ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

Συγκριτική Μελέτη δικτύων ATM με εισιτήρια και Wormhole υπό κίνηση εκρηκτική και με θερμά σημεία

Εμμανουήλ Σπυριδάκης

Μεταπτυχιακή Εργασία

Ηράκλειο, Μάιος 1996

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

Συγκριτική Μελέτη δικτύων ATM με εισιτήρια και Wormhole υπό κίνηση εκρηκτική και με θερμά σημεία

Εργασία που υποβλήθηκε από τον
Εμμανουήλ Σπυριδάκη
ως μερική εκπλήρωση των απαιτήσεων
για την απόκτηση
ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΙΔΙΚΕΥΣΗΣ

Συγγραφέας:

Εμμανουήλ Σπυριδάκης
Τμήμα Επιστήμης Υπολογιστών

Εισηγητική Επιτροπή:

Δημήτριος Σερπάνος, Επίκουρος Καθηγητής, Επόπτης

Μανόλης Κατεβαίνης, Αναπληρωτής Καθηγητής, Μέλος

Γεώργιος Σταμούλης, Επίκουρος Καθηγητής, Μέλος

Δεκτή:

Πάνος Κωνσταντόπουλος, Αναπληρωτής Καθηγητής
Πρόεδρος Επιτροπής Μεταπτυχιακών Σπουδών

Ηράκλειο, Μάιος 1996

Συγκριτική Μελέτη δικτύων ATM με εισιτήρια και Wormhole υπό κίνηση εκρηκτική και με θερμά σημεία

Εμμανουήλ Σπυριδάκης

Μεταπτυχιακή Εργασία

Τμήμα Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

ΠΕΡΙΛΗΨΗ

Καθώς τα δίκτυα των υπολογιστών αρχίζουν να πλησιάζουν τις επιδόσεις των δικτύων επεξεργαστών, οι μέθοδοι δρομολόγησής τους μπορούν να χρησιμοποιηθούν σε πολυεπεξεργαστές. Η μέθοδος ATM, που είναι δημοφιλής στα δίκτυα, όταν συνδιαστεί με έλεγχο ροής εισιτηρίων μπορεί να χρησιμοποιηθεί σε δίκτυα επεξεργαστών που χρησιμοποιούν άλλες μεθόδους, όπως η μέθοδος wormhole που υλοποιείται από πολλά συστήματα. Ο τρόπος όμως διακίνησης των πληροφοριών μεσα σε ένα δίκτυο ATM με έλεγχο ροής εισιτηρίων είναι διαφορετικός από τον τρόπο διακίνησης σε ένα δίκτυο wormhole, και σε αυτό οφείλονται και οι διαφορές στις επιδόσεις των δικτύων αυτών. Στην εργασία αυτή συγκρίνουμε τις επιδόσεις δικτύων που αποτελούνται από μεταγωγείς που χρησιμοποιούν την μέθοδο ATM με έλεγχο ροής υλοποιημένο με εισιτήρια, με δίκτυα που αποτελούνται από μεταγωγείς που χρησιμοποιούν την μέθοδο wormhole. Υλοποιήθηκε ένας προσομοιωτής για την λειτουργία μεταγωγέων και των δύο προαναφερθέντων μεθόδων, που η λεπτομέρειά του είναι σε επίπεδο κύκλου ρολογιού. Τα μοντέλα που προσομοιώθηκαν είναι όσο το δυνατό πιο ρεαλιστικά. Μελετάμε την καθυστέρηση που υφίστανται πακέτα καθώς και κύτταρα (cells) πακέτων από τα παραπάνω δίκτυα καθώς επίσης και την παροχή (throughput) αυτών των δικτύων. Οι συνθήκες κίνησης κάτω από τις οποίες εξετάζονται τα δίκτυα, είναι εκρηκτική κίνηση (bursty traffic) και κίνηση με θερμά σημεία (hot spots). Μελετάμε ακόμα το πως επιρρεάζονται οι επιδόσεις των δικτύων καθώς αυτά ή οι μεταγωγείς που τα αποτελούν, γίνονται μεγαλύτεροι. Όπως δείχνουν τα αποτελέσματα το δίκτυο στην περίπτωση του ATM συμπεριφέρεται καλύτερα από εκείνο στην περίπτωση του wormhole, παρουσιάζοντας μικρότερες καθυστερήσεις στην κίνησή του καθώς και καλύτερη παροχή σε ανάλογες διαμορφώσεις των πειραμάτων. Ειδικά στην περίπτωση του ATM, τα δίκτυα που παρέχουν λωρίδες κυκλοφορίας των δεδομένων έχουν σημαντική βελτίωση στην καθυστέρηση που υποβάλλουν την κίνηση που διέρχεται από αυτά.

Επόπτης : Δημήτριος Σερπάνος, Επίκουρος Καθηγητής

Comparative Study of Credit Flow Controled ATM and Wormhole Networks Under Bursty Traffic and With Hot Spots

Emmanouil Spyridakis

Master of Science Thesis

Department of Computer Science
University of Crete

ABSTRACT

As the performance of computer networks becomes similar to the performance of multiprocessors, the efficiency of the routing methods used, is also converging. Methods used for networks can also be applied for a high performance network having similar performance to a multiprocessor. The ATM method, which is frequently used in computer networks, combined with credit based flow control can be used in multiprocessors where popular routing method is wormhole. Data flow is different for the two methods and that is the main reason that the two methods have different performance. In our work we measure and compare the performance of ATM with credit based flow control and wormhole. The two kind of switches were modeled and a simulator was created based on those models. The models are detailed, realistic and they describe the operations of a switch in one clock cycle. We study the cell delay and throughput for networks consisting of either ATM or wormhole switches under uniformly distributed (poisson), bursty traffic and with hot spots at the exits of the network. We also study the performance as the network or the switches become larger. The results show that a network consisting of ATM switches have better performance than the one consisting of wormhole switches, having smaller delays and better throughput in analogous configurations. For ATM networks an improvement of performace is achieved when more than one lane is available in the network.

Supervisor : Dimitris Serpanos, Assistant Professor

Ευχαριστίες

Θεωρώ χρέος μου να ευχαριστήσω ορισμένους ανθρώπους που με βοήθησαν σημαντικά στην ολοκλήρωση αυτής της εργασίας είτε σε τεχνικά είτε σε επιστημονικά σημεία.

Κατ' αρχήν οφείλω ένα ευχαριστώ στον κ. Μανόλη Κατεβαίνη και στον κ. Δημήτρη Σερπάνο για την βοήθειά τους τόσο στην επιλογή του θέματος της εργασίας όσο και για την βοήθειά τους σε δύσκολα θεωρητικά σημεία της εργασίας. Ιδιαίτερα, μεγάλη βοήθεια είχα από τον κ. Σερπάνο, που με επέβλεψε κατά το μεγαλύτερο μέρος της εργασίας, και μου αφιέρωσε πολλές ώρες στο γραφείο του, αλλά και εκτός αυτού, για να συζητήσουμε θεωρητικά αλλά και τεχνικά θέματα σχετικά με την εργασία.

Το μέλος της εξεταστικής επιτροπής κ. Γεώργιος Σταμούλης μου έδωσε πολύ εκτενή και ουσιαστικά σχόλια σχετικά με την εργασία μου. Οι υποδείξεις του συνέβαλαν στην βελτίωση της συγγραφής καθώς και της παρουσίασης της εργασίας.

Ο Μανόλης Μαραζάκης με βοήθησε πολλές φορές σε τεχνικά ζητήματα που κυρίως είχαν σχέση με θέματα προσομοίωσης αλλά με στήριξε και ηθικά όταν τα πράγματα δεν πήγαιναν κατ' ευχήν.

Θα ήταν παράληψη να μην ευχαριστήσω τους γονείς μου, Νίκο και Ελένη για την ηθική αλλά και υλική υποστήριξη που μου παρείχαν όλα τα χρόνια των σπουδών μου.

Ευχαριστώ τέλος, το Ινστιτούτο Πληροφορικής του Ιδρύματος Τεχνολογίας και Έρευνας για την τεχνική και οικονομική υποστήριξη που μου παρείχε κατά τη διάρκεια των μεταπτυχιακών μου σπουδών και το Τμήμα Επιστήμης Υπολογιστών του Πανεπιστημίου Κρήτης για το υψηλό επίπεδο σπουδών που μού προσέφερε.

Περιεχόμενα

Περίληψη **i**

Ευχαριστίες **v**

Περιεχόμενα **vii**

Κατάλογος Πινάκων **ix**

Κατάλογος Σχημάτων **xi**

1 Εισαγωγή **1**
1.1 Ιστορική αναδρομή . . . 1
1.2 Asynchronous Transfer Mode (ATM) . . . 4
1.3 Wormhole routing . . . 5
1.4 Συμφόρηση δικτύων – Έλεγχος ροής . . . 5
1.5 Το πρόβλημα που μελετήθηκε . . . 8

2 Δίκτυα ATM **9**
2.1 Μέθοδοι ελέγχου ροής με εισιτήρια . . . 9
2.2 Εικονικές συνδέσεις και ομάδες ροής . . . 10
2.3 Μέθοδος ελέγχου ροής του δικτύου ATM . . . 12
2.4 Συγχώνευση ομάδων ροής και οικονομία πόρων . . . 12

3 Προσομοίωση **15**
3.1 Περιγραφή μοντέλου μεταγωγέα ATM . . . 15
3.2 Περιγραφή μοντέλου μεταγωγέα Wormhole . . . 18
3.3 Περιγραφή μοντέλου παραγωγού για δίκτυα ATM . . . 23
3.4 Περιγραφή μοντέλου παραγωγού για δίκτυα Wormhole . . . 23
3.5 Περιγραφή μοντέλου καταναλωτή . . . 24
3.6 Περιγραφή της βιβλιοθήκης υποστήριξης . . . 25

4 Πειραματικά αποτελέσματα **27**
4.1 Μεταβολή στα μοντέλα κίνησης εισόδου . . . 29
4.1.1 Κίνηση Poisson . . . 29
4.1.2 Εκρηκτική κίνηση . . . 31
4.1.3 Κίνηση με θερμά σημεία . . . 34
4.2 Μεταβολή στις παραμέτρους του δικτύου . . . 37
4.2.1 Μεταβολή μεγέθους δικτύου . . . 37

4.2.2 Μεταβολή μεγέθους μεταγωγέα . . . 39

5 Συμπεράσματα **41**

Βιβλιογραφία **43**

Παράρτημα Α **44**
.1 Εισαγωγή . . . 45
.2 Χρήση . . . 45
.2.1 Είσοδος . . . 45
.2.2 Τοπολογία δικτύου . . . 47
.2.3 Πίνακας δρομολόγησης . . . 48
.2.4 Έξοδος . . . 49
.3 Τεχνικές πληροφορίες . . . 50
.3.1 Συνάρτηση `sim` . . . 51
.3.2 Συνάρτηση `Medium` . . . 53
.3.3 Συνάρτηση `producer_code` . . . 53
.3.4 Συνάρτηση `packet_generator_code` . . . 54
.3.5 Συνάρτηση `credit_collector_code` . . . 55
.3.6 Συνάρτηση `switch_code` . . . 55
.3.7 Συνάρτηση `consumer_code` . . . 55

Κατάλογος Πινάκων

Κατάλογος Σχημάτων

1.1 Ιεραρχική δομή δικτύου . . . 3
1.2 Παραδείγματα τοπολογιών δικτύων παράλληλων υπολογιστών . . . 4
1.3 Συνθήκες για αποδοτική λειτουργία . . . 7

2.1 Εικονικές συνδέσεις και ομάδες ροής. . . . 11
2.2 Έλεγχος ροής με δύο εισιτήρια . . . 13

3.1 Μοντέλο μεταγωγέα ATM 2×2 . . . 16
3.2 Μοντέλο μεταγωγέα wormhole 2×2 . . . 19
3.3 Μοντέλο παραγωγού για την περίπτωση του ATM . . . 23
3.4 Μοντέλο παραγωγού στην περίπτωση του wormhole . . . 24

4.1 Μεταβολή της παροχής θεωρητικά και σε προσομοίωση . . . 27
4.2 Παροχή κορεσμού με κίνηση Poisson . . . 29
4.3 Input vs Central Buffering . . . 30
4.4 Σύγκριση υπό συνθήκες κίνησης Poisson χαμηλού φορτίου . . . 31
4.5 Συγκριση υπό υψηλού φορτίου κίνηση Poisson . . . 32
4.6 Σύγκριση καθυστερήσεων όταν αλλάζει το μέγεθος πακέτου . . . 33
4.7 Παροχή κορεσμού για διαφορετικά μεγέθη πακέτων . . . 33
4.8 Παροχή κορεσμού wormhole με μία λωρίδα . . . 34
4.9 Καθυστέρηση για κίνηση με διαφορετικά πλήθη θερμών σημείων . . . 35
4.10 Απόστάσεις προορισμών σε ένα δίκτυο Banyan 16×16 . . . 36
4.11 Καθυστέρηση σε συνάρτηση της απόστασης από το θερμό σημείο . . . 37
4.12 Μεταβολή καθυστέρησης καθώς μεταβάλλεται το μέγεθος του δικτύου . . . 38
4.13 Μεταβολή παροχής κορεσμού καθώς μεταβάλλεται το μέγεθος του δικτύου . . . 38
4.14 Μεταβολή συνολικής καθυστέρισης καθώς μεταβάλλεται το μέγεθος του μεταγωγέα . . . 39
4.15 Μεταβολή παροχής κορεσμού καθώς μεταβάλλεται το μέγεθος του μεταγωγέα . . . 40

Κεφάλαιο 1

Εισαγωγή

Τα τελευταία χρόνια, η ανάπτυξη των δικτύων των υπολογιστών είναι μεγάλη λόγω της προόδου στις επιδόσεις των γραμμών διασύνδεσης. Οι επιδόσεις ιδίως στον τομέα των τοπικών δικτύων αρχίζουν να πλησιάζουν τις επιδόσεις δικτύων διασύνδεσης μέσα σε παράλληλους υπολογιστές. Τα δίκτυα υπολογιστών αρχικά χρησιμοποιήθηκαν για να συνδέονται μεταξύ τους σταθμοί εργασίας και τερματικά και να μοιράζονται αποτελεσματικά δεδομένα και συσκευές (εκτυπωτές, σχεδιαστές, κ.ά.). Οι σύγχρονες εφαρμογές που χρησιμοποιούν υπηρεσίες που προσφέρουν τα δίκτυα, μπορούν να παράγουν πολύ μεγάλο μέγεθος πληροφορίων που μετακινούνται μεταξύ συστημάτων πάνω σε ένα δίκτυο. Οι υπάρχουσες λύσεις δεν είναι πλέον ικανοποιητικές σε ορισμένες περιπτώσεις [NBKP 95], [Kung 92]. Υποψήφιες τεχνολογίες για να χρησιμοποιηθούν σε αυτού του είδους τα δίκτυα είναι η τεχνολογία *ATM (Asynchronous Transfer Mode)* και *wormhole*. Το wormhole είναι ο τρόπος επικοινωνίας που χρησιμοποιείται κυρίως από δίκτυα διασύνδεσης παράλληλων υπολογιστών, ενώ το ATM σχεδιάστηκε για να χρησιμοποιείται από δίκτυα υπολογιστών ιδιαίτερα για συνδέσεις σε μεγάλες αποστάσεις αλλά και για παροχή υπηρεσιών για μεταφορά ήχου (τηλέφωνο) και εικόνας.

Με τον τρόπο που είναι σχεδιασμένη η τεχνολογία ATM δείχνει να μην είναι κατάλληλη για να χρησιμοποιηθεί σε δίκτυα που μεταφέρουν δεδομένα σε μικρές αποστάσεις. Αυτό όμως δεν είναι αληθές. Όταν προσθέσουμε έλεγχο ροής με εισιτήρια έχουμε μία τεχνολογία που είναι κατάλληλη και για τοπικά δίκτυα. Με την εργασία αυτή συγκρίνουμε τις δύο αρχιτεκτονικές δικτύων και μέσω προσομοίωσης υπολογίζουμε την παροχή (throughput) και την καθυστέριση που υφίσταται η κίνηση που διέρχεται μέσω αυτών. Τα αποτελέσματα υποδεικνύουν σαφή υπεροχή του ATM με έλεγχο ροής με εισιτήρια έναντι του wormhole και η υπεροχή αυτή γίνεται πιο εμφανής όταν οι συνθήκες κυκλοφορίας γίνονται πιο δύσκολες, όταν δηλαδή η κίνηση είναι εκρηκτική (bursty) ή όταν υπάρχουν θερμά σημεία (hot spots). Η παροχή που επιτυγχάνεται από τα δικτυα ATM είναι πολύ κοντά στην βέλτιστη (100%) και οι καθυστερήσεις είναι μικρότερες. Αυτές οι επιδόσεις επιτυγχάνονται με λιγότερη μνήμη από όση χρειάζεται το wormhole το οποίο μονο όταν είναι αρκετά πλούσιο σε πόρους μπορεί να φτάσει στις επιδόσεις του ATM.

1.1 Ιστορική αναδρομή

Η εξέλιξη των δικτύων, ξεκινά από τον 19ο αιώνα με τις ανακαλύψεις του τηλεγράφου (1830) και του τηλεφώνου (1876). Αρχικά η χρήση των δικτύων ήταν για να διασυνδεθούν μεταξύ τους συσκευές που έστελναν και παραλάμβαναν ήχο. Τα τηλεγραφικά και τηλεφωνικά δίκτυα, αναπτύχθηκαν πολύ στα τέλη του 19ου και κατά τον 20ο αιώνα. Μέχρι το 1960, δίκτυα χρησιμοποιούνταν για να διασυνδέσουν κυρίως τηλεφωνικές συσκευές. Σε αυτά τα δίκτυα, για να συνδεθούν δύο τηλέφωνα μεταξύ τους, πρέπει

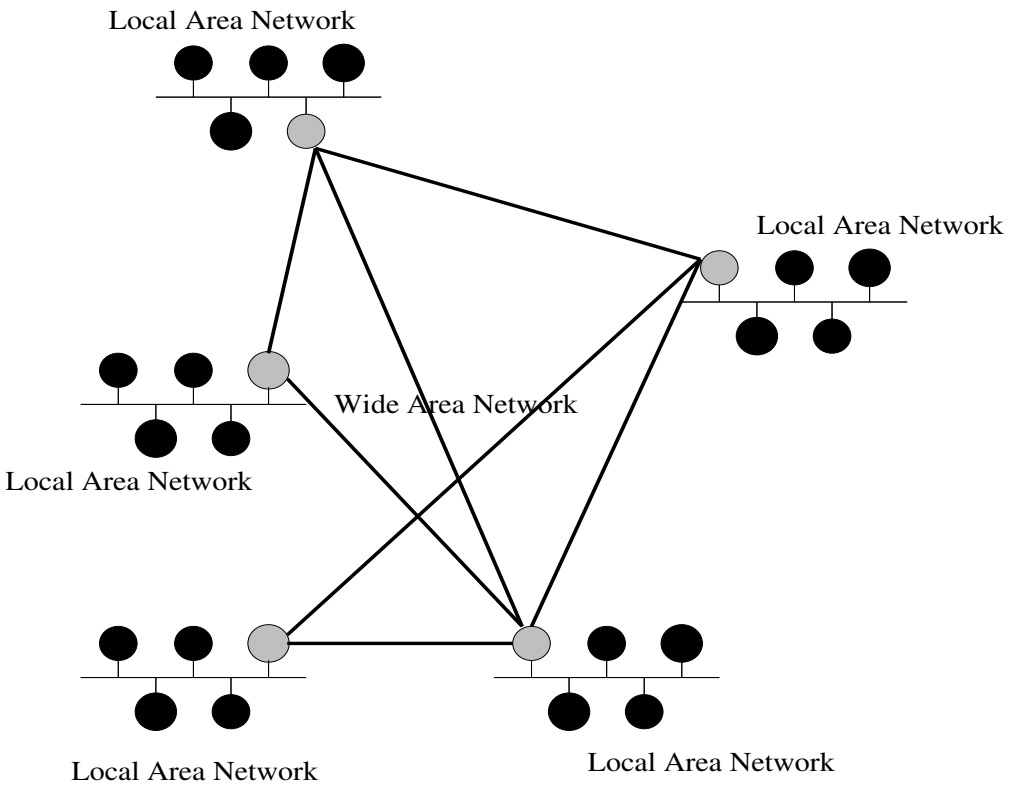
να υπάρχει ένα φυσικό κανάλι που να τα συνδέει. Σε αυτό το κανάλι, εφόσον γίνει δυνατό να διατεθεί, αφιερώνεται σε αυτήν την σύνδεση ένα μέρος του *εύρους ζώνης (bandwidth)* και ακόμα και αν για κάποια χρονικά διαστήματα η σύνδεση δεν χρησιμοποιείται, το εύρος ζώνης δεν μπορεί να χρησιμοποιηθεί από άλλες συνδέσεις. Η μορφή αυτών των δικτύων σε τοπικό επίπεδο, είναι σαν αστέρι, υπάρχει δηλαδή ένας μεταγωγέας με τον οποίο συνδέονται όλα τα τηλέφωνα του τοπικού κέντρου. Οι μεταγωγείς κάθε κέντρου, συνδέονται κατόπιν μεταξύ τους. Ο λόγος που αυτή η τοπολογία έχει επιλεγεί, είναι ότι χρειάζεται να μπορούν να συνδεθούν όλα τα τηλέφωνα μεταξύ τους, αλλά αυτό δεν χρειάζεται να γίνει ταυτόχρονα για όλες τις συσκευές πάνω στο δίκτυο, γιατί αν χρησιμοποιούσαμε ένα καλώδιο για να συνδέσουμε ένα τηλέφωνο με όλα τα άλλα, θα είχαμε μία πολύ μεγάλη σπατάλη πόρων. Ο τρόπος σύνδεσης μεταξυ μίας πηγής και ενός προορισμού με αποκλειστική ανάθεση του φυσικού μέσου στην σύνδεση λέγεται *μεταγωγή κυκλώματος (circuit switching)* [Walr 91]. Μέσα σε ένα δίκτυο η πληροφορία μεταδίδεται είτε σε αναλογική είτε σε ψηφιακή μορφή. Μετατροπή σε αναλογικό σήμα σημαίνει ότι η πληροφορία μετατρέπεται σε ηλεκτρικό κύμα και μεταδίδεται μέσα από τις γραμμές διασύνδεσης. Η μετατροπή σε ψηφιακό σήμα σημαίνει ότι η πληροφορία μετατρέπεται σε μορφή ακολουθίας από bit. Ο δεύτερος αυτός τρόπος μετάδοσης της πληροφορίας είναι περισσότερο ανεκτικός στον θόρυβο και γενικά επιτυγχάνει μεταφορά καλύτερης ποιότητας.

Από το 1970 αρχίζει να αναπτύσσεται ο τομέας των δικτύων υπολογιστών. Η ανάπτυξη κινήθηκε προς δύο κατευθύνσεις. Στον τομέα των τοπικών δικτύων (LANs) και τον τομέα των δικτύων ευρείας περιοχής (WANs). Υπάρχει και μία τρίτη κατηγορία δικτύων που είναι σχεδιασμένα για να καλύπτουν τις ανάγκες μίας περιοχής μεγέθους μερικών χιλιομέτρων (MANs).

Στα δίκτυα ευρείας περιοχής, οι υπολογιστές βρίσκονται γεωγραφικά μακριά μεταξύ τους και κατά κανόνα το φυσικό μέσο που χρησιμοποιείται για να επικοινωνούν μεταξύ τους είναι τηλεφωνικές γραμμές. Αναπτύχθηκαν ειδικά πρωτόκολλα επικοινωνίας μεταξύ υπολογιστών που ήταν συνδεδεμένοι με αυτόν τον τρόπο. Τα πρωτόκολλα διασύνδεσης δεδομένων (data link protocols) περιγράφουν την διαδικασία που πρέπει να ακολουθηθεί για να γίνει δυνατή η επικοινωνία [Walr 91]. Η πληροφορία που είναι προς μετάδοση, χωρίζεται σε κομμάτια που ονομάζονται *πακέτα (packets)*. Για να γίνει πιο αποδοτικός ο τρόπος μετάδοσης, ένα μήνυμα χωρίζεται σε πακέτα μικρού μεγέθους και κάθε πακέτο μεταδίδεται ξεχωριστά. Η τοπολογία αυτών των δικτύων είναι με μορφή πλέγματος (mesh) και οι συνδέσεις είναι από υπολογιστή σε υπολογιστή άμεσα (point to point). Επειδή δεν είναι δυνατό να υπάρχει μία σύνδεση μεταξύ κάθε υπολογιστή που ανήκει στο δίκτυο με κάθε άλλον μέσα σε αυτό, υπάρχουν και υπολογιστές που λειτουργούν ως μεταγωγείς και εφόσον δεν είναι οι αποδέκτες ενός πακέτου που φτάνει σε αυτούς, έχουν πληροφορία να το προωθήσουν προς τον παραλήπτη του. Έτσι προκύπτει η τεχνολογία *μεταγωγής πακέτου (packet switching)*, που χαρακτηριστικό της είναι ότι το φυσικό μέσο δεν χρησιμοποιείται αποκλειστικά από μία σύνδεση αλλά πολλές συνδέσεις πολυπλέκονται πανω στο φυσικό μέσο.

Τα τοπικά δίκτυα αναπτύχθηκαν παράλληλα με τα δίκτυα ευρείας περιοχής και η διαφορά τους από αυτά είναι ότι οι συσκευές που συνδέονται μέσω τοπικών δικτύων βρίκονται γεωγραφικά κοντά. Κύριο χαρακτηριστικό αυτών των δικτύων είναι ότι η επικοινωνία επιτυγχάνεται μέσω ενός κοινόχρηστου μέσου και την πληροφορία που μία συσκευή βάζει μέσα στο δίκτυο, τελικά όλοι μπορούν να την δουν, άρα και ο παραλήπτης. Το κοινόχρηστο μέσο που χρησιμοποιήθηκε αρχικά ήταν ο αέρας, κάνοντας μετάδοση των δεδομένων μέσω ραδιοφωνικού καναλιού (ALOHA) [Walr 91]. Αργότερα όμως, για να γίνει πιο αξιόπιστη η μετάδοση χρησιμοποιήθηκε ομοαξωνικό καλώδιο. Αφού το μέσο είναι κοινόχρηστο, κάθε συσκευή, συνδέεται με κάθε άλλη μέσα στο δίκτυο. Άρα εδώ δεν υπάρχει θέμα για ύπαρξη μεταγωγεών μέσα στο δίκτυο. Το κύριο πρόβλημα σε ορισμένα από αυτά τα δίκτυα είναι οι συγκρούσεις που μπορεί να συμβούν όταν κάποιες συσκευές ξεκινήσουν να εκπέμπουν τα δεδομένα τους ταυτόχρονα και αφού το μέσο είναι διαμοιραζόμενο, τα δεδομένα όλων γίνονται μη αναγνώσιμα

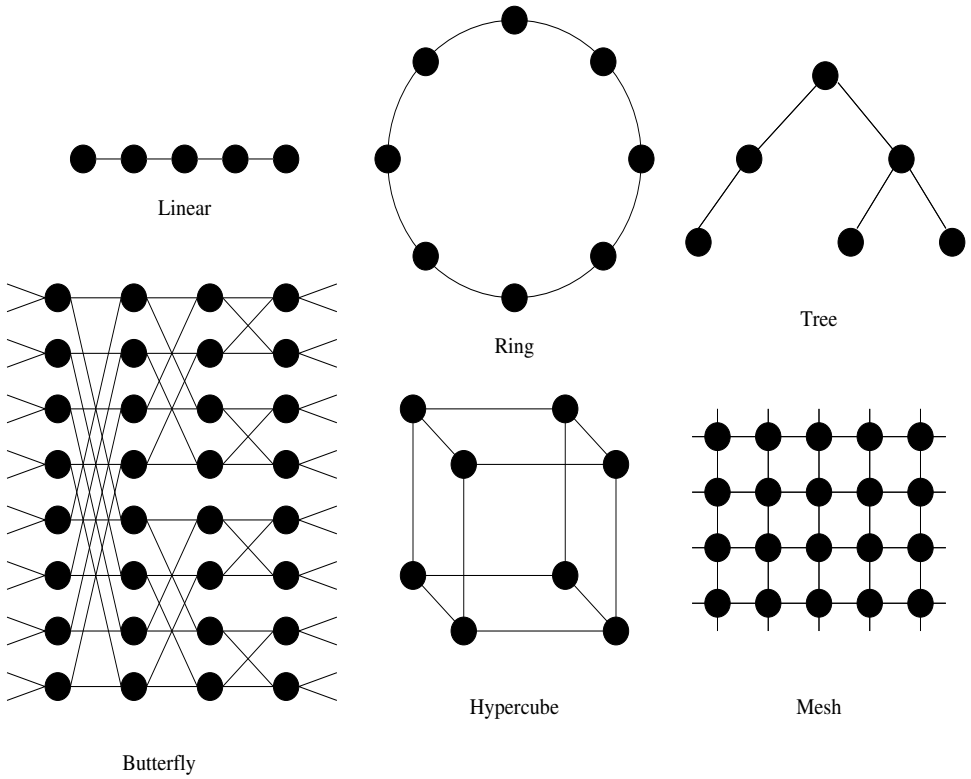
(Ethernet) [Walr 91]. Η μετάδοση τότε πρέπει να επαναληφθεί από όλους όσους συγκρούστηκαν. Αρκετά πρωτόκολλα για να λύνουν τις συγκρούσεις έχουν υλοποιηθεί για να λύσουν το πρόβλημα μετά από μία σύγκρουση. Για να αντιμετωπιστεί αυτό το πρόβλημα και να μην υπάρχουν συγκρούσεις, ορισμένα πρωτόκολλα υποχρεώνουν κάθε συσκευή που θέλει να στείλει δεδομένα μέσω του δικτύου να έχει ένα κουπόνι (token). Μόλις το αποκτήσει μεταδίδει τα δεδομένα που θέλει και μετά περνάει το κουπόνι αυτό σε κάποιον άλλο που πιθανόν περιμένει για να μεταδόσει δεδομένα. Ανάλογα με την δυνατότητα του δικτύου μπορεί να υπάρχουν ένα ή και περισσότερα κουπόνια.



Σχήμα 1.1: Ιεραρχική δομή δικτύου

Συνήθως οι παραπάνω τύποι δικτύου χρησιμοποιούνται σε συνδιασμό, έχοντας ορισμένους υπολογιστές συνδεδεμένους με ένα δίκτυο ευρείας περιοχής και ο κάθε ένας από αυτούς, είναι συνδεδεμένος με ένα τοπικό δίκτυο (σχήμα 1.1). Όταν ένας υπολογιστής από ένα τοπικό δίκτυο θέλει να επικοινωνήσει με έναν άλλο υπολογιστή ενός άλλου τοπικού δικτύου, χρησιμοποιεί υπηρεσίες τις οποίες προσφέρει ο υπολογιστής που βρίσκεται άμεσα συνδεδεμένος με το δίκτυο ευρείας περιοχής. Παράδειγμα τέτοιου δικτύου είναι και το Internet.

Η επιθυμία για μεγαλύτερη υπολογιστική ισχύ, οδήγησε στην ανάπτυξη πολυεπεξεργαστών ή παράλληλων μηχανών που συνδέουν επεξεργαστές σε πολύ μικρή απόσταση. Οι μηχανές αυτές αποτελούνται από μονάδες επεξεργασίας που επικοινωνούν μεταξύ τους μέσω ενός δικτυου διασύνδεσης υψηλών επιδόσεων. Κάθε μονάδα επεξεργασίας μπορεί να αποτελείται από επεξεργαστή με την κρυφή του μνήμη, τοπική κύρια μνήμη καθώς και μονάδες που αναλαμβάνουν να διαχειριστούν τα δεδομένα που διακινούνται στο δίκτυο διασύνδεσης [LLGW 92] [KOHR 94] [ABCJ 92]. Ανάλογα με το μοντέλο της παράλληλης μηχανής μπορεί πάνω στο δίκτυο να βρίσκονται και μονάδες κεντρικής μνήμης που μπορούν να δουν όλοι οι επεξεργαστές. Οι τοπολογίες που συνήθως χρησιμοποιούνται γι αυτά τα δίκτυα είναι (σχήμα 1.2): η γραμμική (linear), ο δακτύλιος (ring), το δένδρο (tree), το πλέγμα (mesh), ο υπερκύβος (hypecube) και τα δίκτυα πολλαπλών επιπέδων (multistage networks) όπως η πεταλούδα (butterfly) ή το



Σχήμα 1.2: Παραδείγματα τοπολογιών δικτύων παράλληλων υπολογιστών

δέλτα (delta) δίκτυο [BeTs 89]. Υπάρχουν δύο βασικές κατηγορίες παράλληλων μηχανών: οι μηχανές κοινόχρηστης μνήμης και οι μηχανές με πέρασμα μηνυμάτων. Στις μηχανές κοινόχρηστης μνήμης, όπου η μνήμη του συστήματος είναι κοινή σε όλους τους επεξεργαστές, υπάρχει ένα πεδίο διευθύνσεων για όλες τις μονάδες επεξεργασίας. Διεργασίες του ιδίου προγράμματος που τρέχουν σε διαφορετικές μονάδες επεξεργασίας επικοινωνούν μεταξύ τους μέσω κοινόχρηστων δομών στην μνήμη. Στις μηχανές με πέρασμα μηνυμάτων δεν υπάρχει κύρια μνήμη που να την βλέπουν όλοι οι επεξεργαστές και το πεδίο διευθύνσεων είναι ξεχωριστό για κάθε μονάδα επεξεργασίας. Διεργασίες του ιδίου προγράμματος που τρέχουν σε διαφορετικές μονάδες επεξεργασίας, επικοινωνούν μεταξύ τους περνώντας μηνύματα. Σε κάθε κόμβο του δικτύου διασύνδεσης υπάρχει και μία μονάδα που παρακολουθεί την κίνηση στο δίκτυο διασύνδεσης και εφόσον κάποια πληροφορία που διακινείται σε αυτό, αφορά τον τοπικό κόμβο, την εξάγει από το δίκτυο και την παραδίδει στον τοπικό επεξεργαστη. Αν δεν απευθύνεται στον τοπικό κόμβο η πληροφορία που έφτασε, πρέπει να προωθηθεί στην σωστή κατεύθυνση για να φτάσει στον προορισμό της.

