Threshold Schemes for Optical Flow Switching

by

Dmitry Brant

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degrees of
Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

May 17, 1999

[June 1999]

Author____
Department of Electrical Engineering and Computer Science
May 17, 1999

Certified by____________________________________
Eytan Modiano
Thesis Supervisor

Accepted by____________________________________
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

Threshold Schemes for Optical Flow Switching

by

Dmitry Brant

Submitted to the Department of Electrical Engineering and Computer Science
May, 1998

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

IP routers are difficult to scale to very high rates due to the significant amount of electronic processing required for each IP packet. In WDM networks, an alternative to IP routing is Optical Flow switching. The idea behind the Optical Flow switching is to set up WDM cut-trough transmissions for large messages. Unfortunately, current WDM systems only have a limited number of channels that can be used for cut-through transmissions. This work investigates the performance of several schemes that calculate a threshold for determining whether a message is sent over IP or over a WDM cut-through. We discuss the performance of static schemes which do not vary the threshold and dynamic schemes which vary threshold in response to changing network conditions.

Thesis Supervisor: Dr. Eytan Modiano
Title: Technical Staff, MIT Lincoln Laboratory

Acknowledgments

I would like to thank my supervisor, Dr. Eytan Modiano, who was a great source of guidance and help over the past year. I would also like to thank my family and my grandma for their support and caring. In addition, I would like to thank Marina Zhurakhinskaya and Natasha Skorodinsky for providing me with their great company and making the long hours spent at Athena more enjoyable. I also thank Aaron Maldonado for proofreading my thesis and Pietro Russo for being the entertaining personality he is.

Table of Contents.

List of Figures

1.Introduction.

Over the past decade the number of Internet users has grown dramatically, with each user adding more demand to the network. Over time, each user's traffic tends to increase, adding further to the demand. This growth in traffic demand is expected to continue for years to come. In response to this expected demand, network equipment vendors and service providers have developed and deployed larger and faster routers. However, the routing speed of existing IP routers is not keeping up with the quickly growing transmission rates within networks, particularly with the rates of Wavelength Division Multiplexing(WDM)-based networks that have rates of hundreds of Gbps. IP routers are difficult to scale to high rates due to the significant amount of electronic processing required for each IP packet. Several schemes such as IP switching [1], Tag Switching [2], ARIS [3] and CSR [4] have been proposed to improve the performance of IP routers.

WDM increases the capacity of embedded fibers by first assigning incoming optical signals to specific frequencies within a designated frequency band and then multiplexing the resulting signals onto one fiber. Because incoming signals are never terminated in the optical layer, the interface can be bit-rate and format independent, allowing the service provider to integrate the WDM technology easily with existing equipment in the network while gaining access to the untapped capacity in the embedded fiber. WDM combines multiple optical signals so that they can be

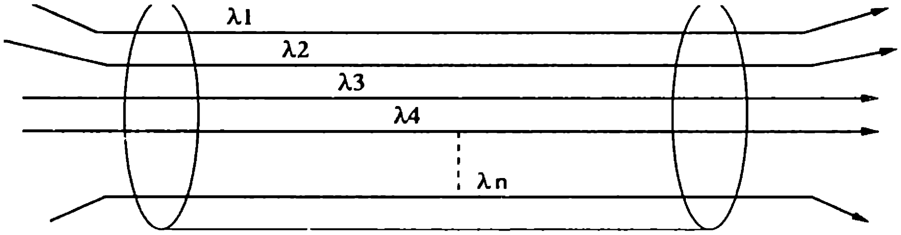


Figure 1. WDM channels in an optical fiber

amplified as a group and transported over a single fiber to increase capacity (see Figure 1). Each carried signal can be at a different rate (OC-3/12/24, etc.) and in a different format (SONET, ATM, data, etc.). For example, a WDM network with a mix of SONET signals operating at OC-48 (2.5 Gbps) and OC-192 (10 Gbps) over a WDM infrastructure can achieve capacities of over 40 Gbps. A system with WDM can achieve all this gracefully while maintaining the same degree of system performance, reliability, and robustness as current transport systems, and in some instances even surpassing it [5].

Delivery of IP services can be greatly enhanced by harnessing the huge capacity and configurability of WDM. The optical layer can provide many additional services such as reconfiguration, multilayer switching and restoration. However, to take full advantage of the underlying optical layer the network protocol stack needs to be simplified. Right now, the multitude of layers in the protocol stack produces bandwidth inefficiencies, adds to the latencies of connections and inhibits QoS assurances.The layers are also unaware of each other, which causes duplication of network services or operation on different and sometimes conflicting virtual topologies. Instead, we can use a simplified stack (Figure 2) with a WDM-aware electronic layer. Such an arrangement will greatly enhance the possible synergy that can be obtained from interactions between IP and the underlying configurable WDM pipes [6].

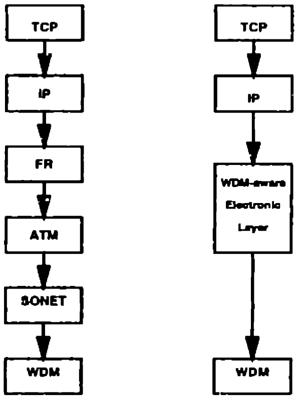


Figure 2.
a) A typical internet connection with a multitude of layers between IP and the WDM
b) Protocol stack with the WDM-aware electronic layer.

In particular, the WDM-aware optical layer is naturally suited for flow switching. Flow switching can be used to alleviate the bottleneck associated with routing at the IP layer. Flow switching was originally conceived in the context of ATM networks as IP switching which was proposed by Ipsilon[1]. The idea of IP switching is to set up ATM virtual circuits for those connections that are perceived to be of long duration [1]. A flow is defined as a sequence of packets having some common properties such as the same destination or port number. The focus of this project is to investigate the incorporation of flow switching in WDM networks. In such networks. IP routers will be connected using separate wavelengths. Instead of allocating an ATM Virtual Circuit (VC) to the flow, the WDM-aware layer will allocate a complete wavelength to the flow of packets (Figure 3). Flow switching can yield higher throughput and lower latencies by avoiding the IP routing bottleneck. The main question addressed in this paper is: how does the network decide when to optically switch a flow and when to switch the flow electronically?

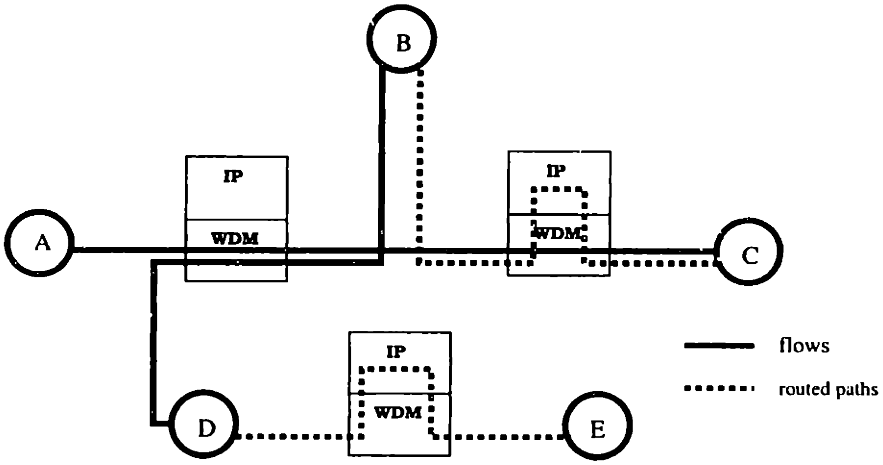


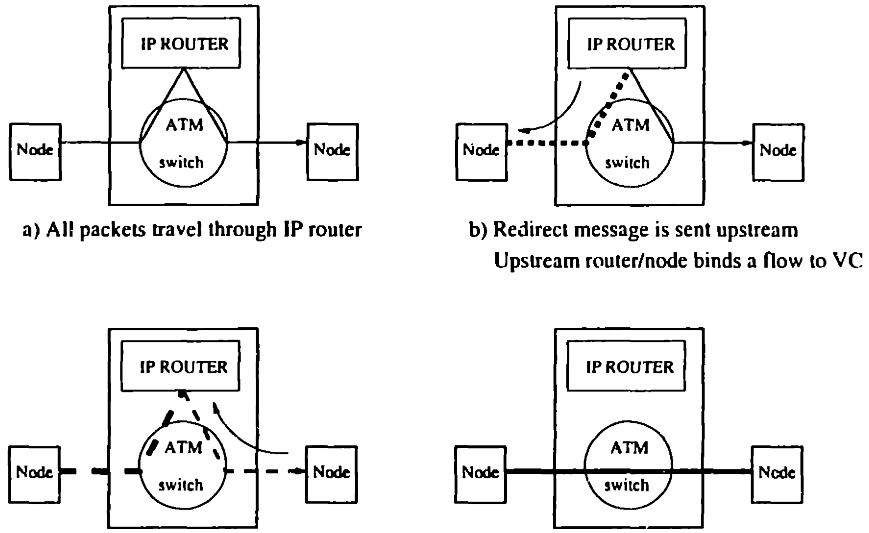
Figure 3. WDM network with flows between several nodes

2.Background Information

2.1 Introduction to IP switching.

IP switching integrates IP routing with the fast lookup and copying mechanisms of ATM switching. Consider the IP switch shown in Figure 3. Assume that the switch is surrounded by a number of identical neighbors. The switch begins by establishing adjacency with its neighbors, identifying them as fellow IP switches. This is done using a special Ipsilon Flow Management Protocol (IFMP). IP datagrams arrive at the switch from various sources. These datagrams are part of a flow between two user processes, or between two end points in the network. Initially, all datagrams from all flows are received and transmitted on a single, default ATM VC. The flow classifier in the switch inspects the contents of the fields that characterize the flow and makes its decision based upon a local policy. For example, a flow classifier might look for well-known source or destination port numbers to identify the application. Flows belonging to FTP data connections might be configured to be switched at the ATM layer, but DNS queries could be routed through IP. Another example of a flow classifier might count the number of packets received on each flow. If the number of packets received within a specified time interval exceeds a threshold the flow is switched at the ATM layer[7].

If the switch decides to switch the flow at the ATM layer, it will attempt to establish a unique VC for the flow, so that it may be switched quickly in hardware [1]. The switch tells its upstream neighbor to begin sending all datagrams that are in that flow on a separate VC (Figure 4 b). This is achieved by sending a redirect message to the next switch that is upstream in the flow. Similarly, the switch will receive a redirect message from its downstream neighbor when the downstream neighbor detects a flow (Figure 4c). When the flow has been successfully bound, the switch can perform hardware, layer-2 switching for the incoming VC to the outgoing VC, without consulting any routing tables or performing any other processing (Figure 4d).



a) All packets travel through IP router

b) Redirect message is sent upstream
Upstream router/node binds a flow to VC

c) Redirect message is recieved from downstream

d) Cut-through established

Figure 4. IP switching

2.2 *Optical Flow switching*

The idea of Optical Flow Switching is similar to the idea of IP/ATM switching. Instead of the ATM switch, a Frequency Selective Switch (FSS) is used. FSS can take any wavelength on any of the incoming ports and switch it to any of the output ports (Figure 5) Other, more limited versions of a WDM switches also exist, such as Optical Broadcast Star, where each port can transmit only on the frequencies not occupied by other ports.

