες κατηγορίες, ανάλογα με το αν φιλοξενούν και τον ίδιο τον εξυπηρετητή στο δίκτυο τους ή όχι, με το αν ένα μέρος των αντικειμένων εξυπηρετείται από τους CDNs και το υπόλοιπο από τον εξυπηρετητή ή όχι.

2.5.1 Διασπορά Αιτήσεων και Ανακατεύθυνση Αυτών

Ο μηχανισμός προώθησης των περιεχομένων στους χρήστες πρέπει να είναι διαφανής. Με άλλα λόγια η χρήστες δεν πρέπει να κάνουν καμίας ρύθμιση ούτε πρέπει να γνωρίζουν από που ανακτούν τελικά το αντικείμενο που ζητούν.

Ένας απλός τρόπος ανακατεύθυνσης (redirection) των αιτήσεων είναι ο εξής: η εφαρμογή του χρήστη επικοινωνεί με τον εξυπηρετητή ο οποίος στη συνέχεια της επιστρέφει μια λίστα με τους κόμβους του CDN στους οποίους βρίσκεται το ζητούμενο αντικείμενο. Επιλέγεται ένας από τους κόμβους στην λίστα και κατευθύνεται η αίτηση του χρήστη σε αυτόν για να εξυπηρετηθεί. Ένα από τα πλεονεκτήματα αυτής της

μεθόδου είναι ότι η εφαρμογή του χρήστης μπορεί να επιλέξει με διάφορα κριτήρια τον κόμβο από την λίστα που θα προσφέρει την καλύτερη απόδοση (π.χ. μειωμένο χρόνο απόκρισης). Από την άλλη όμως αυτή επιλογή μπορεί να είναι αποδοτική για τον χρήστη όχι όμως και για όλο το σύστημα, δηλαδή μπορεί για μικρά χρονικά διαστήματα να οδηγούνται οι αιτήσεις των χρηστών σε συγκεκριμένους μόνο κόμβους αφήνοντας άλλους κόμβους ανεκμετάλλετους.

Μια δεύτερη προσέγγιση είναι η χρήση ενός switch στο οποίο θα διοχετεύονται όλες οι αιτήσεις των χρηστών. Αυτό το switch στη συνέχεια θα αναλαμβάνει να επιλέγει τον κατάλληλο κόμβο κάθε φορά για κάθε αίτηση που δέχεται. Έτσι δίνεται η εντύπωση στους χρήστες ότι επικοινωνούν με έναν μόνο εξυπηρετητή.

2.5.2 Συνέπεια των Αντιγράφων

Ένα βασικό πλεονέκτημα των CDNs σε σχέση με τα συστήματα συνεργαζόμενων αντιπροσώπων, είναι ότι προσφέρουν μια στενότερη σχέση μεταξύ των κόμβων που φιλοξενούν τα αντίγραφα και του εξυπηρετητή. Αυτό δίνει την δυνατότητα παροχής ισχυρής συνέπειας (strong consistency) στους χρήστες, παρέχοντας τους πάντα το πιο πρόσφατο, από πλευράς περιεχομένου, αντικείμενο.

Εξαιτίας αυτής της στενής σχέσης είναι δυνατή η ανάπτυξη ενός μηχανισμού ενημέρωσης των αντικειμένων. Οι κόμβοι που έχουν αποθηκευμένα τα αντίγραφα του περιεχομένου ενός εξυπηρετητή εγγράφονται στην υπηρεσία που θα αναλάβει να τους ειδοποιήσει για πιθανές αλλαγές. Έτσι όταν μια αλλαγή προκύψει, ο εξυπηρετητής ενημερώνει την υπηρεσία για την αλλαγή και το αντικείμενο στο οποίο έγινε αυτή, και η υπηρεσία αναλαμβάνει να ενημερώσει όλους τους κόμβους που έχουν αποθηκευμένο το αντίγραφο του συγκεκριμένου αντικείμενου. Κάνοντας στη συνέχεια χρήση των γραμμών υψηλού εύρους ζώνης με τις οποίες είναι εξοπλισμένο το δίκτυο του CDN, γίνεται η παράδοση των νέων αντιγράφων σε όλους τους εμπλεκόμενους κόμβους.

2.6 Επιλογή Κατάλληλων Κόμβων για Φιλοξενία Αντικειμένων

Τα τελευταία χρόνια έχει γίνει αρκετή δουλειά πάνω σε τεχνικές και σχετικούς αλγορίθμους που αποσκοπούν στη αποδοτική ανακατεύθυνση των αιτήσεων των

χρηστών στους κατάλληλους εξυπηρετητές ενός CDN. Ωστόσο δεν έχει δοθεί αρκετή προσοχή στην ανάπτυξη τεχνικών που σκοπό έχουν την κατάλληλη τοποθέτηση των αντιγράφων των εξυπηρετητών (που εμπιστεύονται το περιεχόμενο τους στους CDNs) σε κομβικά σημεία του δικτύου. Μία κατάλληλη τοποθέτηση αντιγράφων είναι συμφέρουσα τόσο για τους ίδιους τους εξυπηρετητές αφού θα μειωθεί ο χρόνος απόκρισης των χρηστών τους αλλά και ο φόρτος εργασία τους, όσο και για τα δίκτυα παροχής περιεχομένου όπου θα παρατηρήσουν εμφανή μείωση της κατανάλωσης του πολύτιμου για αυτά, εύρους ζώνης.

Ας υποθέσουμε ότι ένας εξυπηρετητής επιθυμεί να βελτιώσει την απόδοση του (π.χ. να μειώσει τον χρόνο απόκρισης των αιτήσεων των χρηστών) με το να εμπιστευθεί την διαχείριση του περιεχομένου του σε ένα CDN. Το πρόβλημα τοποθέτησης αντιγράφων μπορεί να εκφραστεί ως εξής: πόσα αντίγραφα (M) πρέπει να τοποθετηθούν και σε ποιους κόμβους τους δικτύου (N όπου $N > M$), έτσι ώστε μία συνάρτηση κόστους που περιγράφει την απόδοση όλου του συστήματος να βελτιστοποιηθεί. (αυτή η συνάρτηση θα μπορούσε να εκφράζει τον μέσο χρόνο απόκρισης των χρηστών, την κατανάλωση εύρους ζώνης, κ.α.)

Η θεωρητική προσέγγιση του προβλήματος της επιλογής των κατάλληλων κόμβων αλλά και της εύρεσης του αριθμού αυτών με σκοπό την τοποθέτηση αντιγράφων, είναι γνωστή ως *facility location problem* [38] και περιγράφεται ως εξής:

Ας υποθέσουμε ότι έχουμε ένα σύνολο από κόμβους σε ένα δίκτυο και ότι η τοποθέτηση ενός αντιγράφου σε έναν από αυτούς, i , προκύπτει με ένα κόστος f_i . Μία κοινότητα χρηστών, j , ανατίθεται σε ένα κόμβο i από τον οποίο ανακτά τα αντικείμενα που ζητάει, με κόστος $d_{ij}c_{ij}$, όπου d_i είναι οι απαιτήσεις της κοινότητας χρηστών (για παράδειγμα η ποσότητα πληροφορίας όπου ζητείται) και c_{ij} είναι απόσταση μεταξύ του i και του j (για παράδειγμα αριθμός hops). Στόχος μας είναι να βρούμε τόσο τον αριθμό των κόμβων όσο και την θέση τους στο δίκτυο ώστε το συνολικό κόστος που περιλαμβάνει τους όρους $d_{ij}c_{ij}$ και f_i για όλα τα i, j να είναι το μικρότερο δυνατό.

Ένα παρόμοιο με το προηγούμενο πρόβλημα είναι το *minimum K-Median* [38]. Η διαφορά του με το προηγούμενο είναι ότι γνωρίζουμε από πριν τον αριθμό των κόμβων που θα φιλοξενήσουν τα αντικείμενα (αυτός είναι K) και το κόστος για την τοποθέτηση ενός αντικειμένου όπου στην προηγούμενη περίπτωση είναι f_i , σε αυτή την περίπτωση απουσιάζει.

2.6.1 Αλγόριθμοι Τοποθέτησης Αντιγράφων σε Δίκτυα Παροχής Περιεχομένου

Είναι προφανές ότι τα παραπάνω θεωρητικά προβλήματα βρίσκουν εφαρμογή σε CDNs όπου είναι επιθυμητό να βρεθούν τόσο ο αριθμός όσο και οι κόμβοι που θα φιλοξενήσουν τα αντίγραφα των αντικειμένων ενός εξυπηρετητή, δεδομένου των χαρακτηριστικών της ροής αιτήσεων των χρηστών προς τα αντικείμενα αυτά. Κατά καιρούς έχουν προταθεί αλγόριθμοι για επίλυση των προβλημάτων αυτών [46]. Στη συνέχεια θα παρουσιάσουμε μερικούς από αυτούς.

□ Αλγόριθμοι Βασισμένοι Σε Δέντρα

Η βασική παραδοχή, είναι η ύπαρξη δεντρικής τοπολογίας με ρίζα του δέντρου να είναι ο εξυπηρετητής, του οποίου τα περιεχόμενα πρέπει να τοποθετηθούν κατάλληλα.

Το πρόβλημα μοντελοποιείται σαν πρόβλημα δυναμικού προγραμματισμού [36]. Ας υποθέσουμε λοιπόν ότι έχουμε ένα δέντρο T . Αυτό διαιρείται σε πολλά μικρότερα υποδέντρα T_i . Ο καλύτερος τρόπος για να τοποθετηθούν $t > 1$ αντίγραφα του εξυπηρετητή (δηλ t αντίγραφα όλου του περιεχομένου του εξυπηρετητή) στο δέντρο T είναι να τοποθετηθούν t_i' με τον καλύτερο τρόπο σε κάθε υποδέντρο T_i , όπου το άθροισμα των t_i' είναι ίσο με t . Ο αλγόριθμος βρίσκει την βέλτιστη λύση με την προϋπόθεση δεντρικής τοπολογίας και ότι οι αιτήσεις των χρηστών ταξιδεύουν από τα φύλλα προς την ρίζα χωρίς να υπάρχει η δυνατότητα η αίτηση να εξυπηρετηθεί από γειτονικό κόμβο του ίδιου επιπέδου. Είναι δυνατό παρόλα αυτά, αυτές οι δύο προϋποθέσεις να αποκλείουν τις περιπτώσεις αποδοτικότερης τοποθέτησης, όπως έχουν δείξει σχετικές μελέτες [37]. Η πολυπλοκότητα του συγκεκριμένου αλγορίθμου είναι $O(N^3M^2)$.

□ Greedy Αλγόριθμος

Ο Greedy αλγόριθμος [37] δουλεύει ως εξής: ας υποθέσουμε ότι θέλουμε να τοποθετήσουμε M αντίγραφα του εξυπηρετητή ανάμεσα σε N πιθανούς κόμβους. Διαλέγουμε έναν κόμβο κάθε φορά. Στην πρώτη επανάληψη ελέγχονται όλοι οι κόμβοι για πιθανή καταλληλότητα τοποθέτησης αντιγράφου. Υπολογίζεται το συνολικό κόστος του συστήματος θεωρώντας ότι όλες οι αιτήσεις των χρηστών διοχετεύονται στον κόμβο

που επιλέγεται κάθε φορά. Στο τέλος της επανάληψης θα επιλέξουμε τον κόμβο που θα μας δώσει το μικρότερο συνολικό κόστος. Στην δεύτερη επανάληψη προσπαθούμε να επιλέξουμε τον δεύτερο κόμβο που πρόκειται να φιλοξενήσει αντίγραφο (δεδομένης φυσικά της ύπαρξης του πρώτου κόμβου) και επιλέγεται ο κόμβος με το συνολικό μικρότερο κόστος. Το συνολικό κόστος υπολογίζεται θεωρώντας ότι οι αιτήσεις των χρηστών διοχετεύονται στον κόμβο με αντίγραφο που είναι πιο κοντά σε αυτούς. Αυτή η διαδικασία επαναλαμβάνεται μέχρις ότου να βρεθούν οι M κόμβοι. Η πολυπλοκότητα του Greedy αλγορίθμου είναι $O(N^2M)$.

□ **Hot Spot Αλγόριθμος**

Η βασική ιδέα αυτού του αλγορίθμου [37] είναι η τοποθέτηση αντιγράφων κοντά σε κοινότητες χρηστών που προκαλούν αρκετή κίνηση (με άλλα λόγια ζητούν πολλά από τα αντικείμενα του εξυπηρετητή του οποίου αντίγραφο, θέλουμε να τοποθετήσουμε). Έτσι οι N κόμβοι κατανέμονται ανάλογα με την ποσότητα της κίνησης που γεννιέται στην γειτονιά τους. Τα αντίγραφα στη συνέχεια τοποθετούνται στους κόμβους των οποίων οι γειτονιές παράγουν την μεγαλύτερη κίνηση. Γειτονιά του κόμβου A ορίζεται ως ο κύκλος με μία συγκεκριμένη ακτίνα. Η ακτίνα μπορεί να έχει μία τιμή από 0 έως την μεγαλύτερη απόσταση μεταξύ δύο κόμβων (εκφραζόμενη σε hops). Η πολυπλοκότητα του Hot Spot αλγορίθμου είναι $O(N^2 + \min(N \log N, NM))$.

□ **Random Αλγόριθμος**

Όπως υποδηλώνει και το όνομα του, ο αλγόριθμος αυτός αποφασίζει με τυχαίο τρόπο. Έτσι επιλέγονται τυχαία οι κόμβοι που θα φιλοξενήσουν τα αντίγραφα. Η πολυπλοκότητα του Random αλγορίθμου είναι $O(NM)$.



Κεφάλαιο 3

Δομή Συστήματος για την Μελέτη Απόδοσης των Αντιπροσώπων Διαδικτύου

3.1 Στήσιμο Συστήματος για την Μελέτη της Απόδοσης

Με σκοπό την μελέτη της απόδοσης του συστήματος συνεργαζόμενων κρυφών αντιπροσώπων, αναπτύχθηκε ένα μοντέλο και έγιναν ορισμένες παραδοχές για χάρη απλότητας οι οποίες περιγράφονται παρακάτω.

3.1.1 Μοντέλο Διαδικτύου

Γενικά, ο χρόνος πρόσβασης ενός αντικειμένου μέσω διαδικτύου είναι η συνιστώσα καθυστερήσεων που οφείλονται σε αρκετά επιμέρους στοιχεία [23]. Ο συνολικός χρόνος περιέχει για παράδειγμα DNS χρόνους, καθυστερήσεις λόγω TCP πρωτοκόλλου, I/O χρόνους και καθυστερήσεις πρόσβασης των ενδιαμέσων συστημάτων αλλά και των εξυπηρετητών / κρυφών αντιπροσώπων. Επιπλέον, το διαδίκτυο αποτελείται από ένα σύνολο διαφορετικών συστημάτων εξυπηρετητών, δρομολογητών και γραμμών

μεταφοράς, με διαφορετικά χαρακτηριστικά απόδοσης. Αν λάβει επίσης κανείς την συνεχή ανάπτυξη του διαδικτύου το να μοντελοποιήσει κανείς όλο αυτό το σύστημα είναι αρκετά δύσκολο.

Οι περισσότεροι ερευνητές χρησιμοποιούν πληροφορία για τον χρόνο πρόσβασης των αντικειμένων η οποία περιέχεται στις ροές αιτήσεων αρκετά δραστήριων εξυπηρετητών. Μετά από σχετική μελέτη πραγματικών ροών αιτήσεων [17], παρατηρήσαμε ότι παρουσιάζεται αρκετά μεγάλη απόκλιση στο χρόνο πρόσβασης για ένα αντικείμενο, ακόμα και μετά από σχετικά σύντομο χρονικό διάστημα (κάτι το οποίο επισημαίνεται και στο [19]). Αυτό μπορεί να οδηγήσει σε λανθασμένες αποφάσεις όταν μελετάμε και συγκρίνουμε την απόδοση αλγορίθμων αντικατάστασης βασιζόμενοι στο χρόνο απόκρισης.

Μία πρόσφατη μελέτη εντοπίζει το μεγαλύτερο μέρος του χρόνου πρόσβασης να οφείλετε στο ίδιο το διαδίκτυο [15]. Έτσι εστιάζουμε στο επικοινωνιακό κομμάτι του χρόνου πρόσβασης θεωρώντας ότι υπάρχουν αρκετά καλά ρυθμισμένοι και αποδοτικοί εξυπηρετητές, κρυφοί αντιπρόσωποι, δρομολογητές, DNS όπως επίσης και αποδοτικό πρωτόκολλο μεταφοράς.

Για τους σκοπούς της μελέτης αυτής επιθυμούμε να χρησιμοποιήσουμε ένα απλό μοντέλο το οποίο να ικανοποιεί τις εξής απαιτήσεις: *i)* απεικονίζει τις αποκλίσεις στο χρόνο απόκρισης για αντικείμενα διαφορετικών μεγεθών, *ii)* είναι παραμετρικό και ευαίσθητο στο φορτίο στο Διαδίκτυο, *iii)* απεικονίζει το γεγονός ότι οι διαφορετικοί αλγόριθμοι αντικατάστασης κρυφών αντιπροσώπων ασκούν διαφορετικές επιδράσεις στο φόρτο του Διαδικτύου, λόγω των διαφορετικών hit ratios που επιτυγχάνουν, και *iv)* απεικονίζει τα διαφορετικά χαρακτηριστικά γραμμών μεταφοράς των μονοπατιών που ακολουθούνται κάτω από διαφορετικά σενάρια.

Πιστεύουμε ότι ένα μοντέλο το οποίο ικανοποιεί αυτές τις απαιτήσεις μπορεί να είναι αρκετά απλό, αφ' ενός, και επίσης αρκετά ισχυρό για να επιτρέψει την κατάλληλη αξιολόγηση των διαφορετικών αλγορίθμων αντικατάστασης όσον αφορά την απόδοση στον χρόνο απόκρισης. Έτσι χρησιμοποιήσαμε ένα μοντέλο βασισμένο σε M/M/1 ουρές αναμονής για τον υπολογισμό του χρόνου απόκρισης. Λεπτομέρειες σχετικά με αυτό παρουσιάζονται στο [32].

3.1.2 Δημιουργία Συνθετικών Ροών Αιτήσεων

Για την μελέτη της απόδοσης αρχικά χρησιμοποιήθηκαν πραγματικές ροές αιτήσεων όπως αυτές έχουν καταγραφεί από αρκετά δραστήριους εξυπηρετητές [17]. Ωστόσο με σκοπό την μελέτη επίδρασης διαφόρων παραμέτρων (όπως τα χαρακτηριστικά της κατανομής της δημοτικότητας των αντικειμένων) στη απόδοση του συστήματος, χρησιμοποιήθηκαν συνθετικές ροές αιτήσεων.

Οι συνθετικές ροές αιτήσεων δημιουργήθηκαν με το εργαλείο SURGE (Scalable URL Reference Generator [13]). Κάθε ροή περιέχει 140,000 αιτήσεις, αριθμός ο οποίος ανταποκρίνεται στον φόρτο εργασίας ενός δραστήριου εξυπηρετητή για διάστημα μίας ημέρας. Σε κάθε ροή υπάρχουν 35,000 αντικείμενα.

Χρησιμοποιήσαμε τις εξής κατανομές για τα μεγέθη των αντικειμένων, την δημοτικότητα και της ιδιότητας χρονικής περιοχής:

- i. *Μέγεθος Αντικειμένων*. Η κατανομή του μεγέθους των αντικειμένων έχει αρκετά μεγάλη ουρά, πράγμα που σημαίνει ότι έχουμε αρκετά μεγάλη διακύμανση. Το σώμα της κατανομής περιγράφεται από την *Lognormal* ενώ η ουρά από την *Pareto*.
- ii. *Δημοτικότητα*. Η κατανομή της δημοτικότητας των αντικειμένων ακολουθεί, όπως άλλωστε έχουν δείξει αρκετές μελέτες [14], την *Zipf* κατανομή με παράμετρο θ . Όταν η τιμή του θ πλησιάζει την μονάδα, η κατανομή γίνεται πιο απότομη (πράγμα που σημαίνει ότι έχουμε ελάχιστα, υπερβολικά δημοφιλή αντικείμενα), ενώ όταν πλησιάζει το 0 η κατανομή γίνεται ομοιόμορφη (με όλα τα αντικείμενα να έχουν την ίδια πιθανότητα ζήτησης). Σχετικές μελέτες δείχνουν ότι οι τιμές του θ στις πραγματικές ροές αιτήσεων κυμαίνονται από 0.6 έως 0.8 [14]. Ωστόσο άλλες μελέτες θεωρούν ότι η τιμή της παραμέτρου θ μπορεί να φτάσει και στο 1.0 ([12][13]).
- iii. *Ιδιότητας χρονικής περιοχής (temporal locality)*. Η κατανομή της ιδιότητας χρονικής περιοχής ακολουθεί την *lognormal*, όπως έχουν δείξει σχετικές μελέτες. Η ιδιότητα αυτή των ροών αιτήσεων στο διαδίκτυο περιγράφει πόσο πιθανόν είναι, ένα αντικείμενο που ζητείται τώρα να ξαναζητηθεί στο πολύ κοντινό μέλλον.

3.1.3 Μοντελοποίηση Συνεργασίας Κρυφών Αντιπροσώπων

Οι κρυφοί αντιπρόσωποι συνεργάζονται έτσι ώστε να βελτιωθεί η απόδοση του συστήματος στην περίπτωση όπου δεν είναι αποθηκευμένο ένα αντικείμενο τοπικά στον δίσκο. Έτσι το αντικείμενο αναζητείται πρώτα στους υπόλοιπους κρυφούς αντιπρόσωπους και στην συνέχεια, αν δεν βρεθεί ανακτάται από τον εξυπηρετητή.

Με σκοπό την ποσοτικοποίηση της βελτίωσης της απόδοσης παραμετροποιούμε το σύστημα ως εξής: Το ποσοστό των αντικειμένων τα οποία είναι κοινά σε όλες τις ροές είναι 50%. Η συνολική μάζα πιθανότητας των κοινών αντικειμένων είναι 60% (χρησιμοποιήθηκε επίσης και μάζα πιθανότητας ίση με 80%). Τέλος οι κρυφοί αντιπρόσωποι συνδέονται μεταξύ τους με γραμμές των οποίων το εύρος ζώνης είναι της τάξης του 2 MBps.

3.2 Αλγόριθμοι Αντικατάστασης και Σχετικοί Μηχανισμοί

Στην παρούσα μελέτη γίνεται σύγκριση διάφορων αλγορίθμων αντικατάστασης υπό την παρουσία περιβάλλοντος συνεργασίας. Οι αλγόριθμοι είναι: ο γνωστός LRU, ο Cost Size Popularity (CSP), ο GDS(1) και ο GDS(lat). Αναλυτική περιγραφή τους γίνει παρακάτω, αφού πρώτα παρουσιάσουμε ορισμένους βοηθητικούς μηχανισμούς.

Μοντέλο Πρόβλεψης Δημοτικότητας

Η δημοτικότητα ενός αντικειμένου (εφόσον δεν είναι γνωστή από πριν) μπορεί να υπολογιστεί έμμεσα, με την βοήθεια της εξής τεχνικής, η οποία έχει παρουσιασθεί σε σχετική βιβλιογραφία [18]: η δημοτικότητα του αντικειμένου i ορίζεται από τον λόγο $1/MTTR_i$ όπου $MTTR_i$ είναι ο μέσος χρόνος ζήτησης. Το $MTTR$ υπολογίζεται σταδιακά ώσπου να φτάσει στην πραγματική του τιμή. Έτσι το $MTTR$ την χρονική στιγμή t_1 είναι, $MTTR(t_1) = (1-a)(t_1 - t_0) + aMTTR(t_0)$ όπου t_0 είναι η πιο πρόσφατη χρονική στιγμή αίτησης του αντικειμένου και t_1 είναι η τωρινή χρονική στιγμή. Ο παράγοντας a μπορεί να επιλεγεί κατάλληλα έτσι ώστε να δίνει βάρος σε πρόσφατες χρονικές αφίξεις ή σε πιο παλιές. Στη μελέτη που ακολουθεί η τιμή του a είναι 0.5.

Μηχανισμός Ελέγχου Εισόδου

Βασικός στόχος του μηχανισμού ελέγχου εισόδου (Admission Control Policy) είναι να αποφασίζει ποια αντικείμενα πρέπει να αποθηκευτούν και ποια όχι. Είναι δυνατόν ένα μεγάλο σε μέγεθος και μη δημοφιλές αντικείμενο να αναγκάσει την διαγραφή πολλών αντικειμένων. Ο μηχανισμός ελέγχου εισόδου θα αποτρέψει την αποθήκευση του αφού ενδεχόμενη είσοδος του στο σύστημα δεν θα βελτιώσει την συνολική απόδοση. Παράλληλα με τον μηχανισμό εισόδου υλοποιείται και μία βοηθητική κρυφή μνήμη η οποία λειτουργεί σαν επιπρόσθετο φίλτρο και σκοπό έχει να βοηθήσει στον υπολογισμό του MTTR αλλά και να κρατάει στατιστικά στοιχεία για τα αντικείμενα.

