

# Analysis of Queueing Displacement Using Switch Port Speedup\*

**Israel Cidon<sup>†</sup>**

Department of Electrical Engineering  
Technion - Israel Institute of Technology  
Haifa 32000, Israel  
cidon@ee.technion.ac.il

**Asad Khamisy**

SUN Microsystems Labs  
2550 Garcia Avenue  
Mountain View, CA 94043, U.S.A.  
asad@eng.sun.com

**Moshe Sidi<sup>‡</sup>**

Department of Electrical Engineering  
Technion - Israel Institute of Technology  
Haifa 32000, Israel  
moshe@ee.technion.ac.il

## Abstract

*Current high-speed packet switching systems, ATM in particular, have large port buffering requirements. The use of highly integrated ASIC technology for implementing high-degree and high-speed switch fabrics is facing a technology mismatch in the sense that today's chip technology does not allow to integrate on-chip the high-speed switching fabric with the large buffering requirements. Consequently, many designs are based on the principles of queueing displacement, i.e., they attempt to move the queueing point off-chip. This is usually done by considerably speeding-up the on-chip switch output ports and placing a second external stage of buffering between the switch fabric and the outgoing link circuitry. Such designs are very popular and are used by many current ATM switch vendors. While such schemes are widely used, no rigorous analysis has so far been offered to evaluate the design trade-offs and to quantify the design points.*

*The model we use to analyze the performance of the above system is a two-node tandem queueing system. The first node in the tandem corresponds to the internal buffer while the second node in the tandem corresponds to the external buffer. It is assumed that the internal buffer is capable to transfer  $c_1$  cells per time unit to the external buffer, while the external buffer is served at a lower rate of  $c_2$  cells per time unit.*

\*The work of I. Cidon and M. Sidi was partially supported by Intel, Israel and by the Fund for the Promotion of Research at the Technion.

<sup>†</sup>also with SUN Microsystems Labs, 2550 Garcia Avenue, Mountain View, CA 94043, U.S.A.

<sup>‡</sup>Part of the work of this author was done while visiting SUN Microsystems Labs, Mountain View CA, U.S.A.

## 1 Introduction

This paper addresses basic design and performance issues in the architecture of hardware based fast packet switches. The ability to implement cost-effective high-speed switches in silicon is (along with the advance of fiber-optic technologies) the enabling technology behind the recent rise of broadband integrated networks, in particular the ATM architecture [1, 2, 3, 4, 5, 6]. Fast hardware based switches have also been at the core of the emerging switched LAN (also termed switching hubs) products as well as the new generation of Gigabit routers [[www.bbn.com/magazine/techwatch/router.html](http://www.bbn.com/magazine/techwatch/router.html)].

Current switching system designs in high-speed packet-switching networks and in particular the emerging ATM market have large and consistently growing port buffering requirements. This trend follows the expected use of ATM networks for bursty data applications, the use of reactive flow control over ever increasing link speeds and the separation of different QoS classes and connection groups to dedicated buffers. While in the past ATM vendors have frequently offered switches with less than a hundred cell (5Kbytes) buffers, the numbers today have grown typically to more than 10K cell buffers. These numbers are expected to continue to grow with the speed of links, with the geographical distances, as well as with the use of more sophisticated buffer scheduling mechanisms. For example, Fore System [[www.fore.com/html/products/datasheets/2famcamp.html](http://www.fore.com/html/products/datasheets/2famcamp.html)] advertises buffer sizes of 53K cells (almost 3Mbytes) and 212K cells (11Mbytes) for its ASX-200 and ASX-1000 ATM switches, respectively.

Cisco [[www.cisco.com/warp/public/641/16.html](http://www.cisco.com/warp/public/641/16.html)] advertises 64K cells output buffers per link for its LightStream 2020 ATM switch. Stratacom provides 64K cells per port at its BPX switch [[www.stratacom.com/products/bpxaxis.html](http://www.stratacom.com/products/bpxaxis.html)]. The GTE [[www.gte.com/Cando/Carrier/Docs/Wired/span4000.html](http://www.gte.com/Cando/Carrier/Docs/Wired/span4000.html)] SPANet 4000 ATM switch is designed with a buffer size of 16K cells.

Most hardware based switch designs rely on the new generation of CMOS ASICs which are very popular within products which require fast turn-around and at the same time high integration and high performance. (Nevertheless, these following statements are also correct for today's fully custom VLSI designs.) The use of highly integrated ASIC technology for implementing high-degree and high-speed switch fabrics (8X8 to 32X32 with 155-622Mbps) is facing a technology mismatch in the sense that today's technology does not allow to integrate within the same ASIC the high-speed switching fabric logic of many ports with the large amount of fast memory required to buffer all switch output ports.

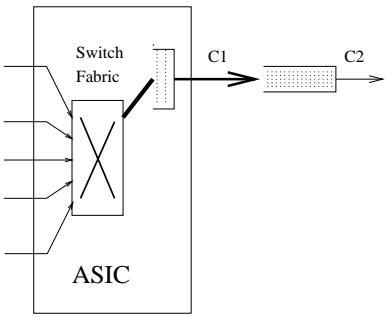


Figure 1: Speedup used to extend the memory in an ASIC based switch

Consequently, many designs are based on the principles of queueing displacement, i.e., the attempt to move the queueing point off-chip and implement the main buffer using standard RAM technology. This is usually done by considerably speeding-up the integrated switch output ports (compared to the actual link speed supported) and placing a second stage (external to the switch) of buffering between the switch fabric and the outgoing link. The much higher speed at which the internal buffer is off-loaded, considerably reduces its memory requirements for a given overflow probability design point. Such a design is very popular and is used by many current

ATM switch vendors (usually termed as *speedup*). In particular, the ones which base their buffering design on output queueing techniques. (Other designs are also possible, for example, pure input queueing or a shared memory design based exclusively on off-the-shelf RAM.) Figure 1 depicts an ASIC with a limited on-chip buffer at the switching fabric being extended with an external memory using the speedup technique. For example, the IBM switch-on-a-chip Prizma chip is limited by the number of on-chip buffers and employs speedup technique [[www.zurich.ibm.com/Technology/ATM/SWOCWPWP](http://www.zurich.ibm.com/Technology/ATM/SWOCWPWP)] and [8]. Bay Networks, uses speedup in its Lattice Cell switch design to increase the effective amount of buffers in its multi-stage switching fabric.

