An Analysis of Deflection-Based Wormhole Routing with Virtual Channels[1]

Emmanouel A. Varvarigos and Jonathan P. Lang

University of California, Department of Electrical and Computer Engineering, Santa Barbara, CA 93106.

Abstract. We analyze a new wormhole routing scheme, which we call the wormhole deflection with virtual channels (abbreviated WDVC) scheme, that combines wormhole routing with virtual channels and deflection routing to provide efficient lossless communication. We use new analytical models to analyze the performance of the WDVC scheme for the Manhattan Street (MS) network topology. In particular, we examine the effect of the traffic load and the number of virtual channels on the throughput and the length of paths followed by the worms, and compare the analytical results obtained with corresponding simulation results. Our results indicate that wormhole deflections combined with virtual channels is efficient under both light and heavy traffic loads, especially when the number of virtual channels on a link is large.

1 Introduction

The efficiency in exchanging messages is an important performance parameter of parallel computing systems, and it relies heavily on the switching scheme used for interprocessor communication. Because of its low latency, wormhole routing has evolved as the desired scheme for multiprocessor networks and its performance has been analyzed by several researchers under a variety of models (see, for example, [DaS87], [KiD94], [BoD95], [Dua93], and [Dal92]).

Conventional wormhole routing allows messages (or worms) to be transmitted as a continuous stream of bits occupying multiple links simultaneously. This is accomplished by dividing each message into smaller message units called *flits* [DaS87], and advancing the flits from the input port to the output port at intermediate nodes, without waiting for the entire message to be received at the node. We distinguish between three types of flits in a message: the *header flit* used to route the message, the *data flits* used to carry the message information, and the *tail flit* used to signify the end of the message. When the header flit arrives at

[1] Research supported by ARPA under the MOST project.

an intermediate node, it is buffered until the output port can be determined and it can be transmitted on the associated link. When the data flits arrive, they are switched automatically to the same output port and transmitted immediately. When the header flit is blocked at an intermediate node, the flits that follow it are also blocked at their current position and the network resources are tied up until the contention is resolved. This scenario can lead to *deadlock*, where no worms can advance towards their destination, because all resources are occupied by worms waiting for resources to become available.

Virtual channels have been proposed to avoid deadlock and alleviate the waste of resources caused by blocked worms in the network ([DaS87], [Dal92]). A physical channel is shared by several virtual channels with parallel queues by time-division multiplexing the resources. Each worm being transmitted on a physical link occupies a virtual channel. When a worm is blocked, only the resources associated with the virtual channels used by the blocked worm are forced to remain idle, and only a portion of the network resources are wasted.

Deflection routing is an adaptive routing scheme that resolves network contentions by temporarily misrouting messages, as opposed to buffering them until the resources become available. As network congestion increases, messages are deflected away from the congested areas, thus balancing the load among the network links. Deflection routing has previously been studied by several researchers for store-and-forward routing (see for example, [Max89], [Max90], [GrG88], [GrH92], and [Bra91]), cut-through routing ([BFB93],[KoS94]), and wormhole routing ([LBB94], [LNG96]).

The wormhole deflection scheme with virtual channels (WDVC) that we propose and analyze in this paper is a wormhole routing scheme that combines the flexibility of virtual channels with the adaptability of deflection routing. Each physical channel consists of m virtual channels, with parallel queues at the receiving nodes. A preferred path is selected from the source to the destination of the message, and the header flit is sent on that path followed by the data flits. If there is no virtual channel available at an intermediate link to accommodate the worm, the worm may have to be routed over a different, longer path; we then say that the worm is *deflected.* As we will see later, for regular networks (with the number of input channels equal to the number of output channels) there are always adequate resources available to accommodate an incoming worm of an intermediate node. This, however, may happen at the expense of interrupting (preempting) existing worms that originate at that node.