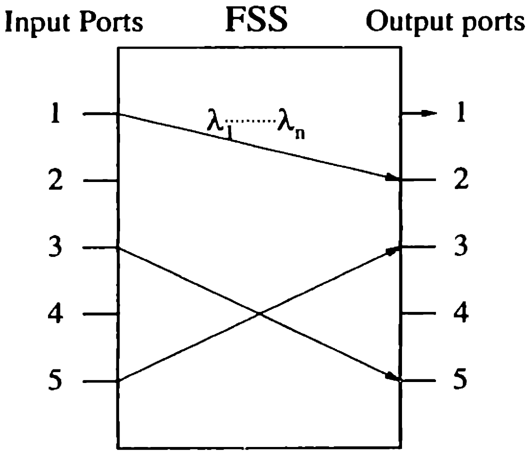


Figure 5. Frequency Selective Switch (FSS)

Located on top of the WDM switch is a routing kernel and a switch controller which makes the decisions about switching the flows. Packets arrive on WDM channels and get processed through the WDM-aware electronic layer and then get passed to the IP router. At that point, if the switch controller detects a flow, it can decide to either switch or route it. The decision can be made either based on the duration of the flow or the Quality of Service requirements. Some flows might require very low delay even though their duration may not be very long. These flows should have higher priorities than longer, but less latency-dependent flows and are more likely to be switched than routed.

However, in certain respects, IP/WDM switching is fundamentally different from IP switching over ATM. For example, the IP/ATM switch is almost unlimited in the number of flows it can support at a time, since only a single VC is necessary for a flow to be established, and ATM switches support thousands of VC's simultaneously. On the other hand, the number of simultaneous flows in an Optical switch system is limited by the physical number of channels supported by the system. Some WDM systems can support as many as 80 different wavelengths, a number that is still insignificant compared to the thousands of VC's supported by ATM. Therefore, if the optical switch controller binds flows to WDM channels unwisely, it will run out of channels very quickly. Also, the price for binding the entire channel to a short flow is very high because of the long setup times for optical channels. Spending 3 milliseconds to set up an optical path for a microsecond flow is very inefficient. The IP/ATM switch's flow classification mechanism classifies flows based on their expected duration. Since the switch does not know the size of the incoming flow beforehand, it has to guess the duration of the flow. In ATM systems, the set-up time for a VC is very short and the price for giving the short flow a VC is minimal, which is not true for the Optical system. Therefore, flow identification algorithms used in IP/ATM switches will not work well with IP/WDM switches.

3.Models

3.1 Flow classification methods.

The design of a better classification algorithm for Optical flow switching is the focus of this project. The new algorithm must take into account the number of available wavelengths, capacity of the IP router and other parameters, such as the setup times for the switch. There exist two different approaches that can be used to do flow classification. In the simpler method, the application is responsible for letting the switch know the complete size of a message. With this information the switch can make a decision on whether to route the message over IP or send it in a single burst over some WDM channel. However, several problems exist with this method. First, we have to trust the application, since it is fair to assume that some applications might be inclined to lie about the message size in order to get a better bandwidth allocation. Also, in many cases, the application doesn't know the size of a message beforehand, particularly for the messages which are generated automatically, such as with Common Gateway Interface (CGI) scripts. It is inefficient for the application to buffer the whole message to get its complete size, since part of the message can be transmitted while the rest of it is being generated. In addition, the application will need to announce the message size to all the switches on the path in order to get a complete end-to-end cut-through path. This will require a lot of synchronization between switches and could be difficult to implement.

The second approach is similar to the one used by IP/ATM switches. The switch tries to classify flows dynamically. If the switch sees a certain number of packets with the same characteristics within a certain amount of time, it will switch the flow. However, with constantly changing traffic patterns, the switch does not have enough information to make a good estimate of the flows future duration[8]. A flow which lasted for several seconds can stop in the next microsecond. With a total of only several dozen channels, each at 10 Gbt/s, a bad decision can be very costly in terms of wasted bandwidth. Perhaps, in the future, when WDM systems will support higher number of channels, this approach could be used. Therefore, for the rest of this paper we focus on the first approach; that is, we assume that the size of a flow is known in advance and devise algorithms to determine if a flow should be switched optically or electronically. Our goal is to devise an algorithm that will minimize message delay in the network.

3.2. Model Architecture.

The main question of this study is: Given a network shown in Figure 3, how do we decide when to switch a flow? The answer depends on the number of available WDM channels, IP ports, delays in the network, set-up times and other factors.We came up with a conceptual model of an IP/WDM switch that would be simple, yet would capture all the essential properties of the actual physical node. These properties are: number of available wavelengths, switch capacity of the IP switch and different delays depending on whether the message gets routed or switched. We came up with a simple single switch model shown in Figure 6.

This model uses a broadcast optical star to model a single WDM Frequency Selective Switch (FSS). The broadcast optical star can take an input wavelength on any of its input ports and transmit it on all output ports. If more than one node transmits on the same wavelength, it will cause collisions on this wavelength on the output ports. We can represent a FSS as a broadcast star if we assume that there is no wavelength changing done within the FSS, so that the message incoming to the input port of FSS on some wavelength λ will leave the FSS on the same wavelength λ.

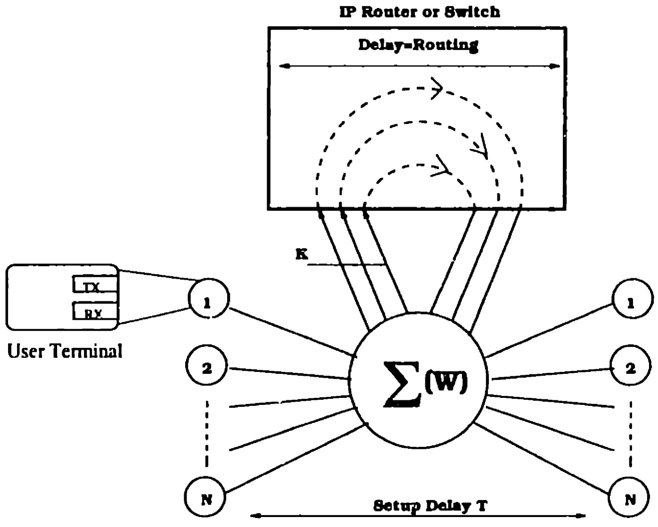


Figure 6. IP/WDM switch

If one of the wavelengths is taken by some port, no other input ports will be able to transmit on that wavelength. As shown in Figure 6, N user terminals are connected to each other by the optical star. Each user terminal has a separate TX/RX port for WDM and IP. It can therefore transmit or receive on both the WDM and the IP channels simultaneously. We also assume that the star can connect at most W nodes at a time, where W is the maximum number of WDM channels in the system. In addition, the optical star has K connections to the IP router, where K is the number of ports on the router. The routing capacity of the IP router is limited by the number of ports it has, since at any given time, it can only receive at most K messages (one message for each port). If there are more transmitting nodes than IP ports, there will be contention between the nodes for the ports on the IP router. Therefore, K captures the effect of port blocking on the IP router.

Each of the N nodes generates messages which represent flows. If messages go through the IP router, they get broken into IP packets, otherwise they get switched directly through the optical star. The delays of each path will obviously be different, depending on the router's load. The basic trade-off between sending a message over WDM vs. IP is the set-up delay required for establishing a cut-through optical path. For the large messages, this set-up delay will be small compared to their transmission delay, but the routing delay will be very high due to the large number of IP packets that will have to be sent out, causing congestion in the IP router. On the other hand, for small messages, this set-up delay will be much larger then their routing delay so it will make sense to send those packets through the IP router. This reasoning lead us to believe that a threshold scheme would work best for the switch controller.

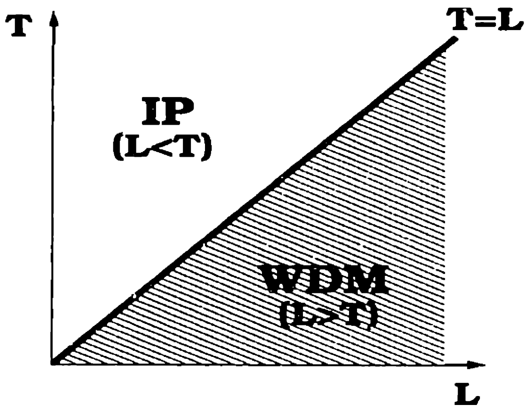


Figure 7. L vs. T

The threshold scheme is simple and it is based on the length (L) of the incoming message and the size of the threshold T. The rules are outlined below:

if **L** >**T**, switch the message optically over WDM
if **L**< **T**, switch message electronically over IP

Figure 7 shows a graph of a 45 degree line L=T. All the points below the line represent the region where the decision would be "WDM", and all the points above L=T represent the region of the "IP" decision.

3.3. Simulation Architecture.

We simulated the proposed model using OPNET. The overall network topology is shown in Figure 8. Two multi-channel buses were connected by the IP/WDM switch. Each bus supports W distinct channels, simulating the optical broadcast star. All transmitters are placed on the left side of the IP/WDM switch and the receivers are placed on the right side. If the node has a message to transmit, it sends a request packet to the IP/WDM switch telling it the size of the message. The switch then makes a decision and sends it back to the transmitter, which then sends out a message either over WDM or IP.

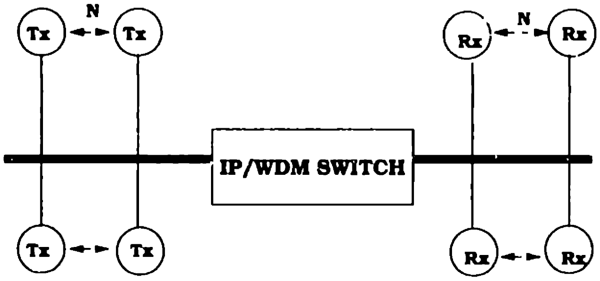


Figure 8. Simulation network topology

The general structure of the IP/WDM switch is shown on Figure 9. It consists of the routing component, controller and WDM component. We modeled the routing component as an input queue switch with CxC switching fabric and constant delay for the routing table lookup and header processing. Although this is a very simplified model of a router, it captures the queuing delay of the packets in the system. The delay associated with header manipulation and table lookup can be considered to be constant.

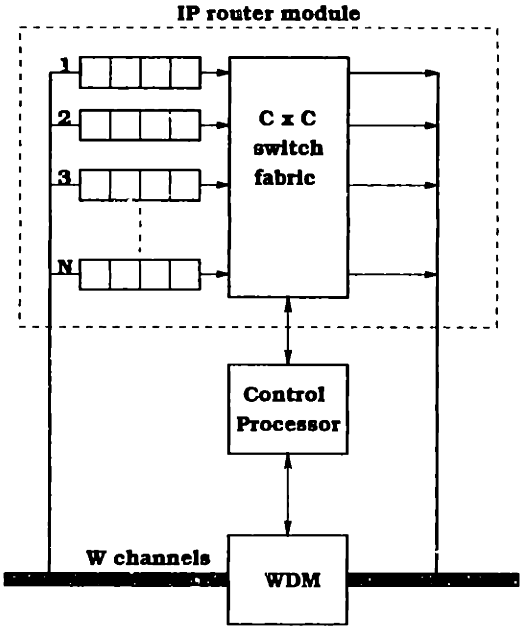


Figure 9. *Simulated IP/WDM switch*

In the type of queueing architecture displayed in Figure 9, a separate buffer is placed on each input port of the switch. The input buffers may operate in a First In First Out (FIFO) or First In Random Out (FIRO) fashion. If no scheduling algorithm is used to select which cells to transmit at the beginning of the routing cycle, head-of-line blocking may occur with more frequency.