1.2 Asynchronous Transfer Mode (ATM)

Η μέθοδος ATM είναι αυτή που έχει υιοθετηθεί για πολλά δίκτυα τοπικά και ευρείας περιοχής για μεταφορά δεδομένων σε υψηλές ταχύτητες. Η μέθοδος ATM ορίζει τον τρόπο με τον οποίο μετακινούνται οι πληροφορίες μέσα στο δίκτυο. Έστω ότι μία πηγή θέλει να στείλει ένα μήνυμα σε κάποιο προορισμό. Το μήνυμα χωρίζεται σε μικρότερα κομμάτια που λέγονται *πακέτα (packets)* και ειδοποιείται ο διαχειριστής του δικτύου να κατασκευάσει μία σύνδεση. Ο διαχειριστής κατασκευάζει τότε μία *εικονική σύνδεση (Virtual Connection)* βρίσκοντας ένα μονοπάτι μέσα στο δίκτυο που συνδέει την πηγή με τον προορισμό και ενημερώνει τους μεταγωγείς πάνω σε αυτό το μονοπάτι για την δημιουργία

της νέας σύνδεσης. Στην συνέχεια κάθε πακέτο χωρίζεται σε μικρότερα κομμάτια σταθερού μεγέθους που ονομάζονται cells. Κάθε cell αποτελείται από την επικεφαλίδα και το σώμα του που περιέχει μέρος του πακέτου [RoCG 94]. Τα cells διαφορετικών πακέτων μπορεί να πολυπλέκονται στους συνδέσμους. Τα cells κάθε πακέτου μεταδίδονται με την σειρά και παραδίδονται στον προορισμό τους με την ίδια σειρά με την οποία μεταδόθηκαν. Ένα άλλο πακέτο του μηνύματος μπορεί να χρησιμοποιήσει την ίδια εικονική σύνδεση. Όταν τελειώσει η πηγή με ό,τι θέλει να μεταδόσει, ειδοποιεί τον διαχειριστή του δικτύου να αποσυνθέσει την εικονική σύνδεση. Ο έλεγχος ροής που εφαρμόζεται στο standard ATM γίνεται μέσω ελέγχου του ρυθμού μετάδοσης δεδομένων από τις πηγές (παράγραφος 1.4).

1.3 Wormhole routing

Η μέθοδος wormhole είναι από τις πιο δημοφιλείς σήμερα μεθόδους διασύνδεσης παράλληλων επεξεργαστών. Στην περίπτωση του wormhole και πάλι ένα μήνυμα χωρίζεται σε μικρότερα πακέτα και το κάθε πακέτο χωρίζεται σε μικρά κομμάτια σταθερού μεγέθους που λέγονται *flit*. Τα flits μπορούν να θεωρηθούν αντίστοιχα των ATM cells. Στην περίπτωση του wormhole δεν δημιουργούνται εικονικές συνδέσεις όπως στην περίπτωση του ATM, αλλά κάθε πακέτο δημιουργεί την σύνδεση καθώς προχωράει μέσα στο δίκτυο και την αποσυνθέτει όταν ολοκληρωθεί η διέλευσή του από αυτό. Παρ' όλα αυτά στα περισσότερα δίκτυα, όλα τα πακέτα που πηγαίνουν από μία πηγή στον ίδιο προορισμό, ακολουθούν το ίδιο μονοπάτι. Ένα πακέτο αποτελείται από τριών ειδών flit:

1. Επικεφαλίδα : το flit αυτό περιέχει τον προορισμό του πακέτου. Αναλαμβάνει να ενημερώσει για την καινούργια σύνδεση κάθε μεταγωγέα από τον οποίο περνάει δημιουργώντας τον δρόμο που τα υπόλοιπα flit του πακέτου θα ακολουθήσουν.

2. Σώμα : περιέχει λίγες πληροφορίες ελέγχου και μέρος τών δεδομένων του πακέτου. Ένα flit σώμα δεν περιέχει την πληροφορία του προορισμού του αλλά απλώς ακολουθεί την επικεφαλίδα. Γι αυτόν τον λόγο, όταν ξεκινήσει ένα πακέτο να μεταδίδεται, κρατάει τους πόρους που έχει καταλάβει μέχρι να ολοκληρώσει την μετάδοσή του. Αν μπλοκαριστεί, οι πόροι αυτοί δεν μπορούν να χρησιμοποιηθούν από άλλα πακέτα που θα μπορούσαν ίσως να εξυπηρετηθούν.

3. Ουρά : αναλαμβάνει να αποσυνθέσει την σύνδεση που είχε δημιουργήσει η επικεφαλίδα. Μόνη πληροφορία ελέγχου που περιέχει, όπως και στο flit−σώμα, είναι το πακέτο στο οποίο ανήκει. Μπορεί ακόμα να περιέχει και το τελευταίο μέρος των δεδομένων του πακέτου.

Όταν μόνο ένα πακέτο αρκεί να καταλάβει όλο τον χώρο αποθήκευσης που είναι αφιερωμένος σε έναν σύνδεσμο ενός μεταγωγέα τότε αν αυτό το πακέτο μπλοκαριστεί μπορούν να εμποδιστούν άλλα πακέτα που περνούν από τον ίδιο σύνδεσμο αλλά κατευθύνονται σε άλλους προορισμούς. Για να βελτιωθούν οι επιδόσεις του δικτύου, οργανώνεται η μνήμη του μεταγωγέα με τέτοιον τρόπο ώστε ένα μέρος της μόνο να αφιερώνεται σε ένα πακέτο. Ετσι δημιουργούνται *λωρίδες (lanes)* ώστε να μπορούν περισσότερα από ένα πακέτα να βρίσκονται στην μνήμη ενός μεταγωγέα. Μόνο ένα πακέτο μπορεί να χρησιμοποιήσει μία λωρίδα [DaSe 87] [Dally 90]. Και πάλι πολυπλέκονται πακέτα στους συνδέσμους αλλά σε αυτήν την περίπτωση μπορούν να πολυπλεκτούν το πολύ τόσα πακέτα όσες λωρίδες.

1.4 Συμφόρηση δικτύων – Έλεγχος ροής

Ένα δίκτυο, είτε τηλεφωνικό είναι είτε δίκτυο υπολογιστών, έχει δυνατότητα μεταγωγής δεδομένων ανάλογα με το πόσο πλούσιο σε πόρους είναι. Πόροι ενός δικτύου είναι οι σύνδεσμοι που συνδέουν τις συσκευές στα άκρα του δικτύου, οι μεταγωγείς εφόσον υπάρχουν, και ο χώρος αποθήκευσης που

υπάρχει στο δίκτυο, για να μπορεί αυτό να ανεχθεί μεταβολές στην κίνηση που παρέχεται στο δίκτυο από τις πηγές δεδομένων. Ορίζουμε σαν φόρτο που εισέρχεται στο δίκτυο από έναν σύνδεσμο εισόδου, τον λόγο του ρυθμού με τον οποίο παρέχεται κίνηση στο δίκτυο λ, προς τον μέγιστο ρυθμό με τον οποίο το δίκτυο μπορεί να δεχθεί την κίνηση αυτή μ.

$$l = \frac{\lambda}{\mu}$$

Όταν για παράδειγμα, ο ρυθμός με τον οποίο εισέρχεται η κίνηση στο δίκτυο από έναν σύνδεσμο εισόδου είναι ο μισός από τον ρυθμό με τον οποίο το δίκτυο μπορεί να δεχθεί δεδομένα, ο φόρτος που παρέχεται στο δίκτυο από τον σύνδεσμο αυτόν είναι 0.5. Όταν η κίνηση που εισέρχεται μέσα στο δίκτυο είναι μεγάλη, το δίκτυο αρχίζει να γεμίζει με κίνηση. Αυτό το φαινόμενο συνεχίζεται όσο αυξάνει το φορτίο μέχρι να επέλθει *κορεσμός (saturation)*. Σε αυτήν την περίπτωση, το δίκτυο δεν μπορεί να μεταφέρει περισσότερη κίνηση και αρχίζουν οι ουρές των μεταγωγέων μέσα σε αυτό να γεμίζουν. Ο χώρος αποθήκευσης που είναι διατεθειμένος για κάθε τέτοια ουρά είναι περιορισμένος και όταν δεν υπάρχει έλεγχος ροής εφόσον το μέγεθος κάποιας ουράς ξεφύγει από τα επιτρεπτά όρια, αρχίζουν να **χάνονται πακέτα**. Αυτό, αν και δεν είναι επιθυμητό, σε μερικές περιπτώσεις είναι αποδεκτό. Τέτοιες περιπτώσεις έχουμε όταν μεταφέρουμε φωνή (στο τηλέφωνο) ή εικόνα, όπου το να χαθούν ορισμένα cell είναι προτιμότερο από το να καθυστερήσουν πολύ. Στην περίπτωση όμως που μεταφέρονται δεδομένα, η απώλεια cell δεν είναι αποδεκτή. Το πρόβλημα αυτό μπορεί να αντιμετωπιστεί με δύο τρόπους:

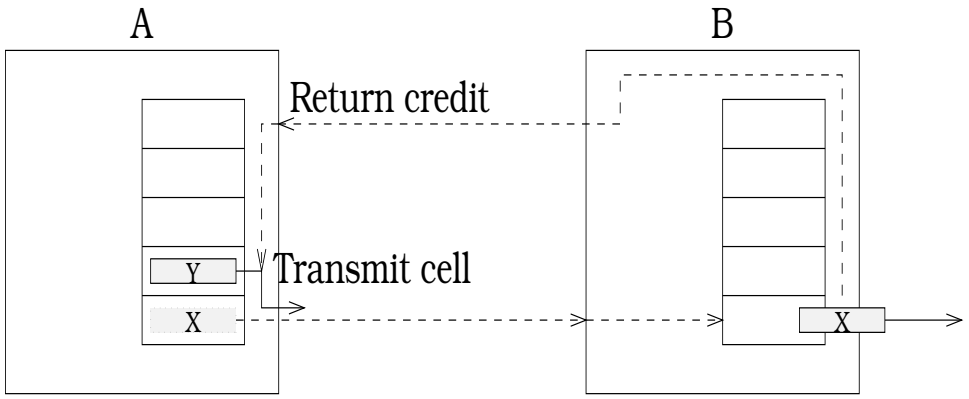
1. σε περίπτωση που επιτρέπεται να χαθούν δεδομένα λόγω υπερχείλισης, η ανάρρωση από την απώλεια γίνεται μέσω επαναμετάδοσης. Στα άκρα της σύνδεσης εφαρμόζεται ένα πρωτόκολλο το οποίο αναλαμβάνει να βρει αν υπάρχουν απώλειες από δεδομένα και αν κάτι τέτοιο παρατηρηθεί, ο παραλήπτης ειδοποιεί τον αποστολέα να ξαναστείλει τα δεδομένα που του λείπουν. Η μέθοδος αυτή είναι απλή να υλοποιηθεί αλλά τα αποτελέσματά της δεν είναι ικανοποιητικά σε ορισμένες περιπτώσεις. Μπορεί για παράδειγμα το δίκτυο να έχει μεγάλη κίνηση, με αποτέλεσμα, τα cell να χάνονται επειδή οι ουρές γεμίζουν. Το λογισμικό που θα ανιχνεύσει αυτήν την απώλεια, θα παράγει επιπρόσθετη κίνηση για να ειδοποιήσει την πηγή για τις απώλειες, και η πηγή θα παράγει νέα κίνηση από την οποία, εφόσον συνεχίζεται η *συμφόρηση (congestion)* κατά πάσα πιθανότητα θα χαθούν και πάλι δεδομένα προς τον ίδιο προορισμό κ.ο.κ. [Kung 92] και το προβλημα να οδηγήσει στην *κατάρρευση (collapse)* του δικτύου.

2. όταν δεν επιτρέπεται υπερχείλιση μπορούμε να έχουμε αποφυγή του λάθους. Εφόσον κατά την διάρκεια της μετάδοσης, εξασφαλιστεί ότι δεν θα χαθεί cell εξ' αιτίας κάποιας υπερχείλισης ουράς, τότε δεν είναι πιθανή η απώλεια cell. Πάντως δεν αποκλείεται εντελώς η πιθανότητα αυτή γιατί υπάρχουν και τα λάθη λόγω θορύβου στις γραμμές μετάδοσης. Άρα ο μηχανισμός ελέγχου μέσω λογισμικού δεν είναι περιττός αλλά δεν χρειάζεται να χρησιμοποιείται συχνά για να αναρρώνει από απώλειες cell και η απόδοση βελτιώνεται.

Οι μέθοδοι που χρησιμοποιούνται για τον έλεγχο της ροής της κίνησης είναι οι ακόλουθες:

- Έλεγχος ρυθμού μετάδοσης (rate based flow control): Η μέθοδος αυτή εφαρμόζεται όταν έχουμε επιτύχει σύνδεση μεταξύ πηγής και προορισμού και εφαρμόζεται στα άκρα της σύνδεσης αυτής. Η γενική ιδέα της μεθόδου είναι ότι όταν σε κάποιο σημείο του δικτύου εμφανιστεί συμφόρηση (congestion), ο μεταγωγέας που την παρατηρεί γράφει την πληροφορία αυτή σε ένα ειδικό πεδίο κάθε cell που περνάει από αυτόν και όταν ο προορισμός, διαπιστώσει ότι κάπου εμφανίστηκε συμφόρηση, ειδοποιεί την πηγή να μειώσει τον ρυθμό με τον οποίο μεταδίδει τα δεδομένα της. Υπάρχουν διάφορες παραλλαγές αυτής της μεθόδου με κυριότερη ότι η πηγή μειώνει τον ρυθμό της κάθε φορά που στέλνει ένα cell και εφόσον δεν παρατηρηθεί συνωστισμός στο δίκτυο, ο

προορισμός ειδοποιεί την πηγή να αυξήσει και πάλι τον ρυθμό με τον οποίο στέλνει τα δεδομένα της [OMSI 95]. Αυτή η μέθοδος δεν εξασφαλίζει ότι δεν θα χαθούν cells εξ'αιτίας της υπεχείλισης κάποιας ουράς αλλά όταν σχεδιαστεί καλά μπορεί να μειώσει την πιθανότητα να συμβεί μία τέτοια περίπτωση, καθώς θα διαπιστωθεί η συμφόρηση εγκαίρως και θα μειωθεί ο ρυθμός με τον οποίο μεταδίδονται τα δεδομένα από τις πηγές. Όταν χαθεί κάποιο cell, χρησιμοποιείται ο κατάλληλος μηχανισμός για να επαναμεταδοθεί εφόσον αυτό είναι απαραίτητο.



Σχήμα 1.3: Συνθήκες για αποδοτική λειτουργία

- Χρησιμοποίηση εισιτηρίων (credit based flow control): Η μέθοδος αυτή εφαρμόζεται ανάμεσα σε μεταγωγείς. Η γενική ιδέα της μεθόδου είναι ότι ένας μεταγωγέας δεν στέλνει στον επόμενό κάποιο cell, εφόσον δεν ξέρει ότι αυτός θα έχει χώρο ελεύθερο για να το δεχθεί. Η πληροφορία για το αν υπάρχει ελεύθερη θέση γίνεται γνωστή μέσω των εισιτηρίων. Ένας μεταγωγέας έχει αρχικά ένα πλήθος από εισιτήρια για κάθε άλλο μεταγωγέα με τον οποίο συνδέεται. Κάθε φορά που μεταδίδει ένα cell, μειώνει και το πλήθος των εισιτηρίων που διαθέτει για τον μεταγωγέα στον οποίο θα πάει αυτό το cell. Εφόσον τελειώσουν τα εισιτήρια προς κάποιο μεταγωγέα, αυτό σημαίνει ότι δεν είναι βέβαιο ότι υπάρχει ελεύθερος χώρος εκεί και άρα σταματάει η αποστολή νέων cells. Ο μεταγωγέας τότε περιμένει μέχρι να λάβει σήμα από τον επόμενο ότι μετέδοσε κάποια cells και υπάρχουν ελεύθερες θέσεις σε αυτόν. Το σήμα (cell) αυτό προκαλεί την παροχή εισιτηρίων και μόλις ο πρώτος μεταγωγέας το λάβει, αυξάνει τον μετρητή των εισιτηρίων προς τον μεταγωγέα από τον οποίο έλαβε την άδεια. Αν έχει cells που πρέπει να δρομολογηθούν εκεί, συνεχίζει την μετάδοσή τους. Η μέθοδος αυτή εξασφαλίζει ότι δεν θα χαθούν cells επειδή κάποια ουρά υπερχειλίζει, αλλά και εδώ χρειάζεται η ύπαρξη πρωτοκόλλων γιατί θα πρέπει να μπορούν να ανιχνευτούν cells που χάνονται εξ'αιτίας του θορύβου στις γραμμές μετάδοσης, αν και το φαινόμενο είναι σπάνιο λόγω της αξιοπιστίας των γραμμών. Συμφόρηση με την έννοια που περιγράψαμε προηγουμένως, δεν μπορεί να παρατηρηθεί, αφού η κίνηση που θα μπορούσε να την προκαλέσει δεν εισέρχεται στο δίκτυο. Για να μην χάνεται εύρος ζώνης (bandwidth) θα πρέπει να ικανοποιείται η ακόλουθη σχέση [Kate 94]:

$$\text{πλήθος εισιτηρίων} = \frac{\text{χρόνος επαναφοράς}}{\text{μέγιστη παροχή}}$$

Ο *χρόνος επαναφοράς (round−trip delay)* είναι ο ελάχιστος απαιτούμενος χρόνος που μεσολαβεί μεταξύ της αναχώρησης ενός cell X από έναν μεταγωγέα A προς έναν μεταγωγέα B, μέχρι την στιγμή που η μετάδοση ενός νέου πακέτου μπορεί να ξεκινήσει από τον A εξ' αιτίας της άφιξης του εισιτηρίου που δημιουργήθηκε από την αναχώρηση του X από τον B (σχήμα 1.3). Η μέγιστη παροχή είναι ο μέγιστος ρυθμός με τον οποίο ο μεταγωγέας A μπορεί να στείλει cells στον μεταγωγέα B (σχήμα 1.3).

1.5 Το πρόβλημα που μελετήθηκε

Η εργασία αυτή μελετάει την συμπεριφορά των δικτύων που έχουν αρχιτεκτονική ATM με έλεγχο ροής με εισιτήρια και την συγκρίνει με την συμπεριφορά των δικτύων που χρησιμοποιούν wormhole routing. Καθώς η τεχνολογία μας οδηγεί στην σύνδεση σταθμών εργασίας μέσω δικτύων υψηλών επιδόσεων [NBKP 95] για να επιτύχουμε επιδόσεις που πλησιάζουν τις επιδόσεις παράλληλων υπολογιστών, οι δύο προαναφερθήσες τεχνολογίες είναι υποψήφιες να χρησιμοποιηθούν για την κατασκευή τους.

Η τεχνολογία ATM, είτε χωρίς έλεγχο ροής είτε με έλεγχο του ρυθμού μετάδοσης δεδομένων από τις πηγές, εφαρμόζεται σήμερα, με επιτυχία, σε δίκτυα ευρείας περιοχής. Αν προστεθεί έλεγχος ροής με εισιτήρια, παύει να υπάρχει η μη επιθυμητή ιδιότητα του να χάνονται cells και η τεχνολογία ATM γίνεται ελκυστική για να χρησιμοποιηθεί και στον χώρο των τοπικών δικτύων. Προσθέσαμε ακόμα στο ATM την έννοια της λωρίδας για να μην επιτρέπουμε μία ροή δεδομένων προς ένα προορισμό που δέχεται πολύ κίνηση, να γεμίσει άσκοπα την μνήμη ενός μεταγωγέα και να μην αφήσει άλλες ροές να εξυπηρετηθούν.

Η τεχνολογία wormhole χρησιμοποιείται σήμερα σε πολλούς παράλληλους υπολογιστές και θεωρείται κατάλληλη για την δημιουργία δικτύων υψηλών επιδόσεων. Η απόδοση δικτύων που χρησιμοποιούν wormhole routing μελετήθηκε στην εργασία [Dally 90]. Σε εκείνη την εργασία μελετήθηκε ένα μόνο μέγεθος πακέτου και τα πακέτα που παρήγαγε μία πηγή κατανέμονταν ομοιόμορφα στους προορισμούς. Επεκτείναμε την μελέτη για την περίπτωση του wormhole και στις περιπτώσεις που το μέγεθος πακέτου μεταβάλλεται καθώς και όταν υπάρχουν προορισμοί που δέχονται περισσότερη κίνηση από άλλους. Για να συγκρίνουμε τα δύο είδη αρχιτεκτονικών, επαναλάβαμε τα πειράματα αυτά και στην περίπτωση του ATM με έλεγχο ροής με εισιτήρια.

Οι προσομοιώσεις μάς δείχνουν ότι το ATM δίκτυο έχει πολύ καλή παροχή (throughput) με μικρότερο μέγεθος μνήμης. Τα δίκτυα μάλιστα με περισσότερες από μία λωρίδες δεν επηρεάζονται από μεγαλύτερα μεγέθη πακέτων. Όταν υπάρχουν θερμά σημεία, η κίνηση που δεν κατευθύνεται προς αυτά, δεν επηρεάζεται από την κίνηση που κατευθύνεται στα θερμά σημεία, αρκεί να υπάρχουν λίγες περισσότερες λωρίδες από το πλήθος των θερμών σημείων.

Το wormhole έχει χαμηλότερες επιδόσεις σχεδόν σε όλες τις περιπτώσεις. Μόνο σε χαμηλά φορτία και ομοιόμορφη Poisson κίνηση εμφανίζει παρόμοιες επιδόσεις όταν έχει αρκετές λωρίδες. Ιδιαίτερα υπό συνθήκες εκρηκτικής κίνησης και κίνησης με θερμά σημεία, όταν το πλήθος των λωρίδων είναι μικρό, οι επιδόσεις του δεν είναι καλές.

Η υπόλοιπη εργασία έχει την ακόλουθη δομή. Στο κεφάλαιο 2 περιγράφονται οι βασικές αρχές ενός δικτύου ATM καθώς και ο τρόπος που υλοποιήσαμε τον έλεγχο ροής για την εργασία. Στο κεφάλαιο 3 περιγράφονται τα μοντέλα που υλοποιήθηκαν στον προσομοιωτή καθώς και η βιβλιοθήκη υποστήριξης που χρησιμοποιήθηκε. Στο κεφάλαιο 4 αναφέρονται τα πειραματικά αποτελέσματα, και στο κεφάλαιο 5 τα συμπεράσματα από την εργασία αυτή.

Κεφάλαιο 2

Δίκτυα ATM

Τα δίκτυα ATM αποτελούνται από *μεταγωγείς (switches)*. Μεταξύ μίας πηγής κίνησης και ενός προορισμού παρεμβάλλονται ένας ή περισσότεροι μεταγωγείς. Ένας μεταγωγέας μπορεί για απλότητα να θεωρηθεί ότι είναι ένα "κουτί" με Ν εισόδους και Μ εξόδους. Σκοπός του είναι να προωθήσει στις κατάλληλες εξόδους του τα πακέτα που φτάνουν στις εισόδους του. Όταν ένα πακέτο φτάσει σε μία είσοδό του, ο μεταγωγέας πρέπει να έχει πληροφορία για να βρει ποια έξοδο θα χρησιμοποιήσει αυτό το πακέτο. Κατά την ορολογία, ο μεταγωγέας *δρομολογεί (routes)* το πακέτο στην κατάλληλη έξοδο. Επειδή όμως είναι πιθανό από διαφορετικές εισόδους να εισέρχονται ταυτόχρονα πακέτα που πηγαίνουν στην ίδια έξοδο, ένας μεταγωγέας πρέπει να έχει και την δυνατότητα της *αποθήκευσης (buffering)* μερικών από τα πακέτα που συγκρούονται (confict) στις εξόδους [Toba 90].

Οι μέθοδοι με τις οποίες γίνεται η δρομολόγηση διαφέρουν ως προς τον χρόνο που μπορεί να ξεκινήσει ένα πακέτο να μεταδίδεται από τις εξόδους του μεταγωγέα. Παλιότερα θα έπρεπε να φτάσει ολόκληρο το πακέτο σε ένα μεταγωγέα για να αρχίσει η μετάδοσή του από τον κατάλληλο σύνδεσμο εξόδου. Η μέθοδος αυτή είναι γνωστή ως *αποθήκευση και προώθηση (store–and–forward)*. Με την βελτίωση της τεχνολογίας και κάτω από διαφορετικές συνθήκες έγινε δυνατή η έναρξη της μετάδοσης ενός πακέτου στον επόμενο μεταγωγέα να γίνει ακόμα και όταν ένα μικρό κομμάτι του πακέτου (συνήθως η επικεφαλίδα του) φτάσει στον παρόντα μεταγωγέα. παραλλαγές αυτού του τρόπου είναι το *cut through* και το *wormhole*. Η βασικότερη διαφορά μεταξύ τους είναι ότι στο cut through αφιερώνεται χώρος για ολόκληρο το πακέτο σε έναν μεταγωγέα και έτσι αν αυτό μπλοκαριστεί καταλαμβάνει μόνο αυτόν τον χώρο, ενώ στην περίπτωση του wormhole δεν αφιερώνεται χώρος σε έναν μεταγωγέα για ένα πακέτο [Dally 90] και όταν αυτό μπλοκαριστεί, τμήματά του βρίσκονται σε διαφορετικούς μεταγωγείς.

Εφόσον πολλά πακέτα συγκρούονται στις εξόδους του μεταγωγέα, είναι δυνατό να υπερχειλίσει ο χώρος αποθήκευσης και να υπάρχει απώλεια πακέτων (παράγραφος 1.4). Για να μην παρουσιάστούν αυτού του είδους τα προβλήματα εφαρμόζεται έλεγχος ροής, ώστε μην χάνονται πακέτα λόγω υπερχείλισης του χώρου αποθήκευσης.

2.1 Μέθοδοι ελέγχου ροής με εισιτήρια

Για την διαχείριση της μνήμης κατά την υλοποίηση μίας μεθόδου ελέγχου ροής με εισιτήρια υπάρχουν διάφορες μεθοδολογίες:

- διαμοιραζόμενος χώρος αποθήκευσης χωρίς έλεγχο. Σ' αυτή την οργάνωση μνήμης ο διαθέσιμος χώρος αποθήκευσης ενός μεταγωγέα είναι κοινόχρηστος για όλες τις ροές δεδομένων που διέρχονται μέσα από αυτόν. Εφόσον όλες οι συνδέσεις που χρησιμοποιούν ένα μεταγωγέα συναγωνίζονται για τον χώρο αυτόν, υπάρχει περίπτωση κάποια σύνδεση που έχει μεγάλη κίνηση

να μονοπωλεί την χρήση του χώρου σε βάρος συνδέσεων που δεν έχουν συνεχή ροή. Έτσι δεν υπάρχει έλεγχος για το ποια ροή καταλαμβάνει πόσο χώρο, μέσα στον μεταγωγέα. Αυτό μπορεί να δημιουργεί ορισμένες αδικίες ως προς την χρήση του χώρου, αλλά αυτή η μέθοδος επιτυγχάνει καλή χρησιμοποίηση του χώρου αποθήκευσης, γιατί κάποιο cell που θέλει να χρησιμοποιήσει τον μεταγωγέα που διαθέτει ελεύθερο χώρο, μπορεί να περάσει. Ένα άλλο μειονέκτημα της μεθόδου είναι ότι η κίνηση μιας ροής μπορεί να επηρεάζει την κίνηση των άλλων, όπως για παράδειγμα όταν υπάρχει ένα θερμό σημείο, μπορεί να δημιουργούνται σημεία συμφόρησης ενώ το υπόλοιπο δίκτυο να υποχρησιμοποιείται.

- κατανεμημένος χώρος αποθήκευσης χωρίς έλεγχο. Σ' αυτή την οργάνωση μνήμης ο διαθέσιμος χώρος αποθήκευσης διαμοιράζεται μεταξύ όλων των συνδέσεων στατικά. Εδώ δεν υπάρχει το μειονέκτημα να μπορεί κάποια σύνδεση να αδικεί τις άλλες, αφού όλες καταλαμβάνουν την ίδια ποσότητα χώρου στην μνήμη. Κατανέμοντας τον χώρο αποθήκευσης μεταξύ των συνδέσεων δεν επιτυγχάνεται καλή χρησιμοποίηση του χώρου. Ορισμένες συνδέσεις μπορεί να μην χρησιμοποιούν συνέχεια τον χώρο που τους είναι ανατεθημένος και έτσι αυτός να μένει αχρησιμοποίητος, ενώ ταυτόχρονα δεν μπορούν να δημιουργηθούν νέες συνδέσεις που θα μπορούσαν να τις χρησιμοποιήσουν. Η περίπτωση αυτή μοιάζει με την περίπτωση της διασύνδεσης με μεταγωγή κυκλώματος (circuit switching). Σ' αυτή την περίπτωση όμως, ο πόρος του δικτύου που υποχρησιμοποιείται δεν είναι το φυσικό μέσο (σύρμα), αλλά ο χώρος αποθήκευσης που ανατίθεται στην σύνδεση.

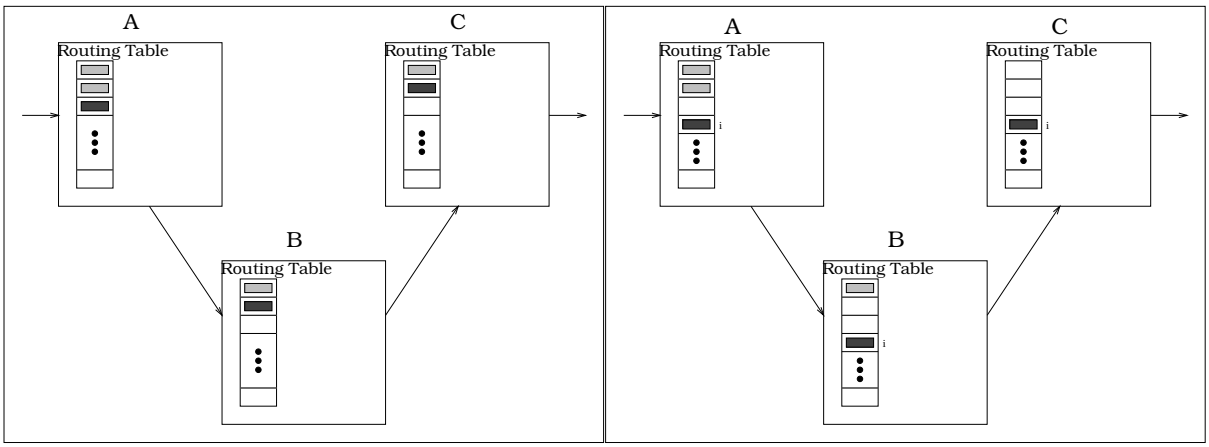
- διαμοιραζόμενος χώρος αποθήκευσης με έλεγχο. Σ' αυτή την οργάνωση μνήμης ο διαθέσιμος χώρος αποθήκευσης είναι κοινόχρηστος σε όλες τις συνδέσεις που διέρχονται από αυτόν. Αυτό σημαίνει ότι τώρα υπάρχει έλεγχος για το ποια σύνδεση χρησιμοποιεί πόσον χώρο. Μπορούν λοιπόν να περάσουν πολλές συνδέσεις από τον μεταγωγέα, αλλά δεν μπορεί μία σύνδεση να μονοπωλήσει όλον τον διαθέσιμο χώρο. Με αυτόν τον τρόπο ξεπερνούνται τα προβλήματα που είχαμε στις προηγούμενες μεθόδους γιατί ούτε μία σύνδεση μπορεί να επηρεάσει όλες τις άλλες, ούτε έχουμε υποχρησιμοποίηση του αποθηκευτικού χώρου. Για να υλοποιηθεί αυτή η μέθοδος χρειάζεται η λογική του ελέγχου κάθε ροής καθώς και ότι ο μεταγωγέας θα πρέπει να μπορεί να χειριστεί κάθε ροή ξεχωριστά. Για τους σκοπούς της εργασίας, μία ροή δεδομένων καθορίζεται από τον προορισμό της. Θεωρούμε ότι, εκτός του κανονικού εισιτηρίου που απαιτεί η μέθοδος ελέγχου ροής με εισιτήρια, υπάρχει και άλλο ένα εισιτήριο που πρέπει να είναι διαθέσιμο για να προχωρήσει ένα cell και αυτό είναι το εισιτήριο προορισμού. Αυτό προστίθεται για να μην επιτρέπεται να προχωρήσουν περισσότερα cells που κατευθύνονται στον ίδιο προορισμό από όσα υποδεικνύει το μέγιστο πλήθος εισιτηρίων προορισμού, χωρίς προηγουμένως να επιστρέψει πίσω εισιτήριο που να υποδεικνύει ότι άδειασε εν τω μεταξύ μία θέση στον επόμενο μεταγωγέα γι αυτή την ροή. Με αυτόν τον τρόπο δεν μπορεί μια ροή να μονοπωλεί τον χώρο αποθήκευσης του μεταγωγέα.

2.2 Εικονικές συνδέσεις και ομάδες ροής

Σε ένα δίκτυο ATM η σύνδεση μεταξύ μίας πηγής και ενός προορισμού γίνεται μέσω *εικονικών συνδέσεων (virtual connections)*. Μία εικονική σύνδεση καθορίζει ένα μονοπάτι μέσα σε ένα δίκτυο το οποίο θα πρέπει να ακολουθήσει οποιοδήποτε cell θέλει να μεταδοθεί από την πηγή στον προορισμό που συνδέει αυτή η εικονική σύνδεση [Walr 91]. Οι εικονικές συνδέσεις ήρθαν να καλύψουν μία αδυναμία που είχαν τα τηλεφωνικά δίκτυα: το να καταλαμβάνει μία σύνδεση ένα φυσικό κανάλι(σύρμα) και να έχει αποκλειστική χρήση αυτού ασχέτως του αν το χρησιμοποιεί συνέχεια ή όχι. Όταν μία σύνδεση

επιτυγχάνεται μέσω μίας εικονικής σύνδεσης, το φυσικό κανάλι δεν ανατίθεται αποκλειστικά σ' αυτήν την σύνδεση, αλλά τμήματά του μπορούν να ανατεθούν και σε άλλες εικονικές συνδέσεις. Με αυτόν τον τρόπο αποσυνδέεται η σύνδεση από το φυσικό μέσο και επιτυγχάνεται αποτελεσματικότερη χρήση του δικτύου. Για να δημιουργηθεί μία εικονική σύνδεση, κάθε μεταγωγέας χρειάζεται να διαθέτει έναν πίνακα που να κρατάει πληροφορίες για την σύνδεση. Ο πίνακας αυτός είναι μέρος του *πίνακα δρομολόγησης (routing table)*. Έχοντας δεδομένους τον σύνδεσμο εισόδου και τον αριθμό της εικονικής σύνδεσης στον παρόντα μεταγωγέα, ο πίνακας παρέχει τον σύνδεσμο εξόδου και τον αριθμό της εικονικής σύνδεσης στον επόμενο μεταγωγέα. Η πληροφορία για τον αριθμό της εικονικής σύνδεσης βρίσκεται μέσα στην επικεφαλίδα του cell, στην οποία γράφεται ο αριθμός της εικονικής σύνδεσης που το cell θα χρησιμοποιήσει στον επόμενο μεταγωγέα. Η εικονική σύνδεση φτιάχνεται από τον χειριστή του δικτύου και μπορεί είτε να είναι μόνιμη είτε να διαρκεί μόνο όσο διαρκεί η μεταφορά και μετά να αποσυντίθεται.

