Removing Instability and Maximizing Throughput in a Multicast Shuffle-Exchange Network

Cathy W. Chan Soung C. Liew

Department of Information Engineering
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
Email: *wcchan4@ie.cuhk.edu.hk, soung@ie.cuhk.edu.hk*
Tel: +852 2609-8367 +852 2609-8352
Fax: +852 2603-5032

Abstract

Multicast capability can be incorporated into any interconnection networks by using a general packet replication scheme previously proposed in [4]. The network can then be used for both packet replication and routing processes. Unfortunately, such a multicast network can easily evolve to saturation due to instability. Once the network is saturated, the throughput drops to zero. The operation of the multicast network must therefore be carefully controlled to avoid the unstable region. Not well-thought-out control schemes, however, may result in a small throughput and inefficient utilization of the network. This paper investigates the stability issue in the multicast shuffle-exchange network in detail. Several schemes are then proposed to remove network instability. Our study indicates that a dynamic access control scheme can potentially achieve high network throughput even under non-uniform traffic conditions.

I. Introduction

A broadband multicast packet switch is often constructed by the cascade combination of a copy network and a point-to-point switch [1, 2, 3]. The copy network replicates the packets on request and the point-to-point switch delivers them to their respective destinations. Multicast capability can also be incorporated, in a similar fashion, into recirculating interconnection networks, such as the shuffle-exchange network and the Manhattan-street network. The multicast process is divided into two phases – the replicating phase and the routing phase. In the replicating phase, a packet wanders in the network without a destination, and duplicates itself whenever an empty link appears until all the requested copies are generated. Each copy that requires no further duplication becomes a routing packet and subsequently travels to its destination. The packet replication scheme proposed in [4] further generalizes this strategy for arbitrary interconnection networks. The performance of the general multicast network is studied in [5]. It is shown in [5] that the operation of a multicast network is unstable when the network load is high. In this paper, we will focus on the multicast shuffle-exchange network and investigate the instability issue and propose some possible solutions.

II. Performance of the Multicast Shuffle-Exchange Network

The shuffle-exchange network is a single-stage recirculating network in which the outgoing links of the nodes are fed back to the same stage after a shuffle. All packets are assumed to be of equal length and the network operates on a time-slotted basis. An example 4-node network is shown in Fig. 1. The performance of the multicast shuffle-exchange network that uses deflection routing and a random contention resolution scheme to resolve packet conflicts is studied in [5] and is highlighted in this section. In the analysis, it is assumed that packets in the network are independent of each other and the destinations of packets are uniformly distributed over all the nodes.

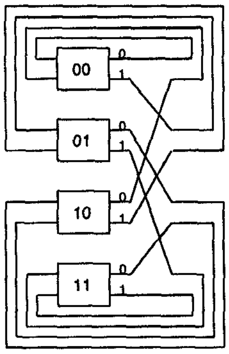


Figure 1: A 4-node shuffle-exchange network.

Consider a multicast shuffle-exchange network with $N=2^n$ nodes. The structure of a network node is shown in Fig. 2. Arriving packets that are destined for the node are removed at the output processor. Packets originating from the source of the node are buffered in the input queue

0-7803-3925-8/97 $10.00 ©1997 IEEE

and a packet is injected into the network through the input processor when an empty link is available. The replicating switch duplicates a replicating packet if the other link is empty, and switches routing packets to their desired links, with contentions resolved by deflection routing. Replicating packets are forwarded to arbitrary outgoing links if duplication is unsuccessful.

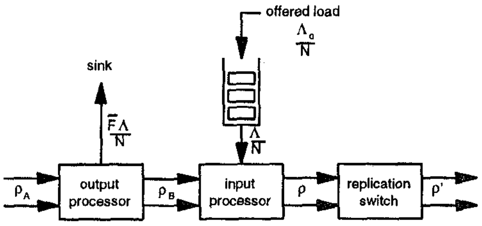


Figure 2: A 2 × 2 switch node.