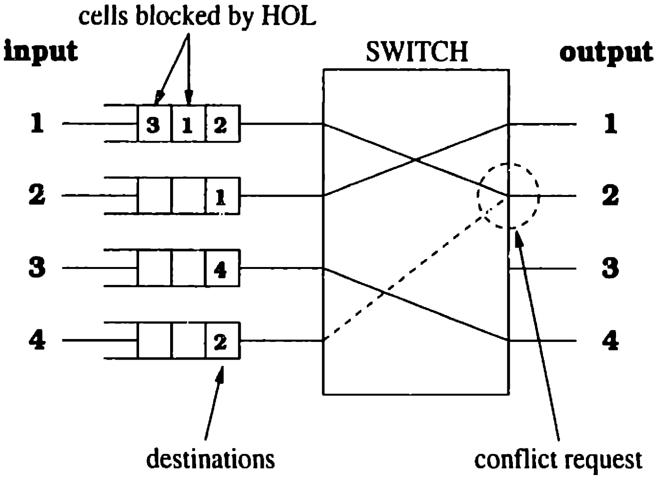


Figure 10. HOL blocking in 4x4 switch

When 'k' cells at the head of their queues compete for the same output, only one is allowed to pass through, and 'k-1' cells must wait for the next routing cycle. In the meantime, while one of the 'k-1' cells waits for its turn, other cells are queued in the FIFO, and blocked from reaching possible idle output ports in the switch. Figure 10 shows head-of-line blocking for a 4 by 4 switch.

A theoretical performance analysis of a strictly input queued space-division packet switch, without arbitration, was conducted in [9]. For independent and identical Bernoulli traffic sources, a large number of input ports, and incoming packets uniformly distributed among all outputs, it was determined from the analysis that the maximum saturated throughput is approximately 58.6%.

The control processor makes the decision on whether the message is to be transmitted over the WDM channel or over IP. If the message size is larger than a certain threshold and a processor can allocate a channel to the message, then it is sent over the WDM. If no channels are available, two possibilities exist: the controller can either tell the transmitter to send the message over IP or it can tell the transmitter to wait with the transmission until a wavelength is available. Two reasons exist for why the channel can't be allocated. Either, all of the WDM channels are already used or the receiver on the receiving node is already used.

4.Static Threshold Schemes

4.1 Algorithm.

Our first scheme is a simple static threshold scheme that was described in section 3. The IP/WDM switch uses the following static decision policy: the size of the message (**L**) is compared to the threshold **T**. **L** has some probabilistic distribution which we normally do not know in advance. In this chapter, however, we assume that the distribution of **L** is known. By knowing the distribution of **L**, we can choose a single optimal threshold **T** that will minimize the delay of the messages in the system. Thus our goal is to determine the optimal threshold for which the delay of the messages in the network is minimal. **T** is chosen only once and it does not change in response to the state of the network, so in this sense, the threshold scheme is static.

The relationship between L and T can be best observed from Figure 7. It follows that

if **L** >**T**, the message is switched optically over WDM

if **L**< **T**, the message is routed electronically over IP

If the message cannot be switched optically right away because there are no more WDM channels left, or because of the receiver contention, then the message is put into the tail of FIFO queue where it waits until it can be serviced.

4.2 Analytical Model (M/G/1 approximation)

We devised an analytical model of the system in order to approximate the delay that messages will encounter through the system. We will later use this approximation to obtain the optimal threshold values which will minimize the message delay in the system.

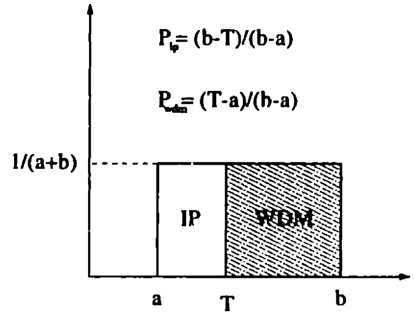


Figure 11. Distribution of the message size

In our definition, message duration is the length of the message divided by the transmission rate.

Assume that the message duration is uniformly distributed between *a* and *b* seconds (Figure 11). All messages with duration which falls in the shaded area are sent over WDM, and the rest are sent over IP. We can easily calculate probabilities of the message being sent over WDM, P(L>T) or IP, P(L<T). Therefore, P_{wdm} and P_{ip} are given by

$$P(L>T) = P_{wdm} = \frac{T-a}{b-a}$$

$$P(L<T) = P_{ip} = \frac{b-T}{b-a}$$

In order to analytically calculate the message delay, we model each of the paths the message can take through the system as a queue. Therefore, we have a system with two queues, one for the IP path and one for the WDM path (Figure 12).

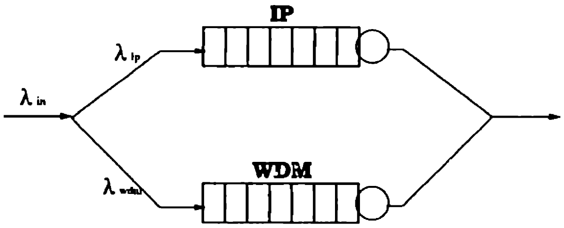


Figure 12. Queuing model of IP/WDM switch

The total arrival rate to the system, λ_{in} ,is the *arrival rate of each individual node multiplied by the number of nodes*. Therefore, the respective arrival rates to the WDM and IP queues are:

$$\lambda_{wdm} = P_{wdm}\lambda_{in} = \frac{(b-T)}{(b-a)}\lambda_{in}$$

$$\lambda_{ip} = P_{ip}\lambda_{in} = \frac{(T-a)}{(b-a)}\lambda_{in}$$

We approximate both the WDM and the IP systems as an M/G/1 queue (Figure 13). Even though, the WDM queue should be an M/G/w queue, where *w* is the number of WDM channels, we do not have solutions for the delays in such a queue. However, we can approximate M/G/w queue with the arrival rate λ_{in} as w independent M/G/1 queue with an arrival rate of $\lambda'_{wdm} = \frac{\lambda_{wdm}}{w}$. This approximation appears reasonable when the system is very lightly loaded, so that λ_{in} is approximately 0. In this case, there will be no queueing and every arriving packet will be served immediately. Therefore, the delay for both M/G/w and M/G/1 systems will be the transmission time of the packet. For very heavy loads, the delay in the M/G/1 system is also approximately the same as the delay in the M/G/w system, because all the servers in both system will always be busy. This means both systems are work-conserving and will have the same delay.

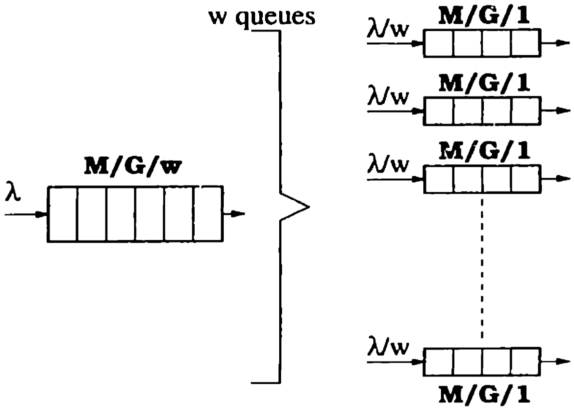


Figure 13. M/G/w approximation for the WDM system

Similarly for the IP queue, we approximate the M/G/C queue, where C is the capacity of the router as an M/G/1 queue with arrival rate $\lambda'_{ip} = \frac{\lambda_{ip}}{C}$.

Using the M/G/1 approximation, we can now calculate the expected delay of the message in the system using the Pollaczek-Khinchin (P-K) formula [10]:

$$\overline{W} = \frac{\lambda\overline{X^2}}{2(1-\rho)} + \overline{X}$$

where $\overline{X}$ is average message length, $\overline{X^2}$ is the second moment of $\overline{X}$, and $\rho = \lambda\overline{X}$ is the utilization.

For the WDM queue, the average service time, second moment and utilization are given by

$$\overline{X}_{wdm} = \frac{(b+T)}{2} + D_{setup}$$

$$\overline{X^2_{wdm}} = \int_{T+D_{setup}}^{b+D_{setup}} \frac{x^2}{b-T}dx = \frac{(b+D_{setup})^3 - (T+D_{setup})^3}{3(b-T)}$$

$$\rho_{wdm} = \lambda'_{wdm}\overline{X}_{wdm}$$

The expected delay in WDM queue is then:

$$W_{wdm} = \frac{\lambda'_{wdm}\overline{X^2{}_{wdm}}}{2(1-\rho_{wdm})} + \overline{X}_{wdm}$$

We can also calculate W_{ip} in the same manner. Figure 14 shows the approximation of the IP queue as an M/G/1 system. Originally, the IP queue system consists of N queues which represent input ports on the router and C servers which represent the CxC switching fabric of the router. The approximation contains two steps. First, N queues, each with input rate λ are approximated as one queue with arrival rate of $N\lambda$. Then, the system with one queue and C servers is approximated as a system with a single server and one queue with the new arrival rate $\frac{N\lambda}{C}$.

Now, if we assume that all IP packets constituting a message arrive to the same queue within a router simultaneously, then all the packets from that queue will need to be sent out sequentially before the new message in the queue will get serviced.

Therefore, the average service time of the message in an IP queue is its average duration,

$$\overline{X}_{ip} = \frac{(a+T)}{2}$$

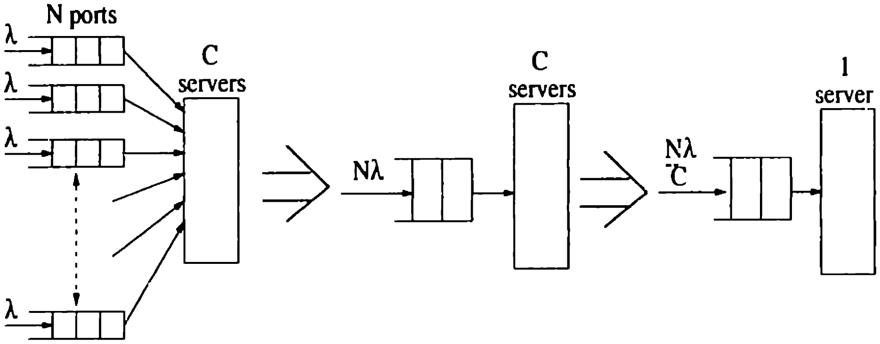


Figure 14.M/G/C approximation for the IP system

The calculations of $\overline{X^2_{ip}}$, ρ_{ip} and W_{ip} are similar to the WDM case:

$$\overline{X^2_{ip}} = \int_a^T \frac{x^2}{T-a}dx = \frac{T^3-a^3}{3(T-a)}$$

$$\rho_{ip} = \lambda'_{ip}\overline{X}_{ip}$$

$$W_{ip} = \frac{\lambda'_{ip}\overline{X^2}_{ip}}{2(1-\rho_{ip})} + \overline{X}_{ip}$$

Averaging over the IP and WDM systems, we can obtain an expression for the expected delay of a message entering an IP/WDM system:

$$W_{total} = P_{wdm}W_{wdm} + P_{ip}W_{ip} = \frac{(b-T)}{(b-a)}W_{wdm} + \frac{(T-a)}{(b-a)}W_{ip}$$

Using Maple, we can plot graphs of W_{total} vs. T (Figure 15). We can see that the system is stable only in one region between 0.45 and 0.8 (a). On the left of the stable region, the system is unstable because of the delay in the WDM queue, and on the right it is unstable due to the delays in the IP queue. If we increase D_{setup}, then the delay in the WDM queue will increase and the region of stability will begin to shrink(b). Decreasing of W or C will decrease the capacities of the WDM and IP systems which will cause increase delays in the WDM and IP queues. We can also shrink the stability region by increasing the load, since too much traffic to the IP and WDM queues will also cause increased delays in the system. Eventually, we can reach the point where the system is not stable for any value of T.