We analyze the performance of the proposed scheme for the Manhattan Street (MS) network topology. In our model, all worms require one virtual channel, and their destinations are uniformly distributed over all nodes of the network. We obtain results on the throughput, the average number of deflections, and other performance parameters of interest as a function of the traffic load, the network size, and the number of virtual channels. Our analytical results are in close agreement with corresponding simulation results, and they indicate that the WDVC scheme uses the available resources very efficiently, especially when the number m of virtual channels on a link is large (say, 10). Our model is different than the ones used in previous works, where unslotted deflection schemes with cut-through [BFB93] and wormhole [LNG96] routing were studied via simulation. In our scheme, multiple worms can share the same link and unsuccessful worms are not dropped, but are rescheduled for transmission. Our analysis does not assume nodes to have any global information about the utilization of the network links, other than for their own outgoing links.

2 Wormhole Deflection with Virtual Channels Scheme

Messages (worms) are assumed to be generated at a source with a specified destination. A path is then computed at the source, and the source transmits the header flit, followed immediately by the data flits. As the header flit travels along the path, it may use any of the m virtual channels provided on each physical link. If the header flit is successful in finding a free virtual channel on all the links on the path to the destination, the WDVC scheme looks like the usual wormhole routing with virtual channels (except that in our scheme there are no constraints on the virtual channel chosen on each link). If, however, the header flit arrives at an intermediate node and is unable to obtain a virtual channel on the desired outgoing link of the path, an existing worm may have to be interrupted (preempted) or the arriving worm may have to be deflected, as described below.

We focus on a particular intermediate (i.e., non-source, non-destination) node, where the header flit of a transit worm arrives. If none of the virtual channels of the desired outgoing link are available for the transit worm, and one of the channels is used by a worm originating at this particular node, the originating worm is preempted (interrupted) to release the necessary virtual channel. When a worm is preempted, the source ceases transmitting new data and sends a tail flit to release the resources along the path to accommodate the new worm.

Worms that are interrupted may resume transmission when they see the tail flit of the worm that preempted them (either because the worm has propagated normally through the network, or because it has been preempted at its source). If none of the virtual channels of the desired outgoing link are occupied by originating worms, then the new worm is deflected to an alternate output link. It is possible that all of the virtual channels of the alternate link are occupied by existing worms, and preemption must be used to make a channel available for the deflected worm. Although it is possible for worms to preempt themselves, our simulations indicate that self preemptions happen infrequently.

Although deflection routing is deadlock-free (since worms are never blocked at intermediate nodes), livelock can occur if a worm circulates endlessly without reaching its destination. In addition, allowing worms to follow very long paths can waste network resources, increasing the probability that future worms will be blocked or will be forced to take even longer paths. To avoid livelock and the waste that occurs when a worm follows a very long path due to deflections, we may request that a worm is dropped when the header flit has traveled more than H hops without reaching its destination. The parameter H can be chosen to be equal to a multiple (e.g. 2 or 3 times) of the shortest distance between the source and the destination of the worm, and it may also be made to depend on the current congestion in the network.

3 Analysis for the Manhattan Street Network

We assume that the underlying network topology is a square Manhattan Street (MS) network [Max85], which is a two-connected regular mesh network with $\sqrt{N}$ nodes along each dimension and unidirectional communication links which alternate directions in adjacent rows [columns]. We assume worms are generated at each node over an infinite time horizon according to a Poisson process of rate λ, and their destinations are uniformly distributed over all nodes of the network. The worm lengths are independent and exponentially distributed with mean $1/\mu$. Each physical channel is divided into m virtual channels.

A worm using a given link l is called an *originating worm* if l is the first link on the worm's path, a *transit worm* if l is an intermediate link, and a *terminating worm* if it reaches its destination over link l. When both of the outgoing links of a node lie on a shortest path to the destination, then the node is called a *don't care* node for that destination; otherwise it is called a *preference* node. A transit worm arriving at an intermediate node selects a preferred outgoing link according to the *persistent rule*: If the current node is a "don't care" node, one of the links is chosen with equal probability as the preferred one. If the current node is a "preference" node, the preferred link is the one that lies on the shortest path.

