On the speedup required for a multicast parallel packet switch*

Sundar Iyer, Nick McKeown

Abstract -- **A parallel packet switch (PPS) is a switch in which the memories run slower than the line rate. Arriving packets are load-balanced packet-by-packet over multiple lower speed center stage packet switches. It is known that, for unicast traffic, a PPS can precisely emulate a FCFS output-queued (OQ) switch with a speedup of two and an OQ switch with delay guarantees with a speedup of three. In this paper we ask: Is it possible for a PPS to emulate the behavior of an OQ *multicast* switch? The main result is that for multicast traffic an N-port PPS can precisely emulate a FIFO OQ switch with a speedup of $S > 2\sqrt{N} + 1$, and a switch that provides delay guarantees with a speedup of $S > 2\sqrt{2N} + 2$.**

***Keywords*--Clos' network; inverse-multiplexing; parallel packet switch; load-balancing; multicasting.**

I. Introduction

All packet switches require memories that buffer packets during times of congestion. For several years, the capacity of high performance switches and routers has been limited by the bandwidth of commercially available memories. This is, in part, because standard DRAM memories are optimized for density rather than speed of random access. Given that this situation is unlikely to change in the near future, there is interest in packet-switch architectures that overcome the memory bottleneck.[1]

The memories in well-known packet switch architectures — such as input queued (IQ) switches, output queued (OQ) and combined input and output queued (CIOQ) switches — must be capable of buffering and retrieving packets at a speed equal to, or faster than the line rate. As line rates increase (from OC48c, OC192c, to OC768c and above), it becomes difficult or impractical to buffer packets as fast as they arrive.

In attempt to overcome this problem, the PPS was proposed in [1] allowing each memory device in the switch to run slower than the line rate. The architecture is based on the 3-stage Clos Network [3] and is illustrated in Figure 1. The figure shows an example of a 4×4 PPS with a central stage consisting of three layers — each made from an output queued switches. The demultiplexor at each port connects to all layers which operate independently and in parallel. Arriving packets are sent by the demultiplexor to one of the slower speed layers. In other words, packets from each external line (operating at rate R) are sent over one of k links, each of which operates at a data rate of at least R/k. Packets are stored in the output-queues of the center stage switches and are sent to the multiplexor at the time of departure. In general, the internal links in the center stage switches operate at a rate $S(R/k)$, where S is the *speedup*.

It is interesting to know how a PPS will perform; in particular, whether it can emulate an OQ switch and its capabilities to provide guaranteed qualities of service (QoS). We answer this question in [1][2].

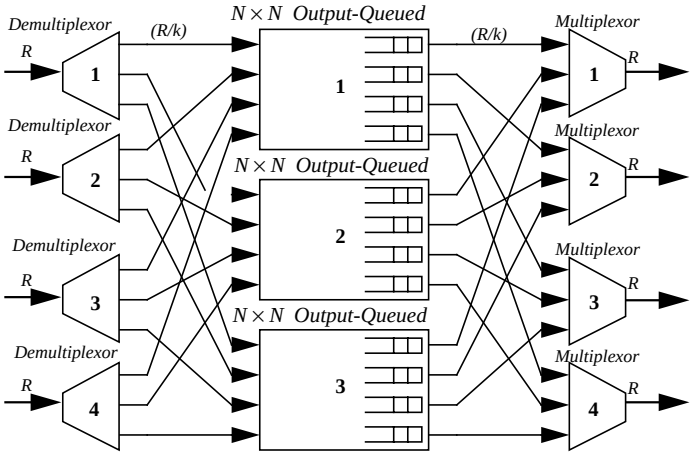

Figure 1: The architecture of a PPS based on output-queued switches.

In this paper we consider multicast traffic. There have been different multicast packet switch architectures which have been proposed in the past. A survey of these switches and the issues involved in their design can be found in [4][5]. In this paper we ask whether it is possible for a PPS to precisely emulate a multicast OQ switch.

II. Background

A. Terminology

Cell: A fixed-length packet.

Time slot: This is the time taken to transmit or receive a cell at a link rate of R.

PIFO Queues: A "Push-In First-Out" queue [6] ordering is one where arriving cells can be placed at any location, but may depart only from the head of line of the queue. PIFO queues are quite general and can be used to implement QoS scheduling disciplines such as WFQ [7], GPS [8], and strict priorities.

