2006

Improving Routing Efficiency, Fairness, Differentiated Servises And Throughput In Optical Networks

BIN ZHOU
University of Central Florida

IMPROVING ROUTING EFFICIENCY, FAIRNESS, DIFFERENTIATED SERVISES AND THROUGHPUT IN OPTICAL NETWORKS

by

BIN ZHOU
B.S. Harbin Institute of Technology, 1990
M.S. Harbin Institute of Technology, 1993

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the School of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2006

Major Professor: Mostafa A. Bassiouni

ABSTRACT

Wavelength division multiplexed (WDM) optical networks are rapidly becoming the technology of choice in next-generation Internet architectures. This dissertation addresses the important issues of improving four aspects of optical networks, namely, routing efficiency, fairness, differentiated quality of service (QoS) and throughput.

A new approach for implementing efficient routing and wavelength assignment in WDM networks is proposed and evaluated. In this approach, the state of a multiple-fiber link is represented by a compact bitmap computed as the logical union of the bitmaps of the free wavelengths in the fibers of this link. A modified Dijkstra's shortest path algorithm and a wavelength assignment algorithm are developed using fast logical operations on the bitmap representation.

In optical burst switched (OBS) networks, the burst dropping probability increases as the number of hops in the lightpath of the burst increases. Two schemes are proposed and evaluated to alleviate this unfairness. The two schemes have simple logic, and alleviate the beat-down unfairness problem without negatively impacting the overall throughput of the system. Two similar schemes to provide differentiated services in OBS networks are introduced.

A new scheme to improve the fairness of OBS networks based on burst preemption is presented. The scheme uses carefully designed constraints to avoid excessive wasted channel reservations, reduce cascaded useless preemptions, and maintain healthy throughput levels.

A new scheme to improve the throughput of OBS networks based on burst preemption is presented. An analytical model is developed to compute the throughput of the network for the special case when the network has a ring topology and the preemption weight is based solely on burst size. The analytical model is quite accurate and gives results close to those obtained by simulation.

Finally, a preemption-based scheme for the concurrent improvement of throughput and burst fairness in OBS networks is proposed and evaluated. The scheme uses a preemption weight consisting of two terms: the first term is a function of the size of the burst and the second term is the product of the hop count times the length of the lightpath of the burst.

ACKNOWLEDGEMENTS

I am very grateful to my wife Daisong Sun for her love and support. She always stands behind me whenever I am in need of her support.

I would like to express my sincere gratitude to my advisor, Dr. Mostafa A. Bassiouni, for his inspiration, guidance and patience. I would also like to thank my committee members Dr. Guifang Li, Dr. Ratan K. Guha and Dr. Kien A. Hua for their help and support.

I especially give my thanks to my parents and my parents-in-law for their encouragement for pursuing my Ph.D. degree.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

Wavelength division multiplexed (WDM) optical networks [1-13] are rapidly becoming the technology of choice in network infrastructure and next-generation Internet architectures. WDM networks have the potential to provide unprecedented bandwidth, reduce processing cost, achieve protocol transparency, and enable efficient failure handling. In this dissertation, we investigate four important issues that affect the performance as well as the fairness [14-19] of WDM optical networks. Below, we introduce each of these four issues, survey the related previous results and outline the new approaches that we proposed for each issue.

1.1 Routing and channel assignment

The routing and channel assignment (RWA) problem can be divided into two categories: static and dynamic. The objective of static RWA is to reduce the blocking probability for scenarios in which the connection requests are known in advance. This problem can be formulated as a mixed-integer linear program (ILP) [1, 20], which is NP-complete [21]. In [22], a simple greedy algorithm to find the maximum edge disjoint paths (an NP-hard problem) is used to improve the wavelength usage in the static RWA problem. The second RWA category is the dynamic RWA problem [4, 6, 9, 23] that aims at reducing the blocking probability when connection requests arrive dynamically and are not known in advance. There are two aspects of handling a connection request: i) finding a route from the source to the destination and ii) assigning a wavelength for this connection. Some RWA methods considered these two aspects separately [1, 3, 5]. The layered-graph RWA approach [4, 6, 8] usually has high computational overhead. In

the method described in [6], for example, every wavelength in every fiber is represented by an edge in the layered graphs. This representation poses scalability problems when network links have multiple fibers and the number of wavelengths is large. Some RWA methods consider multi-fiber networks with arbitrary number of wavelengths [7, 24] while others have restriction on the number of fibers and/or wavelengths [2].

In chapter 3, we propose a new dynamic RWA approach based on the logical compact representation of link states. The algorithm uses efficient logical operators for computing the routing reward function. The proposed bitwise routing algorithm combines the benefits of least loaded and shortest path routing algorithms.

1.2 Improving fairness in optical burst switched networks

Optical burst switching (OBS) [30-34] has been proposed to provide fine bandwidth granularity and improve the utilization of WDM networks. Many aspects of burst switching have been extensively investigated including: signaling and reservation protocols [30, 32, 35, 36, 37], routing and scheduling [38, 39], burst assembly at the ingress nodes [40], burst technology and related switch architectures [41], comparison with optical flow switching [42], and contention resolution at the core nodes [43]. There have been also a number of investigations on the issue of fairness in OBS networks. The term “fairness” has been used to address different issues in different types of optical networks such as the capacity unfairness problem [44]. In OBS networks, however, the main type of unfairness arises because the dropping probabilities of optical bursts traveling through paths with large hop count tend to be higher than those whose

paths have a small number of hops. This type of unfairness is well known in electronic packet switched networks and is often referred to as the "*beat down*" problem. This problem has been studied in several papers as a secondary consideration in the evaluation of RWA and other optical algorithms [29, 45, 46, 48, 63]. In [48], an OBS reservation scheme is proposed for OBS networks operating under the wavelength continuity constraint (i.e., no converters). The scheme uses the backward reservation paradigm in which a probe control message propagates from the source to destination then a reservation message travels back from the destination to the source before the data burst can be transmitted. The simulation results reported in [48] uses a three-node tandem network (i.e., maximum of two hops) and a total of three connections (two 1-hop and one 2-hop). The results show that fairness for the 2-hop connection has improved at the expense of a slight increase in the overall mean blocking probability of the three connections. The authors in [45] proposed a deflection routing algorithm and showed that the blocking probabilities of bursts with various hop count at high load levels are almost the same as the case of no deflection, i.e., this deflection routing does not aggravate the unfairness problem for bursts with large hop count.

In Chapter 3, we investigate the unfairness encountered by bursts traveling through long lightpaths and we propose and evaluate the performance of two schemes for alleviating this type of unfairness. The first scheme, Balanced JIT, uses a simple equation to adjust the size of the search space for a free wavelength based on the number of hops traveled by the burst. The BJIT scheme has a single parameter and is implemented in each OXC. The second scheme, PRED, uses proactive discarding to reduce the probability of dropping bursts with large hop count at the expense of an increase in the dropping probability of bursts with small hop count.

In Chapter 5, we propose a preemption-based scheme for improving fairness. The preemptive weight of a burst at a node is the product of the current hop count of this burst and the length of its lightpath. The control packets of bursts with higher preemptive weight are allowed to preempt the channel resources reserved earlier for bursts with lower preemptive weight. The scheme uses carefully designed constraints to avoid excessive wasted channel reservations and reduce cascaded useless preemptions.

1.3 Quality of service differentiation in optical burst switched networks

In real life applications with differentiated QoS requirements, data bursts should have different priority classes. The growing interest in introducing QoS differentiation in Internet services is motivated by the need to improve the quality of support for IP voice and video services, and in general, by the desire to provide clients with a range of service-quality levels at different prices. In [58], differentiated Quality of Service has been incorporated into Just-Enough-Time (JET) scheduling [30] by assigning different offset times to different classes. In [75], QoS is supported by adjusting the offset time and the lower and upper bounds on blocking probability for two burst classes are analyzed. Burst assembly at the ingress nodes [40] is another method to improve QoS in JET. In this method, the window size and weight of a class determines the number of packets in the window. In [59], the QoS differentiation in JET is investigated by analyzing the loss probabilities of two classes of bursts. As was done in [58], the authors in [59] also focus on changing the offset time to support differentiated QoS. In [60, 69], a burst segmentation method is proposed to address the differentiated QoS problem. Bursts with higher priority can preempt the overlapping segments of lower priority bursts and the preempted segments are dropped or are

deflected to alternate routes. In [61], a proportional model is proposed to enhance the offset-based QoS differentiation method proposed in [58]. In the proportional model, the differentiation of a particular QoS metric can be quantitatively adjusted to be proportional to the factors that a network service provider sets. The lower priority burst is intentionally dropped when the proportional differential model is violated. In [74], the authors combine a service differentiation model based on proportional resource allocation with a partially preemptive burst scheduler. The scheme improves the utilization while providing QoS with controllable service differentiation. In [62], the authors propose a differentiation scheme for JET that does not assign extra time to a higher priority class as in [58]. Rather it uses a priority queuing technique to schedule a higher priority burst earlier than a lower priority burst. Another proportional differentiation model is used in [68]. In this model, the burst control packets are queued in increasing order of the burst preferred scheduling time (defined as a burst arriving time minus its differential time). Each OXC chooses its own differential time function according to its resource availability and QoS requirement. In [63], some undesired characteristics of the offset-time management mechanism for JET are identified. The authors found that the burst drop probability differentiation attained for a given offset-time value strongly depends on the distribution of the burst durations and that controlling the differentiation is difficult. In [64], assured Horizon is introduced for a coarse-grained bandwidth reservation r_i for every forwarding equivalent class (FEC) between ingress and egress. The burst assembler marks bursts as compliant and non-compliant bursts, depending on whether the burst is conforming to r_i. The non-compliant bursts are dropped when congestion occurs. In [70], a generalized Latest Available Unused Channel with Void Filling (LAUC-VF) algorithm is proposed. The LAUC-VF algorithm aims at providing good performance by essentially choosing the wavelength with the smallest possible available window, leaving larger

windows for control packets that arrive later. LAUC-VF is basically another scheduling scheme for OBS networks and is used in [70] to support differentiated services with limited buffers. In [65], an algorithm is proposed to decide about the value of the burst offset-time in JET based on the burst priority class. In [67], preemptive multiclass wavelength reservation is used to provide differentiated services for the JET protocol. The preemption scheme provides QoS differentiation but complicates the control logic in OXC's. In [78], the authors propose QoS-guaranteed wavelength allocation schemes for WDM networks. In their schemes, the wavelengths are classified into different sets based on the QoS requirement and higher priority requests can be allocated more wavelengths. In each set, there are two rules to select the idle wavelength: minimum index numbered or maximum index numbered wavelength. The authors in [78] showed that the connection loss probability for higher priority requests is improved but the overall throughput performance of their schemes is not discussed.

In Chapter 4, we propose two new schemes to support QoS differentiation in OBS networks. The first scheme adjusts the size of the search space for a free wavelength based on the priority level of the burst. The second scheme uses different proactive discarding rates in the network access station of the source node. Both schemes are capable of providing tangible QoS differentiation without negatively impacting the throughput of OBS networks.

1.4 Improving throughput in optical burst switched networks

There are two approaches to assemble the bursts in the network access station (NAS) of OBS networks: timer-based and threshold-based. In the timer-based burst assembly approach, the

formation of the burst is restricted by a maximum (or periodic) time interval [80] and therefore bursts tend to have variable lengths. In the threshold-based approach [81], a limit is used to restrict the size of the burst (or the number of packets in the burst) and therefore bursts tend to have equal, or nearly equal, length. The timer-based approach is more practical for delay sensitive applications and it is also the approach most suitable for throughput improvement using preemption.

In [49], the authors proposed a merit-based scheduling algorithm for OBS networks. The scheme uses the route length and the current value of hop count to decide the preemption weight (priority) of a burst. Bursts with larger weight can preempt the channel resources reserved by bursts with smaller weight. The authors in [49] reported that the blocking probability is reduced by the merit-based scheme. Another preemption-based scheme is proposed in [74] based on proportional service differentiation. In [85], preemption is used as one of the approaches to handle channel contention and a model for analyzing preemptive burst segmentation is developed. Priority-based preemption is also used in [86], and in [73] a probabilistic preemptive scheme is proposed, mainly for the purpose of providing service differentiation.

In Chapter 6, we propose a preemptive scheme to improve the throughput of OBS networks. In this scheme, the preemptive discipline is based on size of the burst size. The burst with larger size can preempt the resources reserved by some other bursts that have smaller burst size. Since preemption incurs overhead and waste of resources, a threshold on the size of the burst and some other design details are used to constrain the rate of preemption and ensure positive impact on

throughput. An analytical model is developed to compute the throughput of the ring topology in the presence of preemption.

In Chapter 7, we propose and evaluate a new preemption-based scheme for the concurrent improvement of network throughput and burst fairness in OBS networks. The scheme uses a preemption weight consisting of two terms: the first term is a function of the size of the burst and the second term is the product of the hop count times the length of the lightpath of the burst. The second term is adjusted by a minimum function to prevent the problem of reverse unfairness.

1.5 Organization of Dissertation

This dissertation is organized as follows. In Chapter 2, the bitwise scheme for WDM optical networks is presented. The fairness problem in OBS networks is addressed in Chapter 3. In Chapter 4, two QoS differentiation schemes for OBS networks are presented. In Chapter 5, an advanced scheme to improve fairness in OBS networks based on constrained preemption is proposed. We introduce a preemption-based scheme to improve throughput in Chapter 6, and we also present an analytical model for computing throughput iteratively for the special case of ring topology. In Chapter 7, we present a preemption-based scheme for the concurrent improvement of throughput and fairness. Finally, the conclusion of this dissertation and future work are presented in Chapter 8.

CHAPTER 2: ROUTING AND WAVELENGTH ASSIGNMENT IN OPTICAL NETWORKS USING LOGICAL LINK REPRESENTATION AND EFFICIENT COMPUTATION

2.1 Introduction

Wavelength division multiplexing (WDM) in optical networks has received considerable attention in recent years [1-13]. In spite of good advancement in the design and implementations of WDM optical networks [14], there are several challenging problems facing the design of these networks, e.g., improving network robustness, increasing the efficiency of routing and wavelength assignment, enhancing the flexibility of network dimensioning, and improving fairness [14-19]. In this chapter, we address the issue of routing and wavelength assignment (RWA) and present an efficient RWA algorithm based on the logical link representation and fast bit-wise computation.

The RWA problem can be divided into two categories: static and dynamic. The objective of static RWA is to reduce the blocking probability for scenarios in which the connection requests are known in advance. This problem can be formulated as a mixed-integer linear program (ILP) [1, 20], which is NP-complete [21]. In [22], a simple greedy algorithm to find the maximum edge disjoint paths (an NP-hard problem) is used to improve the wavelength usage in the static RWA problem. The second RWA category is the dynamic RWA problem [4, 6, 9, 23] that aims at reducing the blocking probability when connection requests arrive dynamically and are not

known in advance. There are two aspects of handling a connection request: i) finding a route from the source to the destination and ii) assigning a wavelength for this connection. Some RWA methods considered these two aspects separately [1, 3, 5]. The layered-graph RWA approach [4, 6, 8] usually has high computational overhead. In the method described in [6], for example, every wavelength in every fiber is represented by an edge in the layered graphs. This representation poses scalability problems when network links have multiple fibers and the number of wavelengths is large. Some RWA methods consider multi-fiber networks with arbitrary number of wavelengths [7, 24] while others have restriction on the number of fibers and/or wavelengths [2].

In this chapter, we propose a new dynamic RWA approach based on the logical representation of link states and using efficient logical operators for computing the routing reward function. The remainder of this chapter is organized as follows. Section 2.2 introduces the model of WDM optical networks and the logical bitmap representation used in our RWA algorithm. Section 2.3 presents the modified Dijkstra algorithm using the logical bitmap representation and the channel assignment algorithm. Section 2.4 describes the simulation model and the routing methods used in our comparison tests. Performance results are given in Section 2.5.

2.2 The Network Model

We consider multi-fiber multi-wavelength optical networks and primarily focus on the RWA problem under the constraint of wavelength-continuity. When a connection request is received at

the destination NAS (network access station), the network establishes a lightpath comprising a number of optical cross-connects (OXCs) and assigns a single wavelength for data forwarding in this lightpath. Although we do not consider wavelength conversion in this chapter, the availability of wavelength converters can be easily incorporated in our approach. The benefit of wavelength conversion can be obtained even when the converters are deployed in only a small number of nodes. For example, [25-27] have shown that the presence of full or limited wavelength conversion capability in a sparse set of nodes selected by a *k-minimum dominating set* heuristic can lead to significant gains in performance. For the sake of clarity and simplicity, however, we shall illustrate the main idea of our RWA approach using OXCs without wavelength converters. Also for the simplicity of illustration, we assume that connections are two-way (duplex) connections. This means that the RWA algorithm can assign the same wavelength to both directions of the connection and the graph representation of the network becomes an undirected graph. Generalizing our RWA algorithm to handle one-way (simplex) connections using directed graphs is straightforward and will not be included in this chapter.

2.2.1 Logical Bitmap Representation

In our model, the optical network is represented by a weighted undirected graph $G=(V, E)$, where V is the set of nodes (vertices) and E is the set of edges. The set V corresponds to the set of OXCs that form the nodes of the optical network. An edge between a pair of nodes in G represents a link between the corresponding pair of OXCs in the optical network. Let $v1$ and $v2$ be two adjacent vertices in G representing two OXCs in the optical network. The link between these two vertices has m fibers denoted $f_1, f_2, \ldots, f_m$. Each fiber can carry data using n wavelengths,

where $n=|W|$ and W is the set $\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ of operational wavelengths in the network. For simplicity of presentation, we shall assume that all links in the network have the same number of fibers, m, and each fiber is provisioned to use all n wavelengths. The n-bit integer variable *fiber_state[v1,v2][j]* is used to keep track of the free channels (free wavelengths) in fiber f_j of the link connecting nodes *v1* and *v2*. The kth bit in this variable is set to 1 if the kth wavelength is free and is set to 0 if this wavelength is used. We adopt the notation that the lowest wavelength ID value corresponds to the least significant bit, i.e., the channel index of a bit increases as the significance of this bit increases. The n-bit variable *Link_state[v1,v2]* is used to keep track of the free wavelengths in the link between *v1* and *v2* and is computed as follows:

$$Link\_state\ [v1,v2] = \bigcup_{r=1}^{m} fiber\_state[v1,v2,][r]$$

where $\cup$ indicates the logical union (bitwise OR) operation. A bit value of 0 in *Link_state* indicates that the corresponding channel is used (i.e., is not free) in all fibers of that link. A bit value of 1, on the other hand, indicates that there is at least one fiber in which the channel is free. For example,

Let m = 3 fibers and n = 5 wavelengths. The link between *v1* and *v2* has the following values

fiber_state[v1,v2][1] = 01011 $\rightarrow \lambda_1, \lambda_2, \lambda_4$ are free in fiber 1

fiber_state[v1,v2][2] = 00110 $\rightarrow \lambda_2, \lambda_3$ are free in fiber 2

fiber_state[v1,v2][3] = 00111 $\rightarrow \lambda_1, \lambda_2, \lambda_3$ are free in fiber 3

The state of the link is given by

$$Link\_state\ [v1,v2] = 01011 \cup 00110 \cup 00111 = 01111$$

The binary value *01111* indicates that a new connection passing through this link can use any of the four free channels λ_1, λ_2, λ_3, λ_4 but cannot use λ_5.

The above logical representation of *Link_state* does not miss any free wavelength in any link but also does not record how many fibers in each link have the free channel. It turns out that this compact logical representation has two major benefits. First, the bitwise representation significantly reduces the storage requirement of the routing algorithm and improves its speed. Second, applying this representation in the computation of the routing cost (reward) function yields a scheme that is a nice compromise between least loaded routing algorithms and shortest path routing algorithms. Our simulation results have shown that this bitwise routing scheme gives very good performance with low blocking probability and can be used satisfactorily to replace the more complex mathematical representation (to be defined in Section 2.3.3).

2.2.2 Logical Reward/Cost Function for Routing Algorithm

Routing algorithms use a cost function (or reward function) to compare the different paths and select the best one. Our RWA method uses a reward function based on the logical representation defined in Section 2.2.1. Let *Count_one_binary(X)* be a function that returns the number of 1-valued bits in the integer value *X* (or alternatively in the binary string representing *X*). The argument *X* for a path is an n-bit integer value obtained from the logical intersection (bitwise AND) of the *Link_state* values of the individual links of this path. The reward function used in

our algorithm is the value returned by the function *Count_one_binary(X)*. Higher reward values indicate higher number of free wavelengths that can be used to serve new connections. Our RWA method gives preference to routes having higher reward values. If two routes have equal reward values, the RWA algorithm selects the route having a smaller length.

2.3 Routing and Wavelength Assignment

2.3.1 Routing Based on Bitmap Representation

Many routing algorithms are based on Dijkstra algorithm [3]. We have the modified Dijkstra algorithm to use the bitwise logical computation described in Section 2.2.1. The modified algorithm uses the reward function defined in Section 2.2.2 in order to find a route that has larger number of free wavelengths as indicated by the bitmap representation of *Link_state*. If there is a tie, the algorithm selects the route having the smallest hop count.

Figure 1 gives the pseudo code of the modified Dijkstra algorithm using the logical representation and the reward function presented in Section 2.2. The operator "$\cap$" in Figure 1 denotes the logical intersection (bitwise AND) operation and the operator "•" denotes string concatenation. The inputs of the modified Dijkstra algorithm are: source node *S,* destination node *D*, the array *Fiber_state* that records the free channels in each fiber and the array *Link_state* that records the free wavelengths in each link. The algorithm returns the selected path from *S* to *D*,

the length of this path and the free wavelengths that can be used in this path. The data structures used by the algorithm are as follows. For each node *v*,

Route[v] is a string that stores the route selected so far from source *S* to node *v*.

Hop_count[v] gives the length (i.e., number of hops) of *Route[v]*.

Avail[v] stores the bitmap representation of the free wavelengths that can be used in *Route[v]*.

Weight[v] is the reward value for the path from S to v specified by *Route[v]*.

```
Modified Dijkstra Algorithm using Logical Representation:
Step 1:  Initialization
        Set current node C = S
        Avail[C] = 2^n – 1    /* This is a binary string of n 1's, i.e., all n channels are free */
        Route[C] = "S"
        Hop_count[C] = 0
        Insert all nodes of the graph except node S into the set L.
        Mark all members of L as unvisited

Step 2: Adjust the path for every neighbor of the current node C.
        For every neighbor of current node C, say v, which is a member of L, do
        {
        If (v is unvisited) {
                /* first time to reach v */
        Route[v]=Route[C] • "v"
        Hop_count[v] = Hop_count[C] + 1
        Avail[v] = Avail[C] ∩ Link_state(C,v)
        Weight[v] = Count_one_binary(Avail[v]) /* Initial reward value for the path from S to v */
        Mark v as visited    }
        else  { /* v has been visited before; a new path to v has now been found via node C */
                TempAvail = Avail[C] ∩ Link_state(C,v),          /* Evaluate the new path to v */
        TempWeight = Count_one_binary(TempAvail)    /* and compute its reward value */
        TempHop_count = Hop_count[C] + 1
        if (TempWeight > Weight[v] OR
                TempWeight = Weight[v]  AND TempHop_count < Hop_count[v])
                        {  /*  better path from S to v has been found */
                         Route[v]=Route[C] • "v"    /* Update the parameters based on the new path */
                        Hop_count[v] = TempHop_count
                        Avail[v] = TempAvail
                        Weight[v] = TempWeight   }
    }
Step 3: /* Find next current node */
        If (L is not empty)
        {   /* select new current node C */
        Select C as  the member of L, say member v,  having the largest value of
                    Weight[v] and
                      if there is a tie, having the smallest value of Hop_count[v] and
                      if there is a tie, having the smallest ID.
        Remove C from L
        Go to Step 2  }
Step 4: The set  L is now empty and the search is completed
        If    (Avail[D] = 0)
        { Failure: there is no path with free channels from S to D}
        else       { return (Route[D], Hop_count[D], Avail[D]  )
```

Figure 2.1. Modified Dijkstra algorithm using logical representation and reward function

2.3.2 Channel Assignment Based on Logical Representation

In [3], Hui Zhang et al observe that first-fit is a reasonably good channel assignment algorithm compared to other methods, e.g., random assignment, SPREAD (least-used assignment), and PACK (most-used assignment). Our simulation tests have also supported this observation. It is easy to explain this observation by noting that, under the constraint of wavelength continuity, the blocking probability (BP) for a lightpath can be estimated as

$$BP = \frac{\text{Number of used wavelengths in the lightpath}}{\text{Total number of wavelengths in the lightpath}}$$

For the first-fit algorithm, the number of used wavelengths will be generally smaller than that of the random assignment algorithm. This is because the first-fit algorithm always uses wavelengths having smaller ID values. This frees upper wavelengths with higher ID values and enhances the chance of finding a wavelength that is free in all the links of a given lightpath. For example, if $W = \{\lambda_1, \lambda_2\}$ and there is a request for a one-hop connection between nodes *v1* and *v2* and another request for one-hop connection between *v2* and *v3*, then the first-fit scheme will assign λ_1 to both connections leaving λ_2 free. The random scheme may however assign λ_1 to one connection and λ_2 to the other connection, thus blocking a future request for a two-hop connection between *v1* and *v3*. This means that under the wavelength continuity constraint, a request for a new connection is more likely to be accepted in the first-fit scheme than under random wavelength assignment.

We have used the first-fit assignment as follows: each wavelength is assigned an ID in the range 1 through *n* and each fiber within a link is assigned an ID in the range 1 to *m*. When the modified Dijkstra algorithm of Figure 1 succeeds in finding a path, the first-fit channel assignment strategy is used to select the free wavelength with the smallest ID. Once the wavelength is selected, we next select the fiber with the smallest ID in which this wavelength is free. All the information needed to perform the first-fit strategy is contained in the link/fiber state information and in the results returned by the routing algorithm of Figure 1. Figure 2 gives the pseudo code of the bitwise first-fit channel assignment algorithm. The input parameters *Path* and *Avail_channels* of this algorithm are obtained from the results *Route* and *Avail* returned by the modified Dijkstra algorithm of Figure 1. The code in Figure 2 selects a wavelength for the new connection, then for each link in the lightpath, it selects a fiber in which this wavelength is free. The algorithm also adjusts *Fiber_state* of the selected fibers and *Link_state* of all links in the lightpath.

```
Channel Assignment (Path, Avail_channels)
Select a channel index k such that k is the smallest positive integer satisfying
                    (2^(k-1) ∩ Avail_channels) ≠ 0    /* Channel k is a free channel with lowest ID */
For j=1 to length(Path)-1 do  { /* Process one link at a time */
        v1 = j^th element of Path
        v2 = (j+1)^th element of Path
        select  a fiber index  i such that i is the smallest integer satisfying
                Fiber_state[v1,v2,][i] ∩ 2^(k-1) ≠ 0    /* Fiber i has channel k free and has lowest ID */
        Fiber_state[v1,v2,][i] = Fiber_state[v1,v2,][i] - 2^(k-1)  /* Adjust status of channel k in fiber i */
        Link_state [v1,v2] = ⋃_{r=1}^{m} Fiber_state[v1,v2,][r]  }  /* Re-compute the state of this link */
```

Figure 2.2. Channel assignment based on the logical representation

2.3.3 Mathematical Representation

The efficient logical representation used in the algorithms presented in Sections 2.3.1 and 2.3.2 can be replaced by a more complex mathematical representation. Recall from the example given in Section 2.2.1, the link state was obtained by the bitwise computation

$$Link\_state\ [v1,v2] \ = \ 01011 \cup 00110 \cup 00111 = 01111$$

The binary value *01111* indicates that a new connection can use any one of the four free channels λ_1, λ_2, λ_3, λ_4 but cannot use λ_5. The corresponding mathematical representation would give

$$Link\_state\ [v1,v2] \ = [0,1,2,3,2]$$

The variable *Link_state [v1,v2]* is now an array of five integer values rather than a single 5-bit integer value. This array indicates that the first channel λ_1 is free in two fibers of the link between *v1* and *v2*, the second channel λ_2 is free in three fibers, the third channel λ_3 is free in two fibers, the fourth channel λ_4 is free in only one fiber and the fifth channel λ_5 is used (not free) in all fibers of the link between *v1* and *v2*.

Let each of A_1 and A_2 be an array of *n* integers and let α_{1i} and α_{2i} be the ith member of A_1 and A_2 respectively, $1 \leq i \leq n$ where *n* is the number of operational wavelengths used in the network. The logical union operator $\cup$ and logical intersection operator $\cap$ used earlier (see Figures 2.2.1 and 2.2.2) are now replaced by the mathematical operators $\oplus$ and $\otimes$, respectively. These latter operators are defined as follows.

Math_Sum $\oplus$: $P=A_1 \oplus A_2$

P is an array of n elements p_i, $1 \le i \le n$ where $p_i = \alpha_{1i} + \alpha_{2i}$

Math_Min $\otimes$: $P = A_1 \otimes A_2$,

P is an array of n elements p_i, $1 \le i \le n$ where $p_i = min\{\alpha_{1i}, \alpha_{2i}\}$

The algorithms of Figures 2.1 and 2.2 can be converted to use the mathematical operators $\oplus$ and $\otimes$ by making all appropriate changes in the different routines, e.g., definition and computation of the reward function, comparisons and selection of routes, wavelength assignment and adjustment of the link state values. We have implemented a mathematical version of the RWA method of Sections 2.3.1 and 2.3.2 and included it in our comparison tests that will be discussed in Section 2.4.

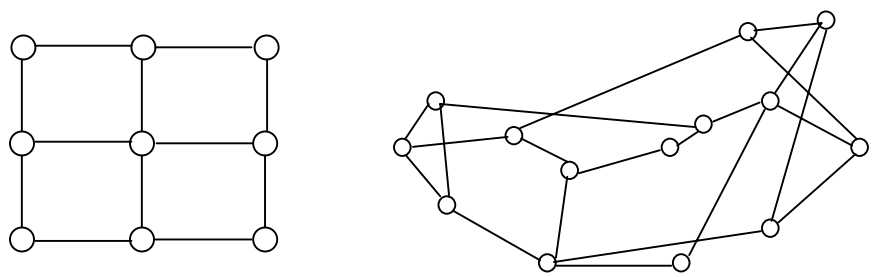
It is important to notice that the Modified Dijkstra algorithm using the logical representation (denoted as DLR) behaves differently than the modified Dijkstra algorithm using the mathematical representation (denoted as DMR). Specifically, DMR is a strict least loaded (i.e., least congested) algorithm. DLR, however, may sometimes select a shortest path route that is not a least loaded path. For example, consider the case of four fibers (m = 4) and five wavelengths (n = 5). Assume that there are two alternative paths from node S to node D, say path P1 and P2 and that P2 is shorter than P1. Wavelengths # 1, 3, 5 are free in all four fibers of P1 but are free in only two fibers in P2. Wavelengths # 2 and 4 are used in all fibers in both P1 and P2. In DLR, the value of *Avail[D]* for both P1 and P2 is equal to the bit string "10101" which has a reward value of 3. Therefore DLR will use the hop count to break the tie and will choose the shorter path P2. In case of DMR, the value of *Avail[D]* for P1 is equal to the array of integers

[4,0,4,0,4] with a reward value of 12 while that of P2 is [2,0,2,0,2] with a reward value of 6. So DMR chooses P1 which is the least loaded path but DLR chooses P2 which is the shortest path.

The above discussion shows that DLR is not a strict "least loaded" routing algorithm and may in fact choose the shortest path rather than the path with the largest number of free wavelengths.

2.4 The Simulation Model

Figure 2.3 shows the two network topologies used in our tests: (a 9-node mesh network and the well-known 14-node NSFNET network).



3x3 Mesh network NSFNET network

Figure 2.3. Two network topologies

Connection requests arrive according to a Poisson distribution with arrival rate λ. The connection holding times are negative exponentially distributed with mean $1/\mu$. For each request, the source

node and destination node are randomly selected. The traffic load ρ for the network is λ/μ. The tests covered a variety of parameter values: the arrival rate λ changed from 5 to 30 in an increment of 1, the mean connection holding time $1/\mu$ changed from 5 to 50 in an increment of 1, the number of wavelengths was 8, 16, or 24 and the number of fibers in each link was 2, 3 or 4. Each simulation result was gotten from the average of six scenarios and a total of 100,000 requests were generated in each scenario. The simulation environment is Sun sun4u E4500/E5500, System clock frequency: 100 MHz, Memory size: 7168MB, CPU: 400MHz and cache: 4MB.

Our extensive simulation tests have compared the following methods:

Fixed shortest-path routing method (FSR):

In this method [3, 6], the connection is blocked if it cannot be serviced by the shortest path from the source to the destination. If the shortest path has free wavelengths, a first fit wavelength assignment is used to select the channel for the new request. Otherwise, the request is blocked.

Modified Dijkstra algorithm using Logical Representation (DLR):

This method uses the logical representation and reward function presented in Section 2.2. The pseudo code of the DLR algorithm was given in Figure 2.1.

Modified Dijkstra algorithm using Mathematical Representation (DMR):

This method uses the mathematical representation and reward function presented in Section 2.3.3.

Modified Flooding algorithm using Logical Representation (FLR):

This method finds the shortest path between the source and destination using the flooding approach. We used the logical representation of Section 2.2 for keeping track of free wavelengths (as was done in the DLR method). We assumed the routing search is terminated once the destination is reached; thus FLR is an ideal flooding scheme that always selects the shortest path but has high signaling overhead.

Buffered methods (BFSR, BDLR, BDMR, BFLR):

In all of the four previous methods, the request for a new connection is immediately discarded if the routing algorithm fails to find a lightpath with a free wavelength to serve this request. In the buffered methods (denoted by BFSR, BDLR, BDMR, and BFLR), we assume there are some electronic buffers of finite size in the source NAS. If a request is blocked, it can be held in these buffers in the hope that a wavelength/lightpath would become available to serve the blocked request soon. Requests waiting in the buffers are discarded if they do not get served before their maximum waiting time expires. In our tests, we used buffers that can store a maximum of 5 requests and we assumed that the maximum waiting time of each request in the buffer is 10 units of time.

2.5 Performance Results

We first discuss the comparison results for the four methods: FSR, DLR, DMR and FLR when two fiber pairs are used per each link (i.e., two fibers in each direction). Figures 2.4 and 2.5 show

the results for the NSFNET topology with 16 and 24 wavelengths, respectively. The corresponding results of the Mesh topology for these two scenarios are presented in Figures 2.6 and 2.7. The results of buffered methods (using five electronic buffers in each source NAS) are given in Figure 2.8 for the NSFNET topology (results for the Mesh topology have similar trends and are omitted). In Figure 2.9, the performance results of NSFNET are compared at a load of 200 for the methods with/without buffers. Figure 2.10 compares the routing speed of the bitwise DLR scheme with that of its mathematical DMR counterpart for the NSFNET topology. The routing speeds are the average running time for each routing process in DLR and DMR. From all these results (Figures 2.4-2.10), we observe the following:

DLR and DMR give very close blocking performance (DLR is usually better). DLR is much faster than DMR and has less storage overhead. This confirms that our proposed logical representation and the bitwise computations used in DLR are not only advantageous in terms of storage requirement and computational overhead, but are also very competitive in terms of improving performance and reducing the blocking probability. It is important to notice that in real-time dynamic routing environments, faster algorithms are highly desirable even at the expense of some loss of performance. Not only the bitwise DLR scheme is faster than DMR, but it is even slightly better in performance.

The blocking performance of DLR is superior to that of the fixed shortest-path routing (FSR) method. This result can be explained by noting that our DLR method uses a reward function that favors the less loaded routes. Thus DLR has benefits similar to those of least-loaded routing algorithms [28]. Li and Somani showed in [28] that least loaded routing can usually achieve

better blocking performance than the fixed alternate routing (FAR) method described in [29]. The reason is that least loaded routing tends to spread the load on different links but is likely to use longer paths. Similarly, our DLR scheme usually achieves better performance than fixed shortest-path routing (FSR) again because of load spreading. Recall, however, that DLR is not a strict least loaded algorithm and, in some scenarios, it may select the shortest path rather than the least loaded path as was explained in Section 2.3.3. It should be noted here that when the number of wavelengths is small (Figure 2.6), the difference between all methods is small since the small number of wavelengths induces congestion more easily and the blocking performance deteriorates rapidly for all methods.

DLR has also better blocking performance than the ideal flooding routing scheme (FLR). Since FLR is a strict shortest path algorithm, it tends to congest the links of the shortest paths at higher loads and its blocking performance is negatively impacted. Our DLR scheme, on the other hand, is a hybrid combination of least-loaded and shortest-path algorithm and often selects less congested routes than FLR, thereby leading to an improved blocking performance.

As expected, the use of buffers at the source NAS improves the blocking performance. Comparing Figure 2.4 and Figure 2.8, the blocking probability is reduced when the source NAS is equipped with electronic buffer. This is also clearly illustrated in Figure 2.9. During congestion situations and when the number of wavelengths is small, the buffer gives only small improvement. This is because the persistence of congestion in these situations may cause the request to get discarded even if it is buffered for some duration. The advantage of buffering is significant at medium loads and when larger number of wavelengths is used.

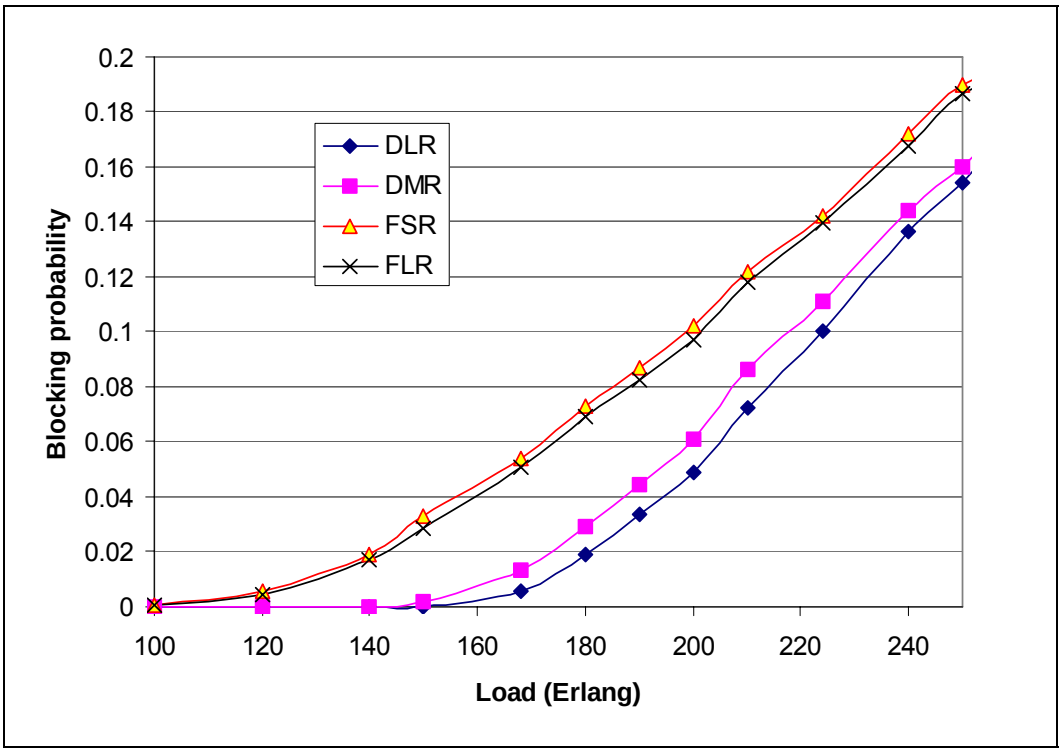


Figure 2.4. Blocking probability vs traffic load. Wavelengths = 16, fibers = 2. NSFNET network

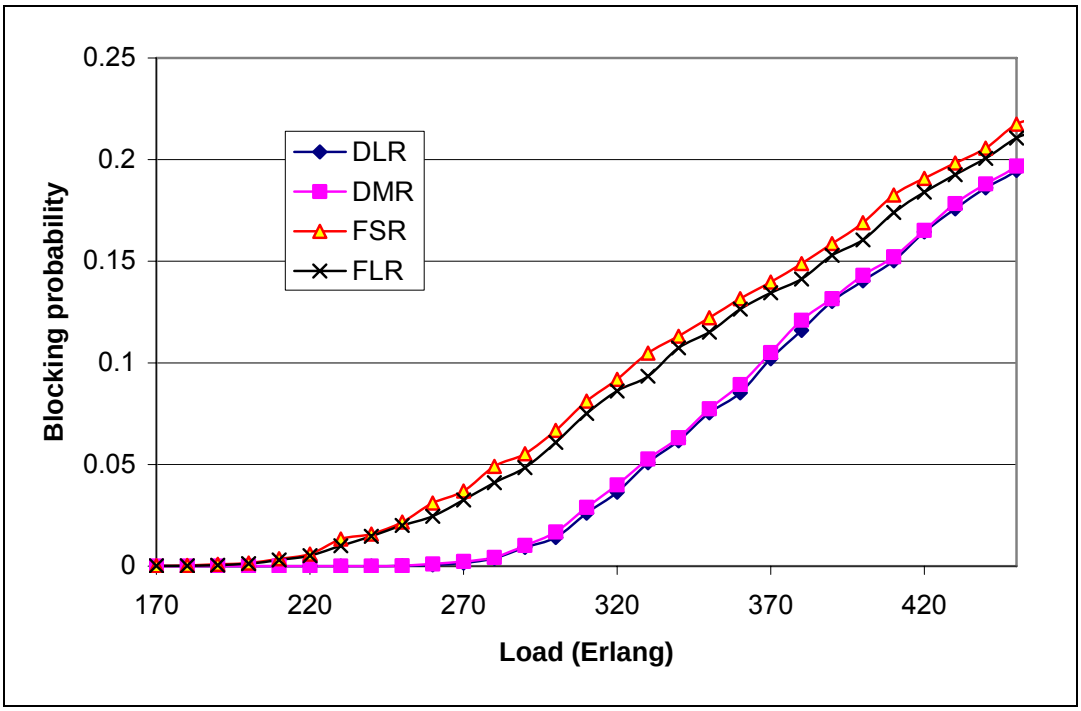


Figure 2.5. Blocking probability vs traffic load. Wavelengths = 24, fibers = 2. NSFNET network

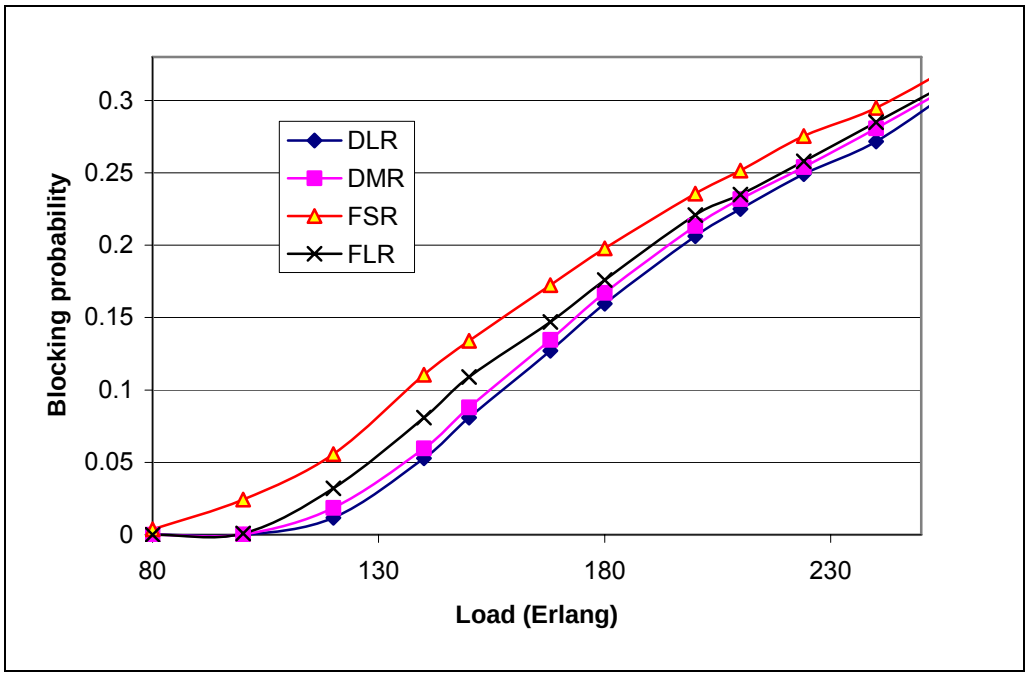


Figure 2.6. Blocking probability vs traffic load. Wavelengths = 16, fibers = 2. Mesh network

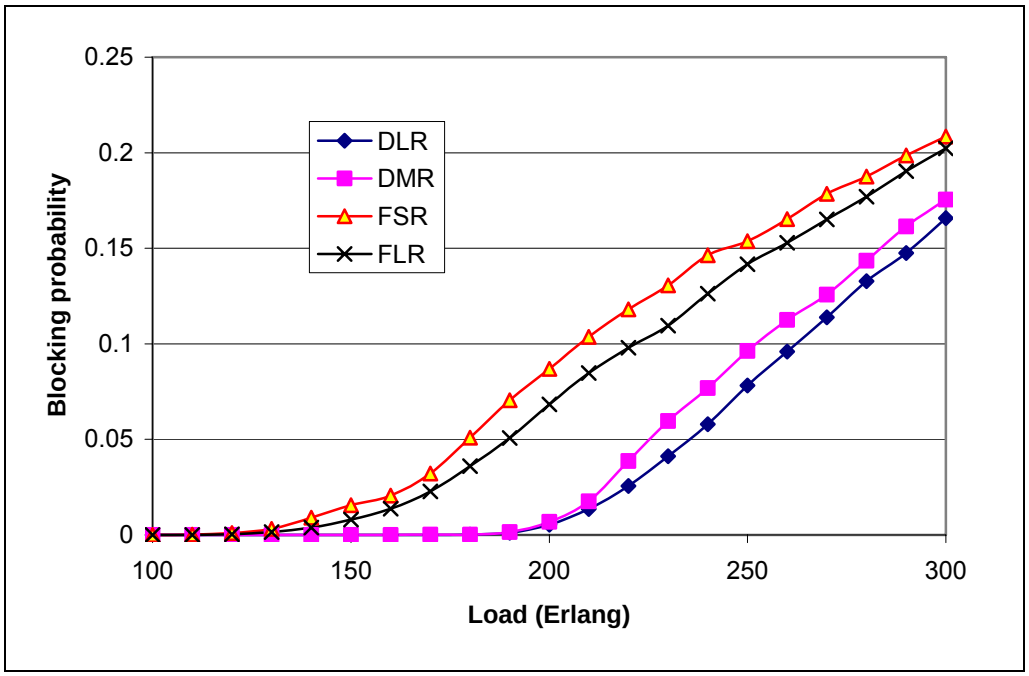


Figure 2.7. Blocking probability vs traffic load. Wavelengths = 24, fibers = 2. Mesh network

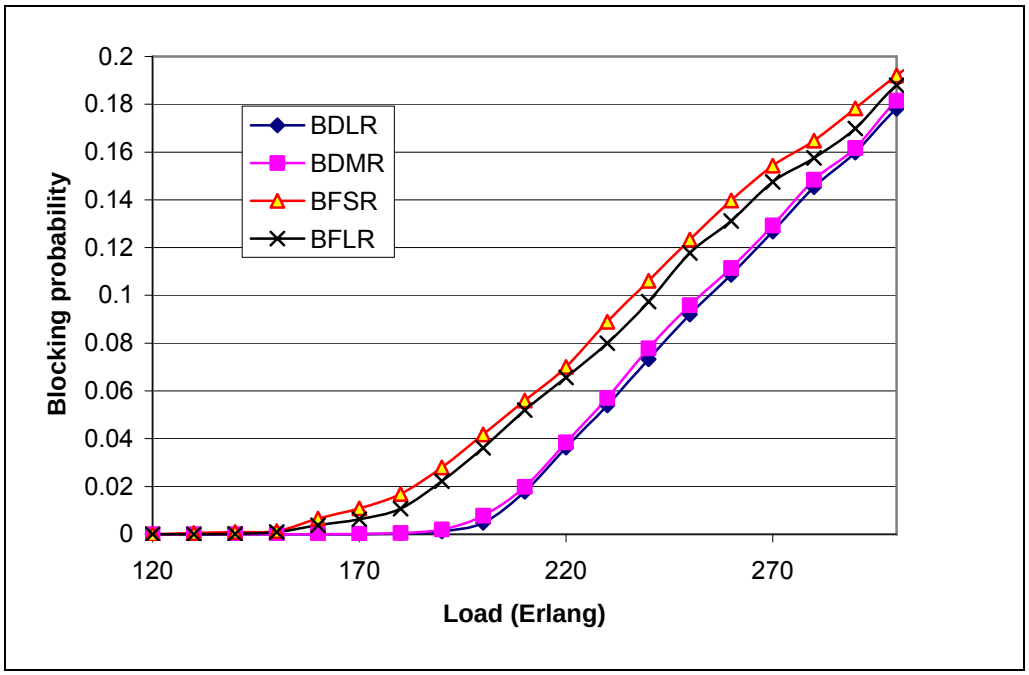


Figure 2.8. Blocking probability vs traffic load. Wavelengths = 16, fibers = 2. NSFNET network with buffer

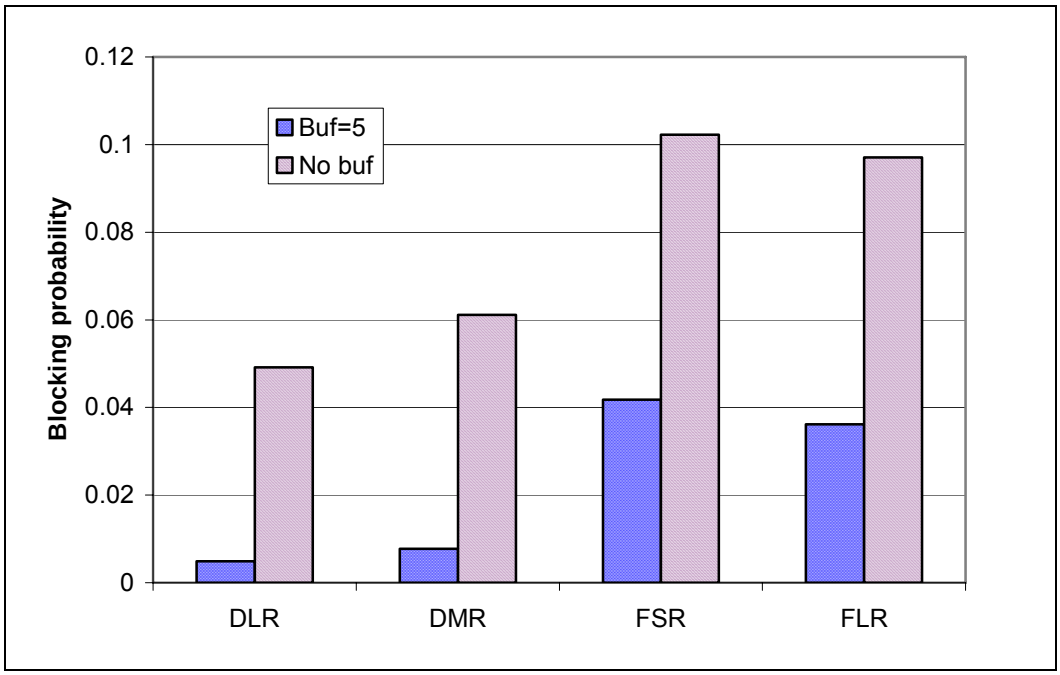


Figure 2.9. Blocking probability comparison for load 200. Wavelengths = 16, fibers = 2.