A transit worm arriving at an intermediate node attempts to transmit on its preferred link, preempting if necessary an originating worm at that link. If this is not possible, the worm is routed over the other link of the node, preempting if necessary some worm originating on that link. An originating worm is accepted only if there is an available channel on its preferred link to accommodate it; that is, worms are never deflected on their first hop. An originating worm that is not accepted is said to be blocked, and must try to retransmit at a later time. A worm that is preempted, reattempts transmission after a random delay, and if it succeeds, continues transmitting data from the point at which the worm was interrupted. Worms that are preempted or blocked are randomly mixed back into the input queues so that the combined process of exogenous and retrial attempts can be approximated by a Poisson process.

We focus on worms with destination $(0,0)$, and let $\overline{D}(i,j)$ [or $D(i,j)$] be the average number of additional links that will be used by a transit [or originating, respectively] worm currently located at node (i,j).. We let p be the probability that an arriving transit worm fails to acquire a virtual channel on its preferred outgoing link (therefore, such a worm is deflected if the current node is a preference node). We then have

$$\overline{D}(i,j)=1+\begin{cases} \frac{1}{2}[\overline{D}(i_1,j_1)+\overline{D}(i_2,j_2)], & \text{if } (i,j) \text{ is don't care node;} \\ (1-p)\overline{D}(i_1,j_1)+p\overline{D}(i_2,j_2), & \text{if } (i,j) \text{ is preference node, and} \\ & (i_1,j_1) \text{ is preferred next node,} \end{cases} \tag{1}$$

and

$$D(i,j)=1+\begin{cases} \frac{1}{2}[\overline{D}(i_1,j_1)+\overline{D}(i_2,j_2)], & \text{if } (i,j) \text{ is don't care node;} \\ \overline{D}(i_1,j_1), & \text{if } (i,j) \text{ is preference node, and} \\ & (i_1,j_1) \text{ is preferred next node,} \end{cases} \tag{2}$$

where (i_1,j_1) and (i_2,j_2) are the outgoing neighbors of (i,j). Also, we clearly have $D(0,0)=\overline{D}(0,0)=0$. If the deflection probability p is known, the preceding equations can be applied iteratively on the MS network to calculate $D(i,j)$ and $\overline{D}(i,j)$ for all nodes (i,j). The total average number of links used by a worm can then be obtained as

$$D=\frac{1}{N-1}\sum_{(i,j)\neq(0,0)} D(i,j). \tag{3}$$

In what follows we present an analytical method for calculating the deflection probability p.

We denote by B the probability that a new worm is blocked at its source, and by E the probability that a worm is interrupted (preempted) before it completes transmission. We assume that the retransmissions of worms that are blocked or preempted are sufficiently randomized so that the total arrival rate of originating worms requesting a particular outgoing link of a node is a Poisson process with rate

$$\lambda_1^* = \frac{\lambda}{2(1-B)(1-E)}. \tag{4}$$

Since the average number of intermediate links used by a worm is equal to $D-1$, the average rate with which transit worms are emitted on a link is

$$\lambda_2 = \lambda_1^*(1-B)(D-1) = \frac{\lambda(D-1)}{2(1-E)}. \tag{5}$$

Also, the average rate with which terminating worms arrive at a node is $\lambda_3 = \lambda_1$.

We will say that a node is in state $\overline{X} = (X_a, X_b, X_c, X_d, X_{ab}, X_{ad}, X_{cb}, X_{cd})$, if there are X_a (or X_c) worms terminating over its horizontal (vertical, respectively) incoming link, X_b (or X_d) worms originating on its horizontal (vertical, respectively) outgoing link, X_{ab} (or X_{cd}) transit worms arriving over the horizontal (or vertical) incoming link and leaving over the horizontal (or vertical, respectively) outgoing link, and X_{ad} (or X_{cb}) transit worms arriving over the horizontal (or vertical) incoming link and leaving over the vertical (or horizontal, respectively) outgoing link. We also let $\pi(\overline{X})$ be the steady-state probability that a node is in state $\overline{X}$.

