Speedup of Clos Packet Switches that Provide Delay Guarantees

Aleksandra Smiljanić, *Belgrade University, Stony Brook University, aleks@ieee.org*
Miloš Petrović, *Belgrade University, pmilos@galeb.etf.bg.ac.yu*

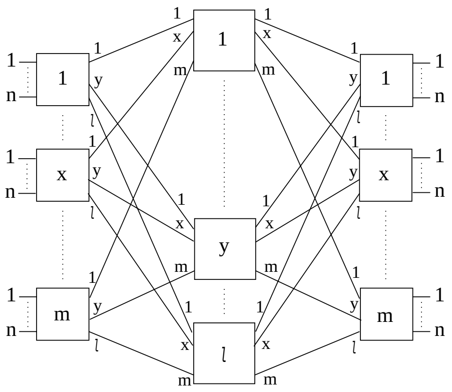


Fig. 1. Clos switching fabric

***Abstract*— A multihop Clos fabric may provide the terabit capacity by using the smaller switching elements (SE). When the traffic load is balanced over the switches in a middle stage, all the traffic would get through the fabric, as long as the switch outputs are not overloaded. We derive formulas of the maximum utilization for which certain tolerable delay is guaranteed, and formulas of the minimum speedup for which tolerable delay is guaranteed and 100% switch utilization is achieved in a Clos packet switch. Finally, performances of the architectures comprising cross-bars and shared buffers as SEs are compared and their scalability is discussed.**

I. Introduction

First generations of packet switches had output or shared buffers, and their capacity was limited by the buffer throughput. Packet switches with input buffers based on cross-bars provide higher capacity. However, terabit capacity cross-bars are still not available on the market. The capacity of a packet switch can be increased by connecting switching elements (SEs) into Clos structure. Figure 1 shows the connections between SEs in a symmetric Clos three-stage switch. The interconnection rule is: the xth SE in some switching stage is connected to the xth input of each SE in the next stage [2], [3]. The Clos switch parameters are: the number of external ports per an input or an output SE denoted by n, the number of input and output SEs denoted by m, and the number of center SEs denoted by l. It has been recognized that a Clos packet switch in which multicast sessions are balanced across the SEs provides non-blocking, i.e. with sufficiently large buffers it passes all the traffic if the outputs are not overloaded [10]. Alternatively, end-to-end sessions can be groomed into the smaller number of flows that are separately balanced [8], [9]. A cell of a flow is transmitted through the SE the number of which equals the counter value of this flow, and the counter is then incremented modulo l.

Recently, load balancing has been proposed in other architectures such as parallel plane switches (PPS) or Birkhoff-von-Neumann switches [1], [4]. Both of these architectures are a special case of Clos packet switches where the number of external ports per input and output SE is $n = 1$. In other words, input and output SEs are ports.

In this paper, we derive the switch utilization under which the tolerable delay can be guaranteed to the most sensitive applications in Clos packet switches with arbitrary parameters.

Often, the switch transceivers that transmit data to the neighboring switches, or receive data from these switches require the most complex and costly hardware. For this reason, switching fabrics have speedups so that the transmission capacity is maximally utilized. Speedup is defined as:

$$S = \frac{lmR_c}{nmR} \geq 1, \tag{1}$$

where R is the external link rate, and R_c is the internal link rate. The internal links connect the SEs, and the external links connect switch with other switches and routers. We will calculate the speedup required to ensure 100% utilization of the transmission capacity. Performance of load balancing algorithms in Clos packet switches based on cross-bars and shared buffers will be analyzed and compared.

II. Performance Analysis of Load Balancing Algorithms

Traffic of each individual flow is balanced independently across the SEs. If there are many flows that transmit cells across some SE at the same time, the cells will experience long delay. Many applications, e.g. voice and video, require stringent delay guarantees. This traffic must be policed either at the edge of the network, or at the switch ports. Policing interval equals F cell time slots. For example, input 1 negotiated to send 10Mbps to output 3, the policing interval is $F = 10^4$ cell time slots, and port bit rate 10Gbps. Then, input 1 will send at most one high-priority cell per frame to output 3. We also assume that SEs are non-blocking and provide rate and delay guarantees. So, they transfer all policed traffic within one frame period. These features hold when the shared buffers are used as SEs. They also hold for the cross-bar SEs with the speedup of two that are run by the maximal matching algorithms [5], [7]. We assume that there is a coarse synchronization in a switch, i.e. that at some point of time the controllers schedule cells belonging to the same frame. In addition, the SEs in each stage schedule packets that have arrived in the previous frame. The total delay that a cell may experience through a three-stage Clos packet switch including the resequencing time is four times the frame duration:

$$D = 4FT_c, \tag{2}$$

where T_c is the cell time slot duration. In our derivations, the number of slots per frame that can be allocated to some port is F_u. Also, all lemmas and theorems hold in large switches where $l > 10$.