Let ρ_A,$\cdot\rho_B$, ρ and ρ' denote the link loading at each input of the output processor, each output of the output processor, each output of the input processor and each output of the replication switch, respectively. Furthermore, let Λ_o be the network offered load and Λ be the packet input rate, which count the average number of new packets that originate from the sources (and join the input queues) and those that actually enter the network, respectively, in each time slot. The packet output rate, or throughput, is the average number of routing packets reaching their destinations in a time slot and equals $\overline{F}\Lambda$ under equilibrium, where $\overline{F}$ is the average fanout of the packets. For simplicity, we will use the notations $\tilde{\Lambda}_o$=Λ_o/N, $\tilde{\Lambda}$=Λ/N and $\overline{F}\tilde{\Lambda}$=$\overline{F}\Lambda/N$ respectively to refer to the offered load, packet input rate and throughput on a per-node basis. Finally, the packet replication probability (P_r) is defined as the probability that a packet at the input link of the replication switch is a replicating packet and the average routing delay D is the average number of time slots taken by a routing packet to reach its destination after it has finished its replication process. The parameters Λ, P_r and D are inter-related by the following equations [5].

$$P_r = \frac{(\overline{F}-1)\Lambda}{2\rho(1-\rho)N} \quad (1)$$

$$\Lambda = \frac{2N(1-\rho)\rho}{\overline{F}-1+\overline{F}(1-\rho)D} \quad (2)$$

$$D = \frac{1-[1-\frac{1}{4}\rho(1-P_r)]^n}{[1-\frac{1}{4}\rho(1-P_r)]^n(\frac{1}{4}\rho(1-P_r))} \quad (3)$$

The first two equations are applicable to arbitrary network topologies whereas the third is determined by the network topology and routing algorithm used. They can be solved iteratively and numerically to obtain the network throughput at any particular link loading ρ. Figure 3 plots the per-node throughput $\overline{F}\tilde{\Lambda}$ against the link loading ρ for a multicast shuffle-exchange network with N=256 and $\overline{F}$=8. The variation of the network throughput can be explained by its interactions with the packet replication probability and the average routing delay [5]. Here we are interested in two closely related phenomena known as deadlock and network instability.

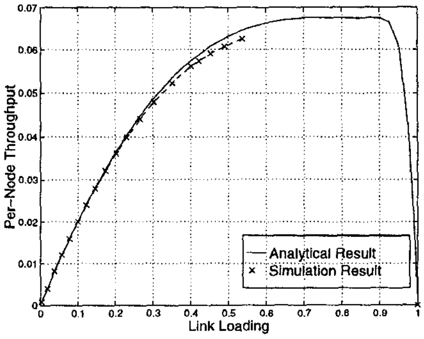


Figure 3: Throughput of the multicast shuffle-exchange network.

Deadlock refers to the situation in which the overall status of the network remains unchanged over time. When the multicast shuffle-exchange network is saturated (ρ=1), packet duplication becomes impossible and replication requests can never be accomplished. The existing packets keep circulating in the network and no new packets can enter. As shown in Fig. 3, the saturation throughput is zero.

The overall network throughput increases with link loading ρ at light load and decreases with ρ at high load. In general, for any networks, operations in regions in which the throughput decreases with link loading is unstable [7]. Any attempt to operate the network in such a region will evolve to a stable operating point at either the beginning or the end of the negative slope. To see this, suppose the network initially operates at some particular ρ and Λ in the negative-slope region. If at any time, the instantaneous link loading increases slightly, the network throughput is reduced according to the negative slope. The reduction in packet clearance results in a further increase in the link loading. This positive feedback effect eventually brings the network to the saturation point where the network is deadlocked with zero throughput. It can be shown (by (2)) that all multicast networks exhibit instability at high load [5].