NSFNET network

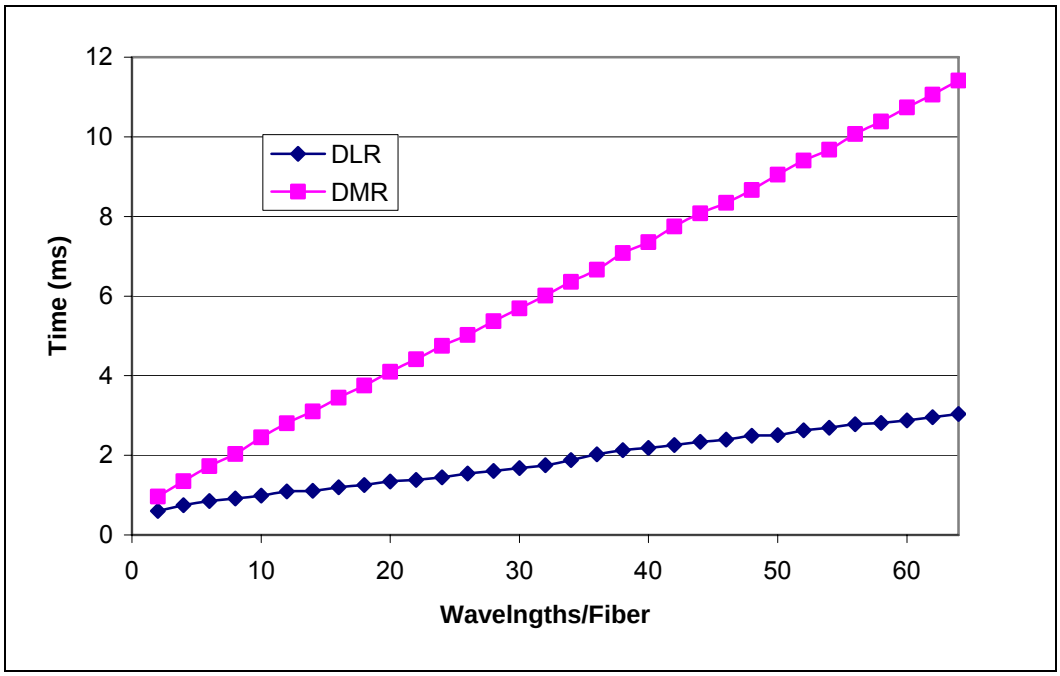


Figure 2.10. Routing speed comparison of DLR , DMR

When we increased the number of fibers per link to three and four, we obtained comparison trends very similar to the case of two fibers per link. Of course, adding more fibers increases the capacity of the network and reduces the blocking probability. For example, Figure 2.11 gives the blocking probability with 4 fibers and 8 wavelengths and Figure 2.12 gives the blocking probability with 3 fibers and 16 wavelengths for the NSFNET topology. Comparing Figure 2.4 (case of 16 wavelengths with 2 fibers) and Figure 2.11 (case of 8 wavelengths with 4 fibers), it is easy to see that both figures have almost the same values of the blocking probability. This is due to the fact that the total number of wavelengths used in each link in both figures is the same (32 wavelengths). Similar observation is obtained when comparing Figure 2.5 and Figure 2.12 (both use a total of 48 channels in each link). The results shown in the various figures clearly

indicate that the blocking probability is reduced when the number of channels in each link is increased.

Figure 2.13 gives blocking probability comparisons at a load of 250 Erlang for the NSFNET network for DLR, DMR, FSR, and FLR with different number of fibers in each link and different number of wavelengths in each fiber. Note that the blocking probabilities for DLR and DMR in the 3x16 and 2x24 scenarios are almost zero and are not visible in the bar chart. Figure 2.13 shows that when the total number of channels in each link (i.e., number of fibers in each link times the number of wavelengths in each fiber) increases, the blocking probability is reduced. Since there is a wavelength-continuity constraint, we can see that the performance of 4 fibers and 8 wavelengths is slightly better than that of 2 fibers and 16 wavelengths (similarly the case of 3 fibers and 16 wavelengths gives slight improvement over the case of 2 fibers and 24 wavelengths). This is because the use of more fibers improves the chance of finding a common free wavelength in the different links of the lightpath.

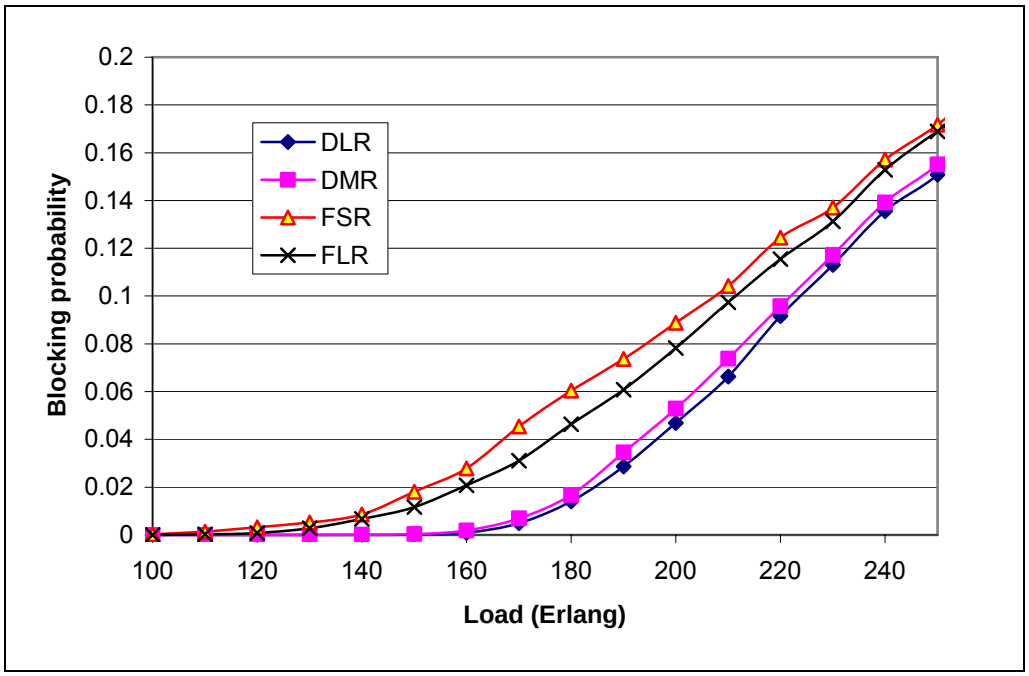


Figure 2.11. Blocking probability vs traffic load. Wavelengths = 8, fibers = 4. NSFNET network

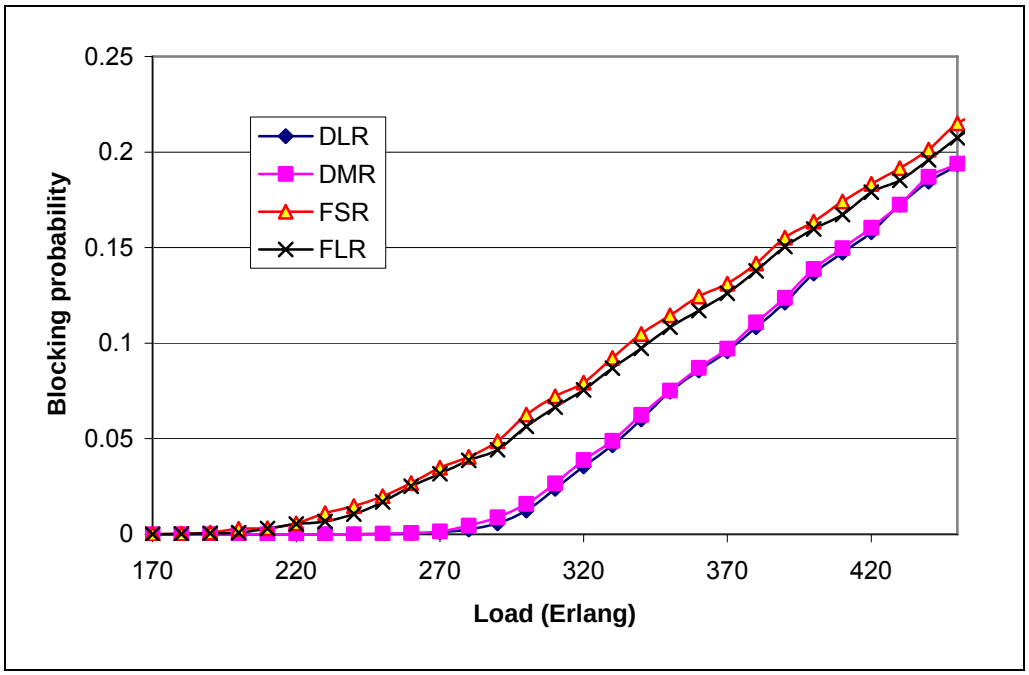


Figure 2.12. Blocking probability vs traffic load. Wavelengths = 16, fibers = 3. NSFNET network

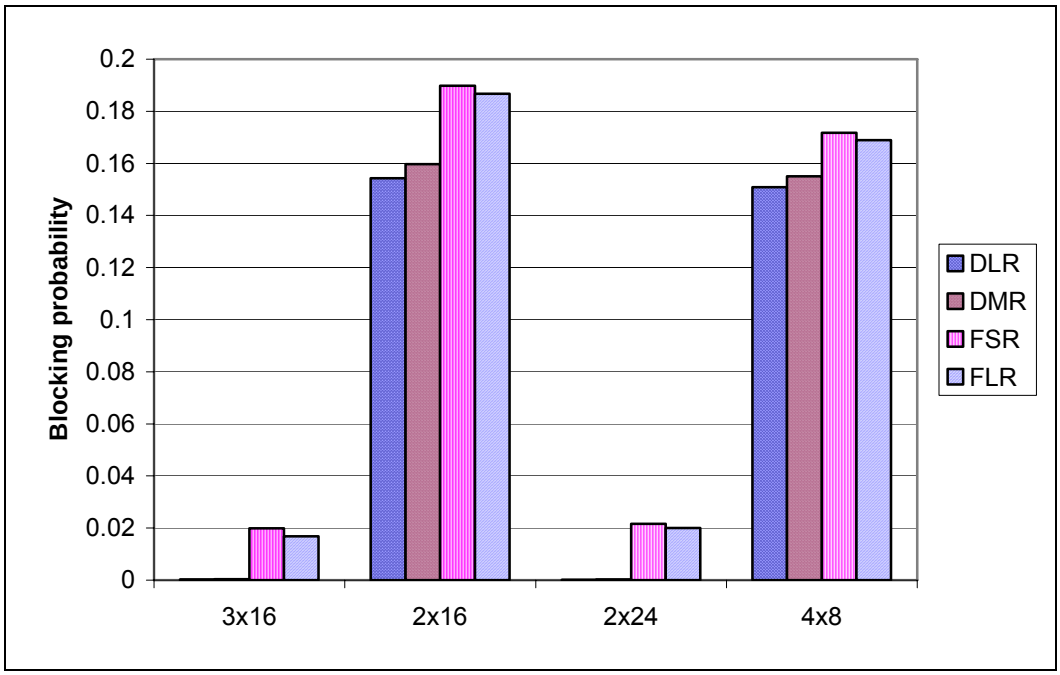


Figure 2.13. Blocking probability comparison for different scenarios. NSFNET network, Load = 250 Erlang (notation: *mxn* means *m* fibers per link and *n* wavelengths per fiber)

CHAPTER 3: IMPROVING FAIRNESS IN OPTICAL BURST SWITCHED NETWORKS

3.1 Introduction

Optical burst switching (OBS) [30-34] has been proposed to provide fine bandwidth granularity and improve the utilization of WDM networks. The basic idea of OBS is that each optical channel is time multiplexed statistically in units of data bursts. Since data bursts are much larger than IP packets, OBS is envisioned to be able to combine the best of optical circuit switching and optical packet switching.

The literature on OBS is rich and growing. Many aspects of burst switching have been extensively investigated including: signaling and reservation protocols [30, 32, 35, 36, 37], routing and scheduling [38, 39], burst assembly at the ingress nodes [40], burst technology and related switch architectures [41], comparison with optical flow switching [42], and contention resolution at the core nodes [43]. In this chapter, we address the issue of fairness in OBS networks. The term "fairness" has been used to address different issues in different types of optical networks. In optical circuit switching, for example, the capacity unfairness problem [44] applies to networks with traffic grooming capability, i.e., the capability of multiplexing and switching lower rate traffic streams onto higher capacity wavelengths. In such grooming environment, a call request that has bandwidth requirement closer to the full wavelength capacity is more likely to experience higher blocking than a call that needs only a smaller fraction. In

OBS networks, there is a different type of unfairness, namely, the dropping probabilities of optical bursts traveling through paths with large hop count tend to be higher than those whose paths have a small number of hops. This latter problem has been studied in several papers as a secondary consideration in the evaluation of routing and channel assignment algorithms [29, 45, 46]. For example, the authors in [45] proposed a deflection routing algorithm and showed that the blocking probabilities of bursts with various hop count at high load levels are almost the same as the case of no deflection, i.e., their proposed deflection routing does not aggravate the unfairness problem for bursts with large hop count. In this chapter, we investigate the unfairness encountered by bursts traveling through long lightpaths and we propose and evaluate the performance of two schemes for alleviating this type of unfairness.

The rest of the chapter is organized as follows. In section 3.2, the *beat down* unfairness problem is discussed and relevant previous work is reviewed. Our first scheme, the balanced JIT scheme, for alleviating the beat down problem is presented in section 3.3. Our second scheme, the prioritized RED scheme, is presented in section 3.4. In section 3.5, the performance results of the two schemes are presented and analyzed.

3.2 The beat down unfairness problem in OBS networks

In OBS, a control packet is transmitted ahead of the data burst on an out-of-band channel to announce the upcoming burst to each optical cross connect (OXC) on the lightpath. A specific offset time is introduced at the source node between the transmission of the control packet and the data burst. During this offset time, the data burst is buffered electronically at the source node

while the control packet propagates forward to reconfigure each OXC along the lightpath. At the end of the offset time, the data burst is transmitted and is switching all-optically through the network without the need of fiber delay lines in any intermediate node. Since there is no connection explicitly established before the burst transmission, the control packet may fail to secure a free channel at some node along its path and the burst may therefore be dropped. Obviously, the larger the number of intermediate nodes in a lightpath the higher the probability that a burst switched through these intermediate nodes will be dropped. This type of unfairness is well known in electronic packet switching networks and is often referred to as the "*beat down*" problem. For example, packets belonging to a TCP connection which traverse many links tend to have more chance to be dropped than packets belonging to a TCP connection with a small number of hops. In particular when TCP is served by "IP over ATM", ATM data cells traveling more hops have a higher probability of getting discarded than those traveling fewer hops. Consequently, long-hop TCP connections may not be able to increase their rates and obtain the bandwidth they deserve. In other words, these long-hop TCP connections are beaten down by their short-hop counterparts.

Few methods have been proposed to deal with the "beat down" problem in electronic packet switching networks such as Fair Buffer Allocation and Selective Packet Dropping [47]. These methods are not suitable for optical networks since they normally have complex logic and require the use of buffers to store data cells/packets. For example, Selective Packet Discard in ATM switches is based on "per-VC accounting" which tracks the number of buffered cells from each virtual connection (VC) in order to reduce cell discarding from low-bandwidth long-hop connections.

There have been very few attempts to design and evaluate schemes to reduce the effect of the beat down problem in OBS networks (note: the term "beat down" has not been used in prior OBS fairness research; we are borrowing this term from the literature of electronic packet switching networks). In [48], an OBS reservation scheme is proposed for OBS networks operating under the wavelength continuity constraint (i.e., no converters). The scheme uses the backward reservation paradigm in which a probe control message propagates from the source to destination then a reservation message travels back from the destination to the source before the data burst can be transmitted. The simulation results reported in [48] uses a three-node tandem network (i.e., maximum of two hops) and a total of three connections (two 1-hop and one 2-hop). The results show that fairness for the 2-hop connection has improved at the expense of a slight increase in the overall mean blocking probability of the three connections. In [49], a preemptive algorithm is proposed to reduce the blocking of bursts with long route lengths in OBS with just-enough-time (JET) signaling. The scheme, called "Merit-based Channel Allocation Algorithm", ranks an arriving burst against those which have been already scheduled for transmission. The value, called "figure of merit", used to determine the ranking is biased in favor of bursts that have long routes, are closer to their destination, or have already consumed large amount of transmission time. If a free channel cannot be found, the scheme then seeks to preempt an already accepted burst based on the computed value of figure of merit for each burst. The simulation results reported in [49] compared five choices (metrics) for computing the figure of merit and showed that the merit based preemption scheme is capable of improving the overall network utilization; there was no fairness analysis and the blocking probabilities for different hop counts were not reported.

In the remainder of this chapter, we propose and evaluate two schemes for dealing with the beat down unfairness problem in OBS. Our schemes have simple logic, do not use preemption, do not require repeated computation to compare accepted bursts and do not introduce any complex modification to the lightpath set-up scheme or the architecture of OXC.

3.3 The Balanced JIT scheme

Our proposed scheme can be applied to OBS networks with either just-in-time (JIT) [32] or just-enough-time (JET) [36] protocol. JIT has a simpler logic than JET and we have selected it for the implementation and evaluation of our schemes. Our OBS network model has the full conversion assumption required in one way reservation protocols such as JIT and JET, i.e., full wavelength conversion capability must be available at each node along the lightpath. However, no optical buffering capability (e.g., fiber delay line) is required in these nodes.

In JIT, the source node delays the transmission of a burst by a certain amount of time after sending the control packet. This delay is called the offset time and is needed to allow each switch in the lightpath to process the control packet and reconfigure the input/ouput ports for the incoming burst. The switch reconfiguration time (also called the cut through time) must be taken into consideration because a burst is dropped if it arrives before the OXC completes its reconfiguration and port setting.

If the lightpath of a burst consists of m hops, the offset time t_d used in JIT can be defined as:

$$t_d \geq m * t_p + t_\delta \qquad (3.1)$$

where t_p is the control packet processing time in each optical cross connect (OXC) including O/E conversion, E/O conversion, and request analysis and routing; t_δ is the extra delay required to assure cut through completion at the last WDM OXC [32]. We next explain the rationale of our first scheme using a simplified high-level model for the probability of burst discarding.

Consider a burst that arrives to an OXC. Let n be the number of wavelengths operational on the output links of this OXC. Let β_i be the probability that the ith wavelength is not free at the time of burst arrival. The blocking (dropping) probability of the burst in this OXC is given by

$$PBLK = \beta_1 * \beta_2 * \beta_3 * \ldots * \beta_n \qquad (3.2)$$

If $\beta_i = \beta$ for all i, then

$$PBLK = \beta^n \qquad (3.3)$$

The "go through" probability of the burst in this OXC is given by

$$PGO = 1 - PBLK \qquad (3.4)$$

If the lightpath of the burst has m hops, the probability that the burst will successfully reach its destination is given by

$$PREACH = PGO_1 * PGO_2 * ... * PGO_m \qquad (3.5)$$

The beat down unfairness problem is clearly explained by equation (3.5). If a shorter lightpath happens to be a prefix of a longer lightpath, the value of *PREACH* for the longer path will be smaller than that of the shorter path. As the number of hops m increases, the probability that the burst will be delivered successfully to the destination decreases. Single-hop lightpaths (i.e., m=1) have the highest probability of successful burst delivery. Our first scheme to alleviate this unfairness is based on a simple observation of equations (3.2)-(3.5): increasing the number of wavelengths n increases the chances of successful burst delivery. As the burst moves from one hop to the next, our scheme gradually increases the number of wavelengths that can be used to switch this burst. Let W be the (maximum) number of wavelengths that are used for burst switching in each OXC. When the control packet of a burst arrives at its first OXC, a certain fraction of the W wavelengths in this OXC is searched for a free wavelength. If no free channel is found, the burst is dropped even though a free channel may be available in one of the other wavelengths that have not been searched. In the second hop (OXC) of the burst, a larger fraction of W is searched for a free channel. Specifically, the number of wavelengths n_i that are searched in the ith hop of a burst is given by

$$n_i = (1-g) * W + g * i * W / D \qquad (3.6)$$

where g is a parameter that is assigned a value between 0 and 1 and D is the diameter of the network (the maximum number of hops in a lightpath in the network). We call this method "balanced JIT" and denote it BJIT(g) where g is a controllable parameter of the scheme. The parameter g divides the search spectrum in each OXC into two parts: a base part and an adjustable part. The base part has a fixed size of $(1-g)*W$ wavelengths regardless of the hop count of the lightpath. The adjustable part has a size that depends on the number of hops traveled by the burst so far, and can reach a maximum size of $g*W$ wavelengths. For example if the network diameter is D=10, the size of the adjustable part is $0.1*g*W$ at the first hop, $0.2*g*W$ at the second hop, and so on. It should be noted that the high-level model of equations (3.2)-(3.5) applies to both JIT and JET and the balanced scheme is therefore suitable for JET as well. The balanced scheme is very easy to implement and does not demand any additional software or hardware resources in the OXC's. The current hop count of the burst, i, used in equation (3.6) is easily passed from one OXC to the next via the control packet. Implementing the reduced (adjustable) search to find a free wavelength requires minor modification to the standard JIT channel allocation scheme; the adjustable search (i.e., searching in a space of size $g*i*W/D$) actually leads to a smaller average search time.

Note that when g=0, the adjustable part of equation (3.6) vanishes and the scheme becomes equivalent to the standard JIT scheme. In other words, BJIT(g=0) has the same level of beat down unfairness as JIT. As the value of g increases, data bursts on long lightpaths get better treatment because the size of their adjustable search space increases and hence their "go thorough" probability gradually improves after each OXC they successfully pass. It is obvious

that larger value of g will be more effective in reducing the beat down unfairness. Ideally, we should choose a value of g that satisfies the following two conditions:

- Eliminate, or reduce the severity of, the beat down unfairness by reducing the probability of dropping bursts with larger hop count at the expense of some increases in the dropping probability of bursts with small hop count.
- Avoid negatively impacting the overall throughput of the system, i.e., do not increase the overall burst dropping probability.

The BJIT scheme and the logic used in equation (3.6) were inspired by our earlier work on reducing the dropping probability of handoff requests in the base stations of cellular wireless networks [50]. Abstractly speaking, the adjustable term of equation (3.6) is a generalization of the guard channels that are exclusively dedicated to handoff requests in order to give them priority over new call cellular requests. Our extensive tests have shown that the BJIT scheme can reduce the beat down unfairness in OBS networks while maintaining (or even slightly improving) the throughput. As explained earlier, bursts with long lightpaths are dropped more frequently by standard JIT (or JET) than those with short paths. As the control packet gets closer to its destination, the time separating the control packet and its burst decreases, making it more difficult to salvage the burst using optimized scheduling techniques such as those used in JET. Thus bursts with longer paths are more likely to get dropped, and they are often dropped in an OXC closer to their destination after having wasted channel resources in the previous OXC's. The BJIT scheme avoids this problem in two ways. On one hand, bursts with long paths have a smaller dropping probability under BJIT than under JIT. On the other hand, the bursts that get

dropped in the BJIT scheme are likely to be dropped near their source node rather than near their destination. The reason is that equation (3.6) ensures that a larger and larger adjustable search space is allocated to the burst as it moves from one OXC to the next. Therefore the BJIT scheme frees up some channel resources that would have been otherwise wasted if the dropped bursts were allowed to travel some more extra hops before being discarded. The discussion on choosing the appropriate value of g and the detailed performance results of the BJIT scheme are given in section 3.5.

3.4 The Prioritized Random Early Dropping (PRED) Scheme

Our second scheme adapts the concept of random early discard (RED) to the OBS environment and prioritizes the levels of discarding based on the length of the lightpath. We call this scheme Prioritized RED or PRED.

The RED concept [51] has received considerable interest in electronic packet switching networks and RED routers have been widely deployed in various commercial applications and in the Internet. There have been numerous studies that support or oppose RED [51-53], present schemes for tuning RED parameters [54], propose modified versions of RED [55], and develop analytical models for RED performance [56]. The basic idea of RED is that routers proactively discard incoming packets with probabilities that depend on the size of the router's queue. The TCP congestion control algorithm [57] reacts to lost packets by throttling the transmission rate of TCP senders. Studies have shown that well-configured RED routers have the potential to avoid

severe congestion and improve the overall throughput while maintaining a small queuing delay within each router.

Our PRED scheme uses proactive burst dropping with a discarding probability that decreases as the length of the lightpath increases. The goal of burst discarding in PRED is not to avoid congestion or throttle TCP senders as in RED (although these could be positive side effects of PRED). Rather, PRED uses proactive discarding to alleviate the beat down unfairness. A major difference between RED and PRED is that our PRED scheme restricts burst discarding to the original source node of the burst while RED allows any router in the path of a packet to proactively drop it. Specifically, all proactive discarding in PRED is done in the network access station (NAS) of the source node that generated the burst. This has the advantage that the discarded bursts will not waste any bandwidth resources in the core of the optical network.

Let α_i be the probability used by PRED at the source NAS to discard a newly incoming burst whose lightpath has a length of i hops. To alleviate the beat down problem, the values of the discarding probabilities must satisfy the following constraint

$$\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_D \qquad (3.7)$$

where D is the diameter of the network as explained in section 3.3.

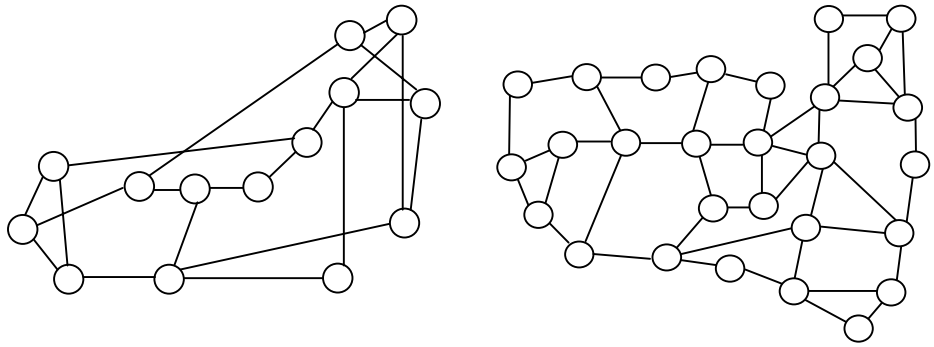
The proactive discarding of relation (3.7) is only applied to the local bursts assembled at this NAS. The NAS may also be servicing transit bursts that have come from some external OXC's

and are being routed to other external OXC's. These transit bursts have already escaped the PRED proactive discarding in the NAS where they were generated. These transit bursts have also already consumed some network bandwidth resources during their partial trip toward their destination. By proactively discarding local bursts and not transit bursts, more bandwidth will be available to transit bursts at the current OXC. This increases the likelihood that transit bursts will reach their destination without wasting the resources that they have already used prior to reaching the current NAS.

As in standard RED, the proactive discarding in PRED should not take place if the load on the OXC is light. This is because at light loads, most bursts are expected to reach their destination successfully and the beat down unfairness is not noticeable. We have adopted a simple mechanism to disable/enable proactive discarding in PRED. In OBS, each NAS uses a buffer to hold assembled bursts until they are sent to the local OXC. Bursts arriving when this buffer is empty do not get discarded by PRED. When the buffer is not empty, new bursts are subjected to the probabilistic discarding of PRED.

3.5 Performance evaluation results and analysis

Figure 3.1 shows the two network topologies that are used in our simulation (NSFNet network with 14 nodes and US LongHaul network with 28 nodes).



NSFNet Topology LongHaul Topology

Figure 3.1. Two network topologies

In our simulation, a static lightpath between any two nodes is established using the shortest path first method as was done in [49, 26, 27]. Notice that the longest shortest path in the LongHaul topology has 7 hops and in the NSFNet topology has 3 hops. Similar to [49], the traffic used in our tests is uniformly distributed among all nodes. This means that all nodes have equal likelihood to be the source of a data burst and for a given source node all other nodes in the network have equal likelihood to be the destination node. This uniform distribution on pairs of nodes gives the following distribution on the number of hops in the burst's lightpath: for the NSFNet topology, the percentage of lightpaths with number of hops equal to 1, 2, and 3 is 23%, 40% and 37%, respectively; for the LongHaul topology, the percentage of lightpaths with number of hops equal to 1, 2, 3, 4, 5, 6 and 7 is 12%, 20%, 23%, 21%, 14%, 8% and 2%, respectively.

In our simulation tests, assembled bursts arrive according to a Poisson distribution with controllable arrival rate. For each burst, the source node and destination node are randomly

selected as explained before. Our simulation tests used values for OXC and network parameters similar to those typically used in the literature: the cut through time in each OXC is 2.5 milliseconds, the link delay per hop is 3 milliseconds, the burst length is 50 microseconds (equivalent to a burst of 250 Kbits at 5 Gbits/sec), the control packet processing time t_p at each hop is 50 microseconds. The number of wavelengths used in each OXC is W=40. Each point in the performance graphs/tables reported in this chapter was obtained by averaging the results of 6 simulation tests using different random generation seed values. Each simulation was run for sufficiently long time to obtain stable statistics; the total number of bursts processed in each simulation test ranged from 2 million bursts at low arrival rates to 12 million bursts at high loads. The time unit (tu) used in the tables/graphs presented in this chapter is equal to 0.05 millisecond. Thus a load of 12 bursts/tu is equivalent to 60 Gbits/second.

3.5.1 Performance of BJIT

Figures 3.2 and 3.3 show the burst drop probability for different hop counts at load 12 burst/tu (60 Gbits/second) in the LongHaul and NSFNet networks, respectively. The horizontal axis gives the value of the parameter g. Notice that the case g=0 corresponds to the standard JIT scheme. The figure shows that the beat down unfairness is gradually alleviated as the value of g increases. When g reaches the value 1, the unfairness is more or less eliminated and the drop probabilities for all hop counts are nearly the same.

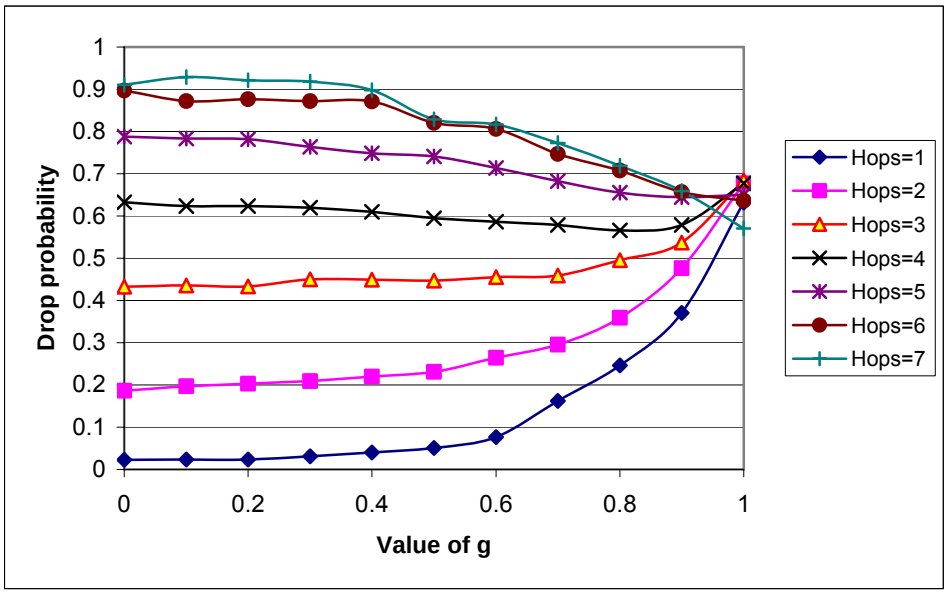


Figure 3.2. BJIT drop probability for different hop counts in LongHaul (load=12)

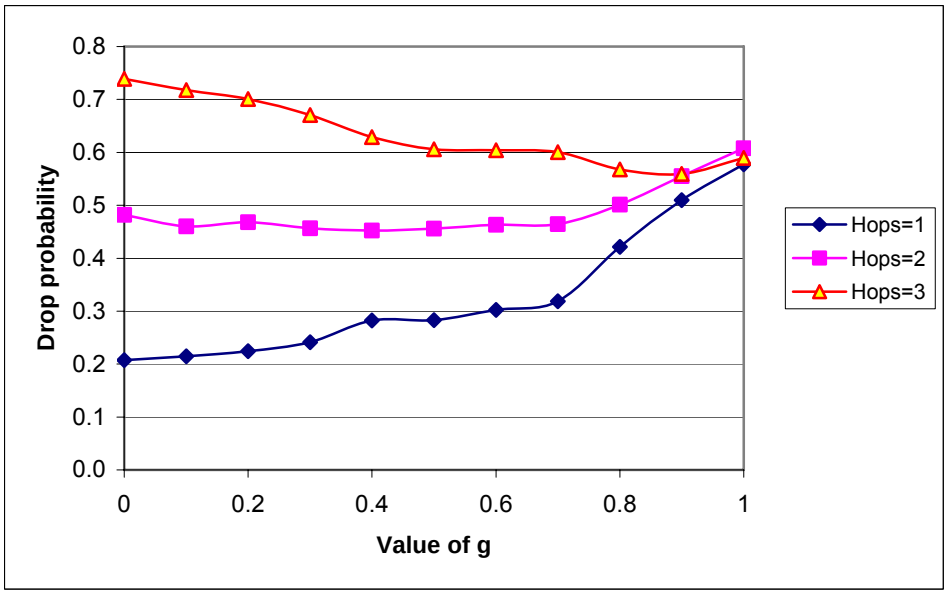


Figure 3.3. BJIT drop probability for different hop counts in NSFNet (load=12)

Table 3.1 shows the distribution of the drop probability at different loads for standard JIT (i.e., BJIT with *g*=0) in the LongHaul network. Table 3.2, 3.3, 3.4, and 3.5 give similar distribution for BJIT with parameter value of *g*=0.2, *g*=0.5, *g*=0.8, and *g*=1.0, respectively. Since the general trends of the drop probability distribution for the NSFNet network are similar to those of the distribution for the LongHaul network, we omit the tables for the drop probability distribution of NSFNet in this chapter.

Table 3.1. Drop probability distribution for standard JIT (g=0) in LongHaul

Load	Hops=1	Hops=2	Hops=3	Hops=4	Hops=5	Hops=6	Hops=7
2	0.0000	0.0002	0.0004	0.0002	0.0003	0.0006	0.0000
4	0.0008	0.0115	0.0329	0.0621	0.0930	0.1580	0.1836
6	0.0023	0.0407	0.1156	0.2284	0.3654	0.4730	0.4683
8	0.0049	0.0748	0.2252	0.4058	0.5470	0.7019	0.7231
10	0.0105	0.1236	0.3356	0.5208	0.7020	0.8103	0.8541
12	0.0230	0.1861	0.4328	0.6325	0.7881	0.8968	0.9106

Table 3.2. Drop probability distribution for BJIT (g=0.2) in LongHaul

Load	Hops=1	Hops=2	Hops=3	Hops=4	Hops=5	Hops=6	Hops=7
2	0.0000	0.0000	0.0002	0.0012	0.0000	0.0007	0.0000
4	0.0019	0.0138	0.0370	0.0636	0.1070	0.1439	0.1707
6	0.0030	0.0489	0.1303	0.2352	0.3530	0.4401	0.4538
8	0.0060	0.1211	0.2270	0.3926	0.5364	0.6954	0.7041
10	0.0115	0.1344	0.3275	0.5235	0.6745	0.8106	0.7814
12	0.0235	0.2033	0.4337	0.6234	0.7816	0.8766	0.9041

Table 3.3. Drop probability distribution for BJIT (g=0.5) in LongHaul

Load	Hops=1	Hops=2	Hops=3	Hops=4	Hops=5	Hops=6	Hops=7
2	0.0008	0.0000	0.0000	0.0007	0.0010	0.0006	0.0000
4	0.0017	0.0128	0.0364	0.0676	0.0990	0.1251	0.1287
6	0.0038	0.0561	0.1181	0.2211	0.3472	0.4332	0.4647
8	0.0098	0.1283	0.2557	0.3921	0.4880	0.5528	0.6019
10	0.0232	0.1678	0.3374	0.4901	0.6328	0.7791	0.7996
12	0.0508	0.2307	0.4467	0.5950	0.7407	0.8204	0.8286

Table 3.4. Drop probability distribution for BJIT (g=0.8) in LongHaul

Load	Hops=1	Hops=2	Hops=3	Hops=4	Hops=5	Hops=6	Hops=7
2	0.0000	0.0010	0.0016	0.0007	0.0007	0.0000	0.0025
4	0.0037	0.0241	0.0584	0.0838	0.1363	0.1137	0.1070
6	0.0066	0.0709	0.1618	0.2426	0.3511	0.4155	0.4174
8	0.0689	0.1773	0.2967	0.3971	0.4846	0.5490	0.5133
10	0.1616	0.2768	0.4013	0.4776	0.5790	0.6436	0.6527
12	0.2462	0.3590	0.4954	0.5661	0.6551	0.7079	0.7189

Table 3.5. Drop probability distribution for BJIT (g=1.0) in LongHaul

Load	Hops=1	Hops=2	Hops=3	Hops=4	Hops=5	Hops=6	Hops=7
2	0.0004	0.0027	0.0013	0.0036	0.0052	0.0025	0.0000
4	0.0925	0.1263	0.1526	0.1571	0.1511	0.1494	0.1018
6	0.3019	0.3531	0.3690	0.3630	0.3615	0.3361	0.2848
8	0.4498	0.5023	0.5208	0.5429	0.4819	0.4729	0.3567
10	0.5634	0.6267	0.6246	0.5911	0.5811	0.5745	0.5268
12	0.6491	0.6818	0.6906	0.6285	0.6323	0.6360	0.5706

Figures 3.2-3.3 and Tables 3.1-3.5 examined the drop probabilities for individual hop count values and showed that higher values of g are more effective in alleviating the unfairness problem. We now examine the overall throughput and the overall average drop probability over

all hop counts. Figures 3.4 and 3.5 show the total throughput (in Gbits per time unit) at different values of *g* and different loads in the LongHaul and NSFNet networks, respectively.

The throughput is calculated by the following method.

Let

Total simulation time = T time units

Total number of bursts transferred successfully from source to destination = N_b

Burst length = 250K bits

then

$$Throughput = N_b * 250/(1000000 * T) \quad \text{Gbits/time unit} \qquad (3.8)$$

As evident from these graphs, the value *g*=1 degrades the throughput significantly while the value *g*=0.5 has the best throughput performance. Notice that the throughput at *g*=0.5 is better than the throughput of standard JIT (i.e., when *g*=0).

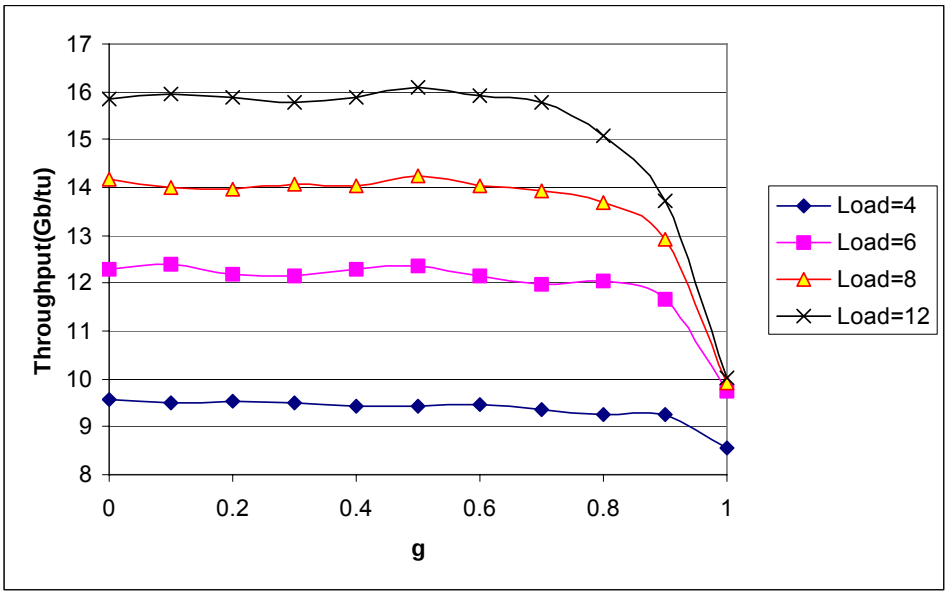


Figure 3.4. BJIT throughput at different values of *g* in LongHaul

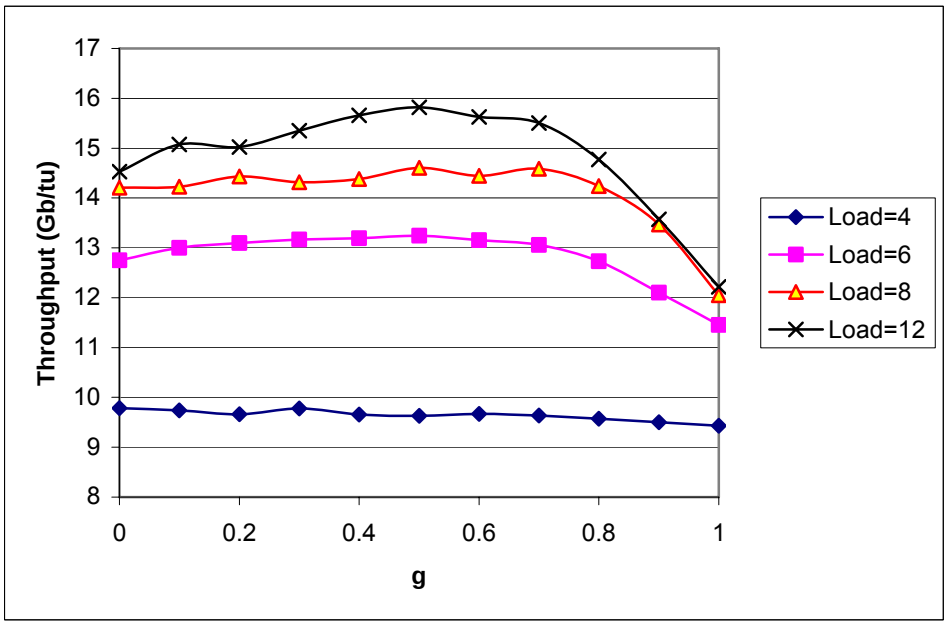


Figure 3.5. BJIT throughput at different values of *g* in NSFNet

Figure 3.6 and 3.7 give bar charts for the average drop probabilities (over all hops counts) for the LongHaul and NSFNet networks, respectively. Again, the value *g*=1 has the worst performance and the value *g*=0.5 has the best overall average drop probability.

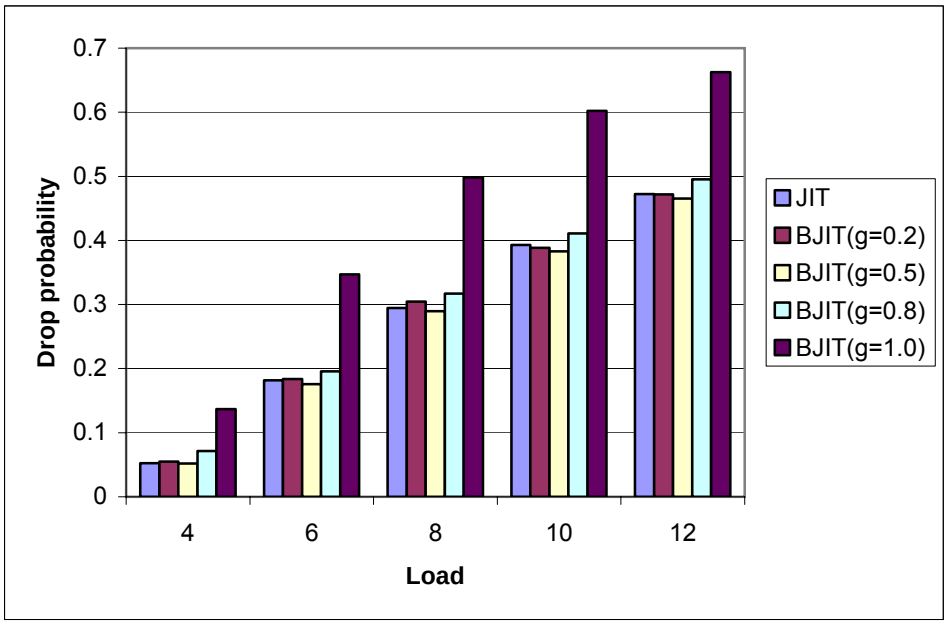


Figure 3.6. BJIT overall average drop probability for different *g* values in LongHaul

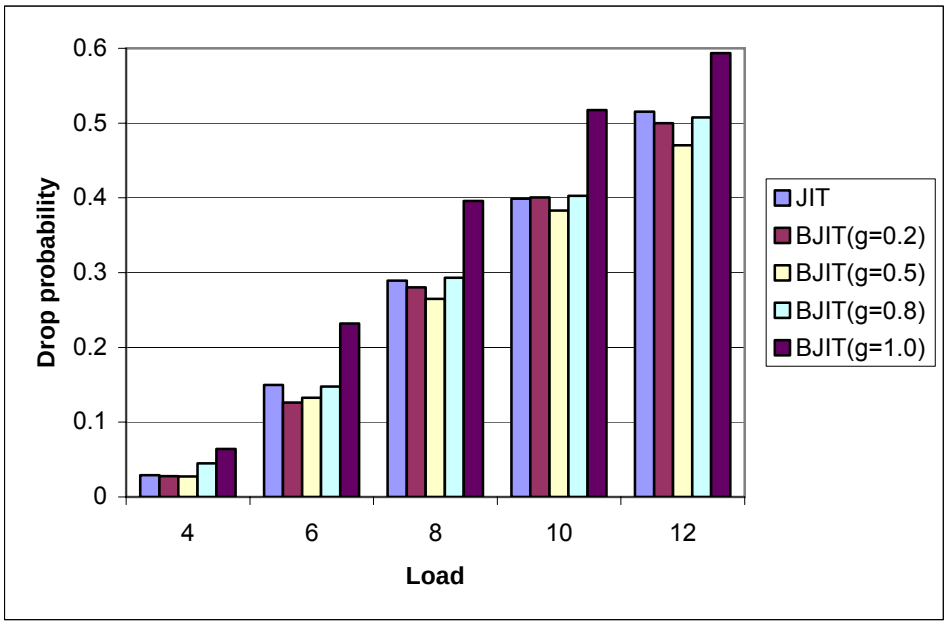


Figure 3.7. BJIT overall average drop probability for different *g* values in NSFNet

The above results suggest that the complete elimination of the beat down unfairness in OBS can be achieved at the expense of degraded throughput performance. However, using values of *g* in the neighborhood of 0.5 offers definite practical advantage compared to standard JIT, namely, the throughput of the system is improved and the severity of the beat down unfairness is alleviated.

3.5.2 Performance of PRED

PRED uses proactive discarding at the source NAS with probability α_k to discard bursts that are k hops away from their destination. Unlike the BJIT(*g*) scheme which has a single parameter, the PRED scheme requires adjusting the values of *D* probabilities. For the LongHaul topology, for

example, the value of D is 7 and there are seven probabilities to adjust. However, the constraint needed to alleviate unfairness (i.e., $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_D$) greatly simplifies the parameter's tuning process.

Table 3.6 shows the burst drop probability for different loads and hop counts in the LongHaul topology with PRED. The values of the proactive drop probabilities used in the source NAS are 0.2, 0.15, 0.1, 0.05, 0.02, 0.0, and 0.0 for paths with 1, 2, 3, 4, 5, 6 and 7 hops, respectively. Notice that the probabilistic discarding is disabled when the NAS's buffer is empty and the actual rate of proactive discarding is therefore much lower than the above probability values. The overall average drop probability (over all hop counts) corresponding to the results of Table 3.6 is shown in Figure 3.8. From Table 3.1, Table 3.6 and Figure 3.8, we can easily see that the PRED scheme has improved the fairness compared to the standard JIT scheme without any negative impact on the overall drop probability (in this experiment, both PRED and standard JIT have the same throughput). The results for the NSFNet topology are similar and will not be given in this chapter.

Table 3.6. Drop probability distribution for PRED in LongHaul

Load	Hops=1	Hops=2	Hops=3	Hops=4	Hops=5	Hops=6	Hops=7
2	0.0012	0.0003	0.0000	0.0002	0.0003	0.0000	0.0000
4	0.0016	0.0161	0.0385	0.0691	0.0976	0.1066	0.0999
6	0.0165	0.0649	0.1538	0.2308	0.3120	0.3781	0.3927
8	0.0243	0.1020	0.2603	0.3740	0.4913	0.5745	0.6402
10	0.0312	0.1681	0.3736	0.5053	0.6362	0.7181	0.7316
12	0.0414	0.2321	0.4741	0.6089	0.7484	0.7673	0.7716

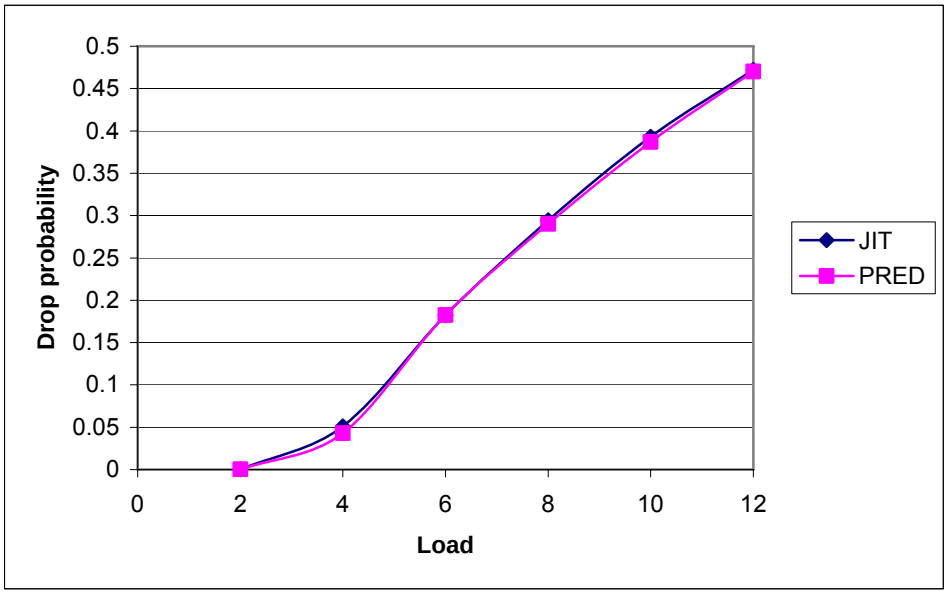


Figure 3.8. Overall average drop probability for PRED in LongHaul

The PRED scheme described above has a very simple logic and is easy to implement. We have found that it is possible to use more complex design variations of the above simple scheme in order to simultaneously improve both the fairness and throughput performance. In one variation, the discarding probabilities are dependent on the load of the network rather than being fixed values as was done in the tests of Table 3.6. In another variation, we implemented two types of proactive discarding with different probability values. The first type of discarding is applied to newly assembled bursts exactly as in the simple PRED scheme. The second type is additional discarding applied to the bursts that are stored in the buffer waiting for transmission to the local OXC. These different variations and/or combinations of them give better performance at the

expense of increased complexity. As an example, Table 3.7 shows the burst drop probability in the LongHaul topology for the variation that uses two types of discarding. Table 3.6 and Table 3.7 show that the two-type proactive discarding scheme improves fairness compared to simple PRED. The throughput of this variation is very slightly better than that of simple PRED.