Lemma 1: Let F_c denote the maximum number of cells per frame sent through some internal switch link (either from input SE to center SE or from center SE to output SE). It holds that

$$\frac{nF_u}{l} + N_f - n \leq F_c < \frac{nF_u}{l} + N_f, \tag{3}$$

where N_f denotes the number of flows passing through that internal link either sourced by some input SE or bound for some output SE.

Proof: Let f_{ig}, $0 \leq g < N_f$, denote the number of time slots per frame that are guaranteed to the individual flows sourced by SE_{1i}, where SE_{1i} denotes the ith SE in the first stage. The maximum number of cells per frame sourced by SE_{1i} fulfills:

$$F_c \leq \sum_g \left\lceil \frac{f_{ig}}{l} \right\rceil < \frac{nF_u}{l} + N_f, \tag{4}$$

where $\lceil x \rceil$ is the smallest integer no less than x, i.e. $\lceil x \rceil < x + 1$. Assume that out of N_f flows sourced by SE_{1i}, $N_f - n$ flows are assigned one time slot per frame, and the remaining n flows are assigned $nF_u - (N_f - n)$ time slots per frame. If it happens that first cells in a frame of all the flows are sent through SE_{2j} (jth SE in the second stage), the total number of cells per frame transmitted through SE_{2j} from SE_{1i} will be:

$$\begin{aligned} F_c &= N_f - n + n\lceil \frac{F_u}{l} - \frac{N_f}{nl} \rceil \\ &= \frac{nF_u}{l} + \frac{(l-1)N_f - (nF_u - N_f) \bmod (nl)}{l} \Rightarrow \\ F_c &\geq \frac{nF_u}{l} + N_f - n \end{aligned} \tag{5}$$

for $l, F_u > 10$. Claim of the lemma follows from inequalities (4,5). The proof is identical in the case of a link bound for some output SE. ■

When different flows bound for the same SE are not properly synchronized, they might send cells within a given frame starting from the same center SE. Alternatively, equal numbers of flows are balanced starting from different center SEs in each frame. For example, flow g of SE_{1i} resets its counter at the beginning of a frame to $c_{ig} = (i + g) \bmod l$. Or, flow g bound to SE_{3k} (kth SE in the third stage) resets its counter at the beginning of a frame to $c_{kg} = (k + g) \bmod l$.

Lemma 2: In load balancing algorithms with the synchronized counters, and $N_f > 10l$ or $N_f \bmod l = 0$:

$$F_c = \begin{cases} \frac{nF_u}{l} + \frac{N_f}{2} & F_u \geq \frac{lN_f}{2n} \\ \sqrt{\frac{2nF_uN_f}{l}} & \frac{10N_f}{8nl} \leq F_u < \frac{lN_f}{2n}. \end{cases} \tag{6}$$

Proof: We follow the calculation from [8] but assume $n \neq l$. We will calculate the maximum number of cells that are transmitted from SE_{1i} through $\mathrm{SE}_{2(n-1)}$ in the middle stage, and the same result would hold for any other center SE. The number of cells in flow g transmitted from SE_{1i} through $\mathrm{SE}_{2(n-1)}$ is $\left\lfloor (f_{ig} + (i+g) \bmod l)/l \right\rfloor$, where $\lfloor x \rfloor$ is the smallest integer not greater than x i.e. $\lfloor x \rfloor \leq x$. So, the number of cells transmitted from SE_{1i} through $\mathrm{SE}_{2(n-1)}$ is:

$$F_c = \sum_{0 \leq g < N_f} \left\lfloor \frac{f_{ig} + (i+g) \bmod l}{l} \right\rfloor \leq \frac{nF_u}{l} + \frac{N_f}{2}, \tag{7}$$

