Multicomputer Interconnection Network Channel Design

Technical Report UW-CSE-93-12-03

Kevin Bolding
Dept. of Computer Science and Engineering – FR-35
University of Washington
Seattle, Washington, USA 98195
kwb@cs.washington.edu

December 14, 1993

Abstract

Massively parallel multicomputers require a high-performance interconnection network. The physical channels which connect nodes in the network are the components that most commonly impose restrictions on the amount of data which can be communicated between nodes of the network. Several different schemes for maximizing the bandwidth of physical interconnection channels are examined.

Conventional technology centers on 5V electrical signalling. Within this paradigm, there are several different methods of maximizing throughput of a channel, given its limited bandwidth. Simplex and duplex channels are examined, as well as arbitration protocols and transmission line pipelining. Non-conventional technologies which achieve higher bandwidth transmission through low-voltage signalling and fiber optic communication are also examined. For the present, it is expected that most networks will continue to use electrical signalling with low-voltage signalling because fiber optic hardware is too expensive. As technology changes, though, it may become more common for networks to be built out of optical or other new technologies.

Keywords: Interconnection networks, Massively parallel machines.

1 Introduction

Modern massively parallel multicomputers require the use of large, high performance, interconnection networks. The primary components of this network are the processing nodes, routers which direct information through the network, and the physical links of the network, or channels. While there has been a large amount of analysis directed at routing algorithms in recent years, most research on channel design has been directed at engineering faster channels by using faster technologies. The analysis of different protocols has been very limited at best. This work examines the basic and common forms of channel design for multicomputer networks and provides a comparative analysis of the advantages and liabilities of each.

Using conventional technologies (i.e. 5V electrical signalling), there is a relatively large design space to be explored. The primary variable is whether the wires connecting two nodes should

be simplex (uni-directional) or half-duplex (bi-directionally multiplexed). Other design variables concern the use of transmission line pipelining and arbitration protocols for multiplexed channels.

When new technologies are introduced, the variables change to some degree. While most existing multicomputer networks use standard cables and PC-board traces for wires and are driven by standard 5V devices, new technologies are being introduced rapidly. Examples of this are lower-voltage signalling, concurrent full-duplex signalling, and fiber-optic signalling. Each of these results in changes in the design space and must be examined.

2 Conventional Design

Conventional channel design uses simple electrical connections between routers. These may be either ribbon cables or, when the topology allows, traces on printed-circuit boards. The channels are driven at the same voltage as the driving chips. This usually translates into 5V drivers for channels driven by CMOS and TTL chips. The advantages of using this technology are clear: it is well-understood and no work has to be done to convert from the on-chip 5V technology to the off-chip 5V technology. Several variations on channel design have been introduced in order to provide maximum performance. These are explored in the following sections.

2.1 Half-Duplex vs. Simplex

In most multicomputer interconnection networks, the nodes are connected together in a bi-directional topology. This means that for any nodes a and b, if there is a directed link from a to b, there is also a directed link from node b to a. There are some networks that do not exhibit this property, such as the Manhattan Street Network and the directed torus networks, but most implemented networks are bi-directionally connected.

When implementing such a network, the physical links may be built either as distinct directed links (simplex), as illustrated in Figure 1, or as a single set of wires that is multiplexed between the two directions (half-duplex[1]), as shown in Figure 2. Since the number of wires in a channel (its width) is constrained by either pin limitations or by limitations on the number of wires in the system, there is a maximum number of wires that can be allocated to a given channel. Thus, as a general rule, simplex channels must be one-half the width of duplex channels because twice as many channels are required.

2.1.1 Bandwidth

Given a fixed number of wires, simplex and duplex channels provide different amounts of bandwidth depending on the design of the channel and the nature of the load presented to the channel. Simplex channels require duplication of control wires that is not needed with duplex channels (see Figures 1 and 2). Because of this, duplex channels can provide a larger amount of data bandwidth using the same number of wires as simplex channels. The magnitude of this effect becomes smaller as data paths become wider relative to the fixed number of control wires. Since this has only a marginal effect on total bandwidth, it will not be considered during the remainder of this section.

[1] Half-duplex channels will be referred to as "duplex channels" throughout the remainder of this paper. References to full-duplex channels will be explicitly identified.