Table 3.7. Drop probability distribution for a modified PRED in LongHaul

Load	Hops=1	Hops=2	Hops=3	Hops=4	Hops=5	Hops=6	Hops=7
2	0.0004	0.0005	0.0000	0.0005	0.0003	0.0000	0.0000
4	0.0095	0.0235	0.0388	0.0506	0.0714	0.0769	0.0845
6	0.0402	0.1017	0.1622	0.2235	0.2871	0.3248	0.3588
8	0.0684	0.1731	0.2786	0.3577	0.4217	0.4747	0.5283
10	0.1032	0.2462	0.3871	0.4804	0.5310	0.6136	0.6151
12	0.1551	0.3159	0.4916	0.5643	0.6386	0.6730	0.6855

CHAPTER 4: SUPPORTING DIFFERENTIATED QUALITY OF SERVICE IN OPTICAL BURST SWITCHED NETWORKS

4.1 Introduction

To provide fine bandwidth granularity and improve the utilization of wavelength division multiplexed (WDM) optical networks [14], optical burst switching (OBS) has been proposed [30, 58]. In real life applications with differentiated QoS requirements, data bursts should have different priority classes. Higher priority bursts should be given preferential treatment in order to reduce their drop probability and their end-to-end delay. The growing interest in introducing QoS differentiation in Internet services is motivated by the need to improve the quality of support for IP voice and video services, and in general, by the desire to provide clients with a range of service-quality levels at different prices. Since WDM optical networks are rapidly becoming the technology of choice in network infrastructure and next-generation Internet architectures, implementing QoS differentiated services and designing network protocols to support a range o of service-quality levels in WDM and OBS networks have received increased recent attention. In [58], differentiated Quality of Service has been incorporated into Just-Enough-Time (JET) scheduling [30] by assigning different offset times to different classes. The higher priority class is given larger offset time. The drawback of this approach is that larger offset times may result in longer delays for higher priority bursts. The review in section 4.2 presents several other proposals for improving QoS differentiation in optical burst switched (OBS) networks. In this

chapter, we propose and evaluate two schemes to improve the QoS differentiation in OBS networks. The two schemes are easy to implement and produce tangible QoS differentiation.

The rest of the chapter is organized as follows. In section 4.2, the QoS differentiation problem in optical networks is discussed and relevant previous works are reviewed. Our first scheme, qualified JIT, is presented in section 4.3. Our second scheme, prioritized RED, is presented in section 4.4. In section 4.5, the performance results of the two schemes are presented and analyzed. In section 4.6, the conclusion of the chapter is given.

4.2 The QoS differentiation problem in OBS networks

In standard OBS networks, a control packet is transmitted ahead of the data burst on an out-of-band channel to reserve a channel for the upcoming burst in each optical cross connect (OXC) along the lightpath of the burst. There is a special offset time that is introduced at the source node between the transmission of the control packet and the data burst. During this offset time, the burst data is buffered electronically in the network access station (NAS) while the control packet propagates forward to configure each OXC along its lightpath. When the offset time expires, the burst is sent out and is switched all-optically from one node to the next until it reaches the destination. It is possible, however, that the control packet fails to secure a free channel in some congested intermediate node along the lightpath. This results in dropping the data burst at the congested node. The data burst dropping probability generally increases as the load on the network increases.

There are two main optical burst scheduling methods: just-in-time (JIT) [32] and just-enough-time (JET) [30]. Several methods to support differentiated QoS in OBS networks have been proposed in the literature. For example, the authors in [58] propose a method to support QoS differentiation in JET by adjusting the offset time for different priority classes. The basic idea of this method is to assign a larger offset time for a higher priority class than the offset time assigned for a lower priority class. The larger offset time in JET increases the chance of securing a successful wavelength reservation for the burst. However, larger offset times also increase the delay for higher priority classes. In [75], QoS is supported by adjusting the offset time and the lower and upper bounds on blocking probability for two burst classes are analyzed. While the method of increasing offset times works well for QoS prioritization in JET, it is not exactly a suitable method for JIT. The JIT scheduling method is less complex than that of JET and cannot fully utilize delayed reservation [30]. In this chapter, we propose and investigate two schemes that can be used to support QoS differentiation in both JIT and JET.

Burst assembly at the ingress nodes [40] is another method to improve QoS in JET. In this method, the window size and weight of a class determines the number of packets in the window. In [59], the QoS differentiation in JET is investigated by analyzing the loss probabilities of two classes of bursts. As was done in [58], the authors in [59] also focus on changing the offset time to support differentiated QoS. In [60, 69], a burst segmentation method is proposed to address the differentiated QoS problem. Using the segmentation scheme, the bursts with higher priority can preempt the overlapping segments of lower priority bursts and the preempted segments are dropped or are deflected to alternate routes. In [61], a proportional model is proposed to enhance the offset-based QoS differentiation method proposed in [58]. In the proportional model, the

differentiation of a particular QoS metric can be quantitatively adjusted to be proportional to the factors that a network service provider sets. The lower priority burst is intentionally dropped when the proportional differential model is violated. In [74], the authors combine a service differentiation model based on proportional resource allocation with a partially preemptive burst scheduler. The scheme improves the utilization while providing QoS with controllable service differentiation. In [62], the authors propose a differentiation scheme for JET that does not assign extra time to a higher priority class as in [58]. Rather it uses a priority queuing technique to schedule a higher priority burst earlier than a lower priority burst. Another proportional differentiation model is used in [68]. In this model, the burst control packets are queued in increasing order of the burst preferred scheduling time (defined as a burst arriving time minus its differential time). Each OXC chooses its own differential time function according to its resource availability and QoS requirement. In [63], some undesired characteristics of the offset-time management mechanism for JET are identified. The authors found that the burst drop probability differentiation attained for a given offset-time value strongly depends on the distribution of the burst durations and that controlling the differentiation is difficult. In [64], assured Horizon is introduced for a coarse-grained bandwidth reservation r_i for every forwarding equivalent class (FEC) between ingress and egress. The burst assembler marks bursts as compliant and non-compliant bursts, depending on whether the burst is conforming to r_i. The non-compliant bursts are dropped when congestion occurs. In [70], a generalized Latest Available Unused Channel with Void Filling (LAUC-VF) algorithm is proposed. The LAUC-VF algorithm aims at providing good performance by essentially choosing the wavelength with the smallest possible available window, leaving larger windows for control packets that arrive later. LAUC-VF is basically another scheduling scheme for OBS networks and is used in [70] to support

differentiated services with limited buffers. In [65], an algorithm is proposed to decide about the value of the burst offset-time in JET based on the burst priority class. In [66], linear predictive filter (LPF) based forward resource reservation is proposed for JET to reduce the burst delay at edge routers. An aggressive reservation method is also proposed to increase the successful forward reservation probability and to improve the delay reduction performance. In [67], preemptive multiclass wavelength reservation is used to provide differentiated services for the JET protocol. The preemption scheme provides QoS differentiation but complicates the control logic in OXC's. In [73], a probability preemptive scheme is proposed in which high priority bursts can preempt low priority bursts in a probabilistic manner. In [76], the authors propose a priority-based wavelength assignment (PWA) algorithm. In this algorithm, each wavelength has its own priority that can be changed according to the burst reachability. In [78], the authors propose QoS-guaranteed wavelength allocation schemes for WDM networks. In their schemes, the wavelengths are classified into different sets based on the QoS requirement and higher priority requests can be allocated more wavelengths. In each set, there are two rules to select the idle wavelength: minimum index numbered or maximum index numbered wavelength. The authors in [78] showed that the connection loss probability for higher priority requests is improved but the overall throughput performance of their schemes is not discussed.

In this chapter, we propose two new schemes to support QoS differentiation in OBS networks suitable for both JIT and JET scheduling. Compared to the previous proposals, our schemes have simple logic that can be easily implemented even in the JIT scheduling method. Specifically, the two schemes do not increase the offset time of high priority bursts (thus do not increase their delay), do not use complex scheduling functions, do not introduce additional

queuing or segmentation mechanisms, do not resort to burst preemption and do not introduce any complex modification to the lightpath set-up scheme or the architecture of OXC. We have implemented a performance simulation model of the two schemes in OBS networks using JIT scheduling. Our extensive performance results show that the two methods improve QoS differentiation compared to the original JIT scheduling scheme without negatively impacting the network throughput. Since our schemes do not modify or depend on the scheduling logic of JIT, they can be also easily implemented in JET. The improved JET channel scheduling algorithms recently proposed in [71] focus on how to assign the void intervals for bursts. Since our proposed schemes merely focus on how to assign wavelengths to different priority bursts, they can be implemented without conflict in OBS networks that use the efficient JET algorithms given in [71].

4.3 Qualified JIT scheme

The JIT scheduling protocol is used in this chapter to illustrate the applicability of our two proposed QoS differentiation schemes and evaluate their efficiency. As has been frequently assumed in JIT/JET scheduling [58, 30, 32], full wavelength conversion capability is assumed to be available at each node along the lightpath of the burst.

In JIT, the source node delays the transmission of a data burst by a certain amount of time after sending the control packet. The amount of this delay (called the *offset time*) is decided by the number of hops along the lightpath and the cut through time in each node. The offset time allows each hop (OXC) along the lightpath to configure its port connection for the incoming burst. The switch reconfiguration time (also called the cut through time) must be taken into

consideration because a burst is dropped if it arrives before the OXC completes its connection reconfiguration.

Normally, if the lightpath of a burst consists of m hops, the offset time t_d used in JIT can be defined as:

$$t_d \geq m * t_p + t_\delta \qquad (4.1)$$

Where t_p is the control packet processing time in each OXC including O/E-E/O conversions and request/routing analysis; t_δ is the extra delay required to assure cut through completion at the last OXC in the lightpath [32].

We next explain the rational of our first scheme using a simplified high-level model for the probability of burst discarding in JIT.

Consider a burst that arrives at an OXC. Let n be the number of wavelengths operational on the destination output link of this OXC. Let β_i be the probability that the ith wavelength is not free at the time of burst arrival. The dropping probability of the burst in this OXC is given by:

$$P_{drop} = \beta_1 \times \beta_2 \times \beta_3 \times \ldots \times \beta_n \qquad (4.2)$$

If $\beta_i = \beta$ for all i, then

$$P_{drop} = \beta^n \qquad (4.3)$$

The "go through" probability of the burst in this OXC is given by

$$P_{go} = 1 - P_{drop} \qquad (4.4)$$

If the lightpath of the burst has *m* hops, the probability that the burst will successfully reach its destination is given by

$$P_{success} = P_{go_1} \times P_{go_2} \times P_{go_3} \times \ldots \times P_{go_m} \qquad (4.5)$$

The above equations apply to an arbitrary burst from any traffic class. This means that standard JIT treats all traffic classes equally and does not provide differentiated QoS to higher priority classes. In this section, we propose a wavelength assignment scheme that skews the search of free wavelength in favor of higher traffic classes. By introducing a bias in the search process, a higher priority burst will get a better chance to go through an OXC than a lower priority burst. The basic idea of the scheme is to make more free wavelengths available to higher priority bursts than to lower priority bursts. The scheme is based on a simple observation of equation (4.2)-(4.4), namely, increasing the number of wavelengths *n* increases the chances of successful burst delivery. As the priority of the burst increases, our scheme gradually increases the number of wavelengths that can be used to switch this burst. Let *W* be the maximum number of wavelengths that are used for burst switching in each OXC and let *P* be the number of burst

classes. We assume that class *P* has the highest priority, class *P-1* has the second highest priority, and class *1* has the lowest priority. When the control packet of a burst arrives at some OXC, it will reserve the wavelength based on its burst priority. For the burst with priority *1*, the control packet is allowed to search just a fraction of the *W* wavelengths in this OXC. For the burst with priority *2*, the control packet is allowed to search a fraction of *W* wavelengths that is larger than the fraction for priority *1*, and so on. For a burst with priority *P*, all of the *W* wavelengths can be searched. Specifically, for the ith priority control packet, the number of wavelengths n_i that are searched in a hop is given by:

$$n_i = (1-g)*W + g*i*W/P \qquad (4.6)$$

where *g* is a parameter that is assigned a value between *0* and *1*. We call this method "qualified JIT" and denote it QJIT(*g*), where *g* is the controllable parameter of the scheme. The parameter *g* divides the search spectrum in each OXC into two parts: a base part and an adjustable part. The base part has a fixed size of *(1-g)*W* wavelengths regardless of the priority level of the burst. The base part ensures that every type of bursts can search some number of wavelengths. The adjustable part has a size that depends on the priority level of the burst, and can reach a maximum size of *g*W* wavelengths for the highest priority level. For example, if the highest priority level is *P=5*, the size of the adjustable part is *0.2*g*W* for priority level *1*, *0.4*g*W* for priority level *2*, and so on. It should be noted that the high-level model of equation (4.2)-(4.5) is suitable for both JIT and JET, and the qualified JIT scheme is therefore suitable for JIT as well as JET. The qualified JIT scheme is very easy to implement and does not demand any major software or hardware resources in the OXCs. The priority level of the burst is easily passed from

one hop to the other hop along a lightpath via the control packet. Implementing the adjustable search for a free wavelength implied by equation (4.6) requires minor modification to the standard JIT channel allocation scheme; the adjustable search (i.e., searching in a space of size $g*i*W/P$) actually leads to a smaller average search time. There are two important remarks about equation (4.6).

Remark 1: when $g=0$, the adjustable part of equation (4.6) vanishes and the scheme becomes equivalent to the standard JIT scheme. In other words, QJIT$(g=0)$ is identical to the standard JIT scheme. As the value of g increases, data bursts with higher priority can get better treatment since the size of their adjustable search space increases and hence their "go through" probability gradually improves. It is obvious that larger value of g will be more effective in providing differentiated QoS services. However, higher values of g could lead to severely deteriorated performance for low priority classes and could adversely impact the overall throughput of the network. Ideally, we should choose a value of g that provides a good compromise between differentiated QoS and network throughput. The ideal value of g should provide tangible improvement in the QoS of high priority traffic without negatively impacting the overall throughput of the system.

Remark 2: the value n_i in equation (4.6) should be rounded to an integer number. For effective QoS differentiation, different values of i (i.e., different priorities) should be mapped to different integer values of n_i. This will ensure that a higher priority burst will have a search space strictly larger in size than that of a lower priority burst and will therefore have a smaller blocking probability. To satisfy this constraint, there should be a limit on the value of P (number of

supported priorities). Simply, the value of P should not exceed the value of the product $g*W$ so that different values of i in equation (4.6) map to different values of n_i. This is formally described by the following constraint on the number of priorities P that can be supported by the QJIT scheme.

$$P \leq g*W \qquad (4.7)$$

The QJIT scheme and the logic used in equation (4.6) were inspired by our earlier work on reducing the dropping probability of handoff requests in the base stations of cellular wireless networks [50]. Abstractly speaking, the adjustable term of equation (4.6) is a generalization of the guard channels that are exclusively dedicated to handoff requests in order to give them priority over new call cellular requests. Our extensive tests have shown that the QJIT scheme can improve the differentiated QoS performance of optical burst switched networks while maintaining the overall throughput of the network. Our performance tests presented in section 4.5 show the effectiveness of the QJIT scheme in providing improved QoS differentiated services in optical burst switched networks. It should be mentioned that while equation (4.6) gives the size of the search space for a given priority, it does not require a fixed set of wavelengths to be searched for that priority. In our QJIT scheme, the starting point of the search process is randomly selected. Even though the size of the search space is fixed, the set of wavelengths searched by QJIT is randomized. Unlike the schemes given in [78], by randomly selecting the sets for different priority bursts, the QJIT can maintain a healthy total network performance while improving QoS for higher priority bursts.

4.4 Prioritized random early dropping (PRED) scheme

Our second scheme adapts the concept of random early discard (RED) to the OBS environment and prioritizes the levels of discarding based on the priority levels of bursts. We call this scheme prioritized RED or PRED.

The RED concept [51] has received considerable interest in electronic packet switching networks and RED routers have been widely deployed in various commercial applications and in the Internet. There have been numerous studies that support or oppose RED [51-53], present schemes for tuning RED parameters [54], propose modified versions of RED [55], and develop analytical models for RED performance [56]. The basic idea of RED is that routers proactively discard incoming packets with probabilities that depend on the size of the router's queue. The TCP congestion control algorithm [57] reacts to lost packets by throttling the transmission rate of TCP senders. Studies have shown that well-configured RED routers have the potential to avoid severe congestion and improve the overall throughput while maintaining a small queuing delay within each router. The PRED scheme performs random proactive dropping different from the burst discarding mentioned in [72]. In [77], the authors proposed a priority based wavelength assignment scheme for all-optical networks. In their scheme, the higher priority requests can get the channel reserved by lower priority bursts. Unlike the scheme in [77], the dropping policy of PRED is not preemptive and is not triggered by the arrival of higher priority requests.

Our PRED differentiated QoS scheme for JIT uses proactive burst dropping with a discarding probability that decreases as the burst priority level increases. The goal of burst discarding in PRED is not to avoid congestion or throttle TCP senders as in RED (although these could be

positive side effects of PRED). Rather, PRED uses proactive discarding to improve the QoS of higher priority bursts at the expense of some deterioration of the QoS of lower priority bursts. A major difference between RED and PRED is that our PRED scheme restricts burst discarding to the original source node of the burst while RED allows any router in the path of a packet to proactively drop it. Specifically, all proactive discarding in PRED is done in the network access station (NAS) of the source node that generated the burst. This restricted mode of discarding has the advantage that the discarded bursts will not waste any bandwidth resources in the core of the optical networks.

Let α_i be the probability used by PRED at the source NAS to discard a newly incoming burst whose priority level is i (larger i means higher priority level). To improve QoS differentiation in OBS networks, the values of the discarding probability should satisfy the following constraint:

$$\alpha_1 > \alpha_2 > \ldots > \alpha_p \qquad (4.8)$$

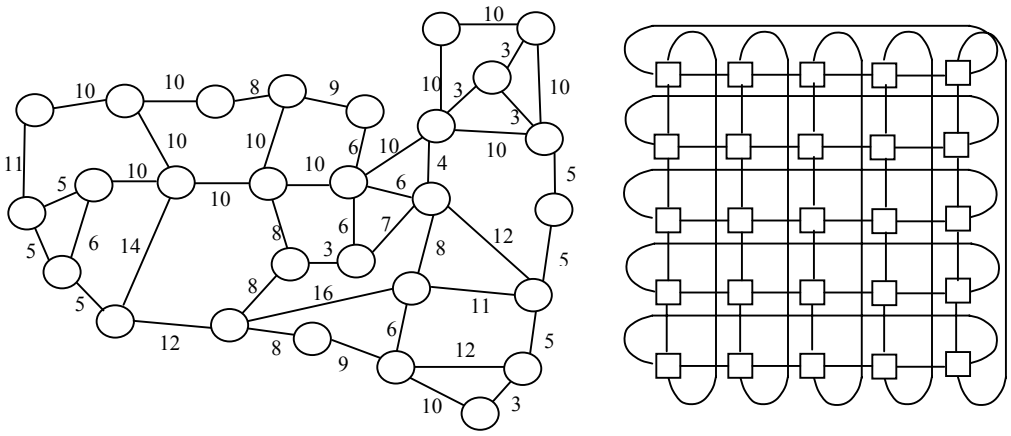
Where P is the number of priority levels in an optical burst switched network as explained in section 4.3. The proactive discarding of relation (4.8) is only applied to the local bursts assembled at this NAS. The NAS may also be servicing transit bursts that have come from some external OXC's and are being routed to other external OXC's. These transit bursts have already escaped the PRED proactive discarding in the originating NAS where they were assembled. These transit bursts have also already consumed some network bandwidth resources during their partial trip toward their destination. By proactively discarding local bursts and not transit bursts, more bandwidth will be available to transit bursts in each OXC. This increases the likelihood that

transit bursts will reach their destination without wasting the resources that they have already used prior to reaching the current OXC.

As in standard RED, the proactive discarding in PRED should not take place if the load on the OXC is not heavy. This is because at light loads, most bursts are expected to reach their destination successfully and the burst dropping probabilities for all priority levels are nearly zero. We have adopted a simple mechanism to disable/enable proactive discarding in PRED. In OBS, each NAS uses some buffers to hold assembled bursts until they are sent to the local OXC. PRED does not discard arriving bursts when the free space in these buffers is greater than or equal to some threshold. When the free buffer space is less than the threshold, new bursts are subjected to the prioritized probabilistic discarding of relation (4.8). Notice that when the network is lightly loaded, the amount of free buffer space will be relatively large and the PRED proactive discarding is disabled. Consequently, the actual discarding probabilities will be smaller than the probabilities α_i given in relation (4.8).

4.5 Performance evaluation results and analysis

Figure 4.1 shows the two network topologies that are used in our simulation (US LongHaul network with 28 nodes and 5x5 mesh-torus network).



LongHaul topology 5x5 mesh-torus topology

Figure 4.1. Two network topologies

In our simulation, a static lightpath between any two nodes is established using the shortest path first method as was done in [30, 56, 3]. Notice that the longest shortest path has 7 hops in the LongHaul topology and 4 hops in the 5x5 mesh-torus topology. The labels on the links of the LongHaul network represent the relative integer ratios of the lengths of the fiber cables of these links. For example, the delay of a link with label 5 is half the delay of a link with label 10. Similar to [56], the traffic used in our tests is uniformly distributed among all nodes. This means that all nodes have equal likelihood to be the source of a data burst and for a given source node all other nodes in the network have equal likelihood to be the destination node. The number of priority levels in our simulation tests is $P=5$ and the traffic is equally distributed among all 5 classes (except the scenario in Figure 4.14 and Figure 4.15, which have P=12 priority levels). Bursts with priority P have the highest priority and those with priority 1 have the lowest priority.

In our simulation tests, assembled bursts arrive according to a Poisson distribution with controllable arrival rate λ. For each burst, the source node and destination node are randomly selected as explained before. Our simulation tests used parameter values similar to those typically used in the literature: the cut through time in each OXC is 2.5 milliseconds, the link delay per hop is 3 milliseconds for the Mesh network, the burst length is 50 microseconds (equivalent to a burst of 250 Kbits at 5 Gbits/second), the control packet processing time t_p at each hop is 50 microseconds. For the LongHaul network shown in Figure 1, the delay of a link in milliseconds is 0.5 multiplied by the length label of that link. Thus the delay for a link with length 6 is 3 milliseconds. The number of wavelengths used in each OXC is W=40 (except Figure 14 and Figure 15 which use 12 wavelengths). Each point in the performance graphs reported in this chapter was obtained by averaging the results of 6 simulation tests using different randomly generated seeds. Each simulation was run for sufficiently long time to obtain stable statistics; the total number of bursts processed in each simulation test ranged from 4 million bursts at low arrival rates to 12 million bursts at high loads for the LongHaul network and from 10 million bursts at low arrival rates to 20 million bursts at high loads for the mesh-torus network. The unit of time (denoted *ut*) used in the graphs presented in this chapter is equal to 0.05 millisecond. Thus a load λ of 12 bursts/ut is equivalent to 60 Gbits/second.

4.5.1 Performance of QJIT

Figure 4.2 shows the burst drop probability for different priority levels at load 12 bursts/ut (60 Gbits/second) in the LongHaul network. Figure 4.3 shows the burst drop probability for different

priority levels at load 20 bursts/ut (100 Gbits/second) in the 5x5 mesh-torus network. Notice that the mesh-torus network has more links than the LongHaul network and it often has multiple shortest-path routes connecting the same source-destination pair. The mesh-torus network therefore requires higher total load than the LongHaul network to induce a certain level of congestion on the individual links. The horizontal axis in Figures 4.2 and 4.3 gives the value of the parameter *g*. Notice that the case *g*=0 corresponds to the standard JIT scheme. The figures show that the QoS for the highest priority levels (level 5 and 4) gradually improves as the value of *g* increases. The value *g=1* gives the largest difference of drop probability between level 5 and level 1.

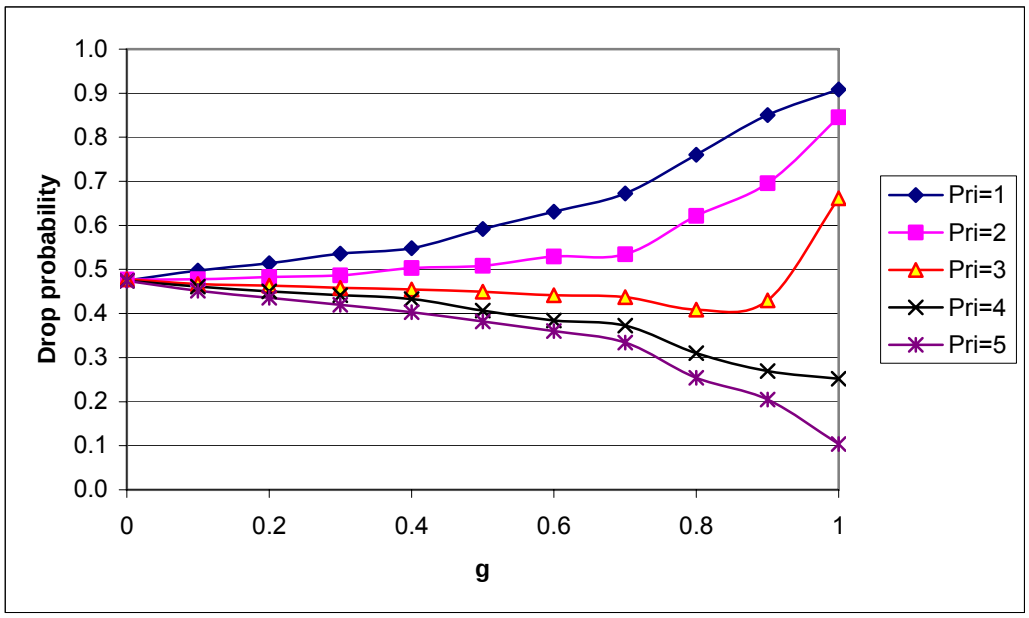


Figure 4.2. QJIT drop probabilities for different priority levels in LongHaul network (load =12)

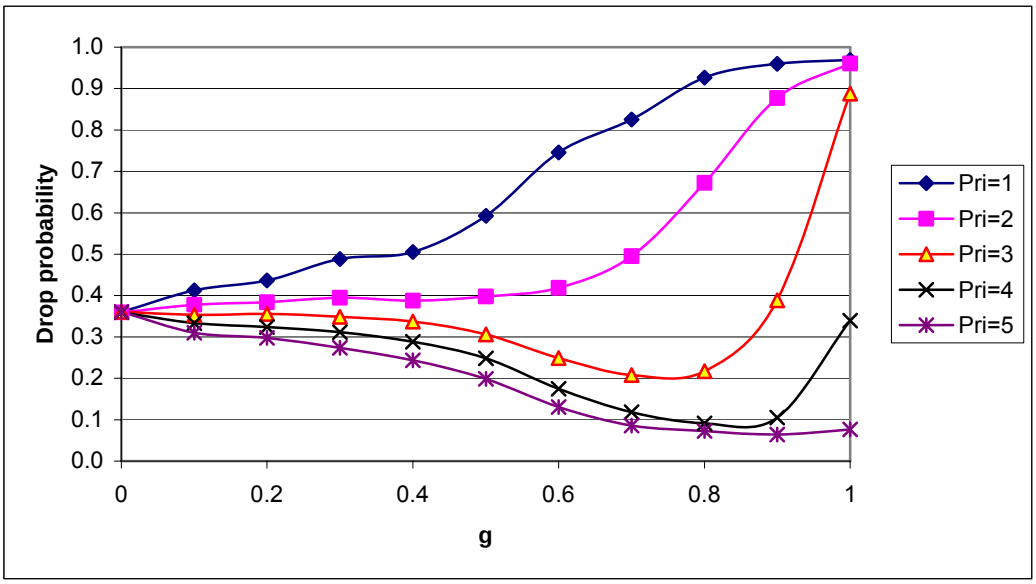


Figure 4.3. QJIT drop probabilities for different priority levels in 5x5 mesh-torus network (load =20)

Figures 4.4-4.6 show the QJIT drop probability in the LongHaul network for priority values 1, 3 and 5, respectively. For each priority value, the drop probability is plotted for different values of *g* and load λ. Figures 4.7-4.9 show the corresponding graphs for the Mesh network. For the lowest priority (priority 1), Figure 4.4 and Figure 4.7 show that the drop probability increases as the value of *g* increases. For the medium priority (priority 3), Figure 4.5 and 4.8 show that the burst drop probability does not change much as the value of *g* increases from 0 to around 0.8. When *g* reaches the value 0.8 there is a slight decrease in the drop probability for priority 3 then there is a significant increase in the drop probability as *g* increases to 1. Figure 4.6 and 4.9 show that as *g* increases, the drop probability for the highest priority (priority 5) decreases.

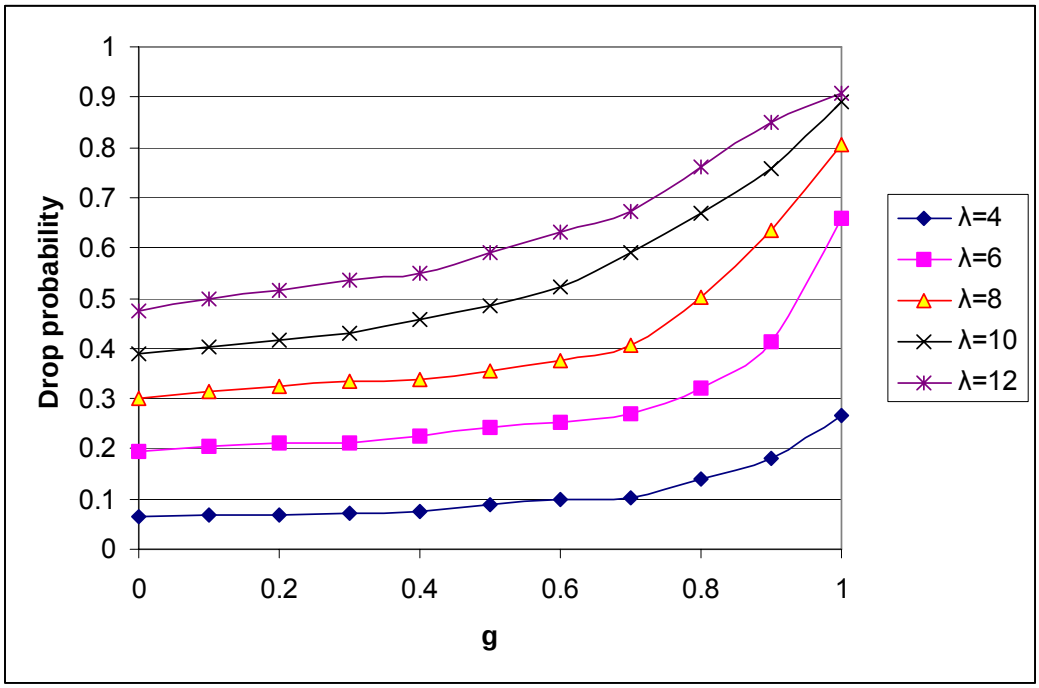


Figure 4.4. Drop probability for different loads with Priority=1, LongHaul network

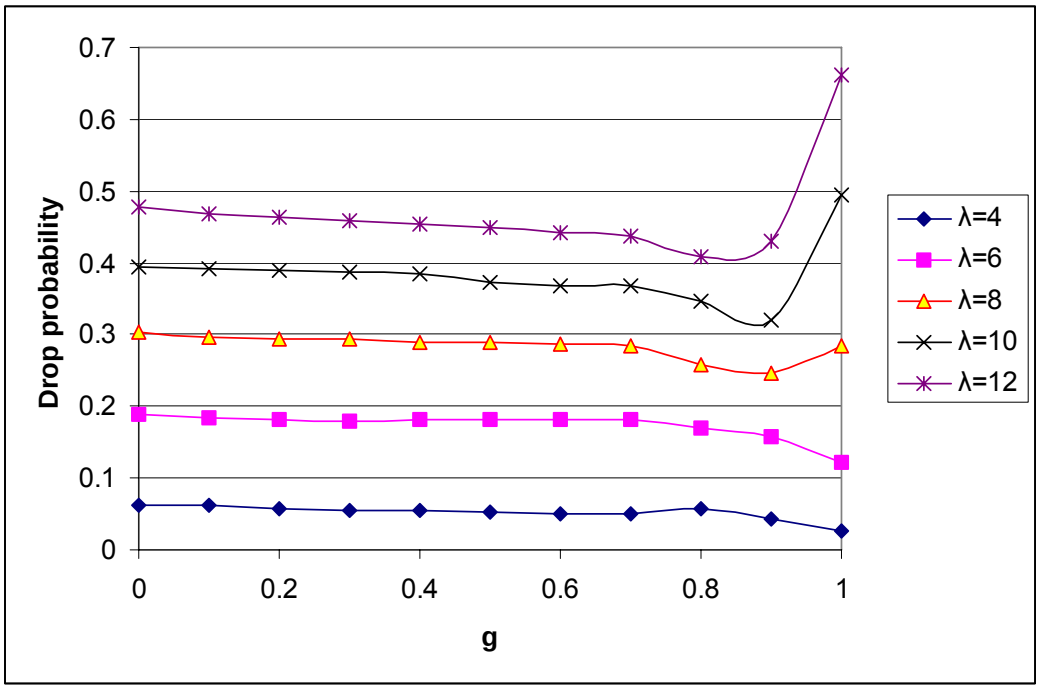


Figure 4.5. Drop probability for different loads with Priority=3, LongHaul network

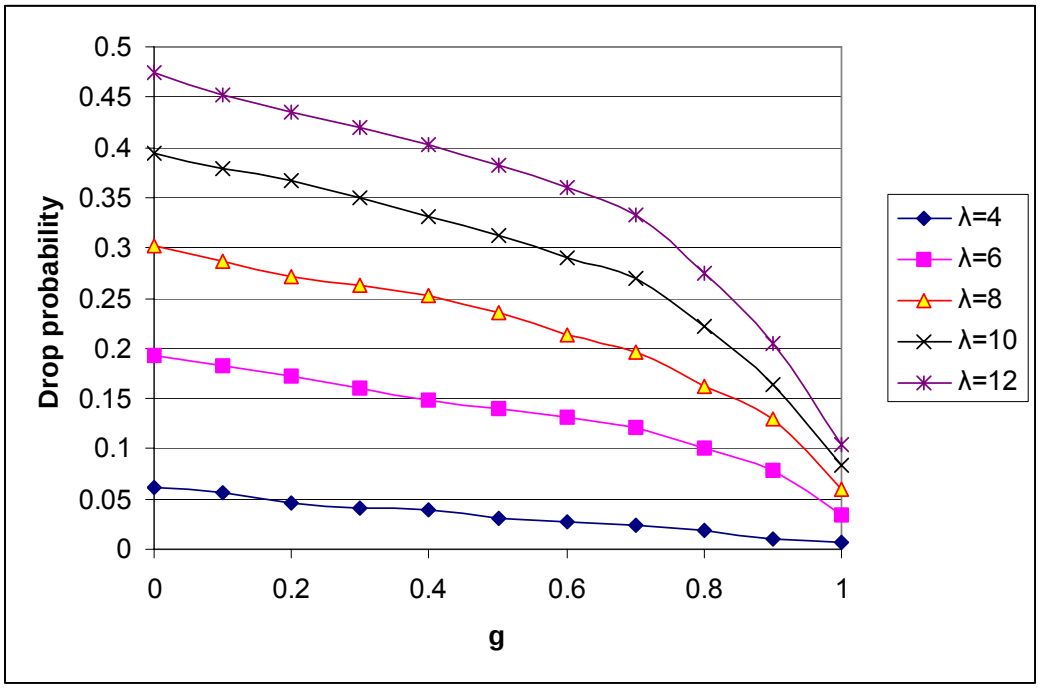


Figure 4.6. Drop probability for different loads with Priority=5, LongHaul network

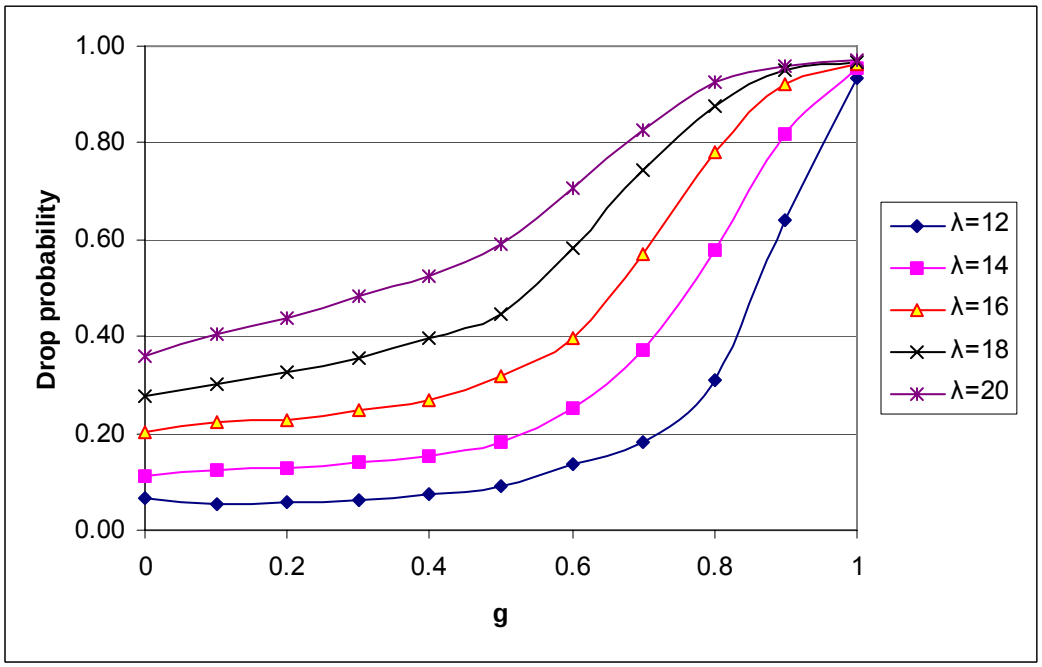


Figure 4.7. Drop probability for different loads with Priority=1, mesh-torus network

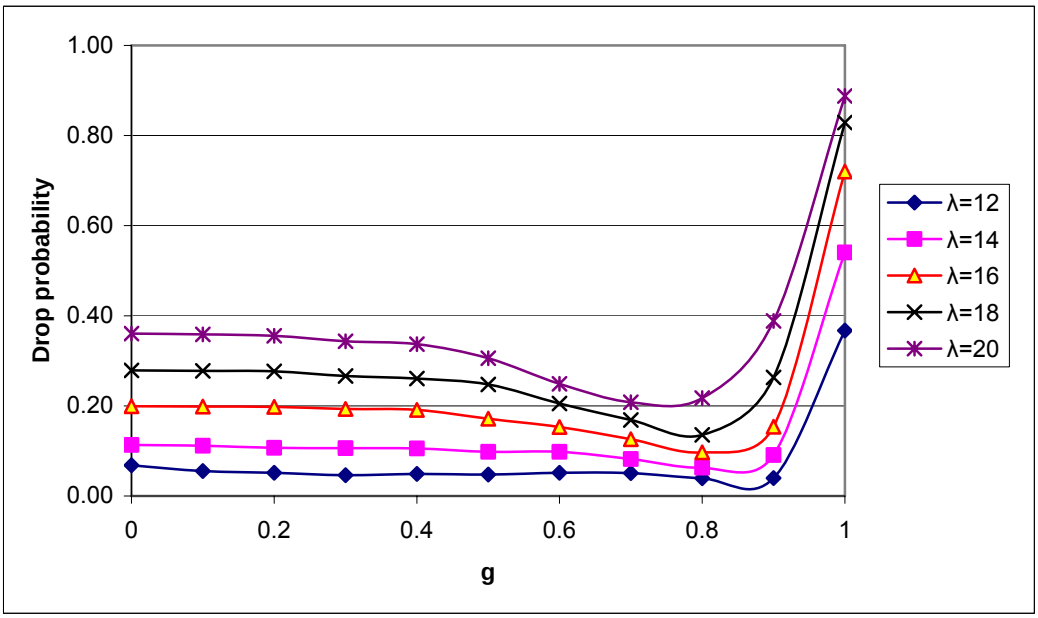


Figure 4.8. Drop probability for different loads with Priority=3, mesh-torus network

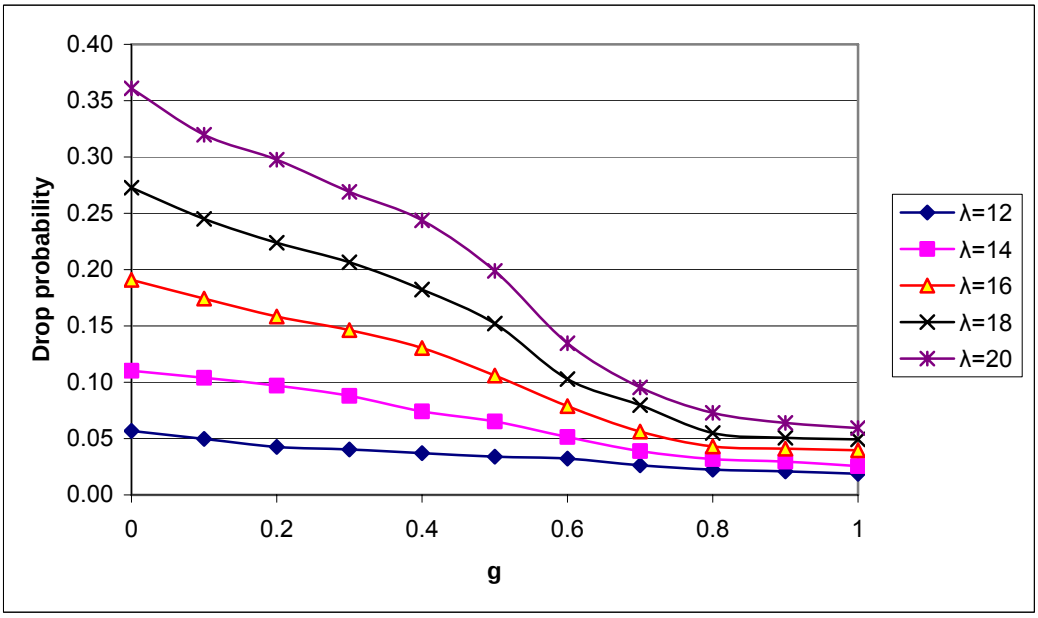


Figure 4.9. Drop probability for different loads with Priority=5, mesh-torus network

Figures 4.2-4.9 examined the QJIT drop probabilities for different priority levels and clearly showed that higher values of *g* are more effective in providing differentiated QoS. We now examine the overall throughput and the overall average drop probability (i.e., averaged over all priority levels). Figures 4.10 and 4.11 show the total throughput (in Gbits per unit time) at different values of *g* and different loads in the LongHaul and 5x5 mesh-torus networks, respectively. As evident from these graphs, the value *g*=1 degrades the throughput significantly. In general, values around *g*=0.5-0.6 give the best throughput performance while still providing noticeable differentiated QoS. Notice that the throughput around *g*=0.5 is almost the same as the throughput of standard JIT (i.e., when *g*=0).

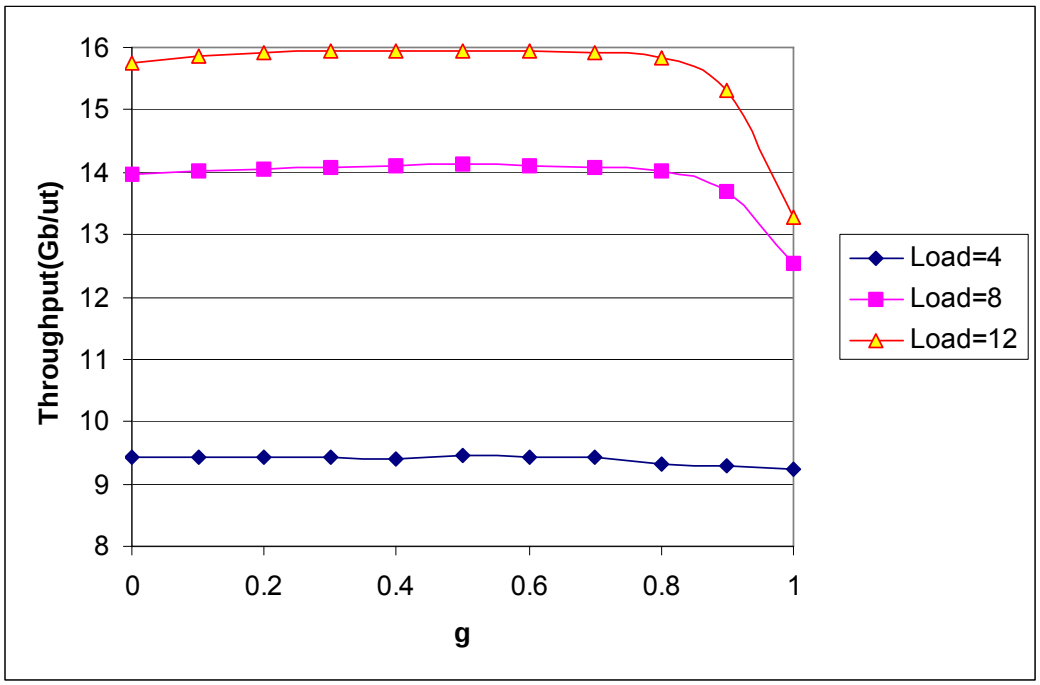


Figure 4.10. QJIT throughput for different values of *g*, LongHaul Network

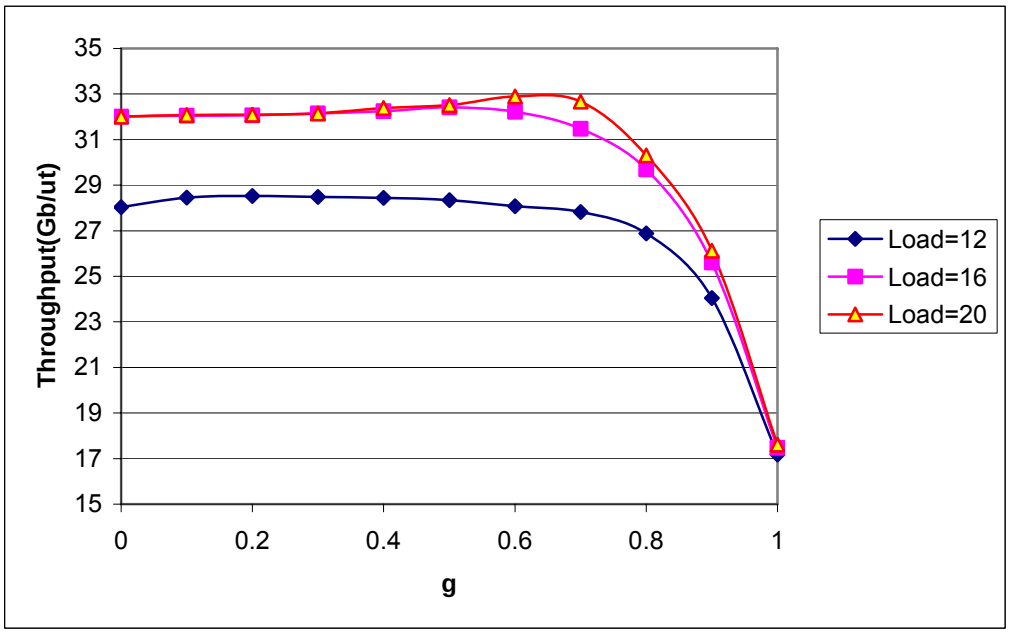


Figure 4.11. QJIT throughput for different values of *g*, mesh-torus network

Figures 4.12 and 4.13 give bar charts for the average burst drop probability (i.e., averaged over all priority levels) for the LongHaul and 5x5 mesh-torus networks, respectively. Again, the value *g*=1 has the worst performance and values around *g*=0.5 give the best overall average drop probability.