We will approximate $\pi(\overline{X})$ as the stationary distribution of an auxiliary system $\hat{Q}$, defined as follows. The system $\hat{Q}$ has four groups of servers (labeled Q_a, Q_b, Q_c, and Q_d), each of which has m identical servers and no waiting space. The groups Q_a and Q_c will be referred to as *incoming* groups, while the groups Q_b and Q_d will be referred to as *outgoing* groups of servers. We also refer to groups Q_a and Q_b as groups of the *top level*, and to groups Q_c and Q_d as groups of the *bottom level.* There are three types of customers, to be referred to as *originating*, *transit*, and *terminating* customers. Originating customers arrive at each outgoing group of servers (Q_b or Q_d) according to a Poisson process with rate λ_1^*. Transit and terminating customers arrive at each incoming group of servers (Q_a or Q_c) according to a Poisson process with rates λ_2^* and λ_3^*, respectively. Originating, transit, or terminating customers that find all servers in the group at which they arrive busy are dropped, never to appear again. An originating or terminating customer that is not dropped, obtains one server in the group at which he arrives. A transit customer that is accepted in an incoming group (Q_a or Q_c) obtains one server in the group of servers at which he arrives, and obtains an additional server in one of the outgoing groups (Q_b or Q_d), in the following way. The transit customer selects one of the outgoing groups as its preferred outgoing group, and then tries to obtain a server in that group. The preferred outgoing group is with probability θ the outgoing group that is at the same level (top or bottom) with the incoming group at which he arrived, and with probability $1-\theta$ it is the outgoing group that is at the other level from the incoming group at which he arrived. The parameter θ is taken to be equal to the average number of straight-through "care" nodes on a path in the MS network. Transit customers have preemptive priority over originating customers in Q_b and Q_d. That is, a transit customer that finds its preferred outgoing group of servers busy, can preempt an originating customer in that group. If all servers in that group are busy serving transit customers, it tries to obtain a server in the nonpreferred group, preempting if necessary an originating worm in that group. Once a transit customer is accepted in Q_a or Q_c, it is guaranteed to always find a server in Q_b or Q_d given the above preemption rule. Originating customers that are accepted in the system use a server for an exponential amount of time with mean $1/\mu$, unless they are preempted before the completion of their service. Transit and terminating customers that are accepted in the system leave the system after an exponential amount of time with mean $1/(\mu+\epsilon)$, where the parameter ϵ is taken to be the "probabilistic rate" at which a transit worm is preempted due to arrivals of header flits at its source, and will be defined precisely later.

We also ask that the rate λ_2 at which transit worms are emitted on a link of the MS network is the same as the rate at which transit customers are accepted in system $\hat{Q}$. For this to hold, we should have $\lambda_2^* = \lambda_2/(1-b)$, where $b = \Pr(\overline{X} : X_a + X_{ab} + X_{ad} = m)$. Furthermore, we ask that the rate λ_3 at which terminating worms are received at an incoming link of the MS network is the same as the rate at which terminating customers are accepted in system $\hat{Q}$. This happens when λ_3^* is defined as $\lambda_3^* = \lambda_3/(1-b)$. The probability that a new worm is blocked at its first hop is

$$B = \sum_{\overline{X}:\ X_b + X_{ab} + X_{cb} = m} \pi(\overline{X}). \tag{6}$$

Assuming that an arriving transit worm finds a node in a typical state, except for states where $X_a + X_{ab} + X_{ad} = m$, the deflection probability p is given by

$$p = \sum_{\substack{\overline{X}:\ X_{ab}+X_{cb}=m,\\ X_a+X_{ab}+X_{ad}\neq m}} \frac{\theta\pi(\overline{X})}{\sum_{\hat{X}:\ X_a+X_{ab}+X_{ad}\neq m}\pi(\hat{X})} + \sum_{\substack{\overline{X}:\ X_{ad}+X_{cd}=m,\\ X_a+X_{ab}+X_{ad}\neq m}} \frac{(1-\theta)\pi(\overline{X})}{\sum_{\hat{X}:\ X_a+X_{ab}+X_{ad}\neq m}\pi(\hat{X})}. \tag{7}$$