Για την εργασία αυτή, αντί της έννοιας της εικονικής σύνδεσης, χρησιμοποιήθηκε η έννοια της *ομάδας ροής (flow group)*. Συγκεκριμένα, κάθε πακέτο ανήκει σε μία ομάδα ροής, η οποία καθορίζεται αποκλειστικά από τον καταναλωτή για τον οποίο προορίζεται. Συνεπώς η ομάδα ροής που χρησιμοποιείται από ένα πακέτο συμπίπτει με την ομάδα ροής που θα χρησιμοποιούσε οποιοδήποτε πακέτο από όπου και να ξεκινούσε, εφόσον είχε τον ίδιο προορισμό. Σημειωτέον ότι, αν και παρόμοια με την έννοια της εικονικής σύνδεσης, η έννοια της ομάδας ροής διαφέρει στο ότι μπορεί στην ίδια ομάδα ροής να ανήκουν πακέτα προερχόμενα από διαφορετικές πηγές. Όπως και στην περίπτωση των εικονικών συνδέσεων, το φυσικό μέσο που συνδέει κάποιον παραγωγό με κάποιον καταναλωτή δεν χρησιμοποιείται αποκλειστικά από μία ομάδα ροής, αλλά περισσότερες της μίας, οι οποίες πολυπλέκονται στους συνδέσμους. Όμως αντίθετα από την περίπτωση των εικονικών συνδέσεων, πακέτα που προέρχονται από διαφορετικές πηγές αλλά πηγαίνουν στον ίδιο προορισμό, χρησιμοποιούν τους ίδιους πόρους του δικτύου. Και πάλι χρειάζεται σε κάθε μεταγωγέα ένας πίνακας δρομολόγησης που να διατηρεί πληροφορίες για τις συνδέσεις και ανάλογα με τον προορισμό του πακέτου να καθορίζεται ο σύνδεσμος εξόδου που πρέπει να ακολουθηθεί. Ο προορισμός του πακέτου περιέχεται στην επικεφαλίδα του. Μία ομάδα ροής αρχικοποιείται στην φάση ενεργοποίησης του δικτύου και παραμένει ενεργή συνεχώς.



(α) Ανάθεση πόρων σε εικονικές συνδέσεις. (β) Ανάθεση πόρων σε ομάδες ροής.

Σχήμα 2.1: Εικονικές συνδέσεις και ομάδες ροής.

Στο σχήμα 2.1 (α) φαίνεται ο τρόπος με τον οποίο δημιουργείται μία εικονική σύνδεση όπου σε κάθε μεταγωγέα, κάθε νέα εικονική σύνδεση καταλαμβάνει μία ελεύθερη θέση στον πίνακα δρομολόγησης και άρα διαφορετικούς πόρους του δικτύου.

Στο σχήμα 2.1 (β) φαίνεται ότι μία ομάδα ροής καταλαμβάνει πάντα την ίδια θέση στον πίνακα δρομολόγησης (την θέση που καθορίζεται από τον προορισμό της ομάδας ροής), και άρα δεν χρειάζεται να φτιαχθεί καινούρια σύνδεση. Με αυτόν τον τρόπο επιτυγχάνεται η συγχώνευση των ομάδων ροής και κάθε πακέτο που χρησιμοποιεί την ίδια ομάδα ροής, χρησιμοποιεί και τους ίδιους πόρους του δικτύου.

2.3 Μέθοδος ελέγχου ροής του δικτύου ATM

Η μέθοδος ελέγχου ροής που χρησιμοποιήθηκε είναι η ακόλουθη: για να προχωρήσει ένα cell από την πηγή ή από έναν μεταγωγέα σε κάποιον άλλο μεταγωγέα, θα πρέπει να έχει διαθέσιμα δύο είδη εισιτηρίων για να καταναλώσει.

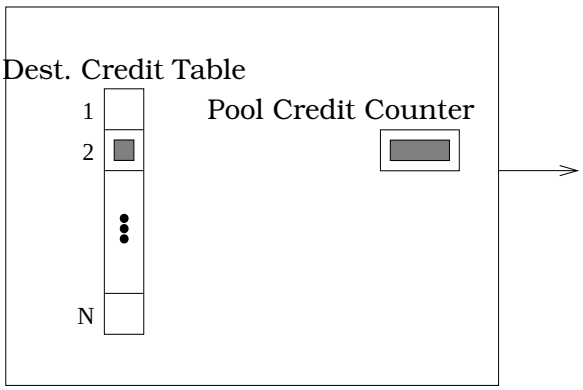
- Εισιτήριο αναχώρησης. Αυτό είναι το εισιτήριο της πρότυπης μέθοδου ελέγχου ροής με εισιτήρια και υποδεικνύει ότι στον επόμενο μεταγωγέα υπάρχει χώρος για να αποθηκεύσει το cell. Για κάθε σύνδεσμο εξόδου από τον μεταγωγέα υπάρχει ένας μετρητής (Pool Credit Counter) που κρατά το πόσες θέσεις, το ελάχιστο, διαθέτει ο επόμενος μεταγωγέας για να δεχθεί cell. Ένα cell δεν μπορεί να φύγει από έναν μεταγωγέα, εφόσον ο μετρητής αυτός είναι μηδενικός. Όταν αρχίσει ένα cell να μεταδίδεται, η τιμή του μετρητή που αντιστοιχεί στον σύνδεσμο εξόδου από τον οποίο μεταδίδεται το cell μειώνεται κατά ένα. Όταν ένα εισιτήριο φτάσει στον μεταγωγέα, η τιμή του αντίστοιχου μετρητή αυξάνεται κατά ένα. Αν ο μετρητής αυτός αρχικοποιηθεί στην τιμή των θέσεων μνήμης που αναθέτει ο επόμενος μεταγωγέας στον παρόντα, τότε εφόσον ο μετρητής δεν είναι μηδενικός, υπάρχει ελεύθερη θέση στον επόμενο μεταγωγέα.

- Εισιτήριο προορισμού. Το εισιτήριο αυτό διατηρείται σε έναν πίνακα (Dest Credit Table) με μέγεθος όσο και οι προορισμοί του δικτύου. Ενα cell δεν μπορεί να φύγει από έναν μεταγωγέα εφόσον ο μετρητής αυτός είναι μηδενικός. Όταν αρχίσει ένα cell να μεταδίδεται, η τιμή του μετρητή που αντιστοιχεί στον προορισμό του μειώνεται κατά ένα. Το μέγιστο πλήθος των εισιτηρίων προορισμού είναι πολύ σημαντική παράμετρος της προσομοίωσης γιατί αυτό κατά κύριο λόγο δείχνει το μέγιστο πλήθος cells με τον ίδιο προορισμό που μπορεί ένας μεταγωγέας να στείλει στον επόμενό του χωρίς να γυρίσει πίσω εισιτήριο και κατ' επέκταση, πόσο χώρο στην μνήμη καταλαμβάνει μία ομάδα ροής, όταν αυτή μπλοκαριστεί. Αυτή είναι και η έννοια της λωρίδας. Διαιρώντας το πλήθος των θέσεων μνήμης που ένας μεταγωγέας διαθέτει για κάθε σύνδεσμο εισόδου του με την μέγιστη τιμή που μπορεί να πάρουν τα εισιτήρια προορισμού, έχουμε το ισοδύναμο του πλήθους των λωρίδων του wormhole για το ATM.

Είναι κατανοητό ότι εφόσον το μέγιστο πλήθος των εισιτηρίων προορισμού είναι μεγαλύτερο ή ίσο του μέγιστου πλήθους των εισιτηρίων αναχώρησης, τότε απενεργοποιούμε τον έλεγχο ροής με εισιτήρια προορισμού και αφήνουμε μόνο τον έλεγχο ροής με εισιτήρια αναχώρησης. Το πλήθος των λωρίδων σε αυτήν την περίπτωση είναι 1 αφού μία ροή, εφόσον μπλοκαριστεί, μπορεί να καταλάβει όλον τον διαθέσιμο χώρο που ο μεταγωγέας διαθέτει στον σύνδεσμο από τον οποίο εισέρχεται η ροή αυτή.

2.4 Συγχώνευση ομάδων ροής και οικονομία πόρων

Όπως σημειώθηκε προηγουμένως, σε αυτήν την εργασία χρησιμοποιείται η έννοια της ομάδας ροής (flow group). Ο πόρος του δικτύου που μπορεί να ανατεθεί σε μία ομάδα ροής είναι ο χώρος αποθήκευσης. Όταν μία μονάδα χώρου αποθήκευσης ανατεθεί σε μία ομάδα ροής, τότε μόνο αυτή μπορεί να τον χρησιμοποιήσει. Τα φυσικά κανάλια, που αποτελούν και αυτά πόρο του δικτύου, δεν ανατίθενται σε ομάδες ροής. Με τον όρο *ανάθεση ενός πόρου σε μια σύνδεση*, σε μία εικονική σύνδεση



Σχήμα 2.2: Έλεγχος ροής με δύο εισιτήρια

ή σε μία ομάδα ροής, εννοούμε ότι σε περίπτωση που οι συνθήκες κυκλοφορίας δεν είναι καλές και αναγκαστεί η σύνδεση ή η ροή να μπλοκαριστεί, τότε δεν υποχρεούται να αφήσει τους πόρους που κατέχει, για να εξυπηρετηθεί άλλη ροή. Η ομάδα ροής μέσα στην οποία κινείται ένα πακέτο, καθορίζεται από τον προορισμό του. Δύο πακέτα που ξεκινάνε από διαφορετικές πηγές του δικτύου και κατευθύνονται στον ίδιο προορισμό, εφόσον κάποια στιγμή συναντηθούν σε έναν μεταγωγέα, θα συνεχίσουν από εκεί και πέρα στην ίδια ομάδα ροής και άρα θα καταναλώνουν τους ίδιους πόρους του δικτύου. Αυτό επιτυγχάνεται με τον ακόλουθο τρόπο:
για κάθε cell ελέγχεται ο προορισμός του. Κάθε φορά που ένα cell φύγει από έναν μεταγωγέα καταναλώνει και ένα εισιτήριο προορισμού. Άρα, ανεξάρτητα από την πηγή του cell, τα εισιτήρια που καταναλώνει έχουν σχέση με τον προορισμό του. Κάθε πακέτο από αυτά που κατευθύνονται στον ίδιο προορισμό, συναγωνίζονται για τα ίδια εισιτήρια και άρα ανήκουν στην ίδια ομάδα ροής.

Αν δεν συνέβαινε συγχώνευση ομάδων ροής, οι ομάδες με τον ίδιο προορισμό θα καταλάμβαναν διαφορετικές λωρίδες κυκλοφορίας, έτσι όπως ορίσαμε τις λωρίδες στην παράγραφο 2.3. Εφόσον λοιπόν η κίνηση προς τον προορισμό και των δύο αυξηθεί, θα μπορούσαν να μπλοκαριστούν και οι δύο και άρα να κατακρατήσουν πόρους του δικτύου που θα μπορούσαν να διατεθούν αλλού. Εφόσον το μέγιστο πλήθος εισιτηρίων προορισμού που δίνονται σε κάθε ομάδα ροής εξασφαλίσει ότι ο έλεγχος ροής δεν επιβραδύνει την κυκλοφορία κάτω από ελαφρύ φορτίο στο δίκτυο, τότε η χρησιμοποίηση διαφορετικών ομάδων ροής από πακέτα που πηγαίνουν στον ίδιο προορισμο δεν ωφελεί σε τίποτα. Η συνθήκη για να λειτουργεί σωστά ο έλεγχος ροής, χωρίς να μειώνεται η απόδοση του δικτύου, περιγράφεται στην παράγραφο 1.4. Υποθέτουμε, για παράδειγμα, ότι το μέγιστο πλήθος από εισιτήρια προορισμού είναι 1 και ότι η κίνηση στο δίκτυο είναι ελαφριά. Για να μην επιβραδύνει ο έλεγχος ροής την κίνηση των cell, θα πρέπει μέχρι να τελειώσει η μετάδοση ενός cell να έχει προλάβει να επιστρέψει το εισιτήριο από τον επόμενο μεταγωγέα γι αυτό το cell. Το cell αυτό έκανε στον επόμενο μεταγωγέα cut through [1], αφού το φορτίο του δικτύου είναι αρκετά ελαφρύ ώστε να του το επιτρέπει, και άρχισε να μεταδίδεται από τον επόμενο μεταγωγέα πριν να τελειώσει η μετάδοσή του από τον παρόντα.

Μπορούμε να παρατηρήσουμε ότι σε έναν μεταγωγέα μπορούν πακέτα με τον ίδιο προορισμό, να καταλαμβάνουν τόσες διαφορετικές λωρίδες όσες και το πλήθος των συνδέσμων εισόδου του μεταγωγέα. Για να βρίσκονται σε διαφορετική ομάδα ροής στον ίδιο μεταγωγέα, δύο πακέτα με τον ίδιο προορισμό, θα πρέπει να προήλθαν από διαφορετικούς συνδέσμους εισόδου. Στον επόμενο μεταγωγέα όμως θα χρησιμοποιήσουν την ίδια ομάδα ροής (λωρίδα). Για να επιτευχθεί όμως cut through σε επίπεδο cell, αυτό θα πρέπει να έχει με την σειρά του δομή, να αποτελείται από επικεφαλίδα και σώμα και άρα να

[1] Οταν ένα cell κάνει cut−through σε έναν μεταγωγέα, αυτό σημαίνει ότι όταν φτάνει το cell αυτό στον μεταγωγέα, ο σύνδεσμος εξόδου που θέλει να χρησιμοποιήσει είτε είναι άεργος ή θα είναι άεργος μετά από μερικούς κύκλους. Τότε το cell μπορεί να αρχίσει να μεταδίδεται πριν φτάσει όλο το σώμα του στον μεταγωγέα.

χωρίζεται με την σειρά του και αυτό σε μικρότερα μέρη. Τα μέρη αυτά ονομάστηκαν *phits* στην εργασία [Dally 90]. Σ' εκείνη την εργασία δεν γίνεται λόγος για cut through σε επίπεδο flit και αυτο γιατί η υπόθεση που γίνεται είναι ότι το μέγεθος ενός flit θεωρητικά μπορεί να είναι και ίσο με το μέγεθος του phit. Σε αυτήν την περίπτωση η μεταφορά των flit μεταξύ μεταγωγέων γίνεται με store−and−forward. Στην εργασία μας χρησιμοποιούμε cut−through και άρα χωρίζουμε κάθε cell ή flit σε phits. Τα phits κατατάσσονται σε δύο κατηγορίες:

- επικεφαλίδα
- σώμα

Υπάρχει ένα bit που διαχωρίζει τα δύο είδη phit και η χρησιμότητά τους είναι ότι επιτρέπεται η ύπαρξη μεγάλων σε μέγεθος cell/flit που δεν μπορούν να μεταδοθούν σε ένα κύκλο ρολογιού.

Κεφάλαιο 3

Προσομοίωση

Η σύγκριση των δύο αρχιτεκτονικών ATM και wormhole έγινε με προσομοίωση. Αναπτύχθηκαν μοντέλα που περιγράφουν με λεπτομέρεια την λειτουργία κάθε μεταγωγέα και χρησιμοποιώντας την βιβλιοθήκη υποστήριξης CSIM, υλοποιήθηκαν οι προσομοιωτές. Και στα δύο μοντέλα δίνεται έμφαση στην ροή της πληροφορίας ελέγχου μέσα στον μεταγωγέα. Οι εξυπηρετητές, που φαίνονται με κύκλους στα μοντέλα που θα δούμε στην συνέχεια, διαχειρίζονται την πληροφορία ελέγχου. Όταν ένα cell/flit εισέλθει στον μεταγωγέα, θα πρέπει να βρεθεί από ποια έξοδο θα πρέπει να βγει και αυτή η πληροφορία βρίσκεται στον πίνακα δρομολόγησης. Κατόπιν θα πρέπει να διαπιστωθεί αν ο επόμενος μεταγωγέας μπορεί να δεχθεί το cell/flit. Αυτή η πληροφορία βρίσκεται στους πίνακες του ελέγχου ροής. Όταν διαπιστωθεί ότι η μετάδοση του cell/flit μπορεί να γίνει, το cell/flit μεταδίδεται το συντομότερο δυνατό. Τα δεδομένα που περιέχονται σε κάθε cell/flit εισέρχονται στην κεντρική μνήμη του μεταγωγέα η οποία είναι *μνήμη παροχής αγωγού (pipeline memory)*. Αυτή η μνήμη είναι σχεδιασμένη ώστε να παρέχει μεγάλους ρυθμούς εξυπηρέτησης για κάθε είσοδο της [KaVE 95]. Χαρακτηριστικό της είναι ότι επιτρέπει να εισέρχονται και να εξέρχονται ταυτόχρονα περισσότερα του ενός cell (8 ή 16) με τον περιορισμό ότι δεν μπορούν να ξεκινήσουν να εισέρχονται ή να εξέρχονται ταυτόχρονα περισσότερα του ενός. Περιγράφουμε στην συνέχεια τα μοντέλα των μεταγωγέων ATM και wormhole καθώς και τα μοντέλα παραγωγού (πηγής) και καταναλωτή (προορισμού) για κάθε μία περίπτωση. Τέλος περιγράφουμε το απλό μοντέλο του καταναλωτή καθώς και την βιβλιοθήκη υποστήριξης.

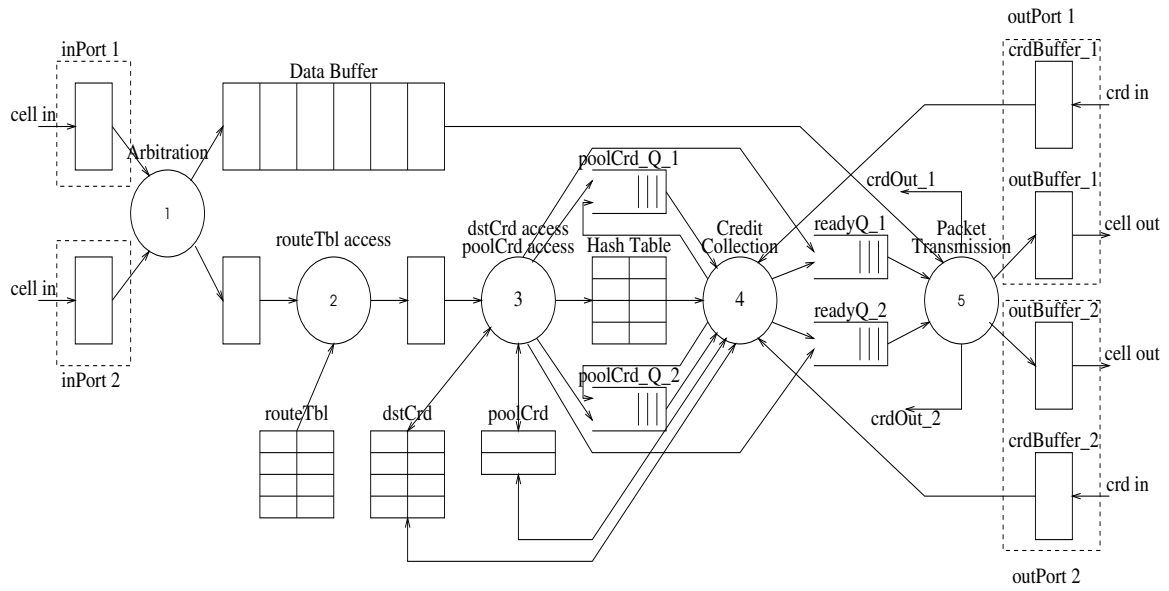
3.1 Περιγραφή μοντέλου μεταγωγέα ATM

Σε έναν μεταγωγέα που ο έλεγχος ροής γίνεται με τον τρόπο που περιγράψαμε στην παράγραφο 2.3, ένα cell δεν μπορεί να φύγει από έναν μεταγωγέα αν δεν πληρούνται οι παρακάτω συνθήκες:

- υπάρχει χώρος στην μνήμη του επόμενου μεταγωγέα που θέλει θα μεταδοθεί το cell, για να το κρατήσει σε περίπτωση που αυτό δεν μπορεί αυτό να προχωρήσει.
- δεν υπάρχουν πολλά cell στον επόμενο μεταγωγέα που να έχουν φύγει από τον παρόντα και να κατευθύνονται στον ίδιο προορισμό.

Η παρατήρηση που μπορεί να γίνει είναι ότι δεν έχει νόημα για μία ροή κυκλοφορίας να καταλάβει πολλές θέσεις μνήμης σε έναν μεταγωγέα γιατί αυτό δεν πρόκειται να επιταχύνει τον ρυθμό με τον οποίο η κίνηση αυτή φεύγει από το δίκτυο. Αυτή η παράμετρος καθορίζεται από την δυνατότητα του παραλήπτη να βγάζει την κίνηση από το δίκτυο. Δεν χρειάζεται λοιπόν να επιτρέπουμε σε ένα μεταγωγέα να κρατά πολλά cells προς τον ίδιο προορισμό. Αυτό μπορεί να έχει δυσάρεστες επιπτώσεις στην αποδοτικότητα του μεταγωγέα, γιατί αν υπάρχει κάποιος προορισμός προς στον οποίο υπάρχει

μεγάλη κίνηση, μπορεί όλος ο διαθέσιμος χώρος αποθήκευσης του μεταγωγέα να καταληφθεί από αυτήν την κίνηση και να μπλοκαριστεί, μην αφήνοντας κίνηση που κατευθύνεται προς άλλους προορισμούς να προχωρήσει.



Σχήμα 3.1: Μοντέλο μεταγωγέα ATM 2×2

Το μοντέλο του ATM μεταγωγέα φαίνεται στο σχήμα 3.1. Οι κύριες δομές μνήμης που φαίνονται είναι οι ακόλουθες:

- routeTbl: ο πίνακας που κρατάει τις απαραίτητες πληροφορίες για την δρομολόγηση. Έχει μέγεθος στην γενική περίπτωση, όσο και το πλήθος των προορισμών στο δίκτυο και δείκτης σε αυτόν τον πίνακα είναι η διεύθυνση προορισμού. Για κάθε προορισμό, υποδεικνύει ποιος είναι ο σύνδεσμος εξόδου του μεταγωγέα που οδηγεί προς τα εκεί.

- dstCrd: ο πίνακας που κρατά το πλήθος των διαθέσιμων εισιτηρίων προς κάθε πιθανό προορισμό. Στην γενική περίπτωση έχει μέγεθος όσο και το πλήθος των προορισμών στο δίκτυο.

- poolCrd: ένα σύνολο μετρητών που κρατά το πλήθος των εισιτηρίων αναχώρησης που είναι διαθέσιμα προς τους επόμενους μεταγωγείς. Το πλήθος τους είναι όσοι και οι σύνδεσμοι εξόδου του μεταγωγέα.

- Hash Table: ένας πίνακας που κρατά περιγραφείς πακέτων που δεν μπορούν να προχωρήσουν στον επόμενο μεταγωγέα γιατί δεν υπάρχει διαθέσιμο εισιτήριο προορισμού. Επειδή θέλουμε να κρατήσουμε το μέγεθός του μικρό, αντί να έχουμε μία θέση του πίνακα για κάθε προορισμό, περισσότεροι του ενός προορισμού αναφέρονται στην ίδια θέση. Ένα cell που δεν έχει εισιτήριο προορισμού, αφήνει τον περιγραφέα του σε αυτόν τον πίνακα, ανεξαρτήτως από το εάν υπάρχει εισιτήριο αναχώρησης για τον επόμενο μεταγωγέα και δεν καταναλώνει κανένα εισιτήριο.

- poolCrd_Q_x: ουρές που κρατάνε περιγραφείς cell που δεν μπορούν να προχωρήσουν προς τον επόμενο μεταγωγέα γιατί δεν υπάρχει εισιτήριο αναχώρησης προς τα εκεί. Υπάρχει μία για κάθε σύνδεσμο εξόδου. Ένας περιγραφέας για να μπει σε αυτήν την ουρά πρέπει πρώτα να έχει καταναλώσει ένα εισιτήριο προορισμού.

- readyQ_x: ουρές που κρατάνε περιγραφείς cells που έχουν καταναλώσει και από τα δύο είδη εισιτηρίων, και άρα είναι έτοιμα να προωθηθούν προς τον επόμενο μεταγωγέα. Υπάρχει μία τέτοια ουρά για κάθε σύνδεσμο εξόδου.

Στο μοντέλο ενός ATM μεταγωγέα 2×2 που φαίνεται στο σχήμα υπάρχουν 5 εξυπηρετητές, ο καθένας από τους οποίους αναλαμβάνει λειτουργίες που μπορούν να γίνουν μέσα σε ένα κύκλο ρολογιού. Οι λειτουργίες κάθε εξυπηρετητή είναι οι ακόλουθες:

- Arbitration: Ο εξυπηρετητής αυτός αναλαμβάνει να δεχθεί τα phits που φτάνουν στις εισόδους του μεταγωγέα. Ελέγχει τι είδους phit είναι το επόμενο που παραλαμβάνει, επικεφαλίδα (header) ή σώμα (body). Όταν διαβάσει μία επικεφαλίδα από κάποια είσοδο, κρατάει κάποια πληροφορία ότι από αυτήν την είσοδο πρέπει να περιμένει ένα ολόκληρο cell που ακολουθεί. Για κάθε είσοδο υπάρχει ένας μετρητής που μετράει πόσα phits ενός cell έχουν παραληφθεί. Επειδή η κεντρική μνήμη του μεταγωγέα είναι της μορφής *μνήμης αγωγού (pipeline memory)*, για να κρατηθεί η ιδιότητα που θέλει ένα το πολύ cell να εισέρχεται στην μνήμη σε κάθε κύκλο ρολογιού, ο εξυπηρετητής δεν αφήνει δύο επικεφαλίδες, προερχόμενες από διαφορετικές εισόδους, να περάσουν στον εξυπηρετητή routeTbl access κατά την διάρκεια του ίδιου κύκλου. Ο εξυπηρετητής Arbitration πρέπει να λειτουργεί σε μεγάλη ταχύτητα γιατί πρέπει να μπορεί να εξυπηρετήσει όλες τις εισόδους ταυτόχρονα[1], αφού ο μόνος περιορισμός είναι ως προς τον αριθμό των επικεφαλίδων που μπορούν να εξυπηρετηθούν στον ίδιο κύκλο. Εφόσον από όλες τις εισόδους έρχονται phits σώματα (body phits), αυτά πρέπει να προωθούνται όλα προς τον Data Buffer.

- routeTbl access: Ο εξυπηρετητής αυτός δέχεται σαν είσοδο ένα phit επικεφαλίδα που περιέχει μία διεύθυνση προορισμού. Χρησιμοποιώντας αυτήν την διεύθυνση, γίνεται πρόσβαση στον Πίνακα Δρομολόγησης (Routing Table) για να βρεθεί από ποια έξοδο θα βγει το cell του οποίου η επικεφαλίδα είναι στον εξυπηρετητή routeTbl access. Ο εξυπηρετητής routeTbl access θα πρέπει να δώσει στον επόμενο εξυπηρετητή ένα περιγραφέα πακέτου. Ο περιγραφέας αυτός περιέχει τις ακόλουθες πληροφορίες:

  -- Διεύθυνση Προορισμού

  -- Σύνδεσμο εξόδου (outgoing link)

  -- Σύνδεσμο εισόδου (incoming link)

- dstCrd/poolCrd access: Ο εξυπηρετητής αυτός δέχεται σαν είσοδο έναν περιγραφέα cell και αναλαμβάνει όλες τις λειτουργίες που σχετίζονται με τον Έλεγχο Ροής (Flow Control). Ελέγχει τον πίνακα dstCrd στην θέση που του υποδεικνύει η διεύθυνση προορισμού που περιλαμβάνεται στον περιγραφέα, καθώς και τον μετρητή poolCrd για τον σύνδεσμο εξόδου που περιλαμβάνεται στον περιγραφέα, για να δει αν υπάρχει εισιτήριο αναχώρησης (pool credit) και εισιτήριο προορισμού. Ανάλογα με την ύπαρξη ή μη εισιτηρίων οι λειτουργίες που μπορεί να γίνουν είναι οι ακόλουθες:

  -- αν και τα δύο είδη εισιτηρίων είναι διαθέσιμα, ο περιγραφέας προωθείται στην ουρά readyQ_x, όπου x ο σύνδεσμος εξόδου από τον οποίο θα φύγει το cell. Σε αυτήν την περίπτωση καταναλώνεται ένα εισιτήριο από κάθε είδος δηλαδή μειώνονται κατά ένα και ο μετρητής poolCrd και η τιμή του πίνακα dstCrd.

  -- αν δεν υπάρχει εισιτήριο προορισμού, τότε ο περιγραφέας προωθείται στο Hash Table και θα περιμένει εκεί μέχρι να φτάσει στον μεταγωγέα κάποιο εισιτήριο για τον προορισμό του. Η

[1] Στην ειδική περίπτωση που το μέγεθος (σε phits) του cell είναι μικρότερο από το πλήθος των εισόδων του μεταγωγέα, αρκεί ο εξυπηρετητής Arbitration να είναι αρκετά γρήγορος ώστε να μπορεί να εξυπηρετήσει τόσες εισόδους όσο το μέγεθος του cell.

περίπτωση αυτή είναι πολύ πιθανό να συμβεί αν υπάρχει μεγάλη κυκλοφορία προς κάποιον προορισμό και καταναλώνονται ταχύτερα τα εισιτήρια προορισμού απ' ότι τα εισιτήρια αναχώρησης. Ακόμα και αν υπάρχει εισιτήριο αναχώρησης, αυτό δεν καταναλώνεται για να αποφευχθούν *αδιέξοδα (deadlocks)*.

-- αν υπάρχει εισιτήριο προορισμού αλλά δεν υπάρχει εισιτήριο αναχώρησης, ο περιγραφέας γράφεται στο τέλος της ουράς poolCrd_Q_x, όπου x ο σύνδεσμος εξόδου από τον οποίο θέλει να μεταδοθεί το πακέτο. Η περίπτωση αυτή είναι πιθανό να συμβεί αν πολλά cell με διαφορετικούς προορισμούς, προσπαθήσουν να βγουν από τον ίδιο σύνδεσμο εξόδου.

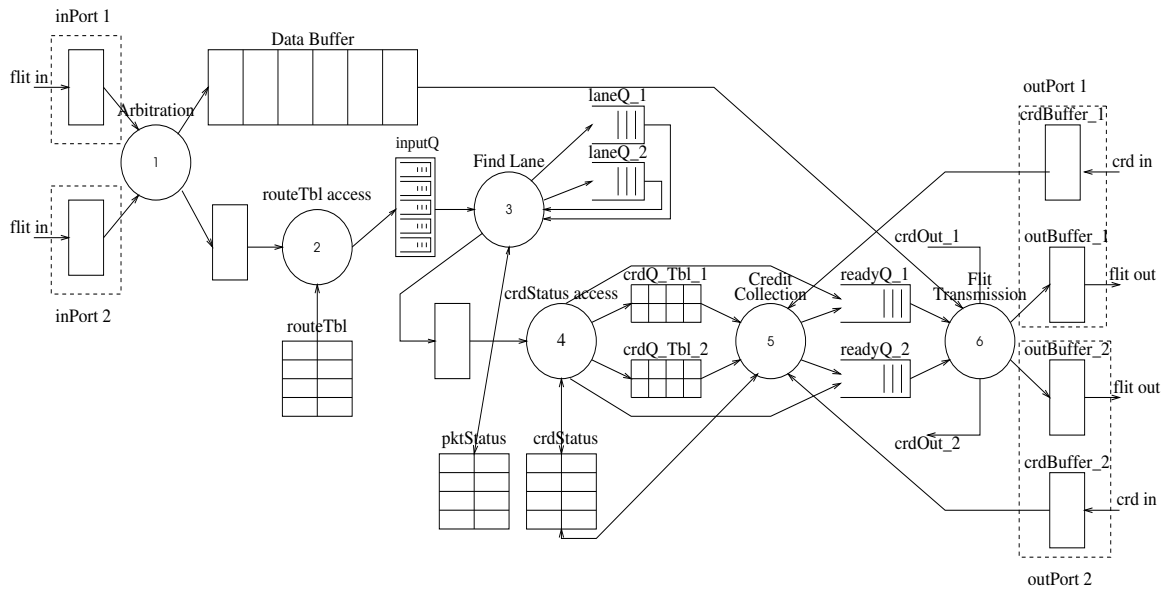
- Credit Collection: Ο εξυπηρετητής αυτός δέχεται σαν είσοδο ένα εισιτήριο και αναλαμβάνει τις λειτουργίες που πρέπει να γίνουν όταν φτάνει ένα εισιτήριο στον μεταγωγέα. Το εισιτήριο μεταφέρει την διεύθυνση προορισμού για την οποία υπάρχει ελεύθερος χώρος στον επόμενο μεταγωγέα. Εστω ότι αυτή η διεύθυνση είναι η Δ. Ελέγχεται κατ' αρχήν αν υπάρχει cell που περιμένει στο Hash Table και πηγαίνει στο Δ. Αν υπάρχει, ο περιγραφέας φεύγει από το Hash Table και προστίθεται στο τέλος της ουράς poolCrd_Q_x, όπου x είναι ο σύνδεσμος εξόδου από τον οποίο πρέπει να φύγει το cell και καταναλώνει το εισιτήριο προορισμού. Αν δεν υπάρχει cell που να περιμένει στο Hash Table, τότε αυξάνεται κατά ένα το πλήθος των εισιτηρίων προς τον Δ στον πίνακα dstCrd. Κατόπιν ελέγχεται η ουρά poolCrd_Q_x για τον σύνδεσμο εξόδου που μόλις έλαβε το εισιτήριο. Αν υπάρχει περιγραφέας που να περιμένει στην κεφαλή της, αυτός φεύγει από την ουρά και προστίθεται στο τέλος της ουράς readyQ_x, όπου x ο σύνδεσμος εξόδου του cell, καταναλώνοντας και το εισιτήριο αναχώρησης. Το cell αυτό είναι έτοιμο να φύγει από τον μεταγωγέα, αφού έχει εξασφαλιστεί ότι υπάρχει ελεύθερος χώρος γι αυτό στον επόμενο μεταγωγέα. Αν η κεφαλή της ουράς poolCrd_Q_x είναι κενή, αυξάνονται κατά ένα τα εισιτήρια αναχώρησης για τον επόμενο μεταγωγέα.