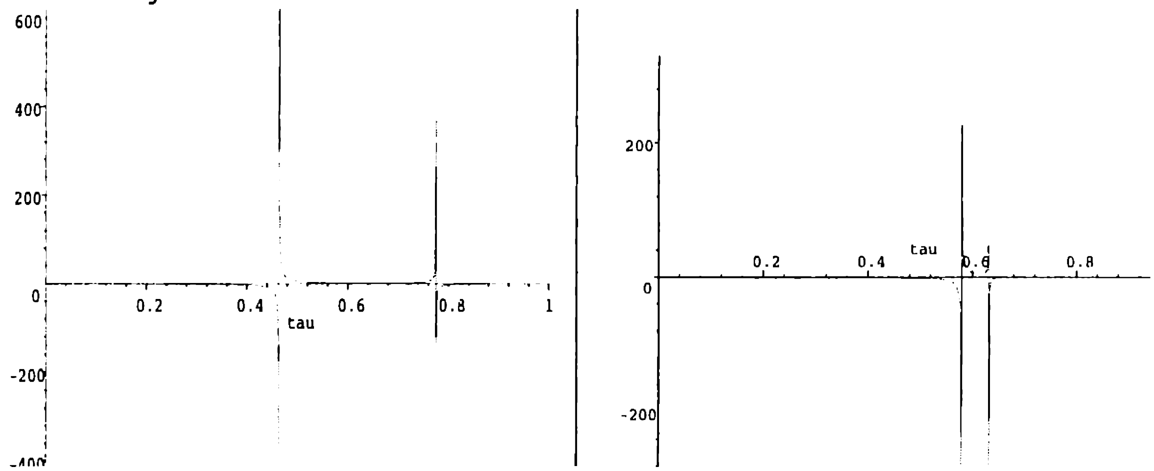


Figure 15. W_{total} vs. T for a) a=0, b=1, D_{setup}=0.1s, W=5, C=3, λ_{in}=10.

b)a=0, b=1, D_{setup}=0.4, W=5, C=2, λ_{in}=10.

Given D_{setup},W, C, λ_{in} and the message size distribution, we want find a value of T which will always place us in the stable region of the graph, if it exists. This is the first and foremost goal of finding an optimal T. Once the system is stable, we would like to find a value of T which minimizes the delay in the network. However, minimization of the delay is possible only in the stable system, so finding a stable region is the most important task. One approach that will guarantee stability is to make sure the utilizations of the WDM and IP queues are equal. Setting $\rho_{ip} = \rho_{wdm}$ and solving for T we obtain:

$$T = \frac{\sqrt{(CD_{setup})^2 + (Cb)^2 + 2C^2bD_{setup} + Ca^2W + Wb^2C + 2WCbD_{setup} + (aW)^2} - CD_{setup}}{C + W}$$

This is a reasonably concise solution for T and it would be easy to implement in the real system. Results shown in Tables 1-6 prove that this T not only guarantees stability, but also achieves a delay which is very close to the minimum delay.

4.3.Simulation Results and Analysis

Figure 16 shows results from the Opnet simulations. We simulated the system for the following conditions: capacity of the router (C) =5, total number of WDM channels (W) =10, message duration uniformly distributed between 0.001 and 1 seconds, and setup delay between 0.1 and 0.5 seconds. The the capacity of the system is a sum of the capacities of the IP and the WDM subsystems:

$$Capacity_{total} = Capacity_{wdm} + Capacity_{ip}$$

Assuming the same transmission rate (TXrate) for both the IP and the WDM systems and ignoring the effects of channel blocking and receiver collisions, the capacity of WDM system can be crudely approximated as $Capacity_{wdm} = W \times TXrate$, where W is the number of channels available for flow switching. Similarly, the capacity of the IP system is $Capacity_{ip} = C \times TXrate$. The load on the overall IP/WDM system is $\lambda_{in}\bar{X}$, where λ_{in} is the total arrival rate due to all the transmitting nodes and $\bar{X}$ is the average length of the message. We can also express λ_{in} as $N\lambda$, where N is the number of transmitting nodes and λ is the arrival rate to each individual node.

The system will be unstable if the incoming load is more than system's capacity, so that

$$(W+C)TXrate < \lambda_{in}\bar{X}$$

In our simulations, $\lambda_{in}=14$. Since the message duration is uniformly distributed between 0.001 and 1 seconds, $\bar{X}$ is approximately 0.5 seconds. TXrate is normalized to be one maximum size message per second. If we substitute the values of C,W and λ_{in} in the previously describe inequality, we get (5+10)*1>14*0.5, so the IP/WDM system can easily handle the applied load of 14 messages/second.

For the plots in Figure 16 we observed that for a small T, the system was unstable since most of the messages try to go through the WDM path. WDM cannot handle the entire load, because of the limited number of WDM channels. The queueing delay on the WDM part of the system became infinite and the system became unstable. For larger values of T, more messages got sent through IP and the system stabilized. As T continued to increase, most of the messages went through IP, which eventually became overloaded and unstable. This condition can be seen in the regions of increasing delay on the right sides of all the graphs. In addition, we observed that as we increased the setup delay from 0.1 to 0.5, we shrunk the stable region by moving the left boundary of the stable region to the left, while the right boundary remained at the same place. We can also drive the system to instability by increasing the load on the system or decreasing the switching capacity of the router. Thus, it is possible, that for very high loads and high setup delays, that the system will not be stable for any T.

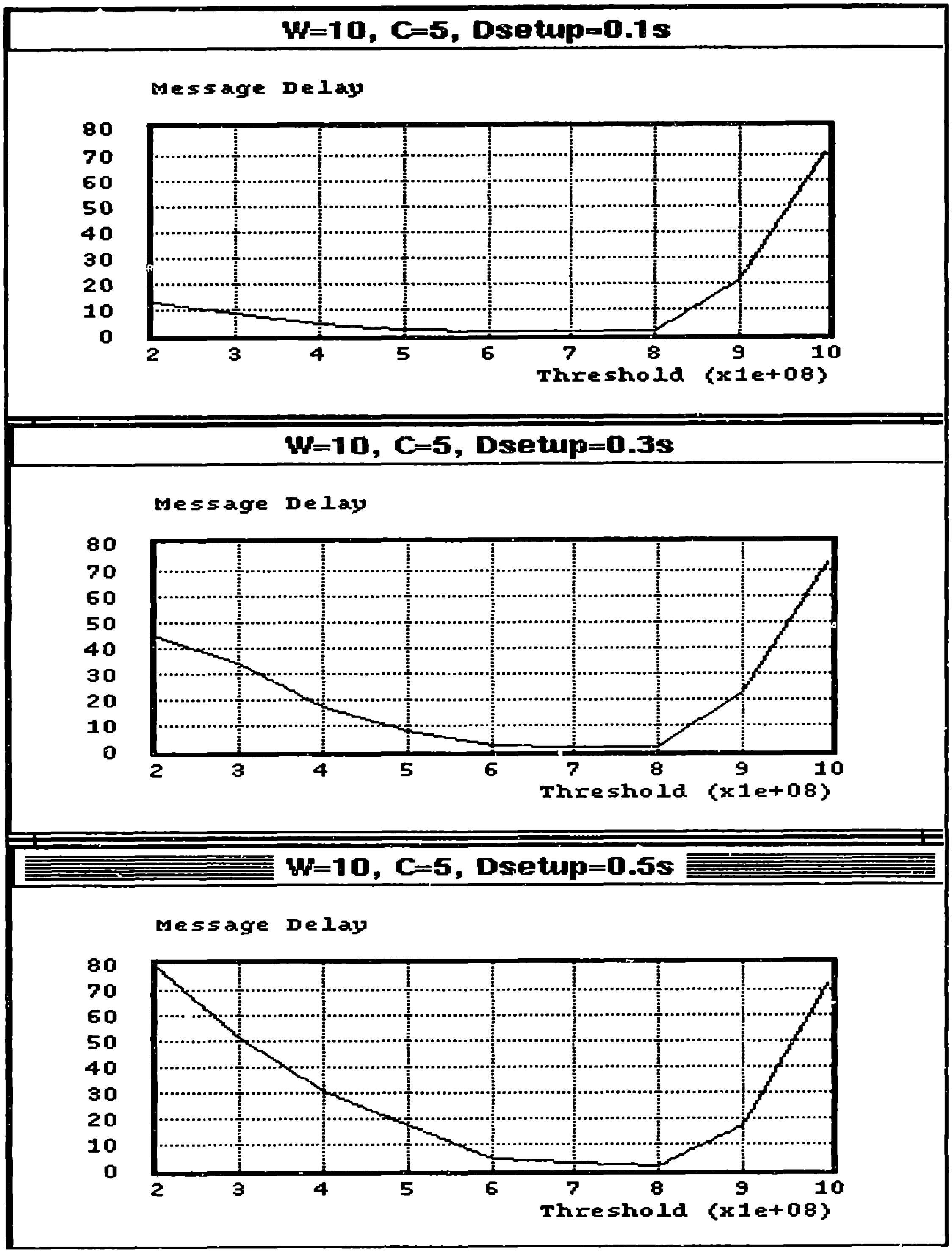


Figure 16. Simulation results for various Dsetup and T
a) Dsetup=0.1, b) Dsetup=0.3, c) Dsetup=0.5

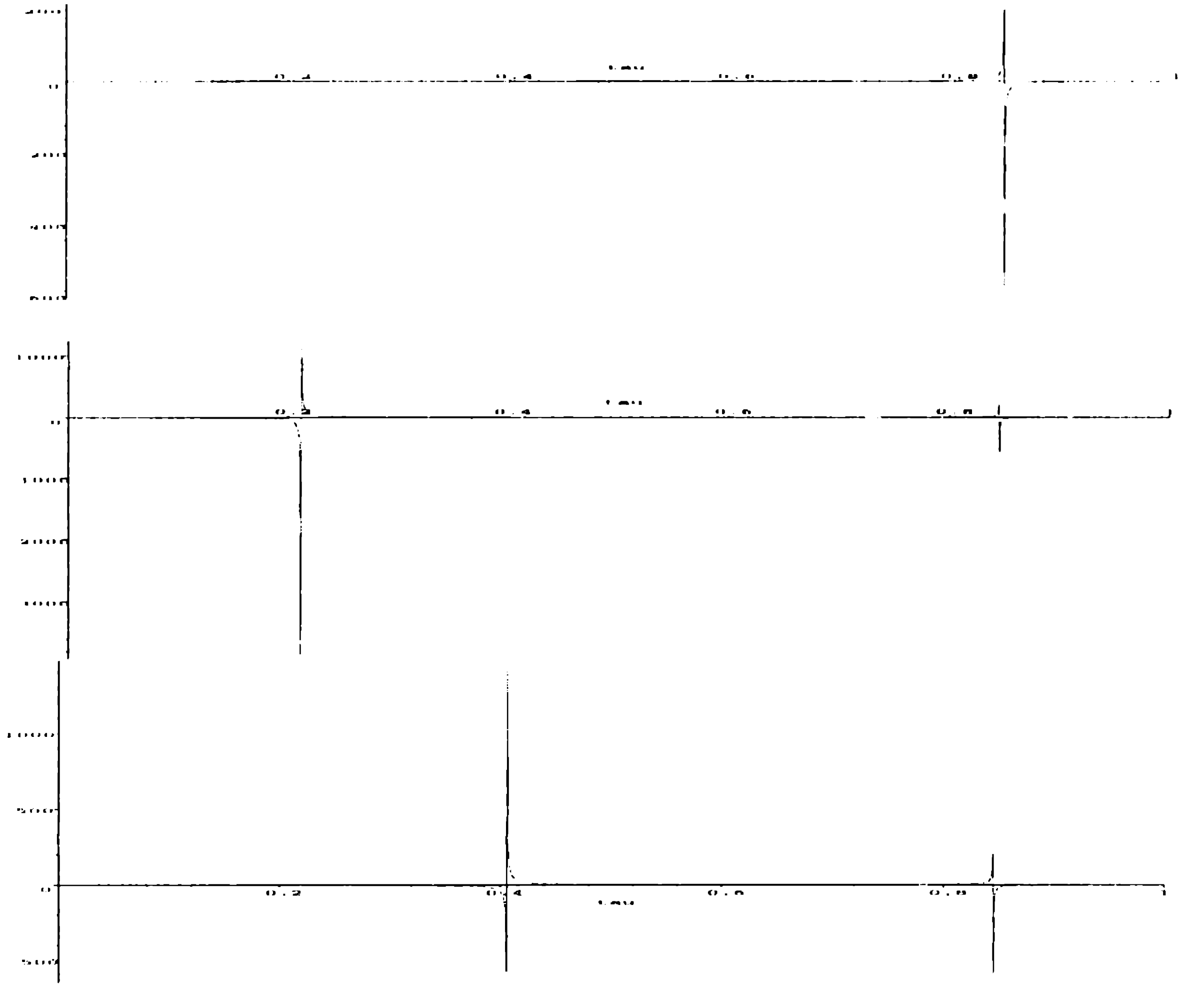


Figure 17. Analytical results for a system with C=5, W=10, λ_{in}=14
a)Dsetup=0.1, b)Dsetup=0.3, c)Dsetup=0.5

Figure 17 shows analytical results for the same parameters as described for the simulation. (C=5, W=10, λ_{in}=14). We also varied the setup times between 0.1seconds and 0.5 seconds while keeping the capacity of the router and the load constant. We can see that the stable region moves to the right as Dsetup is increased. There was no change in the position of the border of the stable region defined by IP, since the capacity of the router was not changed. Had we decreased the capacity of the router, the stable region would shrink to the left. If we compare stable regions in the simulations and analyses, we can see that they are essentially the same. However, in our analysis we did not compensate for HOL blocking within the router. Since HOL blocking is known to reduce throughput to 59%, the capacity of the router may be reduced by a similar number. Also, the capacity of the WDM queue is reduced by the channel blocking and receiver collisions. Thus, the analytical model is only a rough approximation of the real system, but we will later show that it produced reasonably accurate results.