for $l > 10$ and $N_f > 10l$, or $l > 10$ and $N_f \bmod l = 0$. Equality in (7) can be reached iff:

$$F_u \quad \geq \quad \frac{N_f}{n} \cdot \frac{l+1}{2} \approx \frac{lN_f}{2n}, \tag{8}$$

for $l > 10$ and $N_f > 10l$, or $l > 10$ and $N_f \bmod l = 0$. If inequality (8) does not hold:

$$\begin{aligned} \frac{N_f}{l} \cdot \frac{z(z+1)}{2} \quad &\leq \quad nF_u < \frac{N_f}{l} \cdot \frac{(z+1) \cdot (z+2)}{2} \Leftrightarrow \\ z \quad &= \quad \left\lfloor \frac{-1 + \sqrt{1 + \frac{8nlF_u}{N_f}}}{2} \right\rfloor, \end{aligned} \tag{9}$$

where $0 \leq z < l$ is an integer. For $F_u \geq 10N_f/(8nl)$:

$$z \approx \sqrt{\frac{2nlF_u}{N_f}}. \tag{10}$$

It is easy to understand that F_c will be maximal for:

$$f_{ig} = \begin{cases} l - q & l - z \leq q = (i+g) \bmod l < l \\ 0 & 0 \leq (i+g) \bmod l < l - z. \end{cases} \tag{11}$$

If $10N_f/(8nl) \leq F_u < lN_f/(2n)$, from (7,10,11):

$$F_c = \frac{N_f z}{l} \approx \sqrt{\frac{2nF_uN_f}{l}}. \tag{12}$$

Claim of the lemma follows from equations (7,8,12). It can be proven identically that the claim of the lemma holds when F_c is the maximum number of cells that are transmitted to SE_{3k} through $\mathrm{SE}_{2(n-1)}$. ■

A. Switch Utilization

Theorem 1: Maximum utilization of the switch internal link is:

$$\max(0, S - \frac{4lN_fT_c}{nD}) \leq U_a \leq \min(1, S - \frac{4lN_fT_c}{nD} + \frac{4lT_c}{D}), \tag{13}$$

where D is the maximum tolerable delay.

Proof: Note that nSF/l is the number of cells that may pass the link from an input to a center SE within one frame. If it holds that

$$\begin{aligned} U_{ac} = \frac{F_u}{F} \quad &= \quad S - \frac{lN_f}{nF} \Leftrightarrow \\ \frac{nF_u}{l} + N_f \quad &= \quad \frac{nSF}{l}. \end{aligned} \tag{14}$$

from Lemma 1 it follows that

$$F_c < \frac{nF_u}{l} + N_f = nSF/l, \tag{15}$$

and all cells will pass the link within a frame for above U_{ac}. So, the maximum utilization under which all cells pass the switch is $U_a \geq U_{ac}$. From Lemma 1, $F_c \geq nF_u/l + N_f - n$, so it must hold

$$\begin{aligned} \frac{nF_u}{l} + N_f - n &\leq F_c \leq \frac{nSF}{l} \Rightarrow \\ U_a = \frac{F_u}{F} &\leq S - \frac{lN_f}{nF} + \frac{l}{F}. \end{aligned} \tag{16}$$

Theorem 1 follows from equations (15,16) when F is replaced with $D/(4T_c)$. ■

Theorem 2: Maximum utilization of the switch internal link when the counters are synchronized is:

$$U_r = \begin{cases} S - \frac{2lN_fT_c}{nD} & D \geq \frac{4lN_fT_c}{nS} \\ \frac{nS^2D}{8lN_fT_c} & D < \frac{4lN_fT_c}{nS}. \end{cases} \tag{17}$$

where D is the maximum tolerable delay, and $N_f > 10l$ or $N_f \bmod l = 0$.

Proof: Since $F_c \leq nSF/l$, from Lemma 2 it follows that for $F_u \geq lN_f/(2n)$,

$$\begin{aligned} F_c &= \frac{nF_u}{l} + \frac{N_f}{2} \leq \frac{nSF}{l} \Rightarrow \\ U_r &= \frac{F_u}{F} \leq S - \frac{lN_f}{2nF},\ F \geq \frac{lN_f}{nS} \end{aligned} \tag{18}$$

and for $10N_f/(8nl) \leq F_u < lN_f/(2n)$:

$$\begin{aligned} F_c = \sqrt{\frac{2nF_uN_f}{l}} &\leq \frac{nSF}{l} \Rightarrow \\ U_r = \frac{F_u}{F} &\leq \min(\frac{lN_f}{2nF}, \frac{nS^2F}{2lN_f}). \end{aligned} \tag{19}$$

Since $(10N_f)/(8nlF) << 1$ because $N_f \leq F$ and $l > 10$, the range $F_u < (10N_f)/(8nl)$ is not of a practical interest and was omitted in the final formula.