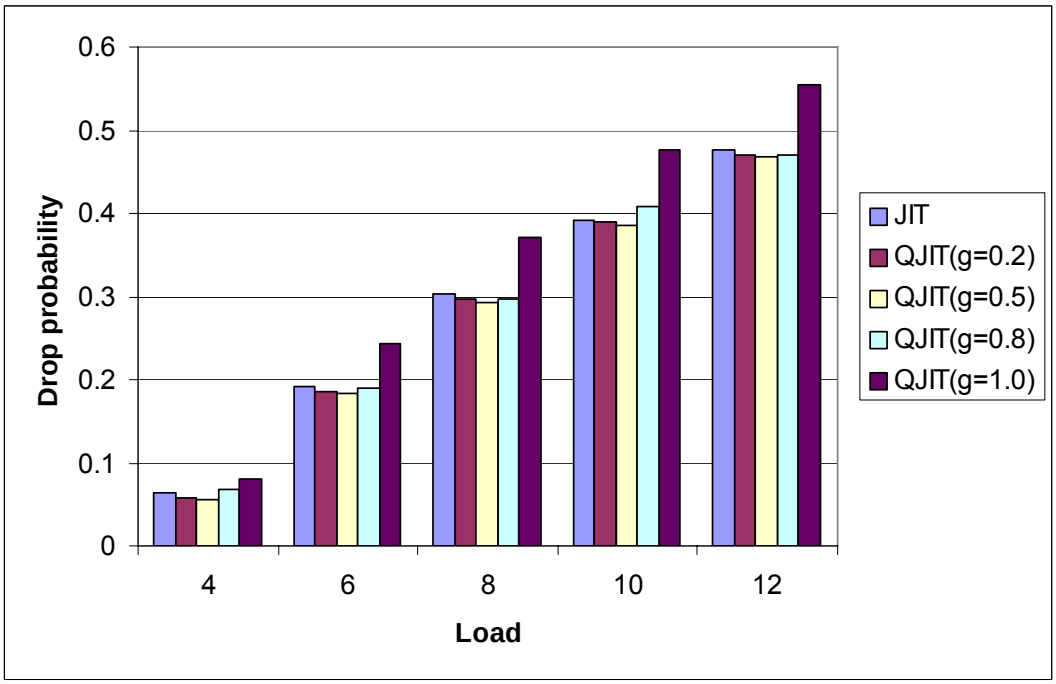


Figure 4.12. QJIT Overall drop probability for LongHaul network

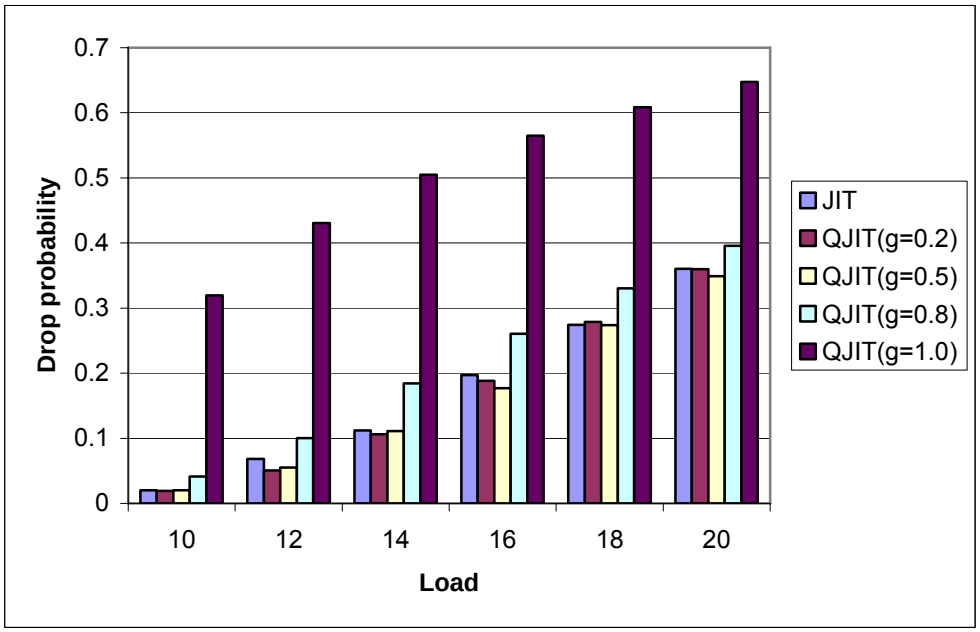


Figure 4.13. QJIT overall drop probability for mesh-torus network

The above results suggest that when using values of g close to 1, the QJIT(g) scheme provides strong QoS differentiation in OBS networks at the expense of degraded total network throughput performance. However, using values of g in the neighborhood of 0.5 offers definite practical advantage compared to standard JIT, namely, the throughput of the system is kept at normal levels and good QoS differentiation among priority levels is achieved.

As explained earlier in relation (4.7), the number of supported priorities P should not exceed $g*W$. Since the above results show that g=0.5 provide a good compromise between QoS differentiation and network throughput, the number of supported priorities P should be less than $0.5*W$. The smaller the value of P we use the better QoS differentiation we get from the QJIT scheme. In all previous graphs, we used P=5 and W=40 and therefore constraint (4.7) was well satisfied. If the value of P exceeds $g*W$, the QJIT scheme will not be able to provide strict differentiation between every pair of priorities. This is illustrated in Figures 4.14 and 4.15 which use P=12, W=12 and g=0.5. The value of P is double the value of $g*W$ and equation (4.6) produces the same value of n_i for two values of the priority i. As show in Figure 4.14 and 4.15, priority 2 and priority 3 have the same drop probability and therefore the same QoS. Similarly, priority 4 and priority 5 have the same QoS, priority 7 and priority 8 have the same QoS, and so on. Assuming the value of g is 0.5, the number of supported priorities P should preferably be less than 0.5*W. For example, if W=8, the maximum number of supported priorities P is recommended to be 3 or less and for W=16 it is recommended to be 5 or less.

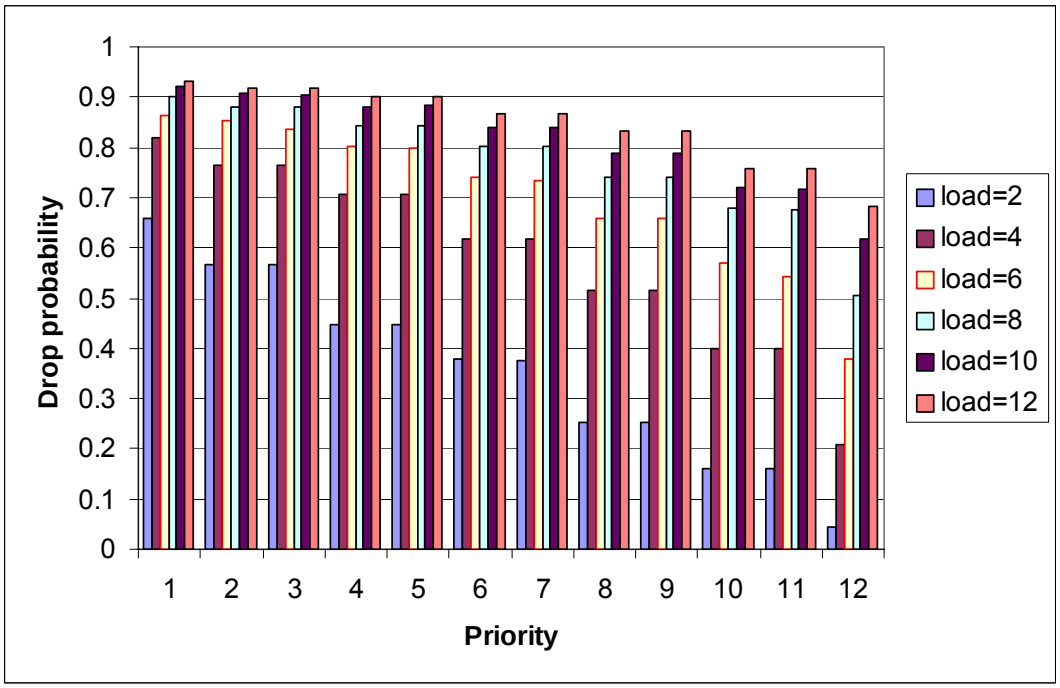


Figure 4.14. Drop probability for LongHaul network with W=12, P=12 and g=0.5

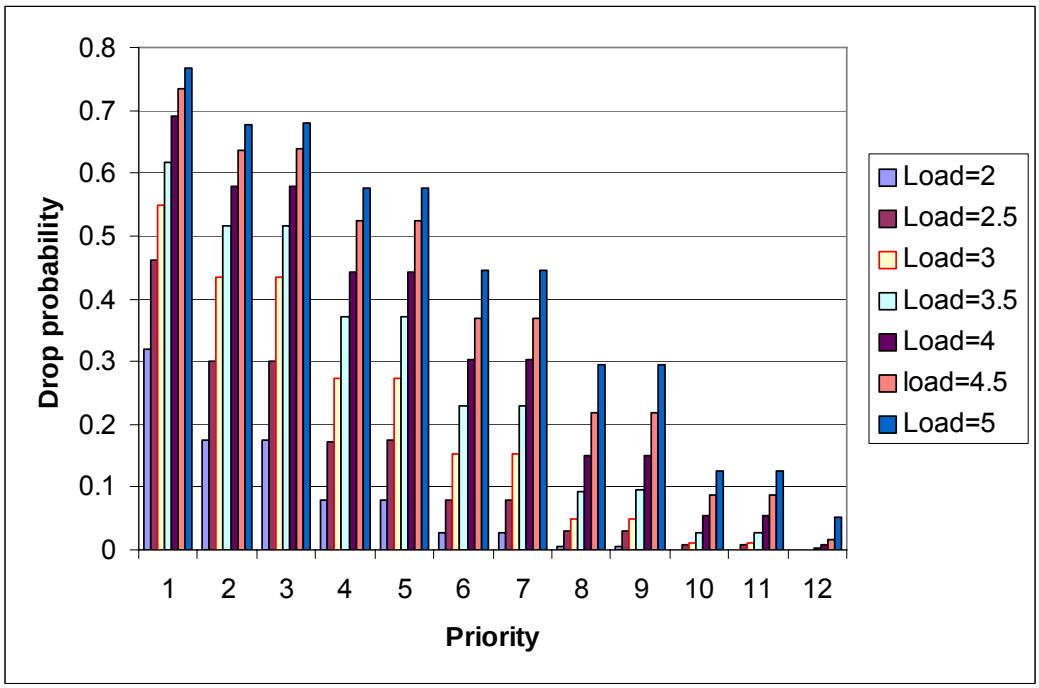


Figure 4.15. Drop probability for mesh-torus network with W=12, P=12 and g=0.5

Table 4.1 shows the distribution of the drop probability for different priority levels at different loads for standard JIT (i.e., QJIT with *g*=0) in the LongHual network. Tables 4.2, 4.3, 4.4 and 4.5 give similar distributions for QJIT(g) with parameter value *g*=0.2, *g*=0.5, *g*=0.8, and *g*=1.0, respectively. The corresponding drop probability distributions for the 5x5 mesh-torus network are given in table 6 for standard JIT and tables 4.7-4.10 for QJIT(g) with different values of g.

Table 4.1. Drop probability distribution in standard JIT (*g=0*), LongHaul

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
4	0.0514	0.0501	0.0514	0.0513	0.0513
6	0.2018	0.2001	0.2025	0.2064	0.2090
8	0.3038	0.2956	0.3022	0.2942	0.3019
10	0.3991	0.3995	0.3992	0.3993	0.3931
12	0.4754	0.4760	0.4759	0.4750	0.4735

Table 4.2. Drop probability distribution in QJIT(*g=0.2*), LongHaul

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
4	0.0674	0.0644	0.0565	0.0465	0.0421
6	0.2040	0.1897	0.1765	0.1632	0.1508
8	0.3350	0.3159	0.2955	0.2762	0.2460
10	0.4322	0.4114	0.3905	0.3650	0.3348
12	0.5370	0.4925	0.4763	0.4464	0.4206

Table 4.3. Drop probability distribution in QJIT(*g=0.5*), LongHaul

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
4	0.0795	0.0591	0.0459	0.0313	0.0220
6	0.2467	0.2119	0.1779	0.1474	0.1217
8	0.3846	0.3334	0.2917	0.2540	0.2217
10	0.4844	0.4246	0.3820	0.3378	0.2947
12	0.6122	0.5261	0.4598	0.4131	0.3612

Table 4.4. Drop probability distribution in QJIT(*g=0.8*), LongHaul

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
4	0.1560	0.0906	0.0487	0.0279	0.0145
6	0.3240	0.2462	0.1759	0.1284	0.0928
8	0.5245	0.3646	0.2687	0.2131	0.1550
10	0.6904	0.4874	0.3358	0.2666	0.2019
12	0.7729	0.6252	0.4165	0.3014	0.2320

Table 4.5. Drop probability distribution in QJIT*(g=1.0*), LongHaul

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
4	0.2745	0.0988	0.0326	0.0124	0.0061
6	0.6935	0.3497	0.1132	0.0553	0.0295
8	0.8612	0.6437	0.2741	0.1091	0.0625
10	0.9247	0.8224	0.5249	0.1535	0.0724
12	0.9582	0.8903	0.6831	0.2665	0.0891

Table 4.6. Drop probability distribution in standard JIT(*g=0.0*), 5x5 mesh-torus

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
10	0.0174	0.0175	0.0171	0.0173	0.0173
12	0.0704	0.0706	0.0707	0.0706	0.0710
14	0.1133	0.1103	0.1134	0.1133	0.1103
16	0.1979	0.1951	0.1969	0.1969	0.1984
18	0.2758	0.2706	0.2757	0.2764	0.2726
20	0.3561	0.3562	0.3579	0.3560	0.3585

Table 4.7. Drop probability distribution in QJIT(*g=0.2*), 5x5 mesh-torus

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
10	0.0210	0.0206	0.0190	0.0175	0.0169
12	0.0590	0.0543	0.0516	0.0446	0.0426
14	0.1280	0.1130	0.1070	0.0971	0.0871
16	0.2295	0.2028	0.1909	0.1684	0.1503
18	0.3310	0.2999	0.2821	0.2511	0.2294
20	0.4363	0.3845	0.3558	0.3239	0.2976

Table 4.8. Drop probability distribution in QJIT(*g=0.5*), 5x5 mesh-torus

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
10	0.0295	0.0221	0.0193	0.0155	0.0153
12	0.0895	0.0654	0.0475	0.0391	0.0340
14	0.1808	0.1321	0.0981	0.0798	0.0653
16	0.3181	0.2191	0.1691	0.1255	0.1032
18	0.4475	0.3184	0.2475	0.1959	0.1602
20	0.5929	0.3978	0.3060	0.2477	0.1988

Table 4.9. Drop probability distribution in QJIT(*g=0.8*), 5x5 mesh-torus

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
10	0.1205	0.0387	0.0175	0.0165	0.0133
12	0.3118	0.1004	0.0393	0.0268	0.0224
14	0.5785	0.2140	0.0629	0.0385	0.0318
16	0.7812	0.3312	0.0971	0.0541	0.0430
18	0.8750	0.5149	0.1359	0.0711	0.0522
20	0.9268	0.6726	0.2175	0.0910	0.0729

Table 4.10. Drop probability distribution in QJIT(*g=1.0*), 5x5 mesh-torus

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
10	0.8840	0.5484	0.1229	0.0245	0.0165
12	0.9347	0.8243	0.3273	0.0426	0.0271
14	0.9560	0.9238	0.5405	0.0734	0.0363
16	0.9615	0.9509	0.7200	0.1387	0.0490
18	0.9661	0.9600	0.8288	0.2250	0.0605
20	0.9693	0.9607	0.8877	0.3389	0.0769

4.5.2 Performance of PRED

As explained in section 4.4, PRED uses proactive discarding at the source NAS with probability α_k to discard bursts that have priority level k. Unlike the QJIT(g) scheme which has a single

parameter *g,* the PRED scheme has *P+1* parameters (*P* discarding probabilities and the threshold on the size of free buffer space used to enable/disable proactive discarding). However, the constraint represented by relation (4.8) greatly simplifies the parameter tuning process.

Figures 4.16, 4.18, and 4.20 show the drop probability distribution for four scenarios with different proactive dropping parameters in the LongHaul network. The empty buffer threshold used in these tests is 10% of the total buffer space. Notice that the probabilistic discarding is disabled when the total number of free buffers is greater than the threshold. Therefore the actual rate of proactive discarding is lower than the values of the discarding probabilities α_k. The overall average drop probability (i.e., averaged over all priority levels) corresponding to the results of Figures 4.16, 4.18, and 4.20 are shown in Figures 4.17, 4.19, and 4.21, respectively. From Figures 4.16, 4.18, and 4.20, we can easily see that the PRED scheme improves the level of QoS differentiation compared to the standard JIT scheme. As the intervals among the discarding probabilities α_k increase, the drop probability difference for different priority levels also increases. Figures 4.17, 4.19, and 4.21 show that as the QoS performance is improved by the PRED scheme, there is no negative impact on the overall drop probability. Similar results for the 5x5 mesh-torus topology are shown in Figures 4.22-4.27.

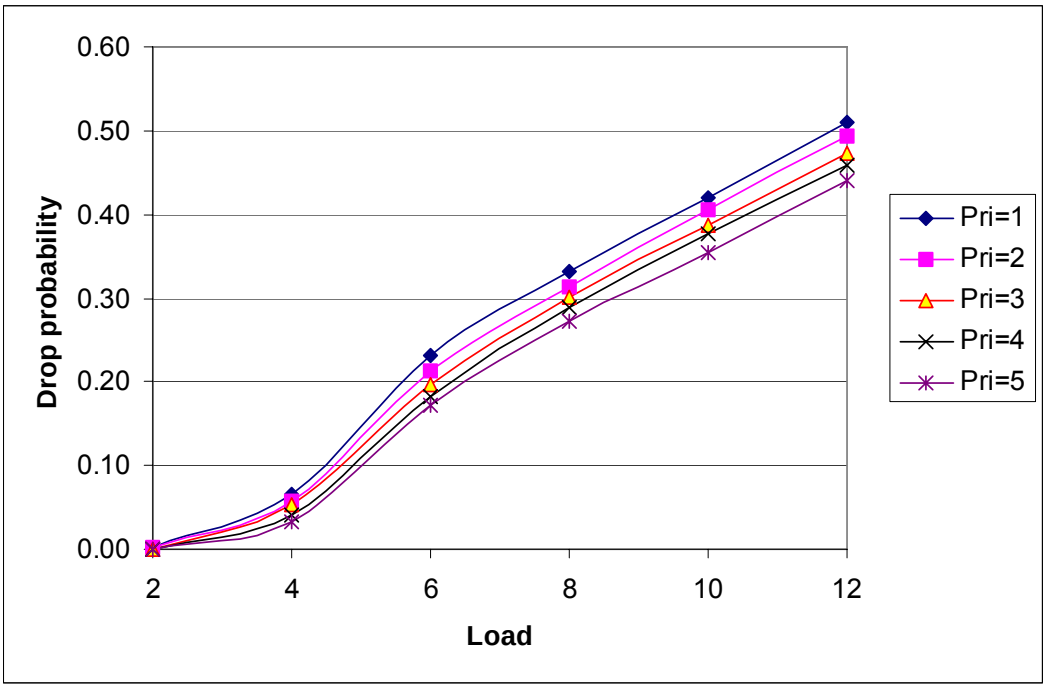


Figure 4.16 . Drop probability distribution using PRED for LongHaul network.

The values α_k of the proactive burst drop probabilities used in the source NAS are *0.8, 0.6, 0.4, 0.2*, and *0.0* for priority levels *1, 2, 3*, *4* and *5,* respectively.

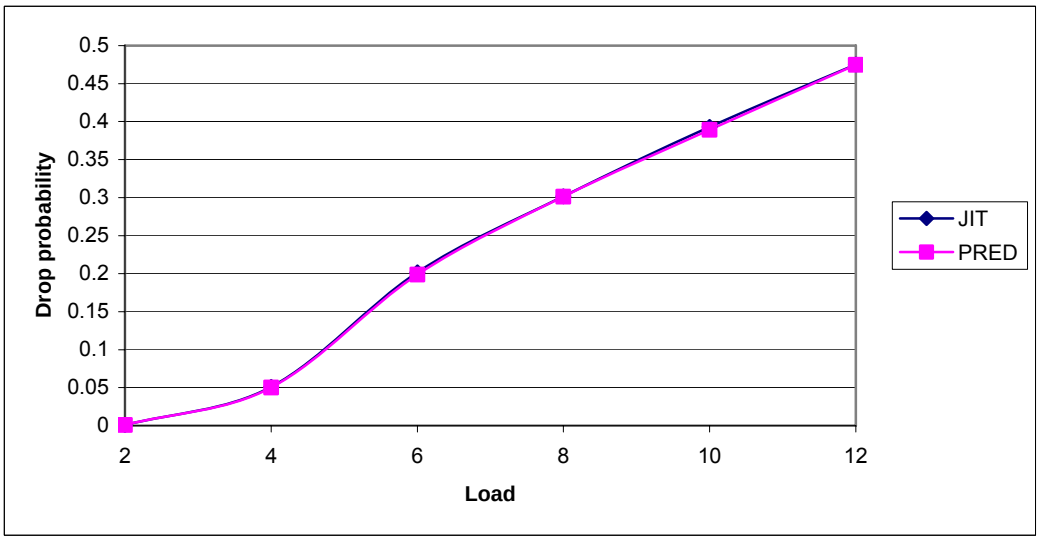


Figure 4.17. Overall average drop probability corresponding to Figure 16.

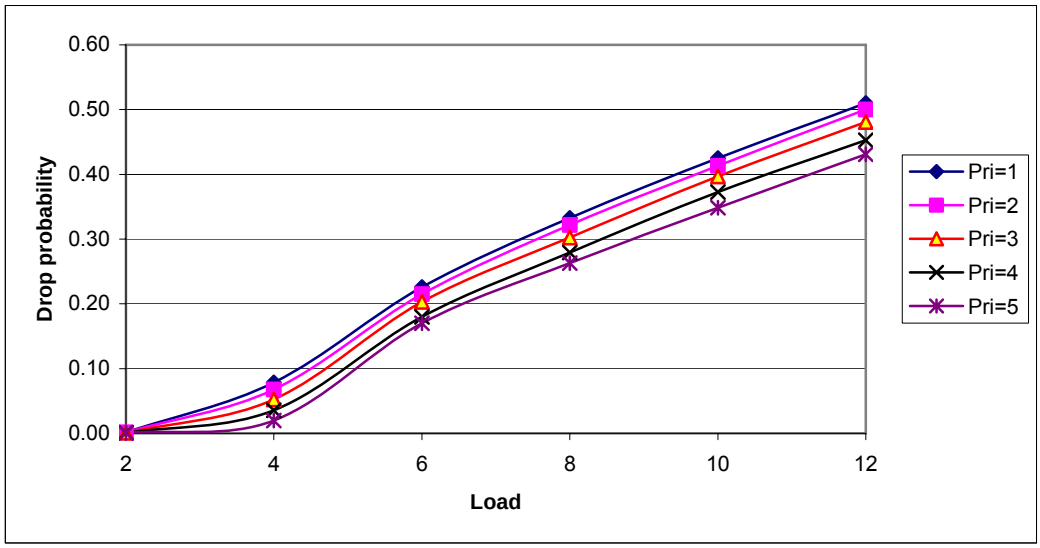


Figure 4.18. Drop probability distribution using PRED for the LongHaul network.

The values α_k of the proactive burst drop probabilities used in the source NAS are *1.0, 0.85, 0.55, 0.25*, and *0.0* for priority levels *1, 2, 3*, *4* and *5,* respectively.

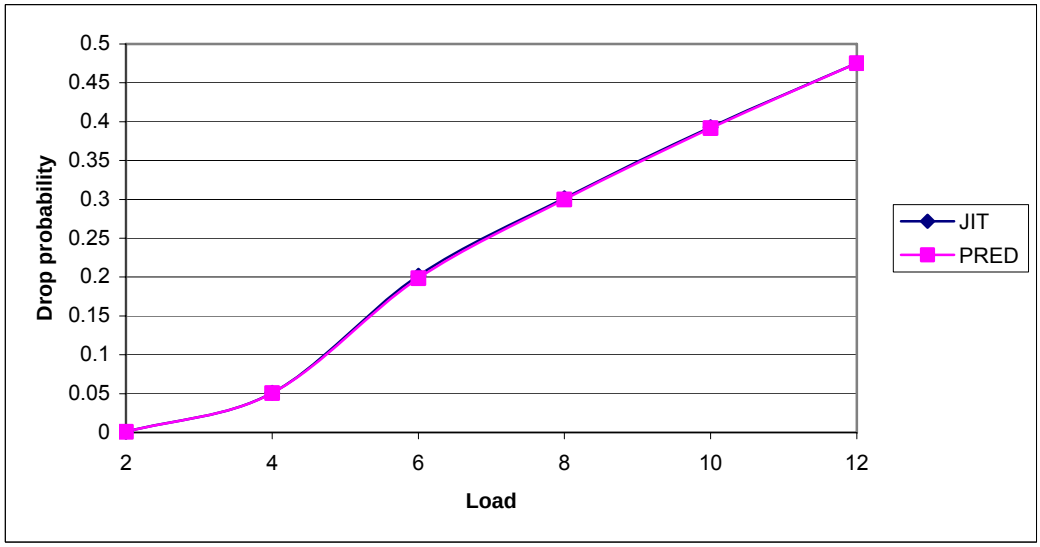


Figure 4.19. Overall average drop probability corresponding to Figure 4.18.

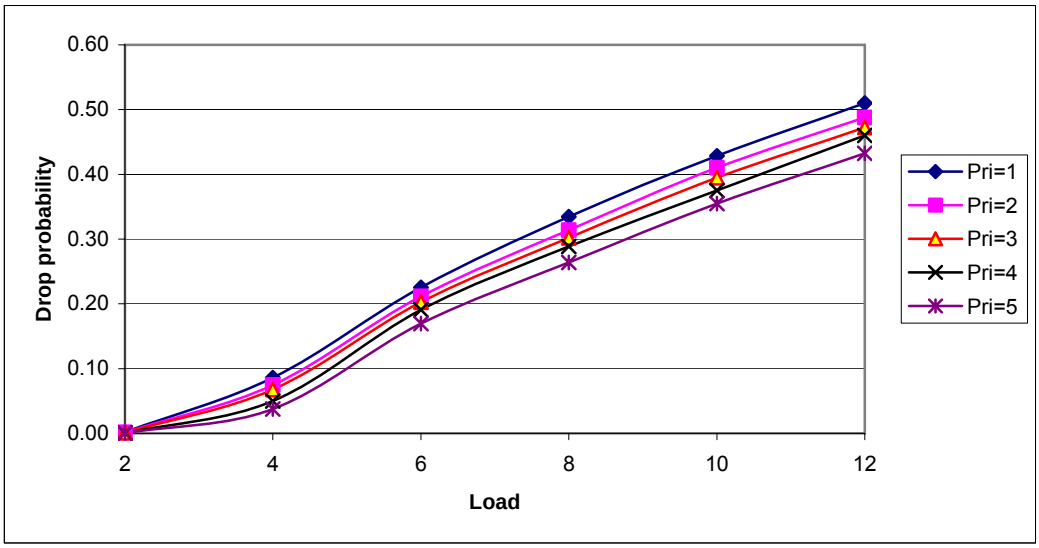


Figure 4.20. Drop probability distribution using PRED for the LongHaul network.

The values α_k of the proactive burst drop probabilities used in the source NAS are *0.9, 0.7, 0.5, 0.3*, and *0.0* for priority levels *1, 2, 3*, *4* and *5,* respectively.

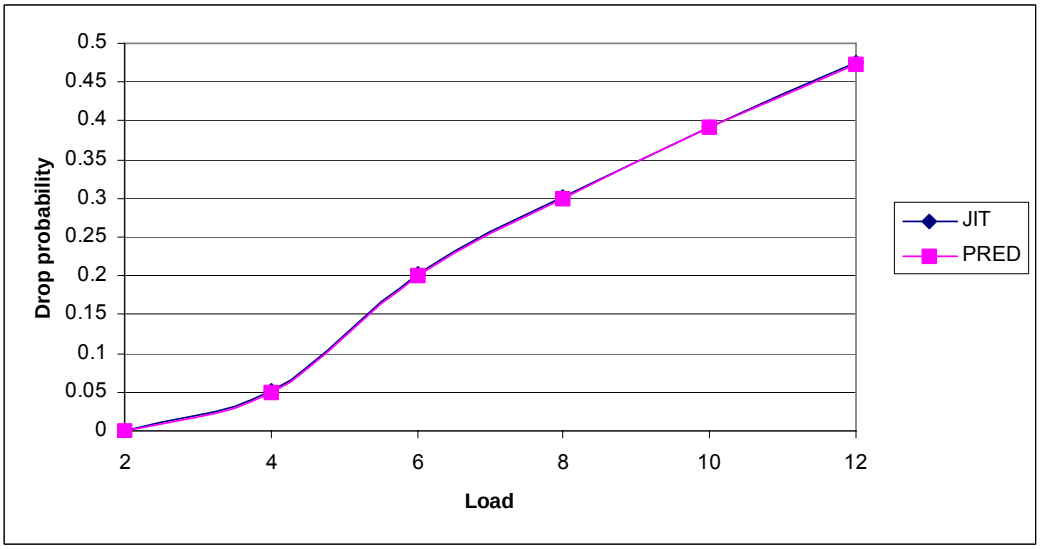


Figure 4.21. Overall average drop probability corresponding to Figure 4.20.

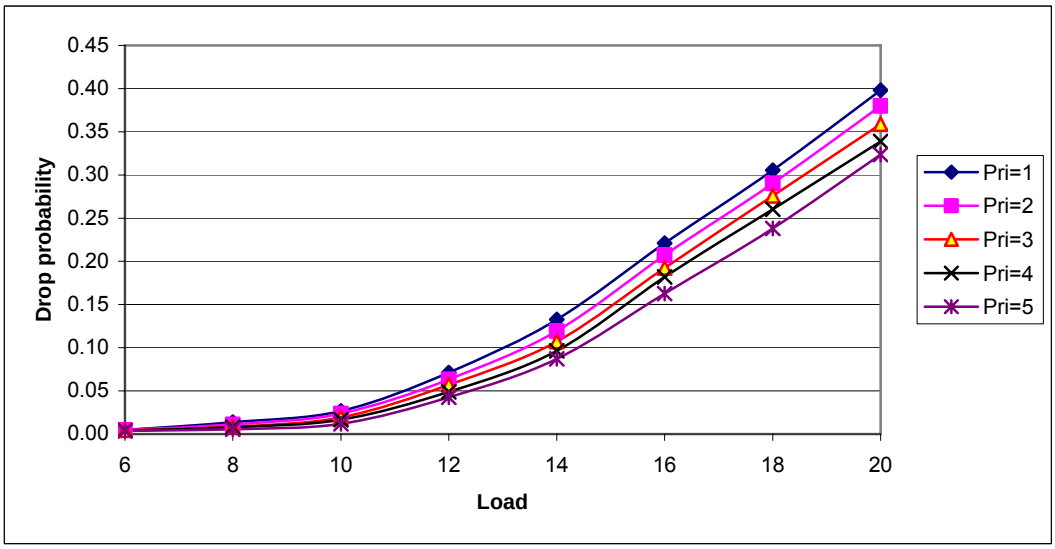


Figure 4.22. Drop probability distribution using PRED for the mesh-torus network.

The values α_k of the proactive burst drop probabilities used in the source NAS are *0.6, 0.45, 0.3, 0.15*, and *0.0* for priority levels *1, 2, 3*, *4* and *5,* respectively.

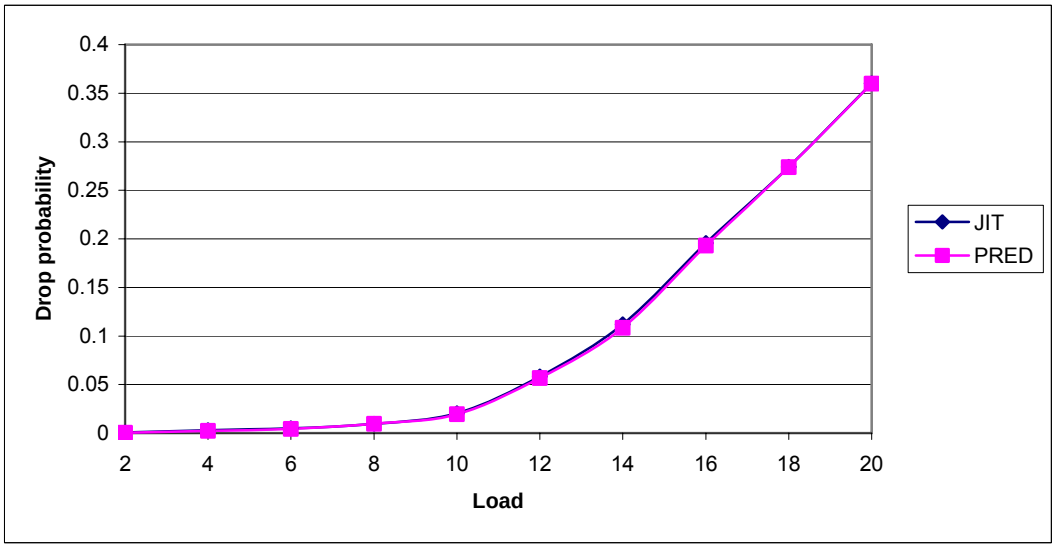


Figure 4.23. Overall average drop probability corresponding to Figure 4.22.

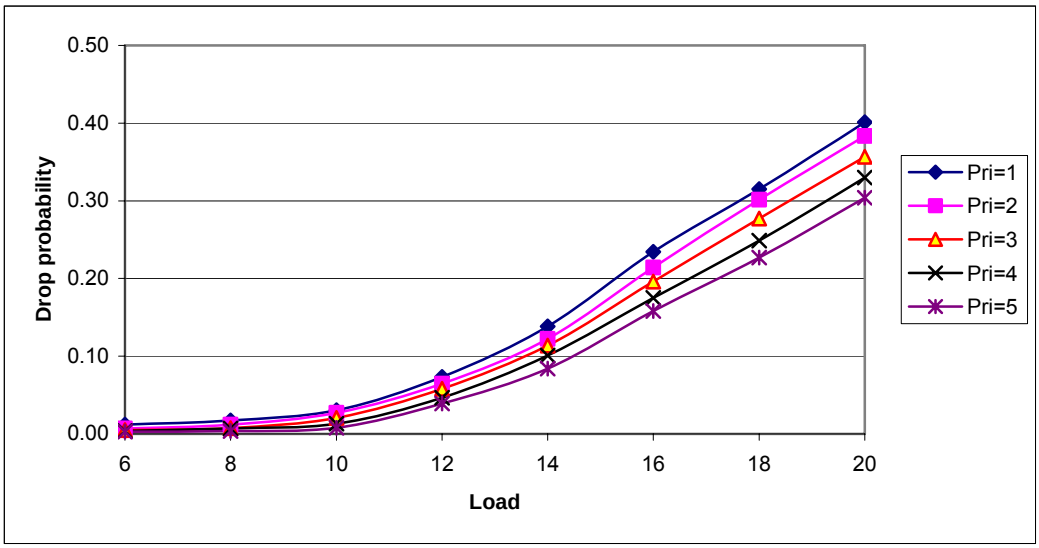


Figure 4.24. Drop probability distribution using PRED for the mesh-torus network. The values α_k of the proactive burst drop probabilities used in the source NAS are *0.8, 0.6, 0.4, 0.2*, and *0.0* for priority levels *1, 2, 3*, *4* and *5* respectively.

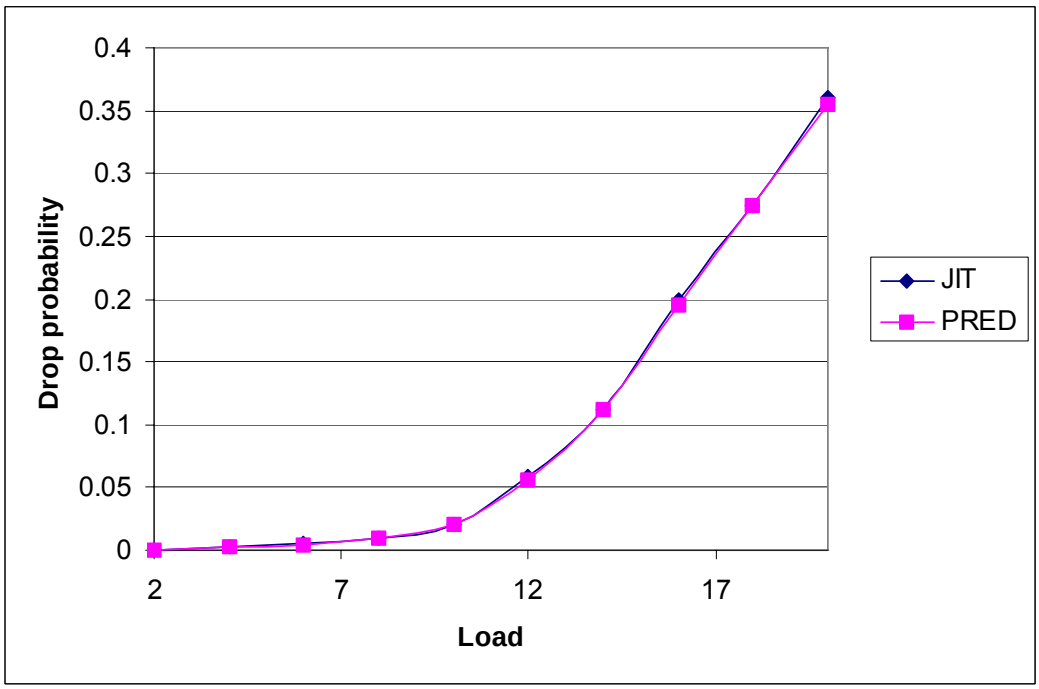


Figure 4.25. Overall average drop probability corresponding to Figure 4.24.

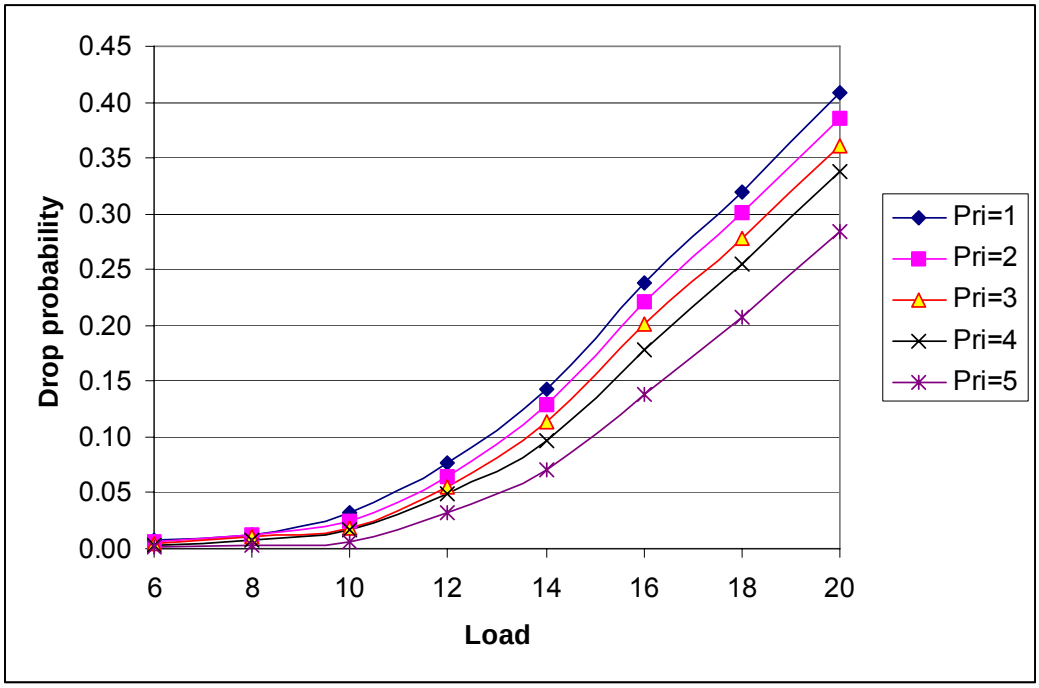


Figure 4.26. Drop probability distribution using PRED for mesh-torus network

The values α_k of the proactive burst drop probabilities used in the source NAS are *1, 0.8, 0.6, 0.4*, and *0.0* for priority levels *1, 2, 3*, *4* and *5* respectively.

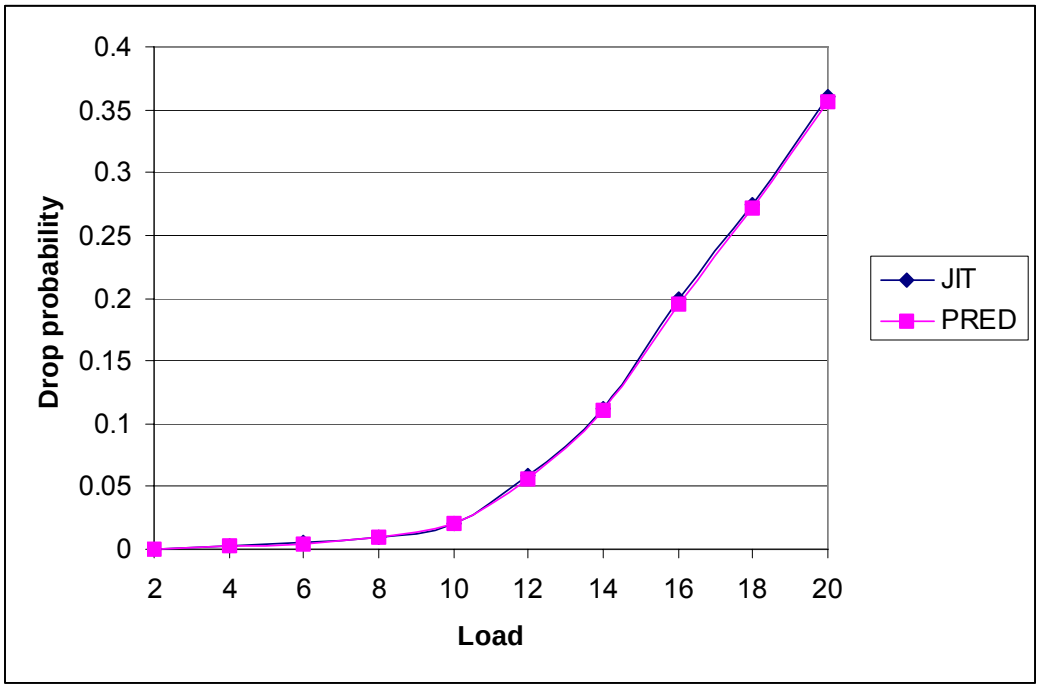


Figure 4.27. Overall average drop probability corresponding to Figure 4.26.

Table 4.11 shows the PRED burst drop probability for different loads using five priority levels in the LongHaul topology. The values of the proactive burst drop probabilities used in the source NAS are *0.8, 0.6, 0.4, 0.2*, and *0.0* for priority levels *1, 2, 3*, *4* and *5* respectively. The empty buffer threshold used in this test is 10% of the total buffer space. Notice that the probabilistic discarding is disabled when the total number of free buffers is greater than the threshold. Therefore the actual rate of proactive discarding is lower than the values of the discarding probabilities α_k. The overall average drop probability (averaged over all priority levels) corresponding to the results of Table 4.11 is shown in Figure 4.17. From Table 4.1 (for standard JIT), Table 4.11 and Figure 4.17, we can easily see that the PRED scheme has improved the QoS differentiation compared to the standard JIT scheme (i.e., has reduced the drop probability of

higher priority bursts), almost without any negative impact on the overall drop probability (in this experiment, we found that both PRED and standard JIT have almost the same throughput). The results for 5x5 mesh-torus topology and some LongHaul topology scenarios are similar and will not be given in this dissertation.

Table 4.11. Drop probability distribution using PRED for LongHaul network

Load	Pri= 1	Pri= 2	Pri=3	Pri= 4	Pri= 5
4	0.0591	0.0549	0.0497	0.0447	0.0431
6	0.2074	0.2001	0.1871	0.1799	0.1603
8	0.3398	0.3200	0.2997	0.2822	0.2575
10	0.4471	0.4140	0.3790	0.3524	0.3206
12	0.5499	0.4996	0.4677	0.4307	0.3920

CHAPTER 5. USING CONSTRAINED PREEMPTION TO IMPROVE DROPPING FAIRNESS IN OPTICAL BURST SWITCHED NETWORKS

5.1 Introduction

In recent years, wavelength division multiplexed (WDM) optical networks have received considerable attention and many researchers proposed schemes to improve the throughput and resource management of these networks [3, 26, 44, 87]. To be able to support the burst traffic on Internet, optical burst switching (OBS) has been proposed to provide fine bandwidth granularity and improve the utilization of WDM optical networks [30]. The basic idea of OBS is that the traffic is formed as units of data bursts. Since the data bursts are much larger than IP packets, OBS is envisioned to combine the advantages of optical circuit switching and optical packet switching.

In optical burst switched networks, a control packet is sent out ahead of the data burst to make channel reservation and configure each optical cross connect (OXC) along the lightpath of the burst. Just-in-time (JIT) signaling and just-enough-time (JET) signaling [30, 32] are the two main scheduling protocols for OBS networks. Both protocols have the shortcoming of the *beat down* unfairness problem, namely, the bursts with longer lightpaths have higher dropping probability than the bursts with shorter lightpaths. Due to the difficulty of optical buffering, schemes proposed to solve the beat down problem in electronic networks are not suitable for implementation in OBS optical networks. In this chapter, we propose a new scheme for the beat

down problem in OBS networks based on constrained burst preemption. Compared to previous work, the new scheme gives improved fairness without degrading network throughput. .

The rest of this chapter is organized as follows. In section 5.2, we review the related schemes proposed in the literature for optical burst switched networks. The burst preemptive strategy is analyzed in section 5.3. In section 5.4, our constrained preemption scheme is presented. The simulation results and performance comparisons are presented and analyzed in section 5.5.

5.2 Related works

Optical burst switching (OBS) has been proposed in recent years to achieve the balance of circuit switching and packet switching [26]. Two important scheduling protocols of OBS networks are just-in-time (JIT) and just-enough-time (JET) [30, 32]. Generally, the JIT protocol is simpler than JET protocol. Compared with JIT protocol, JET attempts to utilize more information of the duration of burst transmission in order to schedule the cross-connect settings more efficiently in each OXC. A number of papers have focused on improving different performance aspects of OBS networks [49, 60, 67, 75]. In [49], the authors proposed a merit-based scheduling algorithm based on JET signaling for optical burst switched networks. In that algorithm, a priority-based preemptive method is used to realize channel reservation along the lightpath of the data burst. The results show that the preemptive scheme improves the overall blocking performance (and hence throughput) of the network. In [67], preemptive multi-class wavelength reservation is used to provide differentiated services.

The problem of improving fairness in OBS networks has received less attention than the problem of improving throughput. The term "fairness" has been used to address different issues in different types of optical networks. For example, in optical circuit switching, the capacity unfairness problem [44] applies to networks with traffic grooming capability, i.e., the capability of multiplexing and switching lower rate traffic streams onto higher capacity wavelengths. In optical burst switching networks, data bursts traveling through longer lightpaths have higher dropping probabilities than bursts with shorter lightpaths. This type of unfairness is well known in electronic packet switching networks and is often referred to as the "*beat down*" problem. Fair Buffer Allocation and Selective Packet Dropping [47] are two of the methods proposed to deal with the "*beat down*" problem in electronic packet switching networks. In WDM and OBS networks, the unfairness problem has often been discussed as a secondary consideration [45, 46, 48, 63]. For example, in [63], the authors mainly discuss the impact of the offset time on the performance of the network and the differentiated QoS that can be given to selected bursts. In [48], an OBS reservation scheme is proposed for OBS networks operating under the wavelength continuity constraint (i.e., no converters). The scheme uses the backward reservation paradigm in which a probe control message propagates from the source to destination then a reservation message travels back from the destination to the source before the data burst can be transmitted. The simulation results reported in [48] uses a three-node tandem network (i.e., maximum of two hops) and a total of three connections (two 1-hop and one 2-hop). The results show that fairness for the 2-hop connection has improved at the expense of a slight increase in the overall mean blocking probability of the three connections. The authors in [45] proposed a deflection routing algorithm and showed that the blocking probabilities of bursts with various hop count at high

load levels are almost the same as the case of no deflection, i.e., this deflection routing does not aggravate the unfairness problem for bursts with large hop count.

In Chapter 3, we proposed two schemes to alleviate the unfairness problem in optical burst switched networks: balanced JIT (BJIT) and prioritized random early discard (PRED). The BJIT scheme uses a simple equation to adjust the size of the search space in order to give bursts with large hop count better chance to find a free wavelength. The PRED scheme uses a probabilistic random early discard strategy to proactively drop some bursts in their first hop before they go to the core of the optical network; larger proactive dropping probabilities are applied to bursts with smaller hop count. The analysis and the test results reported in Chapter 3 show that the BJIT and PRED schemes alleviate, to a limited extent, the unfairness problem in optical burst switched networks without degrading the throughput performance. Both the BJIT and PRED schemes do not use preemption, i.e., the control packet of a data burst makes channel reservation without preempting the channel resources reserved by any other data burst. In this chapter, we present and evaluate a new scheme to improve the fairness performance in OBS networks based on constrained preemptions. The scheme achieves better and more balanced fairness among bursts with different hop counts without degrading the throughput of the network.

5.3 Preemptive scheme in optical burst switched networks

In [49], a preemptive scheme is used to improve the total blocking performance of optical burst switched networks. The authors observed that in OBS networks using JET scheduling, the blocking probability is higher for longer routes. They proposed a set of preemptive schemes

whose main idea is to assign different priorities for different types of bursts. The preemptive priority metric for a burst is the product $H*L$, where H is the current hop count (how far the burst has traveled from its source) and L is the route length (lightpath length from source to destination). The control packet with larger metric value can preempt the channels reserved by the control packet with smaller metric value. The metric $H*L$ gives higher preemptive priority to the burst with longer lightpath and to the burst that is close to arriving to its destination. In our simulation tests, we found that using the metric $H*L$ alone leads to a "*reversed unfairness*" problem, i.e., the blocking probabilities for the bursts with longer route length become much smaller than the blocking probabilities of bursts with shorter route length. The details of our tests will be discussed in section 5.5. Below, we introduce the OBS network model and the notation used in this chapter.

In OBS networks, data bursts are assembled at the network access station (NAS). Before transmitting a data burst, NAS sends out a control packet to do the resource reservation in each one of the optical cross connects (OXC's) along the path of this burst. After some offset time, the data burst is sent out from the NAS and is routed all-optically toward the destination using the resources reserved by the control packet. Failure of the control packet to reserve a channel resource in some intermediate OXC leads to dropping the data burst when it arrives to this OXC. The more the number of hops along the lightpath of a data burst, the larger the probability that the data burst will be dropped. This "beat down" unfairness problem can be modeled at high level as follows.

The blocking probability of a burst in one OXC can be expressed as:

$$PBLK = \beta_1 * \beta_2 * \beta_3 * ... * \beta_n \qquad (5.1)$$

where β_i ($1 \leq i \leq n$) is the probability that the ith wavelength is not free and n is the number of active channels in this OXC.

The "go through" probability of the burst in this OXC is therefore given by

$$PGO = 1 - PBLK \qquad (5.2)$$

If the lightpath of the burst has *m* hops, the probability that the burst will successfully reach its destination is given by

$$PREACH = PGO_1 * PGO_2 * ... * PGO_m \qquad (5.3)$$

Equation (5.3) clearly explains the beat down unfairness problem: as *m* increases, the go-through probability *PREACH* decreases.

Wasted Reservations & Preemptions:

For an effective preemptive scheme in OBS networks, the preemption process in an OXC should only occur during the offset time of the burst, i.e., after the control packet has successfully reserved the free channel and before the data burst has started using the reserved channel. In this

chapter, we assume that no OXC is allowed to preempt an active data burst whose transfer through this OXC is in progress.

Consider the following scenario in an optical cross connect, say OXC_i. At time *t1*, control packet *C1* of data burst *B1* arrives and reserves a channel. Data burst *B1* is expected to arrive and use the reserved channel at time *t1+offset*. At time *t2*, *t1< t2< t1+offset*, control packet *C2* of data burst *B2* arrives and preempts the channel reserved by *C1* because no other free channel exists at that time. Thus the channel reserved by *C1* in OXC_i has been wasted from time *t1* to time *t2*. Furthermore, channel resources are also wasted upstream and downstream of OXC_i. After control packet *C1* has reserved the free channel successfully at time *t1*, it travels downstream to reserve channels in subsequent optical cross connects: OXC_{i+1}, OXC_{i+2}, etc. Furthermore, as *C1* gets closer to its destination, its preemptive priority *H*L* [9] increases, i.e., *C1* gains increasing privilege to preempt the resources reserved by other control packets. Reservations and preemptions done by *C1* in its downstream path from OXC_i to the destination are wasted operations; in particular each preemption by *C1* further exacerbates the "wasted resources" problem and may cause cascaded propagation of other useless preemptions. This issue has not been considered in [9] and has motivated us to design additional constraints that reduce the harmful effect of wasted preemptions.