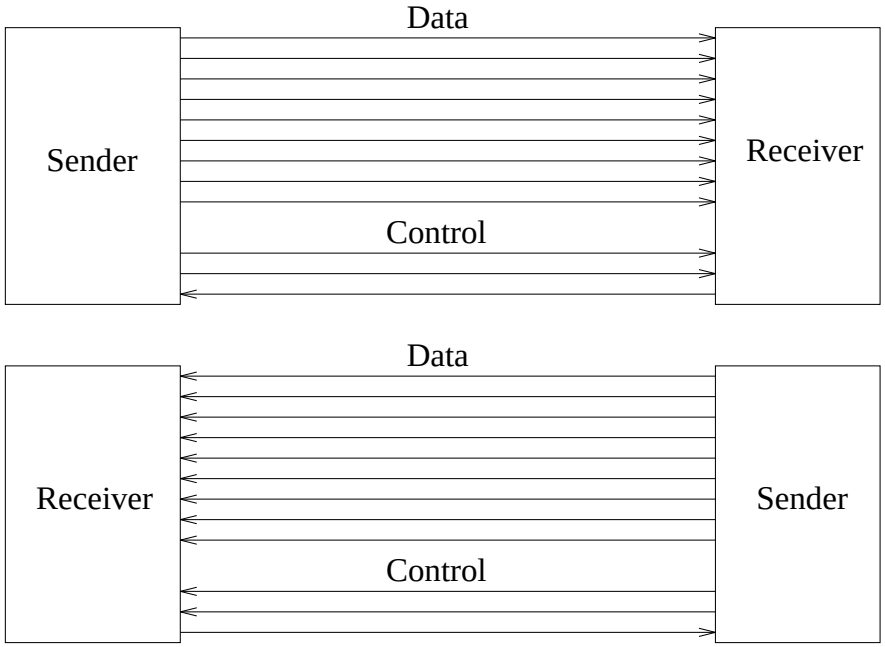

Figure 1: A bi-directional link implemented as two directed simplex channels. 16 bits of data can be transmitted at once, using both channels. The total number of wires for the link, including three bits of control per channel, is 22.

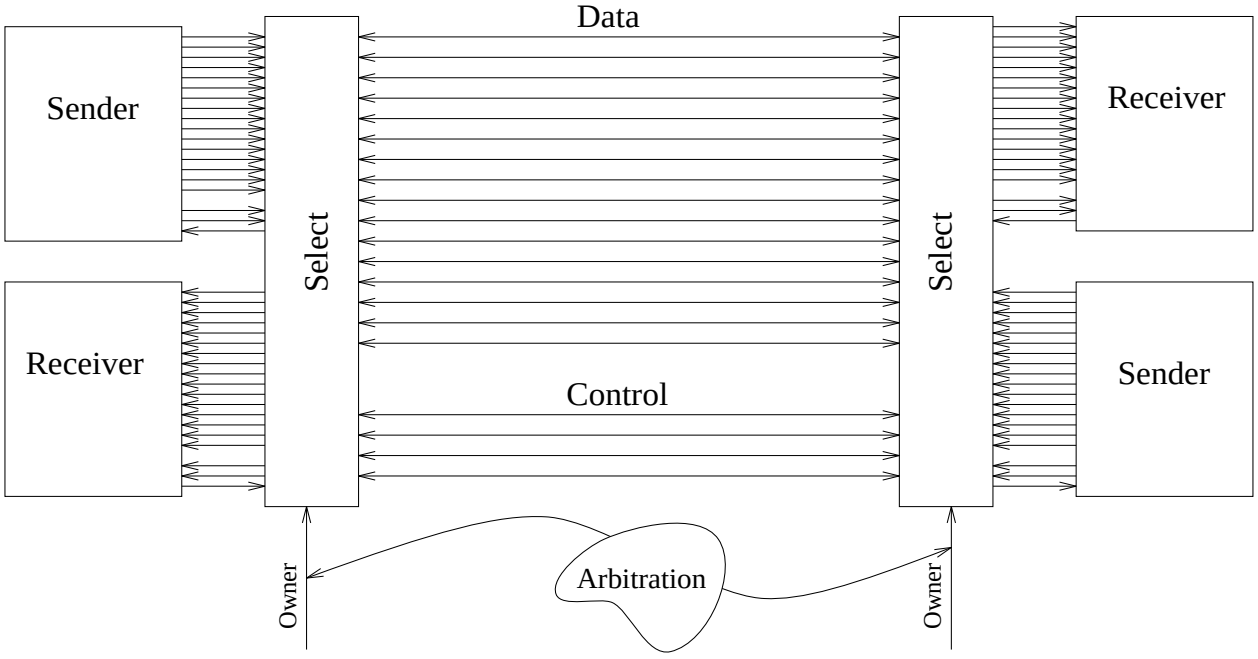

Figure 2: A bi-directional link implemented as a single shared duplex channel. 16 bits of data can be transmitted at once. The total number of wires for the link, with three bits of control per channel plus and additional wire to help in channel arbitration, is 20.

The way that the effective bandwidth changes with the applied load is more interesting. If the load on a channel is such that it is constantly demanded in both directions at all times, both simplex and duplex channels can deliver the same bandwidth. For paired simplex channels, both channels will be constantly utilized at all times. With duplex channels, the channel will only be able to send data in a single direction at a time, but with twice the bandwidth of a single simplex channel. Thus, both simplex and duplex schemes achieve the same bandwidth under heavy, evenly-distributed loads.