- Packet Transmission: Ο εξυπηρετητής αυτός δέχεται σαν είσοδο έναν περιγραφέα cell καθώς και τα δεδομένα του cell από τον Data buffer, και αναλαμβάνει να μεταδώσει το cell αυτό στον επόμενο μεταγωγέα. Σε κάθε κύκλο εξυπηρετεί όλους τους συνδέσμους εξόδου, εφόσον υπάρχουν έτοιμα cell που θέλουν να φύγουν. Και εδώ όπως και στην είσοδο, υπάρχει ο περιορισμός ότι μόνο ένα cell μπορεί να ξεκινήσει να μεταδίδεται στην διάρκεια ενός κύκλου. Αυτο ορίζεται από τις προδιαγραφές της Pipeline Memory που είναι το μοντέλο κεντρικής μνήμης που υποθέσαμε. Ο εξυπηρετητής αυτος, όταν ένας σύνδεσμος εξόδου είναι ή πάει να γίνει *άεργος (idle)*, διαβάζει την κεφάλή της ουράς readyQ_x που του αντιστοιχεί και εφόσον δεν είναι κενή βγάζει από την ουρά τον περιγραφέα που είναι στην κεφαλή και αρχίζει να φέρνει και τα δεδομένα από την μνήμη. Μόλις ξεκινήσει να εκπέμπεται το cell, ένα εισιτήριο δημιουργείται και επιστρέφει, στον μεταγωγέα από τον οποίο έφτασε το cell που μεταδίδεται. Την πληροφορία από που ήρθε το cell, την κρατάμε στον περιγραφέα του. Αν υπάρχει cell το οποίο είχε αρχίσει να μεταδίδεται από κάποιον σύνδεσμο εξόδου σε προηγούμενο κύκλο και δεν έχει τελειώσει ακόμα η μετάδοσή του, ο εξυπηρετητής Packet Transmission μεταδίδει το επόμενο phit αυτού του cell.

3.2 Περιγραφή μοντέλου μεταγωγέα Wormhole

Το μοντέλο ενός μεταγωγέα wormhole φαίνεται στο σχήμα 3.2. Το μοντέλο έχει 6 εξυπηρετητές, ο κάθε ένας από τους οποίους αναλαμβάνει λειτουργίες που μπορούν να γίνουν μέσα σε ένα κύκλο ρολογιού. Οι δομές μνήμης που περιλαμβάνονται σε αυτό το σχήμα είναι οι ακόλουθες:

- routeTbl: ο πίνακας που κρατάει τις πληροφορίες που είναι απαραίτητες για να γίνει η δρομο−λόγηση. Έχει μέγεθος στην γενική περίπτωση, όσο και το πλήθος των προορισμών στο δίκτυο.



Σχήμα 3.2: Μοντέλο μεταγωγέα wormhole 2 × 2

Δείκτης σε αυτόν τον πίνακα είναι η διεύθυνση προορισμού. Για κάθε προορισμό, υποδεικνύει ποιος είναι ο σύνδεσμος εξόδου του μεταγωγέα που οδηγεί προς τα εκεί.

- laneQ_x: ουρές στις οποίες παραμένουν οι περιγραφείς των πακέτων που δεν μπορούν να προχωρήσουν γιατί δεν υπάρχει διαθέσιμη λωρίδα. Υπάρχει μία ουρά για κάθε σύνδεσμο εισόδου. Σε αυτές τις ουρές μπαίνουν επικεφαλίδες πακέτων, αφού μονο μία επικεφαλίδα έχει πιθανότητα να μην μπορεί να βρει λωρίδα, μέσα στην οποία να κινηθεί, αφού μόνο αυτή μπορεί να ζητήσει καινούρια λωρίδα.

- pktStatus: πίνακας που κρατάει πληροφορίες σχετικές με τα πακέτα που θέλουν να περάσουν μέσα από τον μεταγωγέα. Είναι ένας διδιάστατος πίνακας όπου δείκτης στην μία διάστασή του είναι ο σύνδεσμος εισόδου από τον οποίο έρχεται το πακέτο και στην δεύτερη διάσταση είναι η λωρίδα μέσα στην οποία κινείται το πακέτο. Σε κάθε θέση του πίνακα περιέχονται δύο πληροφορίες:

  -- Ο σύνδεσμος από τον οποίο πρέπει να βγουν τα flits του πακέτου.

  -- Ο αριθμός της λωρίδας που θα χρησιμοποιήσουν τα flits του πακέτου στον επόμενο μεταγωγέα.

  Το μέγεθός του είναι το γινόμενο του πλήθους των εισόδων του μεταγωγέα επί το πλήθος των λωρίδων του.

- crdStatus: ο πίνακας που κρατάει την πληροφορία για το πόσα εισιτήρια έχει διαθέσιμα μία λωρίδα. Δείκτης σε αυτόν τον πίνακα είναι η λωρίδα για την οποία θέλουμε να βρούμε τα διαθέσιμα εισιτήρια. Το μέγεθός του είναι όσο το πλήθος των λωρίδων του μεταγωγέα.

- crdQ_Tbl_x: πίνακες μέσα στους οποίους είναι οργανωμένες ουρές, μία για κάθε λωρίδα, στις οποίες μπαίνουν flits τα οποία έχουν μεν λωρίδα για τον επόμενο μεταγωγέα, αλλά δεν έχουν εισιτήριο γι αυτόν. Υπάρχει ένας για κάθε σύνδεσμο εξόδου.

- readyQ_x: ουρές που κρατάνε περιγραφείς flits που έχουν καταναλώσει ένα εισιτήριο, και άρα είναι έτοιμα να προωθηθούν προς τον επόμενο μεταγωγέα. Υπάρχει μία τέτοια ουρά για κάθε σύνδεσμο εξόδου.

- inputQ: Η δομή αυτή είναι μια μνήμη οργανωμένη σε μορφή πίνακα ουρών. Υπάρχει μία ουρά για κάθε λωρίδα και το μέγεθος κάθε ουράς είναι όσο το πλήθος των εισιτηρίων για μία λωρίδα. Η ανάγκη δημιουργίας αυτής της δομής περιγράφεται παρακάτω. Εφόσον μία επικεφαλίδα που φτάσει στον μεταγωγέα, δεν μπορεί να βρει λωρίδα για να προωθηθεί στον επόμενο μεταγωγέα, δεν πρέπει τα flit που ακολουθούν την επικεφαλίδα να φτάσουν στον εξυπηρετητή routeTbl access γιατί δεν υπάρχει έγκυρη περιγραφή στον πίνακα pktStatus που να περιγράφει ποιος είναι ο σύνδεσμος εξόδου από τον οποίο πρέπει να βγεί το flit και ποιος ο αριθμός της λωρίδας που θα έχει στον επόμενο μεταγωγέα που θα πρέπει να πάει. Η περιγραφή αυτή θα φτιαχτεί όταν γίνει δυνατό για την επικεφαλίδα να βρει λωρίδα για να προχωρήσει. Άρα θα πρέπει να υπάρχει κάτι που να εμποδίζει ένα flit ενός πακέτου στο οποίο δεν έχει ακόμα ανατεθεί λωρίδα, να προχωρήσει και αυτό μπορεί να είναι η επικεφαλίδα η ίδια η οποία μπλοκάρει την λωρίδα που κατέχει το πακέτο. Κάθε flit εγγράφεται στο τέλος της ουράς που υπάρχει για την λωρίδα μέσα στην οποία κινείται. Επειδή κάθε λωρίδα έχει ξεχωριστή ουρά, δεν παρατηρείται το φαινόμενο, να εμποδίζει μία μπλοκαρισμένη λωρίδα άλλες λωρίδες να προωθήσουν τα δεδομένα τους κανονικά.

Στο σχήμα παρατηρούμε 6 εξυπηρετητές, των οποίων οι λειτουργίες περιγράφονται παρακάτω:

- Arbitration: ο εξυπηρετητής αυτός αναλαμβάνει να επιλέξει ποια είναι η επόμενη είσοδος που θα εξυπηρετηθεί. Όπως και στο μοντέλο ATM, το μοντέλο μνήμης που υποθέτουμε ότι έχει ο μεταγωγέας, είναι μνήμη παροχής αγωγού. Υπάρχει άρα ο περιορισμός του ότι δεν μπορούν να ξεκινούν να εισέρχονται στην μνήμη περισσότερα από ένα flit σε κάθε κύκλο ρολογιού. Εφόσον από μία είσοδο αρχίσει να γίνεται δεκτό ένα flit, θα εξυπηρετείται αυτή η είσοδος σε κάθε επόμενο κύκλο, μέχρι να φτάσει και να παραληφθεί όλο το flit. Όταν το flit αρχίζει να φτάνει, δημιουργείται ένας περιγραφέας του πακέτου, ο οποίος τοποθετείται στην δομή inputQ και όταν αρχίσει το σώμα του πακέτου να φτάνει, αυτό τοποθετείται στην μνήμη.

- routeTbl access: ο εξυπηρετητής αυτός αναλαμβάνει να κάνει την πρόσβαση στον πίνακα δρομολόγησης και να βρει από ποιον σύνδεσμο εξόδου πρέπει να βγει το flit που βρίσκεται στην κεφαλή της ουράς που εξετάζει, εφόσον αυτό είναι επικεφαλίδα. Αφού βρεθεί ποιος είναι ο σύνδεσμος εξόδου του πακέτου, ο περιγραφέας εμπλουτίζεται με αυτήν την πληροφορία και περνάει στον εξυπηρετητή findLane. Οι ουρές του inputQ εξετάζονται κυκλικά (round robin) και εφόσον μία ουρά είναι κενή παραλείπεται. Αν στην κεφαλή της ουράς βρίσκεται flit που δεν είναι επικεφαλίδα πακέτου, αυτό προωθείται προς τον επόμενο εξυπηρετητή χωρίς να γίνεται πρόσβαση στον πίνακα δρομολόγησης.

- findLane: ο εξυπηρετητής αυτός αναλαμβάνει να χειριστεί όλες τις πληροφορίες σχετικά με τα πακέτα. Έχει σαν είσοδο ένα περιγραφέα flit και σκοπός του είναι να τον εμπλουτίσει με την πληροφορία της λωρίδας ή και του συνδέσμου εξόδου που το flit θα χρησιμοποιήσει για τον επόμενο μεταγωγέα.

  -- Όταν ο περιγραφέας αντιστοιχεί σε flit−επικεφαλίδα τότε θα πρέπει να βρει ελεύθερη λωρίδα να αναθέσει στο καινούριο πακέτο. Εφόσον το καταφέρει αυτό, υπαρχει δηλαδή ελεύθερη λωρίδα στον επόμενο μεταγωγέα, η πληροφορία αυτή προστίθεται στον περιγραφέα και αυτός προχωράει στον επόμενο εξυπηρετητή. Ταυτόχρονα γράφεται στον πίνακα pktStatus η πληροφορία για την λωρίδα που ανατέθηκε στο νέο πακέτο. Αν δεν βρεθεί ελεύθερη λωρίδα, το πακέτο πρέπει να μπλοκαριστεί και ο περιγραφέας του γράφεται στο τέλος της ουράς laneQ_x, όπου x ο σύνδεσμος εξόδου από τον οποίο πρέπει να βγει το πακέτο. Η πληροφορία αυτή υπάρχει στον περιγραφέα του flit−επικεφαλίδα γιατί έχει βρεθεί στον προηγούμενο κύκλο μετά την πρόσβαση στον πίνακα δρομολόγησης.

-- Όταν ο περιγραφέας αντιστοιχεί σε flit−σώμα, τότε δεν περιέχει την πληροφορία του ποιος είναι ο σύνδεσμος εξόδου από τον οποίο θα βγει το flit και αυτήν την πληροφορία θα την ανακτήσει ο εξυπηρετητής findLane. Ξέροντας τον σύνδεσμο εισόδου και την λωρίδα που το πακέτο χρησιμοποιεί στον παρόντα μεταγωγέα, γίνεται πρόσβαση στον πίνακα pktStatus και βρίσκεται ο σύνδεσμος εξόδου και η λωρίδα που θα χρησιμοποιήσει το flit στον επόμενο μεταγωγέα. Αυτές οι πληροφορίες προστίθενται στον περιγραφέα και προχωράει στον επόμενο εξυπηρετητή.

-- Όταν ο περιγραφέας είναι για τελικό flit τότε θα πρέπει η λωρίδα που είχε ανατεθεί στο πακέτο να μαρκαριστεί ως ελεύθερη για να μπορεί να χρησιμοποιηθεί από τα επόμενα flit που θα θελήσουν να περάσουν από τον μεταγωγέα. Όπως και στην περίπτωση του flit−σώματος, γίνεται και πάλι πρόσβαση στον πίνακα pktStatus για να βρεθεί σύνδεσμος και λωρίδα εξόδου του flit. Ταυτόχρονα η είσοδος αυτή σβύνεται από τον πίνακα και ανιχνεύεται η ουρά laneQ_x, όπου x είναι ο σύνδεσμος εξόδου που θα χρησιμοποιήσει το τελικό flit. Αν υπάρχει κάποιος περιγραφέας που περιμένει στην κεφαλή αυτής της ουράς, βγαίνει από αυτήν και του ανατίθεται η λωρίδα που μόλις ελευθερώθηκε. Δίνεται ακόμα ένα σήμα στον εξυπηρετητή 2 που τον ειδοποιεί να μείνει άεργος(idle) κατά τον επόμενο κύκλο και να μην διαβάσει την επόμενη ουρά του inputQ γιατί το επόμενο flit που θα πρέπει να εξυπηρετηθεί είναι αυτό του οποίου ο περιγραφέας μόλις βγήκε από την laneQ_x. Ετοιμάζεται και η έξοδος του εξυπηρετητή που είναι ο περιγραφέας του τελικού flit εμπλουτισμένος με τις πληροφορίες για την λωρίδα και σύνδεσμο εξόδου καθώς και η είσοδός του, που θα είναι ο περιγραφέας του flit−επικεφαλίδας όπως τον είχε ετοιμάσει ο εξυπηρετητής 2 πριν μπει στην laneQ_x ουρά.

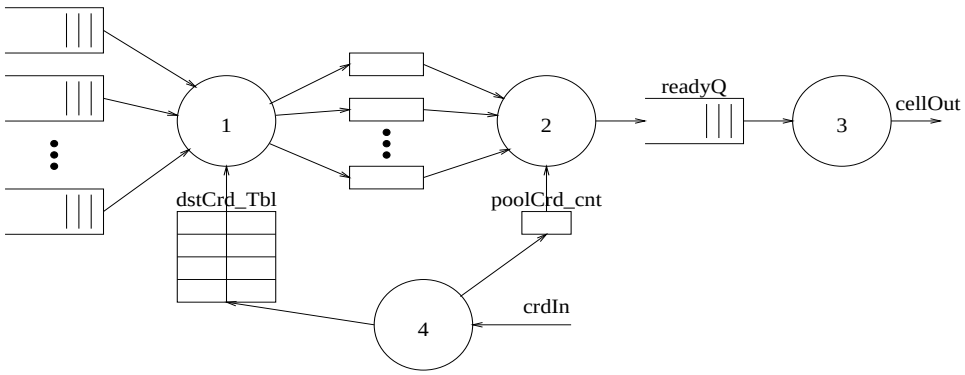
- crdStatus access: ο εξυπηρετητής αυτός αναλαμβάνει τις λειτουργίες που σχετίζονται με τον έλεγχο ροής στον μεταγωγέα. Σαν είσοδο έχει έναν περιγραφέα που περιέχει σύνδεσμο και λωρίδα εξόδου. Με αυτές τις πληροφορίες, γίνεται πρόσβαση στον πίνακα crdStatus για να βρεθεί αν υπάρχουν διαθέσιμα εισιτήρια για την λωρίδα αυτού του συνδέσμου εξόδου. Αν υπάρχουν, τότε ο περιγραφέας γράφεται στο τέλος την ουράς readyQ_x, όπου x ο σύνδεσμος εξόδου, και καταναλώνεται ένα αντίστοιχο εισιτήριο. Αν δεν υπάρχουν εισιτήρια, τότε ο περιγραφέας γράφεται στην crdQ_Tbl_x, όπου x ο σύνδεσμος εξόδου από τον οποίο πρέπει να βγει το flit. Μετά, ο περιγραφέας τοποθετείται στην ουρά που αντιστοιχεί στην λωρίδα που θα κινηθεί το flit στον επόμενο μεταγωγέα.

- Credit Collection: ο εξυπηρετητής αυτός αναλαμβάνει τις λειτουργίες που πρέπει να γίνουν, όταν ένα εισιτήριο φτάσει στον μεταγωγέα. Έχει σαν είσοδό του ένα εισιτήριο το οποίο περιέχει την λωρίδα για την οποία υπάρχει μία ελεύθερη θέση στον επόμενο μεταγωγέα. Επίσης ανιχνεύεται η ουρά που αντιστοιχεί σε αυτήν την λωρίδα στον crdQ_Tbl_x, όπου x ο σύνδεσμος εξόδου τον οποίο αφορά το εισιτήριο που μόλις ήρθε. Εφόσον δεν είναι άδεια, ο περιγραφέας που είναι στην κορυφή, βγαίνει από την ουρά και τοποθετείται στο τέλος της ουράς readyQ_x. Αν η ουρά είναι κενή, κανένα flit δεν περιμένει αυτό το εισιτήριο, οπότε αυξάνεται το πλήθος των εισιτηρίων στον πίνακα crdStatus για την λωρίδα του συνδέσμου εξόδου απ' όπου ήρθε το εισιτήριο.

- Flit Transmission: ο εξυπηρετητής αυτός αναλαμβάνει τις λειτουργίες που πρέπει να γίνουν όταν ένα flit πρέπει να μεταδοθεί. Σαν είσοδο έχει έναν περιγραφέα που βρίσκεται στην κεφαλή της ουράς readyQ_x για όλους τους συνδέσμους εξόδου. Ο περιορισμός που ισχύει και εδώ είναι ότι δεν μπορούν περισσότερα του ενός flit να αρχίσουν να μεταδίδονται ταυτόχρονα από διαφορετικές εξόδους λόγω του περιορισμού της μνήμης παροχής αγωγού του μεταγωγέα. Εφόσον η ουρά readyQ_x ενός συνδέσμου εξόδου δεν είναι άδεια, ο σύνδεσμος είναι ή θα είναι σύντομα άεργος

(idle) και δεν έχει ήδη ξεκινήσει να μεταδίδεται άλλο flit κατά την διάρκεια του ιδίου κύκλου, ανακαλείται από την μνήμη το περιεχόμενο του flit του οποίου ο περιγραφέας βρίσκεται στην κεφαλή της ουράς. Ετοιμάζεται η επικεφαλίδα που θα προηγηθεί του σώματος, για να μεταδοθεί και το αντίστοιχο εισιτήριο στον μεταγωγέα από τον οποίο προήλθε το flit. Στο εισιτήριο περιέχεται και η λωρίδα που το flit χρησιμοποίησε στον παρόντα μεταγωγέα γιατί ο πίνακας των εισιτηρίων που υπάρχει για κάθε σύνδεσμο εξόδου ενός μεταγωγέα, έχει σαν δείκτη τον αριθμό της λωρίδας που το flit θα χρησιμοποιήσει στον επόμενο μεταγωγέα. Αυτή η πληροφορία υπάρχει στον περιγραφέα του flit που είναι στην κεφαλή της ουράς readyQ_x. Αν υπάρχει flit το οποίο είχε αρχίσει να μεταδίδεται από κάποιον σύνδεσμο εξόδου σε προηγούμενο κύκλο και δεν έχει τελειώσει ακόμα η μετάδοσή του, ο εξυπηρετητής Flit Transmission μεταδίδει το επόμενο phit αυτού του flit.

3.3 Περιγραφή μοντέλου παραγωγού για δίκτυα ATM

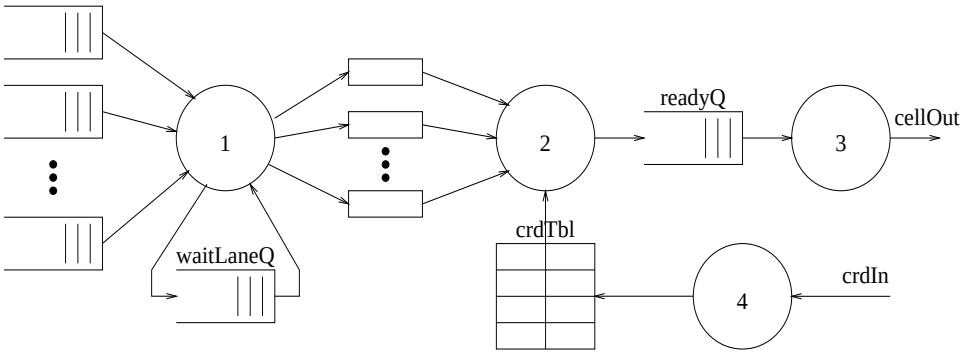


Σχήμα 3.3: Μοντέλο παραγωγού για την περίπτωση του ATM

Στις εισόδους του δικτύου υπάρχουν μηχανισμοί που ελέγχουν την κίνηση που περνάει μέσα σε αυτό και υπακούουν και αυτοί στους κανόνες που επιβάλλει ο έλεγχος ροής(flow control). Αυτό συμβαίνει για να μην μπαίνει στον πρώτο μεταγωγέα του δικτύου κίνηση με διαφορετικές ιδιότητες από αυτήν που μπαίνει στους μετέπειτα μεταγωγείς. Το μοντέλο φαίνεται στο σχήμα 3.3. Αρχικά παράγεται ένα πακέτο και επιλέγεται ποιος είναι ο προορισμός του και το μήκος του. Ανάλογα με το μοντέλο κίνησης ο προορισμός μπορεί να επιλεγεί ομοιόμορφα ή να υπάρχει κάποια προτίμηση προς κάποιον η κάποιους προορισμούς (hot spot traffic). Το καινούργιο πακέτο μπαίνει στην ουρά που προσδιορίζεται από τον προορισμό του. Μία ουρά επιλέγεται για να εξυπηρετηθεί με κριτήριο τον χρόνο άφιξης του πακέτου που βρίσκεται στην κορυφή της. Ο εξυπηρετητής 1 ελέγχει αν υπάρχει εισιτήριο προορισμού για το πακέτο και εφόσον υπάρχει, το πακέτο βγαίνει από την αρχική ουρά, κερματίζεται σε cells, και μπαίνει σε έναν προσωρινό χώρο αποθήκευσης. Υπάρχει ένας τέτοιος χώρος για κάθε προορισμό. Από εκεί θα φύγει κάθε cell με την σειρά του. Οι χώροι αποθήκευσης εξυπηρετούνται από τον εξυπηρετητή 2 κυκλικά (round robin). Ο εξυπηρετητής αυτός ελέγχει τον μετρητή που δείχνει πόσα εισιτήρια είναι διαθέσιμα και εφόσον ο μετρητής δεν είναι μηδενικός, ένα cell περνάει από τον προσωρινό χώρο αποθήκευσης, στην ουρά των έτοιμων για αναχώρηση cell και ο μετρητής μειώνεται. Από εκεί τα cells προωθούνται στο δίκτυο από τον εξυπηρετητή 3, ο οποίος στέλνει το επόμενο cell στον πρώτο μεταγωγέα. Υπάρχει ακόμα και ο εξυπηρετητής 4 ο οποίος δέχεται τα εισιτήρια που επιστρέφουν από τον πρώτο μεταγωγέα, και ενημερώνει τον πίνακα των εισιτηρίων προορισμού καθώς και τον μετρητή των εισιτηρίων αναχώρησης. Για τους σκοπούς της προσομοίωσης, δεν λαμβάνεται υπ' όψη ο χρονισμός σε αυτό το μοντέλο και ο βασικότερος λόγος είναι ότι η πηγή υποχρεούται να στέλνει χωρίς διακοπές. Για να έχουμε χρησιμοποίηση των συνδέσμων εξόδου 100% χρειάστηκε η καθυστέρηση κάθε εξυπηρετητή στον μηχανισμό του παραγωγού (πηγής) να είναι μηδενική. Η μόνη καθυστέρηση που δεν πρέπει να είναι μηδενική, είναι η καθυστέρηση του εξυπηρετητή 3 γιατί πρέπει κάθε cell να μεταδίδεται σε μη μηδενικό χρόνο, ώστε να δίνεται ο χρόνος στον πρώτο μεταγωγέα να το απορροφήσει.

3.4 Περιγραφή μοντέλου παραγωγού για δίκτυα Wormhole

Και στην περίπτωση του wormhole η πηγή κίνησης πρέπει να υπακούει στους κανόνες που επιβάλλει ο έλεγχος ροής. Υπάρχει λοιπόν και εδώ ένας μηχανισμός που μορφοποιεί την κίνηση που μπαίνει μέσα στο δίκτυο, ώστε να μοιάζει με την κίνηση που θα έβγαινε από έναν μεταγωγέα. Όπως και στην περίπτωση ATM, παράγεται ένα πακέτο και επιλέγεται το μήκος και ο προορισμός του ανάλογα με την μορφή της κίνησης που θέλουμε να προσομοιώσουμε. Το μοντέλο φαίνεται στο σχήμα 3.4.



Σχήμα 3.4: Μοντέλο παραγωγού στην περίπτωση του wormhole

Το πακέτο αυτό μπαίνει αρχικά σε μία ουρά ανάλογα με τον προορισμό του. Αυτό δεν είναι απαραίτητο για το μοντέλο wormhole, αλλά για να μην επιτρέπεται να μεταδίδονται ταυτόχρονα δύο πακέτα από την ίδια πηγή προς τον ίδιο προορισμό, τέτοια πακέτα μπαίνουν σε μία ουρά στην είσοδο και περιμένει το ένα μέχρι να τελειώσει το άλλο. Αυτό έγινε για να μοιάζει περισσότερο η μορφή της κίνησης που μπαίνει μέσα σε ένα δίκτυο ATM, με αυτήν που μπαίνει σε ένα δίκτυο wormhole. Από αυτές τις ουρές, ο εξυπηρετητής 1 παίρνει τα πακέτα και εφόσον υπάρχει ελεύθερη λωρίδα και δεν μεταδίδεται ταυτόχρονα άλλο πακέτο προς τον ίδιο προορισμό, τα χωρίζει σε μικρότερα τμήματα (flit). Αν δεν υπάρχει ελεύθερη λωρίδα, το πακέτο μπαίνει στο τέλος της ουράς των πακέτων που περιμένουν να βρουν ελεύθερη λωρίδα. Όταν ένα από τα πακέτα που τους έχει δοθεί λωρίδα, ολοκληρώσει την μετάδοσή του, το πακέτο που βρίσκεται στην κεφαλή αυτής της ουράς, παίρνει την λωρίδα που μόλις ελευθερώθηκε και συνεχίζει την πορεία του προς το δίκτυο. Τα πακέτα από τις ουρές στην είσοδο εξυπηρετούνται με την σειρά με την οποία μπήκαν μέσα σε κάποια από αυτές, δηλαδή FCFS. Το πακέτο, σε μορφή ακολουθίας flits, τοποθετείται σε έναν προσωρινό χώρο αποθήκευσης. Υπάρχει ένας τέτοιος χώρος αποθήκευσης για κάθε λωρίδα (lane).

Από τους χώρους αποθήκευσης ο εξυπηρετητής 2 παίρνει flits με κυκλική σειρά και τα περνάει στην ουρά των έτοιμων προς αναχώρηση flit, εφόσον ο μετρητής των εισιτηρίων για τον πρώτο μεταγωγέα δεν είναι μηδενικός και άρα υπάρχουν διαθέσιμα εισιτήρια. Από την ουρά των έτοιμων για αναχώρηση flit, ο εξυπηρετητής 3 παίρνει το flit που είναι στην κεφαλή της και το μεταδίδει στον πρώτο μεταγωγέα. Υπάρχει ακόμα και ο εξυπηρετητής 4, ο οποίος είναι επιφορτισμένος με το έργο να μαζεύει τα εισιτήρια που γυρίζουν στην πηγή και να ενημερώνει τον μετρητή που λέει πόσα είναι διαθέσιμα ανά πάσα στιγμή. Όπως και στην περίπτωση του ATM, οι καθυστερήσεις όλων των εξυπηρετητών πλην του 3 είναι μηδενικές. Η καθυστέρηση του εξυπηρέτη 3 είναι μη μηδενική γιατί χρειάζεται κάποιο χρόνο μη μηδενικό, ο πρώτος μεταγωγέας, για να απορροφήσει το flit μέσα στο δίκτυο (ανάλογα με το ATM μοντέλο).

3.5 Περιγραφή μοντέλου καταναλωτή

Ο καταναλωτής, βγάζει από το δίκτυο ένα flit/cell μόλις φτάσει σε αυτόν. Μόλις παραληφθεί η επικεφαλίδα του flit/cell και βρεθούν τα στοιχεία που πρέπει να συμπεριληφθούν και στο εισιτήριο που θα επιστρέψει στον τελευταίο μεταγωγέα, τότε παράγεται αυτό το εισιτήριο. Κατόπιν, συνεχίζεται η παραλαβή του flit/cell. Ο καταναλωτής δεν έχει καθυστέρηση αλλά μόλις ένα flit/cell φτάσει σε αυτόν, το αποσύρει αμέσως από το δίκτυο.

3.6 Περιγραφή της βιβλιοθήκης υποστήριξης

Ο προσομοιωτής υλοποιήθηκε χρησιμοποιώντας την γλώσσα προγραμματισμού C και τη βιβλιοθήκη υποστήριξης CSIM. Η βιβλιοθήκη αυτή συνδέεται (links) με τον κώδικα του προσομοιωτή και του παρέχει μία σειρά από εντολές. Η βιβλιοθήκη δίνει έτοιμες ρουτίνες και βοηθάει ώστε να προσομοιώνεται εύκολα οποιοδήποτε σύστημα που μπορεί να μοντελοποιηθεί με μία σειρά από εξυπηρετητές και ουρές ή άλλες δομές μνήμης. Παρέχονται ρουτίνες που υλοποιούν το περιβάλλον προσομοίωσης και ό,τι χρειάζεται ένας προσομοιωτής που οδηγείται από γεγονότα (event driven simulator). Οτιδήποτε συμβαίνει κατά την διάρκεια της προσομοίωσης, αποτελεί ένα γεγονός. Υπάρχει ένα λογικό ρολόι, βάσει του οποίου προγραμματίζονται τα γεγονότα που πρόκειται να συμβούν. Γεγονότα που πρέπει να συμβούν ταυτόχρονα, εκτελούνται σειριακά, αλλά ο χρόνος προσομοίωσης δεν αυξάνει μέχρις ότου εκτελεστούν όλα τα γεγονότα που είναι προγραμματισμένα να συμβούν σε μία χρονική στιγμή. Μόλις ξεκινάει η προσομοίωση υπάρχει μία βασική ρουτίνα που αρχικοποιεί το προς προσομοίωση σύστημα.

Οι ενεργές οντότητες του συστήματος μπορούν να υλοποιηθούν ως ξεχωριστά νήματα (threads). Παρέχεται μηχανισμός ώστε μέσα στα πλαίσια του πεδίου διευθύνσεων μίας διεργασίας να μπορούν διαφορετικά νήματα εκτέλεσης εντολών να είναι ενεργά. Με αυτόν τον τρόπο, χωρίς να χρειάζεται η επέμβαση του λειτουργικού συστήματος, που είναι ιδιαιτέρως χρονοβόρα, μπορεί να εναλλάσσεται η ροή εκτέλεσης μεταξύ των διαφόρων νημάτων που αποτελούν τις ενεργές οντότητες του προσομοιωτή. Σε ένα μοντέλο προσομοίωσης ενός δικτύου από μεταγωγείς, ενεργές οντότητες είναι οι μεταγωγείς, οι παραγωγοί (πηγές) και οι καταναλωτές (προορισμοί). Μπορεί λοιπόν να υπάρχει ένα νήμα εκτέλεσης για κάθε έναν από αυτούς και με κάποιον τρόπο να επικοινωνούν μεταξύ τους.

Στο σύστημα που προσομοιώνεται μπορεί να υπάρχουν πόροι από τους οποίους κάποιο νήμα μπορεί να ζητάει εξυπηρέτηση. Για να προσομοιωθεί ένας πόρος, το CSIM παρέχει την δομή facility. Σε ένα υπολογιστικό σύστημα, facility μπορεί να είναι ένας δίσκος ή μία κεντρική μονάδα επεξεργασίας. Όλες οι αιτήσεις για εξυπηρέτηση από ένα facility μπαίνουν σε μία ουρά και ανάλογα με το πλήθος των εξυπηρετητών που το facility διαθέτει, εξυπηρετούνται είτε σειριακά είτε με κάποιον άλλο τρόπο που απαιτεί η προσομοίωση. Τα νήματα, επικοινωνούν μεταξύ τους είτε μέσω των τρόπων που παρέχει η C, δηλαδή μέσω εξωτερικών (global) μεταβλητών, είτε μέσω ανταλλαγής μηνυμάτων. Είναι δυνατόν να οριστούν γραμματοκιβώτια (mailboxes) στα οποία, ένα νήμα μπορεί να στείλει ένα μήνυμα σε κάποιο άλλο. Με αυτόν τον τρόπο θα μπορούσαν cells να περνάνε από έναν παραγωγό σε έναν μεταγωγέα, ή από έναν μεταγωγέα σε έναν άλλο. Όταν ένα νήμα δημιουργεί ένα άλλο, ένας τρόπος για να του περάσει ορισμένες πληροφορίες είναι με πέρασμα παραμέτρων κατά την κλήση. Δεν είναι δυνατο πάντως ένα νήμα να επιστρέψει μία τιμή σε αυτόν που το ενεργοποίησε.

Ο λόγος για τον οποίο γίνεται η προσομοίωση, είναι για να συλλεγούν ορισμένα στατιστικά στοιχεία σχετικά με το σύστημα που προσομοιώνεται. Το CSIM παρέχει έτοιμες δομές που συλλέγουν και επεξεργάζονται στατιστικά στοιχεία και τα οργανώνουν είτε σε μορφή πίνακα, οπότε δίδονται η μέση τιμή, η διασπορά, το μέγιστο και το ελάχιστο της μέτρησης στην οποία αναφέρεται ο πίνακας, είτε φτιάχνουν το ιστόγραμμα όπου παρέχεται το πλήθος των μετρήσεων που παρατηρήθηκαν μέσα σε καθορισμένες περιοχές του πεδίου τιμών. Από τα πιο σημαντικά εργαλεία που παρέχει το CSIM είναι η γεννήτρια τυχαίων αριθμών καθώς και η δυνατότητα η γεννήτρια αυτή να παράγει αριθμούς που ακολουθούν μία επιθυμητή κατανομή όπως εκθετική, κανονική, ομοιόμορφη και άλλες. Μπορούν μάλιστα να παράγονται είτε ακέραιοι, είτε πραγματικοί αριθμοί που ακολουθούν αυτές τις κατανομές. Αν, για παράδειγμα, χρειαζόμαστε έναν παραγωγό που να γεννάει κίνηση Poisson, αρκεί να διαλέξουμε μία εκθετική καθυστέρηση μεταξύ διαδοχικών γεννήσεων πακέτων, ενώ εάν θέλουμε, μπορούμε να επιλέξουμε ομοιόμορφα έναν προορισμό μέσα στο δίκτυο.

