Performance of Multihop Communications Using Logical Topologies on Optical Torus Networks [*]

X. Yuan, R. Melhem and R. Gupta
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260
{xyuan, melhem, gupta}@cs.pitt.edu

Abstract

We consider multihop communications on optical torus networks with time–division multiplexing where logical topologies are realized on top of the physical network to improve the communication performance. The logical topologies reduce the number of intermediate hops at the cost of a larger multiplexing degree. On the one hand, the larger multiplexing degree increases the packet communication time between hops. On the other hand, reducing the number of intermediate hops reduces the time spent at intermediate hops. We study the trade–off between the multiplexing degree and the number of intermediate hops. Specifically, we study four logical topologies ranging from the most dense logical all–to–all connections to the simplest logical torus topology on top of physical torus networks. We develop an analytical model that models the maximum throughput and the average packet delay of the multihop networks, verify the model through simulations, and study the performance and the impact of system parameters on the performance for these four topologies.

1 Introduction

In optical multihop networks, intermediate hops are responsible for routing packets to their destinations. Hence, optical signals must be converted into the electronic domain and processed at the intermediate nodes to make the routing decisions. Since the electronic processing speed is relatively slow in comparison to the optical data transmission speed, it is important to reduce the number of hops that a packet visits in an optical multihop network. In an optical time-division-multiplexed (TDM) network, multiple virtual channels are supported on each optical link. Reducing the number of intermediate hops in such systems can be achieved by routing packets through a more efficient logical topology, as opposed to routing packets through the physical topology.

In this paper, we study multihop communications on top of physical torus networks with TDM. We chose the torus topology as our underlying physical topology because it has nice characteristics, such as a fixed number of ports, good scalability, and it is currently used by many commercial supercomputers. We study four logical topologies. The first logical topology establishes connections from each node to every other node, which results in single–hop communication. It represents an extreme case where the number of intermediate hops is traded in favor of the multiplexing degree. This topology will be called the logical *alltoall* topology. The second topology is the logical *torus* topology, which has the same topology as the physical network. This represents another extreme case where the multiplexing degree is traded in favor of the number of intermediate hops. The other two logical topologies lie in between these two extremes. The third topology is a 1-hop system. It is formed by having all–to–all connections along each dimension of the torus. Thus, a packet passes at most 1 intermediate hop to reach its destination. We will call this topology the *allXY* topology. The fourth topology is a logical *hypercube* topology. We will discuss these topologies in details in section 3.

We develop an analytical model for the maximum throughput and average packet delay for the four topologies. We then verify the analytical model with simulations and study the impact of system parameters, such as the packet routing time, on the performance of these topologies. We found that in terms of maximum throughput, the $alltoall$ topology is the best. However, the $alltoall$ topology incurs large packet delay even when the network is under light load. In terms of packet delay, the logical torus is good only when the network is under light load and the router is very

[*] This work was supported in part by NSF award CCR-9704350 and MIP-9633729.

fast, while the logical $alltoall$ topology out–performs the other topologies only when the network is almost saturated (under very high load). In general, the logical hypercube and $allXY$ topologies have smaller average packet delay than the torus and $alltoall$ topologies.

The rest of the paper is organized as follows. In section 2, we describe the network architecture for multihop communication. Section 3 introduces the logical topologies and analyzes the performance of these topologies under light load. In section 4, we present the analytical model that takes network contention into consideration and verify the model with simulations. Section 5 studies the performance of the logical topologies. Section 6 concludes the paper.

2 Network Architecture

An optical torus network consists of 5 × 5 switches connected with optical links. All but one input port and one output port of a switch are used to interconnect with other switches, while one input port and one output port are used to connect to a local node (router and/or local PE). See Figure 1 (a). We assume that all links in the system support same number of virtual channels. The number of channels that can be supported concurrently on a link is called the *multiplexing degree*. Connections in the logical topology, which may span a number of links and switches, are all–optical paths (lightpaths) realized using *path multiplexing* [6]. That is, the same channels on all links along a path are used for the connection.