The probabilistic rate ϵ at which a particular transit worm $\mathcal{W}$ is preempted due to arrivals of transit worms at its source can be found to be

$$\epsilon = \lambda_2^*\Bigg[\sum_{\substack{\overline{X}:\ X_b\neq 0,\\ X_b+X_{ab}+X_{cb}=m,\\ X_a+X_{ab}+X_{ad}\neq m}} \frac{1}{X_b}\theta\frac{\pi(\overline{X})}{\sum_{\hat{X}:\ X_b\neq 0}\pi(\hat{X})} + \sum_{\substack{\overline{X}:\ X_b\neq 0,\\ X_b+X_{ab}+X_{cb}=m,\\ X_a+X_{ab}+X_{ad}\neq m,\\ X_{ad}+X_{cd}=m}} \frac{1}{X_b}(1-\theta)\frac{\pi(\overline{X})}{\sum_{\hat{X}:\ X_b\neq 0}\pi(\hat{X})}$$
$$+ \sum_{\substack{\overline{X}:\ X_b\neq 0,\\ X_b+X_{ab}+X_{cb}=m,\\ X_c+X_{cb}+X_{cd}\neq m}} \frac{1}{X_b}(1-\theta)\frac{\pi(\overline{X})}{\sum_{\hat{X}:\ X_b\neq 0}\pi(\hat{X})} + \sum_{\substack{\overline{X}:\ X_b\neq 0,\\ X_b+X_{ab}+X_{cb}=m,\\ X_c+X_{cb}+X_{cd}\neq m,\\ X_{ad}+X_{cd}=m}} \frac{1}{X_b}\theta\frac{\pi(\overline{X})}{\sum_{\hat{X}:\ X_b\neq 0}\pi(\hat{X})}\Bigg]. \tag{8}$$

The probability E that a worm $\mathcal{W}$ that has begun transmission is preempted before it is completed can be approximated as $E = \epsilon/(\epsilon + \mu)$.

To calculate the steady-state probabilities $\pi(\overline{X})$ for all feasible states, we write down the global balance equations of the Markov chain that corresponds to the auxiliary system $\hat{Q}$. If the parameters λ_1^*,λ_2^*, λ_3^*, and ϵ are known, then the global balance equations, together with the equation $\sum \pi(\overline{X}) = 1$ give the steady-state probabilities. These parameters, however, depend on the values of the steady-state probabilities. Equations (1)-(8) together with the global balance equations and the preemption probability E give a system of equations that can be jointly solved by using the method of successive approximations.

4 Analytical and Simulation Results

In this section, we present our results on the throughput, the average path length, and other performance parameters of interest for the WDVC scheme in a MS network topology. These results were obtained by solving numerically the analytical expressions given in Section 3. We also compare the analytical results with corresponding simulation results.

A natural measure of the performance of the WDVC scheme is the *inefficiency ratio* $\eta(\lambda)$, defined as the ratio $\eta(\lambda) = D(\lambda)/D(0)$ of the average path length $D(\lambda)$ taken by a worm for a given arrival rate λ, over the average shortest-path length $D(0)$ of the MS network topology. The inefficiency ratio characterizes the effectiveness with which the WDVC scheme uses the network capacity for a given network load. In Fig. 1 we illustrate $\eta(\lambda)$ as a function of the external arrival rate λ (measured in worms per node per unit of time), for a 6×6 MS network, average worm length $1/\mu = 1$, and the number of virtual channels $m = 1$ and $m = 2$. We also illustrate the deflection probability p at a preference node.