Η λειτουργία λοιπόν αυτών των μηχανισμών περιγράφεται ως εξής: για κάθε αντικείμενο που ζητείται από το διαδίκτυο για πρώτη φορά, εισάγεται στην βοηθητική κρυφή μνήμη ενώ παράλληλα καταγράφονται στατιστικά στοιχεία. Στη δεύτερη αίτηση για το ίδιο αντικείμενο, υπολογίζεται το MTTR και καλείται ο μηχανισμός ελέγχου εισόδου για να αποφασίσει αν τελικά θα πρέπει να αποθηκευτεί ή όχι το αντικείμενο.

Αλγόριθμοι Αντικατάστασης

Οι αλγόριθμοι αντικατάστασης που μελετήθηκαν είναι οι εξής:

□ **LRU (Least Recently Used)**

Ένας από το πιο γνωστούς αλγόριθμους που έχει χρησιμοποιηθεί σε αρκετούς τομείς. Διαγράφει τα αντικείμενα αυτά τα οποία δεν έχουν χρησιμοποιηθεί για το μεγαλύτερο χρονικό διάστημα.

□ **CSP (Cost Size Popularity)**

Αυτός ο αλγόριθμος λαμβάνει υπόψη του το μέγεθος του αντικειμένου (Size), το κόστος ανάκτησης του (Cost, το οποίο εκφράζεται σε χρόνο), και την δημοτικότητα του ($1/\text{MTTR}$). Για κάθε αντικείμενο i υπολογίζεται το κέρδος αποθήκευσης του CG_i , το οποίο είναι μια τιμή η οποία χαρακτηρίζει πόσο ωφέλιμη είναι η διατήρηση ενός αντικειμένου στον τοπικό δίσκο, και δίνεται από την σχέση:

$$\text{CG}_i = \frac{\text{Cost}_i}{\text{Size}_i \cdot \text{MTTR}_i} \quad \text{Σχέση 3-1}$$

Τα αντικείμενα κατατάσσονται με βάση την τιμή του κέρδους αποθήκευσης τους. Αν ο μηχανισμός ελέγχου εισόδου το επιτρέψει, διαγράφονται τα αντικείμενα με το μικρότερο κέρδος, μέχρι να απελευθερωθεί ο απαραίτητος χώρος.

Με σκοπό την μελέτη της επίδρασης του μεγέθους του αντικειμένου και της δημοτικότητάς του στην απόδοση του αλγορίθμου εξετάστηκαν και ορισμένες παραλλαγές του αλγορίθμου CSP [32].

□ **GDS (Greedy Dual Size)**

Κάθε αντικείμενο περιγράφεται από μία τιμή (όπως και στον CSP) η οποία δίνεται από την σχέση $H = c_i / \text{size}_i$. Όπου c_i είναι το κόστος ανάκτησης του αντικειμένου από τον εξυπηρετητή (το οποίο μπορεί να εκφραστεί σε χρόνο, hops ή ακόμα μπορεί να θεωρηθεί ίσο με την μονάδα όταν στόχος είναι η αύξηση του hit ratio) και size_i είναι το μέγεθος του αντικειμένου. Κατά την διαγραφή αντικειμένων επιλέγονται πρώτα αυτά με την μικρότερη τιμή. Στη συνέχεια η τιμή H όλων των αντικειμένων που βρίσκονται στον δίσκο, μειώνεται κατά την τιμή H του αντικειμένου που διαγράφηκε. Όταν προκύψει αίτηση για το αντικείμενο i που βρίσκεται στον δίσκο, η τιμή του (H_i) ανανεώνεται στην αρχική του τιμή (δηλαδή την τιμή που είχε την πρώτη φορά που αποθηκεύτηκε στον δίσκο). Έτσι δημοφιλή αντικείμενα τείνουν να έχουν υψηλές τιμές σε σύγκριση με όχι και τόσο δημοφιλή αντικείμενα. Εξετάσαμε δύο μορφές του αλγορίθμου: τον GDS(1) ο οποίος προσπαθεί να μεγιστοποιήσει το hit ratio (το c_i στον παραπάνω τύπο είναι ίσο με τη μονάδα) και τον GDS(lat) που προσπαθεί να μειώσει το μέσο χρόνο απόκρισης (το c_i στον παραπάνω τύπο εκφράζει χρόνο ανάκτησης αντικειμένου).

3.3 Μετρικές Απόδοσης

Οι μετρικές απόδοσης του συστήματος είναι οι ακόλουθες:

- i. Το *Hit Ratio* το οποίο ορίζεται σαν τον λόγο των αιτήσεων που εξυπηρετήθηκαν από τον κρυφό αντιπρόσωπο προς το σύνολο των αιτήσεων.
- ii. Ο *χρόνος απόκρισης (latency)* το οποίο μετράται σε milliseconds. Εφόσον μας ενδιαφέρει η απόδοση του κρυφού αντιπροσώπου, ο χρόνος απόκρισης δεν

περιέχει την καθυστέρηση μετάδοσης του αντικειμένου, από τον κρυφό αντιπρόσωπο στον χρήστη.

iii. Οι απαιτήσεις σε εύρος ζώνης (network bandwidth requirements), όπου εκφράζονται με την ποσότητα πληροφορίας που διακινείται στις γραμμές μεταφοράς.

Σκοπός της μελέτης μας πάνω στην απόδοση των κρυφών αντιπροσώπων είναι να δώσει απάντηση στα παρακάτω ερωτήματα, στηριζόμενη σε ένα μεγάλο αριθμό προσομοιώσεων.

- i. Ποια είναι η απόδοση του CSP σε σχέση με τον γνωστό και ευρύτατα χρησιμοποιήσιμο LRU; Πιο συγκεκριμένα, πως το μέγεθος του αντικειμένου, το κόστος ανάκτησης του, και η δημοτικότητα του επηρεάζουν την απόδοση;
- ii. Τι συμπεράσματα μπορούμε να βγάλουμε στηριζόμενοι στις μετρικές που προαναφέρθηκαν και κατά πόσο αυτά τα συμπεράσματα ανταποκρίνονται στην πραγματικότητα;
- iii. Πως μεταβάλλεται η απόδοση του συστήματος αλλάζοντα παραμέτρους όπως το μέγεθος του δίσκου, το αριθμό των συνεργαζόμενων αντιπροσώπων κ.α.

Οι απαντήσεις στα ερωτήματα αυτά θα μας δώσουν την δυνατότητα να συνοψίσουμε την απόδοση των κρυφών αντιπροσώπων, δείχνοντας ποιος αλγόριθμος αντικατάστασης είναι ο προτιμότερος κάτω από διαφορετικές τιμές των παραμέτρων και χαρακτηριστικών του συστήματος και διαφορετικές μετρικές απόδοσης.

Κεφάλαιο 4

Μοντέλο Συστήματος για την Τοποθέτηση Κρυφών Αντιπροσώπων στο Διαδίκτυο

Στην ενότητα 2.4 έγινε λόγος για την τοποθέτηση αντιγράφων του περιεχομένου του εξυπηρετητή σε ένα δίκτυο [33],[35],[48](π.χ. το δίκτυο ενός CDN). Μία επέκταση αυτού του προβλήματος είναι η τοποθέτηση κρυφών αντιπροσώπων στο διαδίκτυο [43]. Η διαφορά του από το προηγούμενο πρόβλημα είναι ότι τα περιεχόμενα του κρυφού αντιπροσώπου τα καθορίζουν οι ροές αιτήσεων των χρηστών πράγμα που σημαίνει ότι δεν έχουμε πλήρη αντίγραφα των εξυπηρετητών, αλλά ορισμένα από τα αντικείμενα αυτών. Επίσης στο προηγούμενο πρόβλημα αναφερθήκαμε σε πλήρη αντίγραφα των εξυπηρετητών πράγμα που σημαίνει ότι μία αίτηση είναι σίγουρο ότι θα εξυπηρετηθεί από έναν κόμβο. Σε αυτή την περίπτωση όμως ενδέχεται ένα αντικείμενο που ζητείται από τους χρήστες να μην υπάρχει στον κρυφό αντιπρόσωπο οπότε, όπως θα παρουσιάσουμε παρακάτω, θα ζητηθεί από τον αμέσως κοντινότερο αντιπρόσωπο όπου ενδέχεται να μπορεί να εξυπηρετήσει τον χρήστη. Εισάγεται λοιπόν η έννοια των συνεργαζόμενων κρυφών αντιπροσώπων.

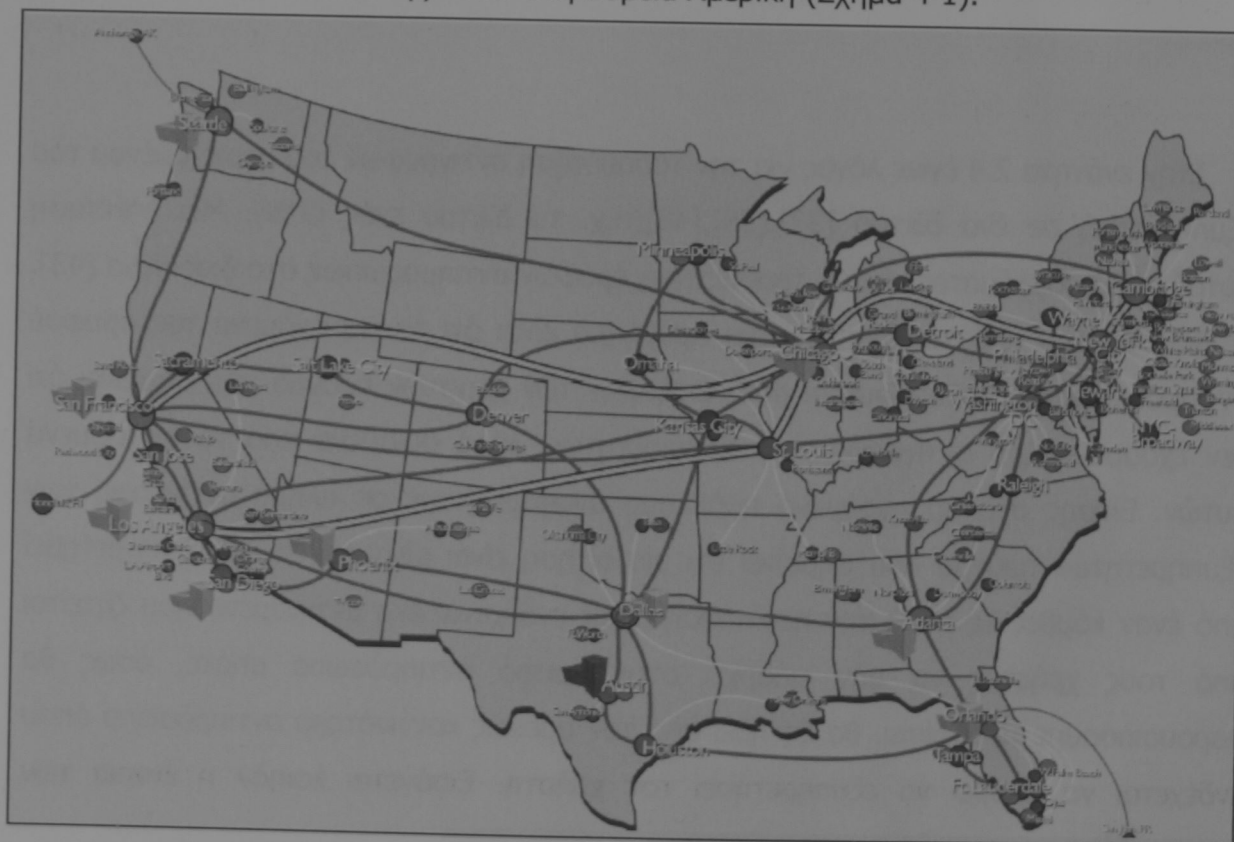
Στο τμήμα που ακολουθεί θα αναφερθούμε στην μοντελοποίηση και στην περιγραφή του συστήματος όπου προσπαθούμε να τοποθετήσουμε αποδοτικά κρυφούς

αντιπροσώπους. Ωστόσο, μιας και αυτό το πρόβλημα είναι επέκταση του προβλήματος τοποθέτησης πλήρων αντιγράφων, θα αφήσουμε για το αργότερα τις αλλαγές/παραδοχές που μπορούμε να κάνουμε έτσι ώστε να προσαρμόσουμε το παρακάτω μοντέλο στο πρόβλημα τοποθέτησης πλήρων αντιγράφων των περιεχομένων του εξυπηρετητή.

4.1 Μοντελοποίηση Συστήματος

4.1.1 Μοντέλο Δικτύου

Το δίκτυο πάνω στο οποίο θα γίνει η κατάλληλη τοποθέτηση των κρυφών αντιπροσώπων μπορεί να θεωρηθεί ότι είναι ένας γράφος με κόμβους οι οποίοι συνδέονται μεταξύ τους με ακμές. Ο γράφος αυτός αντιπροσωπεύει την ραχοκοκαλιά του δικτύου. Ένας τέτοιος γράφος φαίνεται στο Σχήμα 4-2, όπου αντιπροσωπεύει την ραχοκοκαλιά του δικτύου της AT&T στη Βόρεια Αμερική (Σχήμα 4-1).



Σχήμα 4-1. Η ραχοκοκαλιά του δικτύου της AT&T στη Βόρεια Αμερική.

Figure 1 is a network diagram showing 18 numbered nodes (1-18) connected by lines representing distances. Node 1 is highlighted with a thick border. Distances are labeled on the edges. A path is highlighted with thick arrows: 10 to 18 (solid), 18 to 1 (solid), and 1 to 10 (dashed).

Distances between nodes (from the diagram):

- 1-2: 4.97
- 1-8: 3.97
- 1-11: 8.94
- 1-18: 14.57
- 2-8: 4.97
- 5-8: 3.97
- 5-13: 6.18
- 5-9: 19.88
- 5-18: 35.78
- 6-16: 23.85
- 6-18: 16.78
- 7-12: 28.82
- 7-10: 2.98
- 8-11: 18.88
- 9-14: 2.19
- 9-15: 7.95
- 9-16: 39.57
- 9-18: 19.78
- 10-12: 2.99
- 10-17: 21.87
- 10-18: 19.34
- 11-18: 17.62
- 12-18: 3.98
- 13-14: 6.18
- 14-15: 43.13
- 15-16: 40.75
- 15-17: 14.91
- 16-17: 39.75
- 16-18: 6.88
- 17-18: 19.34
- 18-1: 14.57

Όπως φαίνεται στο Σχήμα 4-2, οι ακμές έχουν βάρη. Αυτά τα βάρη αντιπροσωπεύουν τον χρόνο (σε msec) που απαιτείται για να ταξιδέψει ένα πακέτο πληροφορίας από τον

ένα κόμβο στον άλλο. Τα βάρη λοιπόν των ακμών δείχνουν πόσο φορτωμένη (αργή) είναι η σύνδεση μεταξύ δύο κόμβων της ραχοκοκαλιάς του δικτύου. Ο χρόνος αυτός εξαρτάται από την γεωγραφική απόσταση των κόμβων, την καθυστέρηση στους δρομολογητές, την καθυστέρηση στην επεξεργασία των πακέτων κ.α. Τα συγκεκριμένα βάρη των ακμών που παρουσιάζονται στο Σχήμα 4-2 ανταποκρίνονται σε πραγματικές τιμές οι οποίες έχουν συλλεχθεί από τον δικτυακό τόπο της AT&T που είναι αφιερωμένος στην παρουσίαση στατιστικών απόδοσης του δικτύου της [44].

Δεν είναι μόνο οι ακμές οι οποίες έχουν βάρη, αλλά και οι κόμβοι. Το βάρος ενός κόμβου αντιπροσωπεύει την ποσότητα πληροφορίας που ζητούν από τον εξυπηρετητή, όλοι οι χρήστες που είναι συνδεδεμένοι στον κόμβο αυτό. Ας υποθέσουμε λοιπόν ότι υπάρχει ένας εξυπηρετητής και ότι διαθέτουμε ένα αρχείο ροής αιτήσεων προς αυτόν. Για να υπολογιστούν τα βάρη στους κόμβους ακολουθείται η εξής διαδικασία: αρχικά ομαδοποιούνται οι χρήστες σε ομάδες χρηστών που βρίσκονται γεωγραφικά κοντά, με βάση το IP τους (όπου αυτό υπάρχει στο αρχείο ροής αιτήσεων) και σχετική πληροφορία από BGP Routing Tables [34]. Τις N μεγαλύτερες ομάδες τις αναθέτουμε με τυχαίο τρόπο στους N κόμβους (όπου N είναι ο αριθμός των κόμβων του γράφου, στην περίπτωση του Σχήμα 4-2 το N είναι ίσο με 18). Στη συνέχεια υπολογίζουμε την ποσότητα πληροφορίας (σε MByte ή GByte) που ζητάει από τον εξυπηρετητή κάθε ομάδα και αυτή η τιμή αποτελεί το βάρος κάθε κόμβου.

Είναι καμιά φορά δυνατό, το αρχείο ροής αιτήσεων να μην περιέχει πληροφορία σχετικά με την ταυτότητα (IP) του χρήστη που ζητάει κάποιο αντικείμενο (ορισμένοι εξυπηρετητές προσπαθώντας να προστατέψουν την ταυτότητα των χρηστών τους δεν περικλείουν στα αρχεία ροής αιτήσεων τους, το IP των χρηστών). Επίσης έχοντας υπόψη μας την σχετικά ακριβή υπολογιστικά διαδικασία γεωγραφικής ομαδοποίησης των χρηστών μπορούμε να ακολουθήσουμε μια εναλλακτική διαδικασία που αποσκοπεί στον υπολογισμό των βαρών. Έτσι το αρχείο ροής αιτήσεων χωρίζεται σε N ίσα μέρη, οι N διαφορετικές ροές (υποροές) αντιστοιχίζονται στους N κόμβους του γράφου και στη συνέχεια υπολογίζεται η ποσότητα πληροφορίας που ζητείται από κάθε ροή.

Ο αλγόριθμος τοποθέτησης θα χρησιμοποιήσει τα βάρη των ακμών, τα βάρη των κόμβων και την τοπολογία του δικτύου, για να αποφασίσει που και πόσοι κρυφοί αντιπρόσωποι πρέπει να τοποθετηθούν ώστε να βελτιωθεί η τιμή μιας συνάρτησης που περιγράφει την απόδοση του συστήματος.

4.1.2 Εκτιμώμενη Απόδοση Κρυφών Αντιπροσώπων

Εκτός από τα βάρη των ακμών, των κόμβων και την τοπολογία δικτύου, σημαντικό ρόλο παίζει επίσης και η απόδοση του κρυφού αντιπροσώπου. Η μετρική απόδοσης που επιλέχθηκε είναι το Byte Hit Ratio (ορίζεται ως το ποσό πληροφορίας που εξυπηρετήθηκε από τον κρυφό αντιπρόσωπο προς το συνολικό ποσό πληροφορίας που ζητήθηκε) το οποίο εξαρτάται από τα χαρακτηριστικά της ροής αιτήσεων, από το μέγεθος του δίσκου του κρυφού αντιπροσώπου και από τον αλγόριθμο αντικατάστασης του.

Η εκτίμηση του BHR (Byte Hit Ratio) ενός κρυφού αντιπροσώπου στην περίπτωση που θα τοποθετηθεί σε ένα συγκεκριμένο κόμβο, βασίζεται πάνω σε ένα πίνακα απόδοσης ο οποίος συνοψίζει την απόδοση των κρυφών αντιπροσώπων (όπως ο Πίνακας 5-1 που παρουσιάζεται στην ενότητα 5.2) και έχει προκύψει μετά από μια σειρά πειραμάτων κάτω από διαφορετικές συνθήκες περιβάλλοντος. Πιο συγκεκριμένα, υπολογίστηκε η απόδοση ενός κρυφού αντιπροσώπου κάτω από διαφορετικά μεγέθη αποθηκευτικού χώρου, διαφορετικές τιμές της παραμέτρου θ της zipf κατανομής [13],[14],[35] της δημοτικότητας των αντικειμένων, και διαφορετικούς αλγορίθμους αντικατάστασης (GDS(1)[19], GDS(lat)[19], LRU, και CSP(ενότητα 3.2)). Έτσι για κάθε συνδυασμό των τιμών του αποθηκευτικού χώρου και του θ , γνωρίζουμε ποιος είναι ο ποιο αποδοτικός αλγόριθμος με βάση τις μετρικές απόδοσης hit ratio, χρόνος απόκρισης και απαιτήσεις σε εύρος ζώνης, αλλά και την απόδοσή του εκφραζόμενη σε BHR.

Η μεθοδολογία λοιπόν που ακολουθείται για τον υπολογισμό του BHR είναι η εξής. Αρχικά υπολογίζονται για κάθε ομάδα χρηστών τα χαρακτηριστικά της ροής αιτήσεων τους. Δηλαδή η τιμή της παραμέτρου θ αλλά και τον διαθέσιμο αποθηκευτικό χώρο.

Ο υπολογισμός της τιμής της παραμέτρου θ που αντιστοιχεί στην κάθε ροή αιτήσεων γίνεται με την μέθοδο χ^2 . Ποιο συγκεκριμένα: έστω ότι μία ροή αιτήσεων περιέχει ένα αριθμό αιτήσεων προς k αντικείμενα. Τα αντικείμενα κατατάσσονται από τα πιο δημοφιλή στα λιγότερο δημοφιλή, με βάση τον αριθμό αιτήσεων προς αυτά. Έτσι αν το αντικείμενο i είναι το πιο δημοφιλές, τότε κατατάσσεται στην θέση $x_i=1$ και ο αριθμός αιτήσεων προς το αντικείμενο i είναι y_i . Ο αντίστοιχος αριθμός αιτήσεων που προκύπτει για το συγκεκριμένο αντικείμενο i εφαρμόζοντας την zipf κατανομή, είναι $f(x_i)$. Έτσι, για τις τιμές του θ από 0 έως 1, με βήμα 0.1 υπολογίζουμε το άθροισμα:

$$\chi^2 = \sum_{i=1}^k \frac{(y_i - f(x_i))^2}{f(x_i)} \quad \text{Σχέση 4-1}$$

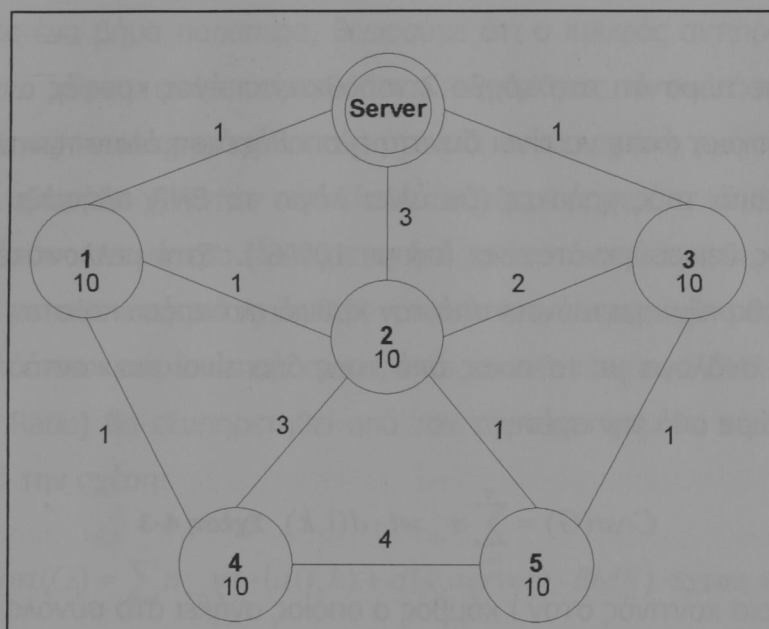
Η τιμή του θ που μας δίνει την μικρότερη τιμή του αθροίσματος, περιγράφει καλύτερα την ροή αιτήσεων.

Ο διαθέσιμος αποθηκευτικός χώρος εκφράζεται σαν το ποσοστό του μέγιστου χώρου που απαιτείται για την αποθήκευση όλων των αντικειμένων που ζητούν οι χρήστες. (Με άλλα λόγια αν θεωρήσουμε ότι όλοι οι κρυφοί αντιπρόσωποι θα έχουν δίσκους μεγέθους 1 GB και το μέγεθος όλων μαζί των αντικειμένων που ζητούνται από την ομάδα χρηστών του κόμβου k είναι 2 GB τότε το ποσοστό του διαθέσιμου αποθηκευτικού χώρου του κόμβου k είναι 50%). Στη συνέχεια συμβουλευόμαστε τον πίνακα απόδοσης (όπως ο Πίνακας 5-1) που αναφέρθηκε πιο πάνω έτσι ώστε να βρούμε το BHR του συγκεκριμένου αντιπροσώπου καθώς και τον αλγόριθμο που πρόκειται να χρησιμοποιηθεί έτσι ώστε να έχουμε την καλύτερη δυνατή απόδοση.