So far, we have discussed several analytical methods for finding T which would place us at the stable region of the system. For example, we can set utilizations of IP and WDM queues equal to each other; in other words $\rho_{ip} = \rho_{wdm}$.

We could also try to find the T which would make the delays of the IP and WDM queues equal (T_W). Finally, using Maple we could find the actual minimum point on the *Message Delay vs. T* graph, and the value of T which corresponds to that point (T_{min}). In addition, we have results from the simulation (T_{sim}), which represent the simulated value of T that minimizes overall delay.

There is one important note about the simulation results. We feel that the length of the OPNET simulations was not sufficiently long to yield the steady state results. While running OPNET simulations, we were limited by the nature of the simulation software which requires a lot of processing power, so we could not run the simulation for very long durations. Therefore, the results obtained in the simulations might differ from the actual steady state results although we feel that they are sufficiently close to provide useful insight.

Tables 1 through 6 show the comparison between the values of T and the message delays (W_{total}) for different methods discussed in the previous paragraph. The heading of each table indicates the number of the WDM channels (W), capacity of the router (C) and the load on the system (λ_{in}). Each row contains the values obtained in the simulation run with different values of D_{setup}. From the results we see that analytical results are similar to the simulation results. We also see that T_{ρ} is usually quite close to the actual minimal point on the graph (T_{min})

Table 1: C=5, W=5, λ_{in}=10.

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{total}	0.721, 0.87	0.765, 1.02	0.795, 1.15
T_{ρ}, W_{total}	0.726, 0.87	0.756, 1.02	0.78, 1.16
T_W, W_{total}	0.859, 1.283	0.883, 1.49	0.90, 1.743
T_{sim}, W_{total}	0.725, 0.66	0.78, 0.73	0.8, 0.89

Table 2: C=5, W=5, λ_{in}=14

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{total}	0.722, 1.39	0.758, 1.92	0.80, 2.67
T_{ρ}, W_{total}	0.726, 1.39	0.756, 1.92	0.78, 2.682
T_W, W_{total}	0.776, 1.791	0.79, 2.49	0.81, 3.47
T_{sim}, W_{total}	0.725, 0.9	0.74, 1.11	0.8, 1.6

Table 3: C=5, W=10, λ_{in}=10

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{total}	0.7, 0.71	0.77, 0.83	0.81, 0.94

Table 3: C=5, W=10, λ_{in}=10

D_{setup}(s)	0.1	0.3	0.5
T_{ρ}, W_{total}	0.60, 0.71	0.637,0.84	0.66, 0.963
T_W, W_{total}	0.844, 1.118	0.871, 1.35	0.89, 1.58
T_{sim}, W_{total}	0.7, 0.675	0.78, 0.75	0.8, 0.83

Table 4: C=5, W=10, λ_{in}=14

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{total}	0.673,0.847	0.725, 1.05	0.75, 1.26
T_{ρ}, W_{total}	0.60, 0.848	0.637,1.05	0.66, 1.28
T_W, W_{total}	0.733, 1.157	0.766, 1.55	0.78, 1.89
T_{sim}, W_{total}	0.69, 0.82	0.72, 1.02	0.76, 1.26

Table 5: C=10, W=5, λ_{in}=10

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{total}	0.82, 0.683	0.874, 0.73	0.9, 0.763
T_{ρ}, W_{total}	0.83, 0.685	0.851, 0.73	0.87, 0.775
T_W, W_{total}	---	---	---
T_{sim}, W_{total}	0.81, 0.688	0.85, 0.71	0.9, 0.772

Table 6: C=10, W=5, λ_{in}=14

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{total}	0.82, 0.811	0.859, 0.887	0.88, 0.942
T_{ρ}, W_{total}	0.83, 0.811	0.851, 0.888	0.87, 0.953

Table 6: C=10, W=5, λ_{in}=14

D_{setup}(s)	0.1	0.3	0.5
T_W, W_{total}	---	---	---
T_{sim}, W_{total}	0.81, 0.764	0.85, 0.86	0.9, 0.99

In general, we would like to have a scheme that would achieve low delays in the network with the least available information about the network. If traffic conditions were fixed and the message distribution was known, our algorithm could be applied to minimize the delay in the network. However, typically, we do not normally know the distribution of the message size or the load in the network which makes the static scheme impractical for implementation in the real system. Therefore we need some kind of dynamic algorithm which will allow us to adjust to the changing traffic conditions in the network.

5. Dynamic Threshold Schemes.

There are several approaches to changing the threshold dynamically in response to the changes in network load. We propose two approaches: a centralized scheme and a distributed scheme. In the centralized scheme, only the IP/WDM switch decides on the designated path of the message. In this scheme, the transmitting nodes do not play a part in the decision process which is localized at the switch. The alternative to the centralized scheme is the distributed scheme, where each transmitting node makes a decision for itself, based on its own estimate of the expected delay of the packets using each path. This method doesn't require the computation of the threshold at the switch, since all the decisions are made locally at each node. However, each node needs enough information to accurately approximate the expected delays through each path.

There are advantages and disadvantages to both schemes. The centralized scheme is much easier to implement for smaller networks, since the decision process is localized and only the switch has to maintain and collect the information needed for the calculation of the threshold. However, the centralized scheme is less practical for a larger network where having a single decision point increases a chance of a failure of the whole network and requires the switch to make too many decisions. On the other hand, the distributed scheme avoids the central decision point, but it requires that each node possess enough information to make a "good" decision, so large amounts of state information need to be maintained and passed through the network using protocols similar to Open Shortest Path First (OSPF) and Border Gateway Protocol (BGP) which are used to compute routes in the internet. Thus the implementation of the distributed approach is going to be much more involved, as it is with many distributed algorithms. However, there is a fairness issue associated with the centralized scheme. It is inherently unfair, since as discussed below, it makes a decision on the *total average* load of the switch, thus ignoring the loads due to the individual nodes. This is unfair, since the adjustment of the threshold in response to the *total average* load will penalize all nodes equally, although their contribution to the average load might be unequal. The distributed scheme remedies this problem by avoiding the use of a fixed threshold and letting each node make an individual decision which will be optimal for that node. Below is a more detailed discussion of the centralized and the distributed schemes.

5.1. Centralized Decision Scheme

In the centralized decision scheme, the switch makes measurements of the average load on the IP and WDM queues and adjusts the threshold in order to balance the load between the two queues.

We have to make sure that neither of the queues gets overloaded, since that would cause higher delays and possible instability. Therefore, we need to dynamically balance the load between the queues, so when one queue gets too much traffic, the switch redirects the traffic to the other queue which is underutilized. This traffic redirection is done dynamically by shifting the threshold in a direction that reduces the load on the overutilized queue. Our goal is to devise an algorithm that keeps the load on both queues balanced and quickly reacts to the sudden changes of the load. We hope that such an algorithm will keep the system in the stable region and achieve the goal of reducing message delays.

5.1a. Average load measurements

We need to make accurate measurements of the load on the IP and the WDM queues. Since our router is slotted to the duration of one IP packet, we measure the load at each slot and then calculate the average load over a reasonably long period of S slots. The load of the IP router per slot is defined as the *number of queues serviced during the slot divided by the maximum capacity of the router* (C). Similarly, the load of the WDM system is defined as *the number of busy channels during a slot divided by the total number of WDM channels* (W). Using these definitions we can calculate average loads Cav and *Wav* over some time period S. However, WDM channels remain occupied for much longer durations of time than one IP slot because every transmission on any WDM channel lasts for *The message transmission time + Setup Time*. Thus we need to choose a measurement period which is presumably much longer then a WDM message.

Having calculated Cav_i and Wav_i for some measurement period *i*, we need to apply some estimation technique to these values to get longer term estimates of the average utilizations $\hat{Cav}$ and

$\hat{Wav}$. One way to do this is to use the first order Auto Regressive Moving Average(ARMA) technique for our estimation [11]. In ARMA, the estimated value of some parameter X that we are trying to measure depends not only on the current measurement of X but also on the previous measurements. For example, α denote the weighting factor of the current measurement X_i of parameter X, and $\hat{X}_i$ as the current estimation of X. The estimated parameter X is obtained recursively by

$$\hat{X}_i = \alpha X_i + (1-\alpha)\hat{X}_{i-1}$$

Applying this to our estimates of average load on IP and WDM queues we obtain:

$$\hat{Cav}_i = \alpha Cav_i + (1-\alpha)\hat{Cav}_{i-1}$$

$$\hat{Wav}_i = \alpha Wav_i + (1-\alpha)\hat{Wav}_{i-1}$$

5.1b. Threshold Scheme

The goal of the dynamic threshold is to keep the load on the IP and the WDM queue equal. This is done by comparing $Ca\hat{v}$ to $\hat{Wav}$ and moving the threshold in a direction that redirects the traffic from the more loaded queue to the less loaded queue. We use additive increase/decrease rule to change the threshold. We change the threshold by the difference between $\hat{Cav}$ and $\hat{Wav}$ which is multiplied by some positive factor M. Thus, the threshold adjustment rule is:

$$T_i = M(\hat{Wav}_i - \hat{Cav}_i) + T_{i-1}$$

Therefore, if $\hat{Cav} > \hat{Wav}$, then we want to decrease the load on the router and the lower threshold to send more traffic through WDM. This will happen, since $\hat{Wav}$ - $\hat{Cav}$ will be negative. On the other hand, if $\hat{Wav} > \hat{Cav}$, then the WDM queue is overloaded and the threshold will shift up. The amount of increase/decrease of the threshold depends on a constant M, which can be tuned for optimal performance.

5.1c. Adaptive α.

We choose the weighting factor α to be adaptive rather than a constant in order to filter the stochastic oscillations of our measurements. With a constant weighting factor, the estimates of $\hat{Cav}$ and $\hat{Wav}$ are also oscillating about the mean value. Decisions based on these estimates are not reliable. For example, if α is large, the value of $\hat{Cav}$ or $\hat{Wav}$ will depend mostly on the current sample value. Since those values might oscillate, the values of $\hat{Cav}$ and $\hat{Wav}$ will also oscillate, which will force the threshold to oscillate as well. If those oscillations are too large, the threshold scheme might not work very well, since it will react too much to the transient changes in the load. On the other hand, if α is too small, then $\hat{Cav}$ and $\hat{Wav}$ will not depend on the current samples and thus the threshold will not change rapidly enough in response to the changes in the load. Therefore, we would like to filter the measurements in such a way that the stochastic fluctuations of the measurements will be smoothed out while the persistent changes of statistics due to overloading or underloading of the queues will be tracked quickly. We opt for an adaptive moving average technique in the sense that it adapts to the changing loads. We want α to be small when the loads are fluctuating about the mean but large when there are abrupt changes.