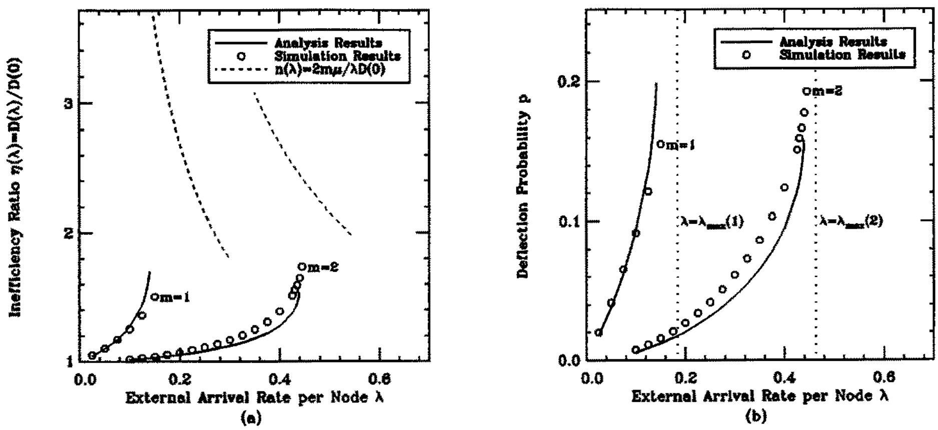

Fig. 1. We illustrate the inefficiency factor $\eta(\lambda)$ and the deflection probability p, as a function of the external arrival rate per node λ, for a 6×6 MS network with $m = 1$ and $m = 2$ virtual channels per link. We also illustrate the upper bound $\frac{2m\mu}{\lambda D(0)}$ on $\eta(\lambda)$.

The analytical model of Section 3 assumes that worms that are blocked or preempted reattempt to establish a connection after a random delay, and that the network is operating in the stable region, where the preemption, deflection, and blocking probabilities are strictly less than one; our simulations show that the preemption probability E is the first of these probabilities to approach one. To obtain a necessary condition for stability, note that external messages are generated in the network at a total rate of λN worms per unit of time, and each of them uses on the average $D(\lambda)$ links. Since the total network capacity is $2Nm$, a necessary condition for stability is $\lambda N D(\lambda)/\mu \leq 2Nm$, or, equivalently, $\eta(\lambda) \leq \frac{2m\mu}{\lambda D(0)}$.

In order to investigate the behavior of the network in the unstable region, where the external arrival rate λ per node is larger than what the network can sustain, we have to rely on simulations. Figure 2 illustrates simulation results for the inefficiency ratio and deflection probability as a function of the normalized external arrival rate λ/m, so that the curves corresponding to different numbers of virtual channels m can appear on the same plot. In Fig. 3, we plot the normalized throughput, that is, the average number of worms per node, per unit of time, and per virtual channel, successfully transmitted without interruptions. Using arguments similar to those used to determine the stability condition, an upper bound on the maximum normalized throughput is given by $\lambda_{max}/m = 2\mu/D(0)$. Note that deflection routing with no virtual channels (case $m = 1$) has a maximum throughput of only 35% of this upper bound. However, a linear increase in the number of virtual channels m corresponds to a better-than-linear increase in the network throughput (for $m = 10$ virtual channels, the maximum throughput is nearly 80% of the upper bound), indicating that the more virtual channels per physical channel, the more efficient is the operation of the WDVC scheme. Also, note that the deflection probability p, the inefficiency ratio $\eta(\lambda)$, and the throughput all increase with λ but eventially reach a plateau. In fact, the load at which the plateau is reached coincides with the load at which the preemption probability E becomes equal to one. This indicates that at heavy traffic load, preemptions act as a "built-in" flow control mechanism that prevents the deflection probability and the average path length from increasing beyond some point. Leonardi et al [LNG96] similarly observed that an input rate control policy was necessary to prevent the throughput from decreasing after reaching a maximum.