The simulation results shown in crosses in Fig. 3 confirm this argument. Due to instability, it is impossible to operate the network in equilibrium in the negative-slope region. Instead, network operation always evolves to the saturation point. In addition, the results reveal that the region immediately preceding the negative-slope region is also unstable – an operating point can be easily shifted to the negative-slope region and evolve to saturation. Hence, the whole region $\rho \geq 0.5$ is unstable and no data can be obtained.

Since the major cause of deadlocks in a multicast network is the indefinite circulation of replicating packets, a natural way to break deadlocks is to discard replicating packets which have been in the network for too long. This is achieved by attaching an age counter to the header of each replicating packet. The age counter is incremented in each time slot, regardless of the success of duplication. When the age of a replicating packet reaches the lifetime limit T_{max}, the packet is discarded. This deadlock-breaking mechanism eliminates indefinite packet circulations and the network is deadlock-free. Note that routing packets can always be delivered to their destinations in finite time and do not contribute to deadlocks.

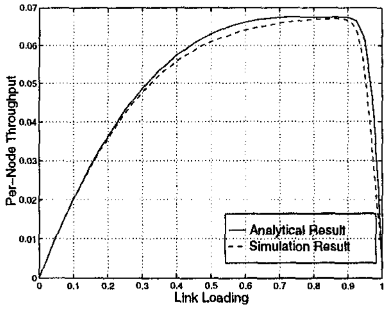


Figure 4: Simulation results of the multicast shuffle-exchange network with deadlock-breaking mechanism.

Figure 4 shows the simulation results when this strategy is applied to the multicast shuffle-exchange network, where T_{max} is 40 time slots. The network can now operate in equilibrium in the high-load region. Even though the network may sometimes evolve from its operating point to the saturation point, packets are removed from time to time and deadlocks do not result. Network operation can then be shifted back to the original operating point. Thus, network stability is achieved and the network can operate at all link loadings.

It is found that the choice of T_{max} has significant effect on network performance. When it is too small, replicating packets are removed so soon that many replication requests cannot be accomplished. The network throughput becomes much smaller than what the network can achieve. On the other hand, if T_{max} is too large, the replicating packets are allowed to stay in the network for too long that the network is easily saturated. Even though the network is deadlock-free, it is still unstable in the high-load region. Hence, the value of the lifetime limit T_{max} has to be carefully chosen to ensure stability as well as a high network throughput.

A major drawback of this mechanism is undesirable packet loss which may lead to retransmission by the packet sources and an increase in the network offered load. Thus, to protect the network against instability, additional access control schemes are required and are discussed in the following sections. Note, however, that the control schemes alone are incapable of completely eliminating deadlocks and when deadlocks do occur, they have to be broken. Hence, the deadlock-breaking mechanism should always be used in conjunction with the control schemes to ensure the network is always deadlock-free. The lifetime limit should be large enough that packet loss is minimized.

III. Static Access Control

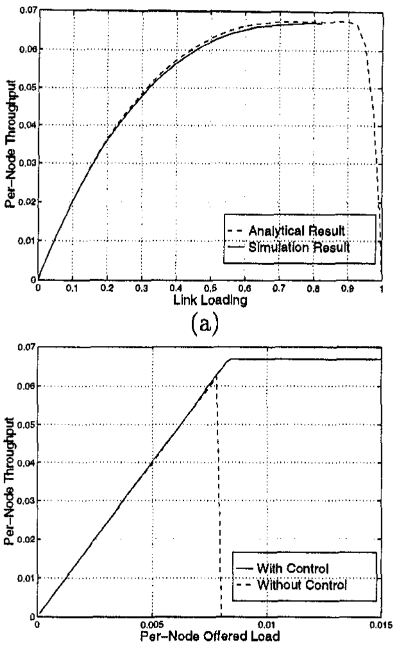
In the previous discussions, we assume that the input processors of the network nodes work in a greedy fashion. An input packet is injected into the network as soon as an empty link appears. This behaviour increases the input rate of the network and the network can easily run into instability. We show in this section how network operation can be stabilized by asserting access control on the network nodes.