While speedup schemes are widely used, no rigorous analysis has so far been offered to evaluate the design trade-offs and to quantify the design points. The designer would like to evaluate the size of the internal and external buffers required for a certain design point as well as the speed of the path connecting them together. In particular, it is attractive to map (or lower bound) such dual stage designs to the performance of an "ideal" output queueing design in order to be able to compare different switches of different design points. The following works have addressed the speedup problem or have used queueing models which are related to our problem. In [10], the speedup effect was explored using a tandem system model composed of an M/M/C queue feeding a dependent M/M/1 queue. This model was investigated by extensive simulation and no rigorous analysis was carried out for the speedup problem. In the relevant tandem queues models only limited results are available. In [9] a tandem of two single server discrete-time queues with independent and identically distributed external arrival processes, where the output of one queue is fed as an additional input for the other queue is solved. This is a special case of our solution when no speedup exists and the server can serve only a single cell at a time slot. In [11, 12] Boxma analyzes two exponential single server queues in series where the first queue has a Poisson arrival process and the second is fed by the output of the first. In both queues the same customers have the same service times. In this case, again, the system in question has no speedup. In [13] a tandem of multiple queues in which message sizes are preserved is considered. Several properties of the joint work load distribution (but no full solution) are presented and proved for such a system.

The model we use to analyze the performance of the system with speed-up is a two-node tandem discrete-

time queueing system. The first node in the tandem corresponds to the internal (in-ASIC) buffer while the second node in the tandem corresponds to the external buffer. We begin the analysis of this system assuming that the external buffer can output at most one cell per slot and derive the generating function of the joint probability distribution of the lengths of the two buffers. Moments of the queue lengths are then derived. In addition, we introduce an explicit recursion to compute the joint probability distribution. From this we are able to compute tail probabilities that enable us to dimension the length of the buffers required to guarantee small loss probabilities when the buffers are finite. Speedup of two has special implementation importance as it is a very common and convenient speedup to design and use. It usually does not require tricky clock resynchronizations as either one clock is derived from the other or the path that connects the two buffer stages is of double width. Intuitively, with such speedup, the load on the first buffer stage is only half the load on the output port. In a well controlled network, no link should be overloaded (on the average) and hence the first port has a load limitation of 0.5. This rather low load should enable us to use very few buffers at the internal first stage. Our main numerical conclusion supports this intuition and shows that the internal buffer can be kept small without damaging the performance of the whole system. We extend the analysis to more general models in which the external buffer can output several cells per slot, and again obtain the generating function of the joint probability distribution of the lengths of the two buffers and the moments. The conclusions obtained are similar to the above.

## 2 The Model

The model we use to analyze the performance of the system with speed-up is a two-node tandem discrete-time queueing system. The first node in the tandem corresponds to the internal (in-ASIC) buffer while the second node in the tandem corresponds to the external buffer. Time is slotted into fixed size slots. It is assumed that during each slot the internal buffer is capable to transfer  $c_1$  cells to the external buffer, while the external buffer is capable to output from the system  $c_2$  cells. Since the internal buffer is faster than the external buffer,  $c_1 > c_2$ . Note that when  $c_2 = 1$  we have integral speedups ( $c_1 : 1$ ), while for a general  $c_2$  we can have any rational speedup ( $c_1 : c_2$ ).

Let  $A_1^{(n)}$  and  $A_2^{(n)}$  be the number of new cells arriving (from outside the system) during slot  $n$  to the first node and the second node, respectively. The arrival process of these new cells at the system is

assumed to be an independent and identically distributed (in each slot) process, namely,  $\{A_1^{(n)}, A_2^{(n)}\}$  is independent of  $\{A_1^{(k)}, A_2^{(k)}\}$  for any  $n \neq k$ , and  $Prob(A_1^{(n)} = i, A_2^{(n)} = j) = a_{i,j}$  is independent of  $n$ . Let  $A(z_1, z_2) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{i,j} z_1^i z_2^j$  be the joint generating function of the arrival processes to the two nodes. Note that arrivals to the two buffers in the same slot may be correlated. We let  $r_l = \sum_{i_1=0}^{\infty} \sum_{i_2=0}^{\infty} i_l a_{i_1, i_2} = \partial A(z_1, z_2) / \partial z_l|_{z_1=z_2=1}$  be the arrival rate of cells to node  $l$  ( $l = 1, 2$ ). For the sake of the analysis, we assume that both buffers are of infinite size. We will later discuss how this assumption can be dealt with to attain the performance of realistic switches. (Note that the speedup based output port system corresponds to the case with no external arrivals to the second node, i.e.,  $a_{i,j} = 0$ ,  $j > 0$ .)