Formula (17) follows from equations (18,19), when F is replaced with $D/(4T_c)$. ■

Note that Theorem 2 provides the maximum utilization when both balancing of flows sourced by an input SE, and balancing of flows bound for an output SE are synchronized. This assumption will hold in all considered algorithms.

B. Switch Speedup

Often, signal transmission over the fibers connecting distant routers requires most complex and costly hardware. Therefore, it is important to provide the highest utilization of the fiber transmission capacity. For this reason, switching fabrics with the speedup have been previously proposed. We have defined the speedup in the introduction (1).

Theorem 3: The minimum speedup S required to pass all incoming packets with a tolerable delay when the counters are not synchronized is:

$$1 + \frac{4l(N_f - n)T_c}{nD} \leq S_a < 1 + \frac{4lN_fT_c}{nD}, \tag{20}$$

and the speedup when counters are synchronized, and $N_f > 10l$ or $N_f \bmod l = 0$ equals:

$$S_r \geq \begin{cases} 1 + \frac{2lN_fT_c}{nD} & D \geq \frac{2lN_fT_c}{n} \\ \sqrt{\frac{8lN_fT_c}{nD}} & D < \frac{2lN_fT_c}{n}. \end{cases} \tag{21}$$

where D is the maximum tolerable delay.

Proof: We are looking for minimum speedup such that for $F_u = F$ it holds $F_c \leq nSF/l$, where F_c is the maximum number of cells per frame passing through some internal link. When the counters are not synchronized from Lemma 1 it follows that:

$$\frac{nS_aF}{l} \geq F_c > \frac{nF}{l} + N_f - n. \tag{22}$$

Also, from Lemma 1 it follows that:

$$\frac{nS_aF}{l} \geq \frac{nF}{l} + N_f, F_u = F \Rightarrow F_c \leq \frac{nS_aF}{l}. \tag{23}$$

The above statement is equivalent to:

$$\frac{nS_aF}{l} \geq \frac{nF}{l} + N_f \Rightarrow (F_u = F \Rightarrow F_c \leq \frac{nS_aF}{l}), \tag{24}$$

meaning that S_a in (24) is sufficient. From (22,23) the minimal required speedup fulfills:

$$1 + \frac{l(N_f - n)}{nF} \leq S_a \leq 1 + \frac{lN_f}{nF}. \tag{25}$$

When the counters are synchronized, from Lemma 2 it follows that:

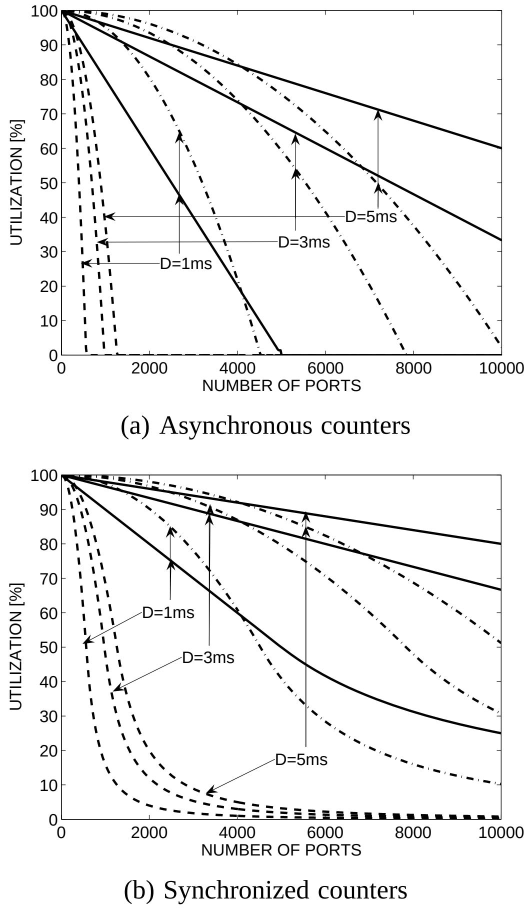
$$\frac{nS_rF}{l} \geq F_c = \begin{cases} \frac{nF}{l} + \frac{N_f}{2} & F \geq \frac{lN_f}{2n} \\ \sqrt{\frac{2nFN_f}{l}} & \frac{10N_f}{8nl} \leq F < \frac{lN_f}{2n}. \end{cases} \tag{26}$$