By using path multiplexing, efficient logical topologies can be established on top of the physical topology. In such systems, the switching architecture consists of an optical component and an electronic component. The optical component is an all–optical switch, which can switch the optical signal from input channels to output channels in the optical domain (i.e., without electronic/optical (E/O) and optical/electronic (O/E) conversions), and which can locally terminate some other lightpaths by directing them to the node's electronic component. The electronic component is a store–and–forward packet router overlaid on top of the optical virtual topology. We assume that each router contains a *routing buffer* that buffers all incoming packets. For each packet, the router determines whether to deliver the packet to the local PE or to the next path toward the packet destination. A separate *output path buffer* is used for each outgoing path that buffers the packets to be sent on that path and thus accommodates the speed mismatch between the electronic router and the optical path. Figure 1 (b) depicts the structure of a router. Note that the output paths are multiplexed in time over the physical link that connect the local PE to its corresponding switch.

In the rest of the paper, we will use R_delay to denote the time a packet spends in the routing buffer and the time for the router to make a routing decision for the packet (packet routing time). We will use T_delay to denote the time a packet spends on the path buffer and the time it takes for the packet to be transferred on the path.

3 Logical topologies and their performance under light load

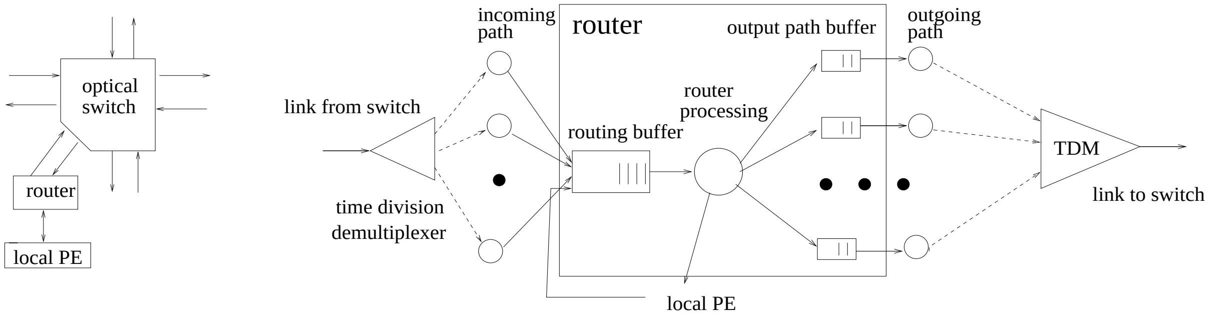
Next we introduce the logical topologies, describe how these logical topologies are realized on top of the torus topology, and discuss how the logical topologies affect the average packet delay assuming that the network is under light load and therefore the network contention is negligible. We later develop an analytical model that takes the effect of network contention into consideration.

For a given logical topology, let h be the average number of intermediate hops and d be the multiplexing degree required to realize the logical topology. Given an $N \times N$ torus, the logical $alltoall$ topology establishes direct connections between all pairs of nodes and thus, totally eliminates intermediate hops, resulting in $h = 0$. Optimal algorithms (with respect to the multiplexing degree) to realize the $alltoall$ connections on ring and torus topologies can be found in [3]. Using an algorithm in [3], a multiplexing degree of $d = \frac{N^3}{8}$ can be used to realize the logical $alltoall$ topology.

Given an $N \times N$ torus, a logical torus topology can be realized using a multiplexing degree of $d = 4$. Notice that although the logical paths are the same as the physical links, the logical torus topology cannot be realized using a multiplexing degree of 1 due to the contention on the links connecting local PEs to switches. Specifically, in one time slot each router can only access one channel, and since each router has four outgoing logical paths, one to each neighbor, a multiplexing degree of 4 is needed to realize the logical torus topology. For a logical $N \times N$ topology, the average number of intermediate hops is $h = \frac{N}{2} - 1$.

For $N = 2^r$, the algorithm in [8] realize a logical hypercube topology on an $N \times N$ torus using a multiplexing degree of $\lfloor \frac{N}{3} + \frac{N}{4} \rfloor + 2$, if r is odd, and $\lfloor \frac{N}{3} + \frac{N}{4} \rfloor + 1$, if r is even. For a logical N^2 node hypercube, the average number of intermediate hops is $h = \frac{lg(N^2)}{2} - 1 = lg(N) - 1$.