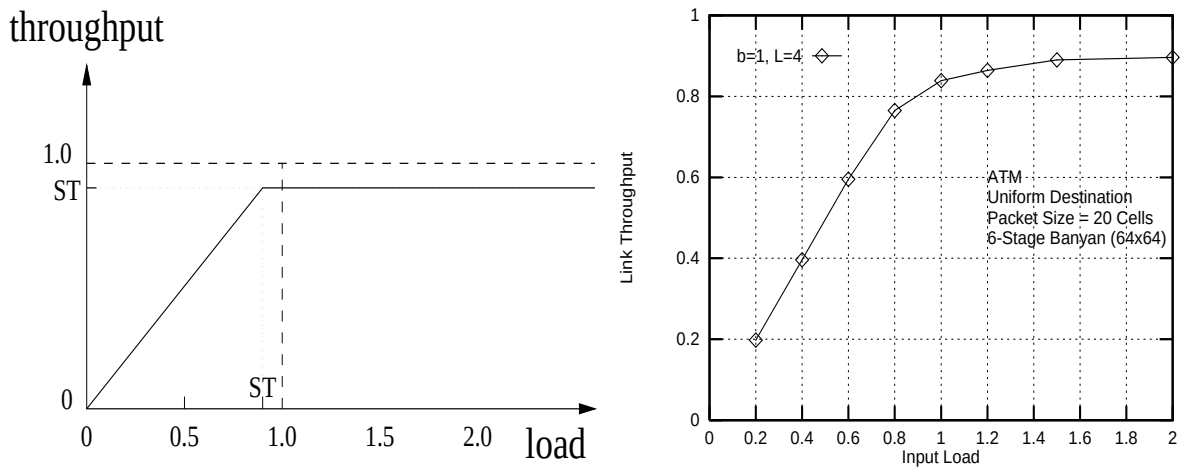
Ορισμένα προβλήματα που εμφανίστηκαν κατά καιρούς σχετικά με την χρήση της βιβλιοθήκης, ξεπεράστηκαν με επιτυχία. Ένα τέτοιο πρόβλημα ήταν ότι αρχικά, ο προσομοιωτής ήταν αδικαιολόγητα

αργός, ακόμα και στα μικρά παραδείγματα που δοκιμάστηκαν. Το πρόβλημα αυτό οφείλονταν στον τρόπο με τον οποίο γίνεται η διαχείριση της στοίβας των νημάτων από το CSIM. Όταν γίνεται αλλαγή του νήματος που εκτελείται, σώζεται η κατάσταση της στοίβας του νήματος που είχε τον έλεγχο και άρα όσο πιο μεγάλη στοίβα έχει ένα νήμα, τόσο πιο χρονοβόρα είναι η διαδικασία εναλλαγής νημάτων. Το πρόβλημα αυτό αντιμετωπίστηκε κάνοντας όσο το δυνατό πιο μικρό το πλήθος των τοπικών μεταβλητών κάθε νήματος, κάνοντας τις υπόλοιπες μεταβλητές εξωτερικές (global). Με αυτήν την αλλαγή, η ταχύτητα του προσομοιωτή περίπου πενταπλασιάστηκε. Ένα άλλο πρόβλημα που αντιμετωπίστηκε στην αρχή εκτέλεσης προσομοιώσεων είχε και αυτό σχέση με την ταχύτητά του και αφορούσε μία εντολή που φαίνονταν να είναι αδικαιολόγητα αργή στην εκτέλεσή της. Συγκεκριμένα, η εντολή timed_receive στην έκδοση της βιβλιοθήκης που δοκιμάσαμε αρχικά καθυστερούσε υπερβολικά την εκτέλεση του προγράμματος. Χρησιμοποιώντας εναλλακτικό τρόπο για την υλοποίηση του διαβάσματος από ένα γραμματοκιβώτιο περιμένοντας για μικρό χρονικό διάστημα (που είναι η λειτουργία της παραπάνω εντολής), ο χρόνος εκτέλεσης της προσομοίωσης υποτετραπλασιάστηκε.

Κεφάλαιο 4

Πειραματικά αποτελέσματα

Στα πειράματα που έγιναν, προσομοιώθηκε η συμπεριφορά ενός δικτύου Banyan butterfly με 64 εισόδους και εξόδους εκτός από την ομάδα των πειραμάτων που μελετάει διαφορετικού μεγέθους δίκτυα. Κάθε μεταγωγέας του δικτύου έχει 2 εισόδους και 2 εξόδους σε όλες τις ομάδες πειραμάτων εκτός αυτής που μελετάει την συμπεριφορά των δικτύων με διαφορετικού μεγέθους μεταγωγείς. Το μέγεθος κάθε πακέτου σε όλες τις ομάδες πειραμάτων εκτός από εκείνη που μελετάει την συμπεριφορά του δικτύου όταν το μέγεθος του πακέτου αλλάζει, είναι 20 cells/flits. Ο αριθμός αυτός επιλέχθηκε αρχικά για να είναι συγκρίσιμα τα αποτελέσματα αυτής της εργασίας με τα αποτελέσματα της εργασίας του W.J.Dally [Dally 90]. Ένα μέγεθος που μετράμε είναι η παροχή κορεσμού (saturation throughput). Το μέγεθος αυτό είναι η παροχή που έχει το δίκτυο όταν το φορτίο εισόδου είναι πολύ μεγάλο και η κίνηση που παράγεται δεν είναι δυνατό να μπει όλη μέσα στο δίκτυο αλλά είναι αναγκασμένη να περιμένει σε ουρές στις εισόδους του δικτύου. Το φαινόμενο που παρατηρήσαμε είναι ότι όταν το φορτίο εισόδου είναι 1.0 η παροχή του δικτύου δεν ήταν δυνατό να υπερβεί το 90% για ορισμένες διαμορφώσεις πειραμάτων. Στα ίδια πειράματα αυξάνοντας το φορτίο εισόδου πάνω από το 1.0 μετρήσαμε την παροχή του δικτύου να πλησιάζει το 100%. Αυτό οφείλεται στην τυχαιότητα με την οποία επιλέγονται τόσο ο προορισμός ενός πακέτου όσο και ο χρόνος μεταξύ διαδοχικών ενάρξεων μετάδοσης των πακέτων από τις πηγές. Στο σχήμα 4.1 φαίνονται η θεωρητική μεταβολή της παροχής καθώς και η μεταβολή που μετράμε με την προσομοίωση.



Σχήμα 4.1: Μεταβολή της παροχής θεωρητικά και σε προσομοίωση

Όταν ο χρόνος που δύο διαδοχικά πακέτα ξεκινούν την μετάδοσή τους, ακολουθεί μία εκθετική

κατανομή με μία μέση τιμή που εξαρτάται από το μέγεθος του πακέτου[1] μπορεί για τυχαίους λόγους να δημιουργούνται μεγάλα κενά μετάδοσης ακόμα και όταν το φορτίο είναι 1. Το φαινόμενο αυτό βέβαια τείνει να εξαφανίζεται όσο μεγαλώνει ο χρόνος προσομοίωσης αλλά για τον χρόνο που χρειάστηκε για να γίνουν οι προσομοιώσεις, οι επιδράσεις δεν είναι αμελητέες. Το φαινόμενο αυτό δεν επηρεάζει την αξιοπιστία των προσομοιώσεων αφού μερικά από τα αποτελέσματα για τυπικά παραδείγματα, ελέγχθηκαν αργότερα και με την χρήση διαστημάτων εμπιστοσύνης, και τα πειράματα που έγιναν, έδειξαν ότι με πιθανότητα 0.95 η μέση τιμή της ως προς μέτρηση μεταβλητής βρίσκεται το πολύ 1% μακρυά από την τιμή που υπολογίζει ο προσομοιωτής. Άλλο μέγεθος που μετράμε είναι η καθυστέρηση που υποβάλλεται ένα cell ή ένα πακέτο μέσα στο δίκτυο αλλά και εκτός αυτού, στις ουρές που δημιουργούνται στις εισόδους του δικτύου όταν το φορτίο είναι υψηλό. Ανάλογα με το πείραμα υπάρχουν 2 είδη καθυστερήσεων που μετράμε.

- Καθυστέρηση του δικτύου. Εδώ δεν μετράμε την καθυστέρηση στις ουρές εισόδου αλλά μόνο αυτήν που παρατηρείται μετά την είσοδο του cell ή του πακέτου μέσα στο δίκτυο.
- Συνολική καθυστέρηση. Εδώ μετράμε την καθυστέρηση από την στιγμή της γέννησης μέχρι την στιγμή της παράδοσης στον καταναλωτή. Εδώ συμπεριλαμβάνεται δηλαδή και ο χρόνος αναμονής στις ουρές εισόδου του δικτύου.

Στα γραφήματα που παρουσιάζονται παρακάτω παραθέτουμε την συνολική καθυστέρηση των cell/flit.

Το κριτήριο με το οποίο τερματίζεται η προσομοίωση είναι η άφιξη του πακέτου με αύξοντα αριθμό 13000 σε κάποιον από τους καταναλωτές (προορισμούς) ανεξάρτητα από το πόσες μονάδες προσομοιωμένου χρόνου χρειάζονται για να γίνει αυτό. Σε κάθε πείραμα, μέχρι να παραδοθούν 3000 πακέτα δεν συλλέγονται στατιστικά στοιχεία. Ο λόγος για τον οποίο περιμένουμε στην αρχή και κατόπιν αρχίζουμε να συλλέγουμε στατιστικά στοιχεία, είναι ότι περιμένουμε να παρέλθει η μεταβατική περίοδος του συστήματος γιατί κατά την διάρκεια αυτής, οι μετρήσεις δεν ανταποκρίνονται στην εικόνα που παρουσιάζει το σύστημα όταν φτάσει σε σταθερή κατάσταση. Ο αριθμός 3000 αποφασίστηκε με βάση τα πειραματικά αποτελέσματα. Η παρατήρηση που έγινε είναι ότι μετά από 2500 περίπου πακέτα, η μεταβολή στην μέση τιμή των μετρούμενων μεγεθών δεν αλλάζει πολύ και άρα μπορεί να γίνει η λογική υπόθεση ότι μετά από 3000 πακέτα έχει φτάσει το δίκτυο σε σταθερή κατάσταση. Για δίκτυα μεγαλύτερου μεγέθους, οι προσομοιώσεις που έγιναν είχαν κριτήριο τερματισμού την παραλαβή 26000 πακέτων και στην αρχή δεν μετρήθηκαν οι επιδόσεις των πρώτων 6000 πακέτων.

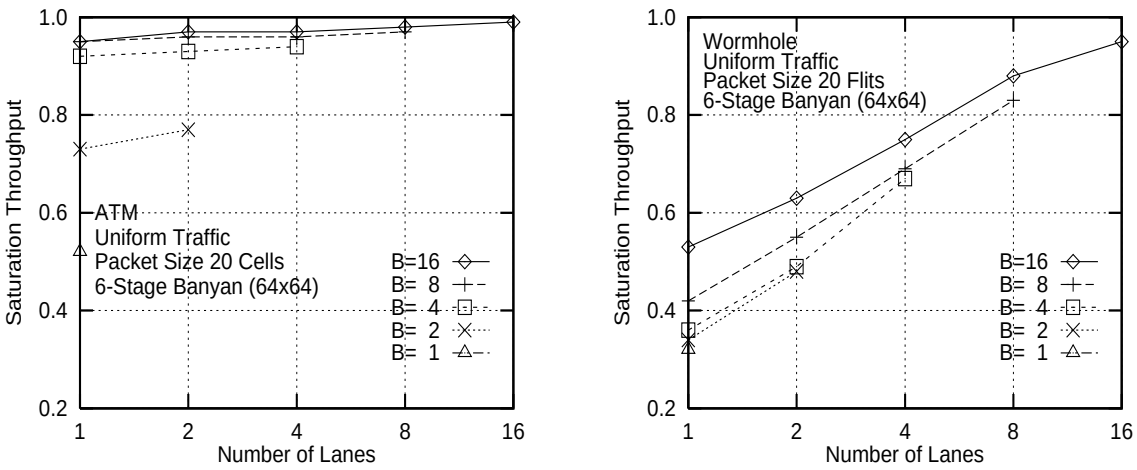
Στην συνέχεια της εργασίας, για να περιγράψουμε την διαμόρφωση του δικτύου, θα χρησιμοποιούμε τον ακόλουθο συμβολισμό:

- B : συμβολίζει το πλήθος των cell/flit που χωράνε μέσα στον χώρο αποθήκευσης που αφιερώνεται σε έναν σύνδεσμο.
- b : συμβολίζει το πλήθος των cell/flit που μπορούν να χωρέσουν στον χώρο που αφιερώνεται για μία λωρίδα.
- L : συμβολίζει το πλήθος των λωρίδων ανά σύνδεσμο.

Από τους παραπάνω ορισμούς των συμβόλων παρατηρούμε ότι ισχύει η ισότητα

$$B = b \cdot L$$

[1] Γνωρίζοντας το μέγεθος του πακέτου, γνωρίζουμε και τον χρόνο που θα χρειαστεί για να μεταδοθεί από τον παραγωγό. Βάσει του ορισμού του φορτίου του δικτύου που είναι $\frac{\lambda}{\mu}$ όπου λ είναι ο ρυθμός με τον οποίο εισέρχεται κίνηση μέσα στο δίκτυο και μ ο ρυθμός με τον οποίο το δίκτυο εξυπηρετεί(δεχεται) την κίνηση αυτή, αν αντιστρέψουμε το κλάσμα, βλέπουμε ότι ο λόγος των χρόνων που ένα πακέτο χρειάζεται για να παραληφθεί από το δίκτυο με τον χρόνο που το πακέτο χρειάζεται για να μεταδοθεί από τον παραγωγο, είναι ίσος με τον φόρτο του δικτύου.



Σχήμα 4.2: Παροχή κορεσμού με κίνηση Poisson

4.1 Μεταβολή στα μοντέλα κίνησης εισόδου

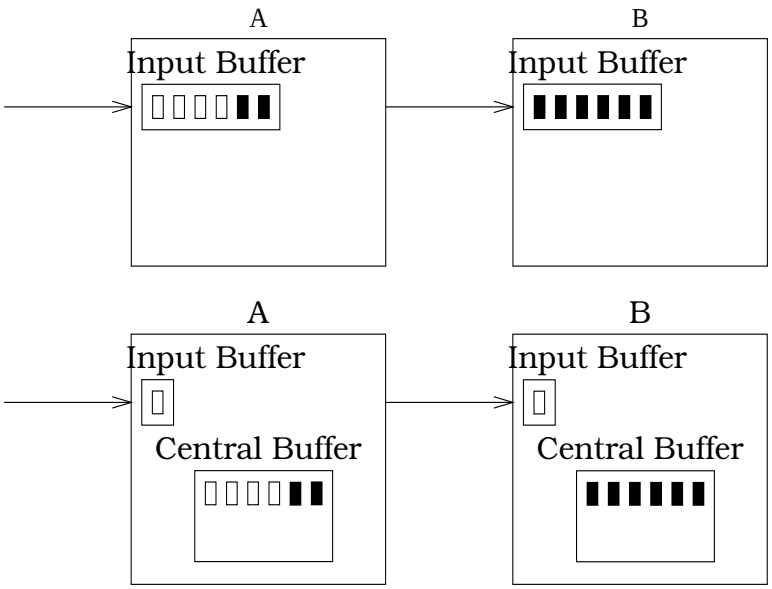
4.1.1 Κίνηση Poisson

Σε αυτήν την σειρά πειραμάτων μελετάμε την συμπεριφορά του δικτύου όταν η κίνηση εισόδου ακολουθεί κατανομή Poisson σε επίπεδο πακέτου. Το μέγεθος κάθε πακέτου είναι σταθερό και ίσο με 20 cell (για το ATM)/flit (για το wormhole). Το χαρακτηριστικό αυτής της μορφής κίνησης είναι ο εκθετικός χρόνος καθυστέρησης μεταξύ της γέννησης δύο πακέτων σε κάθε μία πηγή. Το δίκτυο, του οποίου η συμπεριφορά μελετήθηκε, είναι ένα δίκτυο Banyan με 64 εισόδους και εξόδους το οποίο αποτελείται από μεταγωγείς 2×2.

Στα γραφήματα που παραθέτουμε σε αυτό το κεφάλαιο φαίνεται η παροχή κορεσμού (saturation throughput) για μία σειρά από διαμορφώσεις (configurations) στις οποίες αλλάζει το μέγεθος του χώρου αποθήκευσης (buffer) αλλά και το πλήθος των λωρίδων ανά σύνδεσμο. Ακόμα φαίνεται και η μέση συνολική καθυστέρηση κάθε cell/flit όταν το φορτίο εισόδου είναι ελαφρύ (φορτίο 0.2, 0.4) αλλά και όταν το φορτίο είναι βαρύ (0.8). Η επιλογή 2 περιπτώσεων που το φορτίο είναι ελαφρύ έγινε γιατί στην περίπτωση του wormhole, για μερικές διαμορφώσεις πειραμάτων, το φορτίο 0.4 είναι είτε πάνω από είτε πολύ κοντά στην παροχή κορεσμού του δικτύου και άρα δεν μπορεί να θεωρηθεί ελαφρύ. Ακόμα το μέγεθος του χώρου αποθήκευσης κυμαίνεται από 1 έως 16 (μονάδες μέτρησης σε πλήθος cell/flit που μπορεί να αποθηκεύσει) για όλες τις δυνάμεις του 2 μέσα σε αυτά τα όρια.

Στα γραφήματα του σχήματος 4.2 φαίνεται ότι το ATM έχει καλύτερη παροχή από το wormhole για όλες τις διαμορφώσεις. Επιπλέον φαίνεται ότι ακόμα και για μικρά μεγέθη χώρου αποθήκευσης (π.χ. 4) και ακόμα και με μικρό πλήθος λωρίδων (π.χ. 1) επιτυγχάνεται πολύ καλή παροχή κορεσμού (στο παράδειγμα όπου $B = 4, L = 1$ η παροχή κορεσμού είναι 0.92). Γι αυτόν τον λόγο η προσθήκη λωρίδων δεν βελτιώνει σε αυτήν την περίπτωση τις (πολύ καλές) επιδόσεις του δικτύου. Αντίθετα, στην περίπτωση του wormhole η παροχή κορεσμού, για φτωχά σε πόρους δίκτυα, δεν είναι καλή. Αυτό βελτιώνεται με την προσθήκη λωρίδων. Για να επιτευχθεί όμως παραπλήσια παροχή με αυτή του ATM, χρειάζονται πολύ περισσότεροι πόροι. Παροχή κορεσμού 0.92 για παράδειγμα που επιτυγχάνεται από ένα δίκτυο ATM με $B = 4, L = 1$, για να επιτευχθεί από το wormhole χρειάζεται να είναι $B = 16, L = 16$.

Στο γράφημα για την περίπτωση του wormhole περιέχονται και τιμές που περιλαμβάνονται και στην αναφορά [Dally 90]. Συγκρίνοντας αυτές τις τιμές με τις αντίστοιχες του Dally παρατηρούμε ότι εμείς τις βρίσκουμε υψηλότερες. Αυτό οφείλεται στο ότι στην εργασία [Dally 90], σε κάθε είσοδο μεταγωγέα

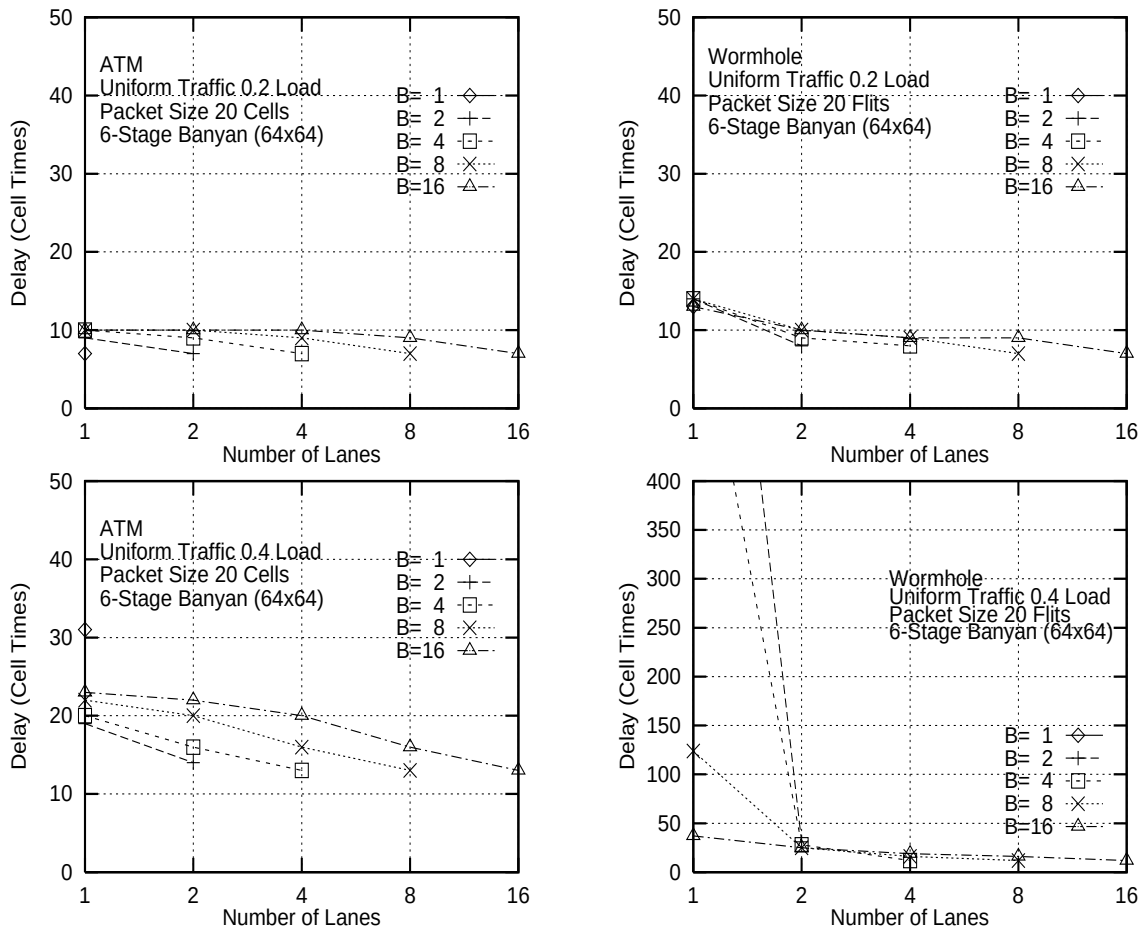


Σχήμα 4.3: Input vs Central Buffering

υπάρχει μία ουρά και κάθε flit περνά από αυτήν πριν διέλθει από τον μεταγωγέα. Χρησιμοποιείται δηλαδή input buffering. Στο σχήμα 4.3 φαίνεται ένα πακέτο το οποίο έχει μπλοκαριστεί στον μεταγωγέα B αλλά επειδή δεν χωράει ολόκληρο σε αυτόν, μερικά από τα flit του πακέτου βρίσκονται στον μεταγωγέα A. Στην υλοποίηση που είχε γίνει στην εργασία [Dally 90], τα flit του πακέτου που βρίσκονται στον μεταγωγέα A μπλοκάρουν και οποιοδήποτε άλλο πακέτο θέλει να χρησιμοποιήσει την λωρίδα αυτή. Στον προσομοιωτή που υλοποιήσαμε δεν χρησιμοποιείται input buffering αλλά central buffering. Cells ενός πακέτου δεν καταλαμβάνουν μνήμη στην είσοδο του μεταγωγέα αλλά αποσύρονται σε μία κεντρική μνήμη και απελευθερώνουν την μνήμη εισόδου. Έτσι επιτρέπεται σε ένα πακέτο που θέλει να χρησιμοποιήσει τις ελεύθερες θέσεις μνήμης αυτής της λωρίδας για να προωθηθεί στην επιθυμητή έξοδο, και να προσπεράσει το μπλοκαρισμένο πακέτο εφόσον μπορεί να μεταδοθεί.

Στα γραφήματα του σχήματος 4.4 παριστάνεται η μέση συνολική καθυστέρηση στην οποία υποβάλλονται τα cells σε συνάρτηση με το πλήθος των λωρίδων και για διάφορες διαμορφώσεις δικτύου. Εδώ βλέπουμε ότι εφόσον είμαστε μακρυά από την παροχή κορεσμού στο wormhole, η καθυστέρηση είναι στα ίδια επίπεδα συγκρινόμενη με το ATM. Η παροχή κορεσμού όμως είναι πολύ καλύτερη στην περίπτωση του ATM και άρα επιτυγχάνει μικρές καθυστερήσεις ακόμα και στην περίπτωση της μιας λωρίδας. Παρατηρούμε εδώ ένα φαινόμενο που δεν ήταν αναμενόμενο: η καθυστέρηση φαίνεται να αυξάνεται καθώς το μέγεθος του χώρου αποθήκευσης μεγαλώνει κρατώντας σταθερό το πλήθος των λωρίδων. Αυτό συμβαίνει γιατί αυξάνοντας το μέγεθος του χώρου αποθήκευσης, αυξάνεται και το πλήθος των cell/flit που μπαίνουν μέσα στο δίκτυο. Κρατώντας όμως σταθερό το πλήθος των λωρίδων παραμένει σταθερή η δυνατότητα του δικτύου να μεταφέρει παράλληλα αυτήν την κίνηση. Περισσότερη κίνηση μπαίνει μέσα στο δίκτυο αλλά παραμένει εκεί για περισσότερο χρόνο.

Στο σχήμα 4.5 προβάλλονται και τα γραφήματα της καθυστέρησης όταν το φορτίο εισόδου είναι βαρύ. Σε αυτήν την περίπτωση ενδιαφέρον παρουσιάζει μόνο το ATM αφού το wormhole συνήθως έχει παροχή κορεσμού μικρότερη από 0.8 και η καθυστέρηση μεγαλώνει συνεχώς. Εδώ δεν παρουσιάζεται πάντα το φαινόμενο που παρατηρήσαμε προηγουμένως. Καθώς πλησιάζουμε στο φορτίο που προκαλεί κορεσμό στο δίκτυο, αρχίζει η καθυστέρηση στις ουρές εισόδου να κυριαρχεί (να είναι δηλαδή μεγαλύτερη) της καθυστέρησης μέσα στο δίκτυο. Αν παρατηρήσουμε προσεκτικά το γράφημα για το ATM, βλέπουμε να εμφανίζεται στην περίπτωση των 8 λωρίδων το φαινόμενο της αύξησης της καθυστέρησης καθώς



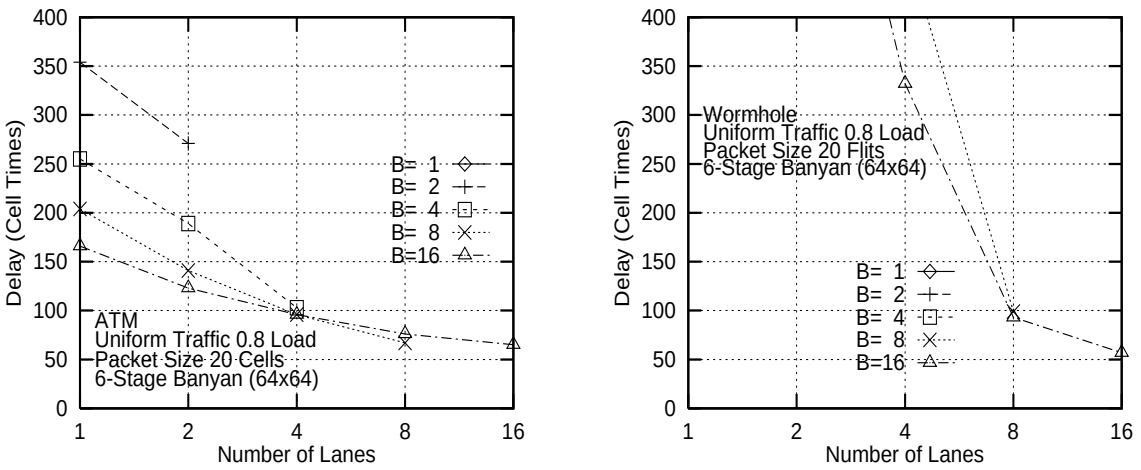
Σχήμα 4.4: Σύγκριση υπό συνθήκες κίνησης Poisson χαμηλού φορτίου

αυξάνεται το μέγεθος του χώρου αποθήκευσης. Όταν οι λωρίδες είναι 4, το φαινόμενο εξαλείφεται, ενώ όταν το πλήθος των λωρίδων μικρύνει ακόμα περισσότερο, η καθυστέρηση στις ουρές εισόδου του δικτύου κυριαρχεί και το φαινόμενο αντιστρέφεται. Σε αυτήν την περίπτωση βλέπουμε ότι η προσθήκη λωρίδων στο ATM μειώνει την καθυστέρηση στην οποία υποβάλλονται τα cells.

Σε γενικές γραμμές μπορούμε να παρατηρήσουμε ότι οι καθυστερήσεις των cell/flit είναι στα ίδια επίπεδα στις περιπτώσεις που το wormhole είναι μακρυά από το σημείο κορεσμού αλλά επειδή το ATM επιτυγχάνει καλύτερη παροχή, επιτυγχάνει πολύ καλύτερες επιδόσεις σε υψηλότερα φορτία. Ακόμα παρατηρούμε ότι η προσθήκη λωρίδων βελτιώνει σημαντικά τις επιδόσεις και του wormhole αλλά και του ATM σε μικρότερο βαθμό, αφού η καθυστέρηση στην οποία υποβάλλονται τα cell είναι σημαντικά μικρότερη όταν αυξάνονται οι λωρίδες και πλησιάζουμε στο φορτίο κορεσμού.

4.1.2 Εκρηκτική κίνηση

Οι επιστημονικές εφαρμογές που χρησιμοποιούν παράλληλους υπολογιστές πολυ συχνά δημιουργούν εκρηκτική κίνηση πάνω στο δίκτυο διασύνδεσης. Μεγάλες ποσότητες δεδομένων πρέπει για παράδειγμα να μεταφερθούν μεταξύ των κόμβων που μετέχουν σε έναν υπολογισμό [DUSH 93]. Καθώς η ταχύτητα με την οποία οι πηγές εισάγουν κίνηση μέσα στο δίκτυο αυξάνει συνεχώς, πρέπει να περιμένουμε ότι μεγαλύτερο εύρος ζώνης (bandwidth) θα χρησιμοποιείται για μικρότερο χρονικό διάστημα [Kung 92]. Άλλο παράδειγμα είναι εξυπηρετητές αρχείων (file servers) που χρειάζεται να μεταδόσουν μεγάλα αρχεία σε ένα κατανεμημένο σύστημα.



Σχήμα 4.5: Συγκριση υπό υψηλού φορτίου κίνηση Poisson

Σαν εκρηκτική μπορεί να χαρακτηριστεί κίνηση της οποίας το μοντέλο έχει τα ακόλουθα χαρακτηριστικά

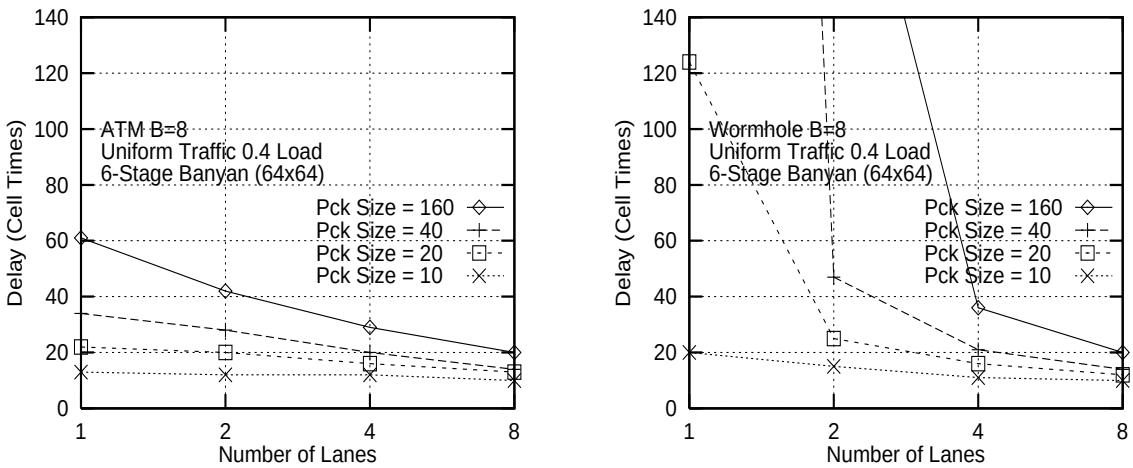
- Εφόσον μία πηγή δεν μεταδίδει, τότε με μεγάλη πιθανότητα δεν θα μεταδόσει στο σύντομο μέλλον και με μικρή πιθανότητα θα μεταδόσει.
- Εφόσον μία πηγή μεταδίδει δεδομένα, τότε με μεγάλη πιθανότητα στο σύντομο μέλλον θα συνεχίσει να μεταδίδει δεδομένα και με μικρή πιθανότητα θα σταματήσει να μεταδίδει.

Με βάση τα παραπάνω χαρακτηριστικά η κίνηση που μπαίνει μέσα στο δίκτυο όταν ένα πακέτο φτάνει στην είσοδο του δικτύου και χωρίζεται σε cell/flit, μπορεί να θεωρηθεί εκρηκτική σε επίπεδο cell/flit.