Formulas (20,21) follow from inequalities (25,26) when F is replaced with $D/(4T_c)$, since $F \geq N_f > 10N_f/(8nl)$ because $l \geq 2$. ■

III. Performance of Load Balancing Algorithms

It can be observed from our previous analysis that the performance of a load balancing algorithm depends on the number of flows that are separately balanced, and the tolerable delay. One way end-to-end packet delay that can be tolerated by interactive applications (e.g. conversational voice, video-conferencing, etc.) is around 150ms, but only 50-60ms of this allowed delay can be budgeted for the queuing. The switch delay below 3ms may be required for various reasons. For example, packets might pass multiple packet switches from their sources to the destinations, and delays through these switches would add. Also, in order to provide flexible multicasting, the ports should forward packets multiple times through the packet switch, and the delay is prolonged accordingly [6], [10].

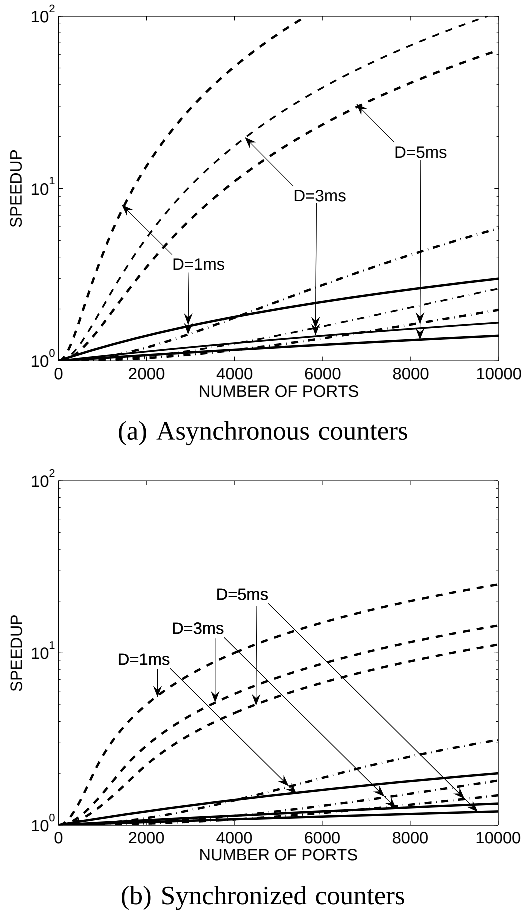
The first two load balancing schemes assume that the Clos packet switch comprises identical $n \times n$ cross-bars as SEs, i.e. $n = m = l = \sqrt{N}$. In the first algorithm a flow comprises cells from some input to some output and $N_f = nN$, while in the second algorithm a flow comprises cells from some input to some output SE and $N_f = N$. The second two load balancing schemes assume that shared buffers with equally limited throughput are used as SEs, i.e. $m = l$. In the third



(a) Asynchronous counters

(b) Synchronized counters

Fig. 2. Switch utilization: solid curves represent the second algorithm with $N_f = N$, dashed curves correspond to the fourth algorithm with $N_f = N/n,\ n = 4$ and dash-dotted curves represent the fourth algorithm with $N_f = N/n,\ n = 16$.