5.4 Constrained preemption fairness scheme

Our scheme uses the preemptive metric *H*L* as was done in [49], but we introduce three new constraints to alleviate the problem of wasted resources, solve the problem of reversed unfairness and improve the throughput of the network.

Constraint 1: "Only the channels reserved for bursts with lightpath length equal to or less than Υ can be preempted", where Υ is a controllable threshold parameter. The purpose of this constraint is to reduce the number of wasted preemptions induced by the control packet of a preempted burst. With Constraint 1, the largest preemptive metric value of the control packet of a preempted burst is Υ^2, which is attained in the last hop when H=L= Υ. In section 5.5, we will present the simulation results to evaluate the improvement obtained by Constraint 1.

Constraint 2: "Preemption is not applied in the first hop along the lightpath of the control packet". This means that the control packet cannot resort to preempt at its first hop to reserve the needed wavelength channel. The purpose of this constraint is to reduce the overall preemption rate, and in particular reduce early (unwarranted) preemptions. Notice that the control packet that generates preemption in its first hop (i.e., H=1 and H*L=1) is likely to have a long lightpath L and is therefore likely to cause more preemption processes before it arrives to its distant destination. Furthermore, the first hop is intrinsically suitable for the implementation of Constraint 2 because of its immediate proximity to the Network Access Station (NAS) that generated the connection request to transmit the data burst. If the control packet of a data burst fails to reserve a channel in the first hop, a negative feedback signal can be sent back to NAS

before the offset period expires, i.e., while the data burst is still stored in the electrical buffer. If NAS is not congested, it can continue to store the data burst in its electrical buffer and attempt the request at a later time. Our performance tests have shown that Constraint 2 is useful in solving the reversed unfairness problem and preventing the degradation of the throughput of the network.

Constraint 3: An additional probabilistic throttle is used to disable the application of preemption in each optical cross connect. The additional throttle has been found to further improve the throughput performance and prevent reversed unfairness. When a control packet cannot find a free channel in an OXC that is not the first hop, Constraint 3 is applied prior to invoking preemption. If Constraint 3 results in disabling preemption, the control packet is dropped and a negative feedback signal is sent to the source. Otherwise, preemption is applied subject to the preemptive metric $H*L$ and Constraint 1. We have evaluated two variations of the probabilistic throttles used by Constraint 3. The first variation, called Route-based Preemption (RP), uses the total length of the route from source to destination to determine the probability of disabling preemption for the control packet. The second variation, called Hop-based Preemption (HP), uses a probability based on the length of the partial path from the source to the current hop. In the RP variation, the throttle parameters to decide when the preemptive action can be executed are denoted $\alpha_R[i]$, where $1 \leq i \leq d$, d is the diameter of the network and each $\alpha_R[i]$ has a value between 0 and 1. If a control packet with route length j cannot find a free wavelength in an OXC along its path, preemption for this control packet is disabled with probability 1- $\alpha_R[j]$, i.e., $\alpha_R[j]$ is the probability to enable preemption. For best performance results, our extensive tests have shown that for small network diameters the values of $\alpha_R[j]$ should increase as j increases. For

networks with large diameters, the values of $\alpha_R[j]$ should increase as j increases until it reaches a maximum value (close to 1) then starts to decrease as j continues to increase toward d, the network diameter.

In the HP variation of Constraint 3, the corresponding parameters are denoted $\alpha_H[i]$, where $1 \le i \le d$ and d is the diameter of the network. Here also, the value of each $\alpha_H[i]$ is between 0 and 1. If a control packet cannot find a free wavelength in an OXC that is j hops away from the source, preemption for this control packet is disabled with probability 1- $\alpha_H[j]$, i.e., $\alpha_H[j]$ is the probability to enable preemption. Generally, the following relation is enforced:

$$\alpha_H[1] \le \alpha_H[2] \le \ldots \le \alpha_H[d] \qquad (5.4)$$

For networks with large diameter d, the value of $\alpha_H[d]$ is set to 1.

Notice that in the RP variation, the control packet uses the same throttle parameter value $\alpha_R[L]$ in all OXC's along its path, where L is the length of the route of this control packet. In the HP variation, however, the control packet uses successively increasing parameter values, $\alpha_H[2]$, $\alpha_H[3]$, ..., $\alpha_H[L]$ as it moves from its second hop to its destination. We now summarize our preemption scheme.

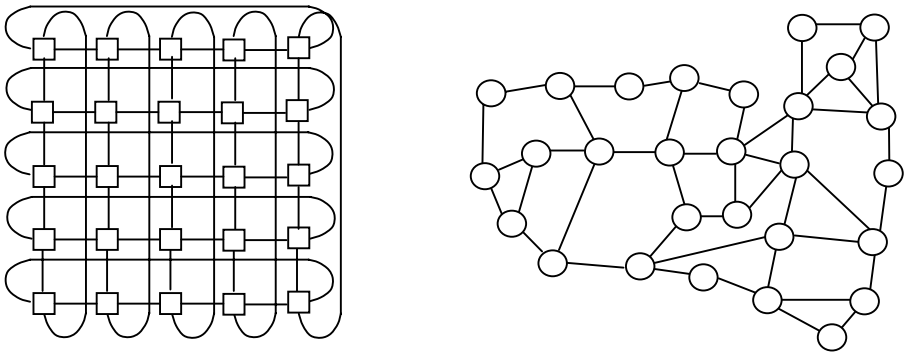
<u>First hop</u>: if no free channel is found in the first hop of the control packet, a negative feedback is sent to NAS and the control packet is discarded. The NAS may keep the data burst stored in its

electrical buffer and then send out a control packet at a later time. If the control packet finds a free channel in the first hop, this channel is reserved for a duration equaling to the offset time plus the data burst transfer time. After the successful reservation, the control packet is sent out from the first hop to the next hop along its lightpath to do channel reservation. If the reservation is not preempted or canceled, the data burst is sent out from the NAS after the offset time expires. The reserved channel is released immediately after the completion of the transfer of the data burst.

Intermediate hop: if there is a free channel, the channel is reserved and if the current OXC is not the final OXC, the control packet is sent out to the next OXC. If no free channel is found, Constraint 3 is applied to determine whether preemption is disabled. If Constraint 3 does not disable preemption, preemption is considered based on the metric $H*L$ and Constraint 1. Once preemption is judged to be feasible, the preemptive action is executed as follows. Among all occupied wavelengths, the wavelength reserved for the data burst with the smallest metric value ($H*L$) is chosen as the preempted wavelength (if there is a tie, a random wavelength is chosen). Once a preempted wavelength is selected, a RELEASE message is sent out to release the reserved channel along the upstream lightpath from the current hop to the first hop. This message travels upstream and is stopped when it reaches the first hop or when the reserved channel has already been released (due to a preemption or after the completion of the transfer of the data burst). Another RELEASE message is sent out to release the channel along the downstream path from the current hop to the destination. This message is stopped when it arrives to the final hop or when it reaches an OXC that has no channel reserved for the preempted burst.

5.5 Simulation results

Figure 5.1 shows the two network topologies that are used in our simulation. The first topology is the 5x5 Mesh-torus network with 25 nodes used in [49] and the second topology is the widely used US LongHaul network with 28 nodes.



5x5 Mesh-torus Topology LongHaul Topology

Figure 5.1. Two network topologies

In our simulation, a static lightpath between any two nodes is established using the "shortest path first" method as was done in [30, 48]. Notice that the longest shortest path has 7 hops in the LongHaul topology and has 4 hops in the 5x5 Mesh-torus topology. Similar to [49, 87], the traffic used in our tests is uniformly distributed among all nodes. This means that all nodes have equal likelihood to be the source of a data burst and for a given source node all other nodes in the network have equal likelihood to be the destination node. This uniform distribution on pairs of nodes gives the following distribution on the number of hops in the burst's lightpath: for the 5x5

mesh topology, the percentage of lightpaths with number of hops equal to 1, 2, 3 and 4 is 16.7%, 33.3%, 33.3%, and 16.7% respectively; for the LongHaul topology, the percentage of lightpaths with number of hops equal to 1, 2, 3, 4, 5, 6 and 7 is 12%, 20%, 23%, 21%, 14%, 8% and 2%, respectively. As in [30, 32, 49], we assume each node in the network has a full wavelength conversion capability. We evaluate the performance of our scheme and assess its benefits using the JIT signaling protocol. Our scheme, however, is equally applicable to the JET protocol since it does not modify the logic of the underlying signaling protocol or the value of the offset time.

In our simulation tests, assembled bursts arrive according to a Poisson distribution with controllable arrival rate. For each burst, the source node and destination node are randomly selected as explained before. Our simulation tests used values for OXC and network parameters similar to those typically used in the literature: the cut through time in each OXC is 2.5 milliseconds, the link delay per hop is 3 milliseconds, the burst length is 50 microseconds (equivalent to a burst of 250 Kbits at 5 Gbits/sec), the control packet processing time at each hop is 50 microseconds. The number of wavelengths used in each OXC is W=20, i.e., each incoming or outgoing fiber can carry 20 wavelengths. Each point in the performance graphs reported in this chapter was obtained by averaging the results of 6 simulation tests using different random generation seed values. Each simulation was run for sufficiently long time to obtain stable statistics; the total number of bursts processed in each simulation test ranged from 2 million bursts at low arrival rates to 20 million bursts at high loads. The unit time (ut) used in the graphs presented in this chapter is equal to 0.05 millisecond. Thus a load of 12 bursts/ut is equivalent to 60 Gbits/second. In our simulation scenario, the number of (electrical) buffers in each NAS is equal to half of the diameter of the network.

We denote the merit-based approach presented in [49] (which uses $H*L$ as the preemption metric) as MJIT; M stands for merit-based. If we modify the MJIT scheme by adding Constraint 2 (i.e., preemption is not allowed in the first hop), the resulting scheme is denoted by MJIT+C2. If we modify the MJIT scheme by adding Constraint1, Constraint 2 and the RP-based variation of Constraint 3, the resulting scheme is denoted RPJIT. If we modify the MJIT scheme by adding Constraint 1, Constraint 2 and the HP-based variation of Constraint 3, the resulting scheme is denoted HPJIT.

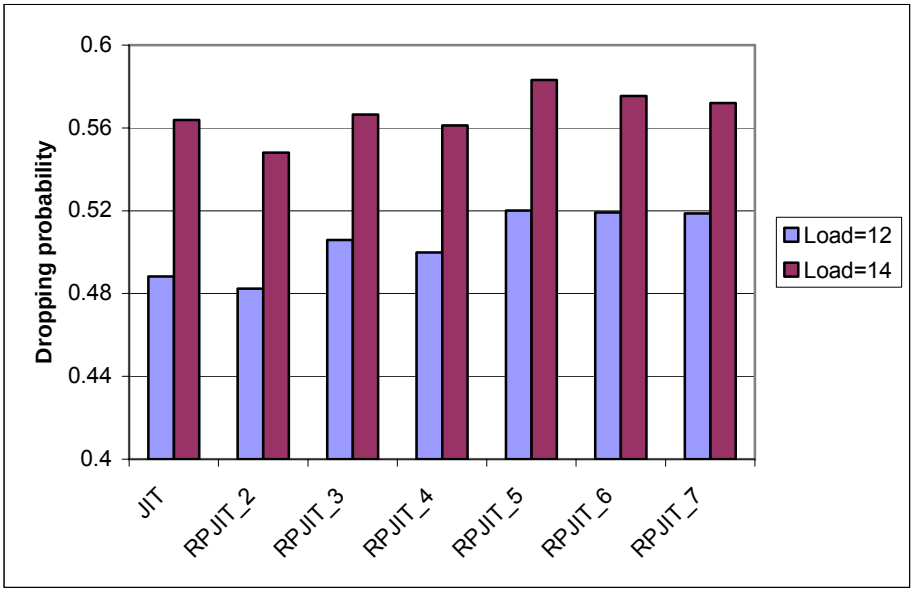


Figure 5.2. Dropping probability comparison with different threshold values ϒ for the LongHaul network

Figure 5.2 investigates the impact of the threshold parameter ϒ used in Constraint 1. The notation RPJIT_*i* means that the test used the RPJIT scheme with threshold value ϒ=*i* for

Constraint 1, i.e., only bursts with route length less than or equal to *i* can be preempted. In Figure 5.2, the values of the throttle used by Constraint 3 for bursts with route lengths 1, 2, 3, 4, 5, 6 and 7 are 0, 0.29, 0.7, 0.9, 0.78, 0.84 and 0.88, respectively. The figure shows that the value ϒ=2 (RPJIT_2) gives the best performance, i.e., the lowest overall dropping probability. Results for the 5x5 mesh-torus network (that has smaller diameter) are similar and are not given in this chapter. The results of the tests on both networks support the use of Constraint 1, i.e., imposing a threshold on the maximum route length of preempted bursts is a useful refinement to the preemption metric *H*L*. Unless otherwise stated, the throttle value ϒ=2 will be used in the various graphs presented in this chapter.

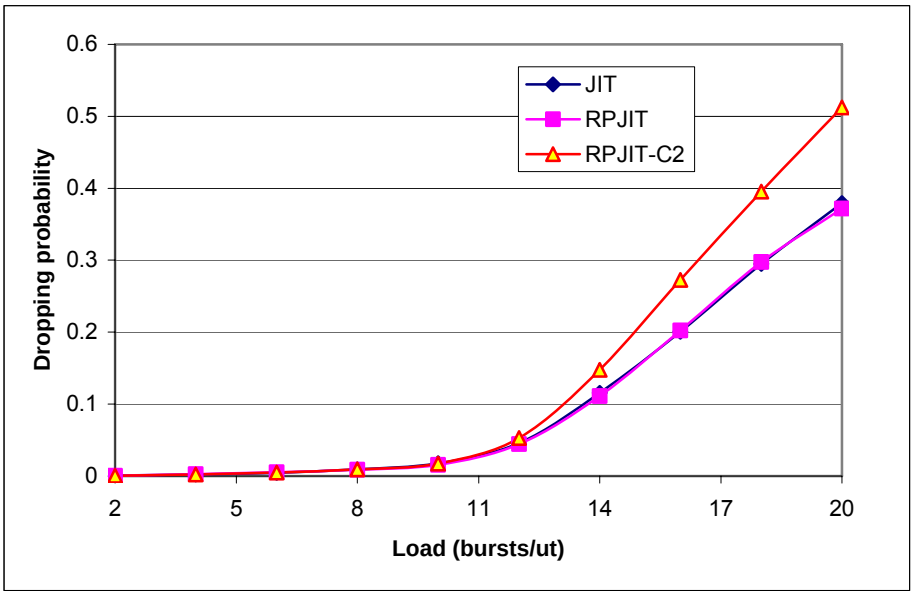


Figure 5.3. Dropping probability comparison among JIT, RPJIT and RPJIT-C2 for the Mesh-torus network

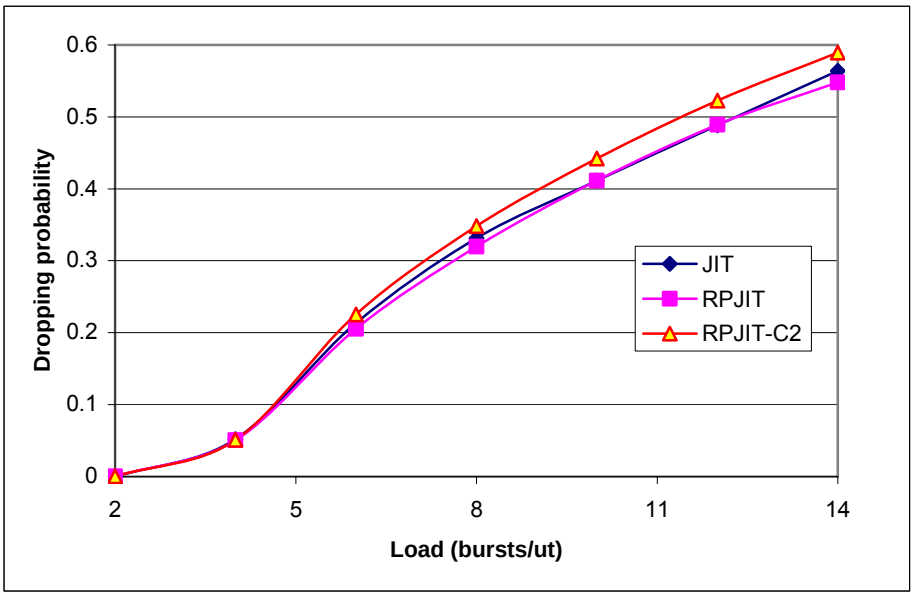


Figure 5.4. Dropping probability comparison for JIT, RPJIT and RPJIT-C2 for the LongHaul network

Figure 5.3 and 5.4 investigate the impact of Constraint 2, i.e., disabling preemption in the first hop of the control packet. The notation RPJIT-C2 means Constraint 2 is not used in the RPJIT scheme and therefore preemption is allowed in the first hop of the control packet. In Figure 5.3 for the mesh-torus network, the values of the throttle used by the RP version of Constraint 5.3 for bursts with route lengths 1, 2, 3 and 4 are 0, 0.4, 0.6, and 0.7, respectively. In Figure 5.4 for the LongHaul network, the values of the throttle used by the RP version of Constraint 3 for bursts with route lengths 1, 2, 3, 4, 5, 6 and 7 are 0, 0.29, 0.7, 0.9, 0.78, 0.84 and 0.88, respectively. Both figures clearly show that Constraint 2 is helpful in reducing the overall dropping probability (and hence improving the throughput) of the network.

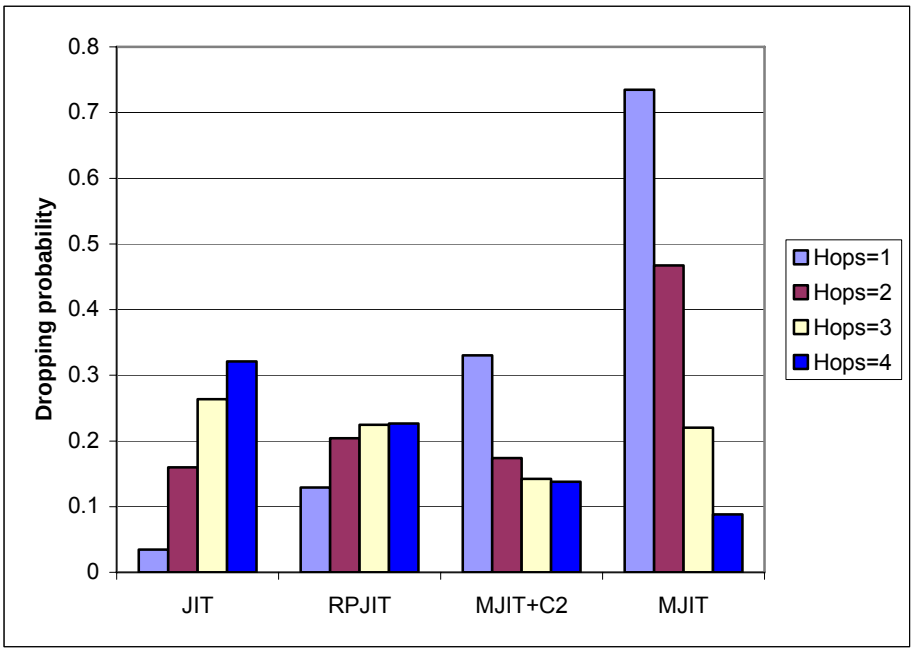


Figure 5.5. Dropping probability fairness comparison for the Mesh network at load =16

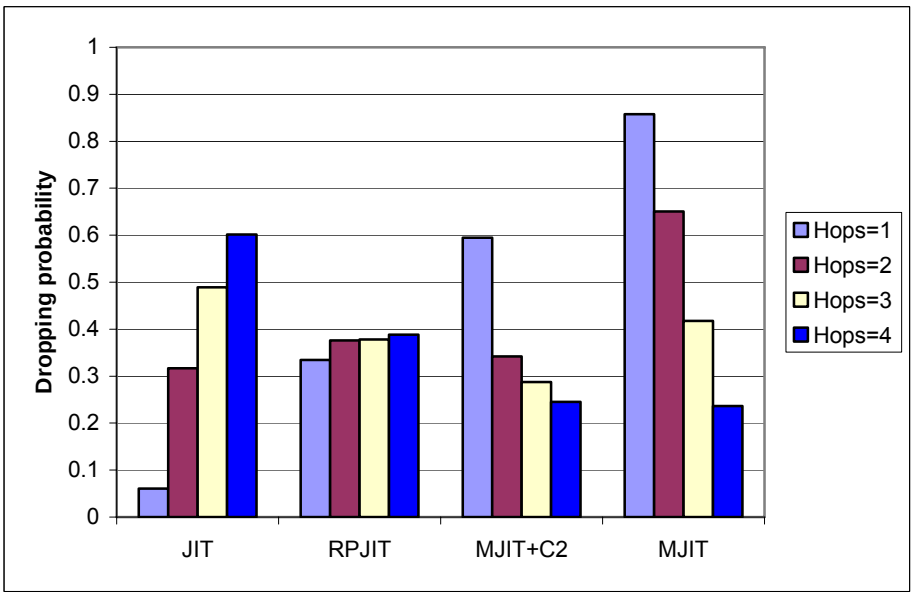


Figure 5.6. Dropping probability fairness comparison for the Mesh network at load =20

Figure 5.5 and Figure 5.6 present fairness comparisons for the Mesh-torus network at loads 16 and 20, respectively. The values of the throttle used by the RP version of Constraint 3 are the same as those used in Figure 5.3. Recall that the MJIT scheme is the merit-based approach [9] that uses *H*L* as the preemption metric, the MJIT+C2 scheme uses the *H*L* metric as well as Constraint 2, and the RPJIT scheme uses the *H*L* metric, Constraint 1, Constraint 2 and the Route-based version of Constraint 3. Figures 5.5 and 5.6 show that MJIT+C2 alleviates the reversed unfairness problem exhibited by MJIT. Thus Constraint 2 is helpful in improving the fairness of the network in addition to being helpful in improving the overall throughput performance (as shown earlier in Figures 5.3 and 5.4). Among the four schemes compared in Figures 5.5 and 5.6, the RPJIT scheme gives the best fairness among bursts with different route lengths.

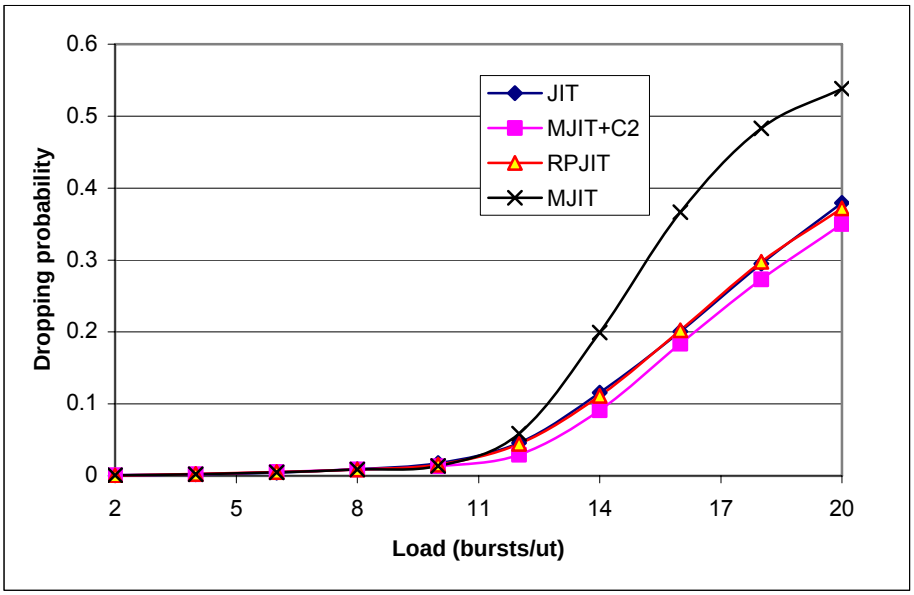


Figure 5.7. Dropping probability comparison among JIT, MJIT, MJIT+C2 and RPJIT in the Mesh-torus network

Figure 5.7 shows that the RPJIT scheme improves fairness without degrading the overall dropping probability (and hence throughput performance) of the network. Notice that the overall average dropping probability (overall all route lengths) of the RPJIT scheme is almost equal to that of the original JIT scheme.

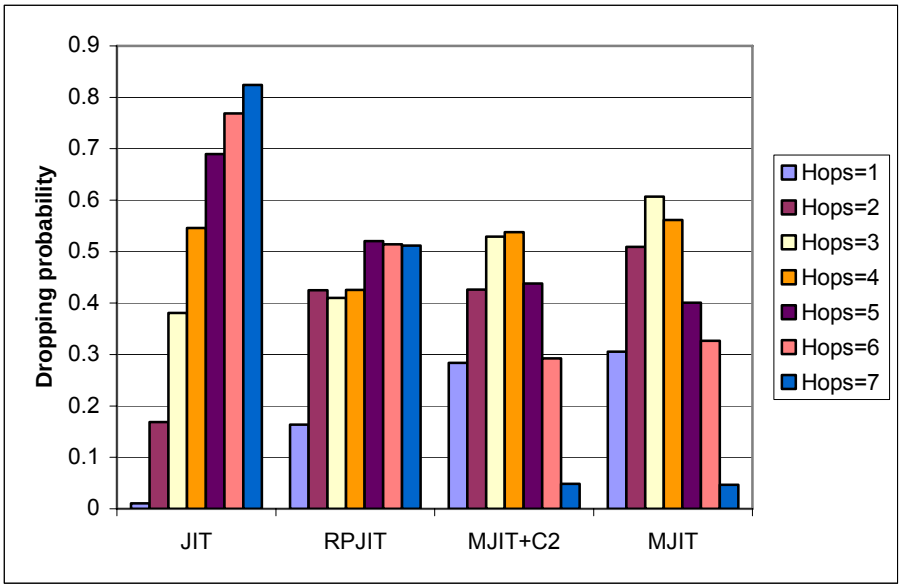


Figure 5.8. Dropping probability fairness comparison for the LongHaul network at load=10

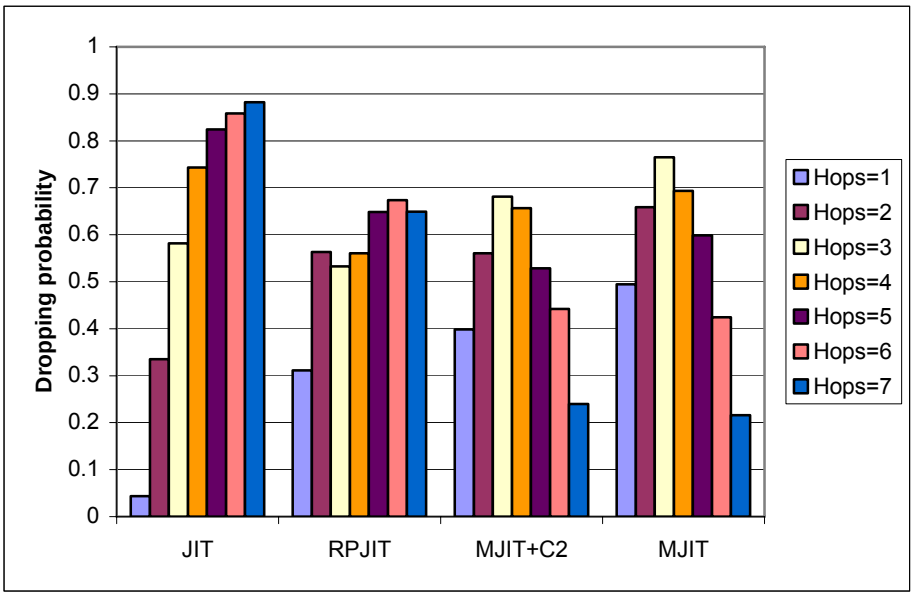


Figure 5.9. Dropping probability fairness comparison for the LongHaul network at load=14

Figure 5.8 and Figure 5.9 present fairness comparisons for the LongHaul network at loads 10 and 14, respectively. The values of the throttle used by the RP version of Constraint 3 are the same as those used in Figure 5.4. Again, both figures show that MJIT+C2 improves fairness compared to MJIT. The RPJIT scheme gives good fairness with less variation in dropping probability among the different route lengths.

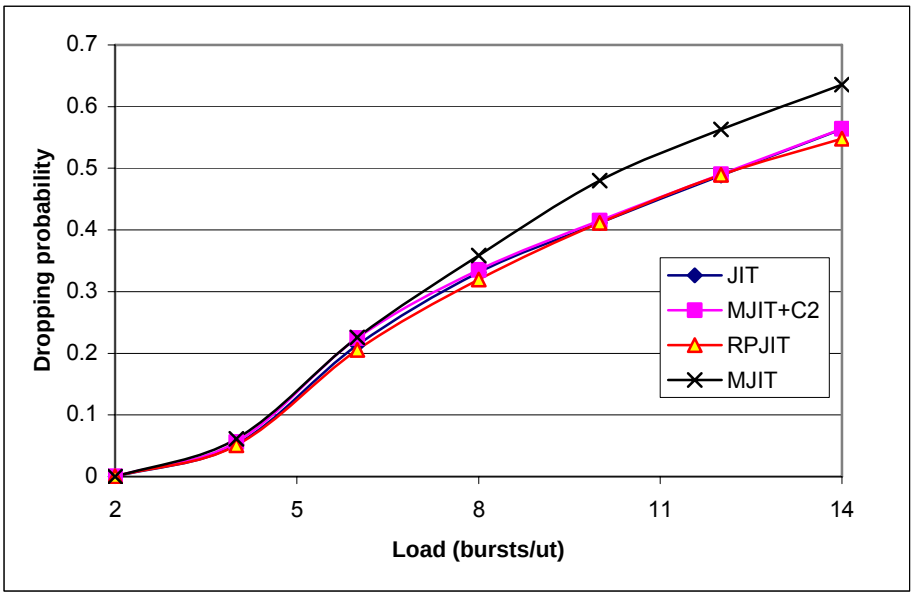


Figure 5.10. Dropping probability comparison among JIT, MJIT, MJIT+C2 and RPJIT in the LongHaul network

Figure 5.10 for LongHaul network confirms the trend explained earlier for the mesh-torus network in Figure 5.7, namely, the RPJIT scheme improves fairness without degrading the overall throughput performance of the network. The overall dropping probability of the RPJIT scheme is almost equal to that of the original JIT scheme.

Figure 5.11 compares the fairness of the Route-based version and the Hop-based version of Constraint 3 for the mesh-torus network at load 16. The throttle parameter values used in the HPJIT scheme for hop counts 1, 2, 3 and 4 are 0, 0.6, 0.75 and 0.9, respectively. The throttle parameter values used in the RPJIT scheme have the same values as those used in Figure 5.3. Figure 5.12 compares the overall dropping probability of the Route-based version and the Hop-based version of Constraint 3 for the Mesh-torus network at different loads. Both Figures 5.11

and 5.12 show that the Hop-based version HPJIT is slightly better than the Route-based version RPJIT both in fairness and throughput.

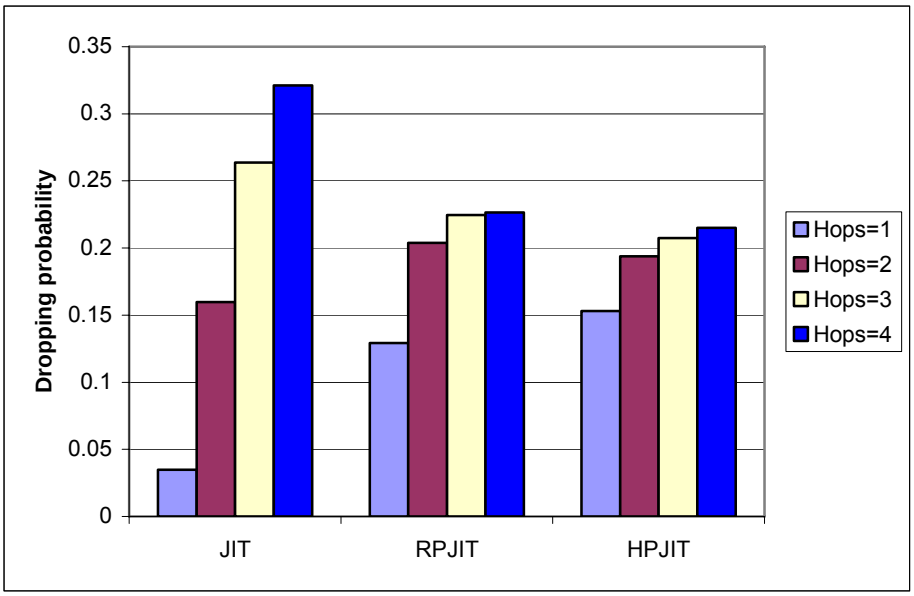


Figure 5.11. Route-based and Hop-based fairness comparison for the Mesh-torus network at load=16

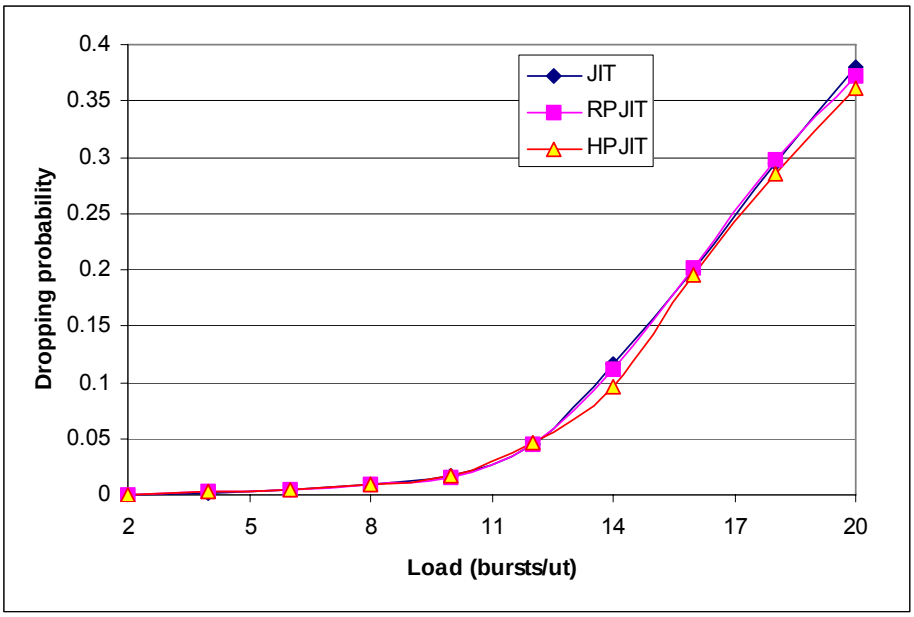


Figure 5.12. Total dropping probability comparison among JIT, RPJIT and HPJIT for the Mesh-torus network

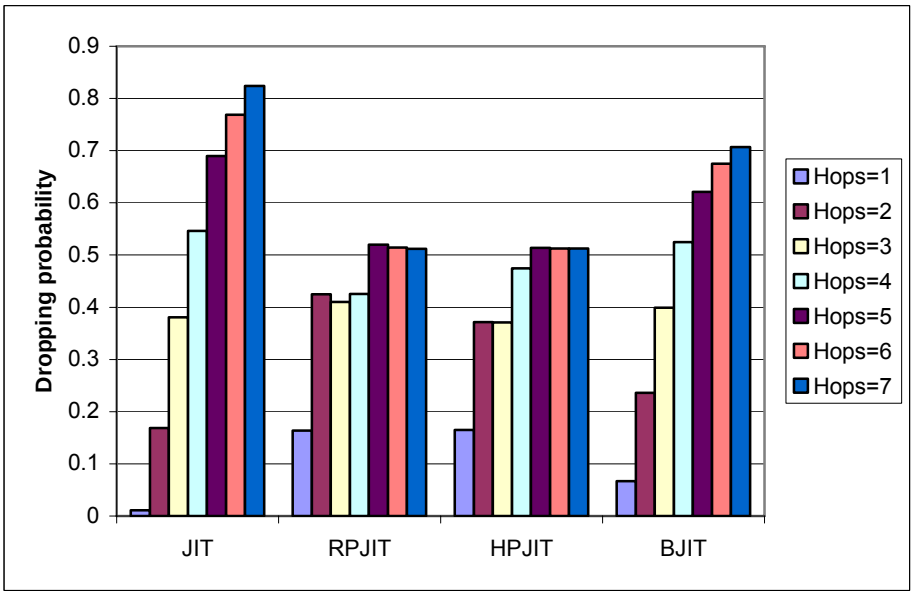


Figure 5.13. Route-based and Hop-based fairness comparison for the LongHaul network at load= 10

Figure 5.13 compares the fairness of the Route-based version and the Hop-based version of Constraint 3 for the LongHaul network at load 10. The throttle parameter values used in HPJIT for hop counts 1, 2, 3, 4, 5, 6 and 7 are 0, 0.7, 0.75, 0.8, 0.9, 0.95 and 1.0, respectively. The throttle parameter values used in RPJIT have the same values as those used in Figure 5.4. Figure 5.14 compares the overall dropping probability of the Route-based version and the Hop-based version of Constraint 3 for the LongHaul network at different loads. Figures 5.13 and 5.14 show that the Hop-based version HPJIT is slightly better than the Route-based version RPJIT with respect to both fairness and throughput. Figures 5.13 and 5.14 also show the performance of the BJIT scheme proposed earlier in Chapter 3 (the BJIT scheme uses the optimal parameter value g=0.5 that improves fairness without degrading throughput). The BJIT scheme, which does not use preemption, has better fairness than JIT but worse fairness than the new scheme HPJIT/RPJIT at roughly the same overall dropping probability.

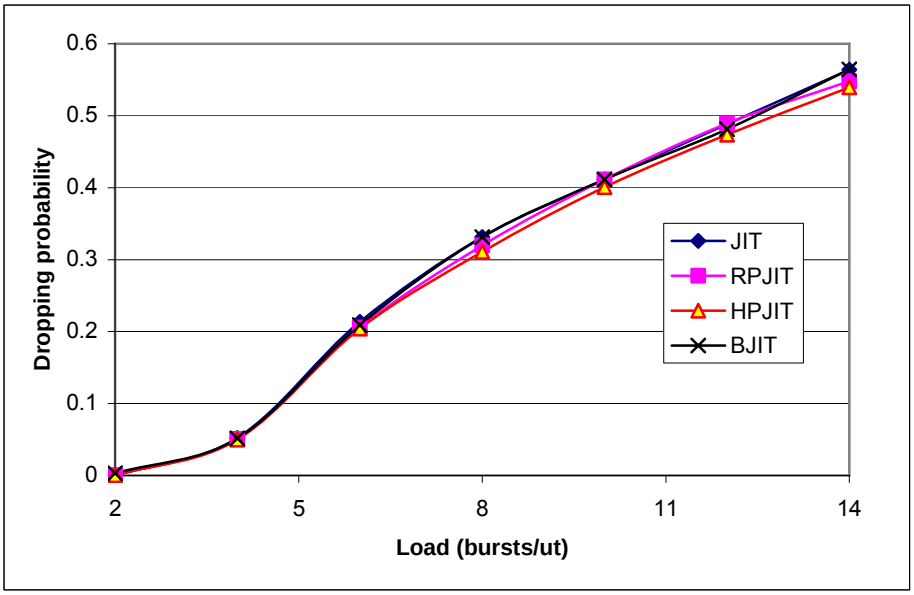


Figure 5.14. Total dropping probability comparison among JIT, RPJIT, HPJIT and BJIT for the LongHaul network

We now discuss how to choose the seven values of the hop-based throttles for Constraint 3 in the LongHaul network. The throttle $\alpha_H[1]$ must be set to 0 and $\alpha_H[7]$ can be set to 1 (or very close to 1). For the remaining five throttles, we found that equal spacing is an easy and good heuristic to use. Simply, we pick appropriate values for $\alpha_H[2]$ and for the spacing Δ, then we apply the following equation:

$$\alpha_H[i] = \alpha_H[2] + (i-2)*\Delta \qquad i>2 \qquad (5.5)$$

Figure 5.15 shows the fairness performance results when equations (5.5) is used with $\alpha_H[2]=0.7$ *and* Δ=0.06 at load 10. The scheme that uses this equation is denoted HPJITe. Figure 5.16 compares the overall dropping probability of HPJIT and HPJITe at different loads.

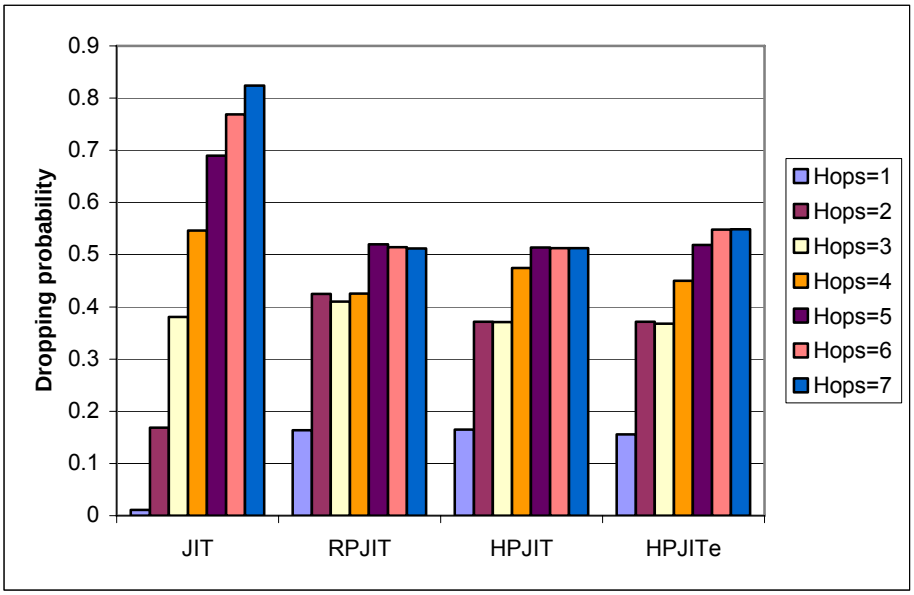


Figure 5.15. Dropping probability comparison, LongHaul network at load=10

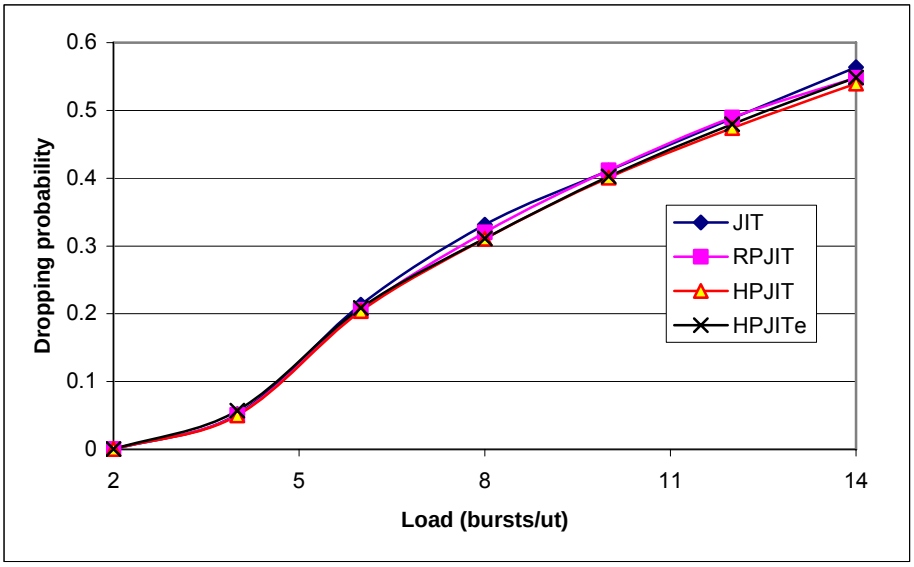


Figure 5.16. Total dropping probability comparison among JIT, RPJIT, HPJIT and HPJITe for the LongHaul network

CHAPTER 6. A PREEMPTION-BASED SCHEME FOR IMPROVING THROUGHPUT IN OBS NETWORKS

6.1 Introduction

In this chapter, we propose a preemption-based scheme to improve the throughput of OBS networks. In our proposal, the preemptive discipline is based on the burst size. The burst with larger size can preempt the resources reserved by a burst with a smaller size. The basic idea of the scheme proposed and analyzed in this section is that bursts with large sizes contribute more positively to the throughput of the network than bursts with small sizes.

The remainder of this chapter is organized as follows. In section 6.2, the preemption-based scheme for improving throughput is presented. In section 6.3, an analytical model for computing the throughput of a ring OBS network is developed. In section 6.4, extensive simulation results are presented to validate the analytical model of the ring topology as well as evaluate the improvement obtained by the preemption scheme in two other topologies: Mesh-torus and LongHaul.

6.2 Preemptive scheme for throughput improvement in OBS networks

In this chapter, we propose a preemptive scheme for OBS networks based on the burst size. We assume that the wavelength conversion is available in all OXCs [30, 32] and we do not consider electrical buffering at the source NAS. Information about the burst size is assumed to be included

in the control packet of the burst. When the control packet makes a channel reservation in an OXC, the burst size is recorded in the OXC along with other reservation information. Our preemptive scheme is outlined below.

The control packet is sent out from the source NAS on an out-of-band channel to reserve a wavelength in each OXC along the lightpath. When the control packet arrives at a node on the path, it tries to reserve a channel using the normal JIT protocol. If there are no free channels, preemption is considered as a last resort before dropping the burst. Let S1 be the size of burst B1 that is trying to reserve a wavelength in the current OXC and let S2 be the size of some burst B2 that already has a reservation in this OXC. If $S1 \geq S2 + \eta$ (where η is a controllable threshold) then the wavelength reserved by B2 is preempted and given to B1. If there are multiple bursts with existing reservations that satisfy the threshold criterion, the preempted burst is chosen to be the one with the smallest size (or selected randomly for the sake of analytical tractability as will be explained in section 6.3). If preemption cannot be executed, burst B1 is blocked.

6.3 Analytical model for computing throughput of ring networks

In this section, we analyze the performance of a ring topology and present a model to compute the throughput of the ring network when the preemption weight is based on the burst size. We assume that the ring network uses the JIT scheduling method. The preempted burst is chosen randomly from eligible bursts. After preemption, the same wavelength cannot be preempted again until the preempting burst is completely transmitted or its channel reservation is cancelled by a release message. The total burst arrival process to the ring network is assumed to be Poisson

with rate λ_{net} and the transmission time (size) of the burst is exponentially distributed with service rate μ (i.e., average burst size $1/\mu$). Let N be the total number of nodes in the ring, W be the number of wavelengths in each fiber link and λ be the local Poisson arrival rate of bursts per one node. The arrival rate is assumed to be symmetric for all nodes and therefore $\lambda_{net} = N*\lambda$. The traffic is assumed to be uniform, i.e., a burst arriving to a node has equal likelihood to be destined to any of the other $N–1$ nodes.

Each fiber link in the network can be modeled as M/M/W/W server. The blocking probability p_b of the link can be approximated by the Erlang-B formula,

$$p_b = \beta(W,\rho) = \frac{\frac{\rho^W}{W!}}{\sum_{m=0}^{W} \frac{\rho^m}{m!}} \qquad (6.1)$$

where ρ is the traffic intensity. An initial estimate of ρ is $0.5*\lambda/\mu$ where $0.5*\lambda$ represents the burst arrival rate for one link (i.e., half the local arrival rate of a node). The go-through probability p_g for this link is given by

$$p_g = 1 - p_b \qquad (6.2)$$

Each of the two links in a node gets two types of traffic. The first type is local traffic and has a rate of $0.5*\lambda$. The second type is traffic generated in downstream nodes and destined to upstream nodes. Let g_j be the probability that the lightpath of a burst has j hops. The lightpath length distribution is given by $g_1, g_2, \ldots, g_d$ for lightpath lengths 1, 2, …,d, respectively, where d is the

diameter of the ring. The arrival rate to the link from the first (closest) downstream node is $p_g * 0.5 * \lambda * (1 - g_1)$. Similarly, the arrival rate from the second downstream node is $p_g^{\ 2} * 0.5 * \lambda * (1 - g_1 - g_2)$, and so on. The actual arrival rate to a link in the current node is therefore given by

$$\lambda' = 0.5 * \lambda * \sum_{i=1}^{d} p_g^{\ i-1} * (1 - \sum_{j=1}^{i-1} g_j) \qquad (6.3)$$

The average go-through probability $\overline{P}_g$ for the entire ring network is given by

$$\overline{P}_g = \sum_{i=1}^{d} g_i * (1 - p_b)^i \qquad (6.4)$$

The controllable threshold of the preemption scheme is η. This means that a burst can be preempted only by a burst whose size is larger than the size of the preempted burst by a value that is equal to or exceeds the threshold η. For a burst with size t and the preemptive threshold is η, the average preempting burst size is

$$\int_{t+\eta}^{\infty} x d(1 - e^{-\mu(x-t-\eta)}) = \tfrac{1}{\mu} + t + \eta \qquad (6.5)$$

The probability that a burst with size x can preempt another burst with size t is

$$p_{\eta(x,t)} = \begin{cases} 1 - e^{-\mu(x-t-\eta)}, & \eta + t < x \\ 0, & 0 < x \le \eta + t \end{cases} . \qquad (6.6)$$

The burst transfer duration is defined to be the entire period in which the wavelength is reserved for or used by the burst (reservation time is the cut-through time and usage time is the burst transmission time). Preemption of a burst can only occur during the cut-through time. If

preemption occurs, a new cut-through time is started and the new (preempting) burst will not be preempted again until the wavelength is released. When the link has no free wavelengths during the cut-through time of a burst, newly arriving bursts will attempt to preempt this burst. The average size, $S_{pted}(M)$, of the preempted burst given that there are M arrivals of bursts during the cut-through time is

$$S_{pted}(M) = \frac{\int_0^\infty t(1-(1-e^{-\mu(t+\eta)})^M)d(1-e^{-\mu t})}{\int_0^\infty (1-(1-e^{-\mu(t+\eta)})^M)d(1-e^{-\mu t})} = \frac{\int_0^\infty t(1-(1-e^{-\mu(t+\eta)})^M)d(1-e^{-\mu t})}{1-\sum_{m=0}^{M}(-1)^m\binom{M}{m}\frac{e^{-\mu\eta m}}{m+1}} =$$

$$\frac{\int_0^\infty td(1-e^{-\mu t}) - \int_0^\infty t(1-e^{-\mu(t+\eta)})^M d(1-e^{-\mu t})}{1-\sum_{m=0}^{M}(-1)^m\binom{M}{m}\frac{e^{-\mu\eta m}}{m+1}} \tag{6.7}$$

$$= \frac{\frac{1}{\mu} - \int_0^\infty t(1-e^{-\mu(t+\eta)})^M d(1-e^{-\mu t})}{1-\sum_{m=0}^{M}(-1)^m\binom{M}{m}\frac{e^{-\mu\eta m}}{m+1}} = \frac{\frac{1}{\mu} - \sum_{k=0}^{M}(-1)^m\binom{M}{m}\frac{e^{-\mu\eta m}}{\mu(m+1)^2}}{1-\sum_{m=0}^{M}(-1)^m\binom{M}{m}\frac{e^{-\mu\eta m}}{m+1}}$$

Note that each of the M bursts attempts to preempt the current burst and once one of the M bursts succeeds, the remaining bursts cannot perform any additional preemption. The probability for M bursts arriving during the cut-through time is assumed to have a Poisson distribution.

$$p_M = \frac{e^{-\overline{M}} * \overline{M}^M}{M!} \tag{6.8}$$

where $\overline{M}$ is the average number of bursts arriving during the cut-through time of the burst. The average size, S_{pted}, of the preempted burst is

$$S_{pted} = \frac{\sum_{M=1}^{\infty} p_M * S_{pted}(M)}{\sum_{M=1}^{\infty} p_M} \qquad (6.9)$$

We can now get the mean size of the preempting burst, S_{pting}.

$$S_{pting} = \int_{\bar{t}+\eta}^{\infty} x\, d\,(1 - e^{-\mu(x - S_{pted} - \eta)}) = \frac{1}{\mu} + S_{pted} + \eta \qquad (6.10)$$

The value of $\overline{M}$ is given by

$$\overline{M} = \frac{0.5*\lambda}{W} * T_c * p_b \qquad (6.11)$$

where T_c is the cut-through time and p_b is the blocking probability of the node (note that all $\overline{M}$ bursts arrive during link congestion).