## 3 Analysis

We begin our analysis with  $c_2 = 1$ , namely, a single cell can be transmitted by the second node and leave the system during a slot. Note that when  $c_1 = 1$ , our system is identical to the system analyzed in [9]. In our analysis we are interested in computing the steady-state probabilities of the lengths of the two nodes. Since at most one cell can leave the system during a slot, steady-state exists if  $r_1 + r_2 < 1$ .

Assume that departures take place at the end of slots and arrivals within slots. Let  $Q_1^{(n)}$  and  $Q_2^{(n)}$  denote the lengths of the first node and the second node at the end of slot  $n$ , respectively. Let  $p_{i,j}^{(n)} = Prob(Q_1^{(n)} = i, Q_2^{(n)} = j)$  and  $G^{(n)}(z_1, z_2) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} p_{i,j}^{(n)} z_1^i z_2^j$  be the generating function of the joint queue length distribution at both nodes. We let  $p_{i,j} = \lim_{n \rightarrow \infty} p_{i,j}^{(n)}$  and  $G(z_1, z_2) = \lim_{n \rightarrow \infty} G^{(n)}(z_1, z_2)$ .

The evolution equation of the queue lengths for  $n \geq 0$  is given by

$$\begin{aligned} Q_1^{(n+1)} &= (Q_1^{(n)} - c_1)^+ + A_1^{(n)} \\ Q_2^{(n+1)} &= (Q_2^{(n)} - 1)^+ \\ &\quad + \min(Q_1^{(n)}, c_1) + A_2^{(n)} \end{aligned} \quad (1)$$

From which we have in steady-state ( $n \rightarrow \infty$ ),

$$\begin{aligned} G(z_1, z_2) &= A(z_1, z_2) \{p_{0,0} + \sum_{i=1}^{c_1-1} p_{i,0} z_1^i \cdot z_1^{-i} z_2^i \\ &\quad + \sum_{i=1}^{c_1-1} \sum_{j=1}^{\infty} p_{i,j} z_1^i z_2^j \cdot z_1^{-i} z_2^{i-1} \\ &\quad + \sum_{j=1}^{\infty} p_{0,j} z_2^j \cdot z_2^{-1} + \sum_{i=c_1}^{\infty} p_{i,0} z_1^i \cdot z_1^{-c_1} z_2^{c_1} \} \end{aligned}$$

$$+ \sum_{i=c_1}^{\infty} \sum_{j=1}^{\infty} p_{i,j} z_1^i z_2^j \cdot z_1^{-c_1} z_2^{c_1-1} \} \quad (2)$$

Simple algebraic manipulations yield the following,

$$G(z_1, z_2) = A(z_1, z_2) \frac{B(z_1, z_2)}{z_1^{c_1} z_2 - z_2^{c_1} A(z_1, z_2)} \quad (3)$$

where  $B(z_1, z_2) = \sum_{i=0}^{c_1-1} [p_{i,0}(z_2 - 1) + g_i(z_2)](z_1^{c_1} z_2^i - z_1^i z_2^{c_1}) + G(z_1, 0)z_2^{c_1}(z_2 - 1)$  and  $g_i(z_2) = \frac{1}{i!} \frac{\partial G^i(z_1, z_2)}{\partial z_1^i}$  at  $z_1 = 0$ .

In (3), we encounter a common phenomenon in dependent queues, namely, that the generating functions  $G(z_1, z_2)$  is expressed in terms of several boundary functions. In order to uniquely determine  $G(z_1, z_2)$  we will have to determine the boundary functions,  $G(z_1, 0)$  and  $g_i(z_2)$ ,  $0 \leq i \leq c_1 - 1$ . In what follows, we develop the method for obtaining these boundary functions. Along this process we mainly use the analytic properties of the generating functions  $G(z_1, z_2)$  in the disk  $|z_i| < 1$ ,  $i = 1, 2$ .

We begin by letting  $z_2 \rightarrow 0$  in (2) to obtain  $G(z_1, 0) = A(z_1, 0)[p_{0,0} + p_{0,1}]$ . When  $z_1 = 0$  we have that

$$p_{0,0} = a_{0,0}(p_{0,0} + p_{0,1}) \quad (4)$$

Therefore we can express  $G(z_1, 0)$  (and the corresponding probabilities  $p_{i,0}$ ,  $i \geq 1$ ) in terms of the constant  $p_{0,0}$  as follows,

$$G(z_1, 0) = A(z_1, 0) \frac{p_{0,0}}{a_{0,0}} ; \quad p_{i,0} = a_{i,0} \frac{p_{0,0}}{a_{0,0}} \quad i \geq 1 \quad (5)$$

To determine  $g_i(z_2)$ ,  $0 \leq i \leq c_1 - 1$ , we use the fact that for any  $|z_2| < 1$ , the denominator  $z_1^{c_1} z_2 - z_2^{c_1} A(z_1, z_2)$  of (3) has exactly  $c_1$  roots within the unit disk  $|z_1| < 1$  (the proof is based on Rouche's Theorem). Let these roots be denoted by  $\sigma_l(z_2)$ ,  $1 \leq l \leq c_1$ . Since  $G(z_1, z_2)$  is an analytic function in the disk  $|z_i| < 1$ ,  $i = 1, 2$ , the nominator of (3) must vanish at each of these roots, i.e.,

$$B(\sigma_l(z_2), z_2) = 0 , \quad 1 \leq l \leq c_1 \quad (6)$$

where in the above we use  $p_{i,0}$  from (5) and  $G(\sigma_l(z_2), 0) = A(\sigma_l(z_2), 0)p_{0,0}/a_{0,0}$ . Expression (6) is a set of  $c_1$  (linear) equations that determines the functions  $g_i(z_2)$ ,  $0 \leq i \leq c_1 - 1$  up to the constant  $p_{0,0}$ .

