From *Proceedings of the Fourth International Conference on Computer Communications and Networks*, Las Vegas, USA, September 20-23, 1995, pp. 344-349

LOSS PROCESS ANALYSIS OF THE KNOCKOUT SWITCH USING STOCHASTIC ACTIVITY NETWORKS

Latha Kant and William H. Sanders
Center for Reliable and High-Performance Computing
University of Illinois at Urbana-Champaign
Urbana, IL 61801 USA
{lkant, whs}@crhc.uiuc.edu

Abstract The proposed use of Asynchronous Transfer Mode (ATM) in B-ISDNs necessitates fast packet switches (FPS) capable of providing adequate quality of service (QoS). The Knockout switch is an FPS with low delay and cell loss at the switch level. However, the performance perceived by a specific input is not simple to determine, since the often computed cell loss probability (CLP) is a time averaged value obtained with respect to the switch. An analysis of the *distribution* of *consecutive cell losses* seen at a tagged port would be more useful. In this paper, we examine this distribution under a wide range of traffic burstiness for the Knockout switch using Markov processes generated automatically from a stochastic activity network (SAN) representation. The results provide useful information on the performance at a specific input, as well as illustrate the usefulness of SANs in modeling and analyzing telecommunication switch designs.

Keywords: Asynchronous transfer mode, Broadband ISDN, Cell loss probability, Distribution of consecutive cell loss, Fast packet switch, Stochastic activity networks, Stochastic Petri nets.

1 Introduction

ATM-based B-ISDNs necessitate fast packet switches (FPS) capable of providing acceptable Quality of Serive (QoS) to the wide range of services that they are expected to support. Research in the FPS arena [1] has resulted in a variety of switch architectures. The Knockout switch [15] is an FPS with low delay and cell loss probabilities (CLP) under uniformly-routed steady-input traffic. Other studies on the Knockout architecture include: (a) analysis with variable length packets and uniform routing in [3], (b) analysis with non-uniform routing in [16], and (c) simulation and implementation of a prioritized Knockout switch in [4].

Both [3] and [16] consider i.i.d. Bernoulli inputs to the switch, which is not appropriate in B-ISDNs, where the traffic is bursty. The authors in [4] considered bursty traffic, but since simulation was used, they were limited to small specific concentrator sizes and high error rates to avoid extremely long simulation times. Further, all prior work on the Knockout switch has concentrated on the overall switch CLP averaged over time. However, the manner in which cells are lost is important, especially to certain applications (for eg., recursively encoded video streams fed to a video port). Some work on loss bunching has been done for simple queues [2, 9, 10, 13, 14], but none on the distribution of consecutive cell loss at a particular port and with respect to a specific FPS architecture.

This paper specifically address this issue and analytically examines the distribution of consecutive cell losses for a tagged port and the switch CLP of the Knockout mechanism with a wide range of bursty traffic. While examining consecutive cell losses, we present results for the fraction of loss bursts of length k, where k is a parameter and a loss burst of length k implies consecutive loss of k cells. We use stochastic activity networks (SANs) [8] to model the switch and workload. SANs, a variant of stochastic Petri nets, allow the representation of alternative switch designs and workloads in a compact manner, and automatic construction of an underlying stochastic process representation that can be solved numerically. We use *UltraSAN* [12], a SAN-based modeling and analysis tool, to generate and solve the required Markov process. The resulting stochastic processes are on the order of tens of thousands states and are solved easily on a typical workstation. The results are significant in that they provide useful information to the switch/network designer, as well as illustrate that SANs provide a useful framework for evaluating telecommunication switch designs.

2 Switch and workload models

The Knockout switch (see [15] for description) is modeled by obtaining the precise sequence of events that occur upon service completion (SC) for each cell. Let n represent a time at which we make a discarding decision, and N the number of input and output ports. Since the ports operate synchronously, and service is non pre-emptive with fixed and deterministic times at

which discarding decisions are made, the sequence of events at each output port is: (a) the *(n-1)th* SC occurs marked by removing the cell in service during the *(n-1)th* SI. (b) high speed discarding (Knockout) decision performed in negligible time. (c) newly admitted cells join the output queue, and (d) the *nth* SI begins, representing the service time for the cell at the head of the output queue.