Borgonovo et al [BFB93] used simulations to analyze an unslotted deflection scheme with cut-through routing. The throughput results that we obtained for the case $m = 1$ are similar to the results obtained in [BFB93] for $\tau \approx 0$ (note that the results in [BFB93] are given in terms of throughput versus offered traffic, and blocked messages do not retry to enter the network). This was expected, because if we view a packet of variable length in [BFB93] as a worm (this corresponds to $\tau \approx 0$ in [BFB93]), and do not allow sharing of links by multiple worms (this corresponds to $m = 1$ in the WDVC scheme), the efficiency with which the two schemes use capacity should be similar. The analogy between the two cases, however, is lost when $m \neq 1$, since the preemption and splitting of worms, which are necessary for the WDVC scheme to work, play no role in the unslotted deflection scheme of [BFB93].

Our results indicate that the WDVC protocol becomes more efficient as the number of virtual channels increase. However, it is possible to increase the per-

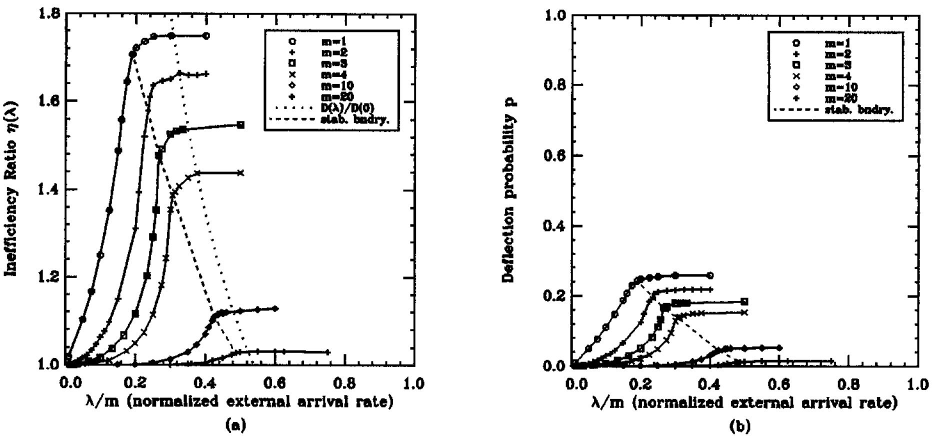

Fig. 2. We illustrate the simulation results for the inefficiency factor $\eta(\lambda)$ and the deflection probability p, as a function of λ/m for a 6×6 MS network, and several values of m. We also illustrate the upper bound $\frac{2m\mu}{\lambda D(0)}$ on $\eta(\lambda)$, and the stability region.

formance of the WDVC protocol (and avoid livelock as mentioned earlier) by restricting the number of hops taken by worms. Figure 3b illustrates simulation results for the normalized throughput obtained for the case where the length of the path followed by a worm is restricted to be at most h times the shortest distance between the source and the destination of the worm. These results indicate that a 35% increase in maximum throughput is possible for $m = 2$ virtual channels and $h = 2$, exceeding the performance for $m = 4$ virtual channels with no path restriction.

Reference

[BFB93] Borgonovo, F., Fratta, L., and Bannister J., "Unslotted Deflection Routing in All-Optical Networks," IEEE GLOBECOM'93, Houston, TX, pp. 119-125, Nov. 1993.

[Bra91] Brassil, J. T., Deflection Routing in Certain Regular Networks, Ph.D. Thesis, UC San Diego, 1991.

[BoD95] Boura, Y. M., and Das C. R., "Modeling virtual channel flow control in hypercubes," in Proc. of 1^{st} IEEE Symp. on HPCA, pp. 166-175, CA, Jan. 1995.

[Dal92] Dally, W. J., "Virtual-Channel Flow Control," IEEE TPDS, Vol. 3, No. 2, pp. 194-205, March 1992.

[DaS87] Dally, W. J., Seitz, C. L., "Deadlock-Free Message Routing in Multiprocessor Interconnection Networks," IEEE Trans. on Comp., Vol. C-36, No. 5, pp. 547-53, 1987.