Now we compute the average size of the burst that does not get preempted given that there are M burst arrivals during the cut-through time of this (**no**n-**pr**eempted) burst.

$$\mathrm{S}_{\text{no-pr}}(\mathrm{M}) = \frac{\int_0^\infty t\mu e^{-\mu t}(1-e^{-\mu(t+\eta)})^M dt}{\int_0^\infty (1-e^{-\mu(t+\eta)})^M \mu e^{-\mu t} dt} = \frac{\int_0^\infty t\mu e^{-\mu t}\sum_{m=0}^{M}(-1)^m \binom{M}{m} e^{-\mu(t+\eta)m} dt}{\sum_{m=0}^{M}(-1)^m \binom{M}{m} e^{-\mu m\eta}\frac{1}{m+1}}$$

$$= \frac{\sum_{m=0}^{M}(-1)^m \binom{M}{m}\frac{e^{-\mu\eta m}}{\mu(m+1)^2}}{\sum_{m=0}^{M}(-1)^m \binom{M}{m} e^{-\mu m\eta}\frac{1}{m+1}} \qquad (6.12)$$

The unconditional average size of the burst that does not get preempted is given by

$$\mathrm{S}_{\text{no-pr}} = \frac{\sum_{M=1}^{\infty} p_M * \mathrm{S}_{\text{no-pr}}(\mathrm{M})}{\sum_{M=1}^{\infty} p_M} \qquad (6.13)$$

where p_M is given by equation (6.8).

The preemptive probability p_{pr} for a burst with size t in a node of the ring network given M bursts arriving during the cut-through time is given by

$$p_{pr}(t,M) = \sum_{m=0}^{M-1}(1-e^{-\mu(t+\eta)})^m e^{-\mu(t+\eta)} = 1-(1-e^{-\mu(t+\eta)})^M \qquad (6.14)$$

The corresponding probability unconditional on the burst size is

$$p_{pr}(M) = \int_0^\infty (1-(1-e^{-\mu(t+\eta)})^M)\, d(1-e^{-\mu t}) = 1-\sum_{m=0}^{M}(-1)^m \binom{M}{m}\frac{e^{-\mu\eta m}}{m+1} \qquad (6.15)$$

Finally the preemptive probability for all bursts in a node is

$$p_{pr} = \frac{\sum_{M=1}^{\infty} \mathrm{p}_{\mathrm{pr}}(\mathrm{M}) * p_M}{\sum_{M=1}^{\infty} p_M} \qquad (6.16)$$

where p_M is given by equation (6.8).

Assuming arrivals of bursts are evenly spaced during the cut-through time T_c, the earliest time for preemption is T_c /M, the second earliest time for preemption is 2*T_c/M and so on. The probability that the mth burst is the one that causes preemption is $(1-e^{-\mu(t+\eta)})^{m-1}e^{-\mu(t+\eta)}$ where t is the size of the preempted burst. Given M arrivals, the average wasted cut-through time due to preemption is

$$T_{cp}(M) = \frac{\sum_{m=1}^{M} \frac{m}{M} * T_c \int_0^\infty (1-e^{-\mu(t+\eta)})^{m-1} e^{-\mu(t+\eta)} d(1-e^{-\mu t})}{\sum_{m=1}^{M} \int_0^\infty (1-e^{-\mu(t+\eta)})^{m-1} e^{-\mu(t+\eta)} d(1-e^{-\mu t})} \qquad (6.17)$$

where, $\int_0^\infty (1-e^{-\mu(t+\eta)})^m e^{-\mu(t+\eta)} d(1-e^{-\mu t}) = \sum_{n=0}^{m} (-1)^n \binom{m}{n} \frac{e^{-\mu\eta(n+1)}}{n+2}$

The unconditional average wasted time is

$$T_{cp} = \frac{\sum_{M=1}^{\infty} T_{cp(M)} * p_M}{\sum_{M=1}^{\infty} p_M} \qquad (6.18)$$

After preemption, a new cut-through time is started and is guaranteed to be safe from further preemptions in this node. The average burst duration, denoted 1/ν, is therefore extended with wasted cut-through time and is given by

$$\begin{aligned} \tfrac{1}{\nu} &= ((T_c + T_{cp}) * p_{pr} + T_c * (1-p_{pr})) * p_b + T_c * (1-p_b) + \tfrac{1}{\mu} + t_p \\ &= T_c + T_{cp} * p_{pr} * p_b + \tfrac{1}{\mu} + t_p \end{aligned} \qquad (6.19)$$

Where, $((T_c + T_{cp}) * p_{pr} + T_c * (1 - p_{pr})) * p_b + T_c * (1 - p_b) = T_c + T_c * p_{pr} * p_b$ is the new extended cut-through time, $\frac{1}{\mu}$ is the average burst size, and t_p is the control packet processing time. Since the preemptive processes are randomly selected, by Jackson's theorem, the combined arrival rate may still be considered Poisson. Thus we can still use the Erlang-B formula, however, the load is changed. That means in equation (6.1), new blocking probability is changed to:

$$p_b = \beta(W, \rho_r) = \frac{\frac{\rho_r^W}{W!}}{\sum_{m=0}^{W} \frac{\rho_r^m}{m!}} \qquad (6.20)$$

where $\rho_r = \lambda'/v$, λ' is computed iteratively using equation (6.3) and (6.20). The corresponding go-through probability is computed as $p_g = 1 - p_b$. Since the go-through probability will be changed when the preemption is executed, we can iteratively change the go-through probability and recalculate the preemptive probability and go-through probability in a node, till the difference of two adjacent go-through probabilities is almost the same.

The total throughput of the network is decided by

$$Thr = \lambda_{gnet} * S$$

where S is the average size of bursts that successfully reach their destination and λ_{gnet} is the go-through arrival rate, i.e., the number of bursts that successfully go from their source to their destination per unit of time. This means that $\lambda_{gnet} = \lambda_{net} * \overline{P_g}$ where $\overline{P_g}$ is the average go-through probability of the entire network given in equation (6.4).

Now we consider the average size of bursts that reach their destination, S. There are three types of such bursts. The first type is bursts that do not experience blocking. The second type is bursts that go through congested nodes but they are not preempted. The third type is bursts that preempt other bursts during their journey from source to destination. The first type has an average size of 1/μ and represents a fraction $\overline{P_g}$ of the total bursts successfully reaching their destination. For the second type, the average size is $S_{no\text{-}pr}$. The blocking probability of a hop is p_b and the preemptive probability during congestion at that hop is p_{pr}. The preemptive probability for a hop during all times (congestion and non-congestion) is $p_b * p_{pr}$. If a successful burst has a lightpath of i hops, its preemption probability is $(p_b * p_{pr} + p_g)^i - p_g^{\,i}$.

Thus, the average preemptive probability for all bursts reaching their destination is:

$$\sum_{i=1}^{d} g_i * (\sum_{j=1}^{i} \binom{i}{j} (p_b * p_{pr})^j p_g^{\,i-j}) = \sum_{i=1}^{d} g_i * ((p_b * p_{pr} + p_g)^i - p_g^{\,i}) = \overline{P_{pr}}$$

Similarly, the average "no preemption" probability for all bursts reaching their destination is:

$$\sum_{i=1}^{d} g_i * (1 - (p_b * p_{pr} + p_g)^i) = 1 - \sum_{i=1}^{d} g_i * ((p_b * p_{pr} + p_g)^i - p_g^{\,i} + p_g^{\,i}) = 1 - \overline{P_{pr}} - \overline{P_g}$$

Hence, the final throughput of the network is:

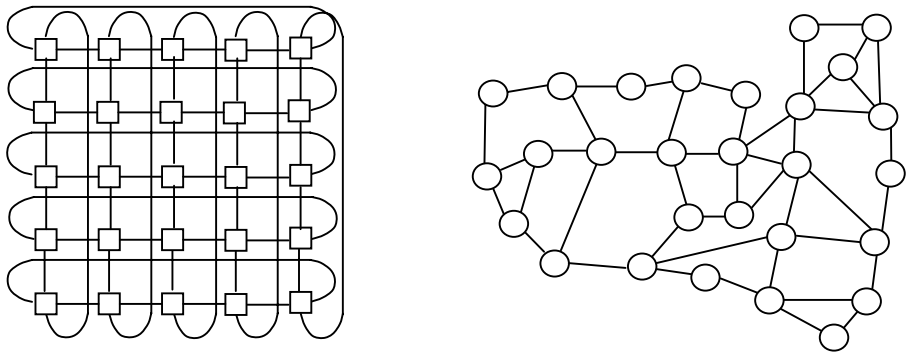
$$Thr = \lambda_{gnet} * (S_{no-pr} * (1 - \overline{P_{pr}} - \overline{P_g}) + S_{pting} * \overline{P_{pr}} + \tfrac{1}{\mu}(1 - \overline{P_b})) \qquad (6.21)$$

The above analytical model can be used to compute the throughput of the ring network iteratively. We first use equations (6.1) and (6.2) to compute the initial values of p_g and p_b. An

initial value of λ' is computed using equation (6.3). In successive iterations, we compute the different values of bursts sizes, the wasted cut-through time using equation (6.18) and the extended duration time 1/ν using equation (6.19). New values of p_b and p_g are computed using equation (6.20) and a new value of λ' is computed using equation (6.3). The iteration process stops when p_g converges to a stable value. The network throughput is finally computed using equation (6.21).

6.4 Simulation results

We present the simulation results and validate the analytical throughput model for a ring network with 7, 17 and 33 nodes (denoted as Ring7 , Ring 17 and Ring 33). In addition, we present simulation results for two additional networks shown in Figure 6.1, a 5x5 Mesh-torus network with 25 nodes and the LongHual network with 28 nodes.



5x5 Mesh-torus Topology

US LongHaul topology

Figure 6.1. Additional network topologies

For our simulation, the burst arrival process is Poisson and the cut-through time is 2.5 milliseconds. The link delay is 3 milliseconds. The burst size has an exponential distribution and its mean size is 10. For every data burst, the source and destination nodes are randomly selected from any nodes in the network, and the lightpath between source and destination nodes is the shortest path between these two nodes. Thus, for 5x5 Mesh-torus network, the network diameter is 4 and the distribution for lightpath length of 1, 2, 3 and 4 is 1/6, 1/3, 1/3 and 1/6, respectively. For Ring17 network, the network diameter is 8 and the lightpath distribution for each lightpath length is 1/8. For Ring33 network, the network diameter is 16 and the lightpath distribution for each lightpath length is 1/16.

The mean size of burst is 10 data units and its transmission time is 0.5 millisecond (time unit). We assume that a data burst has an average size of 5 megabytes, so the data transfer rate is 10 Gigabytes/second. The cut-through time is Tc = 2.5ms. The average control packet process time t_p is 5 μs (microseconds).

Figures 6.2 to 6. 6 show the throughput results for ring networks with nodes 7, 17, and 33 using different number of wavelengths at different loads. Figures 6.7 to Figure 6.11 give the go-through probability comparison corresponding to Figures 6.2 to 6.6.

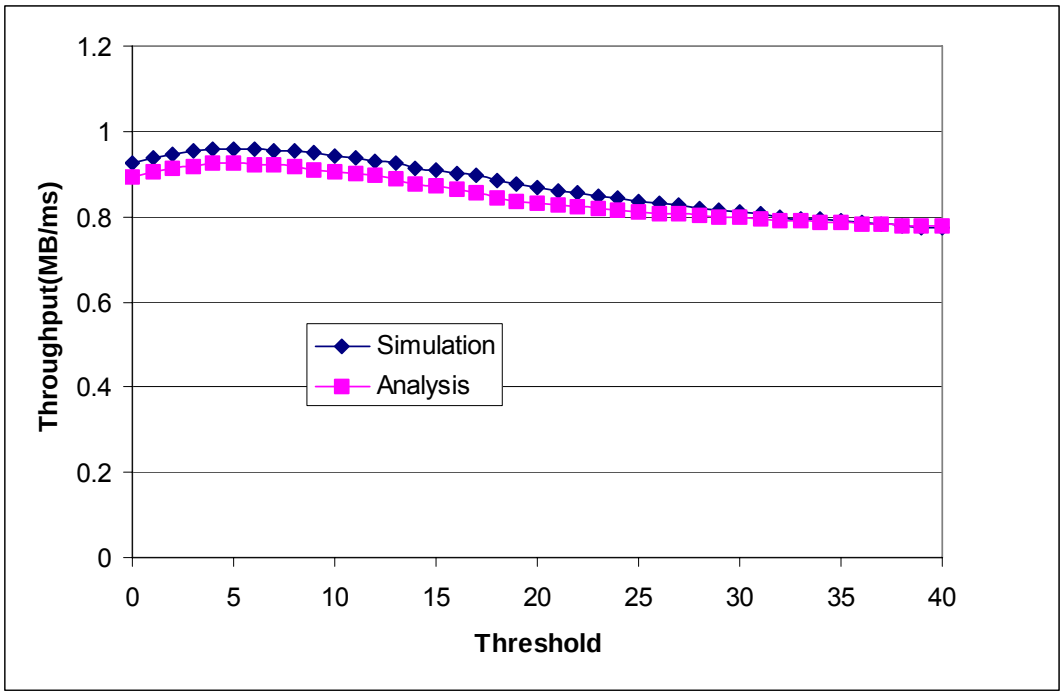


Figure 6.2. Throughput for Ring7, W=8, load λ_{net} =12 bursts/(100ms).

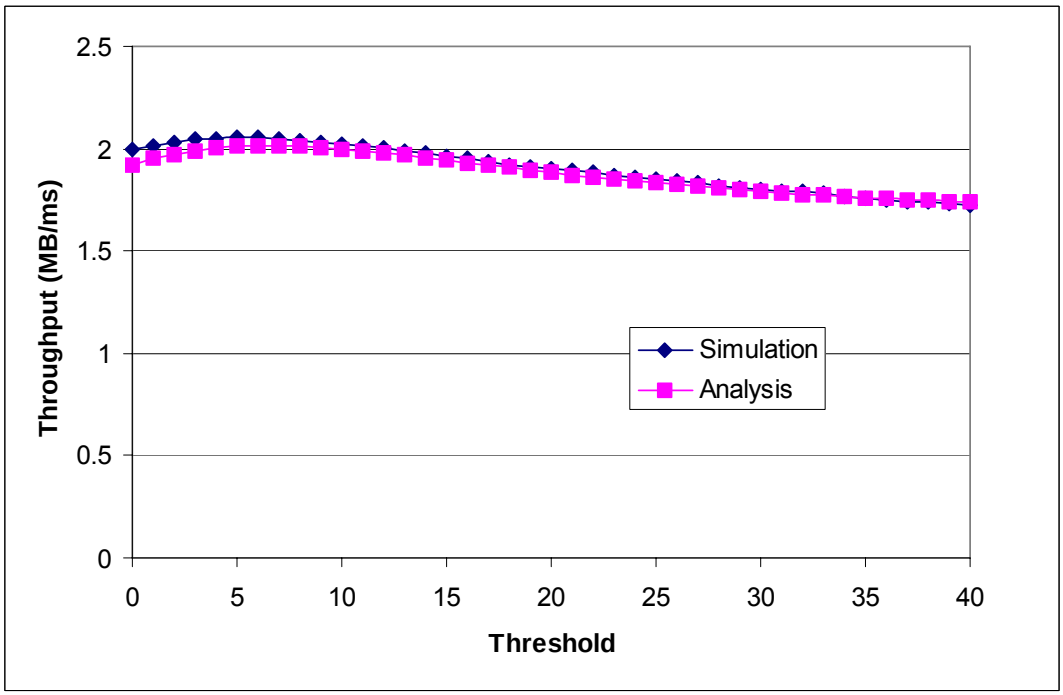


Figure 6.3 Throughput for Ring7 network with W=16, load λ_{net}=24 bursts/(100 ms).

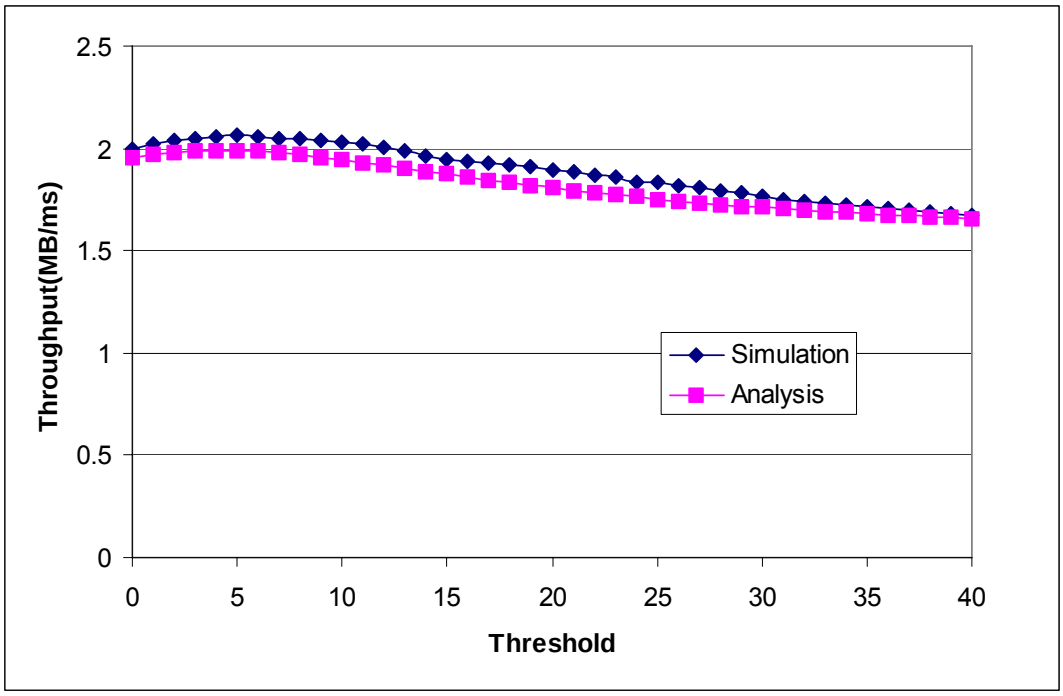


Figure 6.4. Throughput for Ring 17, W=16, load λ_{net} =24 bursts/(100 ms).

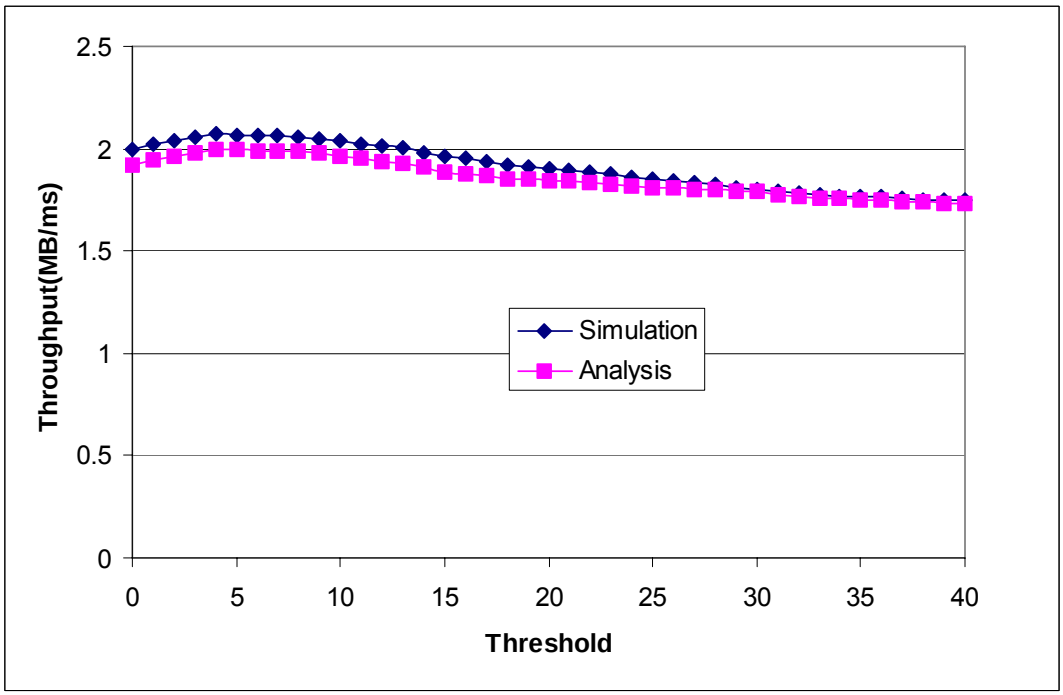


Figure 6.5. Throughput for Ring33, W=16 and load λ_{net} =24 bursts/(100ms).

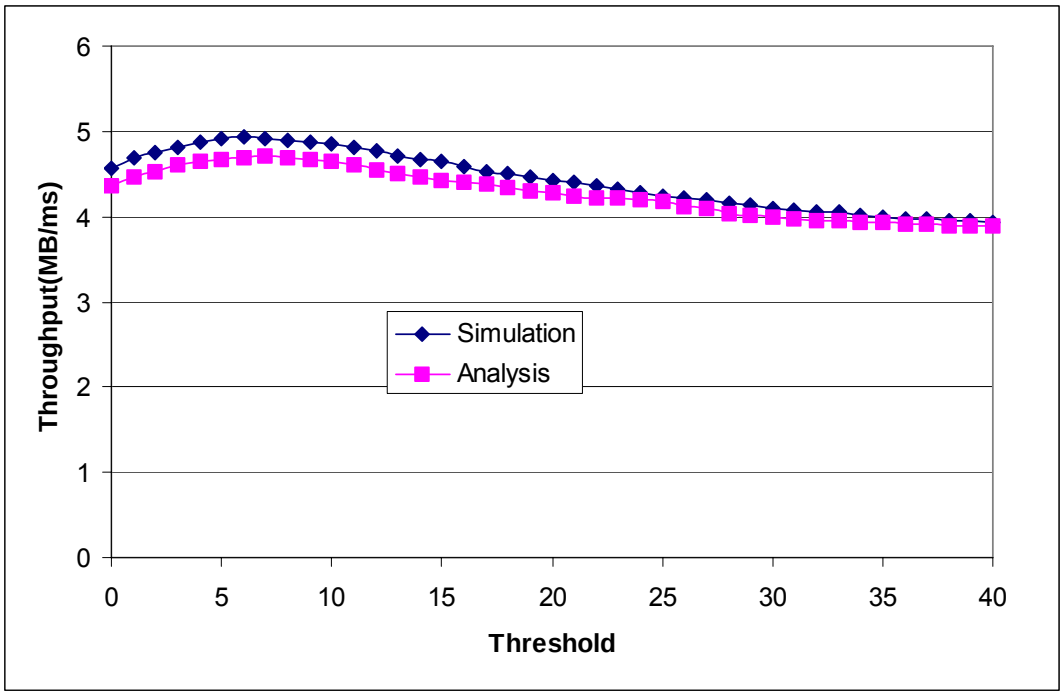


Figure 6.6. Throughput for Ring33, W=32 and load λ_{net} =60 bursts/(100 ms).

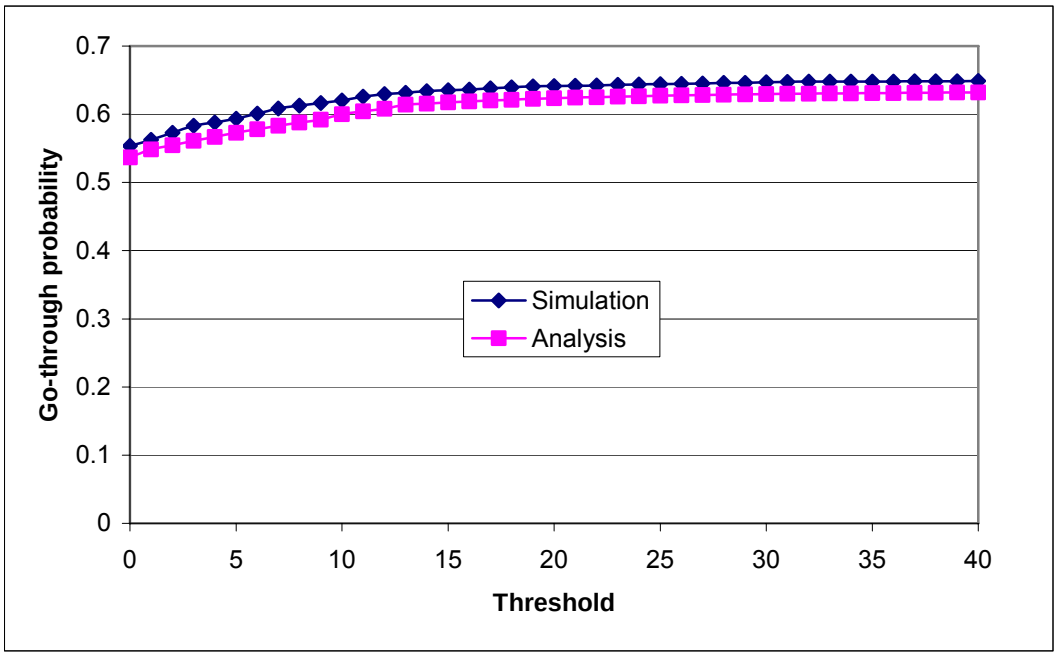


Figure 6.7. Go-through probability of Ring7, W=8 and load λ_{net} =12 bursts/(100 ms).

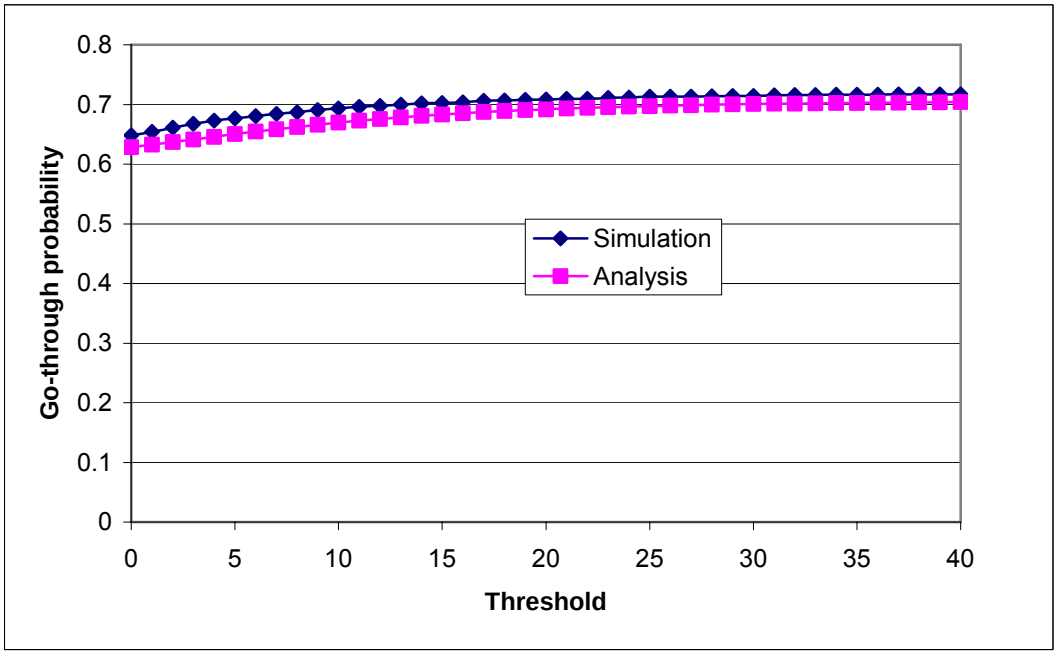


Figure 6.8. Go-through probability of Ring7, W=16 and load λ_{net} =24 bursts/(100 ms).

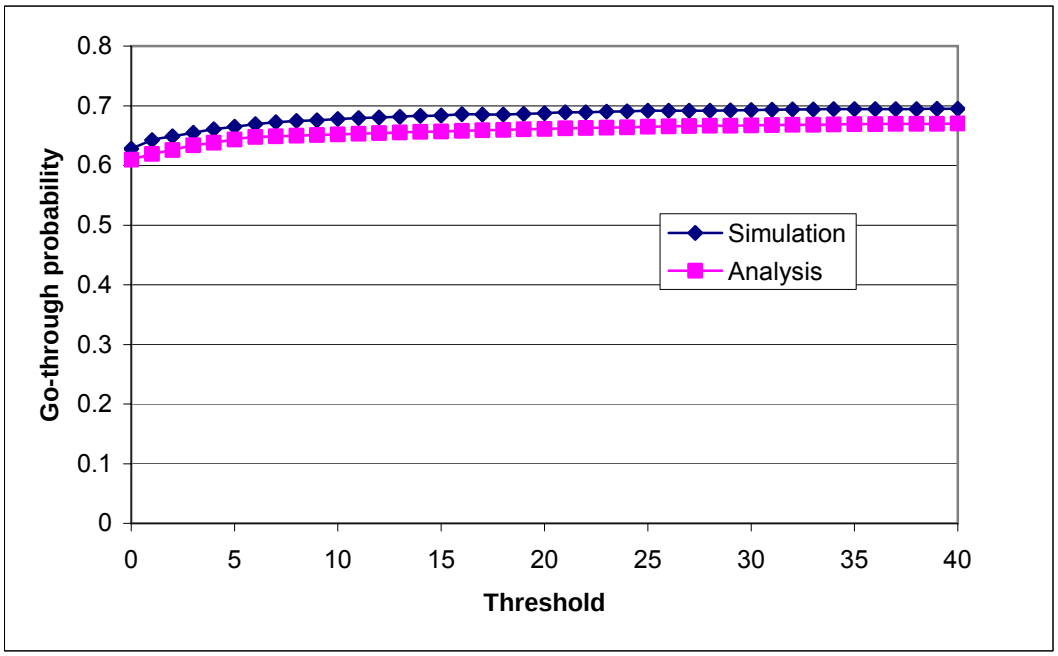


Figure 6.9. Go-through probability of Ring17, W=16 and load λ_{net} =24 bursts/(100ms).

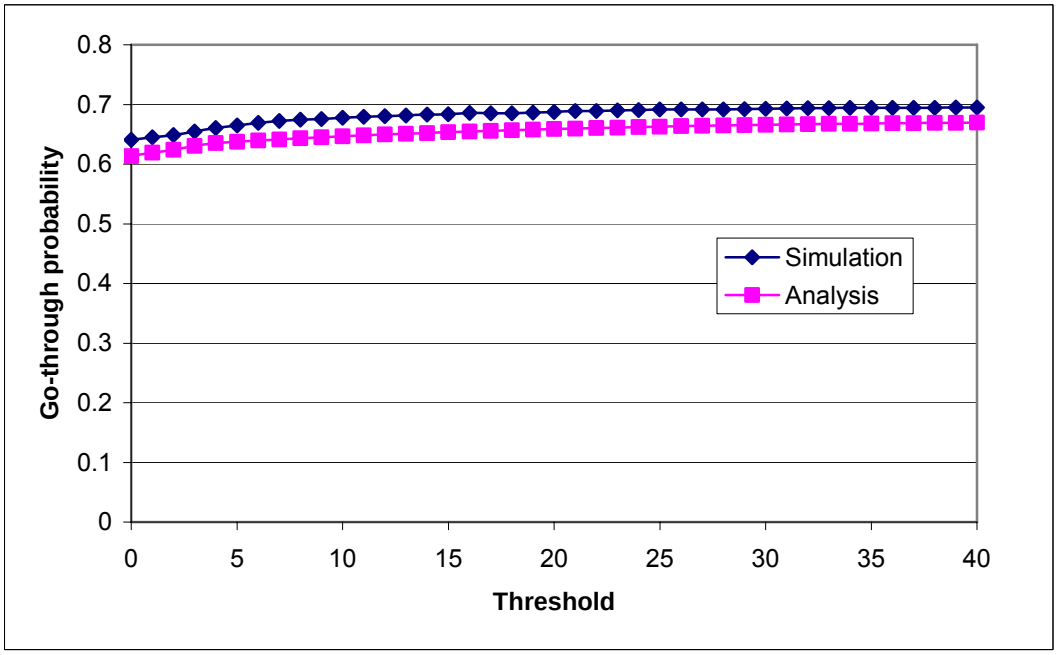


Figure 6.10. Go-through probability of Ring33, W=16 and load λ_{net} =24 bursts/(100ms).

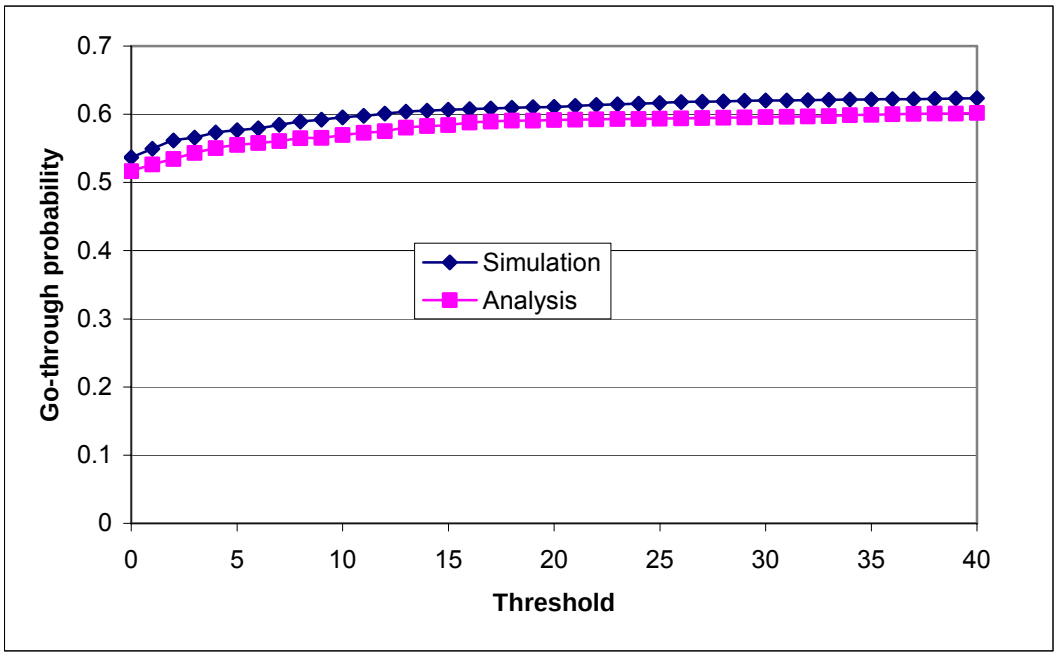


Figure 6.11. Go-through probability of Ring33, W=32 and load λ_{net} =60 bursts/(100ms).

The comparisons in Figures 6.2 through 6.11 all prove that the analytical model works accurately and can be used to analyze the throughput performance of ring networks using burst preemption based on burst size.

Now, we give some extended simulation results to show that the preemptive scheme can improve the throughput of the OBS networks in ring, Mesh-torus and LongHaul networks.

Figure 6.12 to Figure 6.14 give the simulation results comparison for Ring 7 with W=16 and load λ_{net} = 24 bursts/ (100 ms), Ring17 with W=16 and load λ_{net} = 40 bursts/ (100 ms) and Ring 33 with W=16 and load λ_{net} = 60 bursts/ (100 ms). In these figures, the notation *SP random* means only single preemption per wavelength is allowed and the preempted burst is randomly

selected (this is the scheme analyzed in section 6.3). The notation *MP random* means multiple preemptions are allowed and the preempted bursts are randomly selected. The notation *SP smallest* means only single preemption is allowed and the preempted burst is the one that has the smallest size.

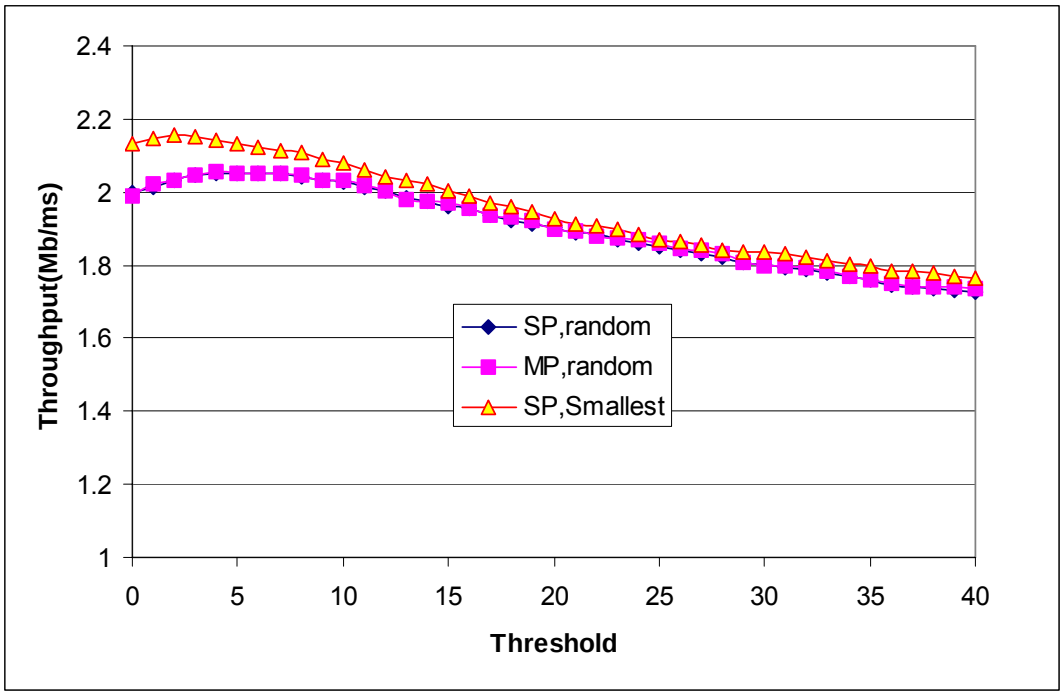


Figure 6.12. Throughput comparison for different preemptive schemes. Ring 7, W=16, load λ_{net} = 24 bursts/ (100 ms)

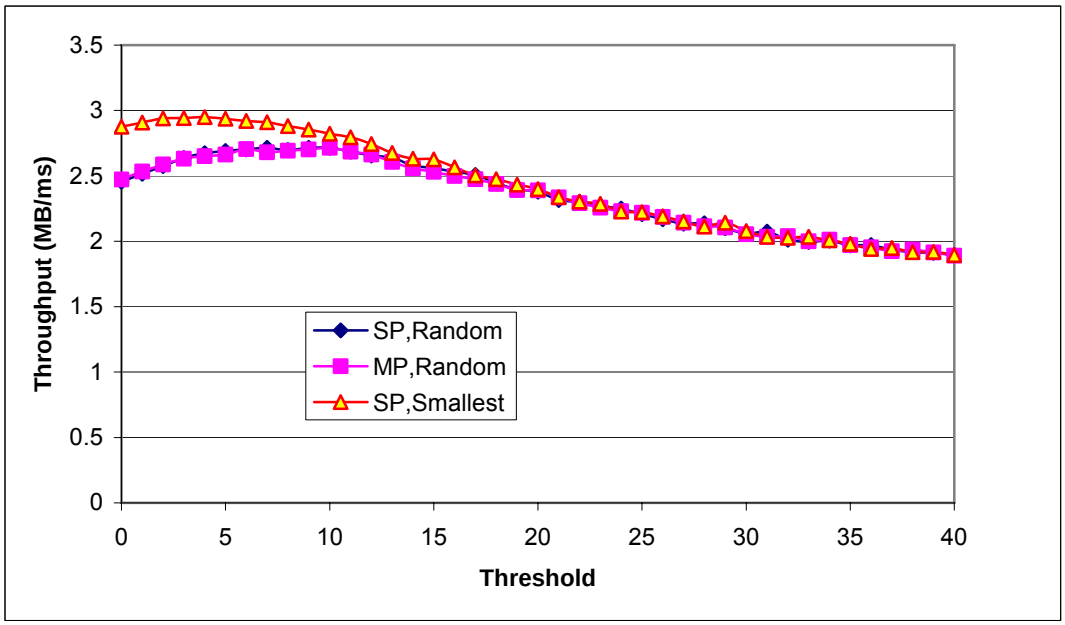


Figure 6.13. Throughput comparison for different preemptive schemes. Ring 17, W=16, load λ_{net} = 40 bursts/ (100 ms)

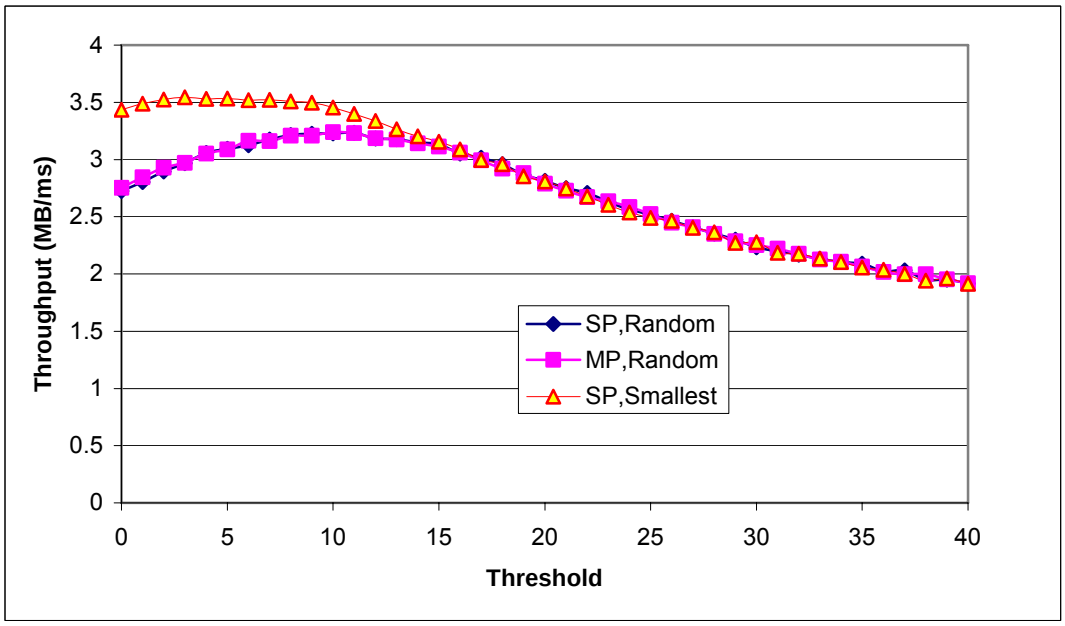


Figure 14. Throughput comparison for different preemptive schemes. Ring 33, W=16, load λ_{net} = 60 bursts/ (100 ms)

Figure 6.12 to Figure 6.14 show that the multiple preemptions and the single preemption schemes have almost the same performance. However, if the burst with smallest size is preempted rather than a random burst, the performance is improved.

Figure 6.15 and Figure 6.16 give the throughput comparison between JIT and the preemptive scheme (i.e., single preemption with random selection) at different thresholds for Ring17 and Mesh-torus networks, respectively. Figures 6.15 and 6.16 show that, compared with the conventional JIT scheme, our preemptive scheme can greatly improve the network throughput performance, especially when the load is large, and the improvement is significant.

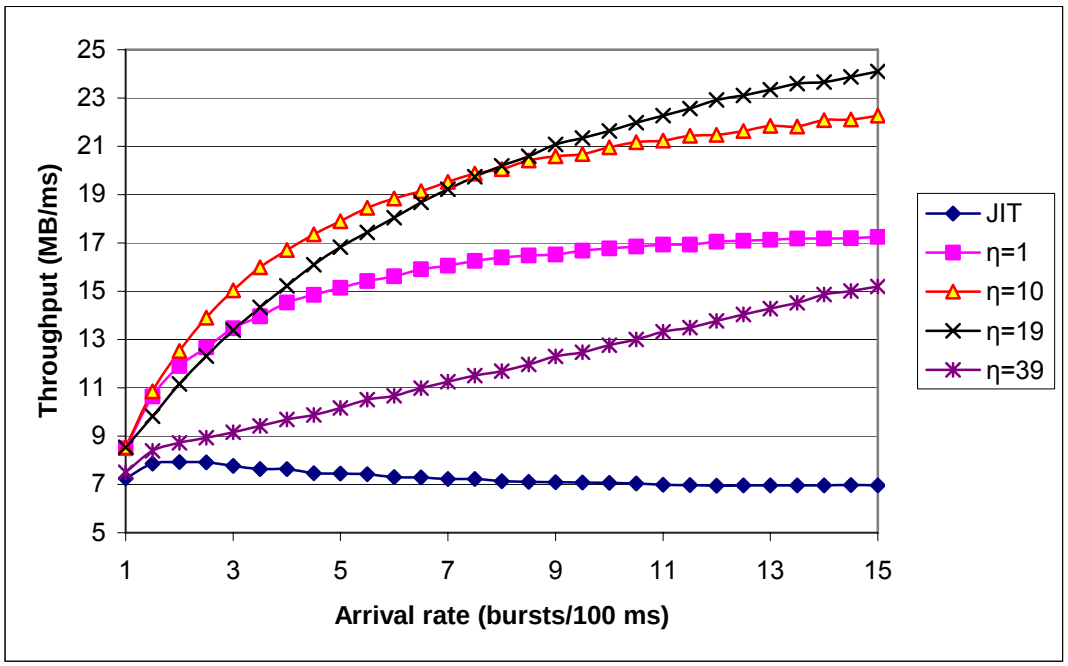


Figure 6.15. Throughput comparison for different Load, Ring17 network, W=8

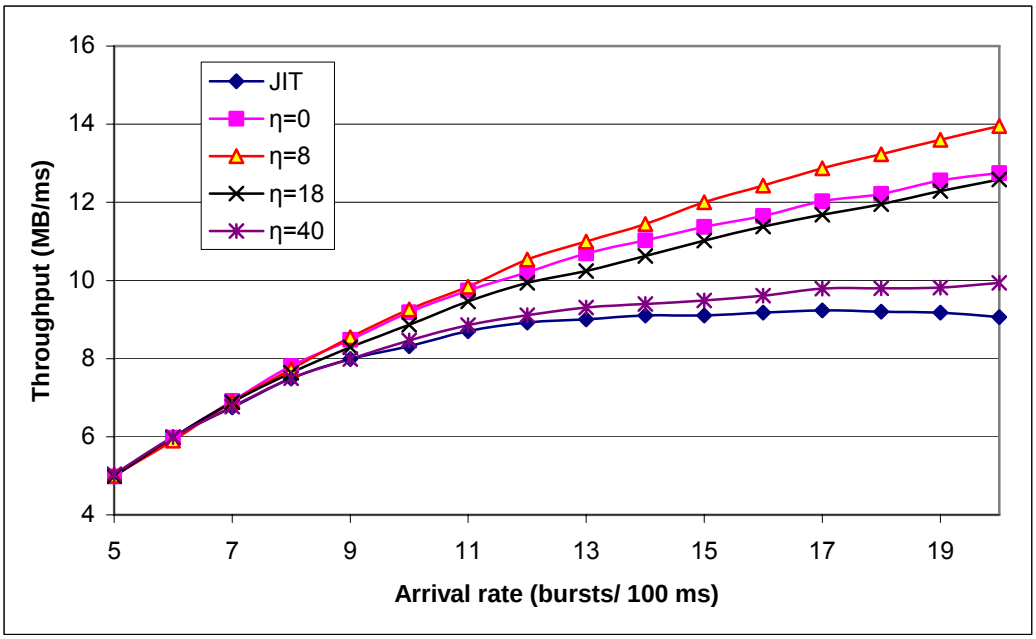


Figure 6.16. Throughput comparison for different threshold, Mesh-torus network, W=16

Figure 6.17 and Figure 6.18 show the throughput of Ring17 with W=8 and 5x5 Mesh-torus network with W=16, respectively. The figures show that the improvement of the network throughput is dependent on the threshold used in the preemption scheme.