The simulation results in Fig. 3 show that the stable operation region of the multicast shuffle-exchange network is $\rho \leq 0.5$. If packets enter the network in such a way that the network link loading ρ is always smaller than 0.5, network stability can be ensured. This is equivalent to requiring $\overline{F}\tilde{\Lambda} \leq 0.06$ or $\tilde{\Lambda} \leq 0.0075$. This condition can be satisfied by forcing the source of each node to maintain its offered load to below 0.0075. Alternatively, we can assert this control at the head of the input queues. In this case, the input processor is responsible for limiting its packet input rate to below 0.0075. After a packet is injected, the node has to wait for at least 1/0.0075 or 133 time slots before transmitting the next packet, even if empty links appear within that time interval. If the source has an offered load larger than 0.0075, packet backlogs build up in its input queue. This acts as a feedback to the source that the network cannot support its offered load. The limitations of this scheme is that the utilization of the network is small: ρ is limited by 0.5 and $\overline{F}\tilde{\Lambda}$ by 0.06 which is smaller than the maximum throughput the network can achieve. Besides, in reality, it is unlikely that the nodes are evenly-loaded and limiting the input rate of all the nodes to the same upper bound is inflexible. The overall performance of the network can be improved if the heavily-loaded nodes are able to use the bandwidth not used by the lightly-loaded nodes. This is achieved by the static access control scheme.

The static access control scheme uses a packet injection probability P_{inj} to control packet input. When an empty link appears and a node has a non-empty input queue, a packet is injected into the network with probability P_{inj}. This slows down the rate at which empty links are filled by new packets and more links can be available for packet replications. The possibility that the network is deadlocked by replicating packets can be greatly reduced.

The advantage of using this scheme is that the packet input rate of a node depends on the probability that its input queue is non-empty, which is determined by the offered load of the node. A node with a higher offered load has a greater probability of having a non-empty input queue and thus has a higher packet input rate. On the other hand, nodes with smaller offered loads also have smaller packet input rates. The empty slots that are not used by these nodes will propagate to, and be used by, the more busy nodes. This enables efficient sharing of network bandwidth among the nodes when their offered loads are different.

Figure 5 shows the simulation results for the multicast



(a)

(b)

Figure 5: Simulation results of the multicast shuffle-exchange network with static access control.

shuffle-exchange network with P_{inj}=0.025. The first figure plots the per-node throughput versus link loading and the second plots the per-node throughput versus the per-node offered load. The operation region of the controlled network is now from ρ=0 to ρ=0.82. When the offered load becomes larger than the sustainable value, the network operates at ρ=0.82 and $\overline{F}\tilde{\Lambda}$=0.067 rather than becoming saturated. In the following discussion, we denote the parameters at the limiting operating point with an asterisk(*). For example, the maximum allowable link loading at P_{inj}=0.025 is ρ^*=0.82 and the corresponding packet input rate is $\tilde{\Lambda}^*$=0.067. In general, the value of P_{inj} can be adjusted such that network operation is restricted to link loading below ρ^* and the corresponding Λ^* obtained from analysis.

To find the value of P_{inj} for a particular operating region, observe that when the offered load is smaller than $\tilde{\Lambda}^*$=Λ^*/N, the packet input rate equals the offered load and there is no packet backlogs in the input queues. When the offered load is larger than $\tilde{\Lambda}^*$, the packet input rate (=$\tilde{\Lambda}^*$) is smaller than the offered load and packets accumulate in the input queues. We can assume that the input queues are always non-empty. Since a packet is injected into the network with probability P_{inj} when an empty link appears at the input processor, the rate at which a node puts packets into the network is

$$\tilde{\Lambda}^* = 2(1-\rho_B^*)P_{inj}$$

Furthermore, the number of packets at the outputs of an input processor is the sum of the number of packets at its inputs and the number of new packets it puts in:

$$2\rho^* = 2\rho_B^* + \tilde{\Lambda}^*$$

Combining these two equations, we obtain

$$P_{inj} = \frac{\tilde{\Lambda}^*}{2-2\rho^*+\tilde{\Lambda}^*} = \frac{\Lambda^*}{2N-2N\rho^*+\Lambda^*}$$