When the applied load is uneven, though, the balance shifts in favor of duplex channels. Assume a load which demands the channel constantly, but only in a single direction. With paired simplex channels, one channel will be fully utilized, while the other sits completely idle. The effective bandwidth will be one-half that of the entire channel. With duplex channels, the entire width of the channel can be devoted to the demanded direction and, thus, the full bandwidth of the channel can be used. Therefore, duplex channels can, when the load is unbalanced, deliver twice the bandwidth of paired simplex channels.

Unfortunately, effective use of duplex channels requires somewhat complicated arbitration schemes. A simple arbitration scheme where each half of the duplex channel is given ownership in a round-robin fashion can be easily implemented. For instance, one side could own the channel during odd-numbered clock cycles and the other during even-numbered clock cycles. The problem with this scheme is that it is oblivious to the load presented and does not achieve any better utilization than paired simplex channels. For example, in the unbalanced load case presented above, it would still give ownership to the no-load side of the channel one-half of the time, resulting in one-half the effective bandwidth.

To provide the maximum channel utilization given any presented load, the channel arbitration logic must grant ownership based on demand. Also, to provide "fairness," the logic must grant ownership in a timely manner whenever one side desires use of the channel. This operation requires that the non-owning side of the channel communicate information to the owning side describing whether or not it desires ownership of the channel. A straightforward protocol can be used if data is sent in bounded-sized packets. In this protocol, channel ownership is transferred whenever the end of a packet is transmitted [Bolding 93]. If one half of the channel gains ownership, but does not have data to transmit, it transmits a single-word "no-op" packet to the other side, yielding ownership back to the other side. This protocol is "fair" in that if both sides of the channel constantly desire ownership, it will be traded back-and-forth on a packet basis. If one half desires the channel more often, it will be given it more often. The drawback to this method is that bandwidth is lost when exchanging information. If only one side of the channel can utilize the channel, it must still waste the time to yield and regain ownership at the end of each packet transmission. When the channel is in constant bi-directional demand, a cycle must be wasted whenever the non-owner receives ownership in order for it to recognize that the channel has been yielded. Thus, bandwidth is reduced due to the complexities of communicating information about the status of each side of the channel.

It is possible to communicate status information by using more control wires. If an additional reverse-direction wire is available, it can be used to indicate to the owner of the channel whether or not the non-owner desires it. If the non-owner does not desire ownership, there is no reason for the owner to yield the channel, so it does not have to waste bandwidth doing so. If packets are always of a known size, the non-owner can assume ownership as soon as the current owner has transmitted the last word of the present packet, and full bandwidth utilization can be achieved. Alternatively,

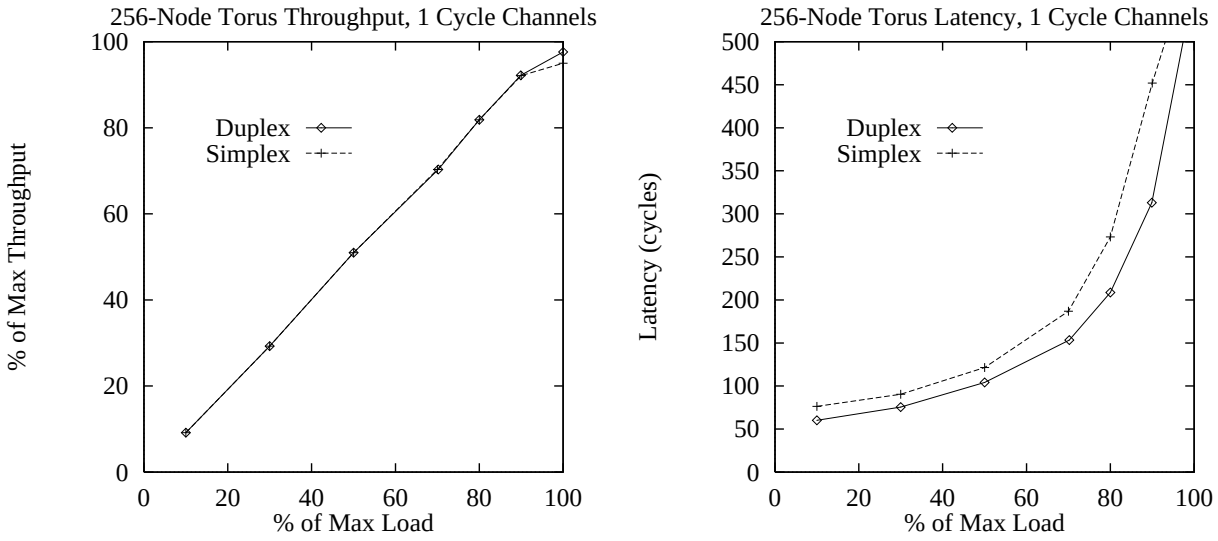

Figure 3: Simplex vs. half-duplex channels. 1-cycle channel transmission time.

if packets are of varying (but bounded) size, the sender can give an early "end-of-message is coming soon" signal on a dedicated wire so the change of ownership can occur without delay. Thus, by using extra wires, full utilization of the data wires can be achieved with duplex channels.