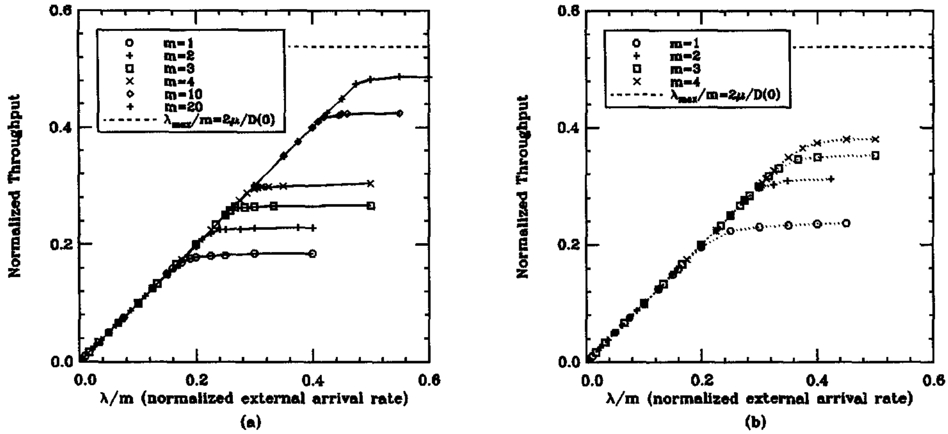

Fig. 3. The normalized throughput is shown for $h = \infty$ (a) and $h = 2$ (b) as a function of λ/m for a 6×6 MS network and several values of m. We also illustrate the upper bound $\frac{2\mu}{D(0)}$ on the throughput.

[Dua93] Duato, J., "A New Theory of Deadlock-Free Adaptive Multicast Routing in Wormhole Networks," IEEE TPDS, Vol. 4, No. 12, pp. 1320-1331, Dec. 1993.

[GrG88] Greenberg, A. G., and Goodman, J., "Sharp Approximate Models of Adaptive Routing in Mesh Networks," Teletraffic Anal. and Comp. Perf. Eval., pp. 255-270, 1988.

[GrH92] Greenberg, A. G., and Hajek, B., "Deflection Routing in Hypercube Networks," IEEE Trans. Communications, Vol. COM-35, No. 6, pp. 1070-1081, June 1992.

[KiD94] Kim, J., Das, C./ R., "Hypercube Communication Delay with Wormhole Routing," IEEE Trans. on Computers, Vol. 43, No. 7, pp. 806-814, July 1994.

[KoS94] Konstantinidou, S., Snyder, L., "The Chaos Router," IEEE Trans. on Computers, Vol. 43, No. 12, pp. 1386-1397, Dec. 1994.

[LNG96] Leonardi, E., and Neri, F., Gerla, M., Palnati, P., "Congestion Control in Asynchronous, High-Speed Wormhole Routing Networks," IEEE Communications magazine, Vol. 34, No. 11, pp. 58-69, Nov. 1996.

[LBB94] Lu, Y., Bagchi, K., Burr, J. B., Peterson, A. M., "A Comparison of Different Wormhole Routing Schemes," Proc. of the 2^{nd} Intl. Work. on Modeling, Anal., and Simulation of Comp. and Telecom. Sys., pp. 323-8, Durham, NC., Feb. 1994.

[Max85] Maxemchuk, N. F., "The Manhattan Street Network," Proc. of GLOBECOM'85, pp. 255-261, New Orleans, LA., Dec. 1986.

[Max89] Maxemchuk, N. F., "Comparison of Deflection and Store-and-Forward Techniques in the Manhattan Street and Shuffle-Exchange Networks," in INFOCOM '89, Vol. 3, pp. 800-809, Ontario, Canada, April 1989.

[Max90] Maxemchuk, N. F., "Problems Arising from Deflection Routing: Live-lock, Lock-out, Congestion and Message Reassembly," Proc. NATO Workshop on Archit. and High Perform. Issues of High Capacity LANs and MANs, Denver, CO., 1990.