To complete the derivation of the generating function we need to compute that constant. To that end, let  $z_1 = z_2 = z$  in (3) to obtain,

$$\begin{aligned} G(z, z) &= A(z, z) \frac{G(z, 0)(z - 1)}{z - A(z, z)} \\ &= A(z, z) \frac{A(z, 0)p_{0,0}(z - 1)}{a_{0,0}[z - A(z, z)]} \end{aligned} \quad (7)$$

Letting  $z \rightarrow 1$  in the above and assuming that the steady-state condition  $r_1 + r_2 < 1$  holds, we obtain by using L'Hôpital's law,

$$p_{0,0} = \frac{a_{0,0}(1 - r_1 - r_2)}{A(1, 0)} \quad (8)$$

This completes the derivation of the joint generating function of the queue-length probability distribution.

Using the generating function we can obtain the average number of cells in node  $i$  ( $i = 1, 2$ ), denoted by  $L_i$ . Letting  $z_2 = 1$  in (3) we have

$$G(z_1, 1) = A(z_1, 1) \frac{\sum_{i=0}^{c_1-1} g_i(1)(z_1^{c_1} - z_1^i)}{z_1^{c_1} - A(z_1, 1)}$$

Taking the derivative of the above expression with respect to  $z_1$  and letting  $z_1 \rightarrow 1$  we obtain,

$$L_1 = r_1 + \frac{\sum_{i=0}^{c_1-1} g_i(1)(c_1^2 - i^2) + r_1 - c_1^2 + \hat{A}}{2(c_1 - r_1)} \quad (9)$$

where  $\hat{A} = d^2 A(z, 1)/dz^2|_{z=1}$ .

Taking the derivative of (7) with respect to  $z$  and letting  $z \rightarrow 1$  we obtain (using (8)),

$$L_1 + L_2 = r_1 + r_2 + \frac{dA(z, 0)}{dz}|_{z=1} + \frac{\frac{d^2 A(z, z)}{dz^2}|_{z=1}}{2(1 - r_1 - r_2)} \quad (10)$$

Subtracting (9) from (10) we obtain the average number of cells in the second node  $L_2$ .

#### 4 The Probability Distribution

The joint generating function derived in the previous section allows us to compute  $p_{0,0}$  and the average quantities easily. However, the computation of the probabilities themselves is difficult. In this section we develop a recursion for the computation of the steady-state probabilities  $p_{n_1, n_2}$ . From equation (4) we obtain  $p_{0,1} = p_{0,0}(1 - a_{0,0})/a_{0,0}$ . Together with equations (5) and (8) we have the initial conditions for the recursion, i.e., we know  $p_{i,0}$ ,  $i \geq 0$  and  $p_{0,1}$ .

Next, we compute the probabilities  $p_{n_1, n_2}$  for  $n_1 \geq 0$  and  $1 \leq n_2 \leq c_1 - 1$  (except for  $p_{0,1}$  that is known already). In order to have  $n_2$ ,  $1 \leq n_2 \leq c_1 - 1$ , cells in the second queue, the number of cells in the first queue at the end of the previous slot have to be less or equal to  $n_2$ . Assume the number of cells in the first queue is  $i$ ,  $0 \leq i \leq c_1 - 1$ , then in order to have  $n_1$  cells in the first queue, all  $n_1$  cells have to arrive from the external source. Further, assuming that  $j$  cells arrive to the second queue from the external source,

the total number of cells that arrive to the second queue is  $i + j$ . If the second queue is not empty at the end of the previous slot, then in order to have  $n_2$  cells, the number of cells at the end of the previous slot have to be  $n_2 + 1 - i - j$ , where the plus 1 accounts for the cell that must have departed the second queue. We have for  $n_1 \geq 0$ ,  $1 \leq n_2 \leq c_1 - 1$  that

$$\begin{aligned} p_{n_1, n_2} &= \sum_{i=0}^{n_2} p_{i,0} a_{n_1, n_2-i} \\ &+ \sum_{i=0}^{n_2} \sum_{j=0}^{n_2-i} p_{i, n_2+1-i-j} a_{n_1, j} \end{aligned} \quad (11)$$

where the first sum corresponds to the case where the second queue is empty at the end of the previous slot. Note that  $n_1$  appears in the right hand side of equation (11) only in  $a_{n_1, l}$  for some  $l$  (and does not appear in any of the  $p$ 's). In equation (11) we compute the probabilities  $p_{n_1, n_2}$  in increasing order of  $n_2$  as follows. We assume that  $p_{i,j}$  for  $i \geq 0$ ,  $0 \leq j \leq n_2 - 2$  as well as  $p_{0, n_2-1}$  are known (initially, for  $n_2 = 2$  this is true as explained at the beginning of this section). Next, we compute  $p_{0, n_2}$ ,  $p_{1, n_2-1}$  by solving two equations with two unknowns. These equations correspond to equation (11) with  $p_{0, n_2-1}$  and  $p_{1, n_2-1}$  in the left hand side. Note that the unknown  $p_{0, n_2}$  appears only in the right hand side of (11), while the unknown  $p_{1, n_2-1}$  appears also in the left hand side. After some simple algebra we obtain the solution for  $p_{0, n_2}$ ,  $p_{1, n_2-1}$  as,

$$\begin{aligned} p_{0, n_2} &= (\beta_{1, n_2}(1)a_{0,0} - \beta_{1, n_2}(0)(a_{1,0} - 1))/a_{0,0} \\ p_{1, n_2-1} &= (\beta_{1, n_2}(0)a_{1,0} - \beta_{1, n_2}(1)a_{0,0})/a_{0,0} \end{aligned} \quad (12)$$

where we define

$$\begin{aligned} \beta_{1, n_2}(l) &\triangleq 1\{l = 0\}p_{0, n_2-1} \\ &- \sum_{i=0}^{\min(c_1, (n_2-1))} \sum_{j=1}^{n_2-i} p_{i, n_2-i-j} a_{l, j-1\{j=n_2-i\}} \\ &- \sum_{i=2}^{\min(c_1, (n_2-1))} p_{i, n_2-i} a_{l, 0} \quad l = 0, 1 \end{aligned} \quad (13)$$