Fanout: The number of outputs that a multicast cell C is destined to is called the fanout and is denoted by m_C.

Maximum Fanout: The maximum number of outputs that any multicast cell can be destined to is called the maximum fanout m, where $m \in \{1, \ldots, N\}$.

Copy Multicast: A multicast cell can be copied to create multiple unicast cells at the input of a packet switch and each unicast cell can be individually switched. We call this a "copy multicast".

Fanout Multicast: In "fanout multicast" the input of the switch delivers a multicast cell into the switch fabric just once. The cen-

[1] Since, optical buffers are as yet commercially unviable, packet switches will continue to use electronic buffers for some time.

*This research was supported by the National Science Foundation, under NGI contract ANI-9872761, the Industrial Technology Research Institute (Taiwan) and the Alfred P. Sloan Foundation.

The authors are in the Computer Systems Laboratory, Stanford University, Stanford, CA 94305-9030 (e-mail: sundaes, nickm @stanford.edu)

ter stage switch takes care of delivering the cell to each output.

B. Definitions

In what follows we will need to use some definitions introduced in [1].

Definition 1: ***Input Link Constraint (ILC) -*** *Because of the data rate of the link connecting each demultiplexor to each layer, a demultiplexor can send a cell to a given layer at most once every $\lceil k/S \rceil$ time slots. This we call the input link constraint.*

Definition 2: ***Available Input Link Set AIL(i,n)*** *- This arises from the input link constraint, and is the set of layers to which demultiplexor i can start sending a cell in time slot n. Note that $|AIL(i, n)| \geq k - \lceil k/S \rceil + 1$.*

Definition 3: ***Output Link Constraint (OLC)*** *- Similarly, each layer is constrained to send a cell to each multiplexor at most once every $\lceil k/S \rceil$ time slots.*

Definition 4: ***Departure Time****- When a cell arrives, the demultiplexor selects a departure time for the cell. A cell arriving to input i at time slot n and destined to output j is assigned the departure time $DT(n, i, j)$.*

Definition 5: ***Available Output Link Set-*** *$AOL(j, DT(n, i, j))$* ***,*** *is the set of layers that can send a cell to multiplexor j at time slot $DT(n, i, j)$ in the future. Note that $|AOL(j, DT(n, i, j))| \geq k - \lceil k/S \rceil + 1$.*

C. Unicast Traffic

Theorem 1:(Sufficiency) For unicast traffic, a PPS can emulate a FCFS OQ switch with a speedup of $S \geq 2$.

Theorem 2:(Sufficiency) For unicast traffic, a PPS can emulate an OQ switch with a PIFO queueing discipline with a speedup of $S \geq 3$.

Proof: Detailed proofs are in [1].

III. Multicast Traffic

We now extend Theorems 1 and 2 to find, first, the conditions under which a PPS can emulate an FCFS OQ and a PIFO OQ multicast switch.[2]

A. Copy multicast

Lemma 1: (Sufficiency) A PPS, with maximum fanout m, can emulate a FCFS OQ switch with a speedup of $S \geq 2m$, using copy multicasting.

Proof: Since the maximum fanout of a multicast cell is m, each cell is replicated upon arrival to form m unicast cells. Thus each input of the PPS can be considered to be operating at a line rate of mR. ❒

[2] Throughout this section, we shall ignore the fact that $S \geq k$ is a trivial bound.

Lemma 2: (Sufficiency) A PPS, with a maximum fanout of m, can emulate any OQ switch with a PIFO queueing discipline with a speedup of $S \geq 3m$.

Proof: The proof is along the lines of Lemma 1 and is based on Theorem 2. ❒

B. Fanout multicast

Lemma 3: (Sufficiency) A PPS, with maximum fanout m, can precisely emulate a FCFS OQ switch with a speedup of $S \geq (m + 1)$, using fanout multicasting.

Proof: Since the maximum fanout of a multicast cell is m, each multicast cell is destined to a maximum of m outputs. Consider a cell C that arrives at demultiplexor i at time slot n and destined for output ports $\langle P_j \rangle$, where, $j \in \{1, \ldots, m\}$. For the ILC and OLC to be met, it suffices to show that there will always exist a layer l such that it meets the ILC for input port i and meets the OLC for output ports $\langle P_j \rangle$, where, $j \in \{1, \ldots, m\}$.