4.1.3 Μοντέλο Απόδοσης Συστήματος

Η μοντελοποίηση της απόδοσης του συστήματος και ο υπολογισμός της είναι βασική στην λύση του προβλήματος της τοποθέτησης κρυφών αντιπροσώπων στο δίκτυο μιας και ο σχετικός αλγόριθμος θα τοποθετήσει τους κρυφούς αντιπροσώπους με τέτοιο τρόπο ώστε να βελτιωθεί όσο το δυνατό περισσότερο η απόδοση του συστήματος.

Ας υποθέσουμε αρχικά, ότι δεν υπάρχει κανένας κρυφός αντιπρόσωπος στο δίκτυο παρά μόνο ο εξυπηρετητής. Έστω n_{wt_i} το βάρος στον κόμβο i (στα παρακάτω σχήματα έχει οριστεί ίσο με 10 για όλους τους κόμβους), $e_{wt_{ij}}$ το βάρος της ακμής που συνδέει τους κόμβους i και j , $d(i,j)$ η μικρότερη απόσταση μεταξύ των κόμβων i και j , N ο αριθμός των κόμβων (στο Σχήμα 4-3 το N είναι ίσο με 6), και $Cost(G)$ το συνολικό κόστος του γράφου G που περιγράφει την απόδοση του συστήματος. Η μικρότερη απόσταση $d(i,j)$ ορίζεται να είναι το μικρότερο άθροισμα των βαρών του μονοπατιού μεταξύ i και j που προκύπτει αφού πρώτα εξεταστούν όλα τα δυνατά μονοπάτια. Ας σημειωθεί ότι $d(i,i)=0$. Έτσι στο Σχήμα 4-3 $d(1,5) = 3$ το οποίο προκύπτει από το μονοπάτι $① \Rightarrow ② \Rightarrow ⑤$. Όλα τα άλλα μονοπάτια δίνουν μεγαλύτερο άθροισμα. Στο Σχήμα 4-4 φαίνονται τα μονοπάτια από κάθε κόμβο προς τον εξυπηρετητή τα οποία δίνουν την μικρότερη απόσταση, καθώς επίσης και η τιμή της απόστασης.



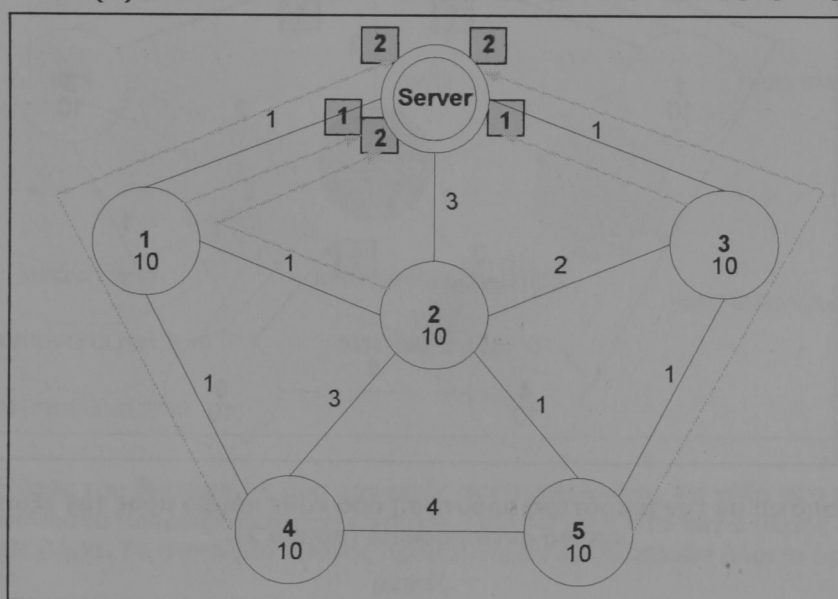
Σχήμα 4-3. Γράφος με βάρη στις ακμές και στους κόμβους. Σε όλους τους κόμβους τα βάρη είναι ίσα με 10.

Το συνολικό λοιπόν κόστος του γράφου δίνεται από την παρακάτω σχέση:

$$Cost(G) = \sum_{i=1}^n n_wt \cdot d(i, server) \quad \text{Σχέση 4-2}$$

Έτσι το συνολικό κόστος του γράφου στο Σχήμα 4-3 είναι ίσο με :

$$^1 Cost(G) = 10 \cdot 1 + 10 \cdot 2 + 10 \cdot 1 + 10 \cdot 2 + 10 \cdot 2 + 10 \cdot 0 = 80$$



Σχήμα 4-4. Μονοπάτια με την μικρότερη απόσταση από κάθε κόμβο προς τον εξυπηρετητή. Στις άκρες των τόξων φαίνεται και η απόσταση.

¹ Το τελευταίο γινόμενο αντιστοιχεί στην περίπτωση όπου ο υπό εξέταση κόμβος είναι ο server οπότε προκύπτει $d(server, server) = 0$

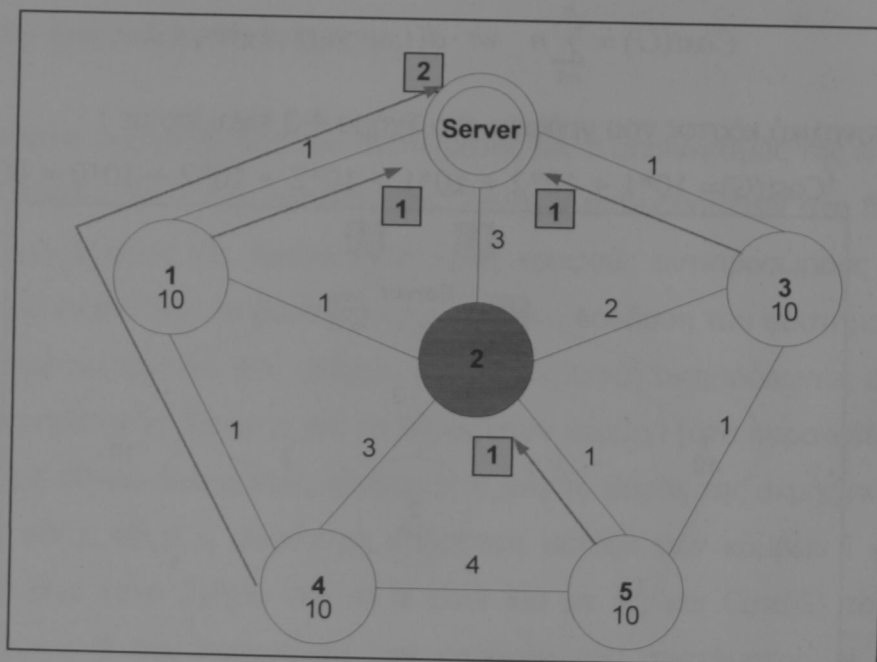
Ας υποθέσουμε τώρα ότι στο κόμβο 2 τοποθετείται ένας κρυφός αντιπρόσωπος με τέτοιο μέγεθος δίσκου, ώστε να είναι δυνατή η αποθήκευση όλων των αντικειμένων τα οποία ζητούνται από τους χρήστες (με άλλα λόγια το BHR πλησιάζει το 100%. Για λόγους απλότητας θεωρούμε ότι είναι ίσο με 100%²). Έτσι μελλοντικές αιτήσεις στα αντικείμενα αυτά θα εξυπηρετούνται από τον κρυφό αντιπρόσωπο στον κόμβο 2 ή από τον εξυπηρετητή ανάλογα με το ποιος από τους δύο είναι πιο κοντά. Το κόστος του γράφου δίνεται τώρα από την σχέση:

$$Cost(G) = \sum_{i=1}^n n_{wt} \cdot d(i, k) \quad \text{Σχέση 4-3}$$

όπου k είναι ο πιο κοντινός στον i κόμβος ο οποίος ανήκει στο σύνολο T των κρυφών αντιπροσώπων, συμπεριλαμβανομένου και του εξυπηρετητή.

Έτσι όπως φαίνεται στο Σχήμα 4-5 το κόστος του γράφου είναι :

$$Cost(G) = 10 \cdot 1 + 10 \cdot 0 + 10 \cdot 1 + 10 \cdot 2 + 10 \cdot 1 = 50$$



Σχήμα 4-5. Μονοπάτια με την μικρότερη απόσταση από κάθε κόμβο προς τον εξυπηρετητή ή τον κρυφό αντιπρόσωπο (κόμβο 2).

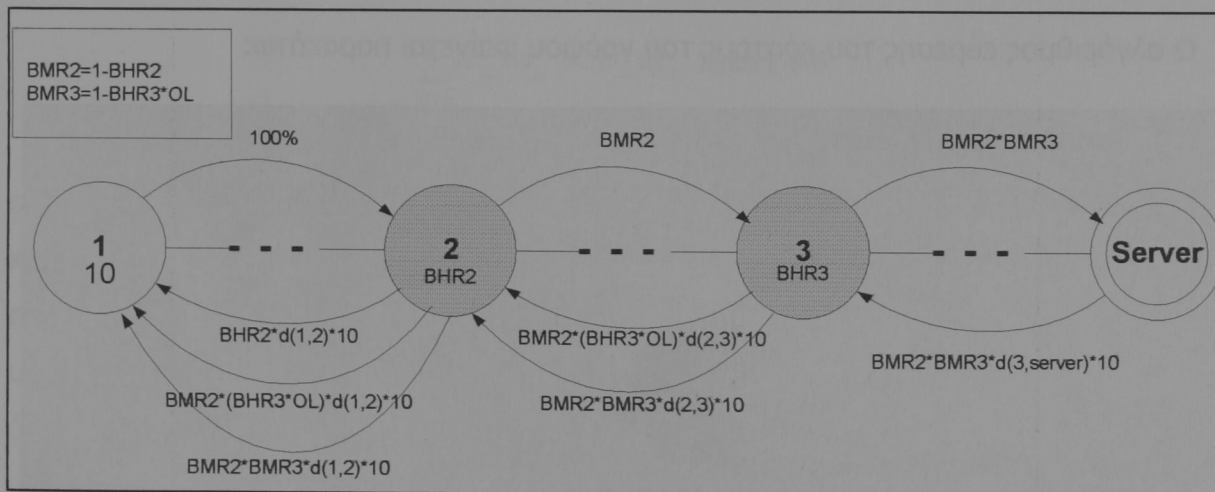
² Η περίπτωση ύπαρξης κρυφού αντιπροσώπου με BHR ίσο με 100% είναι παρόμοια με την ύπαρξη εξυπηρετητή που φιλοξενεί πλήρες αντίγραφο. Οπότε στην περίπτωση τοποθέτησης αντιγράφων ο υπολογισμός του κόστους γίνεται σύμφωνα με την Σχέση 4-3.

Προχωρώντας ένα βήμα παραπέρα, θεωρούμε ότι ο κρυφός αντιπρόσωπος δεν έχει αρκετό ελεύθερο χώρο, οπότε εφαρμόζεται αλγόριθμος αντικατάστασης και κατά συνέπεια η απόδοση του περιγράφεται τώρα με μία συγκεκριμένη τιμή του $BHR < 100\%$. Ομοίως με πριν, η ομάδα χρηστών που είναι συνδεδεμένη σε ένα κόμβο θα ζητήσει τα αντικείμενα από τον κοντινότερο εξυπηρετητή ή κρυφό αντιπρόσωπο. Η μόνη διαφορά με πριν είναι ότι με πιθανότητα ίση με BHR η ποσότητα πληροφορίας που θα ζητηθεί θα εξυπηρετηθεί από τον κρυφό αντιπρόσωπο, ενώ με πιθανότητα $1-BHR$ (όπου $1-BHR = BMR$, Byte Miss Ratio) θα εξυπηρετηθεί από τον εξυπηρετητή. Το κόστος του γράφου δίνεται τώρα από την σχέση:

$$Cost(G) = \sum_{i=1}^n n_wt \cdot (d(i,k) + d(k,server) \cdot BMR) \quad \text{Σχέση 4-4}$$

Όπου k είναι ο κοντινότερος στον i κρυφός αντιπρόσωπος ή ο εξυπηρετητής. Στην περίπτωση όπου ο k είναι ο $server$ τότε το δεύτερο μέλος στην παρένθεση είναι μηδέν αφού εξορισμού $d(i,i) = 0$. Έτσι όπως φαίνεται στο Σχήμα 4-5 το κόστος του γράφου, εάν υποθέσουμε ότι το BHR στον κρυφό αντιπρόσωπο 2 είναι 50%, είναι :

$$Cost(G) = 10 \cdot 1 + 10 \cdot 0 + 10 \cdot 1 + 10 \cdot 2 + 10 \cdot (1 + 0.5 \cdot 2) = 60$$



Σχήμα 4-6. Μέρος του δικτύου με δύο κρυφούς αντιπροσώπους. Τα τόξα στο πάνω μέρος δείχνουν το ποσοστό πληροφορίας που ζητείται, ενώ τα τόξα στο κάτω μέρος δείχνουν το κόστος σε κάθε βήμα. Το συνολικό κόστος προκύπτει αν προστεθούν όλοι οι όροι στο κάτω μέρος.

Η σχέση υπολογισμού του κόστους στον γράφο, φτάνει στην τελική της μορφή αν θεωρήσουμε ότι τα αντικείμενα που δεν εξυπηρετούνται από ένα κρυφό αντιπρόσωπο είναι δυνατό να εξυπηρετηθούν από ένα άλλο με μία συγκεκριμένη πιθανότητα.

Εισάγουμε λοιπόν την έννοια των συνεργαζόμενων κρυφών αντιπροσώπων στο σύστημα μας (το ποσοστό των αντικειμένων που ζητείται από διαφορετικές γεωγραφικά κατανομημένες ομάδες χρηστών, είναι δυνατό να ανέρθει σε αρκετά μεγάλα ποσοστά της τάξεως του 50% ή 60% σύμφωνα με σχετική μελέτη [47], πράγμα το οποίο οδηγεί στην ύπαρξη κοινού περιεχόμενο μεταξύ των κρυφών αντιπροσώπων). Για να γίνει πιο κατανοητή η παραπάνω θεώρηση, ας υποθέσουμε ότι στον γράφο υπάρχουν 2 κρυφοί αντιπρόσωποι, ένας κόμβος και ο εξυπηρετητής, όπως φαίνεται στο Σχήμα 4-6. Κάθε κρυφός αντιπρόσωπος έχει αποθηκευμένη μια ποσότητα πληροφορίας D . Θεωρούμε ότι ένα ποσοστό (ας το ονομάσουμε OL , OverLapping) της ποσότητας D είναι κοινή μεταξύ των κρυφών αντιπροσώπων. Έτσι παρουσιάζεται μία επικάλυψη της αποθηκευμένης ποσότητα πληροφορίας, η οποία είναι ίση με $OL * D$. Η υπόλοιπη ποσότητα πληροφορίας (η οποία δεν είναι κοινή μεταξύ των κρυφών αντιπροσώπων) θεωρούμε ότι είναι δυνατό να εξυπηρετήσει τις αιτήσεις των χρηστών οι οποίες δεν ικανοποιήθηκαν από ένα κρυφό αντιπρόσωπο. Έτσι όπως φαίνεται στο Σχήμα 4-6, η ποσότητα πληροφορίας που δεν εξυπηρετήθηκε από τον κρυφό αντιπρόσωπο 2 (η οποία είναι $MR_2 * 10$) εξυπηρετείται από τον κρυφό αντιπρόσωπο με πιθανότητα $OL * BHR_3$.

Ο αλγόριθμος εύρεσης του κόστους του γράφου φαίνεται παρακάτω:

1. Για κάθε κόμβο i ξεκινώντας από $i = 1$.
2. Βρες τον κοντινότερο κρυφό αντιπρόσωπο. Έστω ότι αυτός είναι ο k
3. $Cost(G) = n_wt_i * d(i, k)$
4. Βρες τον κοντινότερο κρυφό αντιπρόσωπο του k . Έστω ότι αυτός είναι ο j
5. $Cost(G) = Cost(G) + n_wt_i * d(k, j) * BMR_k$, όπου $BMR_k = 1 - BHR_k * OL$
6. $k = j$
7. Πήγαινε στο βήμα 4, εκτός και εάν $k = server$
8. $i = i + 1$
9. Πήγαινε στο βήμα 1 εκτός και εάν $i = N$

Έχοντας λοιπόν κατά νου το αλγόριθμο υπολογισμού του κόστους του γράφου, ο αλγόριθμος αντικατάστασης προσπαθεί να τοποθετήσει τους κρυφούς αντιπροσώπους στα σημεία εκείνα τα οποία θα οδηγήσουν στη μικρότερη τιμή του κόστους.

Αξιζει να παρατηρήσουμε ότι η απόσταση μεταξύ δύο κόμβων $d(i,j)$ δεν είναι απαραίτητο να εκφράζει τον χρόνο μετάδοσης ενός πακέτου πληροφορίας από τον ένα κόμβο στον άλλο. Θα μπορούσε να εκφράζει για παράδειγμα τον αριθμό των κόμβων στο μονοπάτι από το i στον j .

Είναι δυνατό στον γράφο να υπάρχουν περισσότεροι του ενός εξυπηρετητές, έστω s , με διαφορετικό περιεχόμενο (όπως συμβαίνει άλλωστε και στα πραγματικά δίκτυα παροχής περιεχομένου (CDN)). Σε αυτή την περίπτωση, υπολογίζουμε s διαφορετικά κόστη (ένα για κάθε εξυπηρετητή) και στη συνέχεια το άθροισμα τους θα μας δώσει το συνολικό κόστος του γράφου, όπως φαίνεται στην παρακάτω σχέση:

$$Cost(G) = \sum_{j \in S} Cost_j(G) \quad \text{Σχέση 4-5}$$

όπου το $Cost_j(G)$ υπολογίζεται με την βοήθεια του αλγορίθμου που παρουσιάζεται πιο πάνω για κάθε εξυπηρετητή j που ανήκει στο σύνολο S των εξυπηρετητών. Αξιζει επίσης να σημειώσουμε ότι ένα ποσοστό του βάρους n_wt_i κάθε κόμβου i προορίζεται για κάθε εξυπηρετητή. Δηλαδή το n_wt_{ij} είναι η ποσότητα πληροφορίας που οι χρήστες του κόμβου i ζητούν από τον εξυπηρετητή j . Είναι προφανές ότι :

$$\sum_{j \in S} n_wt_{ij} = n_wt_i \quad \text{Σχέση 4-6}$$

Για τον ορισμό των βαρών n_wt_{ij} για κάθε κόμβο i , στην περίπτωση που έχουμε ένα αρχείο ροής αιτήσεων, θεωρούμε ότι όλα τα βάρη είναι ίσα. Έτσι $n_wt_{ij} = n_wt_i / s$ (για κάθε j) όπου s είναι το πλήθος των εξυπηρετητών που υπάρχουν στο σύστημα. Στην περίπτωση που διαθέτουμε s αρχεία ροής αιτήσεων (1 για κάθε εξυπηρετητή), υπολογίζουμε για κάθε αρχείο του εξυπηρετητή j το βάρος στον κόμβο i , n_wt_{ij} . Το άθροισμα των n_wt_{ij} μας δίνει το βάρος του κόμβου i .

4.2 Αλγόριθμοι Τοποθέτησης Κρυφών Αντιπροσώπων

Στην ενότητα 2.6.1 παρουσιάστηκαν ορισμένοι αλγόριθμοι που σκοπό έχουν την επιλογή των κατάλληλων κόμβων ενός CDN ώστε να τοποθετηθούν με τον καλύτερο τρόπο τα αντίγραφα του εξυπηρετητή. Παρά το γεγονός ότι σύστημα που

παρουσιάστηκε στην ενότητα 2 είναι διαφορετικό από αυτό ενός CDN, μπορούμε να εφαρμόσουμε τους αλγόριθμους αυτούς χωρίς κανένα πρόβλημα.

Έτσι εξετάστηκαν οι αλγόριθμοι Greedy, HotSpot και Random. Στην μελέτη μας δεν παρουσιάζεται η απόδοση του tree-based αλγόριθμου, γιατί εξαιτίας των παραδοχών η απόδοση του δεν είναι τόσο καλή όσο των άλλων τριών, όταν αυτός εφαρμόζεται σε γράφο και όχι σε δέντρο.

Σκοπός της μελέτης μας πάνω στους αλγορίθμους τοποθέτησης αντιπροσώπων στο δίκτυο, είναι να βρούμε τον αποδοτικότερο από τους εξεταζόμενους αλγορίθμους. Στη συνέχεια χρησιμοποιώντας το μοντέλο που περιγράφηκε πιο πριν, αναπτύσσουμε ένα εργαλείο (το οποίο περιγράφεται στο κεφάλαιο 7) που αποφασίζει για το πόσοι κρυφοί αντιπρόσωποι και που πρέπει να τοποθετηθούν ώστε να επιτύχουμε την προσδοκώμενη βελτίωση στην απόδοση του συστήματος.

Κεφάλαιο 5

Αποτελέσματα Μελέτης Απόδοσης Αντιπροσώπων Διαδικτύου

Σε αυτό το τμήμα της μελέτης παρουσιάζονται τα αποτελέσματα των πειραμάτων σχετικά με την απόδοση των αλγορίθμων αντικατάστασης των κρυφών αντιπροσώπων. Εκτελέστηκε ένας μεγάλος αριθμός πειραμάτων για τιμές του θ που κυμαίνονται από 0.5 έως 1.0 με βήμα 0.05 και μεγέθη διαθέσιμου χώρου από 1% έως 30% με βήμα 5%. Αν δεν αναφερθεί κάπου συγκεκριμένα θεωρείται ότι ο CSP υλοποιείται με την βοηθητική κρυφή μνήμη και με τον μηχανισμό ελέγχου εισόδου.

5.1 Αποτελέσματα με Βάση τις Τρεις Μετρικές Απόδοσης

5.1.1 Αποτελέσματα Βασιζόμενοι στο Hit Ratio

Πρώτα απ' όλα, όλοι οι αλγόριθμοι αποδίδουν καλύτερα όσο ο διαθέσιμος αποθηκευτικός χώρος αυξάνεται κάτι το οποίο είναι λογικό μιας και αποθηκεύονται περισσότερα αντικείμενα και άρα παράγονται περισσότερα κτυπήματα στον τοπικό δίσκο.

Παρατηρήσαμε ότι ο GDS(1) αποδίδει καλύτερα από τον CSP, τον LRU και τον GDS(lat) όσον αφορά το hit ratio στις περισσότερες περιπτώσεις, πράγμα το οποίο συμβαδίζει με τα αποτελέσματα σχετικών μελετών[19]. Παρ' όλα αυτά όμως, τα αποτελέσματα μας έδειξαν ότι ο GDS(1) και GDS(lat) δεν αποδίδουν το ίδιο καλά όσο θα περίμενε κανείς κοιτάζοντας τις άλλες δύο μετρικές.

Τα συμπεράσματα που βγάλαμε συγκρίνοντας τον CSP με τον LRU, δείχνουν ότι ο LRU είναι καλύτερος σε περιβάλλοντα με αρκετό διαθέσιμο χώρο (με μέγεθος δίσκου 30%) και / ή περισσότερο ομοιόμορφες κατανομές δημοτικότητας αντικειμένων ($\theta=0.5$). Για περισσότερο απότομες κατανομές ($\theta=1.0$) ο CSP είναι καλύτερος εκτός των περιπτώσεων όπου ο διαθέσιμος χώρος είναι μεγάλος. Αυτό μπορεί να εξηγηθεί αν σκεφτεί κανείς το κόστος της βοηθητική κρυφής μνήμης η οποία απαιτείται για τον υπολογισμό του MTTR. Ας υποθέσουμε ότι έχουμε μια αίτηση για το αντικείμενο i , τότε πληροφορία για αυτό εισάγεται στην βοηθητική κρυφή μνήμη και προωθείται στον χρήστη. Στη δεύτερη αίτηση για το αντικείμενο i , ενδέχεται να εισαχθεί στον δίσκο. Το αν θα εισαχθεί ή όχι εξαρτάται από τον μηχανισμό ελέγχου εισόδου. Έτσι στη καλύτερη περίπτωση θα παρατηρηθεί κτύπημα στον δίσκο κατά την τρίτη αίτηση για το αντικείμενο i . Με το αλγόριθμο LRU το πρώτο κτύπημα θα σημειωθεί κατά την δεύτερη αίτηση στο αντικείμενο. Το κόστος λοιπόν της βοηθητικής μνήμης για τον CSP, είναι η απώλεια τουλάχιστον ενός κτυπήματος για κάθε αντικείμενο που τελικά αποθηκεύεται στο δίσκο. Δεδομένου λοιπόν ότι υπάρχει αρκετός χώρος ώστε ακόμα και ο LRU να εκμεταλλευτεί την δημοτικότητα των αντικειμένων καταλαβαίνουμε γιατί στις περιπτώσεις όπου έχουμε αρκετό διαθέσιμο χώρο και απότομη κατανομή της δημοτικότητας, είναι αποδοτικότερος ο LRU.