Finally, let us consider the logical *allXY* topology. By using the 1-dimension communication patterns for all–to–all connections on rings from [3] and mixing the 1-dimensional communication patterns to form 2-dimensional patterns for the $allXY$ topology, it can be shown that when $N \leq 8$, the logical topology can be realized using a multiplexing degree of $2N - 2$. For $N > 8$, the same multiplexing degree needed to realize the all–to–all connections on an N–node ring can be used to realize the *allXY* communication on $N \times N$ torus. Thus, a multiplexing degree of $\frac{N^2}{8}$ can



(a) a nodal switching architecture

(b) the router

Figure 1. The network components

be used to realize the $allXY$ topology. Since for two nodes in the same column or row, no intermediate hop is needed, while in other cases, one intermediate hop is required, the average number of intermediate hops on the logical $allXY$ topology is given by:

$$\frac{2N-2}{N^2-1} \times 0 + \frac{(N^2-1)-(2N-2)}{N^2-1} \times 1 = \frac{N^2-2N+1}{N^2-1}.$$

Table 1 summarizes the average number of intermediate hops (h), the multiplexing degree (d) and the total number of logical connections (P) for the four topologies.

	No. of inter. hops (h)	multipl. degree (d)	total no. of path (P)
$alltoall$	0	$\frac{N^3}{8}$	$N^2(N^2-1)$
$allXY$	$\frac{N^2-2N+1}{N^2-1}$	$\frac{N^2}{8}$ †	$N^2(2N-2)$
hype.	$lg(N)-1$	$\lfloor \frac{N}{3} + \frac{N}{4} \rfloor + 1$ ‡	$N^2 lg(N)$
torus	$\frac{N}{2}-1$	4	$N^2 \times 4$

† Here,we assume that $N > 8$. If $N < 8$, the value is $2N-2$.
‡ Here, we assume that r is even. If r is odd, the value is $\lfloor \frac{N}{3} + \frac{N}{4} \rfloor + 2$.

Table 1. Summary of logical topologies

Let us now consider the communication performance of these topologies when the network is under light load. Assume that a packet can be transferred from source to destination on a path in one time slot and that the network has a multiplexing degree of d. If a packet arrives at a router randomly, then it takes on an average $\frac{d+1}{2}$ time slots to transfer a packet from a router to the next router. Thus, assuming that the packet routing time in each router (including the E/O and O/E conversions) is γ, and the network contention is negligible, the average delay time for each packet can be expressed as follows:

$$delay = (h+2) * \gamma + (h+1) * \frac{d+1}{2}.$$

The first term, $(h+2) * \gamma$, is the average routing time that a packet spends at the h intermediate routers and the 2 routers at the sending and receiving nodes. The second term, $(h+1) * \frac{d+1}{2}$, is the average packet transmission time on paths plus the time that a packet waits in the output path buffers. The average delay time is determined by three parameters, the multiplexing degree d, the packet routing time γ, and the average number of hops per packet transmission h. By replacing h and d by the values in Table 1, we can obtain the average packet delay in terms of γ and N for each logical topology (see Table 2). As shown in Table 2, the packet delay for topologies with less connectivity (torus, hypercube) is affected more by the router speed, γ, while for topologies with high connectivity ($allXY$ and $alltoall$), the packet delay is affected more by the network size.

topology	delay
torus	$O(N\gamma)$
hypercube	$O(\gamma lg(N) + Nlg(N))$
$allXY$	$O(N^2+\gamma)$
$alltoall$	$O(N^3+\gamma)$

Table 2. Average delay under light load

4 An analytical model and its verification

In this section, we will describe an approximate analytical model that takes network contention into consideration. We use this model to study the effect of the network load on the maximum throughput and the packet delay. We assume that in each time slot, a packet can be sent from the source to the destination on a path. For example, if a 1Gbps channel is used with a 53–byte packet (or cell) as defined in the ATM standard, then the slot duration is $0.424\mu s$. All other delays in the system are normalized with respect to this slot duration.

We model the routers and the paths in a network as a network of queues. As shown in Figure 1, each router has a routing queue that buffers the packets to be processed.

The router places packets either into one of the output path queues that buffer packets waiting to be transmitted, or into the local processor. Both a router and a path have a constant service time. The exact model for such network is very difficult to obtain. We approximate the network by making the following assumptions: 1) each queue is independent of each other, and 2) each queue has a Poisson arrival and a constant service time. These assumptions enable us to derive expressions for the maximum throughput and the average packet delay of the four logical topologies by dealing with the M/D/1 queues independently. Our simulation results confirm that these approximations are reasonable. We use the following notations in the model:

- N. Size of each dimension of the torus. Thus, the network has a total of N^2 nodes.
- d, h and P are defined in the previous section. A *frame* consists of d time slots. Within a frame, one time slot is allocated to each path.
- λ. Average packet generation rate at each node per time slot. This implies that the average generation rate of packets to the entire network is $N^2\lambda$. We assume that the arrival process is Poisson and is independently and identically distributed on all network nodes. Furthermore, we assume that all packets are equally likely to be destined to any one of the network nodes. At each router, the newly generated packets and the packets arriving from other nodes are maintained in an infinite routing buffer before being processed as shown in Figure 1.
- λ_s. Average rate of packet arrival at a router per time slot, including both generated packets and packets received from other nodes. This composite arrival rate, λ_s, may be derived as follows. In any time slot the total number of generated packets that arrive at all the routing buffers is λN^2. On average, each of these packets traverses $h+2$ routers within the network. Therefore, under steady state condition, there will be $\lambda N^2(h+2)$ packets in all the routers of the network in each time slot. Under the assumption that each packet is equally likely to be in each router, the total arrival rate is given by $\lambda_s = \lambda(h+2)$.
- λ_p. Average rate of packet arrival at a path buffer per time slot. This arrival rate, λ_p, can be derived as follows. Under steady state condition, in any time slot, the total number of packets in all the routers in the network is $\lambda N^2(h+2)$. Of all these packets, λN^2 packet will exit the network and $\lambda N^2(h+2) - \lambda N^2 = \lambda N^2(h+1)$ packets will be transmitted through paths in the network. Under the assumption that sources and destinations are uniformly distributed in the network, the average arrival rate is given by $\lambda_p = \frac{\lambda N^2(h+1)}{P}$.
- γ. The routing time per packet at a router. Since packets are of the same length, the routing time is a constant value. The average packet departure rate from the routing buffer, denoted by μ_s, is $\mu_s = \frac{1}{\gamma}$.
- μ_p. The average packet departure rate from each path buffer per time slot. Since in our model, each path will be served once in every frame, $\mu_p = \frac{1}{d}$. The average service time in each path is $S_p = \frac{1}{\mu_p} = d$.

Maximum throughput

With the above notation, we can now study the maximum throughput and average packet delay of the logical topologies. We will first study the theoretical maximum throughput and then the average packet delay. Two bottlenecks can potentially limit the maximum throughput.

- If the average packet arrival rate at a routing buffer is larger than the average packet departure rate, that is if $\lambda_s \le \mu_s$, then the throughput will be limited by the router processing bandwidth. The maximum packet generation rate allowed by the router bandwidth, λ_s^{max}, can be derived as follows: $\lambda_s \le \mu_s$, or $(h+2)\lambda \le \frac{1}{\gamma}$, or $\lambda \le \frac{1}{\gamma(h+2)}$. Thus,

$$\lambda_s^{max} = \frac{1}{\gamma(h+2)}.$$

- If the average packet arrival rate at a path buffer is larger than the average packet departure rate, that is $\lambda_p \le \mu_p$, then the throughput will be limited by the path bandwidth. The maximum fresh packet generation rate allowed by the path bandwidth, λ_p^{max}, can be derived as follows: $\lambda_p \le \mu_p$, or $\frac{(h+1)\lambda N}{P} \le \frac{1}{d}$, or $\lambda \le \frac{P}{(h+1)Nd}$. Thus,

$$\lambda_p^{max} = \frac{P}{(h+1)Nd}.$$

The theoretical maximum throughput is the minimum of λ_s^{max} and λ_p^{max}, that is, $\lambda^{max} = min(\lambda_s^{max}, \lambda_p^{max})$. Given a topology, $\lambda^{max} = \lambda_s^{max}$ indicates that the router speed is the bottleneck, while $\lambda^{max} = \lambda_p^{max}$ indicates that the path speed is the bottleneck.

Average packet delay

As mentioned in section 2, we divide packet delay into the *routing delay*, which includes the time a packet spends on routing buffers and the time for routers to process the packets, and the *transmission delay*, which includes the time a packet spends on path buffers and the actual packet transmission time on the paths.