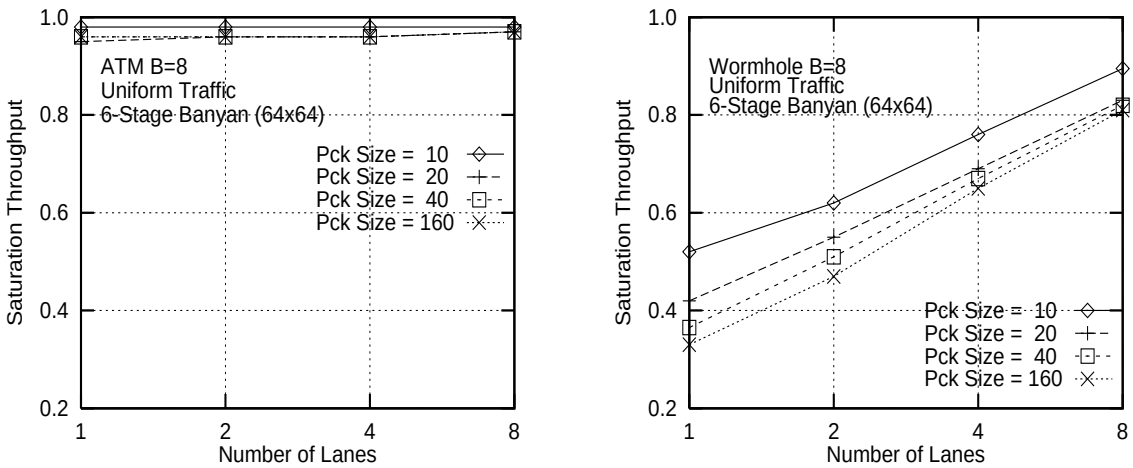
Μελετήσαμε πως επηρεάζονται οι επιδόσεις ενός δικτύου καθώς μεταβάλλεται το μέγεθος των πακέτων που κυκλοφορούν μέσα σε αυτό. Σαν τυπικό δίκτυο επιλέχθηκε ένα Banyan με 64 εισόδους και εξόδους, αποτελούμενο από μεταγωγείς 2×2. Το μέγεθος του χώρου αποθήκευσης είναι $B = 8$ και οι προορισμοί των πακέτων επιλέγονται ομοιόμορφα. Ο χρόνος που μεσολαβεί μεταξύ της γέννησης δύο πακέτων είναι τυχαίος και ακολουθεί την εκθετική κατανομή. Το πλήθος των λωρίδων που έχει το δίκτυο μεταβάλλεται από 1 σε 8 (υποτίθενται οι δυνάμεις του 2) και τα μεγέθη των πακέτων που μελετούνται είναι 10, 20, 40, 160 cell/flit. Στις παραπάνω περιπτώσεις μετρούνται η συνολική καθυστέρηση στην οποία υπαβάλλεται ένα cell/flit καθώς και η παροχή κορεσμού για κάθε μία από τις προαναφερθείσες διαμορφώσεις των δικτύων. Η καθυστέρηση μετράται όταν το φορτίο εισόδου είναι 0.4.

Στο σχήμα 4.6 φαίνονται οι καθυστερήσεις των cell/flit σε συνάρτηση με το πλήθος των λωρίδων, για κάθε μέγεθος πακέτου. Παρατηρούμε ότι όσο μεγαλώνει το μέγεθος του πακέτου, τόσο πιο πολύ επηρεάζεται η καθυστέρηση των cell/flit για μικρό πλήθος λωρίδων. Η επήρεια μειώνεται καθώς μεγαλώνει το πλήθος των λωρίδων. Παρότι στο ATM η καθυστέρηση παραμένει πάντα περιορισμένη, στο wormhole μεγαλώνει πολύ ή και απεριόριστα στις περιπτώσεις όπου η παροχή κορεσμού είναι μικρότερη από το φορτίο εισόδου σε ορισμένες διαμορφώσεις.

Όπως φαίνεται στο σχήμα 4.7, παρατηρούμε ότι και πάλι η παροχή κορεσμού για το ATM είναι πολύ υψηλή σε όλες τις περιπτώσεις, φαινόμενο που δεν παρατηρείται στο wormhole. Στο wormhole φαίνεται ότι οι επιδόσεις πέφτουν, όταν μεγαλώνει το μέγεθος του πακέτου, με μεγαλύτερη την επιβάρυνση όταν το πλήθος των λωρίδων είναι μικρότερο. Μεγάλα πακέτα μπαίνουν στο δίκτυο τα οποία όταν μπλοκαριστούν απλώνονται σε όλο το μήκος του δικτύου και καταλαμβάνουν πόρους σε πολλούς μεταγωγείς μέσα σε αυτό. Το φαινόμενο αυτό μελετήθηκε και πιο ειδικά με μερικά πειράματα για $B = 8, L = 1$ όταν το πακέτο έχει μέγεθος 6, 7, 8, 9, 12, 15, 16, 17 και 20. Όπως φαίνεται



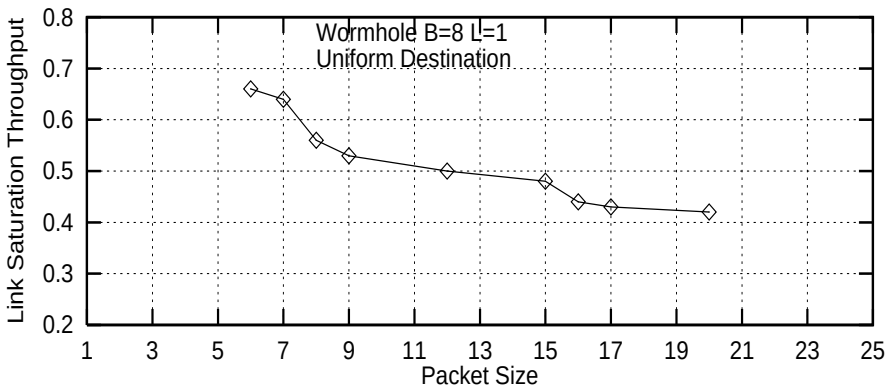
Σχήμα 4.6: Σύγκριση καθυστερήσεων όταν αλλάζει το μέγεθος πακέτου



Σχήμα 4.7: Παροχή κορεσμού για διαφορετικά μεγέθη πακέτων

και στο σχήμα 4.8 όταν το μέγεθος του πακέτου είναι μικρότερο από 8, η παροχή του δικτύου είναι μεγαλύτερη από 0.6. Οταν μπλοκάρεται ένα πακέτο, δεν αρκεί για να γεμίσει τον χώρο αποθήκευσης ενός μεταγωγέα και αφήνεται τουλάχιστον μία θέση κενή για κάποιο άλλο πακέτο να διέλθει από τον μεταγωγέα, προσπερνώντας το μπλοκαρισμένο πακέτο όπως περιγράφηκε στο 4.1.1. Όταν το μέγεθος του πακέτου γίνει ίσο με το μέγεθος του χώρου αποθήκευσης, το φαινόμενο αυτό δεν συμβαίνει και η επίπτωση είναι η απότομη πτώση της παροχής από 0.64, όταν το μέγεθος πακέτου είναι 7, στο 0.56 όταν το μέγεθος πακέτου γίνει 8. Ανάλογο φαινόμενο παρουσιάζεται και στην μετάβαση του μεγέθους πακέτου από 15 σε 16, οπότε όταν μπλοκαριστεί ένα πακέτο, τότε καταλαμβάνει πλήρως τον χώρο αποθήκευσης 2 μεταγωγέων.

Συνοπτικά, είναι εμφανές ότι το ATM επιτυγχάνει και εδώ σημαντικά βελτιωμένες επιδόσεις από το wormhole. Φαίνεται επίσης ότι η προσθήκη λωρίδων βελτιώνει σημαντικά τόσο τις επιδόσεις του wormhole, όπως είδαμε και στο 4.1.1, αλλά και για το ATM την καθυστέρηση στην οποία υποβάλλονται τα cells όταν το μέγεθος του πακέτου είναι μεγάλο.



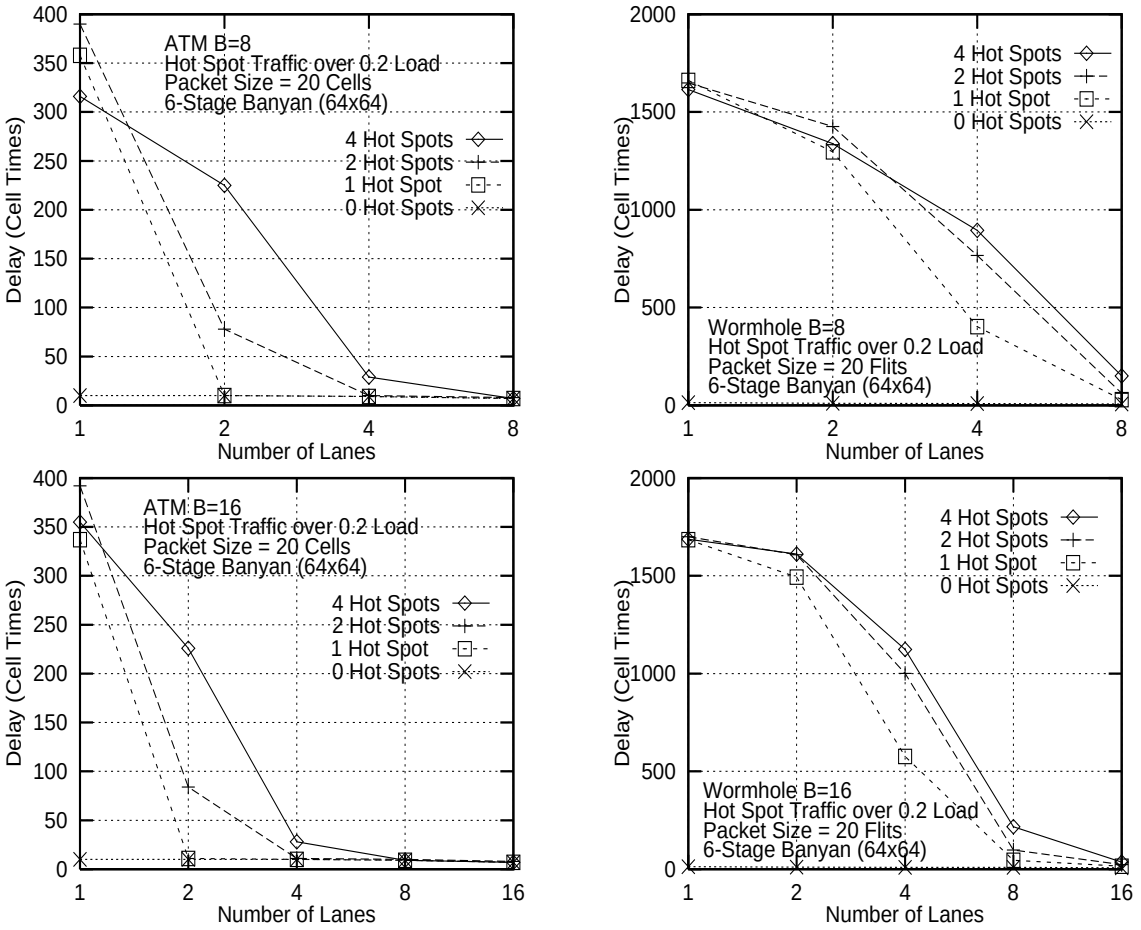
Σχήμα 4.8: Παροχή κορεσμού wormhole με μία λωρίδα

4.1.3 Κίνηση με θερμά σημεία

Η κίνηση που γεννάται από παράλληλες εφαρμογές, εκτός από το ότι συχνά είναι εκρηκτική, συχνά δεν έχει και την ιδιότητα της ομοιομορφίας στην επιλογή προορισμού. Ορισμένοι προορισμοί μπορεί να επιλέγονται πιο συχνά από άλλους [Kung 92]. Παραδείγματα αυτού του είδους της κίνησης μπορεί να παράγονται προς έναν εξυπηρετητή (server) μέσα σε ένα κατανεμημένο σύστημα ή από πράξεις συγχρονισμού μίας παράλληλης εφαρμογής. Η εικόνα που έχουμε λοιπόν από τα προηγούμενα πειράματα, δεν είναι πλήρης. Είναι ενδιαφέρον να δούμε το πώς επηρεάζει τις επιδόσεις ενός δικτύου η ύπαρξη ενός ή και περισσοτέρων *θερμών σημείων (hot spots)* στις εξόδους του δικτύου δηλαδή προορισμών που έχουν συνεχώς φορτίο. Σε αυτήν την περίπτωση είναι ενδιαφέρον να μελετήσουμε τις επιπτώσεις που έχει η ύπαρξη θερμών σημείων στην καθυστέρηση της κίνησης που δεν πηγαίνει σε θερμό σημείο. Η κίνηση που πηγαίνει σε κάποιο θερμό σημείο υποχρεωτικά θα υποβληθεί σε μεγάλες καθυστερήσεις. Σε ένα δίκτυο που είναι δίκαιο, η κίνηση που δεν πηγαίνει σε θερμό σημείο, δεν πρέπει να επηρεάζεται από την κίνηση που πηγαίνει σε κάποιο θερμό σημείο.

Στα πειράματα που ακολουθούν, γίνεται μία επιλογή από τους πιθανούς προορισμούς για το ποιοι θα είναι θερμά σημεία. Όταν τα θερμά σημεία είναι περισσότερα του ενός, τα τοποθετούμε όσο πιο μακρυά μεταξύ τους είναι δυνατό. Για κάθε ένα από τα θερμά σημεία παράγεται κίνηση φορτίου 1 από κάθε πηγή. Για όλους τους υπόλοιπους προορισμούς παράγεται κίνηση φορτίου 0.2 (ελαφρά κίνηση). Η κίνηση αυτή διαμοιράζεται ομοιόμορφα ανάμεσα στους υπόλοιπους προορισμούς, αυτούς δηλαδή που δεν είναι θερμά σημεία. Το δίκτυο που μελετήσαμε είναι Banyan με 64 εισόδους εισόδους και εξόδους. Το μέγεθος του πακέτου είναι 20 cell/flit. Μελετήσαμε την συμπεριφορά των δικτύων με μέγεθος χώρου αποθήκευσης 8 και 16 καθώς το πλήθος των λωρίδων παίρνει όλες τις δυνατές τιμές που είναι δυνάμεις του 2. Το πλήθος των θερμών σημείων που μελετήσαμε είναι 1, 2, 4. Στα γραφήματα που ακολουθούν υπάρχουν και οι καθυστερήσεις όταν δεν υπάρχουν θερμά σημεία για το ίδιο φορτίο εισόδου (0.2).

Στο σχήμα 4.9 φαίνονται τα γραφήματα των καθυστερήσεων σε συνάρτηση με το πλήθος των λωρίδων ανά σύνδεσμο. Συγκρίνοντας τα αντίστοιχα γραφήματα μεταξύ ATM και wormhole παρατηρούμε ότι η ύπαρξη θερμών σημείων έχει πολύ μεγαλύτερη επίπτωση στην καθυστέρηση των flit που δεν πηγαίνουν σε θερμό σημείο στην περίπτωση του wormhole, από την αντίστοιχη καθυστέρηση των cell στην περίπτωση του ATM. Ενώ η πρώτη είναι περίπου 1600 χρόνοι cell, η δεύτερη είναι στην περιοχή των 350 χρόνων cell για την περίπτωση της μίας λωρίδας. Καθώς το πλήθος των λωρίδων αυξάνεται, οι καθυστερήσεις μειώνονται αλλά και πάλι το ATM έχει καλύτερη συμπεριφορά. Με μικρό πλήθος λωρίδων, και συγκεκριμένα λίγο μεγαλύτερο από το πλήθος των θερμών σημείων, η κίνηση που πηγαίνει σε θερμό σημείο δεν επηρεάζει καθόλου την υπόλοιπη κίνηση. Η καθυστέρηση της κίνησης που δεν πηγαίνει σε θερμό σημείο πέφτει στα επίπεδα της καθυστέρησης στην περίπτωση που δεν

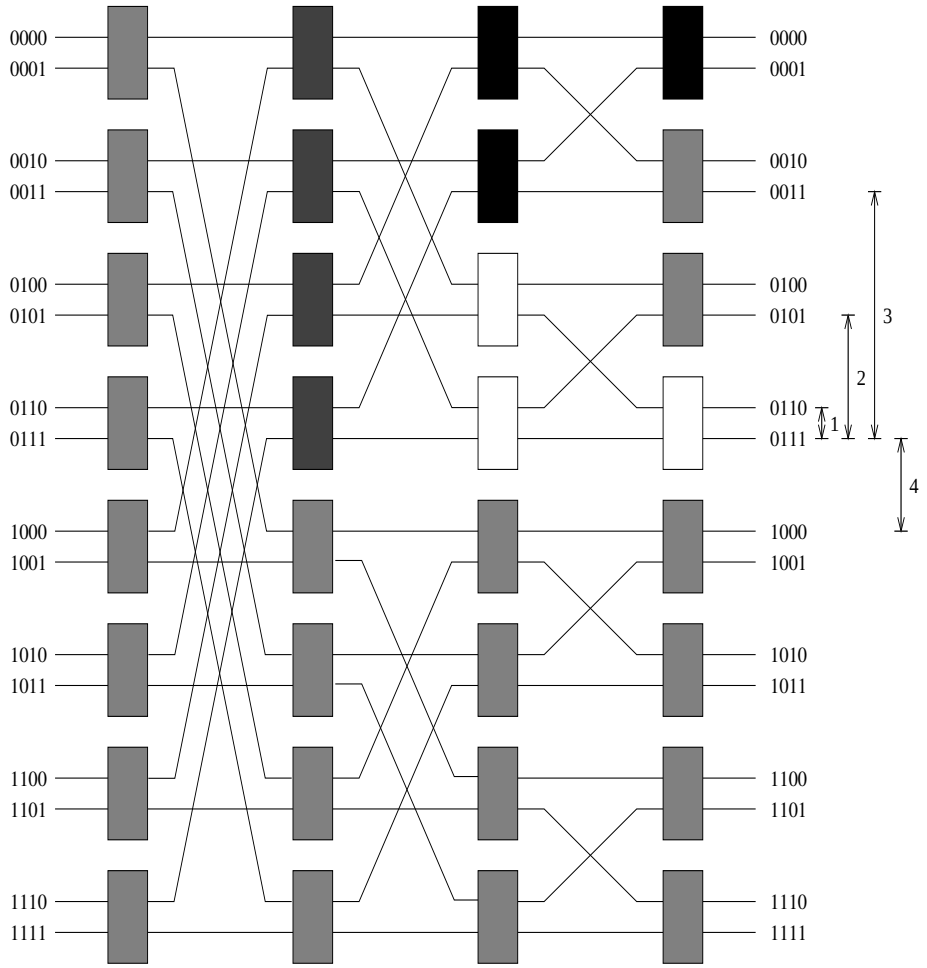


Σχήμα 4.9: Καθυστέρηση για κίνηση με διαφορετικά πλήθη θερμών σημείων

υπάρχουν θερμά σημεία.

Στην περίπτωση του wormhole αντίθετα, ακόμα και στην περίπτωση του ενός θερμού σημείου, δεν αρκούν 4 λωρίδες για να εξαλείψουν τις συνέπειες που έχει στην καθυστέρηση της κίνησης που δεν πηγαίνει σε θερμό σημείο η ύπαρξη αυτού του θερμού σημείου. Αυτό οφείλεται στον τρόπο με τον οποίο κυκλοφορεί η κίνηση μέσα σε ένα δίκτυο wormhole και συγκεκριμένα στο γεγονός ότι δεν συγχωνεύονται τα πακέτα που πηγαίνουν στον ίδιο προορισμό σε μία ομάδα ροής όπως γίνεται στην περίπτωση του ATM που μελετάμε. Η κίνηση προς το θερμό σημείο αφήνεται χωρίς έλεγχο να γεμίσει τον χώρο αποθήκευσης των μεταγωγέων άσκοπα. Δεν επιτρέπεται έτσι στην κίνηση που δεν πηγαίνει σε θερμό σημείο να προχωρήσει χωρίς πρόβλημα και να προσπεράσει την μποτιλιαρισμένη κίνηση προς το θερμό σημείο. Αυτό δεν συμβαίνει στην περίπτωση του ATM όπου η κίνηση που πηγαίνει στον ίδιο προορισμό, συγχωνεύεται σε ομάδες ροής και καταναλώνει μόνο ένα μέρος από τον χώρο αποθήκευσης κάθε μεταγωγέα. Εφόσον το πλήθος των λωρίδων είναι ελαφρά μεγαλύτερο από το πλήθος των θερμών σημείων, πάντα υπάρχει λωρίδα που δεν είναι κατειλημένη από κίνηση που πηγαίνει σε θερμό σημείο. Η κίνηση που δεν πηγαίνει σε θερμό σημείο, μπορεί να κινηθεί ανεπηρέαστη.

Ενδιαφέρον είναι να δούμε αν η κίνηση που προορίζεται "κοντά" σε ένα θερμό σημείο έχει μεγαλύτερες επιπτώσεις στην καθυστέρηση στην οποία υποβάλλεται συγκριτικά με την κίνηση που προορίζεται "μακρύτερα" από αυτό. Γι αυτόν τον λόγο χρειάζεται να ορίσουμε μία έννοια απόστασης δύο προορισμών ενός δικτύου Banyan. Ορίζουμε λοιπόν ως απόσταση δύο εξόδων ενός δικτύου Banyan ως τον μέγιστο αριθμό επιπέδων του δικτύου όπου τα υποδέντρα που οδηγούν σε κάθε μία έξοδο

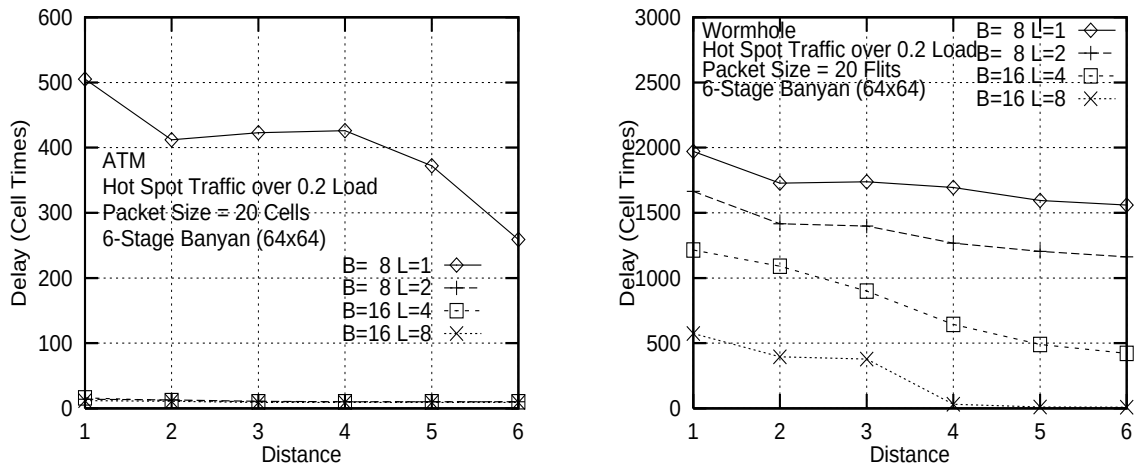


Σχήμα 4.10: Απόστάσεις προορισμών σε ένα δίκτυο Banyan 16×16

είναι ξένα μεταξύ τους. Στο σχήμα 4.10 φαίνεται ένα δίκτυο με 16 εισόδους και εξόδους. Στην δεξιά άκρη του σχήματος φαίνονται οι αποστάσεις διαφόρων προορισμών από τον προορισμό 0111 (δυαδικό). Παρατηρούμε λοιπόν ότι ο προορισμός 0110 βρίσκεται σε απόσταση 1. Το υποδέντρο που συνδέει τον προορισμό 0111 με τις πηγές και αυτό που συνδέει τον 0110 με τις πηγές διαφέρουν μόνο κατά ένα επίπεδο, αυτό που ενώνει τον τελευταίο μεταγωγέα με καθέναν από τους προορισμούς. Κατά τον ίδιο τρόπο σε απόσταση 2 βρίσκονται οι προορισμοί 0100 και 0101, γιατί τα υποδέντρα που τους συνδέουν με τις πηγές διαφέρουν σε δύο επίπεδα από αυτό που συνδέει το 0111 με τις πηγές. Γενικά σε απόσταση κ από ένα προορισμό βρίσκονται $2^{\kappa-1}$ προορισμοί.

Στο σχήμα 4.11 παρουσιάζονται τα γραφήματα της καθυστέρησης της κίνησης σε συνάρτηση με την απόσταση του προορισμού της από το μοναδικό θερμό σημείο. Στην περίπτωση του ATM, βλέπουμε ότι γενικά δεν επιδρά ο παράγοντας απόσταση από το θερμό σημείο στην καθυστέρηση της κίνησης. Αν εξαιρέσουμε την περίπτωση της μίας λωρίδας, στις υπόλοιπες περιπτώσεις και η κίνηση υποβάλλεται σε παρόμοιες καθυστερήσεις ανεξάρτητα της απόστασης από το θερμό σημείο. Ένας προορισμός λοιπόν που βρίσκεται κοντά στο θερμό σημείο δεν υποφέρει από μεγάλη καθυστέρηση εξ' αιτίας κίνησης που δεν τον αφορά. Στην περίπτωση της μίας λωρίδας, η κίνηση που προορίζεται σε προορισμούς σε απόσταση 6 από το θερμό σημείο είναι η μισή περίπου από αυτήν προς τον προορισμό σε απόσταση 1.

Στην περίπτωση του wormhole υπάρχουν περιπτώσεις που η διαφορά είναι σημαντική. Ας προσέξουμε την περίπτωση $B = 16, L = 8$. Η μέση καθυστέρηση που παρατηρήσαμε στο αντίστοιχο



Σχήμα 4.11: Καθυστέρηση σε συνάρτηση της απόστασης από το θερμό σημείο

γράφημα του σχήματος 4.9 είναι 45 χρόνοι cell. Όμως σε αυτήν την περίπτωση η κίνηση που δέχεται ένας προορισμός σε απόσταση 1, έχει υποβληθεί κατά μέσο όρο σε καθυστέρηση 574 χρόνων cell. Τον μέσο όρο τον κρατάνε χαμηλά οι (πολλοί) προορισμοί που βρίσκονται σε απόσταση μεγαλύτερη του 3. Παρόλο δηλαδή που το δίκτυο εμφανίζει μια καλή μέση εικόνα, η κίνηση που προορίζεται σε ορισμένους "άτυχους" προορισμούς υποβάλλεται σε πολύ μεγάλες καθυστερήσεις. Αυτό αποτελεί στοιχείο αδικίας στην μεταχείριση της κίνησης μέσα στο δίκτυο.

Συνοπτικά, βλέπουμε ότι το ATM με εισιτήρια συμπεριφέρεται πολύ καλύτερα κάτω από συνθήκες κίνησης μέ θερμά σημεία και επιτυγχάνει πολύ καλύτερες καθυστερήσεις. Δεν είναι άδικο ως προς τον τρόπο που μεταχειρίζεται την κίνηση που κινείται μέσα στο δίκτυο. Αρκούν ελάχιστα περισσότερες λωρίδες από το πλήθος των θερμών σημείων ώστε η κίνηση που δεν πηγαίνει σε θερμό σημείο να μην επηρεάζει την κίνηση που δεν πηγαίνει. Το wormhole από την άλλη μεριά δεν έχει αναλόγως καλές ιδιότητες. Ακόμα και όταν επιτυγχάνει καλή μέση τιμή στην καθυστέρηση που υποβάλλεται η κίνηση προς μη θερμό σημείο, αυτή φαίνεται να εξαρτάται από την θέση του προορισμού και συγκεκριμένα από την απόστασή του από το θερμό σημείο. Όπως είδαμε και προηγουμένως, η κίνηση που προορίζεται κοντά στο θερμό σημείο μπορεί να υποβληθεί σε ακόμα και 60 φορές μεγαλύτερη καθυστέρηση από αυτήν που κατευθύνεται μακρυά από αυτό (απόσταση 6).

4.2 Μεταβολή στις παραμέτρους του δικτύου

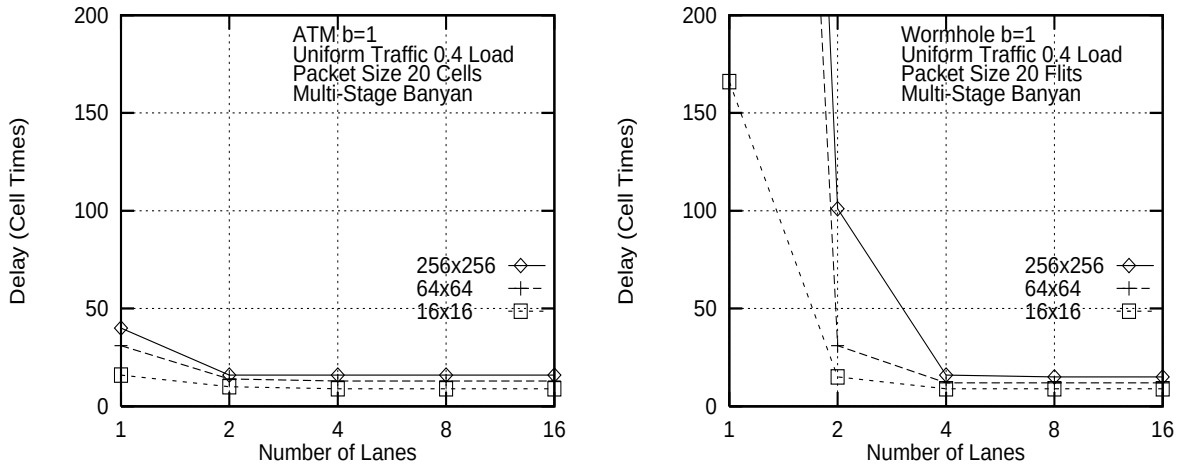
Κατά τα προηγούμενα πειράματα οι παράμετροι του δικτύου όπως το μέγεθός του και το πλήθος εισόδων και εξόδων που έχει κάθε μεταγωγέας από τους οποίους αποτελείται το δίκτυο, περέμειναν αμετάβλητα. Στις σειρές των πειραμάτων που θα ακολουθήσουν, μελετάται η επήρεια της αύξησης των εισόδων και των εξόδων τόσο του δικτύου όσο και των μεταγωγέων που το αποτελούν στις επιδόσεις ενός δικτύου Banyan.

4.2.1 Μεταβολή μεγέθους δικτύου

Το πλήθος των κόμβων που θέλουμε να συνδέσουμε με ένα δίκτυο δεν είναι βέβαια σταθερό για όλες τις ενδιαφέρουσες εφαρμογές. Έχει μεγάλο ενδιαφέρον λοιπόν να μελετήσουμε τις επιδόσεις των δικτύων καθώς μεγαλώνει το μέγεθός τους.

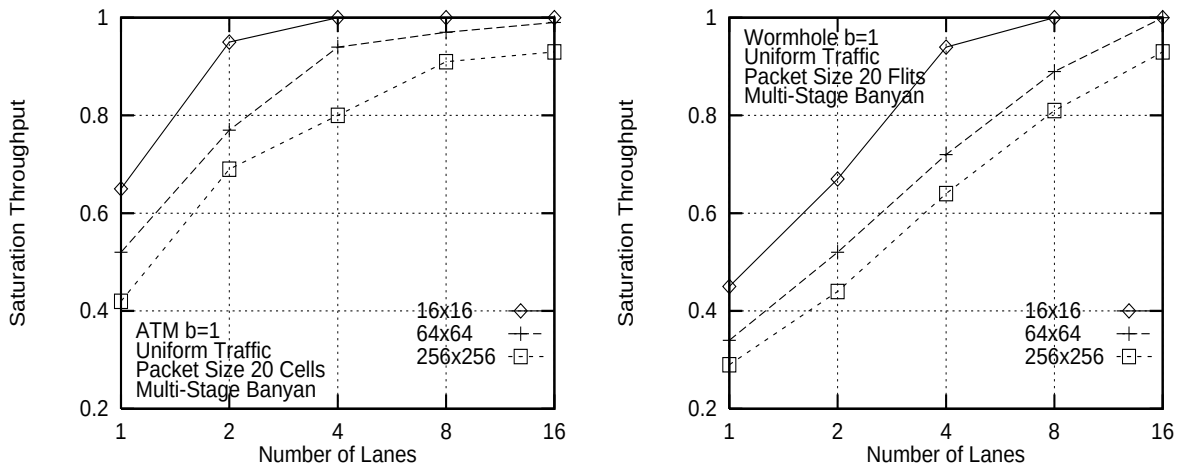
Στα πειράματα που ακολουθούν μελετάται η επίδραση της μεταβολής του μεγέθους του δικτύου

στην συνολική καθυστέρηση στην οποία ένα cell/flit υποβάλλεται όταν το φορτίο εισόδου είναι 0.4. Μελετάμε ακόμα και την επίδραση στην παροχή κορεσμού. Σε όλες τις διαμορφώσεις των πειραμάτων ο χώρος αποθήκευσης που είναι αφιερωμένος για μία λωρίδα είναι 1 (σε μονάδες cell/flit). Αυτό είναι αποδεκτό γιατί η καθυστέρηση περιφοράς (round−trip−delay) είναι μικρότερη του ενός cell time. Η κίνηση που παράγεται στην κάθε πηγή είναι και εδώ Poisson σε επίπεδο πακέτου (όπως στο 4.1.1). Το μέγεθος κάθε πακέτου είναι 20 cell/flit. Το πλήθος των εισόδων και των εξόδων του κάθε μεταγωγέα είναι σταθερό 2. Η τοπολογία του δικτύου είναι Banyan και το πλήθος των εισόδων και των εξόδων του είναι 16, 64 και 256.



Σχήμα 4.12: Μεταβολή καθυστέρησης καθώς μεταβάλλεται το μέγεθος του δικτύου

Στο σχήμα 4.12 βλέπουμε την καθυστέρηση σε συνάρτηση με το πλήθος των λωρίδων. Παρατηρούμε ότι η καθυστέρηση αυξάνει ελαφρά καθώς το μέγεθος του δικτύου αυξάνεται μέχρι το σημείο κορεσμού όπου οι καθυστερήσεις κατόπιν μεγαλώνουν (απεριόριστα στην περίπτωση του wormhole). Αυτό οφείλεται στο ότι καθώς μεγαλώνει το μέγεθος του δικτύου αυξάνονται και τα επίπεδα των μεταγωγέων μέσα σε αυτό. Ένα cell/flit πρέπει να περάσει από περισσότερους μεταγωγείς για να φτάσει στον προορισμό του.



Σχήμα 4.13: Μεταβολή παροχής κορεσμού καθώς μεταβάλλεται το μέγεθος του δικτύου

Στο σχήμα 4.13 βλέπουμε την παροχή κορεσμού σε συνάρτηση με το πλήθος των λωρίδων. Και εδώ παρατηρούμε ότι η παροχή κορεσμού μειώνεται καθώς το μέγεθος του δικτύου μεγαλώνει.