Using the algorithm described in [11], the adapting α is calculated as

$$\alpha = \frac{kE^2}{\varepsilon_{i+1}}$$

$$E = X_i - \hat{X}_{i-1}$$

$$\varepsilon_{i+1} = kE^2 \div (1-k)\varepsilon_i$$

where E and ε are the estimation error and the estimate of the squared estimation error, respectively. By varying k we can change the responsiveness of α to changing load conditions. For a case where C=5, W=5, Dsetup=0.3 and λ_{in}=14, we found that the delay is minimized when k=0.8. (Figure 18) This might seem a bit surprising, since initially we thought that a very responsive α (high k) will lead to the increased delays. However, from the graph, it seems that the delay is almost constant for k greater than 0.4 and does not increase much for higher values of k, which means that α can be very responsive without adversely affecting delay in the system.

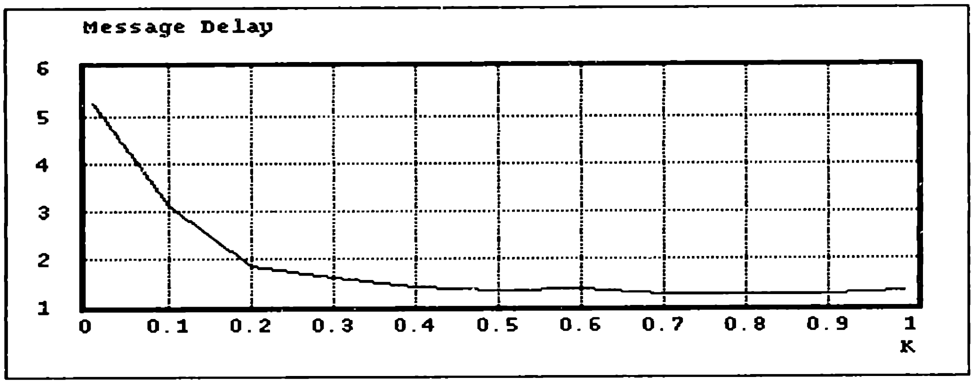


Figure 18. Message Delay vs. k, [W=5, C=5, Lin=14, Desetup=0.3]

One possible explanation of this is that rapid changes of the threshold in response to the transient changes of the load are not necessarily detrimental to the system performance. Several other simulations with different values of C, W and Dsetup showed similar behavior, so we decided to keep k=0.8 for the rest of our experiments.

5.1d Moving Threshold Analysis

We simulated the moving threshold scheme for various values of system parameters. In general, the plots of Threshold vs. Time resemble Figure 19 [12]. We would like the threshold to converge to some steady state value. Convergence is generally measured by the speed with which the system approaches the steady state. However, in most cases, the threshold did not converge to a single steady state, due to the limited nature of feedback. Instead, it reached an "equilibrium" in which it oscillated around the steady state. In Figure 19, we define smoothness. The smoothness is a term which describes the size of oscillations of the threshold. The smoother the plot, the smaller the oscillations are about the goal value.

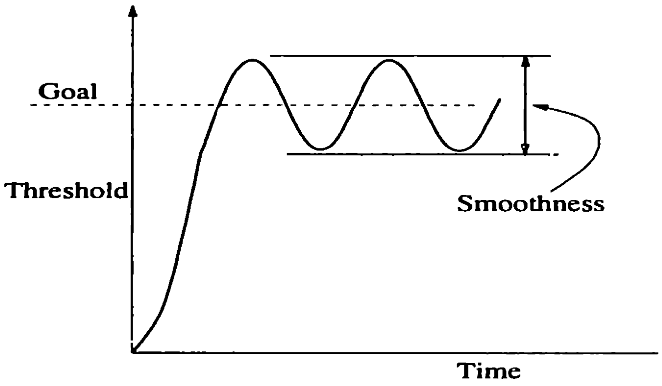


Figure 19. General form of the Threshold vs. Time plot

Figure 20 shows the plots of threshold vs. time for different values of k. As explained above, the value of α affects both the smoothness of the curve. For small k, α is not very responsive and this leads to lower smoothness, because the threshold does not change quickly in response to a changing load, causing larger oscillations. On the other hand, large k leads to a very responsive α, which increases the smoothness of the curve (Figure 20.c). We would like to have a curve that is smooth. In general, the smoothness of the curve depends on the incoming load and the ability of the system to handle it. The better the system can handle the load, the smoother will the threshold curve be. This happens because neither of the queues are heavily loaded and thus there are no large changes in $C\hat{a}v$ and $W\hat{a}v$. Since the threshold depends on the change between $C\hat{a}v$ and $W\hat{a}v$, if the change is small, the threshold will not oscillate much.

This rule, however, does not apply to the case when the system is overloaded. When the system is overloaded, the threshold no longer oscillates, since both the WDM and the IP queues become overloaded and the values of $C\hat{a}v$ and $W\hat{a}v$ remain almost constant and approach 1. Thus the difference between $C\hat{a}v$ and $W\hat{a}v$ will be minimal and the threshold will not oscillate. Figure 21 (a) shows a plot of an overloaded system where the initial threshold was set to 5×10^8. However since the system became overloaded from the beginning, the threshold change was very small over the 70 second time period.

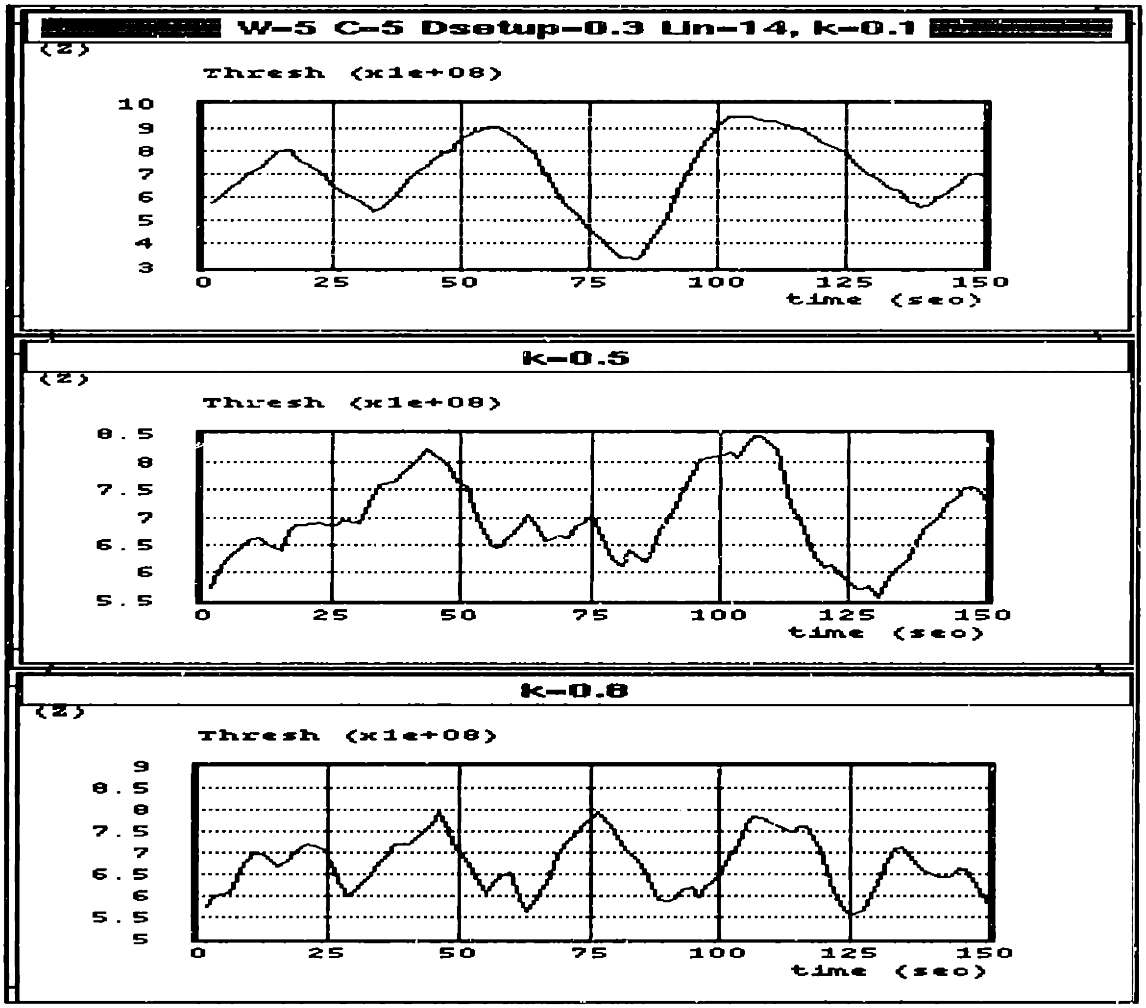


Figure 20. Delay vs. time, for k=0.1, 0.5 and 0.8 in a system where W=5, C=5 and Dsetup=0.3

Unfortunately, this example shows that our scheme does not have a way to lead itself out of the unstable region, once the system gets there. Thus, the threshold scheme works only in the situation where the system is not overloaded.

A similar situation takes place when the system is very lightly loaded. In such cases the loads of the IP and WDM queues approach 0. Since $C\hat{a}v$ and $W\hat{a}v$ are almost constant and close to 0, their difference will be constant and very small, so the value of threshold will barely oscillate.