Thus layer l must meet all the above constraints i.e.

$$l \in \{AIL(i, n) \cap AOL(P_1, DT(n, i, P_1)) \cap AOL(P_2, DT(n, i, P_2)) \cap \ldots (AOL(P_m, DT(n, i, P_m)))\}.$$

From Definition 2 and 5 we know that,

$$|AIL(i, n)| + |AOL(P_1, DT(n, i, P_1))| + |AOL(P_2, DT(n, i, P_2))| + \ldots |AOL(P_m, DT(n, i, P_m))| > mk$$

if, $S \geq (m + 1)$. ❒

Lemma 4: (Sufficiency) A PPS, with maximum fanout m, can emulate an OQ switch with a PIFO queueing discipline, with a speedup of $S \geq (2m + 1)$, using fanout multicasting.

Proof: The proof is similar to the previous lemma. ❒

C. Devising an Optimal Strategy for Multicasting

We can make the following observations on multicasting.

1. For copy multicasting, the speedup required increases in proportion to the number of copies made per cell.
2. For fanout multicasting, a single cell is sent and no additional speedup is required to physically transmit the cell. However higher speedup is required to ensure the existence of a layer which satisfies all the constraints.

Thus, copy multicast does not use the copy capability of each layer, whereas fanout multicast does not utilize the speedup. We now show an optimum strategy which uses both forms of multicast.

1) The bounded copy strategy

Bounded copy multicast bounds the number of copies that can be made from a multicast cell. We define q as the maximum number of copies that are made from a given multicast cell. Since the maximum fanout of any given multicast cell is m at

most $\lceil m/q \rceil$ outputs will receive a copied multicast cell. We now find a lower bound on the size of the available input link set as a function of q.

Lemma 5: $|AIL(i, n)| \geq k - (\lceil k/S \rceil - 1)\, q$, *for all* $i, n \geq 0$; *in a PPS using a bounded copy strategy, where* S *is the speedup on the links connecting each demultiplexor to each layer.*

Consider demultiplexor i. The only layers that i cannot send a cell to are those which were used in the last $\lceil k/S \rceil - 1$ time slots. (The layer which was used $\lceil k/S \rceil$ time slots ago is now free to be used again). $|AIL(i, n)|$ is minimized when a cell arrives to the external input port in each of the previous $\lceil k/S \rceil - 1$ time slots. Since a maximum of q links are used in every time slot, $|AIL(i, n)| \geq k - (\lceil k/S \rceil - 1)\, q$. ❐

Theorem 3:(Sufficiency) A PPS, which has a maximum fanout of m, *can mimic a FCFS OQ switch with a speedup of* $S \geq 2\sqrt{m} + 1$.[3]

Proof: Consider a cell C that arrives at demultiplexor i at time slot n and destined for output ports $\langle P_j \rangle$, where $j \in \{1, \ldots, m\}$. This cell is divided into a maximum of q copies C_y, where $y \in \{1, \ldots, q\}$. Each copy C_y, is destined to a maximum of $\langle P_j \rangle$ distinct output ports, where, $j \in \{1, \ldots, \lceil m/q \rceil\}$. For the ILC and OLC to be met, it suffices to show that there will always exist a layer l such that the layer l meets all the following constraints for each copy C_y, i.e.

$$
\begin{aligned}
l \in \{ & AIL(i, n) \cap \\
& AOL(P_1, DT(n, i, P_1)) \cap \\
& AOL(P_2, DT(n, i, P_2)) \cap \ldots \\
& AOL(P_{\lceil m/q \rceil}, DT(n, i, P_{\lceil m/q \rceil})) \}
\end{aligned}
$$

which is satisfied when,

$$
\begin{aligned}
& |AIL(i, n)| + \\
& |AOL(P_1, DT(n, i, P_1))| + \\
& |AOL(P_2, DT(n, i, P_2))| + \ldots \\
& |AOL(P_{\lceil m/q \rceil}, DT(n, i, P_{\lceil m/q \rceil}))| > (\lceil m/q \rceil)\, k
\end{aligned}
$$