2.1.2 Simulation Results

Simulations were performed to gauge the performance of simplex and duplex channels. The reported results are based on simulations of 256-node torus networks using Chaotic adaptive routing [Konstantinidou & Snyder 90, Bolding 93] with a uniform random traffic load. Throughput results are normalized to the theoretical maximum throughput constrained by the bisection bandwidth of the network. Latency figures reported do not include source queueing. The routing node modeled has an internal delay of 3 cycles, but uses virtual cut-through routing, so the delay is only added to the first flit of each packet traversing a node. Data in the simulated network is sent in fixed-size packets. Because simplex channels can have only half of the width of duplex channels, packets are 20 words long in the duplex networks and 40 words long in the simplex networks.[2] This results in transmitting the same amount of data in each packet. To ensure a fair comparison, the internal router datapaths for both networks are made the same width, which is equal to the width of the duplex channels. In the simulations reported on, there is no penalty for changing the direction of a duplex channel or for the extra control wires needed in simplex channels.

Figure 3 shows a comparison of simplex and duplex channels when the delay across each channel wire is a single cycle. Thus, the minimum latency added at each hop in a packet's path is 4 cycles: 3 cycles for the node delay and 1 cycle for the channel transmission. As can be seen in Figure 3, duplex channels provide all-around better performance. The throughput is essentially the same for both schemes. However, duplex channels give lower latencies throughout the entire loading range due to the higher bandwidth available when channels are not loaded exactly evenly in each

[2] If duplex channels are 16 bits wide and simplex channels are 8 bits wide, these packet lengths allow 32 bytes of payload and 8 bytes of control information per packet.

direction.

2.1.3 Wormhole Routing

Wormhole routing [Dally 87] is a form of routing in which messages are sent in continuous streams through the network. Messages may be any size, and there is no packetization needed. When a message in a wormhole-routed network is blocked due to a resource conflict, it retains ownership of all of the resources it owns, including channels, and waits for the blockage to clear. In general, data flows without interruption behind the head of the message. Modifications to the basic scheme introduce virtual channels [Dally 92] which allow messages to yield channel resources when blocked, while still holding all buffering resources.

Effective utilization of duplex channels with wormhole routing faces two hurdles. The first is that the previously described channel arbitration scheme relies on bounded-length packets. The second is that interrupting a message's transmission by yielding ownership of a channel during the middle of a message creates "segmented" worms which result in lower resource utilization.

In the preceding discussion, it was assumed that data is transmitted in bounded-size packets. This is because channel ownership in the duplex arbitration protocol is exchanged only at the ends of packets. If wormhole routing is used, the message sizes are unbounded and the opportunity for starvation is introduced. This shortcoming may be overcome by forcibly exchanging ownership after some pre-determined number of data words have been transmitted in a single direction. In an ideal case, arbitration could be done on every cycle. However, because of the time needed for arbitration logic to settle, this requires either additional wires to communicate information about each side's future desires for channel ownership, or the insertion of a "dead" cycle for every word transmitted. To overcome this, the time allocated to each owner may be extended. For example, each side may be allocated time to transmit 100 words before requiring arbitration, thus amortizing the cost of arbitration of a longer period.

While this protocol has the desirable result of allowing full utilization of duplex channels, it creates new difficulties as well. Whenever a message is interrupted in order for a channel's ownership to be exchanged, the portion of the message "behind" the channel is blocked from progress until the channel's ownership is re-exchanged. During this time, the "forward" portion of the message may continue on towards its destination, thus segmenting the worm into two or more connected worms. However, the path through the network must remain reserved for the remainder of the message. Thus, there may be many resources which are reserved for the blocked portion of the message that are not utilized by any message in the network. Even if single-cycle arbitration is achieved, resource use per message may be doubled due to the insertion of "empty" words between each actual data word. When the arbitration period is made longer, the situation worsens. This, in turn, results in reduced network performance due to the under-utilization of resources.

Therefore, due to the complexities of using wormhole routing with duplex channels, it is not effective to mix the two techniques. However, since packet-switched routing with virtual cut-through [Kermani & Kleinrock 79] is essentially equivalent to wormhole routing for small message sizes, the remainder of this paper will focus on packet-switched routing.