Let us first consider the routing delay in each router. It takes γ timeslots for a router to process the packet when the packet reaches the front of the routing buffer. As for the packet waiting time in the routing buffer, since we model the routing buffer as an $M/D/1$ queue, the average queuing delay depends on the arrival rate λ_s and is given by:

$$Q = \frac{\lambda_s(\gamma)^2}{2(1-\frac{\lambda_s}{\mu_s})}$$

where λ_s is the average packet arrival rate, γ is the expected service time, and μ_s is the average packet departure rate. Given that $\mu_s = \frac{1}{\gamma}$, the total time that a packet spends in each router is given by:

$$R\_delay = \gamma + \frac{\lambda_s(\gamma)^2}{2(1-\lambda_s\gamma)} \qquad (1)$$

Consider the two components of the transmission delay on each path. The first component is the delay required by a packet to synchronize with the appropriate outgoing slot in the frame on which the node transmits and the actual packet transmission time. The average value of this delay is $\frac{1+2+\ldots+d}{d} = \frac{d+1}{2}$. The second component is the $M/D/1$ queuing delay that a packet experiences at the buffer before it reaches the head of the buffer. This follows the same formula as in the routing delay case, and is given by:

$$\frac{\lambda_p S_p^2}{2(1-\frac{\lambda_p}{\mu_p})} = \frac{\lambda_p d^2}{2(1-\lambda_p d)}$$

Combining the two components, we obtain the total delay a packet encounters on a path,

$$T\_delay = \frac{d+1}{2} + \frac{\lambda_p d^2}{2(1-\lambda_p d)} \qquad (2)$$

Each packet takes $h+2$ hops and $h+1$ paths on average. Thus, given that on average, a packet spends R_delay in each router and T_delay on each path, the average packet delay can be expressed as follows:

$$delay = (h+2) \times R\_delay + (h+1) \times T\_delay$$

Using formula (1) and (2), we obtain the following average delay that a packet encounters from the source to the destination.

$$\begin{aligned} delay = (h+2) \times (\gamma + \tfrac{\lambda_s(\gamma)^2}{2(1-\lambda_s\gamma)}) \\ +(h+1) \times (\tfrac{d+1}{2} + \tfrac{\lambda_p d^2}{2(1-\lambda_p d)}) \end{aligned}$$

Model verification

To verify our analytical model and to further study the performance of these logical topologies, we developed a network simulator that simulates all four logical topologies on top of the torus topology. The simulator takes the following parameters.

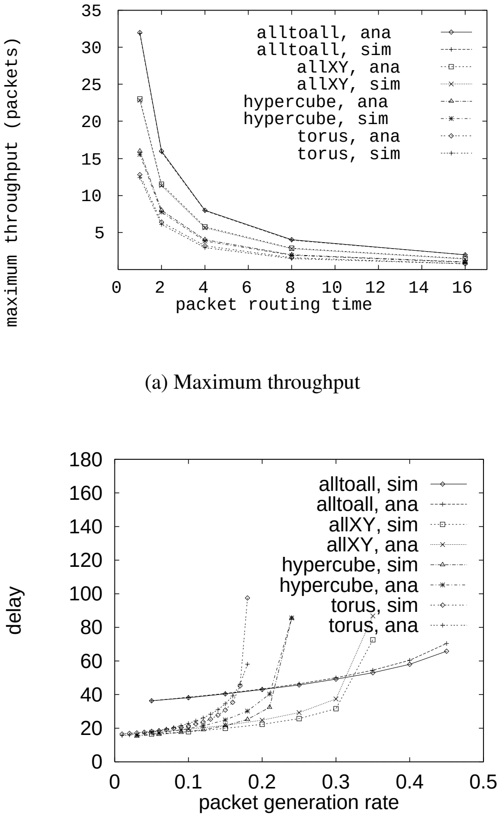
- *system size*, $N \times N$: This specifies the size of the network. Based on the logical topology, the system size also determines the multiplexing degree in the system.
- *packet generation rate*, λ: This is the rate at which fresh packets are generated at each node. It specifies the traffic on the network. The inter–arrival of packets follows a Poisson distribution. When a packet is generated at a node, the destination is generated randomly among all other nodes in the system with a uniform distribution.
- *Packet routing time*, γ.

Fig 2 shows the maximum throughputs and the average packet delays obtained from the analytical model and from simulations for an 8×8 torus. The packet routing time is assumed to be 1 timeslot in the study of the average packet delay. As can be seen from the figure, the analytical results and the simulation results match quite well for all the cases. Networks of different size and/or different packet routing time have also been studied. The results are similar to those in Fig 2.

5 Performance of the logical topologies

In the previous section, we developed an analytical model for performance study for the logical topologies and compared the results of the model with those of simulations. In this section, we focus on studying the performance of the logical topologies. Since the simulation and the analytical model match reasonably well, we will only use the analytical model in this section to study the performance.