Now the probabilities  $p_{n_1, n_2-1}$  for  $n_1 \geq 2$  are computed directly from equation (11). Recursing this procedure completes the computation of the probabilities  $p_{n_1, n_2}$  for  $n_1 \geq 0$  and  $1 \leq n_2 \leq c_1 - 1$ .

Next, we compute the probabilities  $p_{n_1, n_2}$  for  $n_1 \geq 0$  and  $n_2 \geq c_1$ . There are two cases to consider in the recursion, the first is when the number of cells in the first queue is less than  $c_1$  in which case all cells move to the second queue, and the case where the number of cells in the first queue is greater or equal to  $c_1$  in

which case only  $c_1$  cells move from the first queue to the second queue. Similar to the previous recursion, we have for  $n_1 \geq 0$ ,  $n_2 \geq c_1$ ,

$$\begin{aligned} p_{n_1, n_2} &= \sum_{i=0}^{c_1-1} p_{i,0} a_{n_1, n_2-i} \\ &+ \sum_{i=0}^{c_1-1} \sum_{j=0}^{n_2-i} p_{i, n_2+1-i-j} a_{n_1, j} \\ &+ \sum_{i=c_1}^{n_1+c_1} p_{i,0} a_{n_1+c_1-i, n_2-c_1} \\ &+ \sum_{i=c_1}^{n_1+c_1} \sum_{j=0}^{n_2-c_1} p_{i, n_2+1-c_1-j} a_{n_1+c_1-i, j} \end{aligned} \quad (14)$$

In the same way as before, we write two equations for  $p_{0, n_2-1}$  and  $p_{1, n_2-1}$ , from which we obtain the probabilities  $p_{0, n_2}$ ,  $p_{1, n_2-1}$ . Then we use equation (14) to compute the probabilities  $p_{n_1, n_2-1}$  for  $n_1 \geq 2$ . The solution for  $p_{0, n_2}$ ,  $p_{1, n_2-1}$  in this case is given by:

$$\begin{aligned} p_{0, n_2} &= (\beta_{2, n_2}(1)a_{0,0} - \beta_{2, n_2}(0)(a_{1,0} - 1))/a_{0,0} \\ p_{1, n_2-1} &= (\beta_{2, n_2}(0)a_{1,0} - \beta_{2, n_2}(1)a_{0,0})/a_{0,0} \end{aligned} \quad (15)$$

Where we define  $\beta_{2, n_2}(0) \triangleq \beta_{1, n_2}(0)$  and  $\beta_{2, n_2}(1) \triangleq \beta_{1, n_2}(1) + \sum_{i=0}^{n_2-c_1} p_{c_1+1, n_2-c_1-i} a_{0, j-1\{j=n_2-c_1\}}$ . Now the probabilities  $p_{n_1, n_2-1}$  for  $n_1 \geq 2$  are computed directly from equation (14). This completes the computation of the probabilities  $p_{n_1, n_2}$  for  $n_1 \geq 0$  and  $n_2 \geq c_1$ .

The computation complexity of the probabilities  $p_{n_1, n_2}$  for  $0 \leq n_1 \leq N_1$  and  $0 \leq n_2 \leq N_2$  is in the order of  $O(N_1^2 N_2^2)$ .

#### 4.1 Overflow Probability

Since we use infinite buffers in our model, we approximate the cell loss probability by the buffer overflow probability. The first (second) node is in overflow state when the number of cells in the buffer exceeds  $N_1$  ( $N_2$ ).

If the state upon arrival is  $(n_1, n_2)$ , then the number of cells in the second queue in front of this cell, when this cell arrives to the second queue, will be  $n_1 + n_2 - \sum_{i=1}^{\lceil(n_1+1)/c_1\rceil} A_2(i) - \lceil(n_1+1)/c_1\rceil$ , where  $A_2(i)$  denotes the number of cells that arrive to the second queue in slot number  $i$  (the slot of arrival of the cell is marked as slot number 1). This is true since 1. The cell will arrive at the second queue  $\lceil(n_1+1)/c_1\rceil$  slots after it arrived to the system, and 2.  $c_1 > 1$  and hence the second queue will not empty in the next  $\lceil(n_1+1)/c_1\rceil$  slots after the arrival of this cell.

Correspondingly, we define the overflow probability in the system as the probability that the number of

cells in the first and second nodes at slot boundaries fulfil the relation  $(n_1, n_2) \in \{(n_1, n_2) \mid n_1 > N_1, n_1 + n_2 - [(n_1 + 1)/c_1] > N_2\}$ .