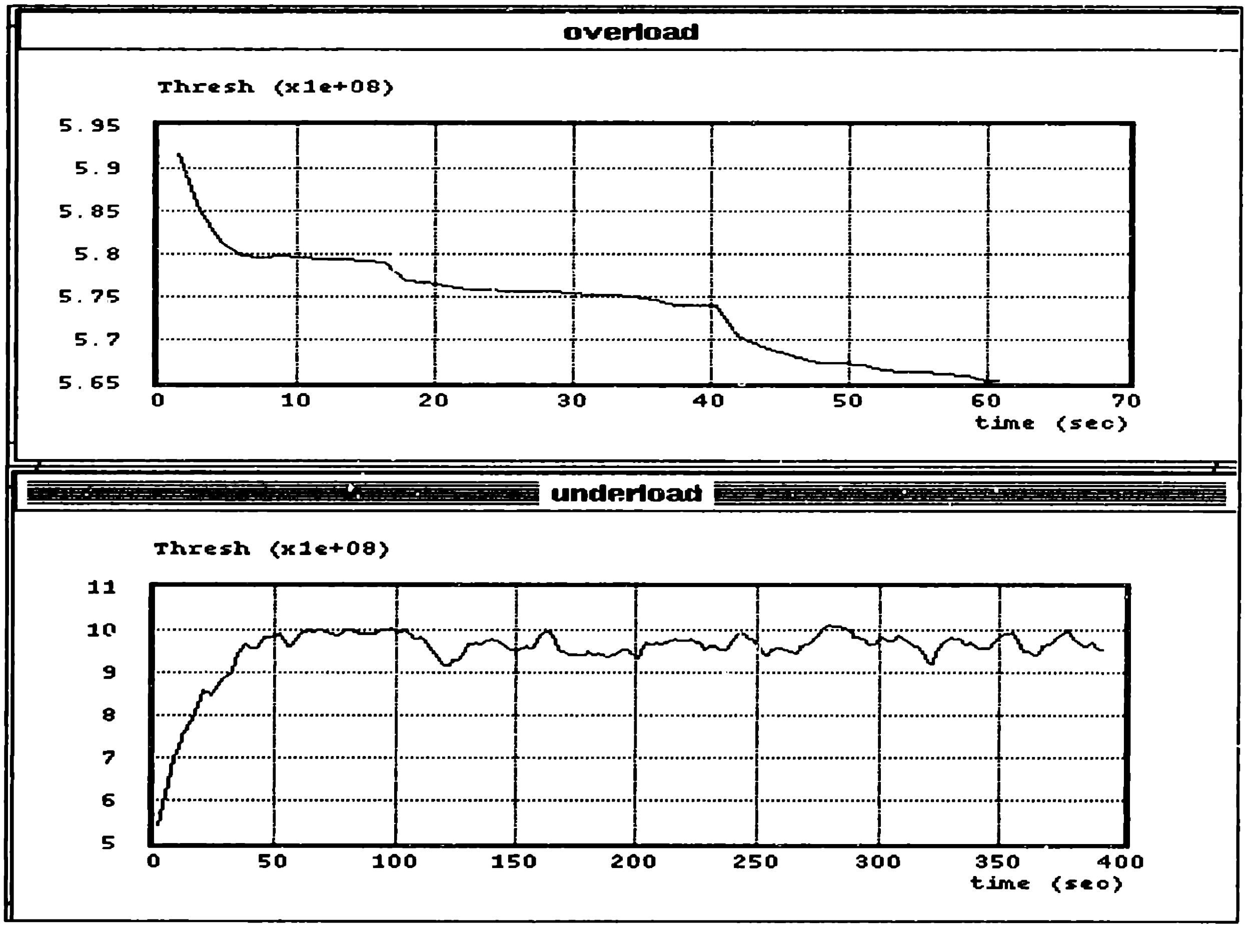


Figure 21. Threshold oscillations in an (a) overloaded system and (b)underloaded system

We can see this happening in Figure 21(b) where the threshold slightly oscillates between 9.5×10^8 and 10^9. The load in this example was so light compared to system capacity

(C=10, W=5, Lin=3) that the IP queue could handle all the traffic and there was no need to use the WDM path, which added additional setup delay to the transmission.

Figure 22 shows the plots of the thresholds vs. time for different system parameters. The average values of thresholds and delays are given later in the chapter. We can see that the size of

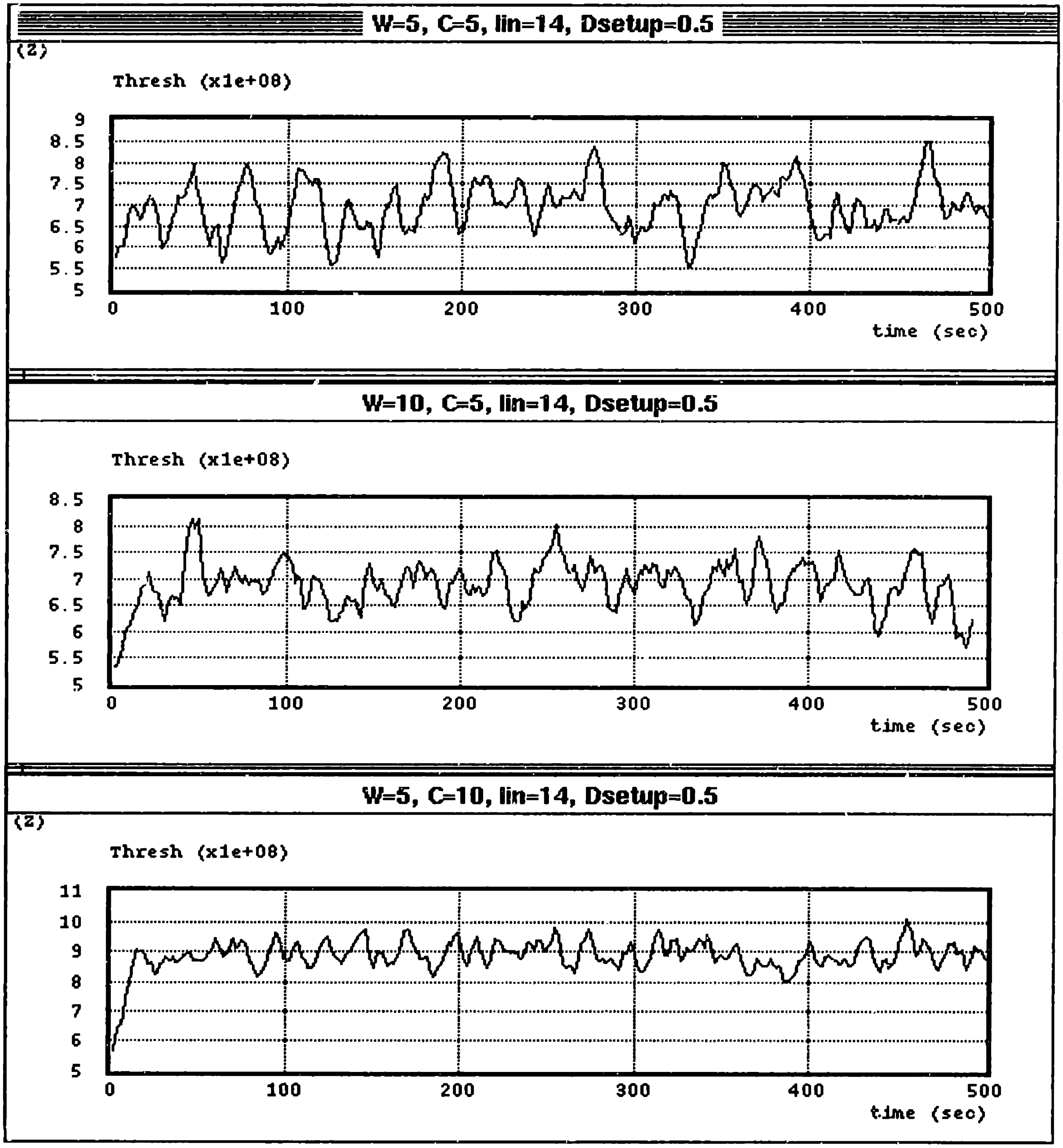


Figure 22 Threshold vs. Time for systems with varying capacity

threshold oscillations depends on the load of the system. In the case when C=5 and W=5 and the total load is 14 packets/second, the threshold curve is not smooth at all - there are large oscillations in the value of the threshold at every observation period. This happens because every observation period the switch tends to distribute the traffic between the IP and WDM queues, but since the load is high, even the small increase of load around the optimal threshold tend to overload the queues. Thus since both queues are heavily utilized, the switch keeps alternatively overloading them, which results in the oscillations of the threshold value.

In case when W=10 and C=5, the overall capacity of the system increases, which in turn increases the smoothness of the threshold curve. However, we see that the plot is smoother for the case where W=5, C=10(c) than for the case where W=10, C=5 (b). This means that the capacity of the system in (c) is higher than in (b). Even though the total number of (IP+WDM) channels is kept constant in both cases, the system with the higher capacity router performs better. This happens because in our simulation the IP queue has higher potential capacity than the WDM queue. The reason is that the throughput of the IP queue is limited only by the HOL blocking. The capacity of the WDM is limited, however, by receiver contention and setup delay. Thus adding extra IP channels leads to better performance than adding extra WDM channels at the same transmission rate.

5.2 Distributed Decision Scheme

5.2a.Decison Scheme

One major problem with the centralized decision scheme is that it is unfair. Since it reacts to the average load on the IP and WDM queues, it does not take into account the needs of individual users. Therefore, while 9 out of 10 subqueues might have low utilization, the 10th subqueue is highly overloaded, but since it contributes only 1/10 of the load to system on average, the overall load might still appear low and the centralized controller will not pay attention to the overloading of that queue. Thus there is a need for the more fair decision scheme. Such a fair scheme is a scheme where each user makes a decision for himself based on the expected delay of its message through the network. If a user has enough information about the delays in the system to calculate the expected delay that the sent packet will experience in the network, it will always send the packet on the path that has the lowest expected delay. Therefore, each of the nodes will make an entirely local decision trying to minimize its own delay and the overall scheme should achieve performance that is more fair than with the centralized scheme. In this scheme, users make estimates of the delays through IP and WDM (D_{ip}, D_{wdm}) and if $D_{ip} > D_{wdm}$ then the message is sent to the WDM queue. If $D_{ip} < D_{wdm}$ then the message is sent to the IP queue.

5.2b.Rate measurements

The difficulty with the distributed scheme is that we are trying to make a reasonable estimation of the delay in the network with the minimal available data. We assume that each node knows the number of the messages that await transmission within its own queue. We also assume that each node has access to statistics such as the average utilization of the IP router and the WDM channels. With these statistics, we try to estimate the expected delay the message will experience when entering the network.

Unfortunately we do not know the service rates for the WDM and the IP queues. However, from the calculations done in the centralized scheme, we know the average values of load on the WDM and IP queues ($C\hat{a}v$ and $W\hat{a}v$). If we can obtain the average number of WDM subqueues (B_{wdm})

and IP subqueues (B_{ip})that are non-empty during the previous sampling time, we can estimate R_{wdm} and R_{ip} which are the respective average service rates of the IP and WDM queues:

$$R_{wdm} = \frac{\hat{Wav}}{B_{wdm}}$$

$$R_{ip} = \frac{\hat{Cav}}{B_{ip}}$$

These rates R_{wdm} and R_{ip} are fractions between 0 and 1 and represent the probability that a busy subqueue is served during the slot.

Next we estimate the minimum delay the message will experience in the WDM queue. The arriving message will have to wait some time D_{wdm} for all the messages in front of it to get transmitted. Since we know the exact size of the messages we can express the overall size of the queue in bits: $Q_{wdm}(bits)$. The time required to transmit all the messages in the queue including the incoming message will be $\left(\frac{Q_{wdm}(bits)}{TXrate} + \frac{L}{TXrate}\right)$, where TXrate is the transmission rate of the network. In addition, we also know the total number of messages in the queue, and we can estimate the delay associated with setting up a WDM channel for each of those messages. Assuming R_{wdm}=1 and that there is no waiting due to receiver contention, we obtain:

$$D_{wdm} = \left(\frac{Q_{wdm}(bits)}{TXrate} + \frac{L}{TXrate}\right) + D_{setup}(Q_{wdm}(messages) + 1) \quad .$$

However, R_{wdm} and R_{ip} may be lower than 1, since they are limited by channel contention. Thus, taking contention into account we can estimate the delay on the WDM system as:

$$D_{wdm} = \left(\frac{Q_{wdm}(bits)}{TXrate} + \frac{L}{TXrate}\right)/R_{wdm} + D_{setup}(Q_{wdm}(messages) + 1)$$

The expected delay for the IP queue can be estimated using similar reasoning. Our IP router is set up in such a way that each node has an individual subqueue within the router. Therefore, all the packets arriving from the same node to the router are inserted in order to the same subqueue. Thus, each subqueue contains only packets from a single node in their arriving order, so when the packet from node N arrives to the router, it only sees the previous packets from node N waiting for transmission in the queue. This means that the messages from the same node cannot get out of order within the router, and message *n* will always get transmitted after message n-1. In other words, the messages in the IP queue are stored in the same format as in the WDM queue, so the only difference between the IP and WDM delays is the absence of the setup delay in the calculations for the IP queue. Therefore, the expected delay of a message going through the IP queue can be estimated by:

$$D_{ip} = \left(\frac{Q_{ip}(bits)}{TXrate} + \frac{L}{TXrate}\right)/R_{ip}$$

Figure 23 shows the comparison of the real average delay experienced by the messages compared to our estimated average delay for a system with C=5, W=5, Lin=10. We can see that the estimated delay tends to be slightly lower than the real delay, although there are cases when the estimated delay is much higher than the real delay. The probable explanation for these discrepancies is the inexact estimation of R_{wdm} and R_{ip} which depend on the average values of $\hat{Wav}$ and $\hat{Cav}$. Since $\hat{Cav}$ and $\hat{Wav}$ are calculated at the beginning of the sampling period S, their values stay the same for the duration of that period until the next period. Since we use a large S in order to get better estimations of $\widehat{Cav}$ and $\widehat{Wav}$, this also means that towards the end of the sampling period, the actual service rate of the WDM and the IP queues changes, although our estimation will not indicate it until the next sampling period, when $\widehat{Cav}$ and $\widehat{Wav}$ are once again recomputed. Thus, there exist cases where estimated and real delays vary greatly, although for most cases the estimated delay is reasonably close to the real delay.