From Definition 2 and 5 we know that,

$$
\begin{aligned}
& |AIL(i, n)| + \\
& |AOL(P_1, DT(n, i, P_1))| + \\
& |AOL(P_2, DT(n, i, P_2))| + \ldots \\
& |AOL(P_{\lceil m/q \rceil}, DT(n, i, P_{\lceil m/q \rceil}))| > (\lceil m/q \rceil)\, k
\end{aligned}
$$

if,

$$
\begin{aligned}
& k - ((\lceil k/S \rceil - 1)\, q) + \\
& (\lceil m/q \rceil)\, (k - (\lceil k/S \rceil - 1)) > (\lceil m/q \rceil)\, k.
\end{aligned}
$$

This will be satisfied if,

$$k - (\lceil k/S \rceil - 1)\, q - \lceil m/q \rceil (\lceil k/S \rceil - 1) > 0$$

i.e. if, $(\lceil k/S \rceil - 1)\, (q + \lceil m/q \rceil) < k$.

Note that the above analysis applies to each copy C_y that is made in parallel. Thus each copy C_y of the multicast packet has the same input link constraint and by definition the same AIL. In the case that two or more distinct copies C_y, $y \in \{1, 2, \ldots\}$ choose the same layer l, the copies are merged and a single cell destined to the distinct outputs of each of the copies C_y is sent.

The speedup is minimized when $(q + \lceil m/q \rceil)$ is minimized. But $(q + \lceil m/q \rceil) < (q + m/q) + 1$ and so the minimum value is obtained when $q = \sqrt{m}$; *i.e.* $S \geq 2\sqrt{m} + 1$. ❐

Theorem 4:(Sufficiency) A PPS with a maximum fanout of m, *can precisely emulate an OQ switch with a PIFO queueing discipline, with a speedup of* $S \geq 2\sqrt{2m} + 2$.

Proof: The proof is almost identical to the one above.

Corollary 1:(Sufficiency) A PPS, can mimic a multicast FCFS OQ switch with a speedup of $S \geq 2\sqrt{N} + 1$ *and a multicast OQ switch with a PIFO queueing discipline with a speedup of* $S \geq 2\sqrt{2N} + 2$.

Proof: $m \leq N$. ❐

IV. Conclusions

While we cannot predict the usage and deployment of multicast, it is likely that Internet routers will be called-upon to switch multicast packets passing over very high speed lines with a guaranteed quality of service. Should this be the case, packet switches might require the characteristic of a PPS in which buffer memories need not run as fast as the line rate.

A conclusion that can be drawn from the results presented here is that the speedup required grows with the size of the allowable multicast fanouts. With small fanouts at each switch, moderate speedup suffices and delay guarantees are theoretically possible. For large fanouts, the speedup may become impracticably large.

References

[1] S. Iyer, A. Awadallah, N. McKeown, "Analysis of a packet switch with memories running slower than the line rate," in *Proc. IEEE INFOCOM '00*, pp.529-537.

[2] S. Iyer, N. McKeown, "Making parallel packet switches practical," in *Proc. IEEE INFOCOM '01*.

[3] C. Clos, "A study of non-blocking switching networks," *Bell Systems Technical Journal* 32, 1953.

[4] Gua, Ming-Huang and Ruay-Shiung Chang, "Multicast ATM switches: survey and performance evaluation," *Computer Communication Review*, Vol. 28, Number 2, April 1998.

[5] J. Turner and N. Yamanaka, "*Architectural choices in large scale ATM switches*," IEICE Trans. Communications., vol.E81-B, no.2, pp.120-137, Feb. 1998.

[6] S-T. Chuang, A. Goel, N. McKeown and B. Prabhakar, "Matching output queueing with a combined input output queued switch", *IEEE Journal of Selected Areas in Communication*, 17:1030--1039. A short version appears in *Proc. Infocom '99*.

[7] A. Demers, S. Keshav; S. Shenker, "Analysis and simulation of a fair queueing algorithm," *J. of Internetworking: Research and Experience*, pp.3-26, 1990.

[8] A. Parekh and R. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single node case," *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 344-357, June 1993.

[3] In these proofs, we derive a conservative bound on the speedup. A tighter bound can be found for specific values of m and k.