Next, we consider a system without external arrivals to the second queue, i.e.,  $a_{i,j} = 0, j > 0$ . Then, the overflow probability in the system is equal to

$$P_{\text{overflow}} = 1 - \sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2 + \lceil(n_1+1)/c_1\rceil - n_1} p_{n_1, n_2} \quad (16)$$

## 5 The case $c_2 > 1$

We now consider a system with  $c_2 > 1$ , namely, several cells ( $c_2$ ) can be transmitted by the second node and leave the system during a slot. As before, we are interested in computing the steady-state probabilities of the lengths of the two nodes. Since  $c_2 > 1$  cells can leave the system during a slot, steady-state exists if  $r_1 + r_2 < c_2$ .

With the same notations as in Section 3 the evolution equation of the queue lengths for  $n \geq 0$  is given by

$$\begin{aligned} Q_1^{(n+1)} &= (Q_1^{(n)} - c_1)^+ + A_1^{(n)} \\ Q_2^{(n+1)} &= (Q_2^{(n)} - c_2)^+ \\ &\quad + \min(Q_1^{(n)}, c_1) + A_2^{(n)} \end{aligned} \quad (17)$$

Similar to the method for obtaining (2), we obtain in this case,

$$\begin{aligned} G(z_1, z_2) &= A(z_1, z_2) \left\{ \sum_{i=0}^{c_1-1} \sum_{j=0}^{c_2-1} p_{i,j} z_1^i z_2^j \cdot z_1^{-i} z_2^{-j} \right. \\ &\quad + \sum_{i=0}^{c_1-1} \sum_{j=c_2}^{\infty} p_{i,j} z_1^i z_2^j \cdot z_1^{-i} z_2^{i-c_2} \\ &\quad + \sum_{i=c_1}^{\infty} \sum_{j=0}^{c_2-1} p_{i,j} z_1^i z_2^j \cdot z_1^{-c_1} z_2^{c_1-j} \\ &\quad \left. + \sum_{i=c_1}^{\infty} \sum_{j=c_2}^{\infty} p_{i,j} z_1^i z_2^j \cdot z_1^{-c_1} z_2^{c_1-c_2} \right\} \quad (18) \end{aligned}$$

and after simple algebraic manipulations we get,

$$G(z_1, z_2) = A(z_1, z_2) \frac{B(z_1, z_2)}{z_1^{c_1} z_2^{c_2} - z_2^{c_1} A(z_1, z_2)} \quad (19)$$

$$\begin{aligned} \text{where } B(z_1, z_2) &= \sum_{i=0}^{c_1-1} \left[ \sum_{j=0}^{c_2-1} p_{i,j} (z_2^{c_2} - z_2^i) + g_i(z_2) \right] (z_1^{c_1} z_2^i - z_1^i z_2^{c_1}) \\ &\quad + \sum_{j=0}^{c_2-1} f_j(z_1) z_2^{c_1} (z_2^{c_2} - z_2^j) \text{ and} \end{aligned}$$

$$g_i(z_2) = \frac{1}{i!} \frac{\partial G^i(z_1, z_2)}{\partial z_1^i} |_{z_1=0}, f_j(z_1) = \frac{1}{j!} \frac{\partial G^j(z_1, z_2)}{\partial z_2^j} |_{z_2=0}.$$

In order to uniquely determine  $G(z_1, z_2)$  we will have to determine the boundary functions,  $f_j(z_1)$ ,  $0 \leq j \leq c_2 - 1$  and  $g_i(z_2)$ ,  $0 \leq i \leq c_1 - 1$ . In what follows, we demonstrate the method for obtaining these boundary functions. Along this process we mainly use the analytic properties of the generating functions  $G(z_1, z_2)$  in the disk  $|z_i| < 1$ ,  $i = 1, 2$ .

We begin by taking the  $l$ -th derivative ( $0 \leq l \leq c_2 - 1$ ) of both sides of (18) with respect to  $z_2$  and let  $z_2 \rightarrow 0$  to obtain,

$$f_l(z_1) = \sum_{k=0}^l \frac{\mathcal{A}_{l-k}(z_1) u_k}{(l-k)!} \quad (20)$$

$$\text{where } \mathcal{A}_i(z_1) = \frac{\partial A^i(z_1, z_2)}{\partial z_2^i} |_{z_2=0}, \text{ and } u_k = \sum_{j=0}^{c_2} p_{k,j} + \sum_{j=0}^{k-1} p_{j,c_2+k-j}.$$

Consequently,  $f_l(z_1)$  (and the corresponding probabilities  $p_{i,l}$ ,  $i \geq 0$ ) are expressed in terms of the constants  $u_k$ ,  $0 \leq k \leq l$ . To determine these constants, we let  $z_1 = z_2 = z$  in (19) to obtain,

$$G(z, z) = A(z, z) \frac{\hat{B}(z)}{z^{c_2} - A(z, z)} \quad (21)$$

where  $\hat{B}(z) = \sum_{j=0}^{c_2-1} (z^{c_2} - z^j) \sum_{k=0}^j \mathcal{A}_{j-k}(z) u_k / (j-k)!$ . Letting  $z \rightarrow 1$  in the above and assuming that the steady-state condition  $r_1 + r_2 < c_2$  holds, we obtain

$$c_2 - r_1 - r_2 = \sum_{j=0}^{c_2-1} (c_2 - j) \sum_{k=0}^j \frac{\mathcal{A}_{j-k}(1) u_k}{(j-k)!} \quad (22)$$