Figure 3 shows the impact of packet routing time on the maximum throughput. The underlying topology is a 32×32 torus. As can be seen from the figure that the $alltoall$ topology achieves higher maximum throughput than the $allXY$ topology, which in turn achieves higher maximum throughput than the hypercube topology. The logical torus has the worst maximum throughput. This observation holds for all packet routing speeds. Under high workload, all paths in the $alltoall$ and $allXY$ topologies are utilized. The algorithms to realize the $alltoall$ and $allXY$ topologies guarantee that in each time slot all links are used if all connections scheduled for that time slot are in use, while the hypercube and



(a) Maximum throughput

(b) Average packet delay

Figure 2. Analytical and simulated performance for 8×8 torus

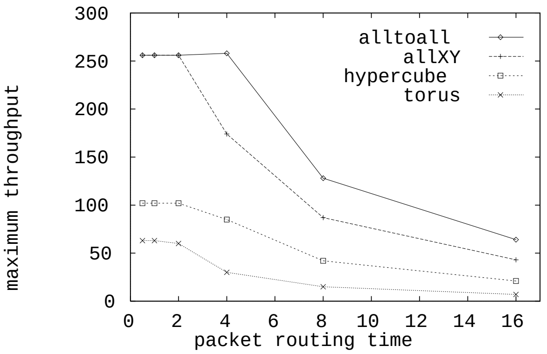


Figure 3. Maximum throughput .vs. packet routing time $(N = 32)$

torus topologies can not achieve this effect. Thus, it is expected that the $alltoall$ topology and the $allXY$ topology will outperform the hypercube and torus topologies in terms of maximum throughput.

Figure 4 shows the impact of network size on the maximum throughput. The results in this figure are based upon a packet routing time of one time slot. We also studied different packet routing times and found similar trends. In terms of maximum throughput, the $alltoall$ topology scales the best, followed by the $allXY$ topology, followed by the hypercube topology. The logical torus topology scales worst among all these topologies. Figures 3 and 4 show that by using time–division multiplexing to establish complex logical topology, we can exploit the large aggregate bandwidth in the network and deliver higher throughput when the network is under high workload.

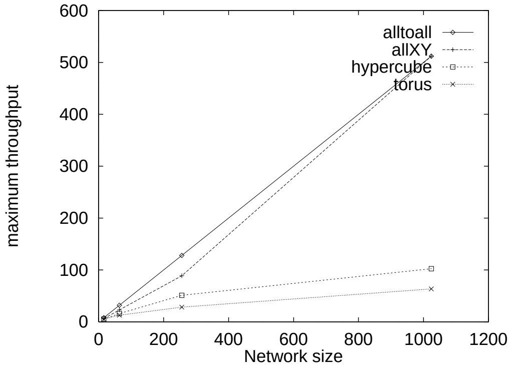


Figure 4. Maximum throughput .vs. network size ($\gamma = 1$)

The average packet delay is another performance metric to be considered. For a network to be efficient, it must be able to deliver packets with a small delay. It is well known that TDM results in larger average packet delay due to the sharing of the links. However, as we have discussed earlier, while using TDM techniques to establish logical topologies increases the per hop transmission time, it reduces the average number of hops that a packet travels. Thus, the overall performance depends on system parameters. Next, we will study this effect for the logical topologies.

Figure 5 shows the delay with regard to the packet generation rate. The underlying topology is a 16×16 torus. We also assume that γ is 1 time slot. As we can see from the figure, the $alltoall$ topology incurs very large delay compared to other logical topologies, this is because of the large multiplexing degree needed to realize the logical $alltoall$ topology. Other topologies have similar delay when the generation rate is small (low workload). However, the $allXY$ topology has a larger saturation point than the hypercube and torus topologies, and thus has a small delay even when the network load is reasonably high (e.g. $\lambda = 0.25$). These results also hold for larger packet routing times.