(a) Asynchronous counters

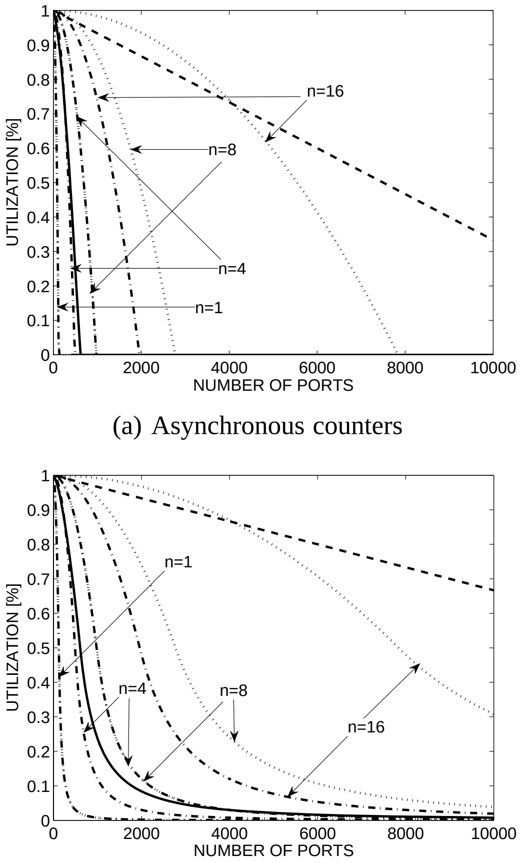
(b) Synchronized counters

Fig. 3. Fabric speedup: solid curves represent the second algorithm with $N_f = N$, dashed curves correspond to the fourth algorithm with $N_f = N/n,\ n = 4$ and dash-dotted curves represent the fourth algorithm with $N_f = N/n,\ n = 16$.

algorithm a flow comprises cells from some input SE to some output and $N_f = N$, while in the fourth algorithm a flow comprises cells from some input SE to some output SE and $N_f = m = N/n$. Note that the bounds in Theorems 1 and 3 become tight in the cases mentioned above. In Figures $2-5$ we will plot the performance of these four load balancing algorithms according to the formulas derived in Theorems 1 – 3.

We have assumed that the duration of cell time slot is $T_c = 50ns$ in further analysis (64 bytes at 10Gb/s). The second and the fourth algorithm have better performance since fewer flows are being balanced in the corresponding architectures. For that reason, Figure 2 examines utilization for these algorithms as the switch size is increasing. Figure 3 shows the fabric speedup required for the 100% utilization when the same algorithms are applied. Solid curves represent the second algorithm with $N_f = N$, dashed curves represent the fourth algorithm with $N_f = N/n,\ n = 4$ and dash-dotted curves represent the fourth algorithm with $N_f = N/n,\ n = 16$. The curves for various tolerable delays of 1,3 and 5ms are plotted. The utilization drops unacceptably when the fourth algorithm with $n = 4$ is applied and the switch size exceeds 1000 ports. The utilization is somewhat improved when the counters are synchronized, but it is still low for the larger switch sizes. The required speedup in this case increases rapidly and equals four for the switch size around 2000 ports, and the tolerable delay of 3ms. Speedup value is reduced when the counters are synchronized, the synchronization having more effect on large switch sizes. On the other hand, utilization of the second and the fourth algorithm with $n = 16$ remains as high as 70% providing a tolerable delay of 3ms even in switches with more than 5000 ports. Or, utilization is 85% in these cases when the counters are synchronized. Also, in these cases 100% utilization is provided for the fabric speedup around two even in switches with 10000 ports, guaranteeing a tolerable delay of 3ms.

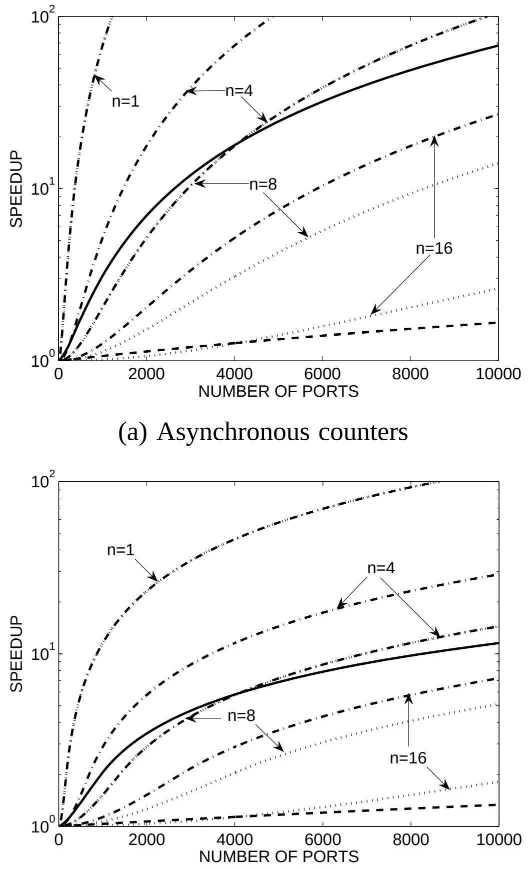
In Figures 4 and 5, the performance of all the load balancing algorithms is more closely examined for a tolerable delay of 3ms. Solid curve represents the first algorithm, dashed curve corresponds to the second algorithm, while dash-dotted curves represent the third algorithm and dotted curves represent the fourth algorithm, with $n \in \{1, 4, 8, 16\}$. Let n_3 denote the number of ports per SE for the third algorithm, and n_4 for the fourth algorithm. It is easy to see from Theorems 1, 2 and 3 that for $n_3^3 = n_4^2$ the performances of the third and the fourth algorithm will be identical and the curves in Figures 4 and 5 that correspond to such cases overlap ($n_3 = n_4 = 1$