Furthermore, the denominator  $z^{c_2} - A(z, z)$  of (21) has one root at  $z = 1$  and exactly  $c_2 - 1$  roots within the unit disk  $|z| < 1$  (the proof is based on Rouché's Theorem). Let these roots be denoted by  $\mu_l$ ,  $1 \leq l \leq c_2 - 1$ . Since  $G(z, z)$  is an analytic function in the disk  $|z| < 1$ , the nominator of (21) must vanish at each of these roots, i.e., for  $1 \leq l \leq c_2 - 1$ ,

$$\sum_{j=0}^{c_2-1} (\mu_l^{c_2} - \mu_l^j) \sum_{k=0}^j \frac{\mathcal{A}_{j-k}(\mu_l) u_k}{(j-k)!} = 0 \quad (23)$$

Equation (22) together with (23) is a set of  $c_2$  (linear) equations that determines the constants  $u_k$ ,  $0 \leq k \leq c_2 - 1$ , and hence the boundary functions  $f_j(z_1)$ ,  $0 \leq j \leq c_2 - 1$  are determined (using (20)). From  $f_j(z_1)$  we can obtain  $p_{i,j}$ ,  $j \geq 0$  by taking the  $i$ -th derivative of  $f_j(z_1)$  with respect to  $z_1$  and letting  $z_1 \rightarrow 0$ .

To determine  $g_i(z_2)$ ,  $0 \leq i \leq c_1 - 1$ , we use the same procedure described in Section 3. In particular, we note that for any  $|z_2| < 1$ , the denominator  $z_1^{c_1} z_2^{c_2} - z_2^{c_1} A(z_1, z_2)$  of (19) has exactly  $c_1$  roots within the

unit disk  $|z_1| < 1$  (the proof is based on Rouché's Theorem). Let these roots be denoted by  $\sigma_l(z_2)$ ,  $1 \leq l \leq c_1$ . Since  $G(z_1, z_2)$  is an analytic function in the disk  $|z_i| < 1$ ,  $i = 1, 2$ , the nominator of (19) must vanish at each of these roots, i.e.,

$$B(\sigma_l(z_2), z_2) = 0 \quad 1 \leq l \leq c_1 \quad (24)$$

Expression (24) is a set of  $c_1$  (linear) equations that determines the functions  $g_i(z_2)$ ,  $0 \leq i \leq c_1 - 1$ .

This completes the derivation of the joint generating function of the queue-length probability distribution. In [14] we provide the procedure for computing the joint probability distribution in this case.

Using the generating function we can obtain the average number of cells in node  $i$  ( $i = 1, 2$ ), denoted by  $L_i$ , as was done in Section 3. In fact, the average number of cells in the first node does not change with  $c_2$ , hence is identical to (9). Taking the derivative of (21) with respect to  $z$  and letting  $z \rightarrow 1$  we obtain,

$$\begin{aligned} L_1 + L_2 &= r_1 + r_2 + \frac{\frac{d^2 A(z, z)}{dz^2}|_{z=1}}{2(c_2 - r_1 - r_2)} \\ &+ \frac{\hat{\mathcal{A}} - c_2(c_2 - 1)}{2(c_2 - r_1 - r_2)} \end{aligned} \quad (25)$$

where  $\hat{\mathcal{A}} = \sum_{j=0}^{c_2-1} \sum_{k=0}^j \frac{u_k}{(j-k)!} \{[c_2(c_2 - 1) - j(j - 1)]\mathcal{A}_{j-k}(1) + 2(c_2 - j)d\mathcal{A}_{j-k}(z)/dz|_{z=1}\}$ . Subtracting (9) from (25) we obtain the average number of cells in the second node  $L_2$ .

## 6 Numerical Results

In the first example cells arrive to the system (through the first node only) according to a Poisson process with rate  $\lambda$ . We set  $c_1 = 2$  and  $c_2 = 1$ . In Figure 2 we plot the overflow probability in the system versus the internal buffer size for external buffer size of 80 cells and load of  $\lambda = 0.9$ . We also plot the overflow probability in the internal buffer, external buffer and in an "ideal" output buffer switch with  $c_1 = \infty$ . From Figure 2 we notice that for internal buffer size greater or equal to 15, the overflow probability in the system is less than the overflow probability in the "ideal" system. Also, the main contribution to the total overflow probability for small (large) internal buffer sizes comes from the internal (external) overflow probability. Notice that the internal buffer reduces the burstiness of the input process to the external buffer by limiting the maximum number of cells that can arrive to the external buffer to  $c_1 = 2$ , and hence the overflow probability in the system and in the external buffer is smaller than the overflow probability in the "ideal" system.

We obtained numerical results for different external buffer sizes and loads where the overflow probability in the "ideal" system was kept at  $10^{-7}$ . In all examples, an internal buffer size between 12 and 14 cells was sufficient to bring the overflow probability in the system below  $10^{-7}$ . The same phenomena described in Figure 2 were observed in all examples.

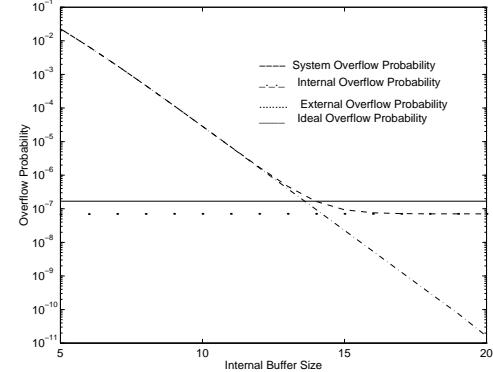


Figure 2: Overflow probability versus internal buffer size for load of 0.9 and external buffer size of 80 cells.

More numerical results are provided in Figure 3 where we plot the overflow probability versus the average load for internal buffer size of 15 and external buffer size of 40. We see again that internal buffer size of 15 is enough to achieve smaller overflow probability in the system compared to the "ideal" system, for average load of 0.75 or larger (for the range of  $\geq 10^{-9}$  overflow probability this is the relevant load range). Numerical results for the case  $c_2 > 1$  are provided in [14].