5.1.2 Αποτελέσματα Βασιζόμενοι στο Μέσο Χρόνο Απόκρισης

Παρατηρήσαμε ότι η απόδοση του GDS(1) για όλα τα μεγέθη δίσκου χειροτερεύει καθώς το θ αυξάνεται. Αποδίδει καλύτερα από τον CSP και LRU για περισσότερο ομοιόμορφες κατανομές της δημοτικότητας, αλλά για $\theta = 1.0$ η κατάσταση αντιστρέφεται. Η απόδοση του GDS(lat) είναι χειρότερη από αυτή του GDS(1) εκτός από την περίπτωση όπου το μέγεθος του δίσκου είναι μεγάλο και το $\theta = 1.0$. Αυτή η

συμπεριφορά του GDS(lat) δεν δικαιολογεί τον στόχο του που, όπως είπαμε, είναι να μειώσει τον μέσο χρόνο απόκρισης.

Για μικρούς δίσκους και $\theta = 1.0$ ο CSP αποδίδει καλύτερα συγκρινόμενος με τον GDS(lat) και τον GDS(1), ενώ για περισσότερο ομοιόμορφες κατανομές ($\theta = 0.5$) ο GDS(1) αποδίδει καλύτερα συγκρινόμενος με τους άλλους αλγορίθμους. Για μεγάλους δίσκους ο LRU αποδίδει μόνο για υψηλές τιμές του θ .

5.1.3 Αποτελέσματα Βασιζόμενοι στις Απαιτήσεις Εύρους Ζώνης

Για μικρούς δίσκους και υψηλές τιμές του θ , ο CSP είναι καλύτερος από τον GDS(1) και από τον GDS(lat), ενώ για μικρότερες τιμές του θ και μεγαλύτερους δίσκους, ο LRU είναι καλύτερος από τον GDS(1) και από τον GDS(lat). Ο χειρότερος είναι ο GDS(1) παρά το ότι αποδίδει από πλευράς hit ratio, διότι προτιμάει τα μικρά αντικείμενα πράγμα το οποίο οδηγεί στην ανάκτηση μεγάλων αντικειμένων από το διαδίκτυο.

5.1.4 Επιπρόσθετα Αποτελέσματα

Ο Ρόλος του Μεγέθους

Μετά από μια σειρά πειραμάτων που σκοπό είχαν να εξετάσουν το βάρος του όρου «μέγεθος αντικειμένου» στον αλγόριθμο αντικατάστασης, φάνηκε ότι αυτός είναι ιδιαίτερα σημαντικός. Έτσι τα αποτελέσματα έδειξαν ότι τελικά αξίζει να συμπεριλαμβάνει κανείς το μέγεθος του αντικειμένου, στον αλγόριθμο αντικατάστασης γιατί διαφορετικά το hit ratio μειώνεται κατά 80.4%. Παράλληλα παρατηρείται μια μικρή αύξηση του χρόνου απόκρισης κατά 10.96% και των απαιτήσεων σε εύρος ζώνης κατά 3.3%. Περιμέναμε μεγαλύτερη αύξηση λόγω της δραματικής μείωσης του hit ratio. Αυτό εξηγείται αν κανείς σκεφτεί τον διπλό ρόλο που παίζει το μέγεθος στον αλγόριθμο αντικατάστασης. Από την μια βοηθάει το hit ratio και άρα το μέσο χρόνο απόκρισης, ενώ από την άλλη οδηγεί στο να ζητούνται μεγαλύτερα αντικείμενα από τους εξυπηρετητές, το οποίο πληγώνει το μέσο χρόνο απόκρισης και τις απαιτήσεις σε εύρος ζώνης.

Ο Ρόλος των Βοηθητικών Μηχανισμών

Η ύπαρξη της βοηθητικής κρυφής μνήμης παίζει σημαντικό ρόλο στην απόδοση του συστήματος. Σε συνδυασμό με έναν μηχανισμό ελέγχου εισόδου μπορεί να οδηγήσει σε καλύτερη απόδοση. Φιλτράρει την ροή αιτήσεων κάνοντας το σύστημα λιγότερο ευαίσθητο σε απότομες αλλαγές του φόρτου εργασίας (workload). Έτσι τα αντικείμενα διατηρούνται για περισσότερο χρόνο στον δίσκο ενώ παράλληλα πρέπει να αποδείξουν ότι αξίζει να αποθηκευτούν, προτού εισαχθούν στο σύστημα.

Ο Ρόλος της Συνεργασίας των Αντιπροσώπων

Όσον αφορά τον ρόλο που παίζει η συνεργασία των κρυφών αντιπροσώπων μπορούμε να πούμε ότι αυτός είναι μάλλον μικρός με βάση τόσο τον χρόνο απόκρισης, όσο και τις απαιτήσεις σε εύρος ζώνης. Αυτό ισχύει παρά το γεγονός ότι όπως και σε άλλες μελέτες παρατηρήσαμε αύξηση της απόδοσης όσον αφορά το hit ratio η οποία αγγίζει το 30%.

5.2 Γενικότερα Συμπεράσματα Σχετικά με τα Αποτελέσματα

Ένα προφανές συμπέρασμα είναι ότι το hit ratio είναι μία μετρική η οποία δεν μπορεί να περιγράψει σωστά την απόδοσης των σύνθετων αλγορίθμων αντικατάστασης (όπως ο CSP) οι οποίοι περικλείουν αρκετούς όρους όπως το μέγεθος του αντικειμένου, ο χρόνος πρόσβασης κ.α. Αυτοί οι όροι δρουν κατά αντίθετο τρόπο οδηγώντας σε παραπλανητικές εκτιμήσεις της απόδοσης του συστήματος αν κάποιος κοιτάει μόνο το hit ratio.

Επίσης παρατηρήσαμε ότι τα αποτελέσματα που πήραμε σχετικά με τον χρόνο απόκρισης, ακολουθούν τις ίδιες τάσεις με τα αποτελέσματα σχετικά με τις απαιτήσεις σε εύρος ζώνης. Το γεγονός ότι τα αποτελέσματα με βάση τις απαιτήσεις σε εύρος ζώνης μετρήθηκαν ρητά στα πειράματά μας, μας οδηγεί στο συμπέρασμα ότι το αναλυτικό μοντέλο που χρησιμοποιήσαμε για το χρόνο πρόσβασης των αντικειμένων (ή καθυστέρηση) είναι αρκετά αξιόπιστο.

Πρέπει να σημειωθεί επίσης ότι ανομοιότητα των αποτελεσμάτων μεταξύ hit ratio και χρόνου απόκρισης είναι μικρότερη από αυτή μεταξύ hit ratio και απαιτήσεων σε εύρος ζώνης. Αυτό ισχύει διότι η βελτίωση στο hit ratio οφείλεται κατά ένα μεγάλο μέρος στην

τάση να διατηρούνται αποθηκευμένα μικρά αντικείμενα (και άρα περισσότερα) και να απορρίπτονται από τον κρυφό αντιπρόσωπο τα μεγαλύτερα. Εντούτοις, το να εξυπηρετηθούν αυτά τα αντικείμενα από τον μακρινό εξυπηρετητή, αυξάνει τις απαιτήσεις σε εύρος ζώνης του δικτύου κατά έναν άμεσο τρόπο. Από την άλλη, ο χρόνος απόκρισης επηρεάζεται λιγότερο από αυτό το γεγονός, δεδομένου ότι άλλοι παράγοντες, όπως ο αριθμός των hops, το εύρος ζώνης των γραμμών, κ.λπ. παίζουν επίσης τον ρόλο τους.

Ο Πίνακας 5-1 συνοψίζει τα αποτελέσματα για τις διαφορετικές τιμές της παραμέτρου θ και του διαθέσιμου μεγέθους του κρυφού αντιπροσώπου, για κάθε μια από τις τρεις μετρικές απόδοσης. Πρέπει να σημειώσουμε ότι αυτός ο πίνακας παρουσιάζει τον καλύτερο από πλευράς απόδοσης αλγόριθμο καθώς επίσης και τον επόμενο καλύτερο στην περίπτωση όπου ο ένας είναι ελάχιστα καλύτερος από τον άλλο.

Μέγεθος Δίσκου	θ						θ						θ					
	Παράμετρος Zipf Κατανομής						Παράμετρος Zipf Κατανομής						Παράμετρος Zipf Κατανομής					
	Καλύτερος αλγόριθμος με Βάση το						Καλύτερος αλγόριθμος με Βάση						Καλύτερος αλγόριθμος με Βάση τις					
	Hit Ratio						τον Χρόνο Απόκρισης						Απαιτήσεις σε Εύρος Ζώνης					
% του μεγ. απαιτ. χώρου	0.50	0.60	0.70	0.80	0.90	1.00	0.50	0.60	0.70	0.80	0.90	1.00	0.50	0.60	0.70	0.80	0.90	1.00
1%	▲	▲	▲	▲■	■	■▲	▲	▲●	■	■▲	■	■	×	●×	×	■×	■×	■×
5%	▲	▲	▲	▲	■×	■	▲×	▲×	▲	■×	■	■×	×	●	×	×	■×	■
10%	▲×	▲	▲	▲	×	■	●	●	●▲	■●	■●	■●	×	●	●	●	■●	●
15%	▲●	×	▲	▲×	●	×	●	×	●	■×	■●	●	●	×	●	●	×	■
20%	●	●×	▲●	▲×	×	×	▲×	×	●	×	×	●×	●×	×	●	●	×	●×
25%	●	×	×	●	×	▲×	▲●	●	●	×	●×	×	●×	●	●	●	●	●
30%	▲×	▲×	▲×	▲●	▲●	▲×	▲×	▲●	●×	▲●	●	●×	●×	●	●	●	×	●
LRU: ●						GDS(1): ▲		GDS(lat): ×				CSP: ■						

Πίνακας 5-1. Συνοπτική παρουσίαση των αποτελεσμάτων με βάση τον καλύτερο αλγόριθμο.

5.3 Παρουσίαση Δυναμικού Αλγόριθμου Αντικατάστασης

Έχοντας αποκτήσει σχετική εμπειρία πάνω στον τρόπο με το οποίο συμπεριφέρεται ο κάθε ένας από τους αλγόριθμους που μελετήσαμε, μπορούμε να προχωρήσουμε στην

ανάπτυξη ενός δυναμικού αλγόριθμου αντικατάστασης. Στις ενότητες που ακολουθούν γίνεται μία σύντομη παρουσίασή του.

5.3.1 Εκτίμηση Απόδοσης Αλγορίθμων

Η βασική ιδέα του δυναμικού αλγορίθμου, είναι η επιλογή του κατάλληλου αλγόριθμου αντικατάστασης ανάλογα με τις συνθήκες περιβάλλοντος. Για αυτό το σκοπό, δημιουργήθηκε ένας πίνακας ο οποίος συνοψίζει την απόδοση των αλγορίθμων αντικατάστασης (που μελετήθηκαν, δηλ. των GDS(1), GDS(lat), LRU και CSP). Η μορφή αυτού του πίνακα είναι παρόμοια με τον Πίνακα 5-1, μόνο που είναι πιο αναλυτικός. Πρέπει να σημειωθεί ότι όσον αφορά τα αποτελέσματα βασιζόμενοι στο χρόνο απόκρισης, μπορούν να αντικατασταθούν από μελέτες απόδοσης άλλων ερευνητών καθώς καλύτερη μοντελοποίηση του επικοινωνιακού κόστους μπορεί να αναπτυχθεί ή να υποβιβαστεί η σημαντικότητα του χρόνου απόκρισης στην αποδοτικότητα του συστήματος (Ειδικότερα αν λάβουμε υπόψη ότι οι περισσότεροι συμφωνούν ότι τα οφέλη όσον αφορά το χρόνο απόκρισης, είναι μικρότερα από αυτά που μπορεί κανείς να περιμένει [25],[26]).

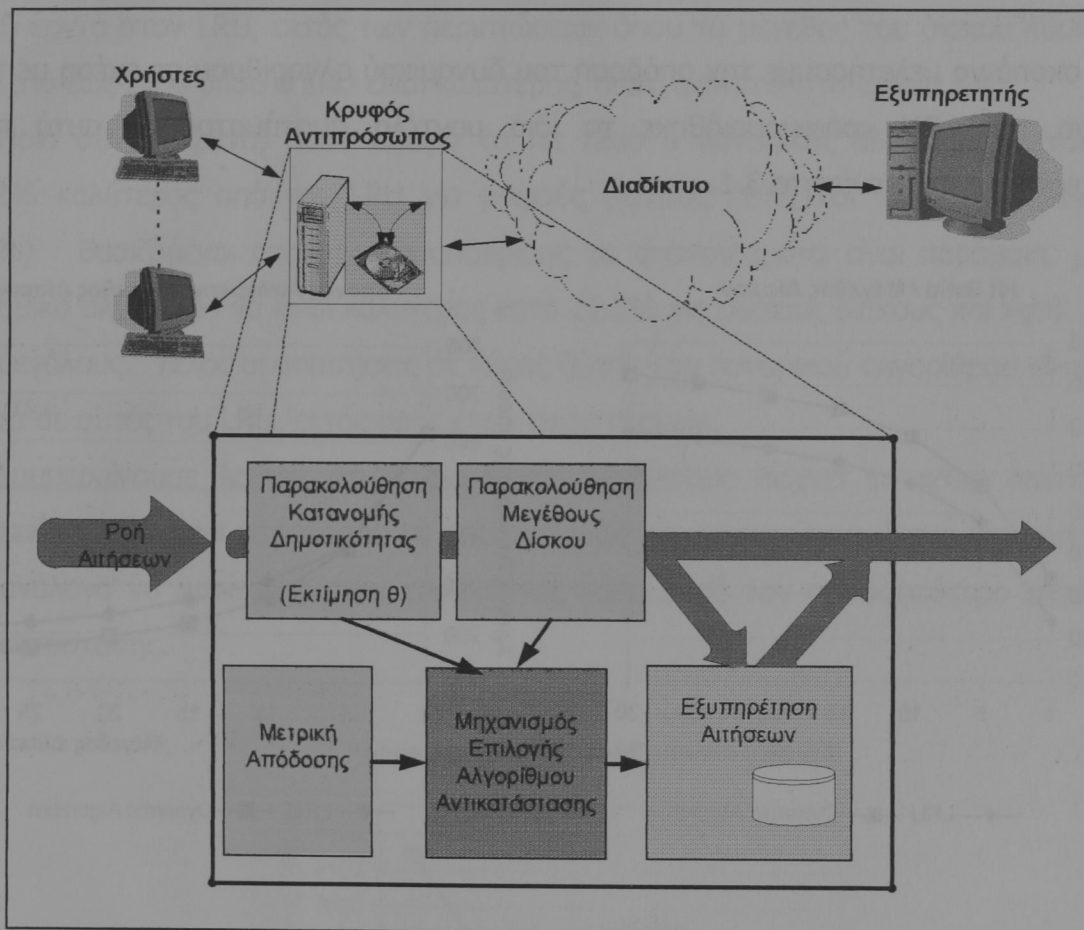
Έτσι λοιπόν, ο δυναμικός αλγόριθμος αντικατάστασης, μελετώντας τα χαρακτηριστικά της ροής αιτήσεων, συμβουλευεται τον σχετικό πίνακα απόδοσης και επιλέγει τον κατάλληλο αλγόριθμο αντικατάστασης.

5.3.2 Περιγραφή Υποσυστημάτων

Στο Σχήμα 5-1 φαίνονται τα βασικά υποσυστήματα ενός κρυφού αντιπροσώπου ο οποίος είναι εξοπλισμένος με τον δυναμικό αλγόριθμο αντικατάστασης.

Το υποσύστημα *παρακολούθησης κατανομής δημοτικότητας*, ελέγχει την ροή αιτήσεων που καταφθάνει στον κρυφό αντιπρόσωπο. Αποθηκεύονται στατιστικά στοιχεία για κάθε αντικείμενο που ζητείται, έτσι ώστε κάθε στιγμή να είναι δυνατή η εκτίμηση της τιμής της παραμέτρου θ της zipf κατανομής της δημοτικότητας των αντικειμένων. Ποιο συγκεκριμένα, ο υπολογισμός της τιμής της παραμέτρου θ γίνεται ανά N αιτήσεις με την μέθοδο χ^2 η οποία παρουσιάστηκε στην ενότητα 4.1.2.

Το υποσύστημα παρακολούθησης μεγέθους δίσκου, σκοπό έχει την εκτίμηση του ποσοστού του μέγιστου χώρου που απαιτείται για την αποθήκευση όλων των αντικειμένων που ζητούνται και αντιστοιχεί στον διαθέσιμο χώρο του κρυφού αντιπροσώπου. Έτσι αν θεωρήσουμε για παράδειγμα ότι το μέγεθος του δίσκου είναι 100MB και μετά από ένα χρονικό διάστημα το συνολικό μέγεθος των αντικειμένων που ζητήθηκαν, είναι 200MB, τότε το ποσοστό που ζητείται είναι 50%.



Σχήμα 5-1. Μοντέλο δυναμικού αλγόριθμου αντικατάστασης.

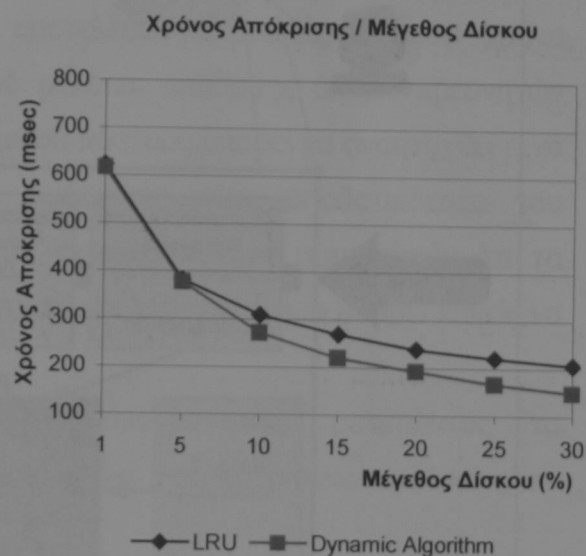
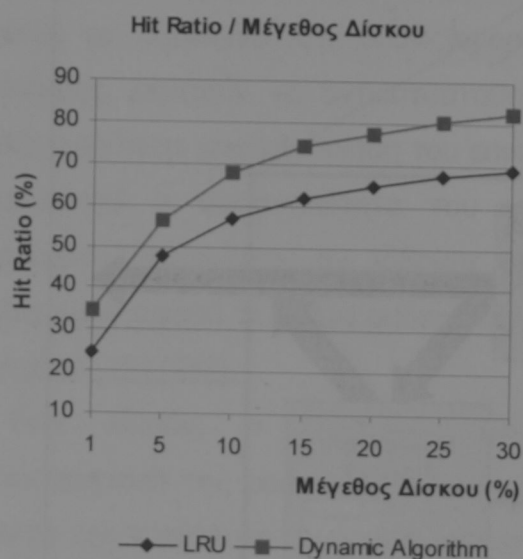
Ο διαχειριστής του κρυφού αντιπροσώπου εμπλέκεται στο σύστημα καθορίζοντας την μετρική απόδοσης. Δηλαδή, ορίζει αν θέλει να μεγιστοποιήσει το hit ratio, να μειώσει τον χρόνο απόκρισης των χρηστών ή να μειώσει τις απαιτήσεις σε εύρος ζώνης.

Τα παραπάνω λοιπόν υποσυστήματα, ανά πάσα στιγμή μπορούν να εκτιμήσουν την τιμή της παραμέτρου θ και το ποσοστό του διαθέσιμου χώρου. Έχοντας διαθέσιμη και την μετρική απόδοσης ο δυναμικός αλγόριθμος αντικατάστασης συμβουλευέται ανά τακτά χρονικά διαστήματα τον πίνακα απόδοσης (ο οποίος περιγράφηκε στην ενότητα

5.1), και αν κρίνει ότι οι συνθήκες περιβάλλοντος έχουν αλλάξει δραματικά, επιλέγει τον κατάλληλο αλγόριθμο αντικατάστασης, ο οποίος είναι ένας από τους GDS(1), GDS(lat), LRU και CSP.

5.3.3 Αποτελέσματα Μελέτης Απόδοσης Δυναμικού Αλγορίθμου

Με σκοπό να μελετήσουμε την απόδοση του δυναμικού αλγορίθμου σε σχέση με την απόδοση του LRU χρησιμοποιήθηκε το ίδιο μοντέλο συστήματος με αυτό που παρουσιάστηκε στην ενότητα 3.1.



Σχήμα 5-2. Hit Ratio μεταβάλλοντας το μέγεθος του δίσκου.

Σχήμα 5-3. Χρόνος απόκρισης με μεταβλητό μέγεθος δίσκου.

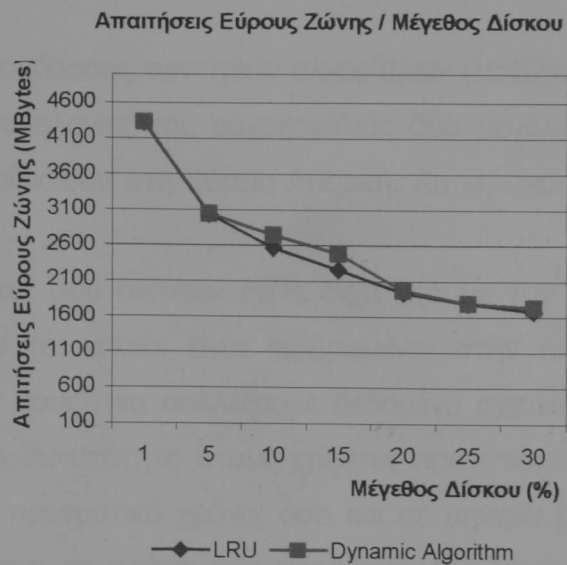
Χρησιμοποιήσαμε συνθετική ροή αιτήσεων η οποία δημιουργήθηκε με την βοήθεια του SURGE [13] και χαρακτηρίζεται από μεταβλητή τιμή της παραμέτρου θ της zipf κατανομής. Έτσι ο φόρτος εργασίας χωρίζεται σε τρία μέρη, με το πρώτο να χαρακτηρίζεται από $\theta=0.6$, το δεύτερο $\theta=0.8$ και το τρίτο 0.9 . Αποτελείται από 450,000 αιτήσεις και περιέχει 35,000 αντικείμενα. Έγιναν πειράματα για επτά διαφορετικά μεγέθη διαθέσιμου αποθηκευτικού χώρου, από 1% έως 30% του μέγιστου χώρου που απαιτείται για την αποθήκευση όλων των αντικειμένων που ζητούνται. Τέλος, η εκτίμηση της τιμής της παραμέτρου θ γίνεται κάθε $N=100,000$ αιτήσεις (μετά από σχετική μελέτη

καταλήξαμε σε αυτή την τιμή. Παρατηρήσαμε ότι δε υπάρχει μεγάλη διαφορά στην απόδοση επιλέγοντας άλλες τιμές, εκτός βέβαια των ακραίων τιμών, δηλαδή πολύ μικρές ή πολύ μεγάλες).

Τα αποτελέσματα δείχνουν ότι όσον αφορά τις μετρικές hit ratio (Σχήμα 5-2) και χρόνος απόκρισης (Σχήμα 5-3) ο δυναμικός αλγόριθμος είναι αποδοτικότερος από τον LRU. Όταν κανείς κοιτάει τις απαιτήσεις σε εύρος ζώνης, ο δυναμικός αλγόριθμος είναι πολύ κοντά στον LRU, εκτός των περιπτώσεων όπου το μέγεθος του δίσκου κυμαίνεται από 5% έως 20% όπου ο LRU είναι καλύτερος, όπως φαίνεται στο Σχήμα 5-4.

Ποιο συγκεκριμένα, όσον αφορά το hit ratio ο δυναμικός αλγόριθμος είναι κατά 42.2% καλύτερος από τον LRU για μικρούς δίσκους (1%) και 19.4% για μεγάλους (30%). Βασιζόμενοι στον χρόνο απόκρισης τα αποτελέσματα είναι παρόμοια, με τον δυναμικό αλγόριθμο να είναι καλύτερος κατά 21.2% για μικρούς δίσκους και κατά 27.5% για μεγάλους. Τέλος οι απαιτήσεις σε εύρος ζώνης του δυναμικού αλγορίθμου είναι πολύ κοντά σε αυτές του LRU, εκτός ορισμένων περιπτώσεων.

Συμπεραίνουμε λοιπόν ότι ο δυναμικός αλγόριθμος δείχνει να κάνει σωστά την δουλειά του μιας και καταφέρνει να παρακολουθεί τα χαρακτηριστικά της ροής αιτήσεων και ανάλογα να προσαρμόζεται, επιλέγοντας κάθε φορά τον αποδοτικότερο αλγόριθμο αντικατάστασης.



Σχήμα 5-4. Χρήση εύρους ζώνης με μεταβλητό μέγεθος δίσκου.

Κεφάλαιο 6

Μελέτη Απόδοσης Αλγορίθμων Τοποθέτησης Αντιπροσώπων Διαδικτύου

6.1 Τοπολογία Δικτύου

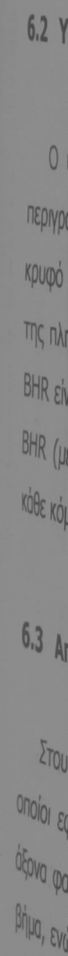
Για την μελέτη της απόδοσης των τριών αλγορίθμων (HotSpot, Random και Greedy) χρησιμοποιήθηκαν οι τοπολογίες της ραχοκοκαλιάς δύο μεγάλων ISPs που παρέχουν υπηρεσίες σύνδεσης διαδικτύου στη Βόρεια Αμερική. Αυτές είναι της AT&T [44] και της C&W [45].