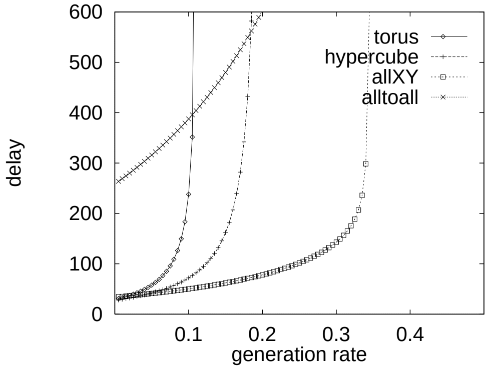


Figure 5. Packet delay as a function of packet generation rate $(\gamma = 1.0, N = 16)$

Figure 6 shows the impact of packet routing time on the average packet delay. The results are based upon a 16×16 torus network and a packet generation rate of 0.005. The packet routing speed has an impact on the delay for all topologies. For very small packet routing time ($\gamma = 0.25$), the torus topology has the smallest delay. When the packet routing time increases, the delay in torus increases drastically, while the delays in the $alltoall$ and $allXY$ topologies increase slightly. In the $alltoall$ and $allXY$ topologies a packet travels through fewer number of routers than it does in the torus topology. Hence the contention at routers does not affect the delay in the $alltoall$ and $allXY$ topologies as much as it does in the torus and hypercube topologies.

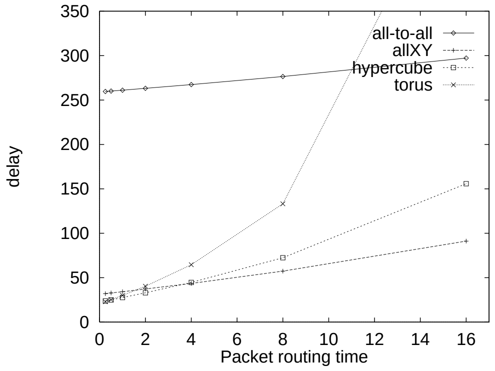


Figure 6. Impact of packet routing time on packet delay ($\lambda = 0.005, N = 16$**)**

Figure 7 shows the impact of network size on the packet delay for the topologies. The results are based upon a packet routing time of 1 time slot and a packet generation rate of 0.01. This figure shows the manner in which the delay time grows with the network size. This figure shows similar trends as the results in section 3. The $alltoall$ topology has very large delay when the network size is large. The delay differences among the other three topologies are relatively small for reasonably large sized networks.

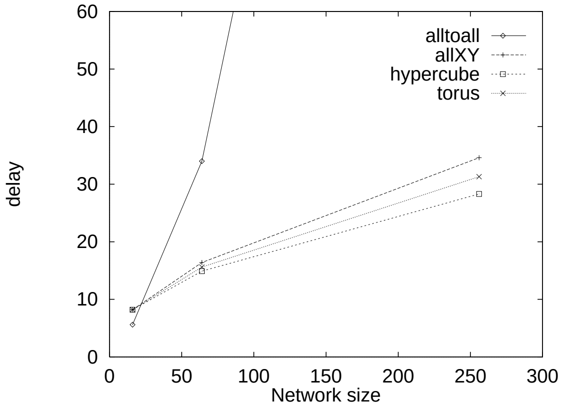


Figure 7. impact of network size on the delay ($\lambda = 0.01$**)**

Three parameters, N, γ and λ affect the average packet delay for all the logical topologies. Next, we will identify the regions in the (N, γ, λ) parameter space, where a logical topology has the lowest packet delay. Figure 8 shows the best topologies in the parameter space (N, γ) with $\lambda = 0.01$. As can be seen from Figure 8, for a given λ, all four logical topologies occupy part of the (N, γ) parameter space. Topologies with higher connectivity suffer less from the router contention and thus, offer higher performance when the router is slow, while topologies with less connectivity require less multiplexing degree, and are better when the router is fast.

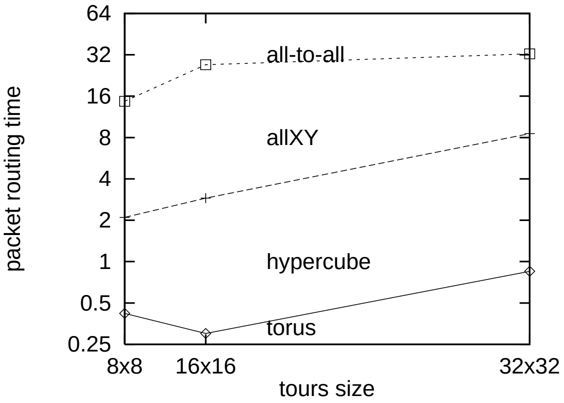


Figure 8. Best logical topology for $\lambda = 0.01$

Figure 9 shows the best logical topologies on the (γ, λ) parameter space for a physical 16×16 topology. Networks of different sizes exhibit similar characteristics. The majority of the (γ, λ) parameter space is occupied by the logical hypercube and $allXY$ topologies. The logical torus topology is good only when λ is small and γ is small. The logical $alltoall$ topology out–performs other topologies only when the network is almost saturated, that is, large λ or large γ. This indicates that in general, the logical hypercube and $allXY$ topologies are better topologies than the logical torus and $alltoall$ topologies in terms of the packet delay.