2.2 Transmission Line Wave Pipelining

Because the wires which form the interconnections in a network are relatively long with respect to the speed of transmission of electro-magnetic waves through them, they must be analyzed as

Table 1: Capacity of transmission lines, v=.2 m/ns.

Data Rate (MBits/s)	Capacity (bits/meter)
50	.25
100	.5
250	1.25
500	2.5
1000	5.0
2000	10.0

transmission lines rather than simple capacitive loads. Accordingly, when a signal is applied to a transmission line, its wavefront travels through the line at speed $v = \frac{c_o}{\sqrt{\epsilon_r}}$, where c_o is the speed of light in a vacuum and ϵ_r is the relative permittivity of the dielectric material the transmission line is made of. Typical values of v for common materials range from 0.1 to 0.2 m/ns [Bakoglu 90]. If the signal being propagated by the transmission line has short rise and fall times with respect to the time of propagation, it is possible to change the signal at its source before the original value has reached the receiving end of the line. If the line is long enough, the signal's value may be changed multiple times before the first value has been received. This allows multiple data words to be pipelined on a single physical channel with no intervening circuitry necessary to separate the words.

The number of words which can be pipelined on a transmission line varies as a function of the clock frequency, f, the length of the line, l, and the speed of propagation of the wave in the line, v: $n = \frac{fl}{v}$. For a typical value of v, 0.2 m/ns, Table 1 shows the capacity of transmission lines per unit length for different clock rates. For low-dimensional networks, links can be kept short, within the 0.1–0.2m range, yielding only a single bit per wire capacity even at 1GBits/s frequencies. For higher dimensional networks, such as hypercubes, the maximum link length grows as the system grows, resulting in links which may be several meters long. In this case, many bits may be pipelined on a single wire.

As long as sufficient buffering is available at the receiving end, the presence of multiple bits on a single wire can be made transparent to the sender and receiver. The advantage to this is that time for transmission of data along the wire can be ignored as long as the wire pipeline is kept full. The cycle time of the network channels thus becomes dependent only on the speed of the channel's drivers and the technology of the wires, but not the actual length of the wires. This becomes more important as the cycle times of network chips are reduced and the size of networks is increased.

Two difficulties arise when mixing duplex channels and wave pipelining, both due to the limitations of pipelining strategies. The first is that changing the direction of the channel requires waiting for all of the data in the channel to drain first. Thus, each change of channel ownership is delayed by a number of cycles equal to the number of words the channel can hold. Moreover, the first word traveling in the new direction incurs the full delay of the transmission line. Essentially, changing the owner of the channel requires draining the pipeline and then waiting for it to refill with new data, so that the delay between the last word sent in the original direction and the first word arriving in the opposite direction is two complete wire delays. As the number of words which can be pipelined on a single wire increases, this delay becomes more and more significant.

The second difficulty with duplex channels is related to the pipeline drainage problem. Efficient

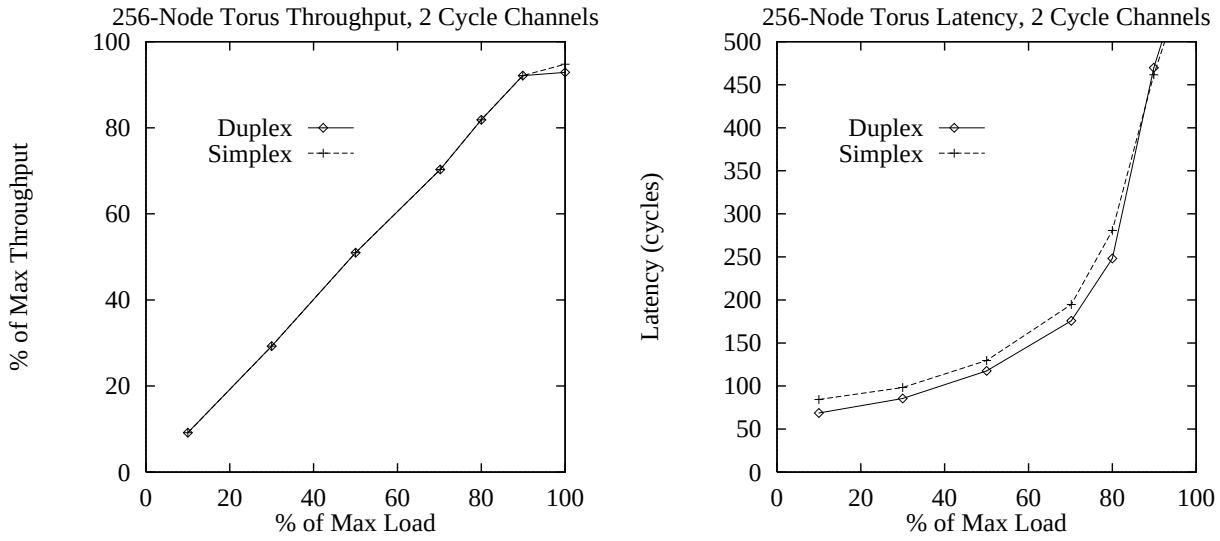

Figure 4: Simplex vs. half-duplex channels using transmission line pipelining. 2 cycle channel delay.