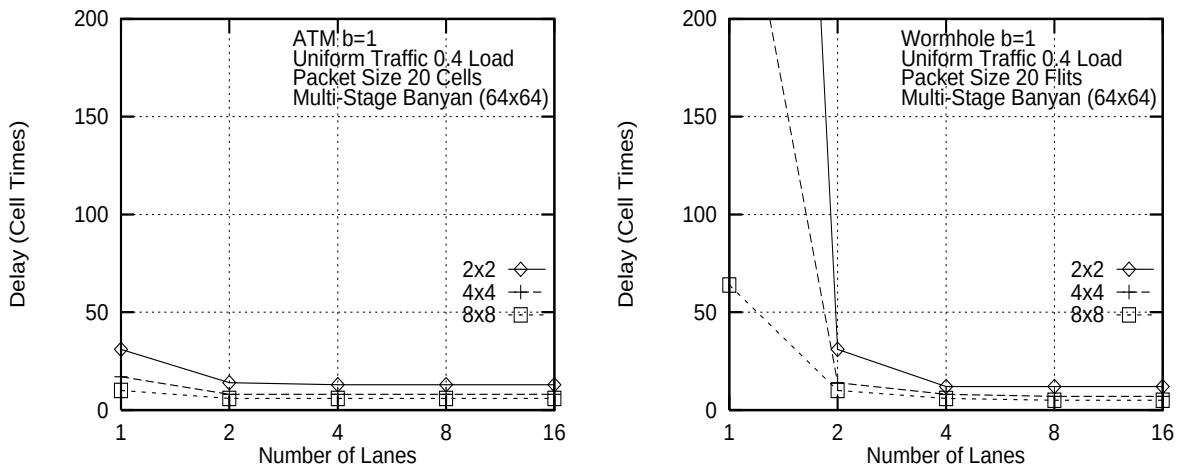
Αυτό οφείλεται στο γεγονός ότι καθώς μεγαλώνει το μέγεθος του δικτύου, αυξάνεται το πλήθος των μεταγωγέων από τους οποίους θα πρέπει να περάσει ένα πακέτο πριν φτάσει στον προορισμό του. Άρα στην γενική περίπτωση αυξάνεται και το πλήθος των πακέτων με τα οποία διεκδικεί τους πόρους του δικτύου για να προχωρήσει. Αφού οι πόροι του δικτύου δεν αυξάνονται, η παροχή από κάθε μεταγωγέα προς έναν προορισμό μειώνεται και άρα η παροχή πακέτων ανά προορισμό, συνολικά μειώνεται.

Συνολικά βλέπουμε ότι οι επιδόσεις του δικτύου χειροτερεύουν καθώς το μέγεθος του δικτύου μεγαλώνει.

4.2.2 Μεταβολή μεγέθους μεταγωγέα

Το μέγεθος των μεταγωγέων ενός δικτύου είναι σημαντική παράμετρος γιατί καθορίζει και το πλήθος των επιπέδων από τα οποία αποτελείται το δίκτυο. Όσο μεγαλώνει το μέγεθος του μεταγωγέα μειώνεται το πλήθος των επιπέδων του δικτύου. Επιπλέον μειώνεται και το πλήθος των μεταγωγέων που χρειάζονται για να κατασκευαστεί το δίκτυο. Είναι λοιπόν ενδιαφέρον, να δούμε πως το μέγεθος του μεταγωγέα επηρεάζει τις επιδόσεις του δικτύου.

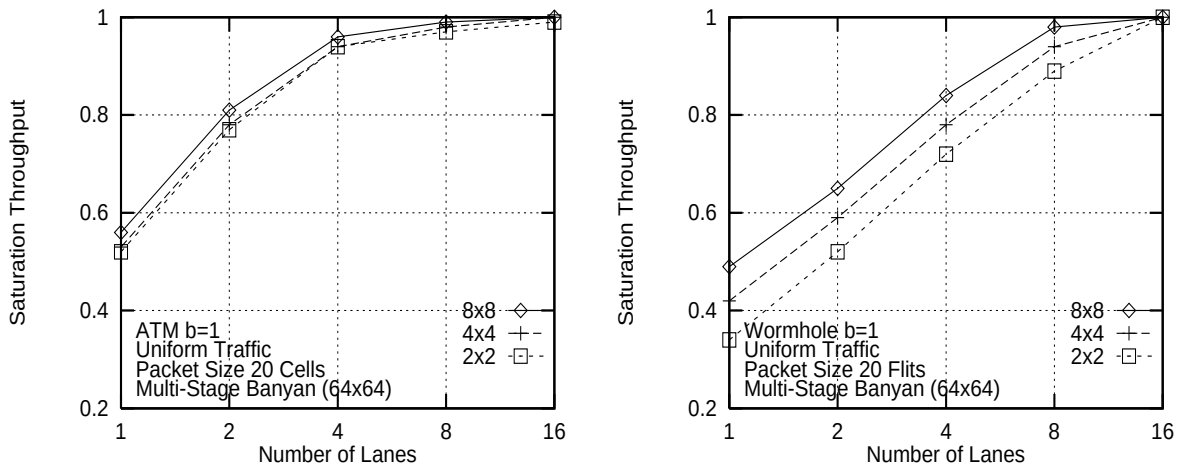
Στα πειράματα που ακολουθούν μελετάται η επίδραση της μεταβολής του μεγέθους του μεταγωγέα στην συνολική καθυστέρηση στην οποία ένα cell/flit υποβάλλεται όταν το φορτίο εισόδου είναι 0.4 καθώς και στην παροχή κορεσμού. Σε όλες τις διαμορφώσεις των πειραμάτων ο χώρος αποθήκευσης που είναι αφιερωμένος για μία λωρίδα είναι 1 (σε μονάδες cell/flit). Η κίνηση που παράγεται στην κάθε πηγή είναι και εδώ Poisson σε επίπεδο πακέτου (όπως στο 4.1.1). Το μέγεθος κάθε πακέτου είναι 20 cell/flit. Η τοπολογία του δικτύου είναι Banyan και το πλήθος των εισόδων και των εξόδων του είναι 64. Το πλήθος των εισόδων και των εξόδων του κάθε μεταγωγέα είναι 2, 4 ή 8.



Σχήμα 4.14: Μεταβολή συνολικής καθυστέρισης καθώς μεταβάλλεται το μέγεθος του μεταγωγέα

Στο σχήμα 4.14 βλέπουμε την καθυστέρηση σε συνάρτηση με το πλήθος των λωρίδων. Παρατηρούμε ότι η καθυστέρηση μειώνεται ελαφρά καθώς το μέγεθος του μεταγωγέα αυξάνεται μέχρι το σημείο κορεσμού όπου οι καθυστερήσεις κατόπιν μεγαλώνουν απεριόριστα στο wormhole. Αυτό συμβαίνει γιατί μειώνεται το πλήθος των επιπέδων του δικτύου και κατά συνέπεια το πλήθος των μεταγωγέων από τους οποίους ένα cell/flit πρέπει να περάσει για να φτάσει στον προορισμό του.

Στο σχήμα 4.15 βλέπουμε την παροχή κορεσμού σε συνάρτηση με το πλήθος των λωρίδων. Και εδώ παρατηρούμε ότι η παροχή κορεσμού αυξάνεται ελαφρά καθώς το μέγεθος του μεταγωγέα μεγαλώνει. Η τοπολογία στο εσωτερικό ενός μεταγωγέα είναι πλήρης δηλαδή είναι δυνατή η σύνδεση οποιασδήποτε εισόδου με μια έξοδο που δεν χρησιμοποιείται, ανεξάρτητα από το πλήθος των συνδέσεων που υπάρχουν ήδη και χωρίς εσωτερικές συγκρούσεις. Παρατηρούμε λοιπόν ότι όσο πιο λίγοι (και πιο μεγάλοι) είναι



Σχήμα 4.15: Μεταβολή παροχής κορεσμού καθώς μεταβάλλεται το μέγεθος του μεταγωγέα

οι μεταγωγείς από τους οποίους αποτελείται ένα δίκτυο τόσο πιο πλούσιο σε συνδέσεις είναι γιατί η τοπολογία του πλησιάζει περισσότερο στην πλήρη τοπολογία. Αφού το δίκτυο γίνεται πιο πλούσιο καθώς μεγαλώνει το μέγεθος των μεταγωγέων που το αποτελούν, γίνεται πιο εύκολη η διέλευση από ένα τέτοιο δίκτυο. Άρα η παροχή ενός δικτύου βελτιώνεται καθώς μεγαλώνει το μέγεθος των μεταγωγέων που το αποτελούν.

Συνολικά βλέπουμε ότι οι επιδόσεις του δικτύου βελτιώνονται καθώς το μέγεθος του μεταγωγέα μεγαλώνει.

Κεφάλαιο 5

Συμπεράσματα

Μελετήσαμε τις επιδόσεις δικτύων που αποτελούνται από μεταγωγείς ATM με έλεγχο ροής με εισιτήρια σε σύγκριση με τις επιδόσεις δικτύων wormhole με ανάλογες διαμορφώσεις, μέσω προσομοίωσης. Μεταβάλαμε δύο παράγοντες στο περιβάλλον της προσομοίωσης για να μελετήσουμε την συμπεριφορά των αρχιτεκτονικών κάτω από διαφορετικές συνθήκες. Μεταβάλαμε το μοντέλο κίνησης του δικτύου και το μέγεθος του δικτύου και του μεταγωγέα. Τα μοντέλα κίνησης που δοκιμάσαμε ήταν

- κίνηση Poisson, όπου ο προορισμός ενός πακέτου επιλέγεται τυχαία και ομοιόμορφα και το μέγεθος των πακέτων είναι σταθερό.
- εκρηκτική κίνηση, όπου μεταβάλλεται το μέγεθος των πακέτων αλλά ο προορισμός επιλέγεται και πάλι ομοιόμορφα.
- κίνηση με θερμά σημεία, όπου ορισμένοι προορισμοί επιλέγονται πιο συχνά από τους άλλους αλλά το μέγεθος των πακέτων είναι σταθερό.

Οι επιδόσεις του δικτύου στην περίπτωση του ATM, είναι σχεδόν πάντα καλύτερες από τις επιδόσεις στην περίπτωση του wormhole, εκτός από τις περιπτώσεις που το φορτίο στο σύστημα είναι χαμηλό και δεν υπάρχουν θερμά σημεία, όπου εκεί οι επιδόσεις είναι παρόμοιες. Ειδικά στην περίπτωση ύπαρξης θερμών σημείων, στην περίπτωση του ATM επιτυγχάνονται σημαντικά καλύτερες επιδόσεις από ότι στην περίπτωση του wormhole, στην οποία περίπτωση η κίνηση που δεν πηγαίνει στο θερμό σημείο, επιρρεάζεται πολύ από την κίνηση που πηγαίνει σε αυτό, ακόμα και όταν το πλήθος των λωρίδων είναι μεγαλύτερο από το πλήθος των θερμών σημείων. Στο ATM δεν συμβαίνει κάτι τέτοιο και αρκεί ελάχιστα μεγαλύτερο πλήθος λωρίδων από πλήθος θερμών σημείων, για να μην υπάρχει επίδραση. Στην περίπτωση που το μέγεθος του πακέτου μεγαλώνει, σε συνθήκες χαμηλού φορτίου, οι επιδόσεις των δύο δικτύων είναι παρόμοιες, εκτός από την περίπτωση όπου το πλήθος των λωρίδων είναι 1 στην οποία το wormhole, σε μεγάλα μεγέθη πακέτου, δεν συμπεριφέρεται καλά.

Ειδικά για την περίπτωση του ATM, παρατηρούμε ότι κάτω από εύκολες συνθήκες κυκλοφορίας της κίνησης, η προσθήκη περισσοτέρων από μίας λωρίδων, δεν φέρνει σημαντικά μεγάλη βελτίωση στις επιδόσεις. Όταν όμως τα μεγέθη των πακέτων αρχίζουν να μεγαλώνουν και ιδιαίτερα όταν υπάρχουν θερμά σημεία στο δίκτυο, η βελτίωση στην καθυστέρηση που υπάρχει είναι σημαντική. Στην περίπτωση που υπάρχει ένα θερμό σημείο, η προσθήκη μιας μόνο λωρίδας παραπάνω προκαλεί μείωση στην καθυστέρηση κατά περίπου 65 φορές.

Για την περίπτωση του wormhole, ήταν ήδη γνωστό [Dally 90], ότι η προσθήκη λωρίδων σε ένα δίκτυο που αποτελείται από μεταγωγείς wormhole, προκαλεί μεγάλη βελτίωση στις επιδόσεις. Με την δική μας εργασία, επιβεβαιώσαμε αυτά τα αποτελέσματα και είδαμε το φαινόμενο εντονότερο κάτω από πιο δύσκολες συνθήκες κυκλοφορίας.

Συνολικά μπορούμε να πούμε ότι το ATM επιτυγχάνει καλύτερες επιδόσεις από το wormhole σε ανάλογες διαμορφώσεις δικτύου. Σε ορισμένες διαμορφώσεις πειραμάτων οι επιδόσεις του ATM με έλεγχο ροής με εισιτήρια είναι πολύ καλές ακόμα και με ελάχιστες λωρίδες και άρα η προσθήκη λωρίδων δεν τις βελτιώνει (κίνηση Poisson). Σε άλλες περιπτώσεις παρατηρείται βελτίωση των επιδόσεων καθώς αυξάνεται το πλήθος των λωρίδων (κίνηση εκρηκτική και με θερμά σημεία). Η προσθήκη λωρίδων προκαλεί μεγάλες βελτιώσεις στην απόδοση του δικτύου στην περίπτωση του wormhole σε κάθε διαμόρφωση πειράματος.

Βιβλιογραφία

[ABCJ 92] A. Agarawal, R. Bianchini, D. Chaiken, K. Johnson, D. Kranz, J. Kubiatowitcz, B. Lim, K. Mackenzie and D. Yeung "The MIT Alewife Machine: Architecture and Performance" Proc. of the 22nd Int. Symposium on Computer Architecture, Santa Margherita Ligure, Italy, June 22−24 1995, pp. 2−13.

[BeTs 89] D.P.Bertsekas and J.N.Tsitsiklis "Parallel and Distributed Computation: Numerical Methods" Engelwood Cliffs, NJ: Prentice Hall, 1989.

[CSIM 94] "CSIM17 User's Guide", Mesquite Software, Inc.

[DUSH 93] R. Das, M. Uysal, J. Saltz, Y. Hwang "Communication Optimizations for Irregular Scientific Computations on Distributed Memory Architectures" University of Maryland, Department of Computer Science UMIACS−TR−93−109 1993.

[DaSe 87] W. J. Dally, C. L. Seitz, "Deadlock − Free Message Routing in Multiprocessor Interconnection Networks", IEEE Transactions on Computers, vol. 36, no. 5, May 1987, pp. 547−553.

[Dally 90] W. J. Dally, "Virtual − Channel Flow Control", Proc. of the 17th Int. Symp. on Computer Architecture, ACM SIGARCH vol. 18, no. 2, May 1990, pp. 60−68.

[KOHR 94] J. Kuskin, D. Ofelt, M. Heinlein, R. Simoni, K. Gharachorloo, J. Chapin, D. Nakahira, J. Baxter, M. Horowitz, A. Gupta, M. Rosenblum and J. Henessy, "The Stanford FLASH Multiprocessor", Proc. of the 21st Int. Symposium on Computer Architecture, Chicago Illinois, April 18−21 1994, pp. 302−313.

[KaVE 95] M. Katevenis, P. Vatsolaki, A. Efthymiou "Pipelined Memory Shared Buffer for VLSI Switches" Proceedings of the ACM SIGCOMM '95 Conference, Cambrigde, MA USA 30 August−1 Sep. 1995, pp. 39−48.

[Kate 94] M. Katevenis, "*Telegraphos*: High−Speed Communication Architecture for Parallel and Distributed Computer Systems", Techical Report FORTH−ICS/TR−123, ICS, FORTH, Heraklio, Crete, GR, May 1994

[Kung 92] H. T. Kung, "Gigabit Local Area Networks: A System Perspective", IEEE Communications Magazine, vol. 30, no. 4, April 1992, pp. 79−89.

[LLGW 92] D. Lenoski, J. Laudon, K. Gharachorloo, W.−D. Weber, A. Gupta, J. Henessy, M. Horowitz and M. S. Lam, "The Stanford Dash Multiprocessor", IEEE Computer, vol. 25, no. 3, March 1992, pp. 63−79 March 1992.

[NBKP 95] A. G. Nowatzyk, M. C. Browne, E. J. Kelly, M. Parkin, "S−Connect: From Networks of Workstations to Supercomputer Performance", 22nd Int. Symposium on Computer Architecture, Santa Margherita Ligure, Italy, June 22−24, 1995, pp. 71−82.

[Nara 88] M. J. Narasimha, "The Batcher−Banyan Self−Routing Network: Universality and Simplification", IEEE Transactions On Communications, vol. 36, no. 10, October 1988, pp. 1175−1188.

[OMSI 95] H. Oshaki, M. Murata, H. Suzuki, C. Ikeda, H. Miyahara "Rate−Based Congestion Control for ATM Networks", Computer Communication Review, ACM SIGCOMM, vol. 25, no. 2, April 1995, pp. 60−72.

[RoCG 94] R. Rooholamini, V. Cherkassky, M. Garver "Finding the Right ATM Switch for the market", IEEE Computer Magazine, vol 27, April 1994, pp. 16−28.

[Toba 90] F. A. Tobagi, "Fast Packet Switch Architectures For Broadband Integrated Services Digital Networks", Proceedings on the IEEE, vol. 78, no. 1, January 1990, pp. 133−167.

[Walr 91] Jean Walrand, "Communication Networks: A First Course", Aksen Associates, Boston, MA, 1991.

.1 Εισαγωγή

Ο προσομοιωτής sim υλοποιήθηκε για να μελετηθεί η λειτουργία δικτύων από μεταγωγείς και να συλλεγούν στατιστικά στοιχεία σχετικά με τις επιδόσεις αυτών των δικτύων. Τα δίκτυα αυτά μπορεί να αποτελούνται είτε από μεταγωγείς ATM, οι οποίοι ελέγχουν την ροή της κυκλοφορίας που διέρχεται από αυτούς χρησιμοποιώντας εισιτήρια (credit based flow control) είτε από μεταγωγείς wormhole οι οποίοι χρησιμοποιούν παρόμοιο τρόπο ελέγχου ροής.

Ο προσομοιωτής είναι υλοποιημένος ώστε να είναι δυνατό να εκτελεστεί σε SPARC Station της Sun Microsystems αλλά και σε σταθμούς εργασίας ALPHA της Digital. Ο πηγαίος κώδικας είναι γραμμένος στη γλώσσα προγραμματισμού C αλλά χρησιμοποιείται και η βιβλιοθήκη υποστήριξης CSIM. Η βιβλιοθήκη παρέχει ρουτίνες που βοηθούν στο να υλοποιηθεί ένας προσομοιωτής που οδηγείται από γεγονότα (event driven simulator).

.2 Χρήση

.2.1 Είσοδος

Όταν έχουμε το εκτελέσιμο αρχείο μπορούμε να το χρησιμοποιήσουμε είτε εκτελώντας το χωρίς είσοδο, οπότε θα αρχίσει να δέχεται είσοδο από το πληκτρολόγιο, είτε βάζοντας με ανακατεύθυνση (redirection) σαν είσοδο ένα αρχείο στο οποίο περιγράφεται η διαμόρφωση του πειράματος που θέλουμε να κάνουμε. Τα δεδομένα που ζητάει ο προσομοιωτής, με την σειρά που τα ζητάει, είναι τα ακόλουθα :

- Το πλήθος των πακέτων μετά την παράδοση των οποίων θα σταματήσει η προσομοίωση. Το κριτήριο τερματισμού της προσομοίωσης είναι η παράδοση στους προορισμούς ενός καθορισμένου από την αρχή πλήθους από πακέτα.

- Το πλήθος των πακέτων μετά την παράδοση των οποίων αρχίζει η συλλογή των στατιστικών στοιχείων σχετικά με τις επιδόσεις του δικτύου. Στην αρχή της προσομοίωσης το δίκτυο βρίσκεται σε μια κατάσταση που ονομάζεται *μεταβατική (transient)*. Κατά την διάρκεια της μεταβατικής κατάστασης, οι επιδόσεις του δικτύου δεν είναι ακριβείς και σταθερές. Δεν συμφωνούν με τις τιμές όταν το δίκτυο βρίσκεται σε *σταθερή κατάσταση*, όπου οι τιμές των μετρούμενων χαρακτηριστικών του δικτύου έχουν σταθεροποιηθεί και δεν μεταβάλλονται. Αν συμπεριλάβουμε τις επιδόσεις κατά την διάρκεια της μεταβατικής περιόδου στην στατιστική μελέτη των μετρήσεων, τότε αλλοιώνεται η εικόνα των στατιστικών. Η διάρκεια της μεταβατικής κατάστασης του δικτύου συνήθως βρίσκεται με πειραματικό τρόπο.

Τα επόμενα δύο ορίσματα διαφέρουν ανάλογα με το ποιου είδους δικτύου την λειτουργία θέλουμε να προσομοιώσουμε (ATM ή Wormhole).

1. Στην περίπτωση του ATM τα δύο επόμενα ορίσματα είναι τα:

   - Μέγιστο πλήθος εισιτηρίων προορισμού (destination credits) για κάθε σύνδεσμο εξόδου ενός μεταγωγέα. Είναι ο αριθμός με τον οποίο αρχικοποιούνται οι μετρητές των εισιτηρίων προορισμού για κάθε έξοδο κάθε μεταγωγέα και παραγωγού.
   - Μέγιστο πλήθος εισιτηρίων αναχώρησης (pool credits) για κάθε σύνδεσμο εξόδου ενός μεταγωγέα. Είναι ο αριθμός με τον οποίο αρχικοποιούνται οι μετρητές των εισιτηρίων αναχώρησης για κάθε έξοδο κάθε μεταγωγέα και παραγωγού.

2. Στην περίπτωση του wormhole τα δύο επόμενα ορίσματα είναι τα:

- Πλήθος λωρίδων (lanes) ανά σύνδεσμο. Αυτός είναι ο αριθμός των πακέτων που μπορούν ταυτόχρονα να πολυπλεχθούν πάνω σε ένα σύνδεσμο ότα το δίκτυο αποτελείται από μεταγωγείς wormhole.
- Μέγιστο πλήθος εισιτηρίων (credits) ανά λωρίδα. Ο αριθμός αυτός υποδεικνύει το πλήθος των flit [1] που μπορεί να φύγουν χρησιμοποιώντας μία λωρίδα ενός συνδέσμου χωρίς να χρειάζεται ένα εισιτήριο να επιστρέψει. Ισοδύναμα είναι το μέγιστο πλήθος flit ενός πακέτου που μπορούν να βρεθούν σε ένα μεταγωγέα.

- Ελάχιστο μέγεθος πακέτου. Το μέγεθος του πακέτου μετριέται σε πλήθος cell/flit. Ένα πακέτο πρέπει να έχει οποσδήποτε ένα cell/flit επικεφαλίδα και άλλο ένα για ουρά. Γι αυτόν τον λόγο ελάχιστο μέγεθος πακέτου είναι το 2. Αν αυτή η παράμετρος για παράδειγμα πάρει την τιμή 0, αυτό σημαίνει ότι το ελάχιστο μέγεθος πακέτου είναι 2.

- Μέγιστο μέγεθος πακέτου. Και εδώ ισχύουν ότι και παραπάνω.

Αν το μέγιστο πλήθος πακέτου είναι μεγαλύτερο από το ελάχιστο τότε γίνεται και άλλη μία ερώτηση σχετική με την πολιτική επιλογής μεγέθους πακέτου. Η πολιτική επιλογής μεγέθους πακέτου μπορεί να είναι μία από τις ακόλουθες:

1. Ομοιόμορφη (uniform): όταν η πολιτική επιλογής μεγέθους πακέτου είναι ομοιόμορφη, το μέγεθος ενός πακέτου που δημιουργείται, επιλέγεται ομοιόμορφα μεταξύ του ελάχιστου και του μέγιστου μεγέθους πακέτου. Αν η επιθυμητή πολιτική επιλογής μεγέθους πακέτου είναι η ομοιόμορφη, τότε η ερώτηση

   `Enter packet size selection policy (uniform/bimodal) :`

   πρέπει να απαντηθεί με το string `uniform`.

2. Διπολική (bimodal): όταν η πολιτική επιλογής μεγέθους πακέτου είναι διπολική, το μέγεθος ενός πακέτου που δημιουργείται είναι είτε το μέγιστο είτε το ελάχιστο. Μέσα στο δίκτυο κυκλοφορούν δύο ειδών πακέτα (μικρά και μεγάλα) και ο όγκος της κίνησης από αυτά τα δύο είδη πακέτων είναι ίδιος. Αυτό σημαίνει ότι το πλήθος των cell/flit που ανήκουν σε κάθε είδος πακέτου είναι ίδιο. Κατά συνέπεια, το πλήθος των μικρών πακέτων που παράγονται είναι μεγαλύτερο από το πλήθος των μεγάλων πακέτων. Αν, για παράδειγμα, το μέγεθος του μεγάλου πακέτου είναι 10 φορές το μέγεθος του μικρού, τότε παράγονται 10 φορές περισσότερα πακέτα με μικρό μέγεθος. Αν η επιθυμητή επιλογή πολιτική επιλογής μεγέθους πακέτου είναι η διπολική, τότε η ερώτηση

   `Enter packet size selection policy (uniform/bimodal) :`

   πρέπει να απαντηθεί με το string `bimodal`.

- Φορτίο εισόδου. Ο αριθμός αυτός είναι σε μορφή κλάσματος της μονάδας. Το φορτίο του δικτύου ορίζεται σαν τον λόγο $\frac{\lambda}{\mu}$ όπου λ είναι ο ρυθμός με τον οποίο εισέρχεται κίνηση μέσα στο δίκτυο και μ ο ρυθμός με τον οποίο το δίκτυο εξυπηρετεί (δέχεται) την κίνηση αυτή. Νόμιμες τιμές που μπορεί να πάρει αυτό το όρισμα είναι για παράδειγμα 0.2, 0.5, 1.0, 2.0 κ.ά. Φορτίο πάνω από 1.0 σημαίνει ότι οι πηγές παράγουν κίνηση με ρυθμό μεγαλύτερο από αυτόν που το δίκτυο μπορεί να την απορροφήσει. Αυτό σημαίνει ότι στις εισόδους του δικτύου θα δημιουργηθούν ουρές που θα μεγαλώνουν απεριόριστα και άρα η συνολική καθυστέρηση ενός πακέτου θα είναι και αυτή απεριόριστη και θα μεγαλώνει όσο αυξάνεται η διάρκεια της προσομοίωσης. Ειδικά στην περίπτωση που προσομοιώνεται κίνηση με θερμά σημεία (hot spots), αυτή η παράμετρος πρέπει να περιέχει το συνολικό φορτίο της κίνησης που εισέρχεται στο δίκτυο. Αν θέλουμε το φορτίο

[1] Flit είναι η ελάχιστη ποσότητα πληροφορίας στην οποία εφαρμόζεται έλεγχος ροής.

της κίνησης που προορίζεται σε κάθε θερμό σημείο να είναι 1.0 ,το πλήθος των θερμών σημείων είναι 2 και το φορτίο της κίνησης που δεν προορίζεται σε θερμά σημεία να είναι 0.4, τότε το συνολικό φορτίο του δικτύου είναι 2.4 (1.0 για κάθε ένα από τα θερμά σημεία και 0.4 το κανονικό φορτίο).

- Μέγεθος cell/flit. Κάθε cell/flit έχει και αυτό με την σειρά του δομή και χωρίζεται σε μικρότερα κομμάτια. Το μέγεθος cell/flit ουσιαστικά καθορίζει το πλήθος των κύκλων που χρειάζεται ένα cell/flit για να μεταδοθεί.

- Μονοπάτι του αρχείου που περιέχει πληροφορίες για την τοπολογία του δικτύου. Ο τρόπος με τον οποίο η τοπολογία ενός δικτύου περιγράφεται παρουσιάζεται αναλυτικά στην παράγραφο .2.2.

- Μονοπάτι του αρχείου που περιέχει τον πίνακα δρομολόγησης κάθε μεταγωγέα. Ο τρόπος με τον οποίον κατασκευάζεται αυτό το αρχείο περιγράφεται αναλυτικά στην παράγραφο .2.3.

- Μέγιστο πλήθος συνδέσμων που εισέρχονται σε έναν μεταγωγέα. Δεν είναι υποχρεωτικό κάθε μεταγωγέας να έχει ακριβώς τόσους συνδέσμους εισόδου. Μπορεί να έχει και λιγότερους. Δεν θα πρέπει όμως να εμφανίζονται περισσότεροι σύνδεσμοι στην περιγραφή του δικτύου από το πλήθος που αναφέρεται εδώ.

- Μέγιστο πλήθος συνδέσμων που εξέρχονται από έναν μεταγωγέα. Και εδώ ισχύουν τα παραπάνω και μπορεί ένας μεταγωγέας να έχει λιγότερους από το μέγιστο πλήθος συνδέσμους εξόδου.

- Πλήθος των θερμών σημείων (hot spots) του δικτύου. Σαν θερμό σημείο μπορεί να επιλεγεί να γίνει ένας προορισμός. Αυτό σημαίνει ότι από κάθε πηγή, εκτός του φορτίου που κατανέμεται ομοιόμορφα σε όλους τους προορισμούς, παράγεται και κίνηση με επιπλέων φορτίο για κάθε θερμό σημείο. Εφόσον το πλήθος των θερμών σημείων δεν είναι 0 τότε πρέπει να δηλωθούν ποιοι προορισμοί θα είναι θερμά σημεία. Θα πρέπει λοιπόν να ακολουθήσουν οι αύξοντες αριθμοί των προορισμών που επιλέγονται για θερμά σημεία (οι αριθμοί αυτοί πρέπει να βρίσκονται μεταξύ 0 και του μέγιστου πλήθους προορισμών μείον έναν).

- Πιθανότητα με βάση την οποία ο προορισμός ενός πακέτου που μόλις κατασκευάζεται είναι ένας από τα θερμά σημεία. Έστω ότι το δίκτυο έχει ν θερμά σημεία και το φορτίο της κίνησης που προορίζεται σε προορισμούς που δεν είναι θερμά σημεία είναι κ. Έστω λ το συνολικό φορτίο της κίνησης που μπαίνει μέσα στο δίκτυο (συμπεριλαμβανομένου και του φορτίου για κάθε θερμό σημείο). Η πιθανότητα με την οποία επιλέγεται σαν προορισμός ενός καινούριου πακέτου ένα θερμό σημείο είναι $1 - \frac{\kappa}{\lambda}$.

.2.2 Τοπολογία δικτύου

Το αρχείο που περιέχει την τοπολογία του δικτύου αποτελείται μόνο από αριθμούς που ανάλογα με την θέση τους περιγράφουν μία τοπολογία. Η δομή του αρχείου είναι η ακόλουθη:

1. πλήθος και αριθμοί−ταυτότητα παραγωγών (πηγών).
2. πλήθος και αριθμοί−ταυτότητα μεταγωγέων.
3. πλήθος και αριθμοί−ταυτότητα καταναλωτών (προορισμών).
4. περιγραφή των συνδέσμων του δικτύου.

Αρχικά δηλώνουμε το πλήθος των παραγωγών που υπάρχουν στο δίκτυο και κατόπιν γράφουμε τους αριθμούς−ταυτότητα (IDs) των παραγωγών αυτών. Οι αριθμοί αυτοί πρέπει να είναι συνεχόμενοι ακέραιοι. Στην παρούσα υλοποίηση του προσομοιωτή ο αριθμός−ταυτότητα 0 δεν είναι νόμιμος. Για λόγους οικονομίας μνήμης είναι καλό να κρατηθούν οι αριθμοί−ταυτότητα όσο το δυνατό μικρότεροι. Μετά το τέλος της δήλωσης των αριθμών−ταυτότητα των παραγωγών ακολουθεί η δήλωση του πλήθους των μεταγωγέων (switches) και μετά οι αριθμοί−ταυτότητά τους. Και εδώ ισχύει ότι και προηγουμένως ότι οι αριθμοί−ταυτότητα πρέπει να είναι συνεχόμενοι και είναι καλό να είναι όσο το δυνατό μικρότεροι για να γίνεται οικονομία σε μνήμη. Κατόπιν ακολουθούν το πλήθος και οι αριθμοί−ταυτότητα των καταναλωτών που και εδώ ισχύουν τα παραπάνω.

Μετά την δήλωση των παραγωγών, των μεταγωγέων και των καταναλωτών ακολουθεί η δήλωση των συνδέσμων του δικτύου. Ένας σύνδεσμος καθορίζεται από τους κόμβους που συνδέει. Άρα για να δηλώσουμε έναν σύνδεσμο πρέπει να δηλώσουμε διαδοχικά τον αριθμό−ταυτότητα του κόμβου από τον οποίο ξεκινάει και τον αριθμό−ταυτότητα του κόμβου στον οποίο καταλήγει. Σε αυτό το σημείο πρέπει να αναφέρουμε ότι στον προσομοιωτή όλες οι συνδέσεις μεταξύ κόμβων είναι από σημείο σε σημείο (point−to−point). Οι αριθμοί−ταυτότητα που εμφανίζονται στις δηλώσεις των συνδέσμων είναι αυτοί που έχουν οριστεί παραπάνω κατά την διάρκεια των δηλώσεων των κόμβων (παραγωγοί, μεταγωγείς, καταναλωτές).

.2.3 Πίνακας δρομολόγησης

Για κάθε μεταγωγέα πρέπει να υπάρχει ένας πίνακας δρομολόγησης που για κάθε πακέτο που φτάνει σε έναν μεταγωγέα να είναι δυνατό να βρίσκεται ο σύνδεσμος από τον οποίο θα φύγει το πακέτο έχοντας σαν δεδομένο τον προορισμό του πακέτου. Αυτές είναι οι σταθερές πληροφορίες που εξαρτώνται από την τοπολογία του δικτύου. Αυτές οι πληροφορίες πρέπει να περιέχονται σε ένα αρχείο που περιέχει όλες τις πληροφορίες δρομολόγησης που χρειάζεται κάθε μεταγωγέας. Κάθε μεταγωγέας του δικτύου πρέπει να αναφέρεται σε αυτό το αρχείο. Στην συνέχεια περιγράφεται ο τρόπος με τον οποίο περιγράφεται ο πίνακας δρομολόγησης ενός μεταγωγέα μέσα στο αρχείο.

Εμφανίζεται αρχικά μία γραμμή της μορφής:

`switch <switch-id> <dest_number>`

όπου η λέξη `switch` είναι λέξη−κλειδί (keyword), το <switch−id> είναι ο αριθμός−ταυτότητα του μεταγωγέα όπως αυτός έχει δηλωθεί στο αρχείο που περιγράφει την τοπολογία (βλέπε και .2.2) και <dest_number> είναι το πλήθος των προορισμών στους οποίους μπορεί να φτάσει ένα πακέτο από τον μεταγωγέα. Αυτή η γραμμή δηλώνει την έναρξη δήλωσης του πίνακα δρομολόγησης.