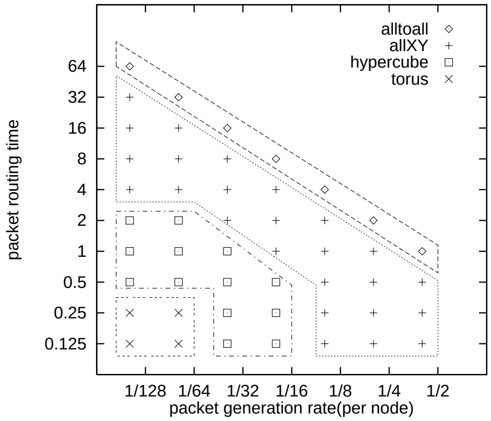


Figure 9. Best logical topology for a 16×16 torus

6 Conclusion

In this paper, we have studied the performance of logical topologies for routing messages on top of the torus topology. We developed an analytical model for the maximum throughput and packet delay for the multihop communication and confirm the model with simulation results. We further studied the performance of these topologies and identified the cases where each logical topology out–performs the other topologies.

In our study of the impact of system parameters on the maximum throughput and the average packet delay, we have concluded that in general, the performance of the logical topologies with less connectivity, such as the torus and hypercube topologies, are more sensitive to the network load and the router speed while the logical topologies with more connectivity, such as the $alltoall$ and $allXY$ topologies, are more sensitive to network size. Logical topologies with dense connectivity achieve higher maximum throughput than the topologies with less connectivity. In addition, they also scale better with network size. In terms of the maximum throughput, the topologies can be ordered as follows,

$$alltoall > allXY > hypercube > torus.$$

In term of average packet delay, the logical torus topology achieves best results only when the router is fast and the network is under light load, while the logical $alltoall$ topology is best only when the router is slow and the network is almost saturated. In all other cases, logical hypercube and $allXY$ topologies out–perform logical torus and $alltoall$ topologies. Comparing the logical $allXY$ to the logical hypercube, the $allXY$ topology is better when the network is under high load. These results hold for all network sizes.

References

[1] C. A. Brackett, “Dense wavelength division multiplexing networks: Principles and applications,” *IEEE Journal on Selected Areas of Communications*, Vol. 8, pp. 948-964, Aug. 1990.

[2] I. Chlamtac, A. Ganz and G. Karmi. “Lightpath Communications: An Approach to High Bandwidth Optical WAN’s” *IEEE Trans. on Communications*, Vol. 40, No. 7, pages 1171–1182, July 1992.

[3] S. Hinrichs, C. Kosak, D.R. O’Hallaron, T. Stricker and R. Take. “An Architecture for Optimal All–to–All Personalized Communication.” In *6th Annual ACM Symposium on Parallel Algorithms and Architectures*, pages 310-319, June 1994.

[4] R. Melhem, “Time–Multiplexing Optical Interconnection Network; Why Does it Pay Off?” In *Proceedings of the 1995 ICPP workshop on Challenges for Parallel Processing*, pages 30–35, 1995.

[5] C. Qiao and R. Melhem, “Reconfiguration with Time Division Multiplexed MIN’s for Multiprocessor Communications.” *IEEE Trans. on Parallel and Dist. Systems*, Vol. 5, No. 4, pages 337–352, April 1994.

[6] C. Qiao and R. Melhem, “Reducing Communication Latency with Path Multiplexing in Optically Interconnected Multiprocessor Systems”, *IEEE Trans. on Parallel and Distributed Systems*, vol 8, no 2, pp.97-108, 1997.

[7] S. Subramanian, M. Azizoglu and A. Somani, “Connectivity and Sparse Wavelength Conversion in Wavelength-Routing Networks.” *Proc. of INFOCOM’96*, pages 148–155, 1996.

[8] X. Yuan and R. Melhem “Optimal Routing and Channel Assignments for Hypercube Communication on optical Mesh–like Processor Arrays.” *Fifth International Conference on Massively Parallel Processing Using Optical Interconnections*, Las Vegas, June 1998