There is always a one-to-one correspondence between the value of the packet injection probability and the range of operation. When the packet injection probability is small, the maximum allowable link loading (ρ^*) is small, and the range of operation is small. Adjusting the value of P_{inj} thus determines the operation region. Conversely, one can first choose a region of operation (ρ^* and Λ^*) and calculate the value of P_{inj} that should be used. For instance, to limit network operation to $\rho \leq \rho^*$=0.7 and $\tilde{\Lambda} \leq \tilde{\Lambda}^*$=0.0084, P_{inj}=0.0138 should be used. Note that $\tilde{\Lambda}^*$=0.0084 corresponds to the maximum network throughput and can be achieved by different values of ρ^* (from 0.7 to 0.9) and P_{inj} (from 0.0138 to 0.0403). In other words, at very high load, a network that uses P_{inj}=0.0138 operates at a link loading of ρ=ρ^*=0.7 whereas one that uses P_{inj}=0.0403 operates at ρ=ρ^*=0.9, both yielding the maximum network throughput with $\tilde{\Lambda}^*$=0.0084. Recall that packet duplication can only be performed when a replicating packet comes across an empty link. Replication can be completed in a shorter time when the link loading is small. Moreover, the routing delay is also proportional to the link loading for a given throughput. The network delay (i.e. replication and routing delay) for the highly-loaded network is, therefore, smaller with a smaller ρ^* and P_{inj}. The tradeoff, however, is an increase in the waiting time in the packet input queues due to the small packet injection probability that has to be used.

IV. Dynamic Access Control

Instead of using a fixed packet injection probability to restrict the network load, we can vary this probability dynamically according to network traffic, which can be measured by the average link loading of the network. While it may be too complicated for the nodes to exchange loading information and obtain the global link loading, a node can easily measure the local link loading. Since both the deflection routing and general replication schemes are capable of spreading network traffic among the network nodes [4], the local link loading is a good approximation of the global one. Figure 2 shows that at a network node, all packets appearing in the links connecting the output and input processors are only passers-by. In other words, they are not originating from or destined for the node. The link loading at these links is thus independent of the packet input and output rates of this node and best reflects the actual network traffic. By averaging this local link loading over a time frame of T time slots, we have a good estimate of the network link loading. This estimate will be used to control packet input.

The dynamic access control scheme operates as follows. At each time slot, each node adjusts its own packet injection probability according to the link loading observed locally over the previous T time slots. For a particular network, we choose two values P_1 and P_2, where $P_1 > P_2$, such that if the link loading exceeds a threshold ρ_t, packet input is reduced by using $P_{inj} = P_2$. If it is smaller than ρ_t, we use $P_{inj} = P_1$ to increase packet input.

We simulated a 256-node multicast shuffle-exchange network using $\rho_t = 0.5$, $T = 3$, $P_1 = 1$ and $P_2 = 0$. With these parameters, a network node can freely inject packets if the local link loading averaged over the previous three time slots is smaller than 0.5. Otherwise, it stops packet input until more empty slots appear. Figure 6 plots the per-node throughput against the network link loading ρ_B averaged over all the nodes and over the duration of the simulation.

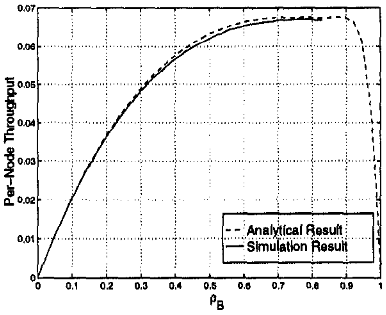


Figure 6: Simulation results of the multicast shuffle-exchange network under dynamic access control.