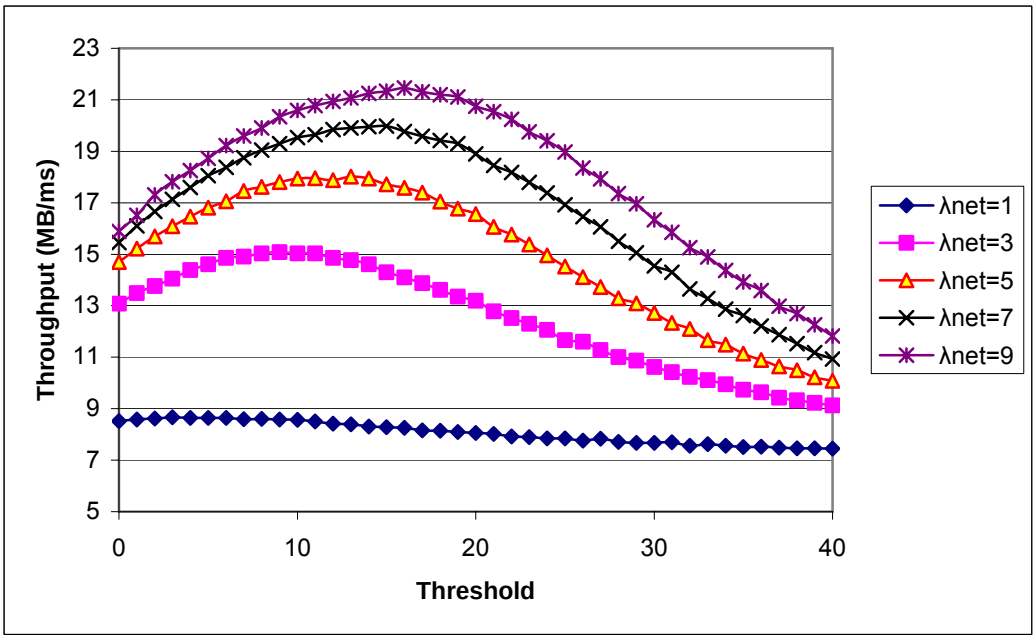


Figure 6.17. Throughput comparison for different thresholds, Ring17 network, W=8

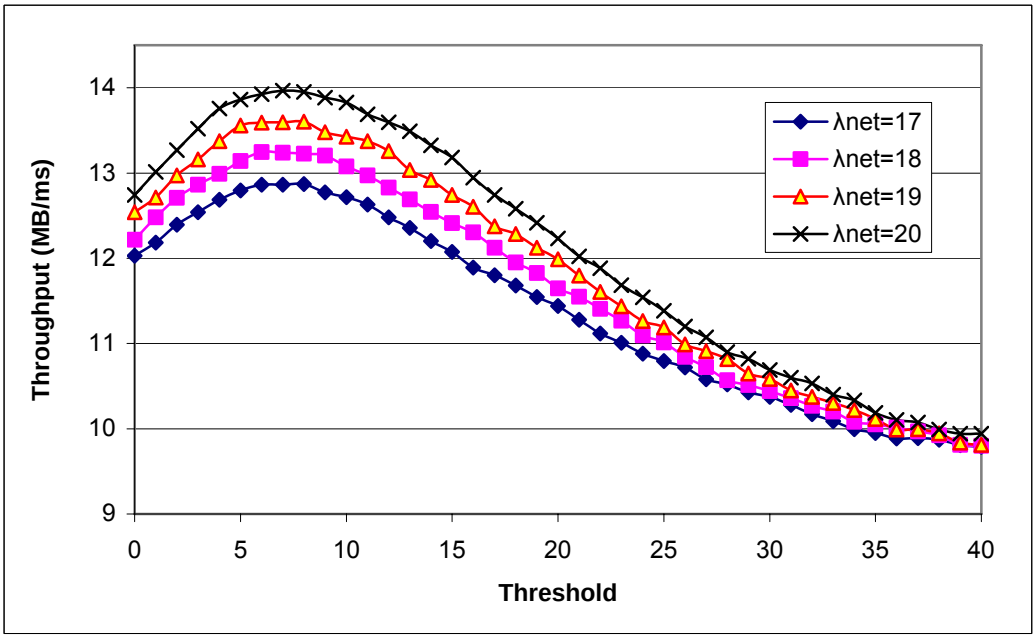


Figure 6.18. Throughput comparison for different thresholds, Mesh-torus network, W=16

Figure 6.19 gives the comparison for the multiple preemptions and single preemption schemes for 5x5 Mesh-torus network. The figure shows that the performance of multiple preemptions and single preemption have almost the same values. Figure 6.20 shows that the throughput when the smallest burst is chosen for preemption is better than when the preempted burst is chosen randomly.

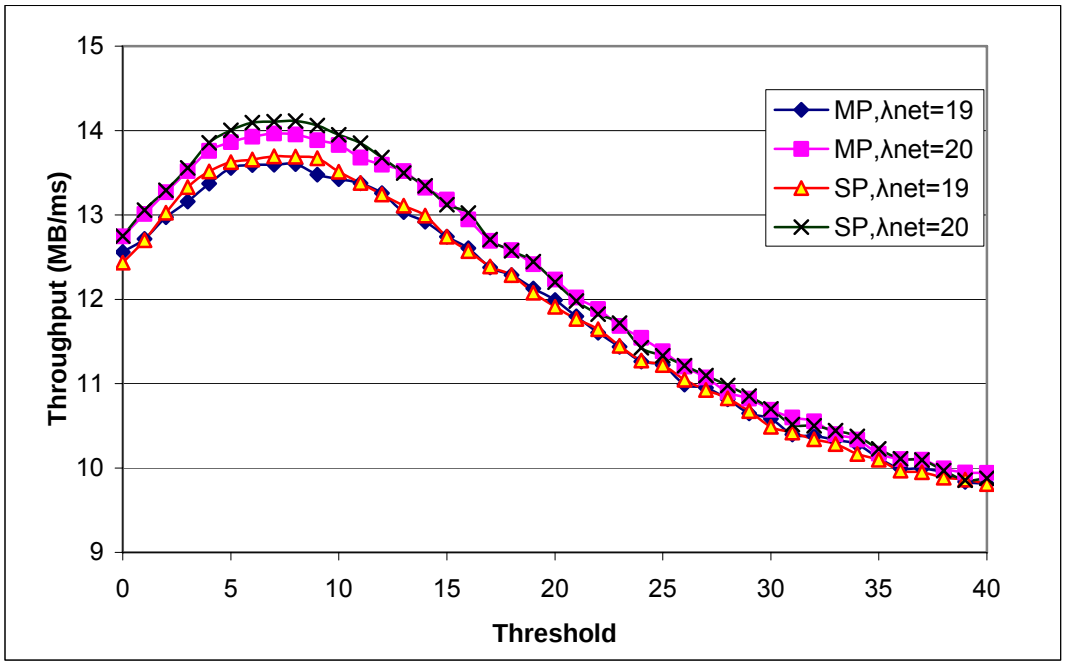


Figure 6.19. Throughput comparison for multi-preemption versus single-preemption, Mesh-torus 5x5 network, W=16

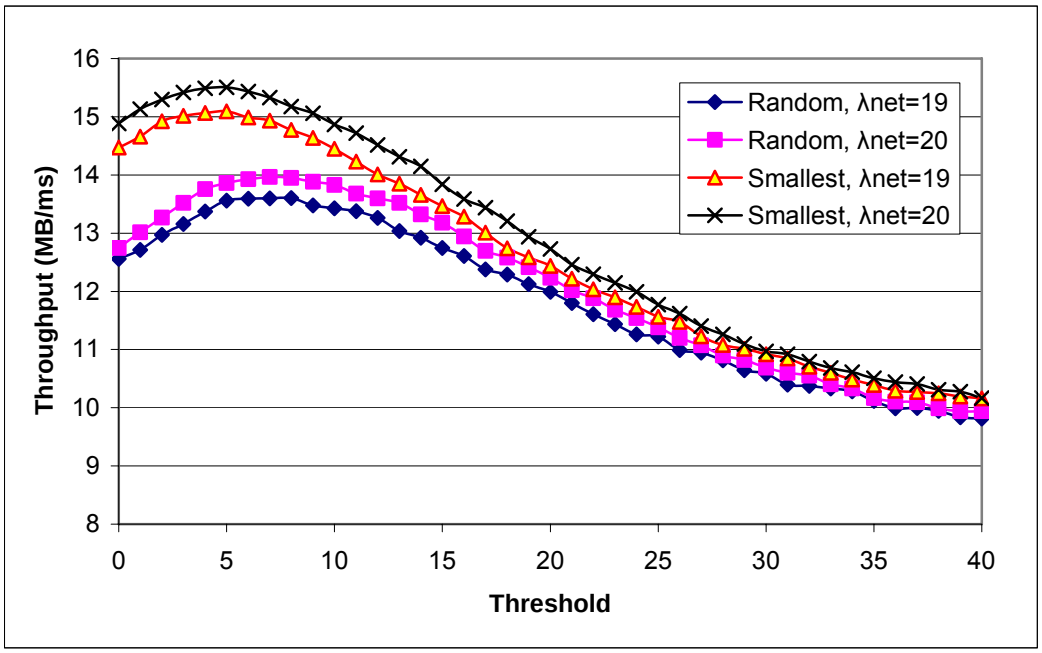


Figure 6.20. Throughput comparison for randomly selected burst versus smallest selected burst, Mesh-torus 5x5 network, W=16

Figures 6.21, 6.22 and 6.23 give the performance results for the LongHaul network under different scenarios. These results confirm the previous conclusions for the ring and Mesh-torus network. The three figures show that the throughput is improved by the preemptive scheme and the level of improvement is dependent on the threshold. Figure 6.23 shows that when the smallest burst size is chosen for preemption, the throughput performance is better than when the preempted burst is randomly selected.

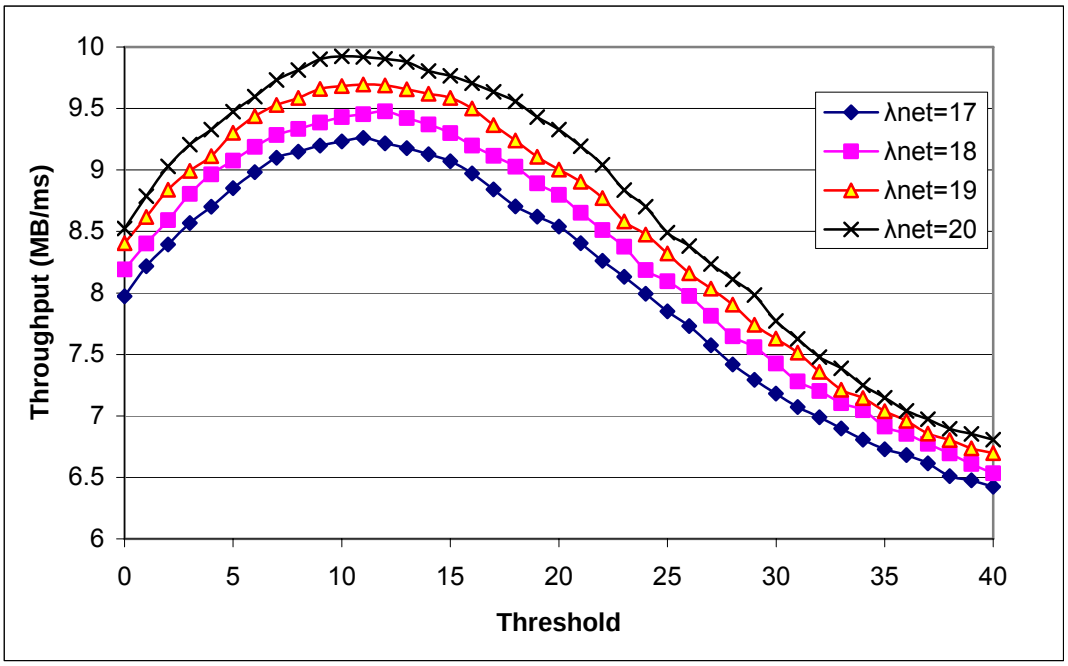


Figure 6.21. Throughput comparison for different threshold, LongHaul network, W=16

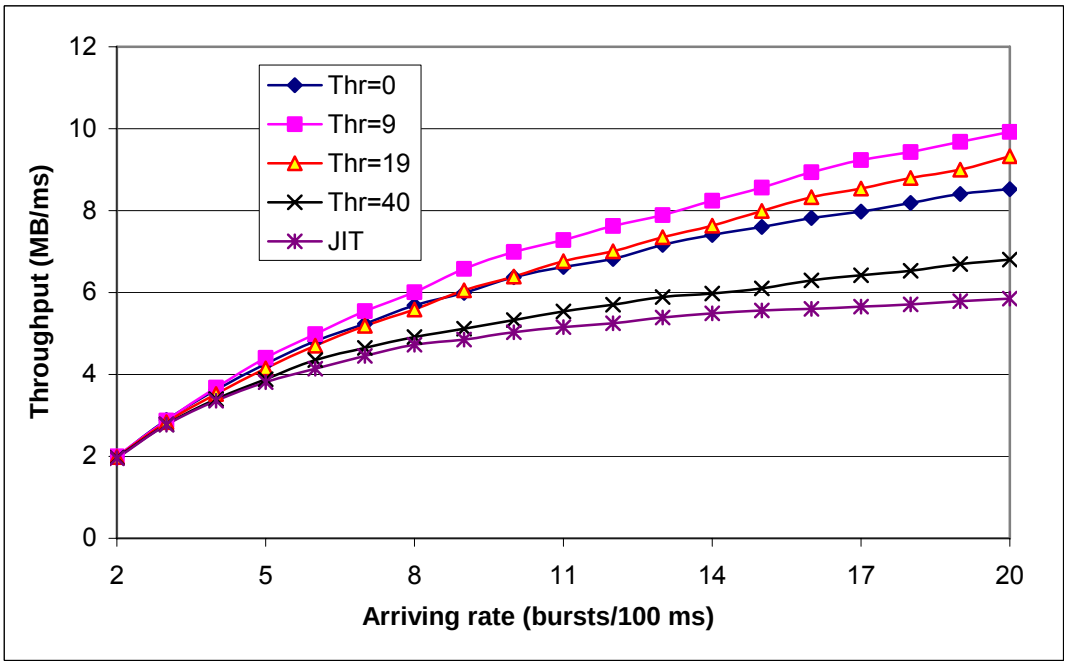


Figure 6.22. Throughput comparison for different thresholds, LongHaul network, W=16

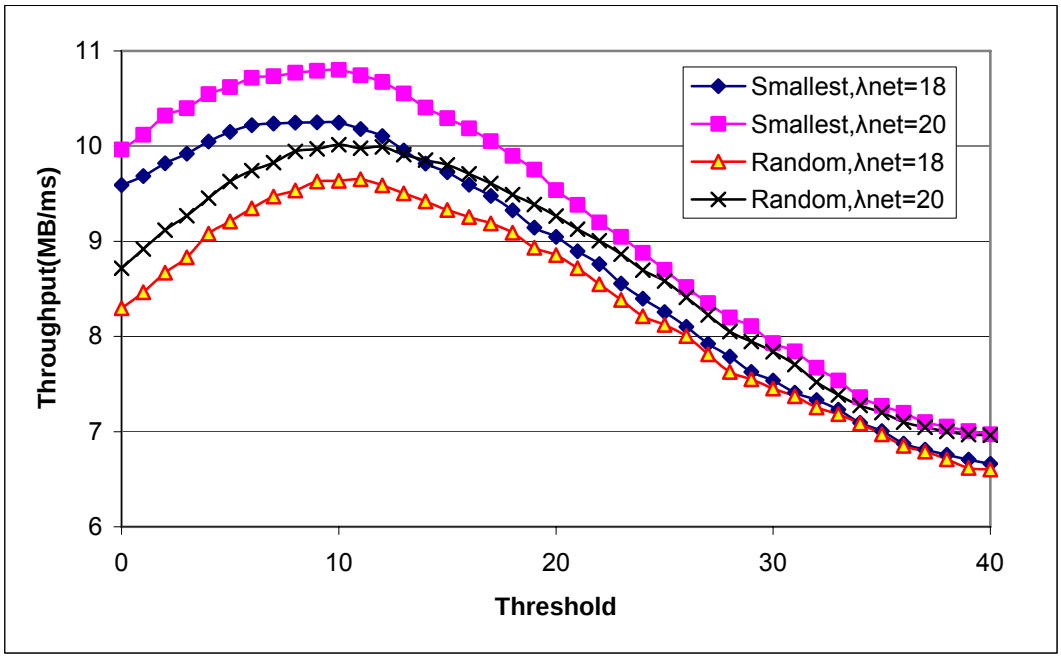


Figure 6.23. Throughput comparison for randomly selected burst versus smallest selected burst, LongHaul network, W=16

CHAPTER 7. CONCURRENT ENHANCEMENT OF NETWORK THROUGHPUT AND FAIRNESS IN OPTICAL BURST SWITCHING ENVIRONMENTS

7.1 Introduction

Optical burst switched (OBS) networks [30, 32] have received considerable attention in the literature. Research proposals to improve the performance and reliability of OBS networks have addressed various issues including QoS support [69, 73, 75], scheduling efficiency [71], blocking probability [12, 69, 73, 83, 84] and fairness.

Standard OBS scheduling protocols [30, 32] suffer from the hop count unfairness problem, i.e., bursts with longer lightpaths have higher dropping probability than bursts with shorter lightpaths. To alleviate this unfairness, bursts with large hop counts must be given higher scheduling priority over bursts with small hop counts. Schemes for improving fairness, however, can result in degrading the overall throughput performance of the OBS network. Likewise, schemes to reduce the blocking probability and improve the throughput of OBS networks can aggravate the burst unfairness problem. In this chapter, we propose schemes based on burst preemption to improve the fairness performance as well as the throughput performance in OBS networks.

The rest of the chapter is organized as follows. In section 7.2, a brief overview of related work is presented. In section 7.3, the model for our proposed preemption-based scheme is presented. In section 7.4, the preemption-based scheme is further enhanced by combining it with a non-preemptive fairness-improving scheme. In section 7.5, the simulation results are presented.

7.2 Some related works

There are two approaches to assemble the bursts in the network access station (NAS) of OBS networks: timer-based and threshold-based. In the timer-based burst assembly approach, the formation of the burst is restricted by a maximum (or periodic) time interval [81] and therefore bursts tend to have variable lengths. In the threshold-based approach [82], a limit is used to restrict the size of the burst (or the number of packets in the burst) and therefore bursts tend to have equal, or nearly equal, length. In this chapter, bursts are assumed to have variable sizes; our tests used exponential and uniform distributions for burst sizes.

There are several schemes in the literature that can also produce variable length bursts, even if the threshold-based burst assembly approach is used. Examples of these schemes are burst segmentation methods [12, 69, 80, 83, 84, 85]. These schemes are mostly used to reduce the packet loss and improve the throughput of OBS networks. Since an optical burst is an aggregation of many IP packets, the packet loss probability is reduced by applying segmentation that discards the initial part of a burst until a wavelength becomes free on the output fiber. Once a wavelength becomes free, the switch transmits only the remainder of the burst [80]. Another scenario occurs when preemption of a burst is allowed while the burst is being transmitted. In [12, 69] when contention occurs, the higher priority burst preempts the resource used by the lower priority burst, and the tail of the lower priority burst is discarded. In [80, 83], the benefit of burst segmentation is analyzed.

In [49], the authors proposed a merit-based scheduling algorithm for OBS networks. The scheme uses the route length and the current value of hop count to decide the preemption weight (priority) of a burst. Bursts with larger weight can preempt the channel resources reserved by bursts with smaller weight. The authors in [49] reported that the blocking probability is reduced by the merit-based scheme; however, the fairness issue is not discussed. Another preemption-based scheme is proposed in [74] based on proportional service differentiation. In [85], preemption is used as one of the approaches to handle channel contention and a model for analyzing preemptive burst segmentation is developed. Priority-based preemption is also used in [86], and in [73] a probabilistic preemptive scheme is proposed, mainly for the purpose of providing service differentiation.

7.3 Burst preemption for fairness and throughput improvement

The issue of fairness in OBS has received less attention than the issue of reducing burst blocking and improving throughput. In Chapter 3, we proposed two schemes to improve the fairness performance in OBS networks. In this chapter, we address the issue of improving fairness and throughput concurrently.

The preemption priority used in [49] is based on the weight $H*L$, where H is the current hop count of the burst (distance from the source node of the burst to the current node) and L is the route length (distance from the source node to the destination node). A burst can be preempted by another burst in an OXC node if the latter burst has a higher $H*L$ value than the former. Our extensive tests showed that the metric $H*L$ is not capable alone to improve fairness and

throughput concurrently. Guided by our tests, we modified the preemption weight function to include information about the burst size. The rationale for this is straightforward; bursts with large size contribute positively to the throughput of the network. Below, we briefly discuss some general preemption strategies in OBS networks then outline our proposed scheme for the concurrent improvement of fairness and throughput.

In normal OBS networks, a control packet is sent out from the source NAS at an earlier time before the actual data burst. Specifically, the transmission of the control packet and that of the data burst are separated by a time duration called the *offset time*. The control packet is used to reserve an optical channel for its data burst in each OXC along the lightpath of the burst. If the reservation is successful in all OXC's, the data burst can be transferred along the lightpath all-optically. If one OXC is congested and all channels are reserved, the control packet will fail to make a reservation for its data burst and the burst is dropped. Data bursts with longer lightpaths have higher probability of being dropped than bursts with shorter lightpaths. One way to alleviate this unfairness is to allow the control packet to reserve a channel by preempting the channel resource reserved by another control packet. We adopt the policy that the preemption process in any OXC should only occur before the arrival of the preempted burst. Specifically, we assume that no OXC is allowed to preempt an active data burst whose transfer through this OXC is in progress. When preemption occurs, two release processes take place: backward propagation to release the channels already reserved in the upstream path (toward source), and forward propagation to release the channels that may have been reserved in the downstream path (toward destination).

In order to improve both fairness and throughput, we use the following preemption weight function:

$$\omega = f(S) + H*L \qquad (7.1)$$

Where S is the burst size and $f()$ is a function to transfer the burst size into the catalog value of $H*L$. Our tests showed that better results are obtained when the mean value of $f(S)$ is roughly equal to the mean value of $H*L$. A simple way to construct $f(S)$ is to divide burst sizes into different groups (ranges), and assign an integer value to each group. The smallest value for $f(S)$ is 1 and corresponds to the burst group whose members have sizes that fall in the smallest range. The burst group whose members have the second smallest size range is assigned $f(S)$ value 2, and so on. The number of groups is chosen such that the mean value of $f(S)$ is roughly equal to the mean value of $H*L$. To offset the overhead of preemption, the preempting burst must have a higher preemption weight than the weight of the preempted burst. We use a parameter $\Re>1$, called the *preemption threshold*, to control the preemption process. Specifically, burst preemption is allowed only if the following constraint is satisfied:

$$\omega_{\text{preempting burst}} \geq \Re * \omega_{\text{preempted burst}}$$

We refer to the scheme that uses equation (7.1) as the Fairness and Throughput Improvement 1 (FATI1) scheme. Notice that the preemption weight of equation (7.1) has two terms. The goal of the first term, $f(S)$, is to improve the throughput performance by helping the bursts with larger size to preempt the resources reserved by bursts with smaller size. The second term, $H*L$, helps

improve fairness by giving advantage to bursts with long lightpaths and/or bursts that are close to reaching their destination.

We further improve equation (7.1) in order to alleviate the problem of reverse unfairness (i.e., bursts with long lightpaths start to gain advantage over bursts with shorter lightpaths). The improved weight function is given by:

$$\omega = f(S) + \mathrm{Min}\{H*L, c*E(H*L)\} \qquad (7.2)$$

Where *E(H*L)* is the average value of *H*L,* and *c* is a constant that is slightly larger than but very close to 1. The *Min* function ensures that bursts with long route paths will not have very high preemption weight and will be confined to the average value of the product *H*L*. The value *E(H*L)* can be easily computed for a given network topology. We denote the scheme that uses equation (7.2) as the Fairness and Throughput Improvement 2 (FATI2) scheme. The simulation results reported in section 7.5 demonstrate that using equation (7.2) gives better performance than using equation (7.1), i.e., the FATI2 scheme is better than the FATI1 scheme.

The FATI1 and FATI2 scheme are applicable to both types of signaling protocols: just-enough-time (JET) signaling [30] and just-in-time (JIT) signaling [32]. In this chapter, we used JIT to implement the FATI1 and FATI2 schemes.

7.4 Preemption combined with Balanced JIT (BJIT)

In Chapter 3, we proposed the balanced JIT (BJIT) scheme to improve the fairness performance of JIT. The idea of BJIT is to constrain the size of the search space allowed for a control packet within an OXC based on the length of the partial path of this control packet. Specifically, the number of channels n_i that are searched in the ith hop of a control packet is constrained by:

$$n_i = (1-g)*W + g*i*W/D \qquad (7.3)$$

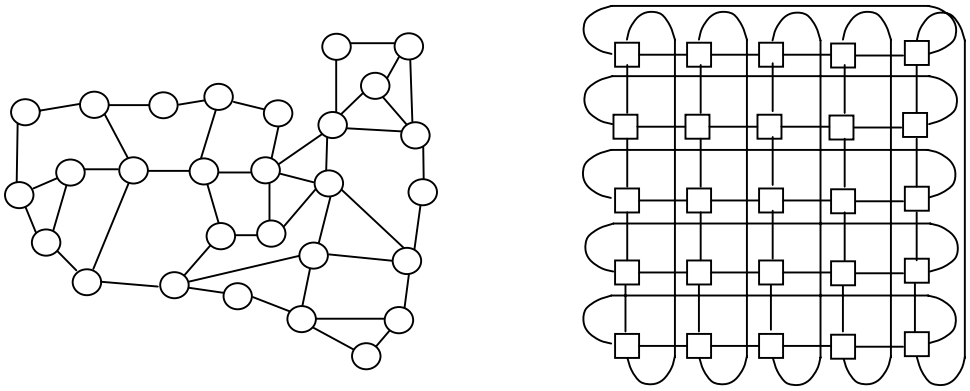
where g is a parameter that is assigned a value between 0 and 1, D is the diameter of the network (the maximum number of hops in a shortest path in the network), i is the current hop count of the control packet (i.e., number of hops from the source node to the current node), and W is the total number of channels available in the current node. The BJIT scheme improves fairness but does not improve throughput. In this chapter, we further improve the FATI2 scheme proposed in section 7.3 by combining it with the BJIT scheme. The resulting scheme is called FATI2B.

In the FATI2B scheme, a control packet is allowed to search only n_i channels when it arrives at its ith hop. The starting position of the search (within the circular space of W channels) is randomized to improve performance. If the constrained search does not find a free channel, the preemption scheme FATI2 proposed in section 7.3 takes place. In section 7.5, the test results show that this combined scheme gives a further fairness-throughput performance improvement over the FATI2 scheme. In Chapter 3, the value g=0.5 was found to be the best parameter value that improves the fairness performance of BJIT without negatively impacting the throughput

performance. The value g=0.5 means that each control packet is guaranteed to search at least 50% of the total search space of *W* channels. However, since equation (7.2) of the FATI2 scheme already has a term to improve fairness, a smaller value of *g* is needed in the FATI2B scheme. Our tests showed that a value of g=0.2 is the best value when the BJIT is combined with FATI2. The value g=0.2 means that each control packet is guaranteed to search at least 80% of the total search space of *W* channels.

7.5 Simulation results

In this section, we report the results of our extensive simulation tests using two network topologies: the US LongHaul network with 28 nodes and a 5x5 mesh-torus network, as shown in Figure 7.1.



LongHaul topology 5x5 Mesh-torus topology

Figure 7.1. Two network topologies

In our simulation, a static lightpath between any two nodes is established using the shortest path first method as was done in [30, 49]. Notice that the diameter of the LongHaul topology has 7 hops and the diameter of the 5x5 Mesh-torus topology has 4 hops. Similar to [26, 49, 87], the traffic used in our tests is uniformly distributed among all nodes. This means that all nodes have equal likelihood to be the source of a data burst and for a given source node, all other nodes in the network have equal likelihood to be the destination node. As in [30, 49], we assume that full wavelength conversion capability is available in all OXCs.

In our simulation tests, assembled bursts arrive according to a Poisson distribution with controllable arrival rate. For each burst, the source node and the destination node are randomly selected as explained before. Our simulation tests used parameter values similar to those typically used in the literature: the cut through time in each OXC is 2.5 milliseconds, the link delay per hop is 3 milliseconds, the average burst length is 50 microseconds (equivalent to a burst of 250 Kbits at 5 Gbits/second), the control packet processing time t_p at each hop is 50 microseconds. The number of wavelengths used in each fiber is W=16 in each direction. Each point in the performance graphs reported in this chapter was obtained by averaging the results of 6 simulation tests using different randomly generated seeds. We used multiple batch means at 95% confidence interval and each simulation was run for sufficiently long time to obtain stable statistics; the total number of bursts processed in each simulation test ranged from 2.5 million bursts at low arrival rates to 25 million bursts at high loads for the LongHaul network and from 4 million bursts at low arrival rates to 40 million bursts at high loads for the Mesh-torus network.

The unit of time (denoted *ut*) used in the graphs presented in this chapter is equal to 0.05 millisecond. Thus a load of 12 bursts/ut is equivalent to a total network load of 60 Gbits/second.

Figures 7.2 to 7.5 give the fairness results of the three schemes: original JIT, FATI1 (using equation 7.1) and FATI2 (using equation 7.2) for the Mesh-torus network. We experimented with various values of the preemptive threshold $\Re$ and selected values that improve both fairness and throughput. The preemptive threshold used for FATI1 and FATI2 in Figures 7.2-7.5 is $\Re$=1.2, i.e., preemption occurs only if the weight of the preempting burst is equal to or greater than 1.2 times the weight of the preempted burst. As shown in Figure 7.2, the standard JIT scheme has poor fairness performance. Bursts with longer lightpaths have higher blocking probability than bursts with shorter lightpaths and the unfairness problem is exacerbated as the load increases. Figures 7.3 and 7.4 show that the FATI1 and the FATI2 schemes improve the fairness performance compared to JIT. Figure 7.5 gives detailed comparison of the three schemes at load 4; the figure shows that the FATI2 scheme has the best fairness (least variation in blocking probabilities) among the three schemes.

Figure 7.6 shows that for loads higher than 2, the throughput of FATI2 and FATI1 in the Mesh-torus network is significantly better than the throughput of the original JIT scheme.

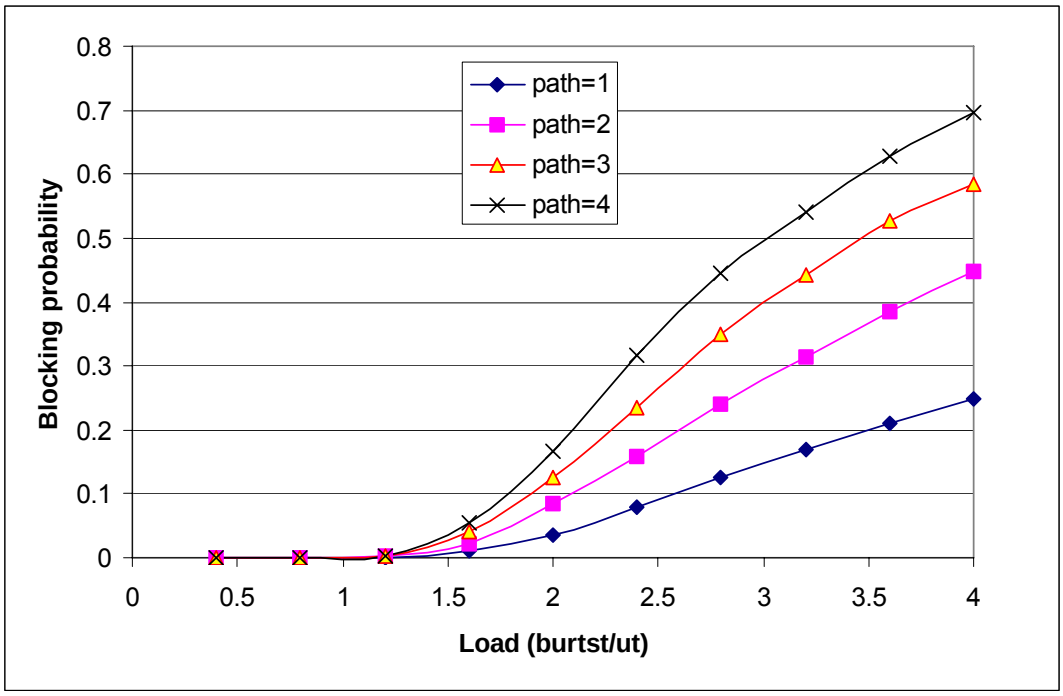


Figure 7.2. JIT blocking probability distribution, Mesh-torus 5x5 network

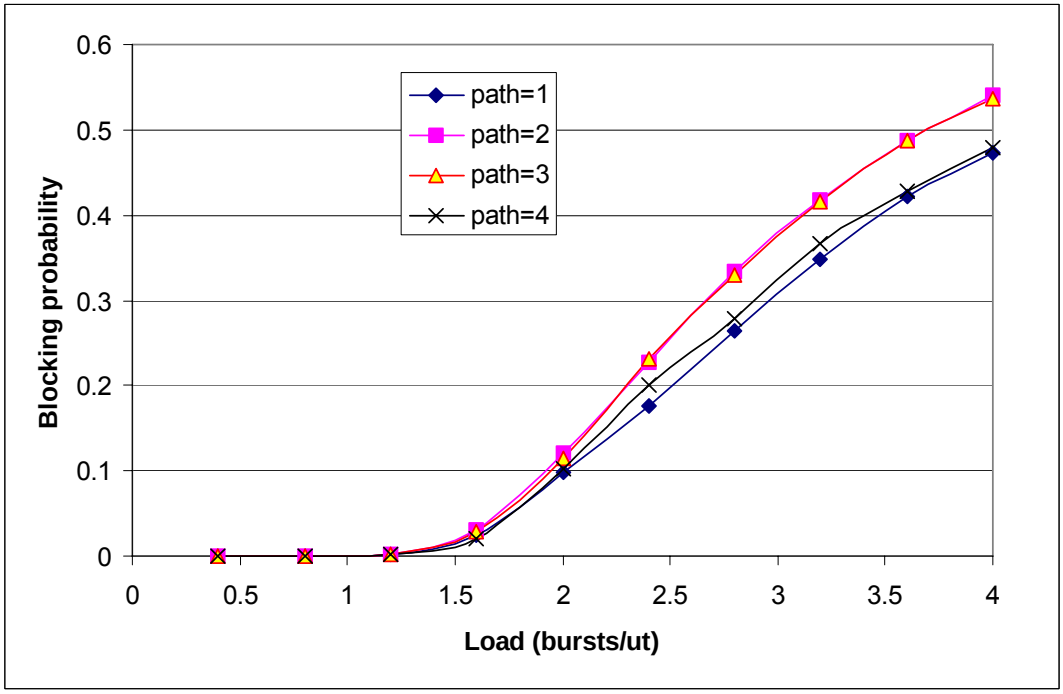


Figure 7.3. FATI1 blocking probability distribution, Mesh-torus 5x5 network

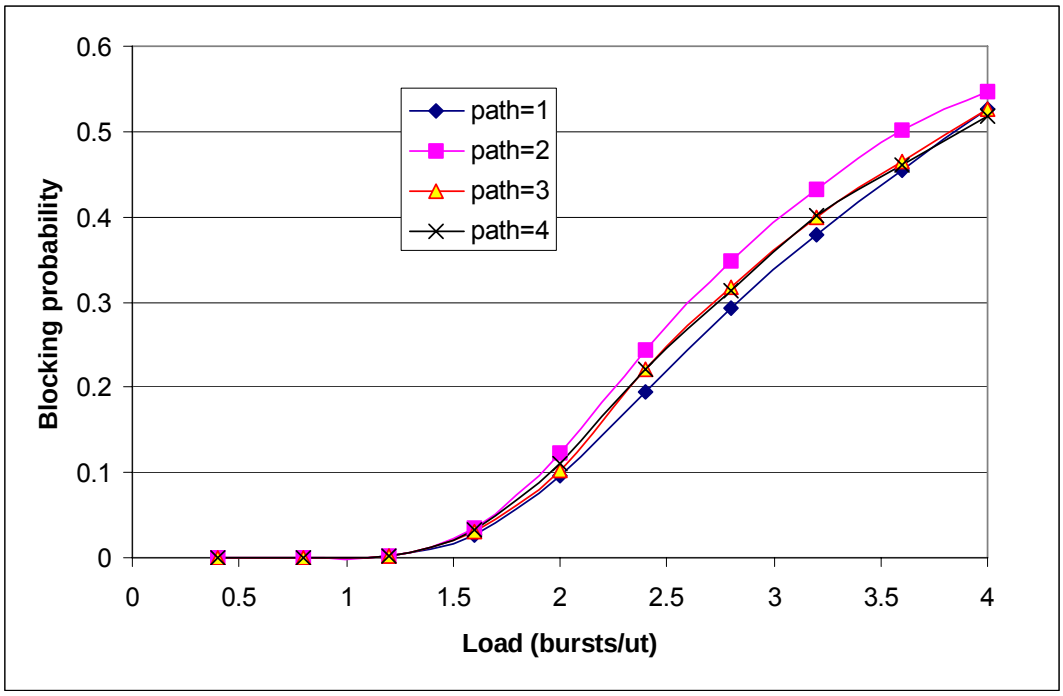


Figure 7.4. FATI2 blocking probability distribution, Mesh-torus 5x5 network

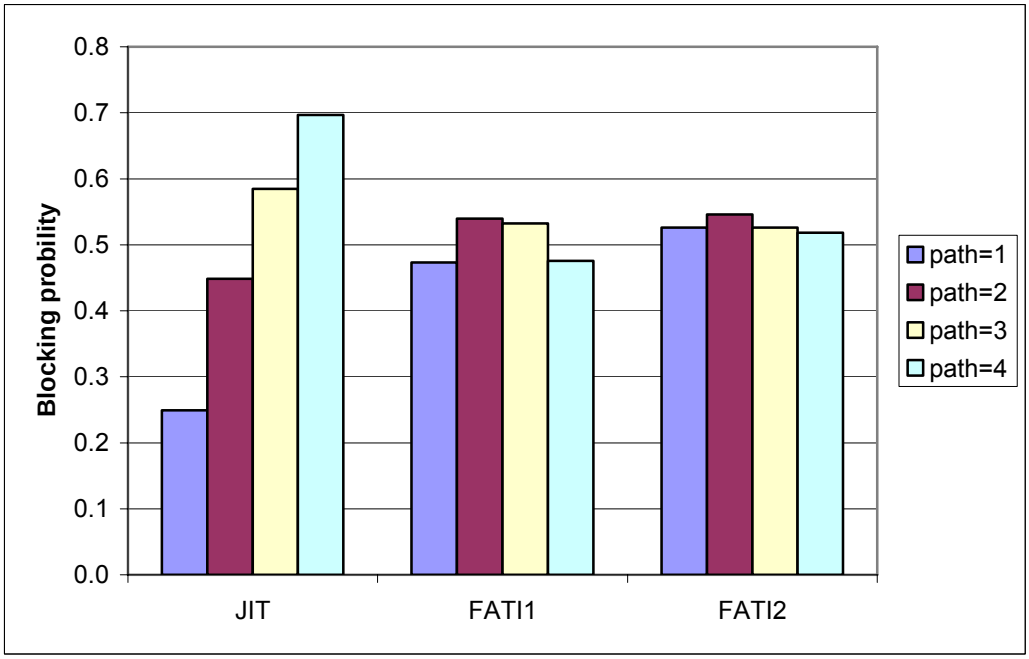


Figure 7.5. Blocking probability comparison among JIT, FATI1 and FATI2 at load =4, Mesh-torus 5x5 network

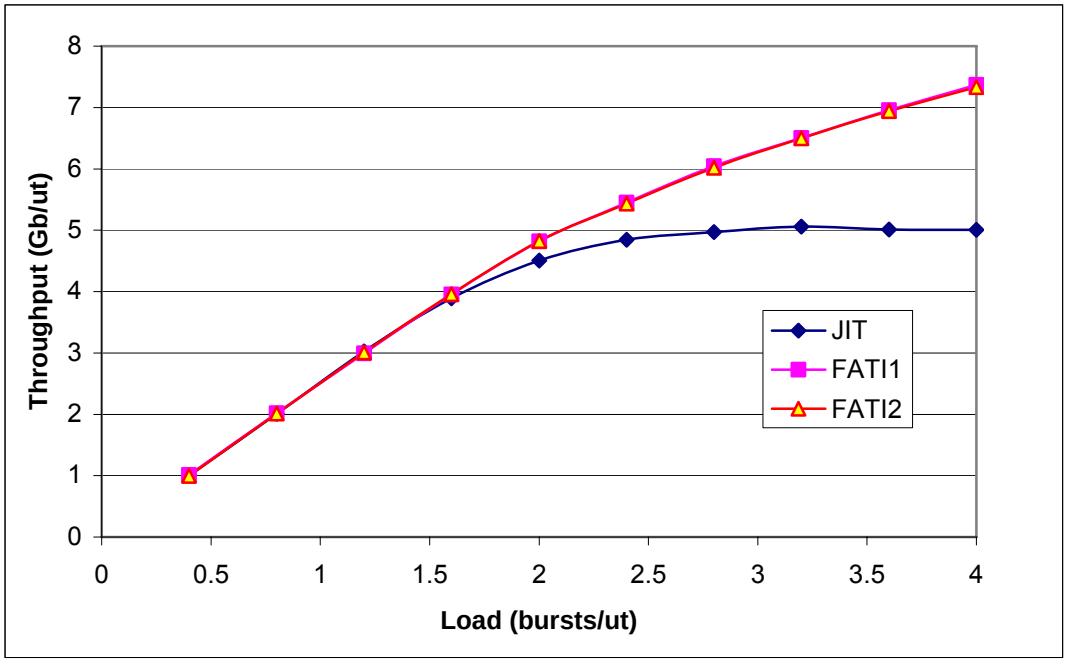


Figure 7.6. Throughput comparison among JIT, FATI1 and FATI2, Mesh-torus 5x5 network

Figure 7.7 and 7.8 give the blocking probabilities of JIT, FATI1, FAIT2 in the U.S. LongHaul network at loads 2 and 2.5, respectively. The preemptive thresholds used in FATI1 and FATI2 is $\Re$=1.2 (i.e., the preempting burst should be at least 20% larger than the preempted burst). Both figures clearly show that FATI2 has the best fairness performance (least variation in blocking probabilities) among the three schemes. Figure 7.9 shows that the FATI2 scheme has the best throughput performance followed closely by the FATI1 scheme. Both FATI1 and FATI2 have significantly better throughput than JIT. It should be noted that the throughput plotted in Figure 7.6 and Figure 7.9 are the average throughput over all bursts. Due to the uniform distribution on source-destination pairs, bursts with different lightpath lengths have different arrival rates. Specifically, the uniform distribution on node pairs gives the following distribution on the

number of hops in the burst's lightpath: for the 5x5 mesh topology, the percentage of lightpaths with number of hops equal to 1, 2, 3 and 4 is 16.7%, 33.3%, 33.3%, and 16.7% respectively; for the LongHaul topology, the percentage of lightpaths with number of hops equal to 1, 2, 3, 4, 5, 6 and 7 is 12%, 20%, 23%, 21%, 14%, 8% and 2%, respectively.

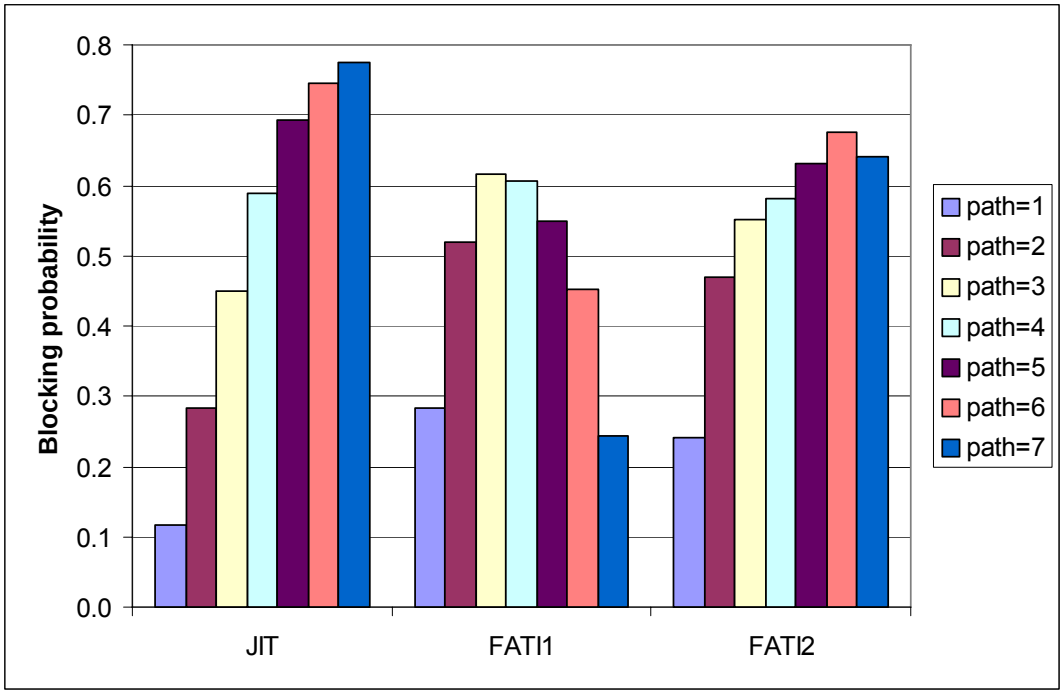


Figure 7.7. Blocking probability comparison among JIT, FATI1 and FATI2 at load =2, LongHaul network

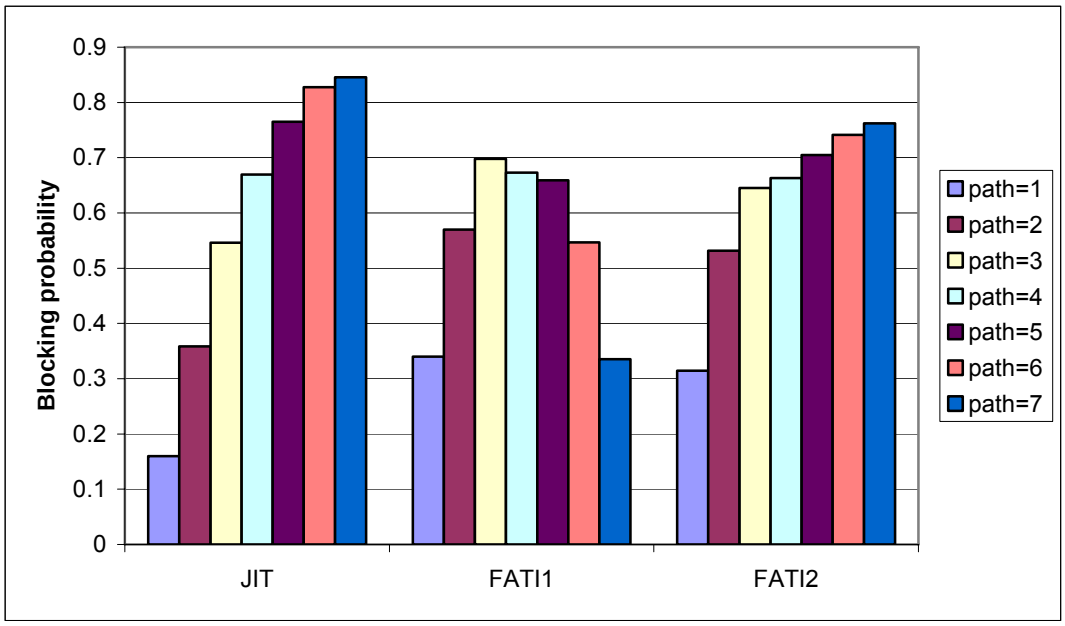


Figure 7.8. Blocking probability comparison among JIT, FATI1 and FATI2 at load =2.5, LongHaul network

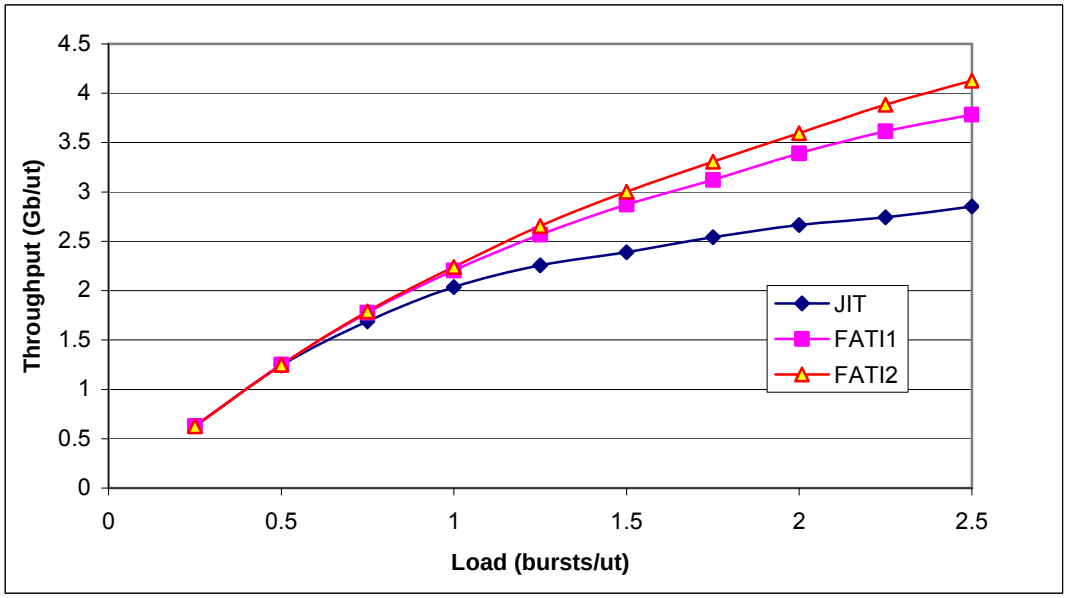


Figure 7.9. Throughput comparison, LongHaul network

We further validate the previous results by a quantitative fairness metric. In [88], the authors discuss methods to alleviate the inherent unfairness of TCP towards connections with long round-trip times. They use the coefficient of variation of the individual throughputs of TCP flows as a metric to evaluate fairness. We borrow this concept and define the *fairness coefficient* in optical burst switched networks as the coefficient of variation (standard deviation over mean) of the individual average blocking probabilities for bursts with different hop lengths. Figure 7.10 shows the values of the fairness coefficient for the three schemes JIT, FATI1, and FATI2 at four different configurations: Mesh-torus network at load 3.2, Mesh-torus network at load 4.0, LongHaul network at load 2.0 and LongHaul network at load 2.5. The FATI2 scheme has the lowest fairness coefficient in all cases.

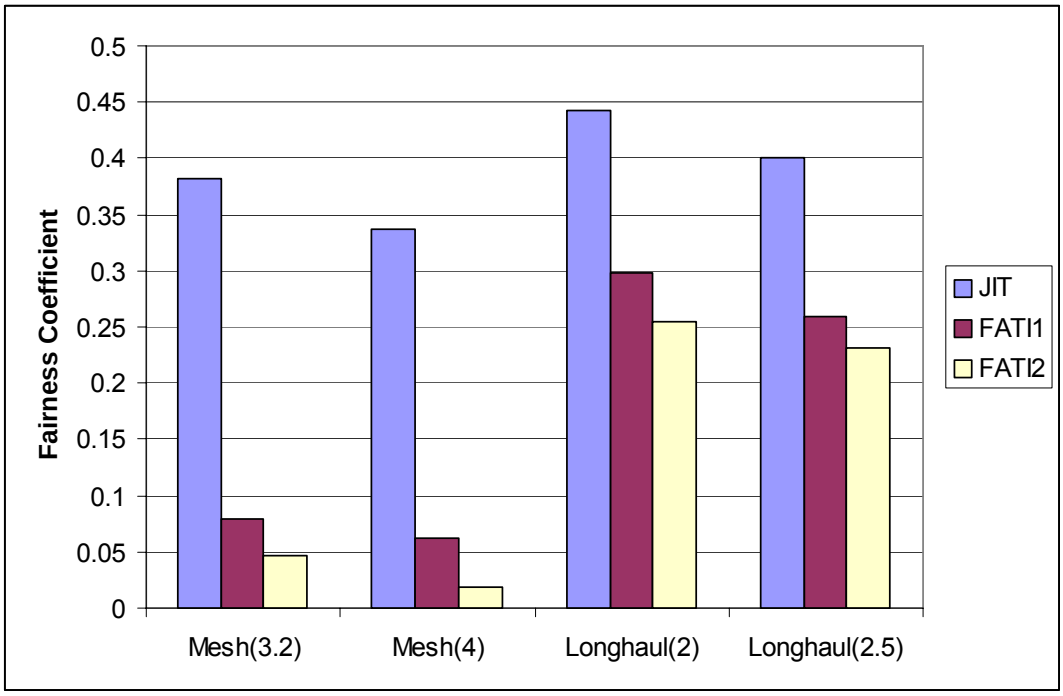


Figure 7.10. Fairness coefficient comparison of JIT, FATI1 and FATI2 for four configurations

As has been shown in Figures 7.2-7.10, the fairness of the FATI2 scheme is better than that of the FATI1 scheme and the throughput of FATI2 is slightly better than that of FATI1. In the remainder of this chapter, the FATI2 scheme will be used for further fine tuning and comparisons.