Η επιλογή αυτών των δύο δικτύων έγινε διότι έχουμε την δυνατότητα μέσω των δικτυακών τους τόπων (οι οποίοι είναι αφιερωμένοι στην παρουσίαση στατιστικών απόδοσης των δικτύων τους), να συλλέξουμε δεδομένα σχετικά με την απόδοση του δικτύου τους. Δίνεται η δυνατότητα στους χρήστες που επισκέπτονται τις ιστοσελίδες τους να δουν τόσο σε πραγματικό χρόνο, όσο και σε μηνιαία βάση την απόδοση του δικτύου του εκφραζόμενη σε χρόνο που χρειάζεται ένα πακέτο πληροφορίας για να ταξιδέψει από ένα σημείο του δικτύου (κόμβο) σε ένα άλλο. Ο χρόνος αυτός μετριέται σε milliseconds και εξαρτάται από αρκετούς παράγοντες όπως ταχύτητα και φόρτος

όσο κ
διακύμ
ΣΤΕ
των β
και τ
C&W
Το
στο π
ρών
εξηγη
αντιστ
τρόπο
που π
ΤΕ
τοπολο

6.2 Υ

Ο
περιγρ
κρυφό
της πλ
BHR είν
BHR (μ
κάθε κ



Αυτή η πληροφορία είναι ιδιαίτερα σημαντική για την μελέτη μας αφού μπορούμε έτσι να χρησιμοποιήσουμε πραγματικές τιμές στα βάρη των ακμών του δικτύου. Έτσι ανά μία ώρα και για έναν ολόκληρο μήνα (Μάρτιος 2002) συλλέχθηκαν τα βάρη των ακμών των δύο δικτύων και μπορέσαμε να κρατήσουμε τόσο ημερήσιους μέσους όρους

όσο και μηνιαίους. Μετά από μελέτη των τιμών αυτών, προέκυψε σχετικά μικρή διακύμανση από μέρα σε μέρα, δείχνοντας ότι το δίκτυο τους είναι αρκετά σταθερό.

Στα πειράματα λοιπόν που ακολουθούν χρησιμοποιήθηκαν οι μηνιαίες μέσες τιμές των βαρών των ακμών. Στο Σχήμα 4-2 φαίνεται η τοπολογία της ραχοκοκαλιάς της AT&T και τα βάρη των ακμών, ενώ στο Σχήμα 6-1 φαίνεται ο αντίστοιχη τοπολογία για την C&W.

Τα βάρη στους κόμβους προκύπτουν από κατάλληλη επεξεργασία (όπως αναφέρεται στο τέλος της ενότητας 4.1.1) τόσο συνθετικών ροών αιτήσεων όσο και πραγματικών ροών αιτήσεων που συλλέχθηκαν από του δικτυακούς τόπους ιδιαίτερα φορτωμένων εξυπηρετητών [17]. Οι ανάθεση των ομάδων χρηστών στους κόμβους (και άρα η αντιστοίχιση των υποροών στους κόμβους) έγινε με 100 διαφορετικούς τυχαίους τρόπους. Εφαρμόστηκαν οι αλγόριθμοι για όλες αυτές τις τοπολογίες και οι μέσες τιμές που προέκυψαν χρησιμοποιήθηκαν για την παρουσίαση των αποτελεσμάτων.

Τέλος θεωρούμε ότι ο εξυπηρετητής είναι εγκατεστημένος στον κόμβο 1 και στις δύο τοπολογίες (AT&T και C&W).

6.2 Υπολογισμός Κόστους Γράφου

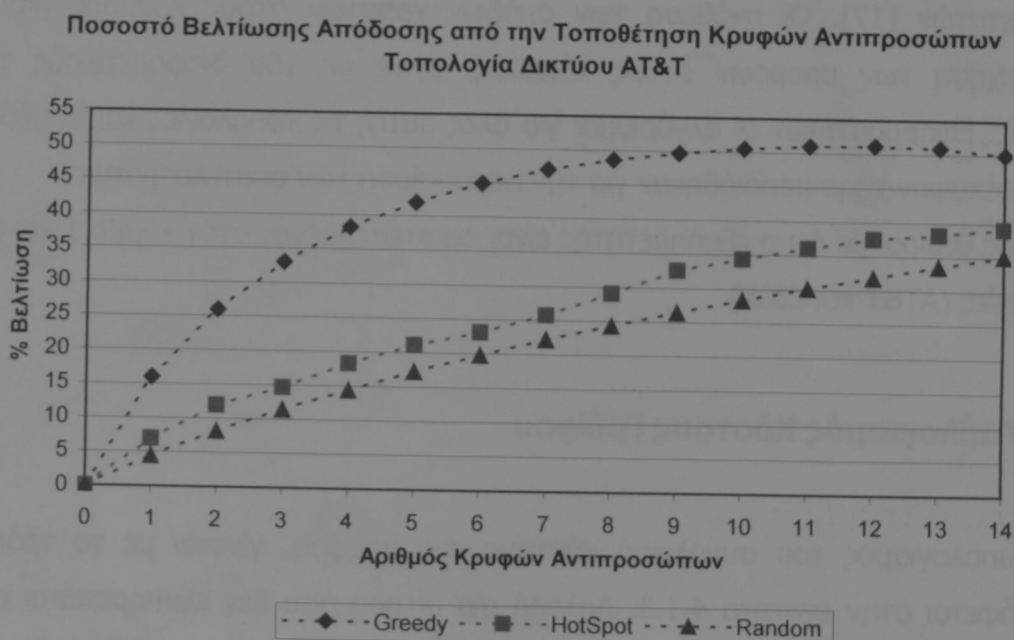
Ο υπολογισμός του συνολικού κόστους του γράφου, γίνεται με το τρόπο που περιγράφεται στην ενότητα 4.1.3. Δηλαδή μία αίτηση που δεν εξυπηρετείται από έναν κρυφό αντιπρόσωπο, μπορεί να εξυπηρετηθεί από τον επόμενο πιο κοντινό. Το ποσοστό της πληροφορίας που είναι κοινή μεταξύ των κρυφών αντιπροσώπων είναι 55%, ενώ το BHR είναι κοινό για όλους και ίσο με 45%. Όπως αναφέρθηκε στην παράγραφο 4.1.2 το BHR (με την βοήθεια της μεθοδολογίας που παρουσιάστηκε) θα είναι διαφορετικό για κάθε κόμβο. Ωστόσο σε αυτή τη μελέτη το θεωρούμε σταθερό και ίσο με 45%.

6.3 Αποτελέσματα Απόδοσης

Στους γράφους που ακολουθούν φαίνεται η απόδοση των τριών αλγορίθμων οι οποίοι εφαρμόζονται στις τοπολογίες δικτύου της AT&T και της C&W. Στον οριζόντιο άξονα φαίνεται ο αριθμός των κρυφών αντιπροσώπων που έχουν τοποθετηθεί σε κάθε βήμα, ενώ στον κατακόρυφο το ποσοστό βελτίωσης του συνολικού κόστους σε σχέση

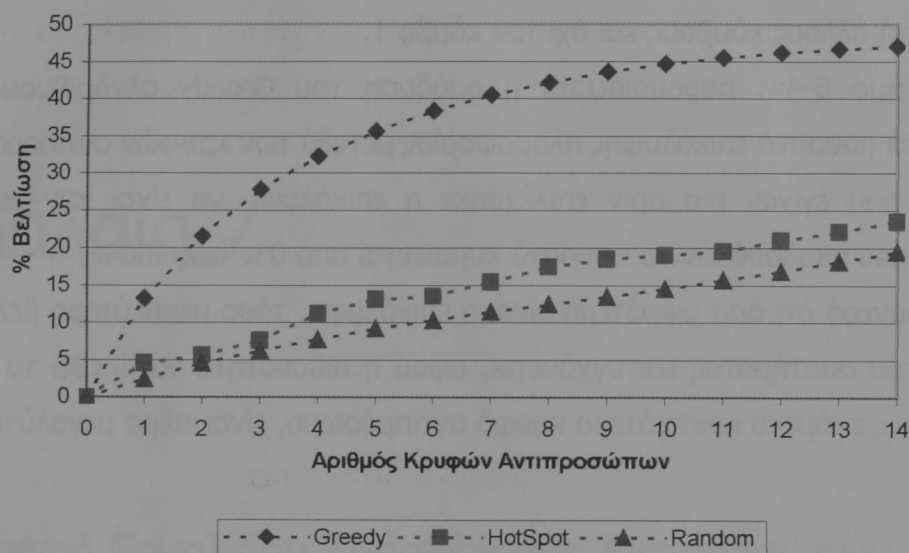
με το συνολικό κόστος του γράφου που προκύπτει χωρίς την τοποθέτηση κρυφών αντιπροσώπων.

Και στις δύο τοπολογίες τα αποτελέσματα είναι παραπλήσια. Φαίνεται καθαρά ότι ο Greedy είναι αποδοτικότερος σε σχέση με τους άλλους δύο, ενώ ο Random όπως είναι αναμενόμενο, είναι ο χειρότερος. Βλέπουμε για παράδειγμα στο Σχήμα 6-2 ότι με την τοποθέτηση 2 κρυφών αντιπροσώπων επιτυγχάνουμε βελτίωση της τάξεως του 26% με τον Greedy ενώ η αντίστοιχη βελτίωση για τον HotSpot είναι γύρω στο 12% και για τον Random γύρω στο 8%. Τα αντίστοιχα ποσοστά για την τοπολογία της C&W (Σχήμα 6-3) είναι για τον Greedy 24% , για τον HotSpot 5.5% και για τον Random 4.5%.



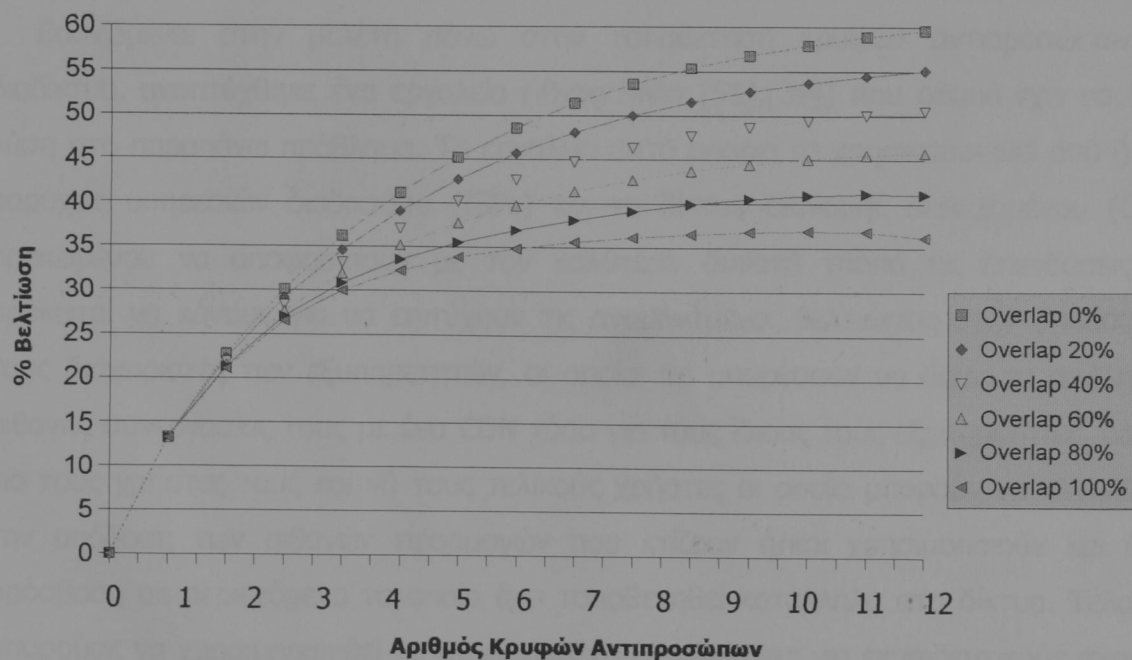
Σχήμα 6-2. Απόδοση των αλγορίθμων μετά από εφαρμογή τους στην τοπολογία δικτύου της AT&T.

Ποσοστό Βελτίωσης Απόδοσης από την Τοποθέτηση Κρυφών Αντιπροσώπων
Τοπολογία Δικτύου C&W



Σχήμα 6-3. Απόδοση των αλγορίθμων μετά από εφαρμογή τους στην τοπολογία δικτύου της C&W.

Ποσοστό Βελτίωσης Απόδοσης από την Τοποθέτηση Κρυφών Αντιπροσώπων
με Μεταβλητό Ποσοστό Επικάλυψης



Σχήμα 6-4. Απόδοση του αλγορίθμου Greedy μετά από εφαρμογή τους στην τοπολογία δικτύου της C&W, με μεταβλητό ποσοστό επικάλυψης.

Αξίζει να σημειωθεί ότι την ίδια συμπεριφορά ακολουθούν οι τρεις αλγόριθμοι (με πρώτο τον Greedy και στη συνέχεια HotSpot και Random), ακόμα και αν επιλέξουμε σαν εξυπηρετητή άλλους κόμβους και όχι τον κόμβο 1.

Στο Σχήμα 6-4 , παρουσιάζεται η απόδοση του Greedy αλγόριθμου κάτω από διαφορετικά ποσοστά επικάλυψης πληροφορίας μεταξύ των κρυφών αντιπροσώπων. Στις μετρήσεις που έγιναν πιο πριν επιλέχθηκε η επικάλυψη να είναι ίση με 55%. Στις μετρήσεις που ακολουθούν το ποσοστό κυμαίνεται από 0% έως 100%.

Είναι φανερό ότι όσο μικρότερη είναι η επικάλυψη, τόσο μεγαλύτερη βελτίωση στην απόδοση του συστήματος επιτυγχάνεται, αφού η πιθανότητα να βρεθεί το αντικείμενο στον αμέσως επόμενο κοντινότερο κρυφό αντιπρόσωπο, είναι τώρα μεγαλύτερη.

Κεφάλαιο 7

Περιγραφή Εργαλείου Τοποθέτησης Αντιπροσώπων Διαδικτύου

Βασιζόμενοι στην μελέτη πάνω στην τοποθέτηση κρυφών αντιπροσώπων στο διαδίκτυο, αναπτύχθηκε ένα εργαλείο (*ProxyTeller* [51],[39]) που σκοπό έχει να δώσει λύση στο παραπάνω πρόβλημα. Το εργαλείο αυτό μπορεί να χρησιμοποιηθεί από i) τους παροχείς υπηρεσιών διαδικτύου (ISPs) και τα δίκτυα διανομής περιεχομένου (CDNs) προκειμένου να αποφασίσουν με τον καλύτερο δυνατό τρόπο τις επενδύσεις που πρόκειται να κάνουν για να επιτύχουν τις αναμενόμενες βελτιώσεις στην απόδοση, ii) τους διαχειριστές των εξυπηρετητών, οι οποίοι θα μπορέσουν να δουν τα οφέλη μιας πιθανής συνεργασίας τους με ένα CDN τόσο για τους ίδιους τους εξυπηρετητές, όσο και για τους χρήστες τους και iii) τους τελικούς χρήστες οι οποίοι μπορούν να καταλάβουν την απόδοση των πιθανών εφαρμογών που κτίζουν ή/και χρησιμοποιούν και έχουν πρόσβαση σε περιεχόμενο το οποίο έχει τοποθετηθεί κατάλληλα στο δίκτυο. Τέλος, θα μπορούσε να χρησιμοποιηθεί το συγκεκριμένο εργαλείο και για εκπαιδευτικούς σκοπούς. Στις ενότητες που ακολουθούν παρουσιάζονται τα βασικά τμήματά του.

7.1 Πλατφόρμα Ανάπτυξης

Χρησιμοποιήθηκαν εργαλεία ανάπτυξης αλληλεπιδραστικών εφαρμογών στο διαδίκτυο, με σκοπό το εργαλείο να είναι εύκολα προσβάσιμο από μεγάλο αριθμό χρηστών από όλο τον κόσμο. Πιο συγκεκριμένα χρησιμοποιήθηκε η γλώσσα PHP σε συνδυασμό με υποστήριξη βάσεως δεδομένων MySQL. Για την παρουσίαση του εργαλείου και την αλληλεπίδρασή του με τους χρήστες, χρησιμοποιήθηκε φυσικά HTML, ενώ για τους ελέγχους των δεδομένων που εισάγει ο χρήστης επιλέχθηκε η javascript. Οι αλγόριθμοι τοποθέτησης κρυφών αντιπροσώπων υλοποιήθηκαν σε C++, ενώ σε ορισμένα σημεία ήταν και απαραίτητη η χρήση της java.

Το σύστημα πάνω στο οποίο αναπτύχθηκε το εργαλείο, υποστηρίζει μεγάλο αριθμό χρηστών ελέγχοντας την πρόσβασή τους και καταχωρώντας τα στοιχεία τους και τις προτιμήσεις τους στην βάση δεδομένων.

ProxyTeller
A Proxy Cache Placement Tool

Signup

Username :

Password :

Re-type Password :

Security Question :

Answer :

First Name :

Last Name :

e-mail :

All fields are necessary.

For comments and/or propositions concerning these web pages contact [Ioannis Askaterinides](#).

Σχήμα 7-1. Φόρμα εισαγωγής στοιχείων χρήστη με σκοπό την εγγραφή του στο σύστημα.

7.2 Περιγραφή του ProxyTeller

7.2.1 Είσοδος Χρηστών στο Σύστημα

Όπως αναφέρθηκε πιο πριν, το εργαλείο αυτό είναι διαθέσιμο σε ολόκληρο τον κόσμο μέσω του διαδικτύου. Ωστόσο για να μπορέσει κάποιος να το χρησιμοποιήσει πρέπει να ακολουθήσει ορισμένα απλά βήματα ώστε να γίνει μέλος του συστήματος. Στο Σχήμα 7-1 φαίνεται η σχετική φόρμα που καλείται να συμπληρώσει ο χρήστης για να γίνει μέλος. Πιο συγκεκριμένα, ζητείται το όνομα χρήστη (username), ο κωδικός εισόδου (password), η απάντηση σε μια ερώτηση ασφαλείας (σκοπό έχει να βοηθήσει στην διαδικασία ανάκτησης του password στην περίπτωση που ο χρήστης το ξεχάσει), το ονοματεπώνυμο του χρήστη και το e-mail του.

here.' The 'Existing User' section contains fields for 'Username:' and 'Password:', a link 'Forgot your password? Click [Here](#)', and two buttons: 'Cancel' (with an X icon) and 'Ok' (with a checkmark icon). At the bottom, there is a footer text: 'For comments and/or propositions concerning these web pages contact [Ioannis Aekaterinides](#).'"/>

Σχήμα 7-2. Φόρμα εισαγωγής στοιχείων χρήστη με σκοπό την είσοδο του στο σύστημα.

Αν ο χρήστης είναι ήδη μέλος του συστήματος, μπορεί να εισάγει το όνομα χρήστη (username) που έχει δηλώσει κατά την εγγραφή του και τον κωδικό του (password) (

Σχήμα 7-2) και να εισέλθει στην περιοχή του συστήματος που απευθύνεται στα μέλη. Σε περίπτωση που ο χρήστης έχει ξεχάσει τον κωδικό του, μπορεί να πατήσει στον σχετικό σύνδεσμο (hyperlink) και ακολουθώντας ορισμένα απλά βήματα, να παραλάβει το κωδικό του μέσω e-mail.

7.2.2 Βασικά Σενάρια Δράσης του ProxyTeller

Μετά την επιτυχημένη εισαγωγή του στο σύστημα, ο χρήστης καλείται να επιλέξει ανάμεσα σε 6 διαφορετικά σενάρια δράσης (Σχήμα 7-3) κάτω από τα οποία ο ProxyTeller

καλείται να δώσει λύση στο πρόβλημα της τοποθέτησης κρυφών αντιπροσώπων ή πλήρων αντιγράφων των εξυπηρετητών. Αυτά είναι τα παρακάτω:

1. Μοναδικός Εξυπηρετητής και Τοποθέτηση Κρυφών Αντιπροσώπων

Σε αυτό το σενάριο υπάρχει ένας μόνο εξυπηρετητής ο οποίος μπορούμε να θεωρήσουμε ότι δραστηριοποιείται στα πλαίσια ενός μικρού δικτύου(π.χ. στην περιοχή ενός μικρού οργανισμού). Στόχος του ProxyTeller είναι να προτείνει την καλύτερη δυνατή τοποθέτηση κρυφών αντιπροσώπων σύμφωνα με τις δυνατότητες και προσδοκίες του οργανισμού (π.χ. το οικονομικό κόστος της τοποθέτησης περισσότερων κρυφών αντιπροσώπων, την προσδοκώμενη βελτίωση της απόδοσης του δικτύου).

2. Μοναδικός Εξυπηρετητής και Τοποθέτηση Πλήρων Αντιγράφων

Το σενάριο αυτό είναι παρόμοιο με το προηγούμενο, με την διαφορά ότι σε αυτή την περίπτωση τοποθετούνται εξυπηρετητές οι οποίοι έχουν την δυνατότητα να φιλοξενήσουν αντίγραφα όλου του περιεχομένου του αρχικού εξυπηρετητή.

3. Εξυπηρετητές και Τοποθέτηση Κρυφών Αντιπροσώπων

Θεωρείστε την περίπτωση όπου οι διαχειριστές ενός δικτύου παροχής περιεχομένου αποφασίζουν να τοποθετήσουν κρυφούς αντιπροσώπους με σκοπό να εξυπηρετήσουν με την βοήθεια αυτών, τις αιτήσεις χρηστών προς εξυπηρετητές οι οποίοι προσφέρουν διαφορετικό περιεχόμενο.

4. Εξυπηρετητές και Τοποθέτηση Πλήρων Αντιγράφων (Συνηθισμένο πρόβλημα σε δίκτυα παροχής περιεχομένου)

Σε αυτό το σενάριο ο ProxyTeller δίνει λύση σε ένα συνηθισμένο πρόβλημα που αντιμετωπίζουν τα δίκτυα παροχής περιεχομένου. Αν υποθέσουμε ότι υπάρχουν M εξυπηρετητές οι οποίοι έχουν εμπιστευθεί το περιεχόμενο τους σε ένα CDN τότε ο ProxyTeller προσπαθεί να τοποθετήσει στο δίκτυο εξυπηρετητές, οι οποίοι έχουν την δυνατότητα να φιλοξενήσουν M διαφορετικά πλήρη αντίγραφα, κατά τον καλύτερο δυνατό τρόπο.

5. Τοποθέτηση Επιπλέον Κρυφών Αντιπροσώπων σε Δίκτυο Παροχής Περιεχομένου

Αν υποθέσουμε ότι στο δίκτυο ενός CDN είναι τοποθετημένα κατάλληλα ορισμένα αντίγραφα ενός εξυπηρετητή, τότε σκοπός του ProxyTeller είναι να προτείνει την επιπλέον τοποθέτηση κρυφών αντιπροσώπων ώστε να βελτιωθεί παραπάνω η απόδοση του συστήματος.

ProxyTeller Logout

Basic Scenarios

A Description of ProxyTeller

At this page you are asked to choose the scenario for which ProxyTeller is called to provide an answer. These six scenarios are:

1. Single Web Origin Server with Proxies	Scenario 1
Consider a web origin server in the bounds of a small organization and the need to decide where to place proxy servers within the network so as to deliver efficiently content to clients. Supplying the expected workload (which as an approximation, could be the web servers' trace file for a large period of time), and the network topology to ProxyTeller, the tool can compute the best placement based on the expressed needs (i.e., how many proxy servers the organization can afford).	
2. Single Web Origin Server with Full Replicas	Scenario 2
This is similar to the previous scenario with the exception of placing full replicas, instead of proxies, able to store all objects of the web origin server.	
3. Several Web Origin Servers with Proxies	Scenario 3
This scenario could emerge when there are many web origin servers storing different content. For example, a large CDN may wish to decide where to place reverse proxies within its network to improve efficiency.	
4. Several Web Origin Servers with Replicas: The Common Problem in CDNs	Scenario 4
Suppose that there are M origin servers managed by a CDN. Our goal is to find the nodes of the network that should accommodate the full replicas of each origin server. This scenario can be applied in the case where a given proxy has the capacity to store M different replicas (1 replica for each origin server).	
5. Incremental Improvement: Additional Surrogate Servers in a CDN	Scenario 5
Within the CDN there are already placed a number of full replicas of a given origin server. Our goal is to improve further the overall performance by placing surrogate servers (that is, partial replicas of the origin server content).	
6. Incremental Improvement: Additional full Replica Servers in a CDN	Scenario 6
This scenario is similar to scenario 5, with the difference that we are trying to place full server replicas, not surrogate servers.	

For comments and/or propositions concerning these web pages contact ioannis.Askatennides.