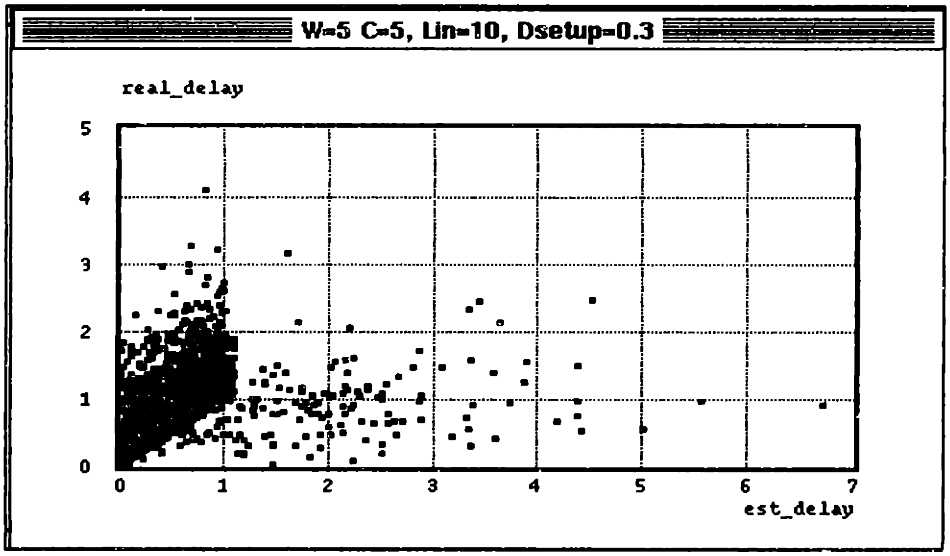


Figure 23. Real vs. Estimated Delay.

5.3 Simulation Results and the Analysis of the Dynamic Schemes

Tables 7 through 12 show the results achieved with the centralized and distributed dynamic schemes. The heading of each table indicates the number of the WDM channels (W), the capacity of the router (C) and the load on the system (Lin). Each row contains the values obtained in the simulation run with different values of D_{setup}. The first two rows show the results obtained from the static case analysis and simulations and are the same as in tables 1-7. T_{min} and W_{min} are the minimal threshold and minimal delay calculated using the analysis for the static scheme, where we chose the value of T that minimizes W. T_{sim} and W_{sim} are the results obtained from the simulations in the static scheme. T_{cen} and W_{cen} are the average threshold and the average message delay obtained using the dynamic scheme with centralized decision. W_{dist} is the average message delay obtained from the dynamic scheme with the distributed decision. There are no threshold measurements for the last case, since there is no threshold in the distributed decision scheme. As for the simulations for the static threshold, we feel that the length of the OPNET simulations was not sufficiently long to yield the steady state results. Thus, the results obtained in the simulations are likely to be inaccurate but provide us with some insight to the behavior of the scheme.

5.3a Centralized Decision Scheme.

In general, we see that the delays obtained with the centralized decision scheme are higher than the delays of the static threshold scheme for the simulations where the load on the system is high (Lin=14). In one particular case (W=5, C=5, Lin=14, Dsetup=0.5) we could not obtain a stable delay since the system always became unstable with the centralized scheme. The reason for this is a very narrow stable region of the system with such parameters. Since the dynamic threshold normally oscillates around the optimal threshold, in this case, where the stable region was narrow, the threshold often oscillated outside of the stable region, thus causing the system to be unstable. We believe that we still could have achieved stability in this system, but this would require further tuning of various parameters, such as M and k.

On the other hand, for the cases where the load is smaller (Lin=10) the delays are close to the static case results. In addition, the average thresholds in centralized and static cases were very close to each other. This tells us that the centralized scheme can achieve performance that is close to that of an optimal scheme where all of the parameters and loads are known in advance.

5.3b Distributed Decision Scheme

On average, the distributed decision scheme performed similarly to the centralized decision scheme. One major advantage of the distributed scheme was its higher tolerance to high load. For example, in the case where the centralized scheme failed to achieve stability (W=5, C=5, Lin=14, Dsetup=0.5), the distributed scheme obtained the delay which was very close to the static case result. In addition, the distributed decision scheme seemed to perform better for the cases where the WDM capacity was greater than the IP capacity.

Table 7: W=5 C=5 λ_{in}=10

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{min}	0.721, 0.87	0.765, 1.02	0.795, 1.15
T_{sim}, W_{sim}	0.725, 0.66	0.78, 0.73	0.8, 0.89
T_{cen}, W_{cen}	0.765, 0.67	0.77, 0.78	0.78, 0.90
W_{dist}	0.71	0.8	0.92

Table 8: W=5 C=5 λ_{in}=14

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{min}	0.722, 1.39	0.758, 1.92	0.80, 2.67
T_{sim}, W_{sim}	0.725, 0.9	0.74, 1.11	0.8, 1.6
T_{cen}, W_{cen}	0.675, 1.02	0.7, 1.67	unstable
W_{dist}	0.91	1.12	1.56

Table 9: W=10 C=5 λ_{in}=10

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{min}	0.7, 0.71	0.77, 0.83	0.81, 0.94
T_{sim}, W_{sim}	0.7, 0.675	0.78, 0.75	0.8, 0.83
T_{cen}, W_{cen}	0.637, 0.72	0.665, 0.85	0.69, 0.97
W_{dist}	0.73	0.81	0.857

Table 10: W=10 C=5 λ_{in}=14

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{min}	0.673,0.847	0.725, 1.05	0.75, 1.26
T_{sim}, W_{sim}	0.69, 0.82	0.72, 1.02	0.76, 1.26
T_{cen}, W_{cen}	0.621, 0.96	0.648, 1.21	0.67, 1.85
W_{dist}	0.91	1.13	1.39

Table 11: W=5 C=10 λ_{in}=10

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{min}	0.82, 0.683	0.874, 0.73	0.9, 0.763
T_{sim}, W_{sim}	0.81, 0.688	0.85, 0.71	0.9, 0.772
T_{cen}, W_{cen}	0.893, 0.685	0.9, 0.72	0.92, 0.77
W_{dist}	0.684	0.777	0.92

Table 12: W=5 C=10 λ_{in}=14

D_{setup}(s)	0.1	0.3	0.5
T_{min}, W_{min}	0.82, 0.811	0.859, 0.887	0.88, 0.942
T_{sim}, W_{sim}	0.81, 0.764	0.85, 0.86	0.9, 0.99
T_{cen}, W_{cen}	0.861, 0.84	0.87, 0.88	0.89, 0.94
W_{dist}	0.82	1.02	1.38

Tables 7-10.Simulation results.

6. Conclusion and Future Work.

Several static and dynamic threshold schemes for Optical flow switching have been presented. Our goal was to design a scheme that would achieve the lowest delay in the network, while requiring the least amount of information about the state of the network. First, we came up with the analytical model for the IP/WDM system. In our analytical model we assumed that the distribution of the message size was known, which allowed us to derive a closed form solution for the optimal threshold which resulted in the lowest network delay. We also used Opnet to find optimal thresholds and minimal network delays as predicted by the simulation. Simulation results were close to the predicted analytical results and have shown that the static scheme would perform well if traffic conditions were fixed and message distributions were known. However, typically, the distribution of the message size and the load in the network are not known which makes the static scheme impractical for implementation in the real system.

In order to adjust to the changing traffic conditions in the network, we developed centralized and distributed dynamic schemes. In the centralized decision scheme, the switch makes the measurements of the average load on the IP and WDM queues and adjusts the threshold in order to balance the load between the two queues. We devised an algorithm that tries to keep the load on both queues balanced and quickly reacts to sudden changes of load. We believed that such an algorithm would keep the system in the stable region and achieve low message delays. Our simulation has shown that the centralized scheme achieved a delay that was close to the delay achieved by the static scheme, although the distribution of the message size was not known in the centralized scheme.

One major problem with the centralized decision scheme is that it relied on a single node to make all of the decisions. We devised a distributed scheme where each user makes a decision for itself based on the expected delay of its message through the network. The simulation has shown that the distributed scheme performed just as well or better than the centralized scheme in most cases.

In general, simulations have shown that both the centralized and the distributed schemes can achieve performance that is close to that of an optimal scheme where all of the parameters and loads are known in advance. Depending on the size of the network and other specific requirements, either of the schemes can be implemented to reduce the load on the IP routers. However, several other concerns, such as performance of the scheme in a multinode network [13] and fair allocation of the bandwidth to the requesting nodes, need further investigation. In the future, we plan to extend the current single-node model to a multinode system, so we can evaluate the delay between several switches. We also hope to evaluate various protocols for reserving a wavelength along the path between several IP/WDM switches, such as OBS [13]. In addition, we will evaluate various wavelength allocation schemes, such as one described in [14].

REFERENCES.

[1] P.Newman, G.Monshall, and T.Lyon, "IP switching - ATM under IP." *IEEE/ACM Transactions on Networking*, 6:117-129, April 1998

[2] Y.Rekhter, B.Davie, D.Katz, E.Rosen, G.Shallow. "Cisco Systems' Tag Switching Architecture Overview."IETF RFC 2105, Feb.1997

[3] A.Viswanathan, N.Feldman, R.Boivie, R.Woundy,"ARIS:Aggregate Route Based IP Switching." IETF Internat Draft, March 1997.

[4] K.Nagami, Y.Katsube, Y.Shobatake, A.Mogi, S.Matsuzawa, T.Jinmei, H.Esaki, "Toshiba's Flow Attribute Notification Protocol (FANP) Specification." IETF RFC 2129, April 1997

[5] R. Ramaswami, K. Sivarajan, *Optical Networking: A Practical Perspective*, Morgan Kaufmann, SF 1998

[6] E.Modiano, R.Berry, "Architectural Consideration in the Design of WDM-based Optical Access Networks," Computer Networks, Feb 1999

[7] S.Lin, N.McKeown, "A Simulation Study of IP Switching", Stanford University, 1998

[8] K.Claffy, H.Braun, and G.Polyzos, "A Parameterizable Methology for Internet Traffic Profiling", IEEE Journal of Selected Areas in Communications, 13 (8), Oct 1995

[9] J. Karol, M.G. Hluchyj and S.P. Morgan, "Input Versus Output Queueing in a Space-Division Packet Switch," IEEE Transactions on Communications, December, 1987

[10] D.Bertsekas,R.Gallager, "Data networks." Englewood Cliffs, Prentice-Hall, NJ, 1992

[11] D.M. Chiu and R. Jain, "Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks", Computer Networks and ISDN Systems, Vol. 17, pp. 1-14,1989.

[12] H.Kanakia, P.Mishra, A.Reibman. "An Adaptive Congestion Control Scheme for Real Time Packet Video Transport", IEEE/ACM Transactions on Networking, vol 3, No.6, December 1995

[13] C.Qiao, M.Yoo, "Optical Burst Switching (OBS) for IP Over WDM", SUNY at Buffalo, 1997

[14] O. Gerstel, G. Sasaki, R. Ramaswami, "Dynamic Channel Assignment for WDM Optical Networks With Little Or No Wavelength Conversion"