(a) Asynchronous counters

(b) Synchronized counters

Fig. 4. Switch utilization: solid curve represents the first algorithm $N_f = nN$, dashed curve corresponds to the second algorithm $N_f = N$, dash-dotted curves represent the third algorithm $N_f = N$, and dotted curves correspond to the fourth algorithm $N_f = N/n$.



(a) Asynchronous counters

(b) Synchronized counters

Fig. 5. Fabric speedup: solid curve represents the first algorithm $N_f = nN$, dashed curve corresponds to the second algorithm $N_f = N$, dash-dotted curves represent the third algorithm $N_f = N$, and dotted curves correspond to the fourth algorithm $N_f = N/n$.

or $n_3 = 4,\ n_4 = 8$). We observe that the performance of the first and the third algorithm in which $n \leq 4$ is unacceptable. For the utilization of 70% the switch size is limited to 1500 ports in the case of the third algorithm with $n = 16$, and to 2000 ports in the case of the fourth algorithm with $n = 8$. All the algorithms require the speedups larger than two except the second algorithm and the fourth algorithm in which the counters are synchronized and $n = 16$. We can see that SEs with shared buffers should have at least 16 ports in highly scalable architecture.

IV. Conclusion

Clos packet switch with moderate speedup can meet stringent delay requirements in arbitrarily large switches, and provide 100% utilization of the switch capacity. Architecture deploying shared buffers as switching elements is efficient if these elements can support more than 16 ports. Otherwise, the preferred architecture deploys cross-bars as SEs that are more scalable, and load balancing algorithm in which the flows comprising cells from input to output SEs are separately balanced.

References

[1] C. S. Chang, D. S. Lee and C. Y. Yue, "Providing guaranteed rate service in the load balanced Birkhoff-von Neumann switches," *Proceedings of INFOCOM 2003.*

[2] C. Clos, "A study of non-blocking switching networks," *Bell Systems Technology Journal,* vol. 32, 1953, pp. 406-424.

[3] J. Hui, *Switching and Traffic Theory for Integrated Broadband Networks,* Kluwer Academic Press 1990.

[4] I. Keslassy, S. T. Chuang, K. Yu, D. Miller, M. Horowitz, O. Solgaard, N. McKeown, "Scaling Internet routers using optics," *Proceedings of ACM SIGCOMM 2003.*

[5] A. Smiljanić, "Flexible bandwidth allocation in high-capacity packet switches," *IEEE/ACM Transactions on Networking,* April 2002, pp. 287-293.

[6] A. Smiljanić, "Scheduling of multicast traffic in high-capacity packet switches," *IEEE Communication Magazine,* November 2002, pp. 72-77.

[7] A. Smiljanić, "Bandwidth Reservations by Maximal Matching Algorithms," *IEEE Communication Letters,* March 2004, pp. 177-179.

[8] A. Smiljanić, "Performance of load balancing algorithms in Clos packet switches," *Proceedings of IEEE Workshop on High Performance Switching and Routing,* April 2004, pp. 304-308.

[9] A. Smiljanić, "High performance Routers," *invited paper at joint Optoelectronic and Communication Conference and International Conference on Optical Internet,* Yokohama, Japan, July 2004.

[10] J. S. Turner, "An optimal nonblocking multicast virtual circuit switch," *Proceedings of INFOCOM 1994,* vol. 1, pp. 298-305.