Σχήμα 7-3. Σελίδα επιλογής σεναρίου δράσης.

6. Τοποθέτηση Επιπλέον Πλήρων αντιγράφων σε Δίκτυο Παροχής Περιεχομένου

Αυτό το σενάριο είναι παρόμοιο με το προηγούμενο με την διαφορά ότι επιθυμούμε να τοποθετήσουμε επιπλέον πλήρη αντίγραφα στο δίκτυο και όχι κρυφούς αντιπροσώπους (οι οποίοι δεν περιέχουν αντίγραφα όλων των αντικειμένων του εξυπηρετητή)

Τα αποτελέσματα που δίνει ο ProxyTeller για τα σενάρια 3,4,5 και 6 μπορούν να συγκριθούν ώστε να μπορέσει κανείς να κρίνει τι είναι ποιο αποδοτικό να κάνει: να τοποθετήσει κρυφούς αντιπροσώπους ή εξυπηρετητές με πλήρη αντίγραφα, λαμβάνοντας βέβαια υπόψη του και τις οικονομικές επιπτώσεις μιας πιθανής επιλογής.

The screenshot displays the ProxyTeller web interface for configuring a new placement. The header shows 'ProxyTeller' and 'New Placement'. The main content area is titled 'Scenario 3: Several Web Origin Servers with Proxies'. Below this, there are tabs for 'Save Results', 'Load Results', 'Your Workloads', and 'Your Topologies'. The 'Input Parameters' section includes the following fields:

- Topology:** A dropdown menu set to 'C&W'.
- Expected Workload:** A dropdown menu set to '0.8'.
- Cache Size:** A text input field set to '65' MBytes.
- Average Requested Data per Node:** A text input field set to '100' MBytes.
- Expected Performance Improvement:** A text input field set to '35' %.
- Performance Metric:** A dropdown menu set to 'Minimize Mean Latency'.
- Define Web Origin Servers (at least 1):** A section with a list of nodes (1, 2, 3, 4, 5, 6) and a text input field containing '3 4'. Below the list, it says 'ctrl+left mouse button for multiple selections' and 'Node numbers must be separated with spaces and should be smaller than 21'. An example '1 4 6' is provided.

On the right side, there are links for 'View Selected Topology', 'Create your own Topology', and 'Upload your Expected Workload'. At the bottom, there is a 'Simulate Placement' button.

Σχήμα 7-4. Φόρμα εισαγωγής παραμέτρων συστήματος.

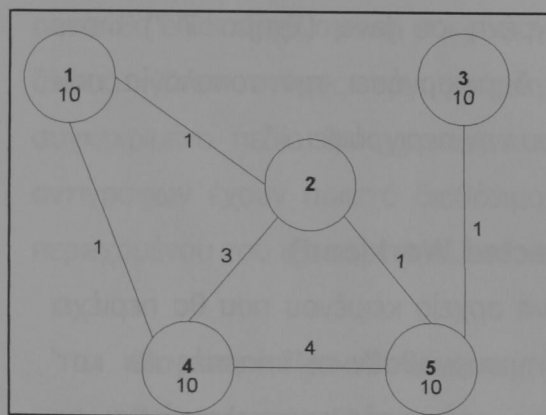
7.2.3 Είσοδος Παραμέτρων Συστήματος

Αφού ο χρήστης επιλέξει το σενάριο δράσης που τον ενδιαφέρει, καλείται να προσδιορίσει ορισμένες παραμέτρους οι οποίες είναι απαραίτητες για την εκτέλεση του αλγορίθμου τοποθέτησης, συμπληρώνοντας κατάλληλα ορισμένα πεδία όπως φαίνεται στο Σχήμα 7-4. Αυτά είναι:

□ Τοπολογία Δικτύου (Topology)

Πρέπει να είναι γνωστή η τοπολογία, οπότε πρέπει να περιγραφεί κατάλληλα. Έτσι το δίκτυο περιγράφεται με τον εξής τρόπο: πρόκειται για ένα αρχείο κειμένου, όπου στην πρώτη γραμμή υπάρχει ένα ζεύγος αριθμών με τον πρώτο να δηλώνει των αριθμό των κόμβων και τον δεύτερο τον αριθμό των γραμμών. Στην συνέχεια ακολουθεί μια γραμμή για κάθε κόμβο που δείχνει με ποιους άλλους κόμβους είναι συνδεδεμένος και το βάρος της κάθε ακμής. Έτσι για το δίκτυο του Σχήμα 7-5 το αρχείο κειμένου που το περιγράφει φαίνεται στο Σχήμα 7-6.

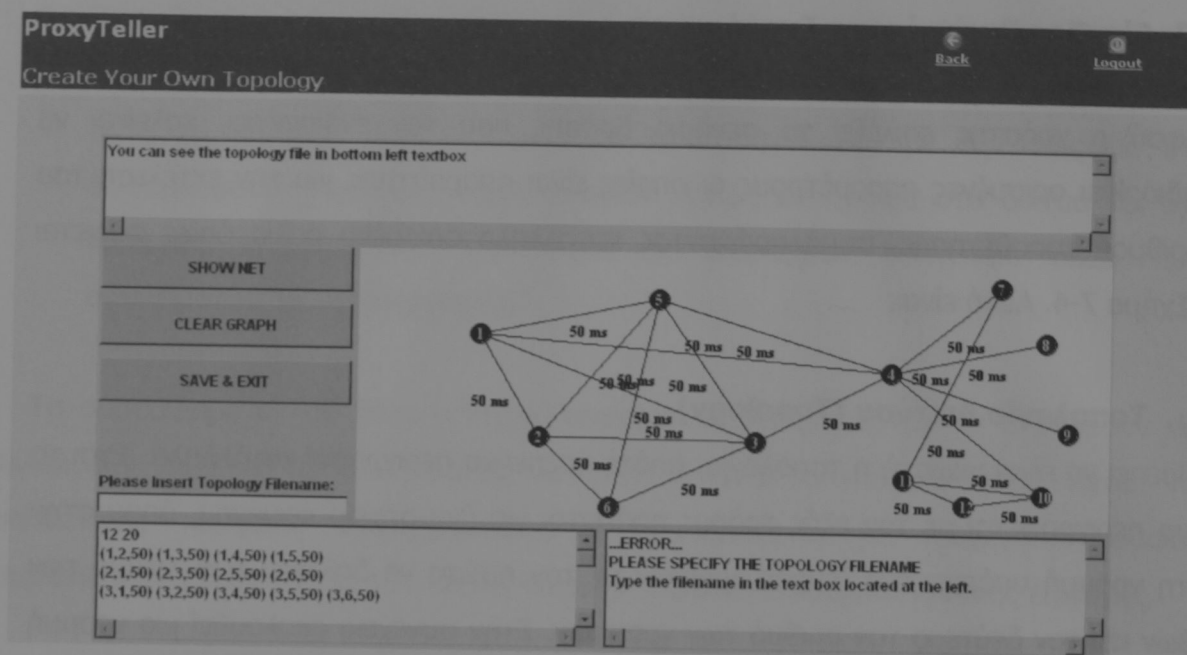
Όπως φαίνεται λοιπόν στο Σχήμα 7-6, στην πρώτη γραμμή κάθε παρένθεση περιγράφει και μία ακμή, δείχνοντας έτσι ότι ο κόμβος 1 είναι συνδεδεμένος με τον 2 και το βάρος της ακμής είναι 1.



Σχήμα 7-5 Παράδειγμα Τοπολογίας Δικτύου.

```
5 6
(1,2,1) (1,4,1)
(2,1,1) (2,4,3) (2,5,1)
(3,5,1)
(4,1,1) (4,2,3) (4,5,4)
(5,2,1) (5,3,1) (5,4,4)
```

Σχήμα 7-6 Περιγραφή της τοπολογίας του Σχήμα 7-5.



Σχήμα 7-7. Δημιουργία νέας τοπολογίας με την βοήθεια του γραφικού εργαλείου.

Ο χρήστης σε αυτό το σημείο έχει τρεις επιλογές. Μπορεί να χρησιμοποιήσει τοπολογίες πραγματικών δικτύων, όπως αυτές της AT&T και C&W οι οποίες αναφέρθηκαν πιο πριν (AT&T Σχήμα 4-2 και C&W Σχήμα 6-1). Μπορεί να εισάγει το δικό του αρχείο κειμένου που περιγράφει το δίκτυο που αυτός επιθυμεί να μελετήσει. Τέλος μπορεί να χρησιμοποιήσει μία εφαρμογή ανεπτυγμένη σε java (Σχήμα 7-7) ώστε τοποθετώντας κατάλληλα κόμβους και ακμές, να δημιουργήσει την τοπολογία που επιθυμεί και κατ' επέκταση και το αρχείο κειμένου που την περιγράφει.

□ Προσδοκώμενος Φόρτος Εργασίας (Expected Workload)

Όπως αναφέρθηκε πιο πριν, είναι απαραίτητο ένα αρχείο κειμένου που θα περιέχει την ροή αιτήσεων των χρηστών, έτσι ώστε να δημιουργηθούν οι υποροές και κατ' επέκταση τα βάρη των κόμβων. Επίσης από την ροή αιτήσεων μελετούνται τα χαρακτηριστικά της κάθε υποροής έτσι ώστε να προσδιοριστεί η προσδοκώμενη απόδοση των κρυφών αντιπροσώπων, όταν αυτοί τελικά τοποθετηθούν στους κατάλληλους κόμβους.

Έτσι σε αυτό το σημείο ο χρήστης μπορεί είτε να εισάγει ένα ή περισσότερα αρχεία ροής αιτήσεων (ακολουθώντας 3 απλά βήματα, Σχήμα 7-8) είτε να χρησιμοποιήσει ήδη υπάρχοντα αρχεία τα οποία παρέχονται από την εφαρμογή.

ProxyTeller
Main Page
Logout

Upload Your Expected Workload

Uploading Status

You uploaded the following file: testtt.txt
The uploading was succesfull.

Workload File Format (step 2 of 3)

The ProxyTeller basically supports workload files that are very close to the Squid log file formats.
In order to correctly analyze and use your workload files, ProxyTeller should be able to locate the requested object and its size.
Thus object's URL must start with 'http://'.
The size of the object can be located by selecting appropriately the radio button below.

This is a sample request line from a Squid log file.
1020816537.022239539134.202.51.180TCP_MISS/5041130GEThttp://www.jdekas.ee/avznpxz/link.html- NONE/-

Size field

Object

This is the first line of the workload file you just uploaded. Please locate the object's size field.

1020816537.022	239539	147.27.7.244	TCP_MISS/504	1130	GET	http://pcp.softnet.tuc.gr/index.htm	-	NONE/-	-
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Next

For comments and/or propositions concerning these web pages contact Ioannis Aekaterinides.

Σχήμα 7-8. Διαδικασία εισαγωγής αρχείων ροής αιτήσεων.

❑ Μέγεθος Δίσκου (Cache Size)

Το μέγεθος του δίσκου παίζει σημαντικό ρόλο στην απόδοση του κρυφού αντιπροσώπου. Έτσι ο χρήστης καλείται να προσδιορίσει το μέγεθος του δίσκου με τον οποίο θα είναι εξοπλισμένοι οι κρυφοί αντιπρόσωποι που πρόκειται να τοποθετηθούν στο δίκτυο. Όταν στόχος του χρήστη είναι η τοποθέτηση πλήρων αντιγράφων, το συγκεκριμένο πεδίο δεν χρησιμοποιείται αφού έχουμε θεωρήσει ότι οι εξυπηρετητές αντιγράφων έχουν αρκετό διαθέσιμο αποθηκευτικό χώρο για την φιλοξενία όλου του περιεχομένου του εξυπηρετητή.

❑ Ποσοστό Προσδοκώμενης Βελτίωσης Απόδοσης (Expected Performance Improvement)

Σε αυτό το μέρος ο χρήστης ζητείται να προσδιορίσει το ποσοστό βελτίωσης στην απόδοση που επιθυμεί να επιτύχει, τοποθετώντας κρυφούς αντιπρόσωπους ή πλήρη αντίγραφα περιεχομένου εξυπηρετητών στο δίκτυο. Το ποσοστό αυτό περιγράφει την μείωση του κόστους του δικτύου (όπως αυτό περιγράφηκε στην ενότητα 4.1.3) που επιτυγχάνεται με την κατάλληλη τοποθέτηση. Εναλλακτικά, μπορεί ο χρήστης να μην εισάγει ποσοστό βελτίωσης αλλά τον αριθμό των κρυφών αντιπροσώπων που είναι

διατεθειμένος να τοποθετήσει.

□ **Μετρική Απόδοσης (Performance Metric)**

Ο χρήστης καλείται να εισάγει την μετρική απόδοσης των κρυφών αντιπροσώπων. Έχει τρεις επιλογές, να μειώσει τις απαιτήσεις σε εύρος ζώνης, να μειώσει τον χρόνο απόκρισης των χρηστών και να αυξήσει το hit ratio. Ανάλογα με το τι θα επιλέξει θα περιγραφεί με μία συγκεκριμένη τιμή στο Byte Hit Ratio η απόδοση του κάθε κρυφού αντιπροσώπου και θα προταθεί στον χρήστη ο κατάλληλος για κάθε κρυφό αντιπρόσωπο αλγόριθμος αντικατάστασης (επιλέγοντας έναν από τους LRU, GDS(1), GDS(lat) και CSP). Στην περίπτωση τοποθέτησης πλήρων αντιγράφων το συγκεκριμένο πεδίο δεν είναι απαραίτητο.

□ **Ορισμός Εξυπηρετητών (Origin Server Definition)**

Ανάλογα με το σενάριο που επιλέγει ο χρήστης, καλείται να προσδιορίσει σε ποιους κόμβους του δικτύου είναι εγκαταστημένοι οι εξυπηρετητές (ή τα ήδη τοποθετημένα πλήρη αντίγραφα τους). Έτσι, είτε επιλέγει από μια λίστα του αντίστοιχους κόμβους, είτε πληκτρολογεί τους πληκτρολογεί σε ένα διαθέσιμο πεδίο.

Αφού λοιπόν ο χρήστης προσδιορίσει τις παραπάνω παραμέτρους, η ρουτίνα που είναι αρμόδια για την εφαρμογή του αλγορίθμου τοποθέτησης κρυφών αντιπροσώπων / αντιγράφων, καλείται. Ο Greedy αλγόριθμος (όπως περιγράφεται στην παράγραφο 2.6) εφαρμόζεται έως ότου το κόστος του δικτύου μειωθεί κατά το ποσοστό που έχει ορίσει ο χρήστης. Πιο συγκεκριμένα, το εργαλείο αρχίζει με την επεξεργασία του αναμενόμενου φόρτου εργασίας προκειμένου να υπολογιστούν τα βάρη των κόμβων για τη δεδομένη τοπολογία, με τον τρόπο που περιγράφεται στην παράγραφο 4.1.1. Επιπλέον, η προσδοκώμενη υποροή αιτήσεων για κάθε κόμβο, αναλύεται προκειμένου να καθοριστεί η τιμή της παραμέτρου θ της zipf κατανομής της δημοτικότητας των αντικειμένων, όπως περιγράφεται στην παράγραφο 4.1.2. Αυτή η τιμή καθώς επίσης και ο διαθέσιμος αποθηκευτικός χώρος, χρησιμοποιούνται έπειτα ως δείκτης στον πίνακα απόδοσης (όπως ο Πίνακας 5-1) ώστε να βρεθεί τόσο ο αλγόριθμος αντικατάστασης του κρυφού αντιπροσώπου όσο και η απόδοση του (η οποία εκφράζονται με το BHR), που πρόκειται να τοποθετηθεί σε κάθε κόμβο. Κατά συνέπεια, στο τέλος αυτής της διαδικασίας το

εργαλείο θα είναι σε θέση να ξέρει την αναμενόμενη απόδοση ενός κρυφού αντιπροσώπου που τοποθετείται σε οποιοδήποτε κόμβο. Το επόμενο βήμα περιλαμβάνει την εύρεση της ελάχιστης απόστασης από έναν κόμβο σε άλλος, για όλα τα ζευγάρια των κόμβων. Αυτό το βήμα είναι ουσιαστικό στον υπολογισμό του συνολικού κόστους τοποθέτησης κρυφού αντιπροσώπου (παράγραφος 4.1.3) Τέλος, ο Greedy αλγόριθμος εφαρμόζεται για την τοποθέτηση όλο και περισσότερων κρυφών αντιπροσώπων, έως ότου επιτευχθεί η επιθυμητή βελτίωση της απόδοσης του συστήματος. Τα αποτελέσματα παρουσιάζονται στον χρήστη με τον τρόπο που περιγράφεται στην επόμενη παράγραφο.

7.2.4 Παρουσίαση Αποτελεσμάτων

Αφού λοιπόν δώσει ο χρήστης την απαραίτητη πληροφορία στο σύστημα καλείται η ρουτίνα η οποία είναι υπεύθυνη για την τοποθέτηση των κρυφών αντιπροσώπων. Μετά από μερικά δευτερόλεπτα, ο χρήστης μπορεί να δει τα αποτελέσματα της τοποθέτησης με βάση τον Greedy αλγόριθμο (Σχήμα 7-9).

Η σελίδα παρουσίασης χωρίζεται σε τρία μέρη. Στο πάνω-αριστερά τμήμα παρουσιάζονται οι τιμές των παραμέτρων που εισήγαγε ο χρήστης (και παρουσιάστηκαν στην προηγούμενη ενότητα). Στο πάνω-δεξιά τμήμα παρουσιάζονται τα αποτελέσματα της τοποθέτησης με τους κόμβους στους οποίους πρόκειται να τοποθετηθούν οι κρυφοί αντιπρόσωποι / αντίγραφα, το ποσοστό βελτίωσης της απόδοσης του συστήματος που επιτυγχάνεται μετά από κάθε τοποθέτηση, το ποσοστό του διαθέσιμου αποθηκευτικού χώρου κάθε κρυφού αντιπροσώπου και τον αλγόριθμο αντικατάστασης που προτείνει το σύστημα. Στο υπόλοιπο μισό της σελίδας παρουσιάζεται το δίκτυο με τοποθετημένους τους κρυφούς αντιπροσώπους / αντίγραφα ακολουθώντας κατάλληλο χρωματικό κώδικα όπως φαίνεται στο Σχήμα 7-9.

Επίσης αξίζει να σημειωθεί ότι ο χρήστης έχει την δυνατότητα να περιηγηθεί γύρω από το δίκτυο του σε τρεις διαστάσεις και να παρατηρήσει έτσι την τοποθέτηση που επέλεξε να κάνει ο ProxyTeller. Μπορεί αν θέλει να κατεβάσει ο χρήστης στον υπολογιστή του ένα αρχείο τύπου εικόνας (GIF) ή postscript με τα αποτελέσματα της τοποθέτησης που προτείνει ο ProxyTeller.

Scenario 3

Several Web Origin Servers with Proxies

Save Results

Load Results

Your Workloads

Your Topologies

Input Parameters

Topology	C&W
Expected Workload	0.8
Cache Size	65 MByte
Expected Performance Improvement	35 %
Performance Metric	Latency

Greedy Placement Results

#	Overall Performance Improvement	Node Label	Cache Size	Replacement Algorithm
[0]	0.0 %	3 (Server)		
[0]	0.0 %	4 (Server)		
[0]	0.0 %	2 (Server)		
[1]	31.78 %	9	49.18 %	GDS(lat)
[2]	42.74 %	17	57.01 %	GDS(lat)

Given desired performance improvement reached or exceeded

2 proxies should be placed.

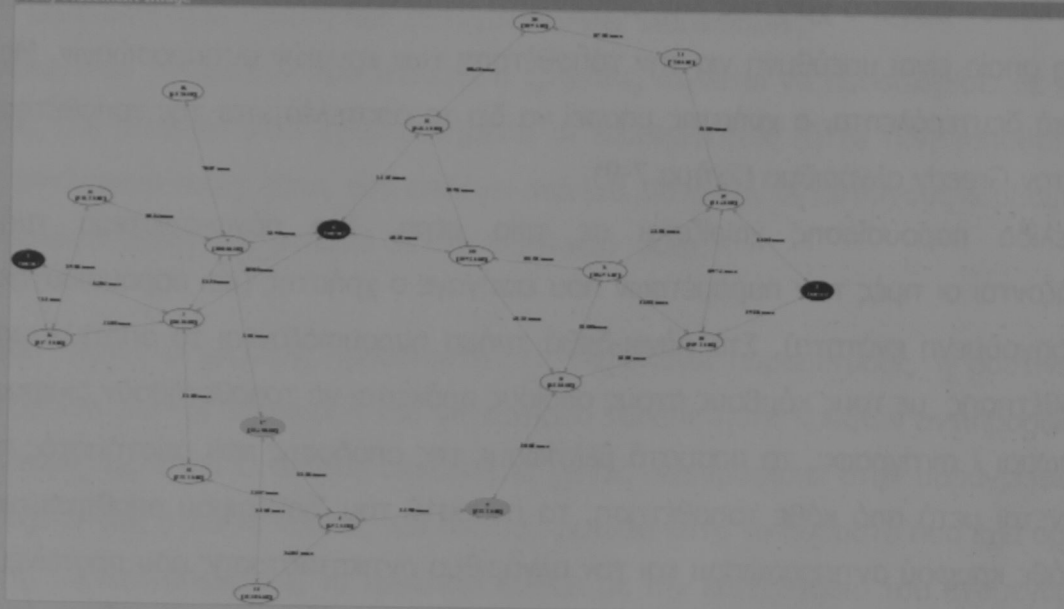
Total number of Nodes in network 21

Download the topology preview file: [gif file](#)

Download the topology preview file: [pscript file](#)

View the 3D topology: [3D](#)

Proxy Placement Image



Σχήμα 7-9. Παρουσίαση αποτελεσμάτων.

7.2.5 Επιπρόσθετες Λειτουργίες του ProxyTeller

Εκτός από τις βασική λειτουργία του εργαλείου που είναι ο προσδιορισμός των παραμέτρων εισόδου και στη συνέχεια η παρουσίαση των αποτελεσμάτων, παρέχονται στον χρήστη και άλλες λειτουργίες που σκοπό έχουν να τον βοηθήσουν στη χρήση του εργαλείου. Αυτές είναι:

ProxyTeller
Back
New Placement
Logout

Load Results

Here you can view the placement results that you have chosen to save. By clicking the corresponding link at the *Name* column you can view the detailed results. You can also insert comments for your results or deleting them by clicking the icon at the right column.

Name	Input Parameters	Comments
run1	Scenario 4 Several Web Origin Servers with Replicas: The Common Problem in CDNs Topology: test Expected Workload: test_workload_2 Cache Size: 50 MByte Performance Improvement: 50 % Performance Metric: Latency Origin Servers: 2 3	Scenario 4 with 2 origin servers Update Comments
run2	Scenario 1 Single Web Origin Server with Proxies Topology: test Expected Workload: test_workload_2 Cache Size: 2 MByte Performance Improvement: 50 % Performance Metric: Latency Origin Servers: 2	Scenario 1 with very small cache size Update Comments
run3	Scenario 4 Several Web Origin Servers with Replicas: The Common Problem in CDNs Topology: test Expected Workload: test_workload_2 Cache Size: 3 MByte Performance Improvement: 60 % Performance Metric: Latency Origin Servers: 1	Scenario 4 with 1 origin server and small cache size Update Comments

For comments and/or propositions concerning these web pages contact [Ioannis Aekaterinides](#).

Σχήμα 7-10. Σελίδα ανάκτησης αποτελεσμάτων.

□ Αποθήκευση και Ανάκτηση Αποτελεσμάτων

Δίνεται η δυνατότητα στον χρήστη να αποθηκεύσει τα αποτελέσματα τοποθέτησης που επιθυμεί. Έτσι μπορεί αν θέλει να κρατήσει ένα αρχείο με τα αποτελέσματα που τον ενδιαφέρουν και να ανατρέξει σε αυτά αργότερα βλέποντας τα ξανά με λεπτομέρεια. Στο Σχήμα 7-10 φαίνεται η σελίδα ανάκτησης αποτελεσμάτων, στην οποία ο χρήστης μπορεί να δει τις τιμές των παραμέτρων εισόδου για κάθε αποτέλεσμα και τα σχόλια που έχει εισάγει ο ίδιος. Τέλος πατώντας πάνω σε σχετικό σύνδεσμο, μπορεί να δει τα συγκεκριμένα αποτελέσματα (στην μορφή του Σχήμα 7-9) ή αν επιθυμεί να τα διαγράψει από το σύστημα.