Figures 7.11-7.13 show the improvement obtained in the LongHaul network by combining the preemption-based FATI2 scheme with the non-preemptive BJIT scheme presented in Chapter 3. The resulting scheme, denoted FATI2B, uses burst preemption based on equation (7.2) and also uses a parameterized wavelength search based on equation (7.3). Figure 7.11 shows the blocking probabilities at load 2 and Figure 7.12 shows the blocking probabilities at load 2.5. The value of g used in the BJIT scheme in Figures 7.11 and 7.12 is equal to 0.5. This value gives the best fairness-throughput performance results for the non-preemptive BJIT scheme. When BJIT is used in conjunction with burst preemption, a smaller value of g is needed to get best fairness-throughput results; the value of g used for the FATI2B scheme in Figures 7.11-7.12 is 0.2. Figure 7.13 shows that the FATI2B scheme has slightly higher throughput than FATI2 and much better throughput than JIT and BJIT at all loads. We conclude that the combined application of equations (7.2) and (7.3) produces a scheme, FATI2B, that has better fairness-throughput properties than the FATI2 scheme that uses equation (7.2) only.

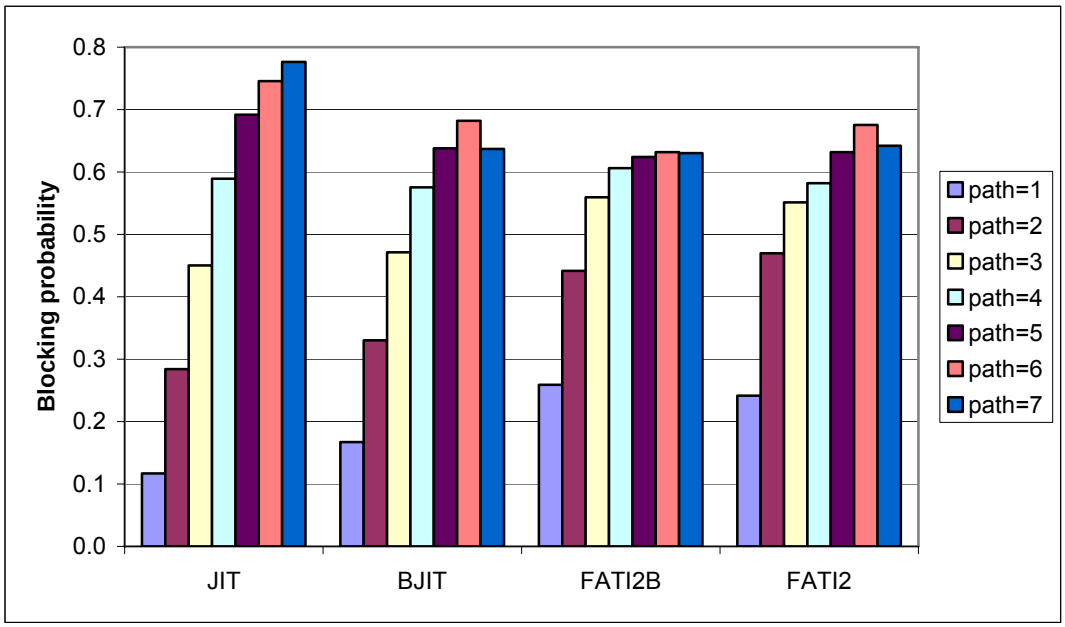


Figure 7.11. Blocking probability comparison among JIT, BJIT, FATI2 and FATI2B for LongHaul network, load = 2 (bursts/ut)

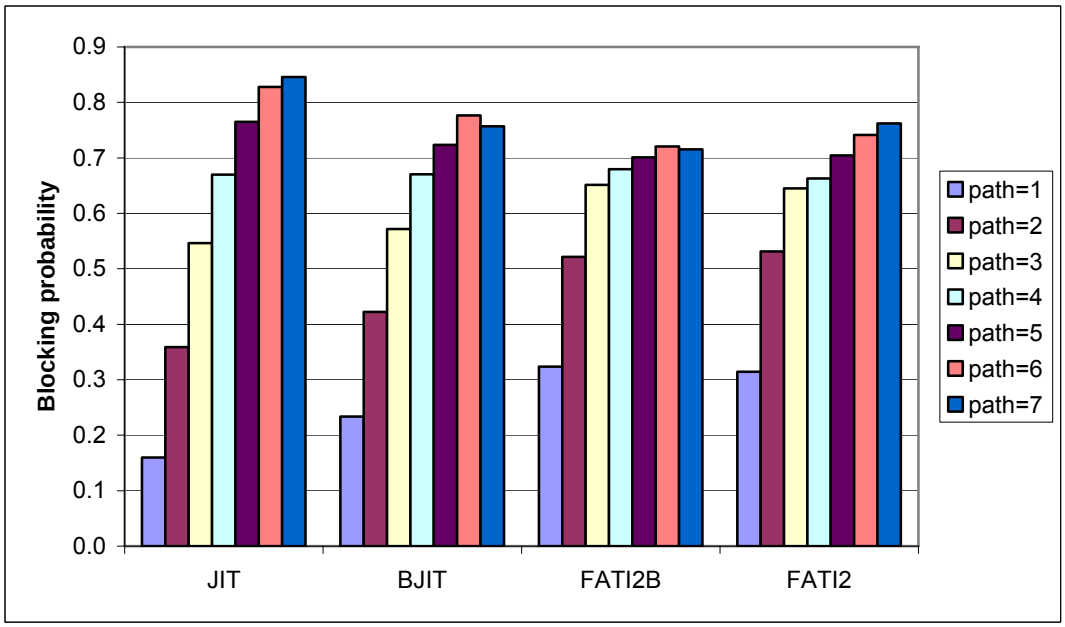


Figure 7.12. Blocking probability comparison among JIT, BJIT, FATI2 and FATI2B for LongHaul network, load = 2.5 (bursts/ut)

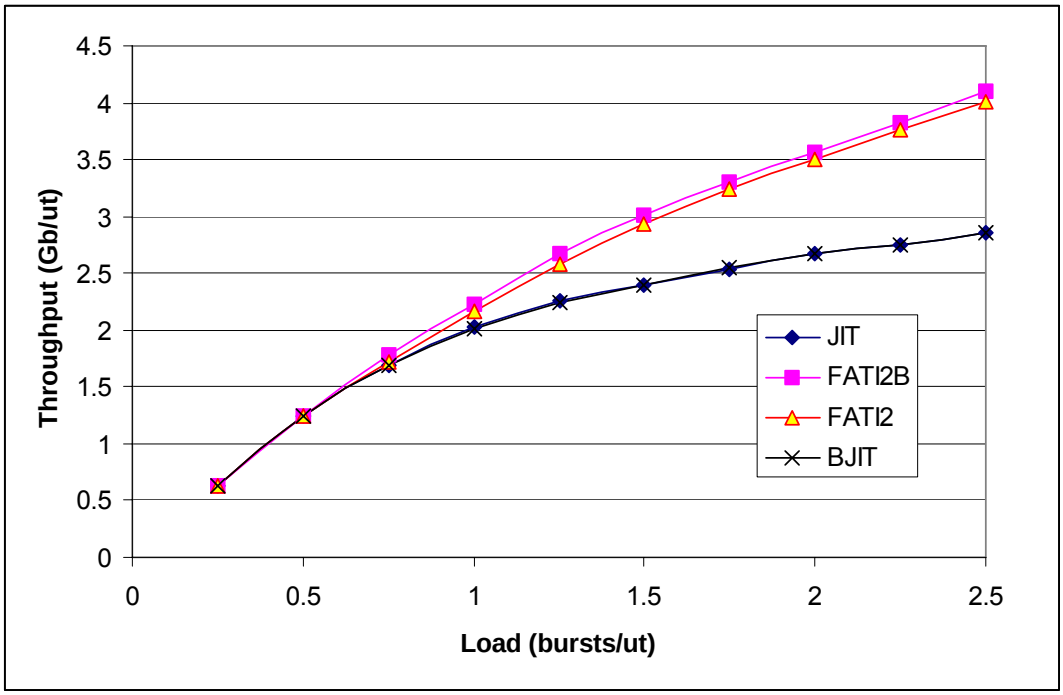


Figure 7.13. Throughput comparison among JIT, FATI2B (g=0.2), FATI2 and BJIT (g=0.5), LongHaul network

Figure 7.14 and 7.15 show the corresponding results for FATI2B in the Mesh-torus network. The FATI2B scheme improves fairness compared to FATI2 (Figure 7.14) and gives equal or slightly better throughput (Figure 7.15).

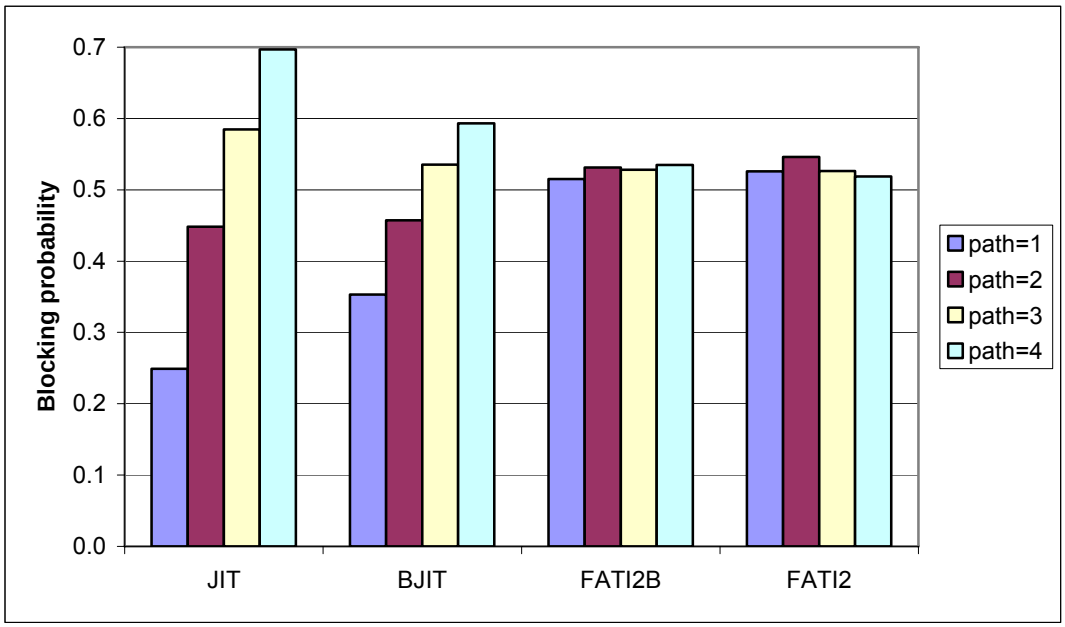


Figure 7.14. Blocking probability comparison among JIT, BJIT(g=0.5), FATI2 and FATI2B(g=0.2) in Mesh-torus 5x5 network at load=4

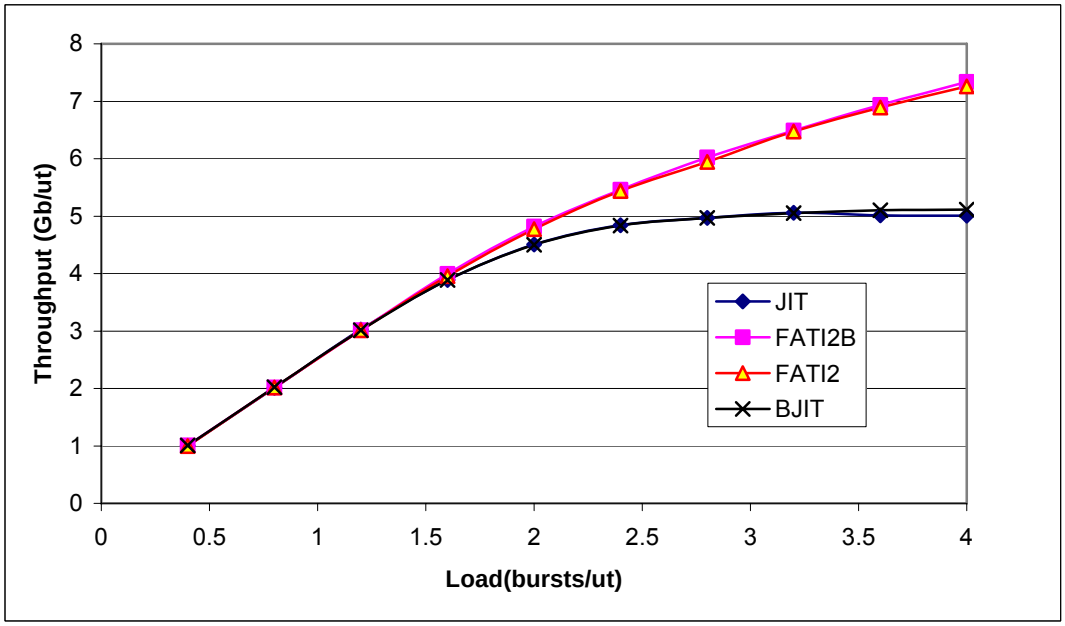


Figure 7.15. Throughput comparison among JIT, BJIT (g=0.5), FATI2 and FATI2B (g=0.2), Mesh-torus 5x5 network

Figure 7.16 gives the fairness coefficient comparison among the four schemes JIT, BJIT, FATI2 and FATI2B in the LongHaul network. The FATI2B scheme gives the lowest fairness coefficient at all loads. Similar results were obtained for the mesh network.

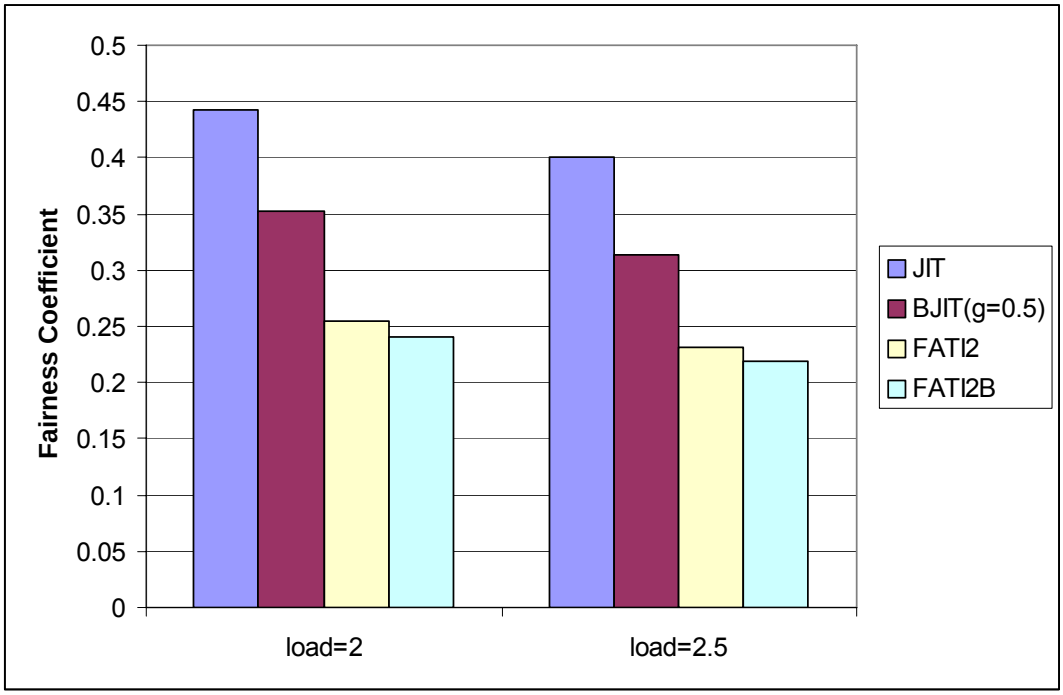


Figure 7.16. Fairness coefficient comparison for JIT, BJIT, FATI2 and FATI2B for LongHaul network at load=2 and load=2.5

Next we further fine tune the FATI2B scheme and present comparison results that offer more insight into the performance of our scheme. Two additional variations of the FATI2 scheme are included in the comparisons, FATI-v4 and FATI-v5. In the FATI-v4 variation, the f(s) term of equation (7.2) is divided by 4 to reduce emphasis on throughput. Thus FATI-v4 replaces equation (7.2) by equation (7.4) given below and represents a FATI2 scheme that is intended to

give less priority to improving throughput. The FATI-v5 variation, on the other hand, replaces equation (7.2) by equation (7.5), i.e., the second term is divided by 3 to increase emphasis on improving throughput.

$$\omega = 0.25*f(S) + Min\{H*L, E(H*L)\} \qquad (7.4)$$

$$\omega = f(S)+ 0.334*Min\{H*L, E(H*L)\} \qquad (7.5)$$

Figure 7.17 gives the blocking probability comparison among JIT, FATI-v4, FATI-v5, FATI2B(g=0.2), FATI2B (g=0.5) for the LongHaul network at load 2 (similar trends were obtained at other loads). Figure 7.18 gives the throughput comparison among these five schemes at different loads. The FATI-v4 scheme, with its dominant *H*L* factor, eliminated the unfairness problem against bursts with long lightpaths but unfortunately created a reverse unfairness problem, i.e., bursts with longer lightpaths enjoy smaller blocking probabilities than bursts with shorter lightpaths. The FATI-v5 scheme, with its dominant *f(s)* factor, improved the throughput over JIT. Figure 7.18 shows that FATI-v5 has the best throughput among all five schemes at all loads. The FATI2B(g=0.2) scheme comes a close second followed by FATI2B(g=0.5). The JIT and FATI-v4 scheme have the worst throughput. Figure 7.19 shows the fairness coefficient of the five schemes for the LongHaul network at loads 2 and 2.5. Comparison tests for using different values of the parameter *g* of equation (7.3) have shown that using a value of g=0.2 is the best compromise for FATI2B. The values g=0.2 and g=0.5 have approximately the same fairness coefficient but the value g=0.2 has consistently given higher throughput than g=0.5. For the

Mesh-torus network, we obtained similar results in the comparison of fairness and throughput among the five schemes (results are omitted).

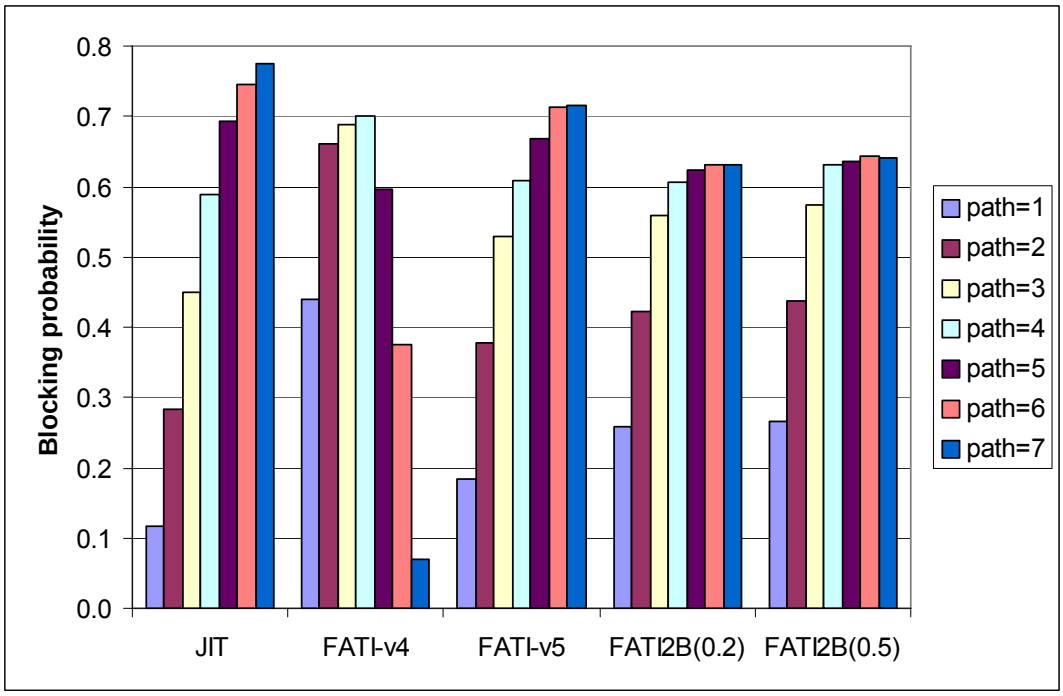


Figure 7.17. Blocking probability comparison among JIT, FATI-v4, FATI-v5, FATI2B(0.2), FATI2B (0.5), at load =2, LongHaul network

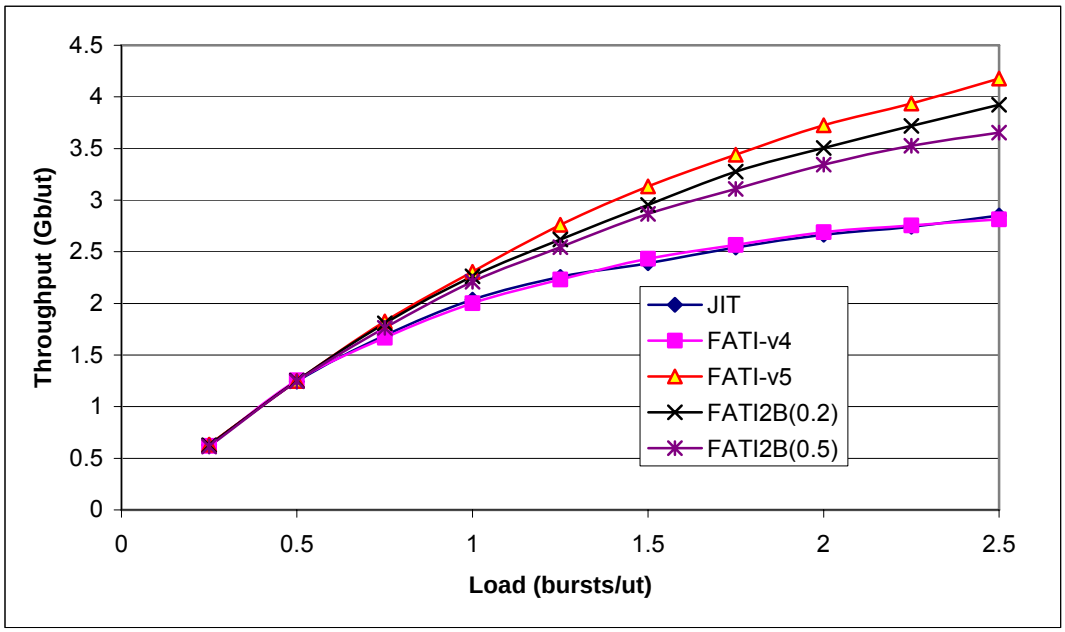


Figure 7.18. Throughput comparison among JIT, FATI-v4, FATI-v5, FATI2B(0.2), FATI2B (0.5) at load =2, LongHaul network

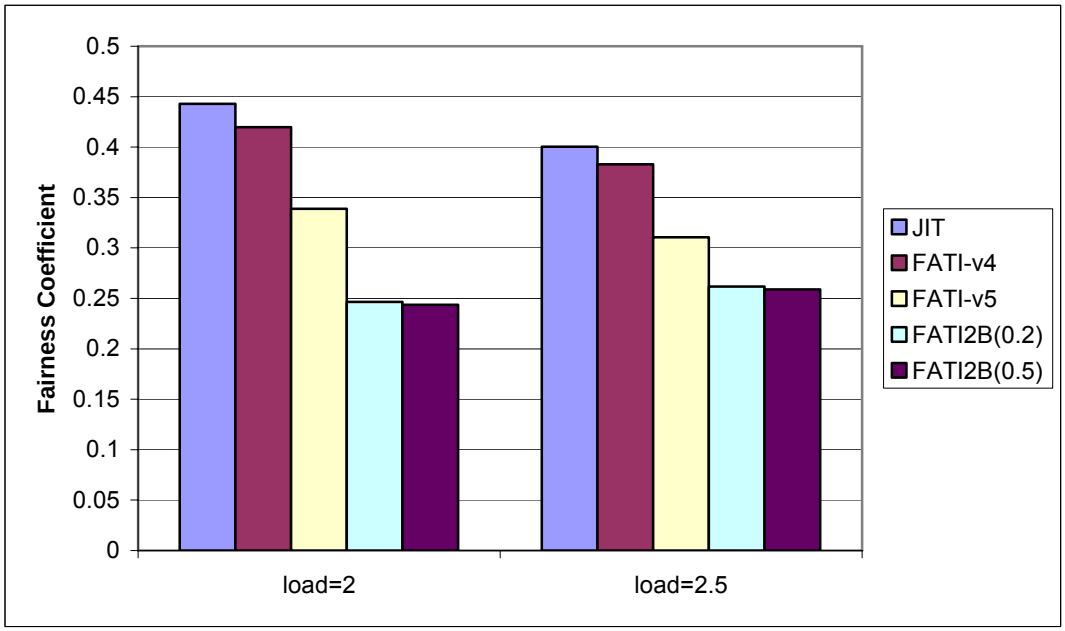


Figure 7.19. Fairness coefficient comparison for JIT, FATI-v4, FATI-v5, FATI2B(0.2) and FATI2B(0.5) for LongHaul network at load=2 and load=2.5

The distributions of burst sizes used in all previous results are exponential distributions. We also performed tests with burst sizes having uniform distributions. For both the LongHaul and Mesh-torus networks, the results for the uniform burst size distributions have given the same general trends as the results obtained for the exponential distributions. Basically, using burst preemption based on equation (7.2) significantly improves the fairness as well as the throughput performance of OBS networks. Combining equation (7.2) with the parameterized wavelength search of equation (7.3) further improves the fairness and throughput performance.

CHAPTER 8. CONCLUSIONS

Below, we present a summary of the contributions of the work presented in this dissertation and discuss future research ideas to extend this work.

8.1 Efficient routing and channel assignment

We proposed and evaluated a new approach for implementing efficient RWA in WDM networks with multiple-fiber links. Our method uses a compact bitmap representation that uniformly applies to fibers, links, and lightpaths. A modified Dijkstra's shortest path algorithm is developed for dynamic routing based on the bitmap representation and the efficient logical intersection operation. A first-fit channel assignment algorithm is developed using a simple computation on the bitmap of the selected route and a shortest hop criterion is used to break ties. The proposed bitwise routing algorithm combines the benefits of least loaded and shortest path routing algorithms. Our simulation tests have shown that the blocking performance of our RWA method is better than that of three previous schemes.

One possible extension of our proposed RWA method is to consider the availability of wavelength converters as a way to reduce the blocking probability. Wavelength conversion is still a maturing and costly technology. Fortunately, it has been shown that the sparse deployment of wavelength converters can achieve significant benefit comparable to that obtained by full deployment of converters in all nodes of the network. The impact of converters on the bitwise RWA method can be evaluated in two steps. First, the converter placement algorithm proposed

in [25-27] can be applied to select the nodes that will be equipped with wavelength converters. Second, the RWA method is modified to take advantage of the presence of converters in some nodes. Encountering a node with a converter during the routing search removes the constraint of wavelength continuity and enables the routing algorithm to start a new search phase, thus improving the prospects of accepting the request.

8.2 Proactive schemes for improving fairness and providing differentiated QoS

We investigated two methods for alleviating the beat down problem in OBS networks. The first scheme, Balanced JIT, uses a simple equation to adjust the size of the search space for a free wavelength based on the number of hops traveled by the burst. The BJIT scheme has a single parameter and is implemented in each OXC. The second scheme, PRED, uses proactive discarding to reduce the probability of dropping bursts with large hop count at the expense of an increase in the dropping probability of bursts with small hop count. The PRED scheme is implemented in the source network access stations and does not therefore waste any bandwidth resources in the core of the optical network. Detailed performance results showed that both BJIT and PRED can alleviate the beat down unfairness without negatively impacting the overall throughput of the system.

We further modified the BJIT and PRED schemes and extended them in order to support QoS differentiation in OBS networks. The first scheme adjusts the size of the search space for a free wavelength based on the priority level of the burst. The second scheme uses different proactive discarding rates in the network access station of the source node. The first scheme has less

number of parameters and is easier to implement/tune than the second scheme. Performance simulation tests showed that both schemes are capable of providing tangible QoS differentiation without negatively impacting the throughput of OBS networks.

In our implementation of the two schemes for fairness and for QoS differentiation, we used the JIT scheduling protocol and assumed the availability of full wavelength converters in each node in the network. One future extension of this work is to apply the two schemes to the JET scheduling protocol. Another extension is to investigate the case in which wavelength converters are not available in all nodes of the network.

8.3 Preemption-based schemes for improving fairness and throughput

We proposed two schemes, RPJIT and HPJIT, to improve the fairness performance in OBS networks based on burst preemption. Our schemes use carefully designed constraints to avoid excessive wasted channel reservations, reduce cascaded useless preemptions, and maintain healthy throughput levels. Our extensive simulation results showed that the constrained preemption schemes improve fairness compared to previous methods without degrading network throughput.

We also proposed and fine-tuned a preemption-based scheme for improving the throughput of OBS networks based on the burst size. Extensive simulation results on different network topologies showed that the scheme significantly improves the throughput of the network. We developed an analytical model to compute the throughput of the network iteratively for the

special case when the network has a ring topology. The analytical model has been shown to be accurate; it gives results close to those obtained by simulation.

Finally, we proposed a preemption-based scheme for the concurrent improvement of throughput and fairness. The preemption weight of an incoming burst is a function of the route length, the current hop count and the burst size. Extensive simulation tests showed that the application of the scheme results in higher throughput and at the same time improves the fairness coefficient compared to the standard JIT scheme.

The preemption-based schemes described above can be extended in many ways. As mentioned in section 8.2, one area of future research is to adapt these schemes to JET scheduling. Adapting preemption-based schemes to JET scheduling requires more sophisticated design than adapting the proactive schemes mentioned in section 8.2. This is because the preemption of a reserved channel must be handled carefully since the size of the burst plays a more critical role in JET's channel reservation. Another area of future research is to extend the analytical model for estimating the throughput of ring OBS networks to other topologies. Extending this analytical model to arbitrary topologies may be quite difficult and we may therefore need to start by tackling topologies with uniform structure such as the mesh-torus topology. A third area of future research is to adapt the schemes to dynamic routing. Static routing is easier to implement than dynamic routing, but is generally less capable of achieving maximum throughput levels. The offset times for the JIT or JET protocols are best suited for static routing and these protocols will need to be redesigned if dynamic routing is used.

LIST OF REFERENCES

[1]. R. Ramaswami and K. N. Sivarajan, "Routing and wavelength assignment in all-optical networks", IEEE/ACM Transactions on networking, vol. 3, no. 5, Oct. 1995, pp. 489-500.

[2]. Z. Zheng and A.S. Acampora, "A heuristic wavelength algorithm for multihop WDM networks with wavelength routing and wavelength re-use", IEEE/ACM Transactions on networking, vol. 3, no. 3, Jun. 1995, pp. 281-288.

[3]. H. Zhang, J.P.Jue and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks", SPIE optical networks magazine, vol. 1, no. 1, Jan. 2000, pp. 47-60.

[4]. S. Xu, L. Li and S. Wang, "Dynamic routing and assignment of wavelength algorithms in multifiber wavelength division multiplexing networks", IEEE Journal on selected areas in communications, vol. 18, no. 10, Oct. 2000, pp. 2130-2137.

[5]. Y. Zhang, K. Taira, H. Takagi, and S. K. Das, "An efficient heuristic for routing and wavelength assignment in optical WDM networks", IEEE International Conference on Communications, ICC 2002 (Copenhagen, Denmark, July 2002), vol. 5, pp. 2734-2739.

[6]. J. Kim and D. C. Lee, "Dynamic routing and wavelength assignment algorithms for multifiber WDM networks with many wavelengths", 2nd European Conference on Universal Multiservice Networks, EDUMN 2002 (CREF, Colmar, France, April 2002), pp. 180-186.

[7]. X. Zhang and C. Qiao, “Wavelength assignment for dynamic traffic in multi-fiber WDM networks”, Computer Proceedings. 7th International Conference on Computer Communications and Networks (Lafayette, LA, October 1998), pp. 479-485.

[8]. C. Chen and S.Banerjee, “A new model for optimal routing and wavelength assignment in wavelength division multiplexed optical networks”, Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation, Proceedings IEEE, INFOCOM '96 (San Francisco, CA, March 1996), vol. 1, pp. 164-171.

[9]. X. Yang and B. Ramamurthy, “Dynamic routing in translucent WDM optical networks”, IEEE International Conference on Communications, ICC 2002 (Copenhagen, Denmark, July 2002), vol. 5, pp. 2796-2802.

[10]. J. Zhang and H.T. Mouftah, “Routing and wavelength assignment for advance reservation in wavelength-routed WDM optical networks”, IEEE International Conference on Communications, ICC 2002 (Copenhagen, Denmark, July 2002), vol. 5, pp. 2722-2726.

[11]. M. Knoke and H. L. Hartmann, “Fast optimum routing and wavelength assignment for WDM ring transport networks”, IEEE International Conference on Communications, ICC 2002 (Copenhagen, Denmark, July 2002), vol. 5, pp. 2740-2744.

[12]. A. Detti, V. Eramo and M. Listanti, “Performance evaluation of a new technique for IP support in a WDM optical network: optical composite burst switching (OCBS)”, IEEE/OSA Journal of Lightwave Technology, vol. 20, no. 2, Feb. 2002, pp. 154–165.

[13]. D. Guo and A. S. Acampora, “Scalable multihop WDM passive ring with optimal wavelength assignment and adaptive wavelength routing”, IEEE/OSA Journal of Lightwave Technology, vol. 14, no. 6, June 1996, pp. 1264–1277.

[14]. Uyless Black, “Optical networks: third generation transport systems”, ISBN 0130607266, published by Prentice Hall, 2002.

[15]. Y. Suemura, I. Nishioka, Y. Maeno,S. Araki, R. Izmailov and S. Ganguly, “Hierarchical routing in layered ring and mesh optical networks”, IEEE International Conference on Communications, ICC 2002(Copenhagen, Denmark, July 2002), vol. 5, pp. 2727-2733.

[16]. S. De Maesschalck, D. Colle, A. Groebbens, C. Develder, A. Lievens, P. Lagasse, M. Pickavet, P. Demeester, F. Saluta and M. Quagliatti, “Intelligent optical networking for multilayer survivability”, IEEE Communications Magazine, vol. 40, no. 1, Jan. 2002, pp. 42–49.

[17]. A.S. Arora, S. Subramaniam and H. Choi, “Logical topology design for linear and ring optical networks”, IEEE Journal on Selected Areas in Communications, vol. 20, no. 1, Jan. 2002, pp. 62–74.

[18]. T.K. Nayak and K.N. Sivarajan, “A new approach to dimensioning optical networks”, IEEE Journal on Selected Areas in Communications, vol. 20, no. 1, Jan. 2002, pp. 134–148.

[19]. Z. Ding and M. Hamdi, “A simple routing and wavelength assignment algorithm using the blocking island technique for all-optical networks”, IEEE International Conference on Communications, 2002. ICC 2002 (Copenhagen, Denmark, July 2002), vol. 5, pp. 2907–2911.

[20]. A. Mokhtar and M. Azizoglu, “Adaptive techniques for routing and wavelength assignment in all-optical WANs”, Circuits and Systems, IEEE 39th Midwest symposium, 1996, vol. 3, pp. 1195-1198.

[21]. I. Chlamtac, A. Ganz, and G. Karmi, “Lightpath communications: an approach to high-bandwidth optical WAN’s”, IEEE transactions on communications, vol. 40, no. 7, July 1992, pp. 1171-1182.

[22]. P. Manohar, D. Manjunath and R.K. Shevgaonkar, “Routing and wavelength assignment in optical networks from edge disjoint path algorithms”, IEEE Communication letters, vol. 6, no. 5, May 2002, pp. 211-213.

[23]. A. Birman and A. Kershenbaum, “Routing and wavelength assignment methods in single-hop all-optical networks with blocking”, Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies. 'Bringing Information to People', Proceedings IEEE, INFOCOM '95 (Boston, MA, April 1995), vol. 2, pp. 431 -438.

[24]. D. Banerjee and B. Mukherjee, “A practical approach for routing and wavelength assignment in large wavelength-routed optical networks”, IEEE Journal on selected areas in communications, vol. 14, no. 5, June 1996, pp. 903-908.

[25]. M. El Houmaidi and M. Bassiouni, “k-Weighted Minimum Dominating Sets for Sparse Wavelength Converters Placement under Non-uniform Traffic”, Proceedings of the 11th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS) (Orlando, FL, October 2003), pp. 56-61.

[26]. M. El Houmaidi, M. Bassiouni and G. Li, “Dominating set algorithms for sparse placement of full and limited wavelength converters in WDM optical networks”, Journal of Optical Networking, Optical Society of America, vol. 2, no. 6, June 2003, pp. 162-177.

[27]. M. El Houmaidi, M. Bassiouni and G. Li, “Architecture and Sparse Placement of Limited Wavelength Converters for Optical Networks”, Journal of Optical Engineering,

International Society for Optical Engineering- SPIE Publishing, vol. 43, no. 1, January 2004, pp. 137-147.

[28]. L. Li and A. K. Somani, "Dynamic wavelength routing using congestion and neighborhood information", IEEE/ACM Trans. Networking, vol. 7, no. 5, Oct. 1999, pp. 779–786.

[29]. H. Harai, M. Murata and H. Miyahara, "Performance of alternate routing methods in all-optical switching networks," in Proc. IEEE INFOCOM'97 (Kobe, Japan, April 1997), pp. 516–524.

[30]. C. Qiao and M Yoo, "Optical burst switching (OBS) – a new paradigm for an optical internet", Journal of high speed networks 8 (1999), IOS Press, pp. 69-84.

[31]. L. Xu, H. G. Perros, and G. Rouskas, "Techniques for optical packet switching and optical burst switching", IEEE communication magazine, Jan. 2001, pp. 136-142.

[32]. J. Y. Wei, and R. I. McFarland, "Just-in-time signaling for WDM optical burst switching networks", Journal of lightwave technology, Vol. 18, No. 12, Dec. 2000, pp. 2019-2037.

[33]. J. S. Surner, "Terabit burst switching", Journal of High Speed Networks, Vol. 8, No. 1, January 1999, pp. 3-16.

[34]. S. Amstutz, "Burst switching--An introduction", Communications Magazine, IEEE, Volume: 21 Issue: 8, Nov 1983, pp. 36 –42.

[35]. C. Loi and W. Liao, "Multiclass wavelength reservation in optical burst switched WDM networks", Communications, Circuits and Systems and West Sino Expositions, IEEE 2002 International Conference, Vol. 1, 2002, pp. 845 –849.

[36]. M. Yoo and C. Qiao, "Just-enough-time (JET): a high speed protocol for bursty traffic in optical networks", IEEE/LEOS Conf. on Technologies For a Global Information Infrastructure, Aug. 1997, pp. 26-27.

[37]. C. Qiao, "Labeled optical burst switching for IP-over-WDM Integration", IEEE Communications Magazine, vol. 38, no. 9, 2000, pp.104-114.

[38]. J. Chang, and C. Park, "Efficient channel-scheduling algorithm in optical burst switching architecture", In Proceedings of Workshop on High Performance Switching and Routing, 2002, pp. 194-198.

[39]. C. Hsu, T. Liu, and N. Huang, "Performance analysis of deflection routing in optical burst-switched networks", INFOCOM 2002, pp. 66 –73.

[40]. M.C. Yuang, J. Shih, and P.L. Tien, "QoS Burstification for Optical Burst Switched WDM Networks", In Proceedings of OFC, 2002, pp. 781-783.

[41]. Y. Xiong, M. Vandenhoute, and H.C. Cankaya, "Control architecture in optical burst switched WDM networks", IEEE JSAC, vol.18, 2000, pp.1838-1851.

[42]. C. Xin, and C. Qiao, "A comparative study of OBS and OFS", Proceedings, OFC 2001, Vol. 4, March 2001, pp. ThG7-1 -ThG7-3.

[43]. C. M. Gauger, "Contention resolution in optical burst switching networks", Advanced Infrastructures for Photonic Networks: WG 2 Intermediate Report, 2002, pp. 62-82.

[44]. S. Thiagarajan and A. K. Somani, "Capacity Fairness of WDM Networks with Grooming Capabilities", OPTICOMM 2000, Dallas, SPIE Proc. Vol. 4233, October 2000, pp. 191-201.

[45]. X. Wang, H. Morikawa, and T. Aoyama, "Burst optical deflection routing protocol for wavelength routing WDM networks", Proceedings of OPTICOMM 2000, pp. 257-266.

[46]. R. Srinivasan and A. K. Somani, “Request-specific routing in WDM grooming networks”, in Proceedings of IEEE International Conference on Communications (ICC 2002), Vol. 5, April-May 2002, pp. 2876-2880.

[47]. M. Labrador and S. Banerjee, “Performance of Selective Packet Dropping Schemes in Multi-hop Networks”, Proc. of IEEE GLOBECOM, 1999, pp. 1604-1609.

[48]. I. Ogushi, S. Arakawa, M. Murata and K. Kitayama, “Parallel reservation protocols for achieving fairness in optical burst switching”, in Proceedings of IEEE Workshop on High Performance Switching and Routing, May 2001, pp. 213-217.

[49]. J. A. White, R. S. Tucker, and K. Long, “Merit-based scheduling algorithm for optical burst switching”, in Proceedings of the International Conference on Optical Internet. Seoul, Korea: Korean Institute of Communication Sciences, July 2002, pp. 75-77.

[50]. M. Chiu and M. Bassiouni, “Predictive Schemes for Handoff Prioritization in Cellular Networks Based on Mobile Positioning” IEEE Journal on Selected Areas in Communications, Vol. 18, No. 3, March 2000, pp. 510-522.

[51]. S. Floyd, and V. Jacobson, “Random Early Detection gateways for Congestion Avoidance”, IEEE/ACM Transactions on Networking, V.1 N.4, August 1993, p. 397-413.

[52]. Woo Chool Park, Sang Jun Park, and Byung Ho Rhe, “RED-priority cell discarding method for improving TCP performance in ATM networks”, Joint 4th IEEE International Conference on ATM (ICATM 2001) and High Speed Intelligent Internet Symposium, 2001. , 22-25 April, 2001, pp. 61 –65.

[53]. M. May, J. Bolot, C. Diot, and B. Lyles, “Reasons not to deploy RED”, INRIA, technical report, June 1999, pp.260-262.

[54]. M. Christiansen, K. Jeffay, D. Ott, and F.D. Smith, "Tuning RED for Web Traffic", ACM SIGCOMM, August 2000, pp.139-150.

[55]. W. Feng, D. Kandlur, D. Saha, and K. Shin, "A Self-Configuring RED Gateway", Proceedings of INFOCOM, March 1999, pp. 1320-1328.

[56]. C. Hollot, V. Misra, D. Towsley and W. Gong. "A Control Theoretic Analysis of RED", Proceedings of INFOCOM, 2001, pp.1510-1519.

[57]. H. El-Aarag and M. Bassiouni, "Performance Evaluation of TCP Connections in Ideal and Non-Ideal Network Environments" Journal of Computer Communications, Elsevier Publishing, Vol. 24, No. 18, December 2001, pp. 1769-1779.

[58]. M. Yoo, C. Qiao and S. Dixit, "QoS performance of optical burst switching in IP-over-WDM networks", IEEE Journal of Selected Areas in Communications, Vol. 18, Issue: 10, Oct., pp.2062 – 2071 (2000).

[59]. C. Gauger, K. Dolzer, J. Späth, and S. Bodamer, "Service differentiation in optical burst switching networks", Beiträge zur 2. ITG Fachtagung Photonische Netze, Dresden, March 2001, pp. 124-132

[60]. V. Vokkarane and J. Jue, "Prioritized Routing and Burst Segmentation for QoS in Optical Burst-Switched Networks", Proceedings, Optical Fiber Communication Conference (OFC) 2002, Anaheim, CA, March 2002.

[61]. Y. Chen, M. Hamdi, and D.H.K. Tsang, "Proportional QoS over OBS networks", Proceedings, IEEE GLOBECOM, vol. 3, pp. 1510 –1514 (2001).

[62]. D.Q. Liu and M.T. Liu, "Priority-Based Burst Scheduling Scheme and Modeling in Optical Burst-switched WDM Networks", Proceedings of International Conference on Telecommunications (ICT 2002), June, 2002.

[63]. F. Poppe, K. Laevens, H. Michiel, and S. Molenaar, "Quality-of-service differentiation and fairness in optical burst-switched networks", Proceedings, Optical Networking and Communication Conference (OptiComm) 2002, Boston, MA, July-Aug 2002.

[64]. K. Dolzer, "Assured Horizon - An Efficient Framework for Service Differentiation in Optical Burst Switched Networks", Proceedings, Optical Networking and Communications Conference (OptiComm 2002), Boston, MA, July-Aug 2002.

[65]. W.H. So and Y.C. Kim, "Offset Time Decision for Supporting Service Differentiation in Optical Burst Switching Networks", Proceedings of COIN-PS 2002, Cheju Island, Korea, July, 2002.

[66]. J. Liu and N. Ansari, "Forward Resource Reservation for QoS Provisioning in OBS Systems", Proceedings, IEEE Globecom 2002, Taipei, Taiwan, November 2002.

[67]. C.H. Loi, W. Liao, and D.N. Yang, "Service Differentiation in Optical Burst Switched Networks", Proceedings, IEEE Globecom 2002, Taipei, Taiwan, November 2002, pp.2313-2317.

[68]. D.Q. Liu and M.T. Liu, "Differentiated services and scheduling scheme in optical burst-switched WDM networks", Proceedings, IEEE International Conference on Networks (ICON), 2002.

[69]. V. Vokkarane and J. Jue, "Prioritized Burst Segmentation and Composite Burst Assembly Techniques for QoS Support in Optical Burst-Switched Networks", IEEE Journal of Selected Areas in Communications, Volume: 21, Issue:7, Sept. pp.1198 – 1209 (2003).

[70]. M.Yang, S.Q.Zheng, and D. Verchere, “A QoS supporting scheduling algorithm for optical burst switching DWDM networks”, GLOBECOM '01. IEEE , Vol. 1 , Nov.,pp.86 – 91(2001).

[71]. J. Xu, C. Qiao, J. Li and G. Xu, “Efficient channel scheduling algorithms in optical burst switched networks”, IEEE INFOCOM 2003, San Franciso, CA, USA (2003), pp. 2268 – 2278.

[72]. Y. Chen, C. Qiao, and X. Yu, “Optical burst switching: A new area in optical networking research”, IEEE Network May/June 2004, pp. 16-23.

[73]. L. Yang, Y. Jiang and S. Jiang, “A probabilistic preemptive scheme for providing service differentiation in OBS networks”, IEEE GLOBECOM 2003, pp. 2689-2693.

[74]. H. C. Cankaya, S. Charcranoon and T. S. El-Bawab, “A preemptive scheduling technique for OBS networks with service differentiation”, IEEE GLOBECOM 2003, pp. 2704-2708.

[75]. M. Yoo and C. Qiao, “A new optical burst switching protocol for supporting quality of service”, In SPIE Proc. of Conf. All-optical Networking, 1998, vol. 3531, pages 396-405.

[76]. X. Wang, H. Morikawa and T. Aoyama, “Priority-based wavelength assignment algorithm for burst switched photonic networks”, OFC 2002, pp.765-767.

[77]. W. Peng and C. Wei, “Distributed wavelength assignment protocols with priority for WDM all-optical networks”, Computer Communications and Networks, 2000. Proceedings, pp.625 – 630.

[78]. T. Tachibana and S. Kasahara, “QoS-Guaranteed Wavelength Allocation for WDM Networks with Limited-Range Wavelength Conversion”, IEICE Transactions on Communiations, vol. E87-B, no. 6, June 2004, pp. 1439-1450.

[79]. H.L. Vu, and M. Zukerman, "Blocking probability for priority classes in optical burst switching networks", IEEE communications letters, Vol.6, No. 5, May 2002, pp.214-216.

[80]. V. M. Vokkarane, J. P. Jue and S. Sitaraman, "Burst segmentation: an approach for reducing packet loss in optical burst switched networks", IEEE International Conference on Communications, 28 April-2 May , 2002, pp. 2673-2677.

[81]. A. GE, F. Callegati and L.S. Tamil, "On optical burst switching and self-similar traffic", IEEE Commun. Letter, Vol.4, Mar. 2000, pp. 98-100.

[82]. V.M.Vokkarane, K.Haridoss and J.P.Jue, "Threshold-based burst assembly policies for QoS support in optical-switched networks", in proc. SPIE OptiComm 2002, vol. 4874, Boston, MA, July 2002, pp. 125-136.

[83]. M.Neuts, Z. Rosberg, H.L.Vu, J.White and M. Zukerman, "Performance enhancement of optical burst switching using burst segmentation", in IEEE International Conference on communications, ICC'03, vol.3. May 2003, pp. 1828-1832.

[84]. Z. Rosberg, H.L.Vu and M. Zukerman, "Burst segmentation benefit in optical switching", IEEE Communications Letters, vol. 7, Issue 3, March 2003, pp. 127-129.

[85]. C.W.Tan, M.Gurusamy and J.C.S.Lui, "Achieving proportional loss differentiation using probabilistic preemptive burst segmentation in optical burst switching WDM networks", IEEE GLOBECOM '04, vol. 3, 29 Nov.-3 Dec. 2004, pp. 1754 – 1758.

[86]. T.K. Moseng, H.Qverby and N.Stol, "Merit based scheduling in asynchronous bufferless optical packet switched networks", In Proceedings of Norsk Informatikk Konferanse (NIK), November 2004, Stavanger, Norway, pp. 126-136.

[87]. M. El Houmaidi and M. Bassiouni, “Dependency Based Analytical Model for Computing Connection Blocking Rates and its Application in the Sparse Placement of Optical Converters”, IEEE Transactions on Communications, Vol. 54, No. 1, 2006, pp. 159-168.

[88]. B. Suter, TV Lakshman, D. Stiliadis and A. Choudhury, “Design considerations for supporting TCP with per-flow queuing”, Proc. of INFOCOMM '98, Apr. 1998, pp.299-306.