When the network is lightly loaded, the link loadings at the network nodes seldom go above ρ_t and packet input is rarely suspended. Network operation is mostly unaffected by the control scheme and the network throughput closely agrees with our analysis. At higher load, packet entry is stopped from time to time due to the access control scheme. The instantaneous link loading of the network varies over a wide range of values and averages to ρ_B. Figure 6 shows that this average ρ_B converges to a maximum of $\rho_B^* \simeq 0.82$. Thus, the network operates within a stable region of $0 \leq \rho_B \leq 0.82$.

The maximum average link loading ρ_B^* achieved by the network is governed by ρ_t and is numerically much larger than ρ_t. To see this, suppose the offered load is large and initially $P_{inj} = 1$. A lot of packets are introduced and the link loadings increase quickly to above ρ_t. Packet entry is suspended at most of the nodes. However, the link loadings continue to increase for some time because of packet replications. Furthermore, it takes some time for these packets to be cleared and for the link loading to decrease back to below ρ_t. As a result, the link loading is much higher than ρ_t most of the time and decreases to below ρ_t only occasionally, giving rise to an average link loading which is much larger than the threshold. This also implies that too large a threshold can cause the network to operate at very high link loading and instability may result. Therefore, a relatively small ρ_t should be used with this scheme.

The window size T is related to the responsiveness of the control scheme and must be carefully chosen. It is found that when T is as small as 1, the network is unstable. In this case, a node changes P_{inj} to 1 too frequently that it may introduce packets when it should not. However, if T is too large, the network nodes react too slowly to the increase in network load. This can cause the network nodes to operate in a synchronized and oscillatory manner. In other words, they all suspend and resume packet input periodically and almost simultaneously. Such kind of oscillations result in a network that is accessible only periodically. This is clearly undesirable and should be avoided.

Just like the static scheme, the use of a packet injection probability in the dynamic scheme allows nodes with higher offered load to have greater packet input rates. Network bandwidth can be shared efficiently among network nodes with different input traffic. Thus, by carefully choosing the parameters ρ_t, T, P_1 and P_2, the network can operate in equilibrium under all kinds of input traffic.

V. Conclusions

In this paper, we studied the stability issue in the multicast shuffle-exchange network. Instability is a common property of all multicast networks that perform packet replication and routing in the same network. Because of network instability, the operation of the multicast shuffle-exchange network has to be restricted to a small operation region. We proposed several schemes to remove deadlocks and prevent the network from becoming unstable. The static access control scheme limits the packet input rate by imposing a common and fixed packet injection probability whereas the dynamic scheme allows each node to vary this probability according to the local link loading. It was shown that these schemes can ensure stable network operation while achieving high throughput. In addition, both schemes allow efficient sharing of network bandwidth among nodes with different input traffic and are applicable under all kinds of network traffic.

References

[1] T.T. Lee, "Nonblocking Copy Networks for Multicast Packet Switching," *IEEE JSAC*, Vol. 6, No. 9, Dec. 1988, pp. 1455-1467.

[2] A. Huang and S. Knauer, "Starlite: A Wideband Digital Switch," *IEEE GLOBECOM '84*, 1984, pp. 121-125.

[3] J.S. Turner, "Design of a Broadcast Packet Switching Network," *IEEE INFOCOM '86*, 1986, pp. 667-675.

[4] S.C. Liew, "A General Packet Replication Scheme for Multicasting in Interconnection Networks," *IEEE Trans. on Commun.*, Vol. 44, No. 8, Aug. 1996, pp. 1021-1033.

[5] C.W. Chan and S.C. Liew, "Performance of Multicasting Closed Interconnection Networks," *IEEE INFOCOM '97*, 1997.

[6] S.C. Liew and T.T. Lee, "Principles of Broadband Switching and Networks," *unpublished lecture notes at The Chinese University of Hong Kong.*

[7] S.C. Liew, "On the Stability of Shuffle-Like Switching Networks with Deflection Routing," *IEEE/ACM Trans. Networking*, Feb. 1997.