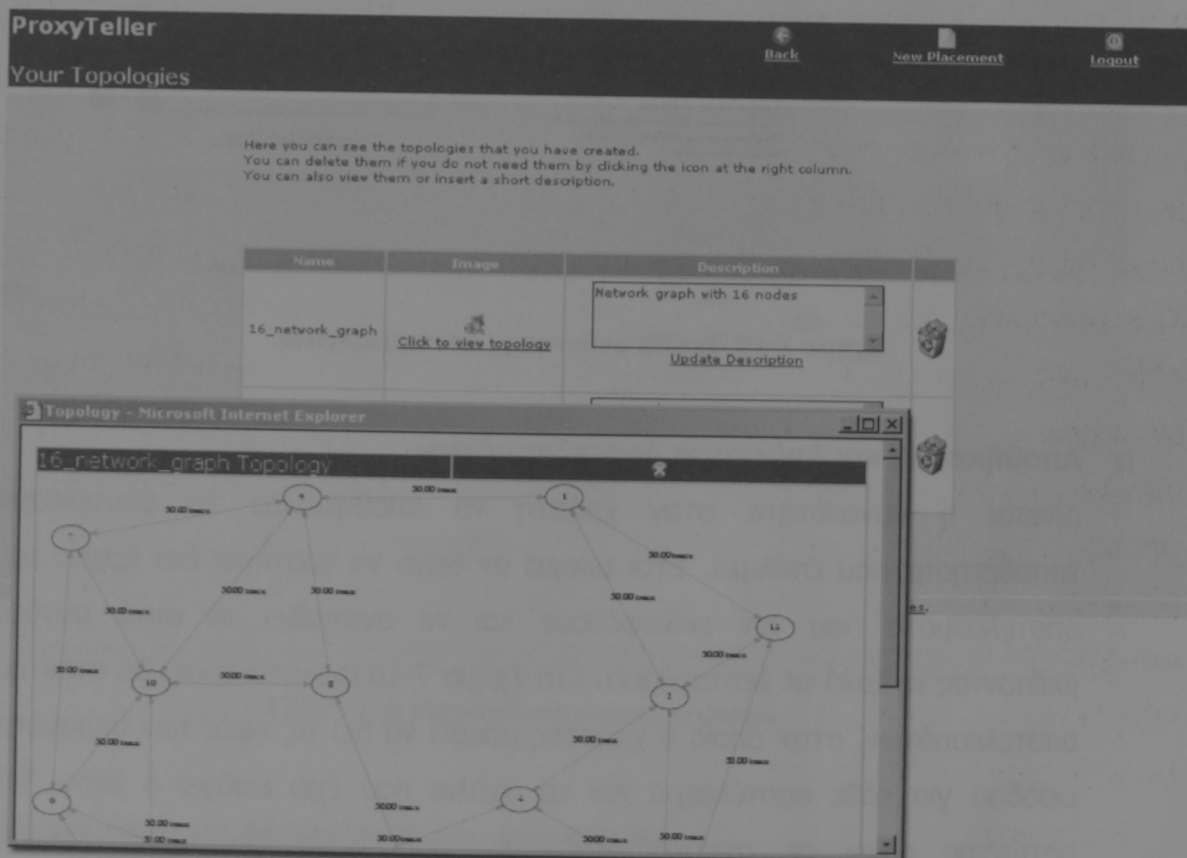
□ Παρουσίαση Τοπολογιών Χρήστη

Στο Σχήμα 7-11 φαίνεται η σελίδα στην οποία παρουσιάζονται οι τοπολογίες που έχει δημιουργήσει ή έχει εισάγει στο σύστημα ο χρήστης. Μπορεί, πατώντας

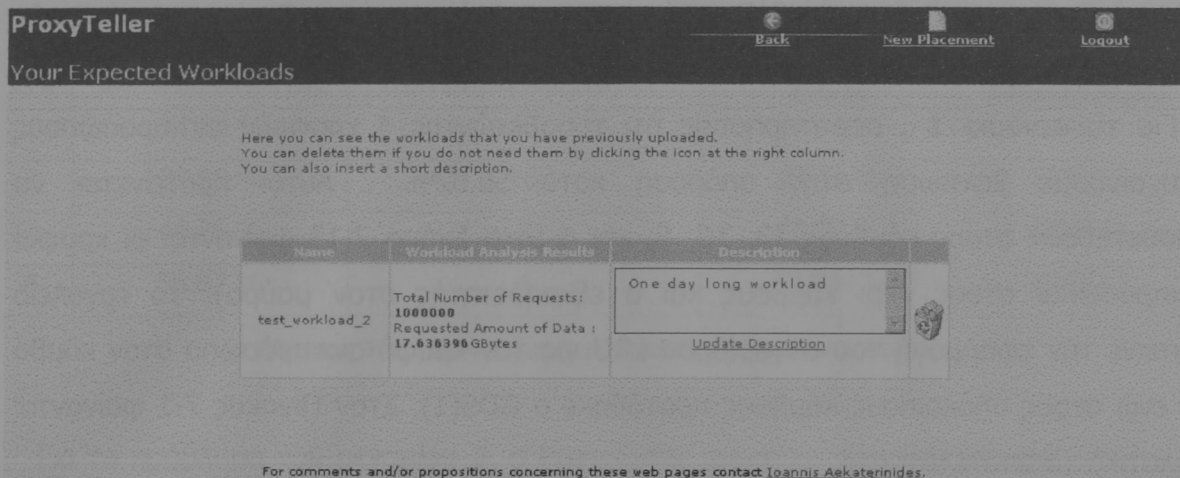
πάνω σε σχετικό σύνδεσμο, να δει μια γραφική απεικόνιση της τοπολογίας που τον ενδιαφέρει, να διαβάσει / δημιουργήσει σχετικά με την τοπολογίας σχόλια και τέλος αν το επιθυμεί να διαγράψει την τοπολογία από το σύστημα.

□ Παρουσίαση Προσδοκώμενων Φόρτων Εργασίας Χρήστη

Παράλληλα με τις τοπολογίες του, ο χρήστης μπορεί να ανατρέξει στα αρχεία ροής αιτήσεων (προσδοκώμενος φόρτος εργασίας) που έχει χρησιμοποιήσει μέχρι τώρα. Όπως φαίνεται στο Σχήμα 7-12, για κάθε αρχείο παρουσιάζονται ορισμένες σχετικές πληροφορίες για αυτό, όπως επίσης και σχόλια τα οποία μπορεί να διαγράψει και να ενημερώσει ο χρήστης. Τέλος ο χρήστης αν επιθυμεί μπορεί να διαγράψει ένα αρχείο ροής αιτήσεων, πατώντας σε σχετικό σύνδεσμο.



Σχήμα 7-11. Παρουσίαση τοπολογιών που έχει εισάγει / δημιουργήσει ο χρήστης.



Σχήμα 7-12. Παρουσίαση φόρτου εργασίας που έχει χρησιμοποιήσει ο χρήστης.

7.3 Παρουσίαση Ενδεικτικών Αποτελεσμάτων Χρήσης του ProxyTeller

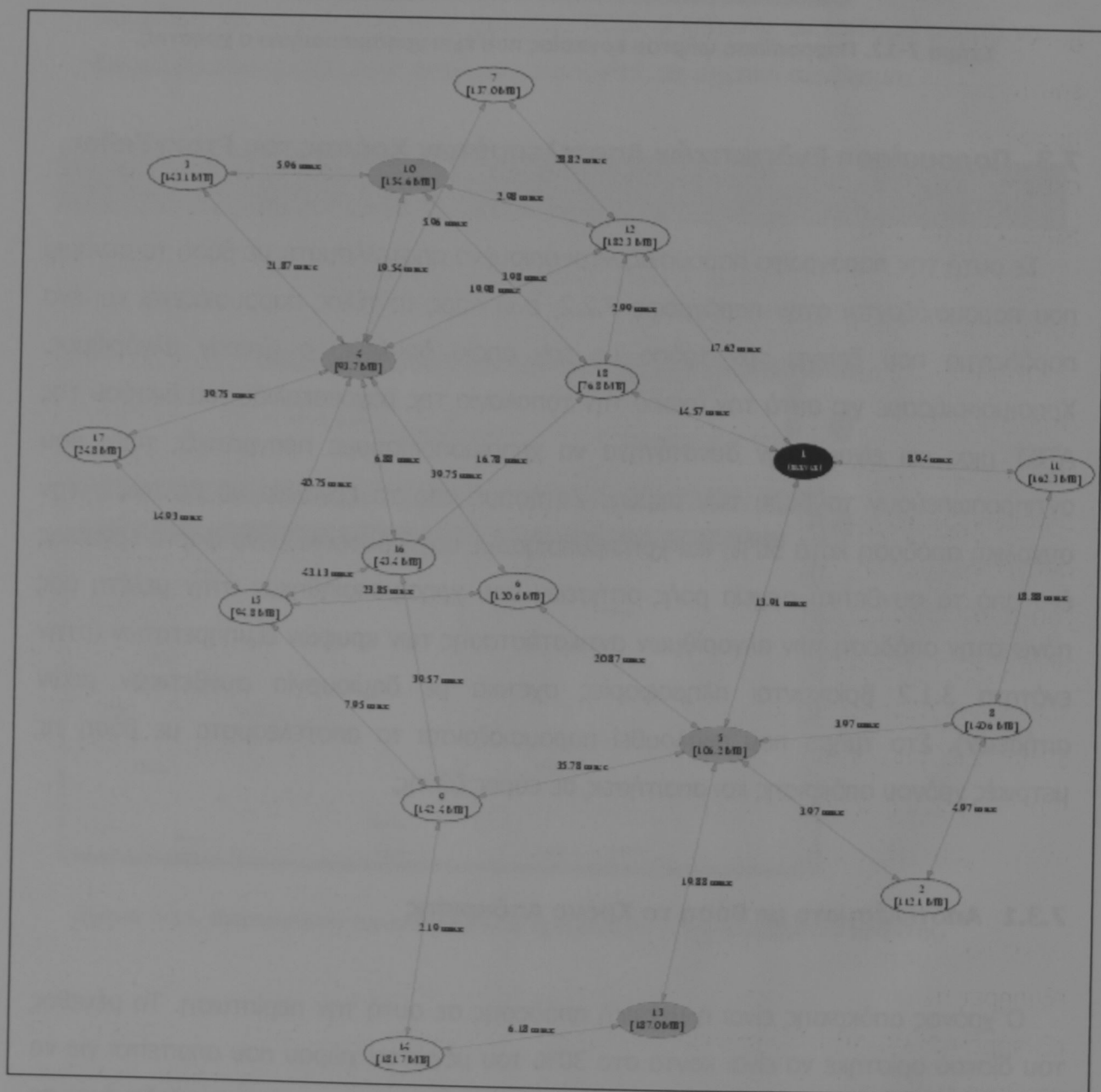
Σε αυτή την παράγραφο παρουσιάζονται ορισμένα αποτελέσματα με βάση τα σενάρια που παρουσιάζονται στην παράγραφο 7.2.2, ενώ προς το τέλος παρουσιάζεται και ένα παράδειγμα που δείχνει τον τρόπο με τον οποίο δουλεύει ο Greedy αλγόριθμος. Χρησιμοποιήσαμε για αυτό τον σκοπό την τοπολογία της ραχοκοκαλιάς του δικτύου της AT&T μιας και είχαμε την δυνατότητα να χρησιμοποιήσουμε πραγματικές τιμές που αντιπροσωπεύουν τα βάρη των ακμών. Ζητήσαμε από το εργαλείο να βελτιώσει την συνολική απόδοση κατά 50%, και χρησιμοποιήσαμε σαν προσδοκώμενο φόρτο εργασίας ένα από τα συνθετικά αρχεία ροής αιτήσεων που χρησιμοποιήθηκαν στην μελέτη μας πάνω στην απόδοση των αλγορίθμων αντικατάστασης των κρυφών εξυπηρετητών (στην ενότητα 3.1.2 βρίσκονται πληροφορίες σχετικά με δημιουργία συνθετικών ροών αιτήσεων). Στο τμήμα που ακολουθεί παρουσιάζονται τα αποτελέσματα με βάση τις μετρικές χρόνου απόκρισης και απαιτήσεις σε εύρος ζώνης.

7.3.1 Αποτελέσματα με Βάση το Χρόνο Απόκρισης

Ο χρόνος απόκρισης είναι η μετρική απόδοσης σε αυτή την περίπτωση. Το μέγεθος του δίσκου ορίστηκε να είναι κοντά στο 30% του μέγιστου χώρου που απαιτείται για να αποθηκευτούν όλα τα αντικείμενα που ζητούν οι χρήστες που είναι συνδεδεμένοι σε

κάθε κόμβο (στην περίπτωση βέβαια που προσπαθούμε να τοποθετήσουμε κρυφούς αντιπροσώπους).

Για το σενάριο **1**, παρατηρήσαμε ότι τοποθετώντας 4 κρυφούς αντιπροσώπους πετυχαίνουμε βελτίωση στην απόδοση κατά 56.02%. Αυτοί προτείνεται να τοποθετηθούν στους κόμβους 13, 10, 5, και 4 (στο Σχήμα 7-13 φαίνονται οι κρυφοί αντιπρόσωποι στους γκρι κόμβους και ο εξυπηρετητής στον μαύρο). Το εργαλείο πρότεινε την εφαρμογή του αλγορίθμου LRU για τον κρυφό αντιπρόσωπο στον κόμβο 13, ενώ στους υπόλοιπους κόμβους προτάθηκε ο GDS(1). Στον Πίνακα 7-1 φαίνονται αναλυτικά τα αποτελέσματα.



Σχήμα 7-13. Λύση του προβλήματος κατά το σενάριο 1.

α/α	Βελτίωση Απόδοσης	Κρυφός Αντιπρόσωπος	Ποσοστό Μεγέθους Δίσκου	Αλγόριθμος Αντικατάστασης
1	24.59 %	13	26.74 %	LRU
2	39.85 %	10	32.33 %	GDS(1)
3	49.26 %	5	47.07 %	GDS(1)
4	56.02 %	4	53.34 %	GDS(1)

Πίνακας 7-1. Αποτελέσματα τοποθέτησης σεναρίου 1.

Για το πρόβλημα του σεναρίου **2**, ο ProxyTeller προτείνει την τοποθέτηση 2 πλήρων αντιγράφων του περιεχομένου του εξυπηρετητή στους κόμβους 9 και 10, επιτυγχάνοντας με αυτό τον τρόπο βελτίωση της απόδοσης κατά 61.96%. Είναι προφανές ότι σε αυτή την περίπτωση επιτυγχάνουμε καλύτερη βελτίωση αλλά πρέπει να σκεφτεί κανείς το κόστος που πρέπει να πληρώσουμε για να εξοπλίσουμε εξυπηρετητές ικανούς να αποθηκεύσουν αντίγραφα όλων των αντικειμένων του εξυπηρετητή.

Στο σενάριο **3**, ας υποθέσουμε ότι έχουμε 3 εξυπηρετητές οι οποίοι είναι τοποθετημένοι στους κόμβους 1, 7 και 17 όπως φαίνεται στο Σχήμα 7-14.

Τα αποτελέσματα δείχνουν ότι πρέπει να τοποθετήσουμε 3 κρυφούς αντιπροσώπους στους κόμβους 5, 9 και 10 για να επιτύχουμε συνολική βελτίωση της απόδοσης της τάξεως του 52.24%. Και στους 3 κρυφούς αντιπροσώπους προτείνεται η εφαρμογή του αλγορίθμου αντικατάστασης GDS(1). Εάν αποφασίσουμε να τοποθετήσουμε πλήρη αντίγραφα (σενάριο **4**), τότε αυτά θα πρέπει να φιλοξενηθούν στους κόμβους 5 και 10 επιτυγχάνοντας βελτίωση κατά 65.17%. Βλέπουμε λοιπόν σε αυτή την περίπτωση, ότι μπορεί να μην αξίζει να τοποθετήσουμε πλήρη αντίγραφα, διότι με την τοποθέτηση κρυφών αντιπροσώπων επιτυγχάνουμε τον στόχο μας (βελτίωση της απόδοσης κατά 50%) ενώ παράλληλα αποφεύγουμε και το κόστος τοποθέτησης πλήρων αντιγράφων. Αυτό το κόστος μπορεί να εκφραστεί σαν την επένδυση σε αποθηκευτικό χώρο για κάθε αντίγραφο του περιεχομένου του εξυπηρετητή.

Ας υποθέσουμε για χάρη απλότητας ότι κάθε ένας από τους εξυπηρετητές διαχειρίζεται ποσότητα πληροφορίας ίση με S GBytes. Έτσι κάθε ένα από τους 2 εξυπηρετητές που πρόκειται να φιλοξενήσουν αντίγραφα των τριών αρχικών εξυπηρετητών, πρέπει να είναι κατάλληλα εξοπλισμένοι έτσι ώστε να μπορούν να αποθηκεύσουν $(3 \cdot S)$ GByte πληροφορία, και συνεπώς απαιτείται συνολικά

προτείνει την τοποθέτηση 4 κρυφών εξυπηρετητών στους κόμβους 9, 10, 5, και 6 επιτυγχάνοντας βελτίωση στην απόδοση κατά 53.88%. Τα αντίστοιχα αποτελέσματα με σκοπό την τοποθέτηση επιπλέον πλήρων αντιγράφων, δείχνουν ότι πρέπει να τοποθετήσουμε 2 αντίγραφα στους κόμβους 14 και 10 για να επιτύχουμε 56.41% βελτίωση στην απόδοση.

7.3.2 Αποτελέσματα με Βάση τις Απαιτήσεις σε Εύρος Ζώνης

Στα αποτελέσματα που ακολουθούν, επιλέχθηκαν οι απαιτήσεις σε εύρος ζώνης σαν μετρική απόδοσης. Σκοπός μας λοιπόν είναι να μειώσουμε όσο το δυνατόν περισσότερο την κίνηση του δικτύου. Παρουσιάζουμε τα αποτελέσματα των σεναρίων 1, 3 και 5 όπου προσπαθούμε να τοποθετήσουμε κρυφούς αντιπροσώπους. Επιλέγοντας σαν μετρική απόδοσης τις απαιτήσεις σε εύρος ζώνης (ή οποιαδήποτε από τις άλλες δύο), μπορεί να οδηγήσει σε διαφορετική επιλογή αλγορίθμου αντικατάστασης για καθένα από τους κρυφούς αντιπροσώπους και συνεπώς διαφορετική τιμή στο BHR κάτι που με τη σειρά του οδηγεί σε διαφορετικές αποφάσεις ως προς την τελική τοποθέτηση τους.

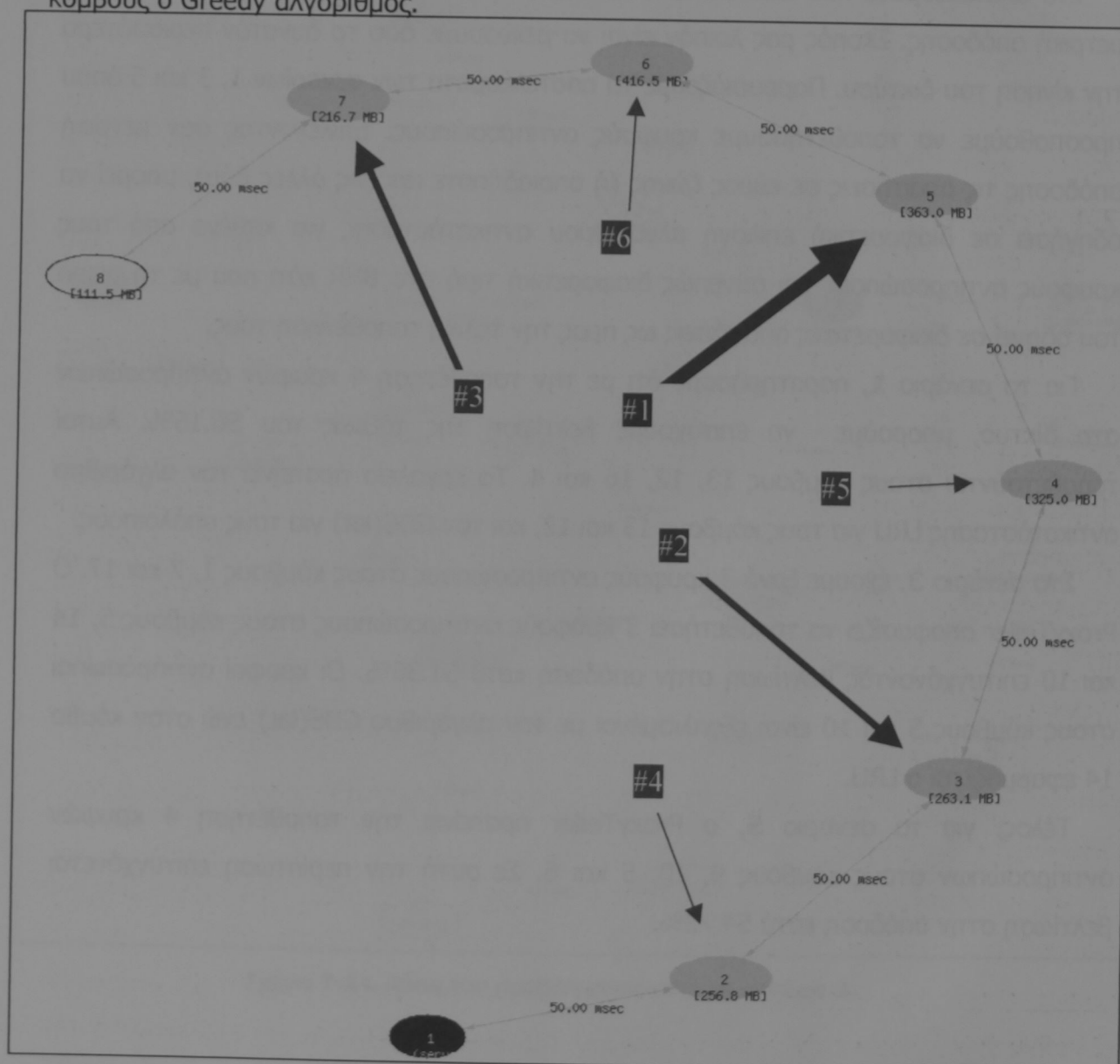
Για το σενάριο **1**, παρατηρήσαμε ότι με την τοποθέτηση 4 κρυφών αντιπροσώπων στο δίκτυο, μπορούμε να επιτύχουμε βελτίωση της τάξεως του 56.16%. Αυτοί τοποθετούνται στους κόμβους 13, 12, 16 και 4. Το εργαλείο προτείνει τον αλγόριθμο αντικατάστασης LRU για τους κόμβους 13 και 12, και τον GDS(lat) για τους υπόλοιπους.

Στο σενάριο **3**, έχουμε ξανά 3 κρυφούς αντιπροσώπους στους κόμβους 1, 7 και 17. Ο ProxyTeller αποφασίζει να τοποθετήσει 3 κρυφούς αντιπροσώπους στους κόμβους 5, 14 και 10 επιτυγχάνοντας βελτίωση στην απόδοση κατά 51.36%. Οι κρυφοί αντιπρόσωποι στους κόμβους 5 και 10 είναι εξοπλισμένοι με τον αλγόριθμο GDS(lat) ενώ στον κόμβο 14 εφαρμόζεται ο LRU.

Τέλος, για το σενάριο **5**, ο ProxyTeller προτείνει την τοποθέτηση 4 κρυφών αντιπροσώπων στους κόμβους 9, 10, 5 και 6. Σε αυτή την περίπτωση επιτυγχάνεται βελτίωση στην απόδοση κατά 54.72%.

7.3.3 Ενδεικτικό Παράδειγμα Εφαρμογής του Greedy Αλγορίθμου

Σε αυτή την περίπτωση ο ProxyTeller λαμβάνει δράση κάτω από το σενάριο **1**. Το μέγεθος του διαθέσιμου αποθηκευτικού χώρου είναι γύρω στο 35%, ενώ χρησιμοποιήθηκε ο ίδιος προσδοκώμενος φόρτος εργασίας με τις προηγούμενες περιπτώσεις. Στόχος μας είναι η βελτίωση της απόδοσης του συστήματος κατά 40% λαμβάνοντας υπόψη μας την μετρική χρόνου απόκρισης. Η τοπολογία που επιλέχθηκε για αυτή την περίπτωση είναι 8 κόμβοι σε σειρά, όπως φαίνεται στο Σχήμα 7-15. Σκοπός μας είναι να δούμε το τρόπο με τον οποίο επιχειρεί να επιλέξει τους καταλληλότερους κόμβους ο Greedy αλγόριθμος.



Σχήμα 7-15. Δίκτυο με 8 κόμβους μετά την τοποθέτηση 6 κρυφών αντιπροσώπων.

Όπως φαίνεται από τον Πίνακα 7-1, πρώτος κρυφός αντιπρόσωπος επιλέγεται ο κόμβος 5 όπου βρίσκεται στην μέση της αλυσίδας. Στη συνέχεια ο κόμβος 3 που βρίσκεται στο μέσο της αλυσίδας από τον κόμβο 5 μέχρι τον εξυπηρετητή, ακολουθεί ο κόμβος 7 που βρίσκεται στο μέσο του άλλου μισού κ.ο.κ.

α/α	Βελτίωση Απόδοσης	Κρυφός Αντιπρόσωπος	Ποσοστό Μεγέθους Δίσκου	Αλγόριθμος Αντικατάστασης
1	14.50 %	5	27.55 %	GDS(1)
2	25.65 %	3	38.01 %	GDS(1)
3	29.89 %	7	46.14 %	GDS(1)
4	32.87 %	2	38.94 %	GDS(1)
5	38.53 %	4	30.77 %	GDS(1)
6	43.57 %	6	24.01 %	GDS(1)

Πίνακας 7-2. Αποτελέσματα τοποθέτησης κρυφών αντιπροσώπων.