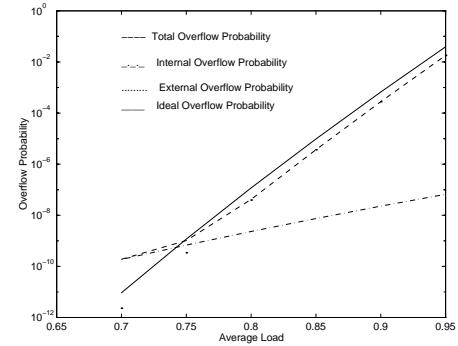


Figure 3: Overflow probability versus average load for internal buffer size of 15 and external buffer size of 40.

We also simulated a system with bursty traffic model that corresponds to VBR traffic class. Here we used three groups of sources. Each group consists of 10 identical sources. Each source corresponds to Interrupted Poisson Process (IPP) model, with sources in the first group having an average ON period of 100 slots and average OFF period of 100 slots. The second and third group sources have average (ON, OFF) periods of (100, 300) and (100, 900), respectively. The average load in the system is termed "VBR Load" and the total load of each group is "VBR Load"/3. Table 6 contains the overflow probability in the "ideal" system and in the system versus "VBR Load" for different internal buffer sizes. The conclusions are similar to the Poisson model.

VBR Load	Ideal	IB = 8	IB = 10	IB = 15	IB = 20
0.62	$2.42 \cdot 10^{-4}$	$3.05 \cdot 10^{-4}$	$2.54 \cdot 10^{-4}$	$2.38 \cdot 10^{-4}$	$2.38 \cdot 10^{-4}$
0.66	$7.15 \cdot 10^{-4}$	$8.08 \cdot 10^{-4}$	$7.31 \cdot 10^{-4}$	$7.07 \cdot 10^{-4}$	$7.07 \cdot 10^{-4}$
0.70	$1.68 \cdot 10^{-3}$	$1.81 \cdot 10^{-3}$	$1.70 \cdot 10^{-3}$	$1.66 \cdot 10^{-3}$	$1.66 \cdot 10^{-3}$
0.74	$3.35 \cdot 10^{-3}$	$3.56 \cdot 10^{-3}$	$3.39 \cdot 10^{-3}$	$3.31 \cdot 10^{-3}$	$3.31 \cdot 10^{-3}$
0.78	$6.07 \cdot 10^{-3}$	$6.29 \cdot 10^{-3}$	$6.09 \cdot 10^{-3}$	$6.02 \cdot 10^{-3}$	$5.61 \cdot 10^{-3}$
0.82	$1.04 \cdot 10^{-2}$	$1.06 \cdot 10^{-2}$	$1.04 \cdot 10^{-2}$	$1.03 \cdot 10^{-2}$	$1.03 \cdot 10^{-2}$

Table 1: Overflow probability versus VBR Load for external buffer size of 100.

## References

- [1] J. S. Turner, "Design of an integrated services packet network," *IEEE Journal on Selected Areas in Comm.*, SAC-4:1373–1380, Nov. 1986.
- [2] I. Cidon and I. S. Gopal, "PARIS: An approach to integrated high-speed private networks," *International Journal of Digital and Analog Cabled Systems*, 1(2):77–85, April 1988.
- [3] J. N. Giacopelli and et al, "Sunshine: A high-performance self-routing broadband packet switch architecture," *IEEE Journal on Selected Areas in Comm.*, SAC-9:1289–1298, Oct. 1991.
- [4] P. Newman, "A fast packet switch for the integrated services backbone network," *IEEE Journal on Selected Areas in Comm.*, SAC-6:1468–1479, 1988.
- [5] Y. S. Yeh, M. G. Hluchyj, and A. S. Acampora, "The knockout switch: A simple, modular architecture for high performance packet switching," *IEEE Journal on Selected Areas in Comm.*, SAC-5:1274–1283, Oct. 1987.
- [6] J. Hui, "A broadband packet switch for multi-rate services," *Proc. of International Conference on Comm. (ICC'87)*: 782–788, June 1987.
- [7] AT&T Microelectronics, *Microelectronic Products Selection Guide*, AT&T Microelectronics, Jan. 1995.
- [8] W. E. Denzel, A. P. J. Engbersen, I. Iliadis, and G. Karlsson, "A highly modular packet switch for GB/S rates," *Proc. of International Switching Symposium*, 2:A8.3, Oct. 1992.
- [9] J. A. Morrison, "Two discrete-time queues in tandem," *IEEE Trans. on Comm.*, COM-27:563–573, 1979.
- [10] J. S. C. Chen and T. E. Stern, "Throughput analysis, optimal buffer allocation, and traffic imbalance study of a generic nonblocking packet switch," *IEEE Journal on Selected Areas in Comm.*, SAC-9, 3:439–449, 1991.
- [11] O. J. Boxma, "On a tandem queueing model with identical service times at both counters, I," *Adv. Appl. Prob.*, (11):616–643, November 1979.
- [12] O. J. Boxma, "On a tandem queueing model with identical service times at both counters, II," *Adv. Appl. Prob.*, (11):644–659, November 1979.
- [13] S. B. Calo, "Message delays in repeated-service tandem connections," *IEEE Trans. on Comm.*, COM-29:670–678, May 1981.
- [14] I. Cidon, A. Khamisy and M. Sidi, "Analysis of Queueing Displacement Using Switch Port Speedup," CC PUB #170, October 1996.