arbitration requires the transmission of data in the reverse direction that gives the status of the non-owning side. Since this information must incur the complete latency of the pipeline, it may not reflect the current status of the channel accurately. For instance, if the non-owner indicated at some time that it had no packet waiting to be transmitted, the owner would not know of that information until d cycles later. In the meantime, a packet may have arrived at the output channel which changes the non-owner's status. However, the decisions made by the owner must be made with the information it has available, so it would not yield the channel. Thus, due to the pipeline, decisions are based on stale data and may result in efficiency reductions.

2.2.1 Simulation Results

Simulations of networks using simplex and duplex channels with longer channel wire delays were performed. The methodology is the same as that described in Section 2.1.2. The channel delay was lengthened to 2 cycles and 5 cycles to gauge the effect on throughput and latency. For simplex channels, flits were pipelined on the data wires when possible, and no penalty was charged for communicating flow-control information. For duplex channels, flits were pipelined where possible as well. However, whenever the channel's direction is changed, it is necessary to wait for the transmission line pipeline to drain for it to restart in the opposite direction.

Figure 4 shows the results when wires have a 2-cycle delay. As with 1-cycle channels, the throughput is essentially the same in each case. Simplex channels provide a slightly higher maximum throughput, while the situation was inverted with 1-cycle channels. This is because when the load is high, the duplex channels must incur the delay due to flushing the pipeline nearly all of the time, while simplex channels do not. The latency results are similar. The low-load figures are about the same as for 1-cycle channels since both the simplex and duplex channel pipelines are likely to be empty most of the time. However, at higher loads, the better utilization of simplex channels allows a lower overall latency.

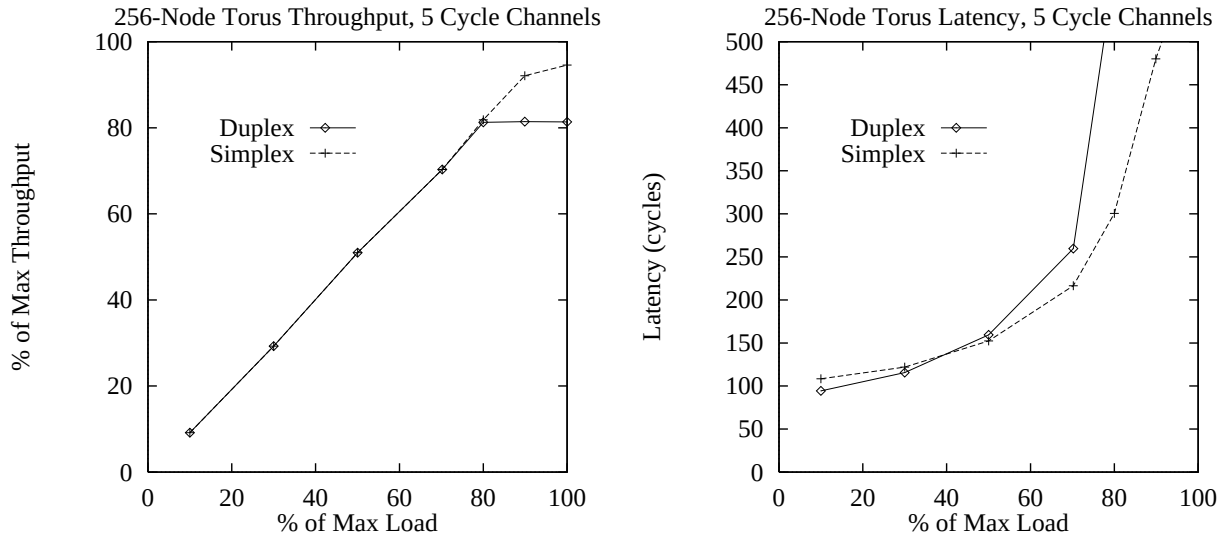

Figure 5: Simplex vs. half-duplex channels using transmission line pipelining. 5 cycle channel delay.

When the channel delay is stretched to 5 cycles, simplex channels outperform duplex channels. Figure 5 shows these results. In this case, duplex channels achieve only about 85% of the throughput of simplex channels. Also, the latency is lower with duplex channels only for very low loads. As the load increases, the latency with duplex channels quickly grows to be larger than that with simplex channels.