7.4 Χρόνος Εκτέλεσης του ProxyTeller

Στη μέχρι τώρα μας μελέτης χρησιμοποιήσαμε τοπολογίες που αντιστοιχούν στη ραχοκοκαλιά του δικτύου της AT&T και C&W και οι οποίες αποτελούνται από 18 και 21 κόμβους αντίστοιχα. Προκειμένου να έχουμε μια εικόνα του χρόνου που απαιτείται για την τοποθέτηση κρυφών αντιπροσώπων σε τοπολογίες δικτύων που αποτελούνται από πολύ περισσότερους κόμβους, ζητήσαμε από τον ProxyTeller να δώσει απάντηση εφαρμόζοντας τον Greedy αλγόριθμο σε δύο τοπολογίες που αποτελούνται από 600 και 1023 κόμβους. Η πολυπλοκότητα του Greedy αλγορίθμου είναι $O(N^2M)$ οπότε είναι προφανές ότι τόσο ο αριθμός των κόμβων όσο και ο αριθμός των αντιπροσώπων που πρόκειται να τοποθετηθούν (και κατ’ επέκταση το ποσοστό βελτίωσης της απόδοσης που επιθυμούμε) παίζουν βασικό ρόλο στο χρόνο εκτέλεσης.

Οι τοπολογίες δημιουργήθηκαν με τη βοήθεια του εργαλείου GT-ITM [49] χρησιμοποιώντας το transit-stub³ μοντέλο [50] το οποίο πλησιάζει την τοπολογία του διαδικτύου. Η ρουτίνα που είναι υπεύθυνη για την εφαρμογή του Greedy αλγορίθμου

³ Το συγκεκριμένο μοντέλο αποτελείται από ένα σύνολο από κόμβους οι οποίοι σχηματίζουν την ραχοκοκαλιά του δικτύου. Σε αυτούς είναι συνδεδεμένοι κόμβοι που σχηματίζουν με την σειρά τους μικρότερα δίκτυα.

εκτελέστηκε σε έναν υπολογιστή εξοπλισμένο με *Pentium III* στα 700 MHz και μνήμη 1 GB.

Και για τις δύο τοπολογίες ο ProxyTeller λαμβάνει δράση στο σενάριο 1, με μέγεθος αποθηκευτικού χώρου να κυμαίνεται από 20% ως 30%, με στόχο την μείωση του χρόνου απόκρισης και επιθυμητή βελτίωση κατά 60%. Ο προσδοκώμενος φόρτος εργασίας δημιουργήθηκε με την βοήθεια του SURGE [13] και αποτελείται από εκατοντάδες χιλιάδες αιτήσεις.

Τα αποτελέσματα έδειξαν ότι στην περίπτωση του δικτύου με τους 600 κόμβους, χρειάστηκαν 9 λεπτά για να τοποθετηθούν 37 κρυφοί αντιπρόσωποι και να επιτευχθεί βελτίωση της απόδοσης κατά 60.02%. Ο αντίστοιχος χρόνος για το δίκτυο των 1023 κόμβων είναι 25 λεπτά και τοποθετήθηκαν 19 αντιπρόσωποι επιτυγχάνοντας βελτίωση κατά 60.06%. Παρατηρούμε λοιπόν ότι ο αριθμός των κόμβων του δικτύου παίζει σημαντικό ρόλο στον χρόνο εκτέλεση.

Κεφάλαιο 8

Συμπεράσματα

Στα πλαίσια αυτής της διατριβής μελετήθηκε αναλυτικά η απόδοση των αλγορίθμων αντικατάστασης σε κρυφούς αντιπροσώπους διαδικτύου. Σχετικές μελέτες έδειξαν ότι με σκοπό να επιτευχθεί υψηλή απόδοση με βάση μία ή περισσότερες από τις μετρικές hit ratio, χρόνος απόκρισης, και απαιτήσεις σε εύρος ζώνης είναι αναγκαίο: i) ο αλγόριθμος αντικατάστασης να συμπεριλαμβάνει στα κριτήρια με τα οποία θα πάρει τις αποφάσεις του, το μέγεθος του αντικειμένου, το χρόνο που απαιτείται για να ανακτηθεί το αντικείμενο από το μακρινό εξυπηρετητή και την εκτιμώμενη δημοτικότητα του αντικειμένου, ii) η χρήση μια βοηθητικής κρυφής μνήμης, η οποία κρατάει πληροφορία για τα αντικείμενα, ενώ παράλληλα ενεργεί και σαν φίλτρο, και η χρήση ενός μηχανισμού ελέγχου εισόδου που μειώνει ακόμα περισσότερο τις ακατάλληλες διαγραφές αντικειμένων, και iii) να εκμεταλλεύονται οι γειτονικοί κρυφοί αντιπρόσωποι κτίζοντας έτσι δίκτυα από συνεργαζόμενους αντιπρόσωπους πράγμα το οποίο μπορεί να προσφέρει επιπλέον βελτίωση της απόδοσης.

Τα αποτελέσματα μας, έδειξαν ότι η χρήση της μετρικής hit ratio για πολύπλοκους αλγόριθμους όπως ο CSP, μπορεί να οδηγήσει σε παραπλανητικές εκτιμήσεις της απόδοσης του συστήματος. Είδαμε για παράδειγμα μικρές διαφορές στις τιμές του hit ratio, να μεταφράζονται σε μεγάλες διαφορές στο χρόνο απόκρισης και στις

απαιτήσεις σε εύρος ζώνης. Ένα ακόμα συμπέρασμα είναι ότι η βελτίωση στην απόδοση που παρατηρείται εξαιτίας του συνεργαζόμενου περιβάλλοντος, είναι μάλλον μικρή, τόσο όσον αφορά τον χρόνο απόκρισης, όσο και τις απαιτήσεις σε εύρος ζώνης. Αυτό ισχύει παρά το γεγονός ότι όπως και σε άλλες μελέτες παρατηρήσαμε αύξηση της απόδοσης όσον αφορά το hit ratio η οποία αγγίζει το 30%.

Όσον αφορά τα βασικά στοιχεία του αλγορίθμου αντικατάστασης, παρατηρήσαμε ότι η βοηθητική κρυφή μνήμη, ο όρος της δημοτικότητας, και ο όρος του κόστους ανάκτησης του αντικειμένου, παίζουν βασικό ρόλο στην απόδοση του CSP. Από την άλλη ο όρος μέγεθος αντικειμένου και ο μηχανισμός ελέγχου εισόδου, δείχνουν να επηρεάζουν περισσότερο το hit ratio και μόνο κατά ένα μικρό μέρος το χρόνο απόκρισης και τις απαιτήσεις σε εύρος ζώνης.

Εκμεταλλευόμενοι τα προηγούμενα αποτελέσματα πάνω στην απόδοση των κρυφών αντιπροσώπων, μπορέσαμε να δώσουμε μια γενικότερη εικόνα της λύσης του προβλήματος δείχνοντας ποιος αλγόριθμος είναι ο προτιμότερος σαν συνάρτηση της κατανομής της δημοτικότητας, του απαιτούμενου αποθηκευτικού χώρου, και μετρικής απόδοσης που μας ενδιαφέρει. Έχοντας αυτή την γενική εικόνα, αναπτύξαμε έναν δυναμικό αλγόριθμο αντικατάστασης, ο οποίος επιλέγει τον προτιμότερο αλγόριθμο αντικατάστασης σύμφωνα με τα χαρακτηριστικά του περιβάλλοντος δράσης του κρυφού αντιπροσώπου. Η απόδοση του δείχνει να είναι ιδιαίτερα ικανοποιητική σε σχέση με τον ευρέως χρησιμοποιούμενο LRU.

Στη συνέχεια, παρουσιάσαμε τον ProxyTeller, ένα νέο εργαλείο που παίρνει αποφάσεις σχετικά με την αποδοτική τοποθέτηση των κρυφών αντιπροσώπων ή αντιγράφων, δίνοντας έτσι απάντηση στις σημαντικές ερωτήσεις που προκύπτουν κατά την εξέταση της αποδοτικής παράδοσης του περιεχομένου των εξυπηρετητών. Ο ProxyTeller είναι βασισμένος : α) στη λεπτομερή μελέτη της απόδοσης των κρυφών αντιπροσώπων, και πως αυτή εξαρτάται από τους αλγορίθμους αντικατάστασης και τους σχετικούς μηχανισμούς, και β) στην προσαρμογή των αλγορίθμων τοποθέτησης πλήρων αντιγράφων εξυπηρετητών σε τοποθέτηση κρυφών αντιπροσώπων (μερικά αντίγραφα) λαμβάνοντας υπόψη τα αποτελέσματα από την προηγούμενη μελέτη.

Συγκεκριμένα, το πρόβλημα που ο ProxyTeller λύνει δέχεται ως είσοδο την τοπολογία του δικτύου στο οποίο οι κρυφοί αντιπρόσωποι θα τοποθετηθούν, το μέγεθος του διαθέσιμου αποθηκευτικού χώρου, τα χαρακτηριστικά του φόρτου εργασίας της κοινότητας που εξυπηρετείται από ένα κρυφό αντιπρόσωπο και τη βελτίωση στην απόδοση που επιδιώκεται. Η έξοδος του ProxyTeller είναι η θέση και ο αριθμός των κρυφών αντιπροσώπων που απαιτούνται προκειμένου να επιτευχθούν οι επιθυμητοί στόχοι απόδοσης.

Το εργαλείο μπορεί να χρησιμοποιηθεί για κρυφούς αντιπροσώπους (τόσο διαδικτύου όσο και δικτύων παροχής περιεχομένου (CDN)) και, ελπίζουμε ότι θα είναι χρήσιμο:

- i. σε ISPs/CDNs, προκειμένου να αποφασίσουν καλύτερα σχετικά με τις απαραίτητες επενδύσεις που διατίθενται να κάνουν προκειμένου να επιτύχουν τις επιθυμητές βελτιώσεις στην απόδοση,
- ii. στους παροχείς περιεχομένου, που θα είναι σε θέση να υπολογίσουν την μείωση του φόρτου εργασίας του εξυπηρετητή τους μετά από μια πιθανή σύμβαση με ένα CDN και τις αναμενόμενες βελτιώσεις απόδοσης που θα απολαμβάνουν οι καταναλωτές του περιεχομένου τους, και
- iii. στους τελικούς χρήστες, που είναι σε θέση να καταλάβουν την απόδοση των πιθανών εφαρμογών που δημιουργούν ή/και χρησιμοποιούν, και έχουν πρόσβαση σε τέτοιο περιεχόμενο.

Ας σημειωθεί επίσης ότι τα διαφορετικά συμβαλλόμενα μέρη που συμμετέχουν σε μια συμφωνία παράδοσης περιεχομένου ενδιαφέρονται για διαφορετικές μετρικές απόδοσης. Οι διαχειριστές των εξυπηρετητών ενδιαφέρονται κυρίως για το πόσο μειώνεται ο φόρτος εργασίας των εξυπηρετητών τους και ο χρόνος απόκρισης των χρηστών τους. Με τη χρησιμοποίηση του ProxyTeller με αυτές τις μετρικές, θα μπορέσουν να δουν ποια είναι τα οφέλη μιας πιθανής συνεργασίας με ένα CDN. Αντίθετα, οι CDNs ενδιαφέρονται κυρίως για τις επενδύσεις σε εύρος ζώνης και αποθηκευτικό χώρο. Χρησιμοποιώντας τον ProxyTeller με τις απαιτήσεις σε εύρος ζώνης σαν μετρική απόδοσης, θα είναι σε θέση να ξέρουν πόσο εύρος ζώνης μπορούν να κερδίσουν και με ποιο κόστος σε αποθηκευτικό χώρο. Συνολικά λοιπόν, πιστεύουμε ότι ο ProxyTeller είναι χρήσιμος για όλα τα ενδιαφερόμενα μέρη. Κατά συνέπεια, η σημασία αυτής της εργασίας βρίσκεται κυρίως στη δυνατότητά της να απαντήσει σε θέματα τοποθέτησης υψηλού ενδιαφέροντος.

Όπως αναφέρεται, έχουμε αναπτύξει το εργαλείο, καθιστώντας το διαθέσιμο μέσω διαδικτύου, προς χρήση από την κοινότητα [39]. Ο τελευταίος στόχος μας είναι, μέσω του ProxyTeller, να συσσωρεύσουμε και να συνδυάσουμε τα σχετικά ερευνητικά αποτελέσματα, τα οποία αποτελούν τη βάση του (δηλ., νέα αποτελέσματα για την απόδοση των διάφορων μηχανισμών που βελτιώνουν την απόδοση των κρυφών αντιπροσώπων, και νέα αποτελέσματα για την τοποθέτηση κρυφών αντιπροσώπων / αντιγράφων) με σκοπό, διαθέτοντας την απαραίτητη υποδομή, να δώσουμε απαντήσεις σε ερωτήσεις κλειδιά που αφορούν την αποδοτική παράδοση περιεχομένου. Επίσης σκοπεύουμε να μελετήσουμε την απόδοση περισσότερων αλγορίθμων τοποθέτησης, έτσι ώστε να εφαρμόσουμε τους καλύτερους από αυτούς στο εργαλείο που αναπτύξαμε. Επίσης επιθυμούμε να επεκτείνουμε τις δυνατότητες του εργαλείου από πλευράς αλληλεπίδρασης με σκοπό να γίνει περισσότερο εύχρηστο.

Βιβλιογραφία

- [1] C. Aggarwal, J. L. Wolf and P. S. Yu, "Caching on the World Wide Web", In IEEE Transactions on Knowledge and Data Engineering, vol. 11, no. 1, January/February 1999.
- [2] J. Shim, P. Scheuermann and R. Vingralek, "Proxy Cache Algorithms: Design, Implementation and Performance", In IEEE Transactions on Knowledge and Data Engineering, vol. 11, no. 4, July/August 1999.
- [3] S. Williams, M. Abrams, C. R. Standridge, G. Abdulla and E. A. Fox, "Removal Policies in Network Caches for World-Wide Web Documents", In Proceedings of the ACM SIGCOMM, pp. 293-305, 1996.
- [4] L. Rizzo and L. Vicisano, "Replacement Policies for a Proxy Cache", Research Note RN/98/13, Department of Computer Science, University College London, 1998.
- [5] E. J. O'Neil, P. E. O'Neil, and G. Weikum, "An Optimal Proof of the LRU-K Page Replacement Algorithm", In Journal of the ACM, Vol. 46 No. 1, 1999.
- [6] D. Neal, "The Harvest Object Cache in New Zealand", In Proceedings of the 5th International WWW Conference, May 1996.
- [7] A. Chankhunthod, P. Danzig and C. Neerdaels, "A Hierarchical Internet Object Cache", In Proceedings of the USENIX Technical Conference, San Diego, CA, January 1996.
- [8] S. Gadde, J. Chase and M. Rabinovich, "A Taste of Crispy Squid", In Proceedings of the Workshop on Internet Server Performance (WISP'98), Madison, WI, June 1998.
- [9] S. Gadde, J. Chase and M. Rabinovich, "Reduce, Reuse, Recycle: An Approach to Building Large Internet Caches", In Proceedings of the Sixth Workshop on Hot Topics in Operating Systems (HotOS-VI), pages 93-98, May 1997.

- [10] R. Tewari, M. Dahlin, H. M. Vin and J. S. Kay, "Design Considerations for Distributed Caching on the Internet", Technical Report TR98-04, Dept of Computer Science, University of Texas at Austin, 1998.
- [11] M. S. Raunak, P. Shenoy, P. Goyal and K. Ramamritham, "Implications of Proxy Caching for Provisioning Networks and Servers", In Proceedings of the ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS 2000), pages 66-77, Santa Clara, CA, June 2000.
- [12] M. Arlitt and C. Williamson, "Web Server Workload Characterization: The Search for Invariants", In Proceedings of the ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems, May 1996.
- [13] P. Barford and M. Grovella, "Generating Representative Web Workloads for Network and Server Evaluation", In Proceedings of the ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems, pages 151-160, 1998.
- [14] L. Breslau, P. Cao, L. Fan, G. Phillips and S. Shenker, "Web Caching and Zipf-like Distributions: Evidence and Implications", In Proceedings of the IEEE INFOCOM, 1999.
- [15] M. A. Habib and M. Abrams, "Analysis of Sources of Latency in Downloading Web Pages", In Proceedings of WebNet 2000, San Antonio USA, November 2000.
- [16] S. Acharya and B. Smith, "MiddleMan: A Video Caching Proxy Server", In Proceedings of the ACM NOSSDAV Conference, 2000.
- [17] Weekly Access Logs at NLNR's Proxy Caches, available online at <ftp://ircache.nlanr.net/Traces/>
- [18] R. Tewari, H. M. Vin, A. Dan and D. Sitaram, "Resource-based Caching for Web Servers", In Proceedings of the SPIE/ACM Conference on Multimedia Computing and Networking, January 1998.
- [19] P. Cao and S. Irani, "Cost-Aware WWW Proxy Caching Algorithms", In Proceedings of the USITS Conference, 1997.
- [20] L. Fan, P. Cao, J. Almeida and A. Broder, "Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol", In Proceedings of the ACM SIGCOMM'98 Conference, Feb 1998.
- [21] S. Jin and A. Bestavros, "GreedyDual* Web Caching Algorithm: Exploiting the Two Sources of Temporal Locality in Web Request Streams." In Proceedings of the 5th International Web Caching and Content Delivery Workshop, May 2000.
- [22] M. Abrams, C. Standridge, G. Abdulla, S. Williams, and E. Fox, "Caching Proxies: Limitations and Potentials", In Proceedings of 1995 World Wide Web Conference, December 1995.

- [23] B. Krishnamurthy and C. E. Wills, "Analyzing factors that influence end-to-end Web performance", In Proceedings of the 2000 World Wide Web Conference / Computer Networks, May 2000.
- [24] M. Rabinovich and O. Spatscheck, "Web Caching and Replication", Addison-Wesley, ISBN: 0-201-61570-3, 2002.
- [25] A. Feldmann, R. Caceres, F. Douglass, G. Glass, M. Rabinovich, "Performance of Web Proxy Caching in Heterogeneous Bandwidth Environments", In Proceedings of the IEEE INFOCOM, pp. 107-116, 1999.
- [26] T. M. Kroeger, D. D. E. Long, and J. C. Mogul, "Exploring the Bounds of Web Latency Reduction from Caching and Prefetching", In Proceedings of the USENIX Symposium on Internet Technologies and Systems, pp. 13-22.
- [27] Squid proxy server. Available online at: <http://www.squid-cache.org/>
- [28] Apache proxy server. Available online at: <http://www.apache.org/>
- [29] S. Gadde, J. Chase, and M. Rabinovich, "Directory Structures for Scalable Internet Caches", Technical Report CS-1997-18, Department of Computer Science, Duke University, November 1997.
- [30] A. Dingle, "Web Cache Coherence", In proceedings of the fifth International World Wide Web Conference May 6-10, 1996.
- [31] V. N. Padmanabhan and J. C. Mogul, "Using Predictive Prefetching to Improve World-Wide Web Latency", ACM Computer Communication Review, 26(3): 22-36, July 1996.
- [32] P. Triantafillou and I. Aekaterinidis, "Web Proxy Cache Replacement: Do's, Don'ts, and Expectations", In Proceedings of The 2nd IEEE International Symposium on Network Computing and Applications (NCA-03), April 2003.
- [33] A. Venkataramani, P. Weidmann, and M. Dahlin, "Bandwidth constrained placement in a WAN", Twentieth ACM Symposium on Principles of Distributed Computing (PODC 2001), August 2001.
- [34] B. Krishnamurthy and J. Wang, "On network-aware clustering of web clients", In Proceedings of ACM Sigcomm 2000, August 2000.
- [35] B. Krishnamurthy, C. Wills, and Y. Zhang, "On the use and performance of content distribution networks", In Proceedings of the ACM SIGCOMM Internet Measurement Workshop, San Francisco, November 2001.
- [36] B. Li, M. J. Golin, G. F. Italiano, and X. Deng, "On the placement of Web proxies in the Internet", in Proceedings of Infocom 2000, March 2000.

- [37] L. Qiu, V. N. Padmanabhan, and G. M. Voelker, "On the placement of Web server replicas", in Proceedings of IEEE Infocom 2001, April 2001.
- [38] M. Charikar, and S. Guha, "Improved Combinatorial Algorithms for the Facility Location and K-Median Problems", in Proceedings of the 40th Annual IEEE Conference on Foundations of Computer Science, 1999.
- [39] ProxyTeller: A Proxy Cache Placement Tool. Demo available at <http://pcp.softnet.tuc.gr>
- [40] Akamai Content Delivery Network. <http://www.akamai.com/>
- [41] AT&T Content Delivery Network. <http://www.att.com/>
- [42] Digital Island Content Delivery Network. <http://www.digisle.net/>
- [43] P. Krishnan, Danny Raz, and Yuval Shavitt, "The cache location problem", IEEE/ACM Transactions on Networking, 8(5), 2000.
- [44] AT&T IP Backbone Network Performance. <http://ipnetwork.bgtmo.ip.att.net/>
- [45] C&W Backbone Network Performance. <http://sla.cw.net/sla/index.jsp>
- [46] M. Karlsson, C. Karamanolis, and M. Mahalingam, "A Framework for Evaluating Replica Placement Algorithms", Hewlett-Packard Laboratories.
- [47] A. Wolman, G. Voelker, N. Sharma, N. Cardwell, M. Brown, T. Landray, D. Pinnel, A. Karlin, and H. Levy, "Organization-Based Analysis of Web-Object Sharing and Caching", In Proceedings of the USENIX Symposium on Internet Technologies and Systems, 1999.
- [48] I. Cidon, S. Kuten, and R. Soffer, "Optimal Allocation of Electronic Content", In Proceedings of the INFOCOM 2001, 2001.
- [49] GT-ITM: Georgia Tech Internetwork Topology Models.
<http://www.cc.gatech.edu/projects/gtitm/>
- [50] K. Calvert, M. Doar and E. W. Zegura, "Modeling Internet Topology", IEEE Communications Magazine, June 1997.
- [51] P. Triantafillou and I. Aekaterinidis, "ProxyTeller: A Tool for Guiding Web Proxy Cache Placement Decisions", Accepted for poster presentation for the 12th International World Wide Web Conference (WWW2003), May 2003.

Ειδικό Λεξιλόγιο

A

- | | |
|------------------------------------|--------------------|
| Ανακατεύθυνση | - Redirection |
| Αντίγραφα Περιεχομένου Εξυπηρετητή | - Replicas |
| Αντιγραφή | - Replication |
| Ασθενής Συνέπεια | - Weak Consistency |

Δ

- | | |
|-----------------------------|-----------------------------------|
| Διαδίκτυο | - Internet |
| Διαφανής | - Transparent |
| Διαχειριστής | - Administrator |
| Δίκτυα Παροχής Περιεχομένου | - CDN – Content Delivery Networks |
| Δικτυακός τόπος | - Web Site |
| Δρομολογητής | - Router |

Ε

- | | |
|-------------------------|------------------|
| Εξερευνητής Ιστοσελίδων | - Browser |
| Εξυπηρετητής Αντιγράφων | - Replica Server |
| Εξυπηρετητής Διαδικτύου | - Server |
| Εύρος Ζώνης | - Bandwidth |

Ι

- | | |
|----------------------------|----------------------|
| Ιδιότητα Χρονικής Περιοχής | - Temporal Locality |
| Ιεραρχία Υποδείξεων | - Hint Hierarchy |
| Ιστοσελίδα | - Web Page |
| Ισχυρή Συνέπεια | - Strong Consistency |

Κ

- | | |
|---------------------|---------------|
| Κρυφός Αντιπρόσωπος | - Proxy Cache |
| Κρυφή Μνήμη | - Cache |

Π

Παροχέας Υπηρεσιών Διαδικτύου

Πελάτης - Χρήστης Διαδικτύου

Προσωρινή Αποθήκευση Αντικειμένων

Πρώθηση Αντικειμένων πριν Ακόμα Ζητηθούν

Πρώθηση με Βάση τις Υποδείξεις του Εξυπηρετητή

- ISP - Internet Service Provider
- Clients
- Caching
- Prefetching
- Server Hint Prefetching

Ρ

Ροή Αιτήσεων

- Request Stream

Σ

Συμβουλευτικός Εξυπηρετητής

Σύνδεσμος

Συνέπεια των Αντικειμένων

Σύστημα Ελέγχου Πρόσβασης και

Προστασίας Τοπικού Δικτύου

- Mapping Server
- Hyperlinks
- Consistency
- Firewall

Τ

Τοπική Πρώθηση

Τοπικό Κτύπημα

(εξυπηρέτηση των αιτήσεων από τον δίσκο του κρυφού αντιπροσώπου)

- Local Prefetching
- Local Hit

Φ

Φόρτος Εργασίας Εξυπηρετητή

- Server Load

Χ

Χρόνος Απόκρισης

- Latency