Κατόπιν αρχίζει μια λίστα από ζευγάρια αριθμών. Το πρώτο μέλος του ζευγαριού είναι ο προορισμός και το δεύτερο ο σύνδεσμος εξόδου του μεταγωγέα από τον οποίο θα πρέπει να βγει το πακέτο για να πάει στον προορισμό του. Ο προορισμός είναι ένας αριθμός−ταυτότητα όπως αυτός εμφανίζεται στο αρχείο της τοπολογίας. Ο σύνδεσμος του μεταγωγέα που είναι το δεύτερο μέλος του ζευγαριού είναι ένας αύξοντας αριθμός. Συγκεκριμένα είναι ο αύξον αριθμός της θέσης στην οποία εμφανίζεται ο σύνδεσμος, μέσα στο σύνολο των συνδέσμων του δικτύου που σαν κόμβο εκκίνησης εμφανίζουν τον κόμβο με αριθμό−ταυτότητα <switch−id>. Η διάταξη αυτού του συνόλου γίνεται σύμφωνα με την σειρά που εμφανίζονται αυτοί οι σύνδεσμοι μέσα στο αρχείο περιγραφής της τοπολογίας του δικτύου ξεκινώντας την αρίθμηση από το 0. Ο πρώτος σύνδεσμος που εμφανίζει τον μεταγωγέα <switch−id> σαν μεταγωγέα εκκίνησης είναι ο σύνδεσμος εξόδου με αύξοντα αριθμό 0 του μεταγωγέα αυτού. Στην λίστα αυτή πρέπει να εμφανίζονται τόσοι προορισμοί όσοι έχουν δηλωθεί αρχικά (<dest_number>) γιατί αλλιώς εμφανίζεται μύνημα λάθους. Με τον ίδιο τρόπο κατασκευάζονται και οι πίνακες δρομολόγησης για όλους τους μεταγωγείς.

.2.4 Έξοδος

Η έξοδος του προσομοιωτή γράφεται σε ένα αρχείο μετά το τέλος της προσομοίωσης. Το αρχείο αυτό λαμβάνει όνομα ανάλογα με την διαμόρφωση του πειράματος που έγινε. Το όνομα έχει επτά πεδία που χωρίζονται με παύλες (underscores) και είναι τα ακόλουθα

- τύπος δικτύου. Ανάλογα με τον τύπο του δικτύου το πρώτο συνθετικό μπορεί να είναι atm ή whl, για δίκτυα ATM και wormhole αντίστοιχα.
- μέγεθος δικτύου. Εδώ μπαίνει το πλήθος των εισόδων του δικτύου.
- πλήθος εισιτηρίων προορισμού (στην περίπτωση ATM) ή πλήθος λωρίδων (στην περίπτωση wormhole).
- πλήθος εισιτηρίων αναχώρησης (στην περίπτωση ATM) ή πλήθος εισιτηρίων (στην περίπτωση wormhole).
- μέγιστο μέγεθος πακέτου.
- πλήθος θερμών σημείων.
- φορτίο της κίνησης σε κάθε σύνδεσμο εισόδου.

Στο αρχείο εξόδου του προσομοιωτή περιέχονται τα εξής στοιχεία

- στοιχεία σχετικά με την ημερομηνία, την πλατφόρμα πάνω στην οποία εκτελέστηκε η προσομοίωση, καθώς και την διάρκειά της.
- στοιχεία σχετικά με την χρησιμοποίηση της κάθε εισόδου ξεχωριστά.
- η καθυστέρηση μέσα στο δίκτυο των επικεφαλίδων πακέτων.
- η καθυστέρηση μέσα στο δίκτυο του πακέτου (από την στιγμή που η επικεφαλίδα μεταδίδεται μέχρι την στιγμή που η ουρά παραλαμβάνεται από τον καταναλωτή).
- η καθυστέρηση μέσα στο δίκτυο του cell/flit.
- στην περίπτωση του bimodal μοντέλου κίνησης η καθυστέρηση των cell/flit που ανήκουν σε μικρά και μεγάλα πακέτα.
- η καθυστέρηση στις ουρές στις εισόδους του δικτύου.
- η καθυστέρηση σε κάθε μία ουρά εισόδου ξεχωριστά.
- η καθυστέρηση των cell/flit μέσα στο δίκτυο για κάθε προορισμό ξεχωριστά.
- η καθυστέρηση των επικεφαλίδων των πακέτων για κάθε προορισμό ξεχωριστά.
- η χρησιμοποίηση του κάθε συνδέσμου εξόδου ξεχωριστά.
- η μέση χρησιμοποίηση των συνδέσμων εξόδου (throughput).

.3 Τεχνικές πληροφορίες

Στα πλαίσια της εργασίας για το Msc έχουν υλοποιηθεί δύο προσομοιωτές, ένας που προσομοιώνει την λειτουργία δικτύων ATM και άλλος για δίκτυα wormhole. Και οι δύο προσομοιωτές έχουν παρόμοια δομή.

Και οι δύο καθοδηγούνται από γεγονότα (event driven). Στο υπό προσομοίωση σύστημα υπάρχει ένα λογικό ρολόι σύμφωνα με το οποίο προγραμματίζεται τι θα συμβεί. Κάθε συμβάν που πρόκειται να γίνει κατά την διάρκεια της προσομοίωσης μπαίνει ανάλογα με την χρονική στιγμή (τιμή του λογικού ρολογιού) που πρόκειται να συμβεί σε μία λίστα συμβάντων (event list). Όλα τα γεγονότα που είναι προγραμματισμένα να συμβούν την ίδια στιγμή, εκτελούνται σειριακά αλλά δεν κυλάει το ρολόι της προσομοίωσης. Όταν όλα τα γεγονότα που είναι προγραμματισμένα να συμβούν την ίδια στιγμή, εκτελεστούν, τότε το λογικό ρολόι προχωράει στην επόμενη χρονική στιγμή που έχει προγραμματιστεί κάποιο γεγονός.

Το δίκτυο αποτελείται από:

- παραγωγούς (πηγές) κίνησης. Υπάρχει ένας παραγωγός για κάθε είσοδο του δικτύου και αυτό που κάνει είναι να παράγει ορισμένου μοντέλου κίνησης και να το εισάγει στο δίκτυο. Οι παραγωγοί υπακούουν στον έλεγχο ροής και δέχονται εισιτήρια από το δίκτυο που τους επιτρέπουν να στείλουν πληροφορία σε αυτό.

- καταναλωτές (προορισμούς). Υπάρχει ένας καταναλωτής σε κάθε έξοδο του δικτύου και κάθε ένας αναλαμβάνει να βγάλει την κίνηση από το δίκτυο που κατευθύνεται σε αυτόν. Ταυτόχρονα ο καταναλωτής, για κάθε κομμάτι πληροφορίας που βγάζει από το δίκτυο, στέλνει και ένα εισιτήριο πίσω στο δίκτυο.

- μεταγωγείς (switches). Οι μεταγωγείς αναλαβάνουν να μεταφέρουν την πληροφορία από τους παραγωγούς στους καταναλωτές. Οι μεταγωγείς είναι στην γενική περίπτωση συνδεδεμένοι μεταξύ τους, σχηματίζοντας ένα δίκτυο. Η όλη προσομοίωση γίνεται για να μετρηθούν οι επιδόσεις αυτών των μεταγωγέων.

Η λειτουργία κάθε καταναλωτή και μεταγωγέα προσομοιώνεται με ξεχωριστό νήμα. Για κάθε ένα καταναλωτή και μεταγωγέα, ο προσομοιωτής έχει και ένα νήμα (thread) για να προσομοιώσει την λειτουργία του. Για κάθε παραγωγό υπάρχουν 2 νήματα. Τα νήματα είναι ροές εκτέλεσης οι οποίες δεν έχουν δικό τους χώρο ανατεθιμένο από το λειτουργικό σύστημα αλλά εκτελούνται μέσα στον χώρο που το λειτουργικό σύστημα έχει αναθέσει σε μία διεργασία. Μέσα σε αυτόν τον χώρο μπορεί να "τρέχουν" πολλά νήματα. Ένα νήμα είναι οικονομικότερο από μία διεργασία σε υπολογιστική ισχύ γιατί κάθε φορά που γίνεται αλλαγή στο νήμα που εκτελείται, δεν χρειάζεται να επέμβει το λειτουργικό σύστημα και έτσι δεν είναι δαπανηρή μία εναλλαγή νημάτων. Στην παρούσα υλοποίηση ξεχωριστό νήμα είναι και κάθε πακέτο που παράγεται από τους παραγωγούς. Τα νήματα δημιουργούνται με την βοήθεια μίας βιβλιοθήκης υποστήριξης, το CSIM.

Το CSIM αποτελεί μία βιβλιοθήκη η οποία συνδέεται (links) με τις ρουτίνες που αποτελούν τον προσομοιωτή και παρέχει λειτουργίες που δημιουργούν νήματα, κρατούν και προγραμματίζουν την λίστα με τα γεγονότα με διαφανή τρόπο για τον προγραμματιστή και παρέχουν ρουτίνες που προσομοιώνουν την λειτουργία κομματιών του υπό προσομοίωση συστήματος. Για να "χτιστεί" το σύστημα που θέλουμε να προσομοιώσουμε, στην αρχή εκτελείται μία συνάρτηση που ονομάζεται `sim()` και η οποία αναλαμβάνει να εισάγει τα δεδομένα και τις παραμέτρους της προσομοίωσης, να ξεκινήσει μερικά από τα νήματα που προσομοιώνουν την λειτουργία κομματιών του συστήματος και περιμένει μέχρι το τέλος της προσομοίωσης, για να τυπώσει τα στατιστικά που μαζεύτηκαν κατά την διάρκεια της προσομοίωσης και να τερματίσει την εκτέλεση του προσομοιωτή.

Η πληροφορία μετακινείται μέσα στο δίκτυο σε μικρά κομμάτια που στην περίπτωση του ATM λέγονται cells και στην περίπτωση του wormhole λέγονται flits. Τα cells και τα flits είναι το μικρότερο κομμάτι πληροφορίας στο οποίο εφαρμόζεται έλεγχος ροής. Τα cells και τα flits έχουν και αυτά δομή και χωρίζονται σε μικρότερα κομμάτια που ο W. J. Dally τα ονομάζει phits. Το μέγεθος του cell/flit μετριέται σε πλήθος phits και ουσιαστικά αντιστοιχεί στο πλήθος των κύκλων ρολογιού που χρειάζεται ένα cell/flit για να μεταδοθεί πάνω σε έναν σύνδεσμο.

Ο έλεγχος ροής που χρησιμοποιείται και στους δύο προσομοιωτές γίνεται μέσω εισιτηρίων (credits) και αυτό γιατί εξασφαλίζεται ότι δεν θα χαθούν cell/flit επειδή ο χώρος αποθήκευσης κάποιου μεταγωγέα γέμισε και έτσι αναγκάστηκε να πετάξει τα cells/flits που του ήρθαν γιατί δεν μπορούσε να τα αποθηκεύσει.

Οι συναρτήσεις από τις οποίες αποτελείται ο προσομοιωτής είναι οι ακόλουθες:

- `sim()`
- `Medium(<arguments>)`
- `producer_code(<arguments>)`
- `packet_generator_code(<arguments>)`
- `credit_collector_code(<arguments>)`
- `switch_code(<arguments>)`
- `consumer_code(<arguments>)`

Ακόμα, σε πειραματικό στάδιο, υπάρχει και η συνάρτηση
`confint(<arguments>)`
η οποία χρησιμοποιείται για να γίνει επεξεργασία των στατιστικών στοιχείων και να μελετηθεί η αξιοπιστία τους, υπολογίζοντας (στο τέλος της προσομοίωσης) τα διαστήματα εμπιστοσύνης για τις μετρήσεις που έχουν συλλεγεί.

Και στους δύο προσομοιωτές οι συναρτήσεις έχουν τα ίδια ονόματα και τα ίδια ορίσματα εκτός από την `packet_generator_code` που έχει λίγο διαφορετικά ορίσματα σε κάθε περίπτωση. Στις επόμενες παραγράφους θα περιγράψουμε αναλυτικά την λειτουργία κάθε συνάρτησης.

.3.1 Συνάρτηση `sim`

Η συνάρτηση `sim` είναι ειδική συνάρτηση και χρησιμοποιείται για την αρχικοποίηση του υπό προσομοίωση συστήματος. Με την σύνδεση της βιβλιοθήκης CSIM με τον υπόλοιπο κώδικα του προσομοιωτή, η συνάρτηση `sim` είναι η πρώτη συνάρτηση που εκτελείται (αντί της main που ορίζει η C).

Αρχικά ζητούνται οι σχετικές με το σύστημα πληροφορίες από τον χρήστη (οπως περιγράφεται στο user's guide) και ορίζονται τα όρια των παραμέτρων προσομοίωσης. Για να ορίσουμε τα όρια αυτά, εισάγουμε τους ακόλουθους συμβολισμούς:

- N_p : πλήθος παραγωγών του δικτύου.
- N_c : πλήθος καταναλωτών του δικτύου.
- N_s : πλήθος μεταγωγέων του δικτύου.
- M : πλήθος πακέτων που χρειάζεται να παραληφθούν από τους καταναλωτές για να τερματίστει η προσομοίωση.

- C_s : μέγεθος cell/flit που μετριέται σε πλήθος phits, και υποδεικνύει το πλήθος των κύκλων ρολογιού που χρειάζεται ένα cell/flit για να μεταδοθεί.
- N_o : πλήθος συνδέσμων εξόδου ενός μεταγωγέα.
- N_i : πλήθος συνδέσμων εισόδου ενός μεταγωγέα.

Τα όρια που είναι χρήσιμα για τον προσομοιωτή είναι τα ακόλουθα:

- max_processes, το μέγιστο πλήθος νημάτων που μπορούν ταυτόχρονα να είναι ενεργά κατά την διάρκεια της προσομοίωσης. Το θεωρητικό μέγιστο είναι όσο το πλήθος των νημάτων που χρειάζονται για τους παραγωγούς, τους καταναλωτές και τους μεταγωγείς ($2N_p + N_c + N_s$) συν το μέγιστο πλήθος των πακέτων που χρειάζεται να παραληφθούν από τους καταναλωτές για να τελειώσει η προσομοίωση (M). Ακόμα υπάρχει συνεχώς ενεργό ένα νήμα που αναλαμβάνει την επικοινωνία μεταξύ των άλλων νημάτων, το `Medium`. Άρα συνολικά το πλήθος των νημάτων που μπορεί να είναι ενεργά (στην χειρότερη περίπτωση) είναι

$$2N_p + N_c + N_s + M + 1$$

- max_messages, το μέγιστο πλήθος των μηνυμάτων που μπορούν ταυτόχρονα να μεταδίδονται μέσα στο δίκτυο. Το μέγιστο πλήθος τέτοιων μυνημάτων υπολογίζεται ως εξής: μπορούν να στέλνουν μήνυμα όλοι οι παραγωγοί και οι μεταγωγείς και εισιτήριο όλοι οι καταναλωτές και οι μεταγωγείς. Ένας παραγωγός, για λόγους οικονομίας σε εναλλαγές εκτελούμενου νήματος, όταν στέλνει ένα cell/flit, στέλνει όλα τα phits μαζί και άρα εφόσον όλοι οι παραγωγοί μεταδίδουν ταυτόχρονα, επιβαρύνουν τον προσομοιωτή με N_pC_s μηνύματα. Ταυτόχρονα μπορεί και όλοι οι μεταγωγείς να μεταδίδουν μηνύματα και να επιβαρύνουν τον προσομοιωτή με N_sN_o. Ταυτόχρονα, κάθε καταναλωτής και μεταγωγέας μπορεί να στέλνει και ένα εισιτήριο και να επιβαρύνει με $N_c + N_iN_s$ μηνύματα. Συνολικά το πλήθος των μηνυμάτων που ταυτόχρονα μπορεί να είναι ενεργά στον προσομοιωτή είναι

$$N_pC_s + N_sN_o + N_c + N_iN_s$$

- max_mailboxes, το πλήθος των γραμματοκυβωτίων που υπάρχουν στον προσομοιωτή. Τα γραμματοκυβώτια είναι δομές στις οποίες καταλήγουν τα μηνύματα ώστε να μπορούν τα νήματα-παραλήπτες να παραλάβουν τα μηνύματα αυτά και να τα επεξεργαστούν. Τα γραμματοκυβώτια χρειάζεται να είναι 2 φορές το πλήθος των συνδέσμων του δικτύου ή αλλιώς να είναι ένα γραμματοκυβώτιο για κάθε σύνδεσμο εξόδου και εισόδου κάθε μεταγωγέα συν το πλήθος των παραγωγών και των καταναλωτών. Συνολικά το πλήθος των γραμματοκυβωτίων είναι

$$(N_i + N_o)N_s + N_p + N_c$$

- max_tables, το μέγιστο πλήθος πινάκων στους οποίους διατηρούνται στατιστικά. Η βιβλιοθήκη CSIM παρέχει μία δομή με την οποία μπορεί ο προγραμματιστής του προσομοιωτή να μαζεύει στατιστικά στοιχεία σχετικά με την προσομοίωση. Αυτά κρατούνται σε δομές που ονομάζονται πίνακες (tables). Το πλήθος των πινάκων που είναι απαραίτητοι για να κρατήσει τα στατιστικά της προσομοίωσης είναι $3N_c$ για τους πίνακες που κρατάνε τα στατιστικά για τις καθυστερήσεις cell πακέτων και πακέτων ανά προορισμό καθώς και ένα μικρό σταθερό πλήθος πινάκων (6) για να κρατούνται στατιστικά σχετικά με μέση καθυστέρηση cell/flit πακέτου επικεφαλίδας κλπ.
- max_facilities, max_servers, το πλήθος των facilities και των εξυπηρετητών τους. Για λόγους πολιτικής, για κάθε πακέτο που ξεκινάει, δημιουργείται επίσης και ένα καινούριο νήμα εκτέλεσης.

Για να μην μεταδίδονται λοιπόν περισσότερα του ενός cell/flit σε κάθε σύνδεσμο εισόδου του δικτύου, οι σύνδεσμοι εισόδου του δικτύου ορίζονται σαν FACILITY, μία δομή που παρέχει το CSIM και μέσω αυτής της δομής μπορούν να συγχρονιστούν τα νήματα−πακέτα για να μεταδίδονται με την σειρά. Το πλήθος τους πρέπει να είναι όσο και το πλήθος των παραγωγών του δικτύου.

- max_events, το μέγιστο πλήθος γεγονότων στον προσομοιωτή. Το CSIM παρέχει την δομή EVENT, τύπος με τον οποίο όταν δηλωθεί μία μεταβλητή μπορεί μέσω αυτής να γίνει συγχρονισμός μεταξύ δύο νημάτων. Τα EVENTs είναι σήματα που μπορεί να ενεργοποιήσει ένα νήμα και να ειδοποιήσει κάποιο άλλο ότι κάτι συνέβει και άρα να το ενεργοποιήσει, αυτό ή και όλα τα νήματα που περίμεναν αυτό το συμβάν. Κατά συνέπεια, ένα νήμα μπορεί να σταματήσει προσωρινά την εκτέλεσή του περιμένοντας ένα γεγονός να συμβεί, και όταν αυτό το γεγονός συμβεί να συνεχίσει την εκτέλεσή του. Νήματα που σε ορισμένες στιγμές περιμένουν κάποιο γεγονός για να συνεχίσουν την εκτέλεσή τους είναι τα credit_collector_code και packet_generator_code και το πως ακριβώς λειτουργεί ο μηχανισμός εξηγείται στην παράγραφο .3.4. Το πλήθος των γεγονότων που χρειάζεται να ορισθούν είναι $3N_pN_c + N_p$ ενώ για τεχνικούς λόγους χρειάζεται να προστεθούν τόσα γεγονότα ακόμα όσο και το πλήθος των γραμματοκυβωτίων.

Στην συνέχεια ανατίθεται χώρος μνήμης στις δομές που την χρειάζονται και το νήμα `sim` περιμένει μέχρι να τελειώσει η προσομοίωση. Επειδή η προσομοίωση τελειώνει με το κριτήριο του πλήθους των πακέτων που έχουν παραδωθεί στους καταναλωτές, υπάρχει ένα γεγονός (το All_finished) που το ενεργοποιεί ο καταναλωτής που παρέλαβε το τελευταίο πακέτο. Τότε όλα τα νήματα τερματίζουν και το `sim` ανακτά τον έλεγχο για τις τελικές λεπτομέρειες που είναι να τυπώσει ορισμένα στατιστικά που δεν κρατούνται σε πίνακες και να τερματίσει την προσομοίωση.

.3.2 Συνάρτηση `Medium`

Στον προσομοιωτή, κάθε κομμάτι πληροφορίας μεταφέρεται με την μορφή μηνυμάτων (messages) που κατευθύνονται στα γραμματοκυβώτια των νημάτων−συστατικών του δικτύου. Τα μηνύματα αυτά παραδίδονται στους προορισμούς τους χωρίς καθυστέρηση. Αυτό δεν είναι αποδεκτό στην δική μας περίπτωση αφού η πληροφορία που μεταφέρεται πάνω στους συνδέσμους, χρειάζεται κάποιον χρόνο για να μεταδοθεί. Για να προσομοιωθεί η καθυστέρηση της πληροφορίας πάνω στους συνδέσμους του δικτύου κάθε μήνυμα δεν μεταδίδεται από αυτόν που το στέλνει κατ' ευθείαν σε αυτόν που το παραλαμβάνει στην άλλη άκρη του συνδέσμου, αλλά το στέλνει στο γραμματοκυβώτιο της Medium.

Η συνάρτηση αυτή παραλαμβάνει όλα τα μηνύματα που στέλνονται κατά την διάρκεια ενός κύκλου μέσα σε όλο το δίκτυο και αφού τα καθυστερήσει κατά μερικά κλάσματα του κύκλου, τα μεταδίδει στον πραγματικό προορισμό τους. Με αυτόν τον τρόπο δεν υπάρχουν μεταφορές πάνω στους συνδέσμους με μηδενική καθυστέρηση και προσομοιώνεται σωστά η καθυστέρησή τους.

.3.3 Συνάρτηση `producer_code`

Σε κάθε είσοδο του δικτύου υπάρχει μία πηγή που παράγει κίνηση, σύμφωνα με ορισμένα σενάρια. Η μορφή της κίνησης εισόδου μπορεί να καθοριστεί στην αρχή της προσομοίωσης, κατά την εισαγωγή των δεδομένων στον προσομοιωτή. Η συνάρτηση `producer_code` αναλαμβάνει να παράγει αυτήν την κίνηση. Ένα νήμα αυτής της συνάρτησης δημιουργείται από την sim για κάθε μία από τις εισόδους του δικτύου. Ένα τέτοιο νήμα−συνάρτηση, αρχικά δημιουργεί ένα νήμα για να συλλέγει τα εισιτήρια που επιστρέφουν στην πηγή (το credit_collector_code για το οποίο θα μιλήσουμε στην παράγραφο .3.5). Κατόπιν επιλέγεται σύμφωνα με την μορφή της κίνησης που προσομοιώνεται, το μέγεθος και ο προορισμός του πακέτου. Κάθε πακέτο που παράγεται είναι και ένα διαφορετικό νήμα. Περισσότερα

του ενός πακέτα μπορεί να μεταδίδονται την ίδια στιγμή από μία είσοδο αλλά μόνο ένα μπορεί να χρησιμοποιεί τον σύνδεσμο εισόδου κατά την διάρκεια ενός κύκλου. Για να μην εμφανίζεται λοιπόν το φαινόμενο, δύο ή και περισσότερα πακέτα να χρησιμοποιούν την ίδια στιγμή (στον ίδιο κύκλο) τον ίδιο σύνδεσμο, χρειάστηκε να ορίσουμε κάποια γεγονότα που χρησιμεύουν για να συγχρονίζονται τα πακέτα μεταξύ τους και χρησιμοποιούν κυκλικά τον σύνδεσμο εισόδου. Γι αυτά τα γεγονότα θα μιλήσουμε αναλυτικά στην παράγραφο .3.4.

.3.4 Συνάρτηση `packet_generator_code`

Δέχεται σαν ορίσματα το μέγεθος και τον προορισμό του πακέτου. Χωρίζει το πακέτο σε μικρότερα κομμάτια, τα cell/flit, και τα μεταδίδει με την σειρά στον πρώτο μεταγωγέα του δικτύου. Ταυτόχρονα συγχρονίζεται και με τα άλλα νήματα−πακέτα που προσπαθούν να μεταδόσουν τα δεδομένα τους στο δίκτυο από τον ίδιο σύνδεσμο εισόδου. Ο συγχρονισμός των νημάτων−πακέτα γίνεται στις ακόλουθες περιπτώσεις:

- ένα cell/flit μπορεί να μεταδοθεί κανονικά (έχει τα απαιτούμενα εισιτήρια) αλλά κάποιο άλλο cell/flit κάποιου άλλου πακέτου μεταδίδεται και καταλαμβάνει τον σύνδεσμο εισόδου του δικτύου. Αυτά τα δύο πακέτα πρέπει να συγχρονιστούν για να μην χρησιμοποιήσουν ταυτόχρονα τον σύνδεσμο. Αυτός ο συγχρονισμός επιτυγχάνεται, χρησιμοποιώντας τις δυνατότητες που παρέχει το CSIM, μέσω του τρόπου με τον οποίο γίνεται ο ορισμός ενός συνδέσμου εισόδου στο δίκτυο. Συγκεκριμένα ο σύνδεσμος εισόδου ορίζεται σαν FACILITY και άρα όταν κάποιο πακέτο θέλει να μεταδόσει πληροφορία πάνω στον σύνδεσμο χρησιμοποιεί την έτοιμη ρουτίνα βιβλιοθήκης `use` και με αυτόν τον τρόπο όποιος άλλος προσπαθήσει να χρησιμοποιήσει τον σύνδεσμο (θα πρέπει και αυτός να κάνει `use`) θα περιμένει μέχρι να τελειώσει την μετάδοσή του ο πρώτος. Όσοι θέλουν να μεταδόσουν δεδομένα, εξυπερετούνται κυκλικά (Round−Robin).

- δεν υπάρχει εισιτήριο για τον επόμενο μεταγωγέα (στην περίπτωση του ATM αυτό το εισιτήριο είναι το εισιτήριο αναχώρησης). Σε αυτήν την περίπτωση συγχρονισμός χρειάζεται μεταξύ των πακέτων που περιμένουν εισιτήριο. Θα πρέπει τα πακέτα που θέλουν να μεταδόσουν cell/flit να περιμένουν μέχρι κάποιο εισιτήριο να επιστρέψει από τον πρώτο μεταγωγέα. Όλα περιμένουν την ενεργοποίηση του γεγονότος (EVENT) της άφιξης εισιτηρίου στην πηγή (το EVENT Leave_next). Όταν ένα εισιτήριο επιστρέψει, το νήμα `credit_collector_code` θα ενεργοποιήσει το γεγονός Leave_next και το πακέτο που περιμένει περισσότερο χρόνο την ενεργοποίηση του συγκεκριμένου γεγονότος, θα αρχίσει να μεταδίδει τα δεδομένα του (εξυπηρέτηση FCFS).

- ένα πακέτο που προορίζεται προς τον καταναλωτή K θέλει να ξεκινήσει από την ίδια πηγή πριν τελειώσει την μετάδοσή του κάποιο άλλο πακέτο που πηγαίνει στον ίδιο καταναλωτή. Αυτή η περίπτωση χρειάζεται να ελεγθεί γιατί για λόγους πολιτικής[2] δεν επιτρέπεται περισσότερα του ενός πακέτα να μεταδίδονται από την ίδια πηγή προς τον ίδιο προορισμό παράλληλα, άρα πρέπει να συγχρονιστούν. Όλα τα πακέτα που περιμένουν να τελειώσει η μετάδοση πακέτου προς τον καταναλωτή που προορίζονται, μπαίνουν σε μία ουρά (ξεχωριστή για κάθε προορισμό) περιμένοντας την ενεργοποίηση του γεγονότος dest_available[dest] όπου dest ο προορισμός τους. Όταν ένα πακέτο τελειώσει την μετάδοσή του, ενεργοποιεί το γεγονός αυτό και επιτρέπει με αυτόν τον τρόπο να αρχίσει την μετάδοσή του το επόμενο πακέτο από την ίδια πηγή, που πηγαίνει στον ίδιο προορισμό.

[2] Αν επιτρέπαμε δύο πακέτα από την ίδια πηγή να μεταδίδονται παράλληλα στον ίδιο προορισμό, αφενός δεν θα κερδίζαμε τίποτα γιατί τα cell/flit των πακέτων αυτών θα πολυπλεχθούν μέσα στο δίκτυο και θα φτάσουν σειριακά στον προορισμό τους, αφετέρου αν υπάρχει συμφόρηση προς τον προορισμό αυτόν, η πολιτική αυτή θα προκαλούσε διόγκωση του προβλήματος.

- ειδικά στην περίπτωση του wormhole συγχρονισμός χρειάζεται μεταξύ των πακέτων από την ίδια πηγή που παρά το ότι είναι έτοιμα να μεταδοθούν, δεν μπορούν να βρούν ελεύθερη λωρίδα για τον πρώτο μεταγωγέα. Αυτό συμβαίνει γιατί στο wormhole υπάρχει περιορισμός στο πλήθος των πακέτων που μπορούν να πολυπλέκονται πάνω σε έναν σύνδεσμο. Όταν ένα πακέτο διαπιστώσει ότι δεν υπάρχει διαθέσιμη λωρίδα για τον επόμενο μεταγωγέα, μπαίνει σε μία ουρά περιμένοντας να ενεργοποιηθεί το γεγονός που υποδεικνύει ότι ένα πακέτο τελείωσε την μετάδοσή του και άρα ελευθέρωσε μία λωρίδα. Όταν κάτι τέτοιο συμβεί, ενεργοποιείται το γεγονός Lane_available και το επόμενο πακέτο που περιμένει λωρίδα, καταλαμβάνει την λωρίδα που μόλις ελευθερώθηκε.

Για λόγους οικονομίας σε εναλλαγές εκτελούμενων νημάτων, όταν μεταδίδεται ένα cell/flit τότε όλα τα phits μεταδίδονται μαζί και η εκτέλεση του νήματος διακόπτεται για την συνολική καθυστέρηση μετάδοσής τους.

Η συνάρτηση `packet_generator_code` κρατάει στατιστικά στοιχεία σχετικά με τον χρόνο αναμονής στις ουρές εισόδου ενός πακέτου. Τον χρόνο δηλαδή που μεσολαβεί από την δημιουργία του πακέτου μέχρι την στιγμή που αρχίζει να μεταδίδεται. Ο χρόνος αυτός προστιθέμενος στον χρόνο που το πακέτο κάνει να περάσει μέσα από το δίκτυο, μας δίνει τον συνολικό χρόνο που το πακέτο ή το cell/flit περιμένει από την στιγμή που δημιουργείται μέχρι την στιγμή που φτάνει στον καταναλωτή.

.3.5 Συνάρτηση `credit_collector_code`

Όπως αναφέρθηκε και παραπάνω, ο ρόλος της credit_collector_code είναι να μαζεύει τα εισιτήρια που επιστρέφουν στην πηγή από τον πρώτο μεταγωγέα του δικτύου. Όταν ένα εισιτήριο επιστρέψει, η συνάρτηση αυτή ανανεώνει τις ενδείξεις στους πίνακες των εισιτηρίων και αν οι προηγούμενες τιμές ήταν 0 αναλαμβάνει να ειδοποιήσει τα πακέτα−νήματα που τυχών περιμένουν την άφιξη εισιτηρίου, ενεργοποιώντας τα κατάλληλα γεγονότα όπως αναφέραμε στην παράγραφο .3.4.

.3.6 Συνάρτηση `switch_code`

Όπως αναφέρθηκε και προηγουμένως, για κάθε μεταγωγέα που υπάρχει στο δίκτυο, δημιουργείται από την συνάρτηση `sim` ένα ξεχωριστό νήμα εκτέλεσης (thread). Κάθε τέτοιο νήμα εκτελεί τον κώδικα της `switch_code`. Το κύριο μέρος της `switch_code` εκτελείται μέσα σε ένα βρόχο. Μέσα σε αυτόν τον βρόχο βρίσκεται ο κώδικας που προσομοιώνει την λειτουργία των εξυπηρετητών του μεταγωγέα. Με λεπτομέρια περιγράφεται η λειτουργία του κάθε εξυπηρετητή αλλά και τα μοντέλα των μεταγωγέων στα κεφάλαια 3.1 και 3.2.

.3.7 Συνάρτηση `consumer_code`

Οι κύριες λειτουργίες που έχει η συνάρτηση `switch_code` είναι να βγάζει την κυκλοφορία από το δίκτυο και να μαζεύει στατιστικά σχετικά με τις εποδόσεις του δικτύου. Και εδώ όπως και στην συνάρτηση `packet_generator_code`, για να γίνεται οικονομία σε εναλλαγές εκτελούμενου νήματος, τα phits που έρχονται από τον τελευταίο μεταγωγέα του δικτύου, μαζεύονται σε ομάδες με μέγεθος όσο και το μέγεθος του cell/flit. Κάθε φορά που μία επικεφαλίδα ενός cell/flit φτάνει σε έναν καταναλωτή, μόλις αυτός την αποσύρει από το δίκτυο, στέλνει και ένα εισιτήριο πίσω στον τελευταίο μεταγωγέα από τον οποίο πήρε το cell/flit. Τα στατιστικά στοιχεία που κρατάει ο κάθε προσομοιωτής είναι τα ακόλουθα:

- καθυστέρηση επικεφαλίδας (μέση και ανά προορισμό). Σαν καθυστέρηση επικεφαλίδας ορίζεται ο χρόνος που μεσολαβεί από την στιγμή που η επικεφαλίδα ενός πακέτου μεταδοθεί από τον παραγωγό μέχρι που παραλαμβάνεται από τον καταναλωτή.

- καθυστέρηση cell/flit (μέση και ανά προορισμό). Είναι η καθυστέρηση που ένα cell/flit χρειάζεται για να περάσει μέσα από το δίκτυο.
- καθυστέρηση πακέτου. Σαν καθυστέρηση πακέτου ορίζεται ο χρόνος που μεσολαβεί από την στιγμή που η επικεφαλίδα ενός πακέτου μεταδοθεί από τον παραγωγό μέχρι που το τελευταίο cell/flit του πακέτου παραληφθεί από τον καταναλωτή.

Κάθε cell/flit περιγράφεται από μια δομή (`msg`) της οποίας ένα πεδίο είναι και ο χρόνος αναχώρησης από τον παραγωγό του cell/flit. Βασιζόμενοι στο γεγονός ότι υπάρχει ένα καθολικό ρολόι του δικτύου, αφαιρώντας από την χρονική στιγμή άφιξης του cell/flit την χρονική στιγμή που αυτό έφυγε από τον παραγωγό, μπορούμε να μετρήσουμε την καθυστέρηση που είχε το cell/flit για να περάσει μέσα από το δίκτυο. Προσθέτοντας σε αυτήν, την καθυστέρηση στις ουρές στην είσοδο του δικτύου, μπορούμε να βρούμε τον συνολικό χρόνο που το cell/flit έκανε για να φτάσει στον καταναλωτή από την στιγμή που δημιουργήθηκε.