As channel data rates and cable lengths increase, the necessity of using wave pipelining increases. Because duplex channels require low latency across the channel, their use becomes impractical after a certain point. If latency across a channel is reasonably small ($1-2$ cycles), duplex channels will still be of benefit, especially if arbitration can be done after long intervals [Wille 92]. As the latency across a channel grows, duplex channels quickly loose their advantage. Thus, the benefits of duplex channels will depend on the ability to design systems with short channels, especially as channel data rates increase. However, as long as interconnection networks can maintain low degree topologies, such as two- or three-dimensional mesh and torus networks, the wire lengths will remain short enough to allow effective use of duplex channels, in conjunction with transmission line pipelining if necessary.

3 Non-Conventional Technologies

Conventional interconnections for multicomputer networks have used the techniques described above: 5V electrical signalling in a single direction at a time. Several techniques have recently begun to be applied to the channel design of multicomputer networks which go beyond the bounds of this "ordinary" technology. Although these techniques are, by and large, not new in principle, their application in this area is a recent development, so they are labeled "non-conventional."

3.1 Low-Voltage Signalling

By reducing the voltage swings required for signalling from the conventional 5V range to lower ranges, the interconnection drivers can operate at higher data rates and consume less power. For example, conventional 5V drivers implemented in 1.2μm CMOS are limited to a data rate of about 66Mbits/s [Bolding et al. 93]. By using voltage swings of 0.25V and a 0.8μm process, a data rate of around 500Mbits/s should be attainable [Dennison et al. 93]. A difficulty with low-voltage signalling is its susceptibility to noise. Even a small spike in data can corrupt a signal with such a small voltage difference between its high and low levels. Differential signalling, where each signal is transmitted on two wires: one for a reference voltage and the other for the actual signal, can overcome this problem. A system using 0.5V differential signalling and a 1.2μm CMOS process was demonstrated to achieve a data rate of 1.2Gbits/sec [Svensson & Yuan 93]. Because differential signalling systems require two wires for every signal, though, their bandwidth must be twice that of conventional single-wire system in order to achieve the same effective bandwidth per wire. It is not currently clear which will scheme will achieve the highest bandwidth for a given technology.

3.2 Concurrent Full-Duplex Signalling

A novel scheme which allows signals to travel concurrently in opposite directions on a single wire is presented in [Lam et al. 90]. This scheme relies on the superposition principle of electro-magnetic waves: two signals sent through the same wire will generate a wave in the wire which is the sum of the original signals. By subtracting the source waveform from the received waveform, each driver can recover the signal transmitted from the other end. The advantage of this scheme is clear: each wire's bandwidth is effectively doubled. This technique can be combined with transmission line pipelining easily.

The primary difficulties with this scheme are its complexity and greater susceptibility to noise. Because the scheme requires subtraction of the input waveform and differential sensing to recover the received waveform, it is somewhat more complex than conventional signalling. Also, because there are two superimposed signals on a single wire, noise introduced at either source will be added together in the resulting signal, complicating the noise problem. Experimental results in [Dennison et al. 93] have demonstrated 100Mbits/sec (each way) concurrent signalling. The complexity of the design and noise problems curtailed performance from its expected levels in this experiment. As this technology matures, it may become a common method of achieving high bandwidth communication in interconnection networks.

3.3 Fiber Optics

Optical fiber communication techniques have been suggested as replacements for electrical communication schemes due to the large bandwidths of fiber optic lines, which can be as high as 32THz/fiber [Agrawal 92]. While this technology is very promising to larger-area networks, its applicability to multicomputer networks is not so clear. Since the data in a computer is stored in electrical form, it must be converted into a light-wave signal by means of a modulated source. Semiconductor lasers are generally used for this purpose, as they provide much faster switching times than LED's do. Unfortunately, the signalling rate of semiconductor lasers is limited to approximately 10GBits/s due to parasitic electrical effects of the modulation circuitry [Agrawal 92], so the full bandwidth of a fiber cannot be utilized using a single bit stream.

Because the fabrication of semiconductor lasers requires GaAs technology [Agrawal 92], integration with routing circuitry is more difficult. Typical routing chips are fabricated using silicon CMOS technology, and changing to the more difficult to use GaAs technology complicates the design and adds to the expense. The expense and area requirements of the multiple lasers needed for a routing chip are considerably greater than for the equivalent bandwidth using electrical signalling. For instance, differential signalling using 1.2μm CMOS has been demonstrated to run at speeds of 1.2GBits/s [Svensson & Yuan 93], corresponding to 0.6Gbits/s/wire. Thus, to achieve the same data rate, a channel can be fabricated using a single optical fiber at 10Gbits/s/fiber or 16 electrical wires.[3] Since VLSI packages may contain 200-300 electrical signal pins, this corresponds to 12-18 fibers and associated laser sources and detectors per routing chip. Since optical fibers cannot be fabricated onto printed-circuit boards in the same manner as conventional wires, their cost and reliability is much worse than electrical wires, especially if low-dimensional networks are used where electrical connections can be made with short wires. Given the cost of the number of laser components required per chip, along with the difficulties of externally wired connections, multicomputers, with short point-to-point connections will probably continue to use electrical network connections in the near future.

4 Conclusions

Conventional-technology channel design has focused on 5V electrical signalling for the interconnect between processors in multicomputers. Within this technology, there are several ways of designing the channels to maximize performance. Duplex channels allow maximum channel utilization over a wide range of applied loads, but cost in terms of complexity. Also, popular wormhole routing techniques cannot be effectively used with duplex channels. When the interconnection wires are relatively long, it is possible to use transmission line pipelining to overcome the latency due to the delay in the wire. However, transmission line pipelining does not mix well with duplex channel sharing due to the need to empty the data pipeline on each change of communication direction. Thus, with conventional technologies, which have a bandwidth limited by the speed of the 5V drivers, duplex channels will give the best results as long as channels are short (less than 30-40cm), but simplex channels with transmission line pipelining are the favored choice for networks with longer interconnection links.

The bandwidth limitation of conventional technology drivers can be made less severe by using other technologies. Low-voltage signalling can increase the bandwidth of electrical signalling to 600Mbits/s/wire or greater using CMOS technology. By applying the techniques of wave superposition, it is possible to transmit data in opposite directions along a single wire. Both of these techniques can be used together to generate very high-bandwidth electrical channels. The primary disadvantages are that noise susceptibility may be greater due to the lower voltages used, and components communicating using special protocols cannot be connected with conventional components without special interface circuitry.

Fiber optics offer an even higher bandwidth communication medium. Unfortunately, the large bandwidth of an optical fiber cannot be realized using a single data signal due to a bottleneck in translating data from electrical to optical and back. This bottleneck typically limits a single transmitter/receiver pair to 10Gbit/s operation. Because the lasers needed to perform the electrical

[3] Additional fibers may be needed to forward clock and/or control signals.

to optical translation are relatively large and require special technologies to implement, their cost currently makes fiber optics too expensive for current systems. As their costs are reduced, though, machines may be built which can take advantage of the very large bandwidth available with optical fiber interconnections.

References

[Agrawal 92] G. P. Agrawal. *Fiber-Optic Communication Systems.* John Wiley and Sons, Inc., 1992.

[Bakoglu 90] H. Bakoglu. *Circuits, Interconnections, and Packaging for VLSI.* Addison-Wesley, 1990.

[Bolding 93] K. Bolding. *Chaotic Routing: Design and Implmentation of an Adaptive Multicomputer Network Router.* PhD dissertation, University of Washington, Seattle, WA, July 1993.

[Bolding et al. 93] K. Bolding, S.-C. Cheung, S.-E. Choi, C. Ebeling, S. Hassoun, T. A. Ngo, and R. Wille. The chaos router chip: Design and implementation of an adaptive router. In *Proceedings of VLSI '93*, September 1993.

[Dally 87] W. Dally. Wire-efficient VLSI multiprocessor communication networks. In P. Losleben, editor, *Proceedings of the Stanford Conference on Advanced Research in VLSI*, pages 391–415. MIT Press, March 1987.

[Dally 92] W. Dally. Virtual-channel flow control. *IEEE Trans. on Parallel and Distributed Systems*, 3(2):194–205, March 1992.

[Dennison et al. 93] L. R. Dennison, W. S. Lee, and W. J. Dally. High-performance bidirectional signalling in VLSI systems. In *Proceedings of the 1993 Symposium on Research on Integrated Systems*, pages 300–319, 1993.

[Kermani & Kleinrock 79] P. Kermani and L. Kleinrock. Virtual cut-through: A new computer communication switching technique. *Computer Networks*, 3:267–286, 1979.

[Konstantinidou & Snyder 90] S. Konstantinidou and L. Snyder. The chaos router: A practical application of randomization in network routing. In *Proceedings of the 2nd Symposium on Parallel Algorithms and Architectures*, pages 21–30. ACM, 1990.

[Lam et al. 90] K. Lam, L. D. Dennison, and W. J. Dally. Simultaneous bidirectional signalling for IC systems. In *ICCD: VLSI in Computers and Processors*, pages 430–433, 1990.

[Svensson & Yuan 93] C. Svensson and J. Yuan. Ultra high speed CMOS design. In *Proceedings of VLSI 93*, pages 7.1.1 – 7.1.10. IFIP, September 1993.

[Wille 92] R. Wille. A high-speed channel controller for the chaos router. Master's thesis, Univeristy of Washington, 1992.