Since there is no sharing between output queues in the Knockout switch, we may restrict our attention to a single output port. In particular, let the state of the system in the *nth* service interval be $X_n = (T_n, D_n, M_n)$, where: (a) T_n is the number of cell arrivals to the given output port, (b) D_n is the number of cells discarded and (c) M_n is the number of cells remaining in the output queue in the *nth* SI. Further, as the system is viewed at discrete time intervals and the next state X_{n+1} depends only upon the current state X_n, the process $\{X_n \mid n \in I\!N\}$ is a discrete-time Markov chain (DTMC).

Defining the cell loss probability at time n (CLP_n) as the fraction of the number of arriving cells discarded, we have

$$CLP_n = E[D_n]/E[T_n] \tag{1}$$

where $E[\cdot]$ denotes the expectation. These expectations are obtained by solving for the state occupancy probabilities $\pi^{(n)}_{i,j,k}$, where $\pi^{(n)}_{i,j,k}$ denotes the probability that the system is in state $X_n = (i,j,k)$ at time n. The denominator in (1) is the traffic intensity ρ. Further, since the DTMC is finite state, irreducible, aperiodic and time homogeneous, the limiting probabilities exist and are independent of initial state. Thus, $\pi_{i,j,k} = \lim_{n\to\infty} \pi^{(n)}_{i,j,k}$ and (1) becomes

$$CLP = \frac{\sum_{i,j,k}\ j\ \pi_{i,j,k}}{\rho}. \tag{2}$$

Equation (2) gives the limiting CLP as seen by the switch.

To calculate the loss seen at a particular port, additional states are required to tag the port. This is done by augmenting the state description of the system to include contributions from the tagged port. Using superscript t to denote the tagged port, the state of the system in the *nth* SI is X^t_n, which has D^t_n and T^t_n in addition to T_n, D_n and M_n in its state description. From this we have (similar to (1)),

$$CLP^t_n = E[D^t_n]/E[T^t_n]. \tag{3}$$

As before, when the limiting probabilities exist, CLP^t is calculated similar to that done in (2), with the denominator now being ρ^t, and the numerator being only those cells discarded from the tagged stream.

Regarding the workload model, due to the bursty nature of traffic in a B-ISDN, we employ a two-state MMPP as in Figure 1 to account for clusters of cell arrivals. Many bursty traffic models exist, including the 2-state [6, 7] and 3-state [11] models . However, as our goal is to analyze switch robustness and sensitivity to burstiness rather than study a particular traffic model, we use the 2-state model [6, 7]. In Figure 1, states 0 and 1 denote idle and active states, respectively. While active, a cell is emitted with probability q, and while idle no cell arrives. The holding time in the two states is geometrically distributed, and the transition probabilities within and between states are as shown in Figure 1.

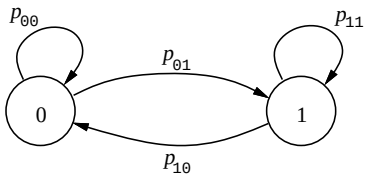

Figure 1: Bursty traffic model

The average time spent in an active period E_{active} (ON time) in cell times is

$$E_{active} = \frac{1}{p_{10}}. \tag{4}$$

Since arrivals occur with probability q only during active periods, the average offered load ρ is

$$\rho = \frac{p_{01}}{p_{01} + p_{10}} \times q. \tag{5}$$

To characterize bursty traffic, we use the average burst size (BS) and activity factor (AF) as in [6, 7]. Defining the AF as the fraction of time the workload is active, we have

$$AF = \frac{E_{active}}{E_{active} + E_{idle}} = \frac{p_{01}}{p_{01} + p_{10}}, \tag{6}$$

and the average burst size is

$$BS = E_{active} \times q = \frac{1}{p_{10}} \times q. \tag{7}$$

The effect of bursty traffic is studied by varying the burst size (BS), activity factor (AF) and cell emission probability (q). Thus (6),(7) and (8) help in selecting the appropriate Markov chain parameters while analyzing CLP and consecutive cell loss distribution with bursty inputs.

3 SAN Representation

Given the above description, we can construct a stochastic activity network (SAN) representation of the switch [8].

A SAN representation for the workload is given in Figure 2 while that for the switch is shown in Figure 3. SANs, a variant of stochastic Petri nets, consist of four primitives: *places, activities, input gates* and *output gates*. Places, represented by circles, represent the "state" of the modeled system and may contain *tokens*. Activities ("transitions" in Petri net terminology) represent actions of the modeled system, and are of two

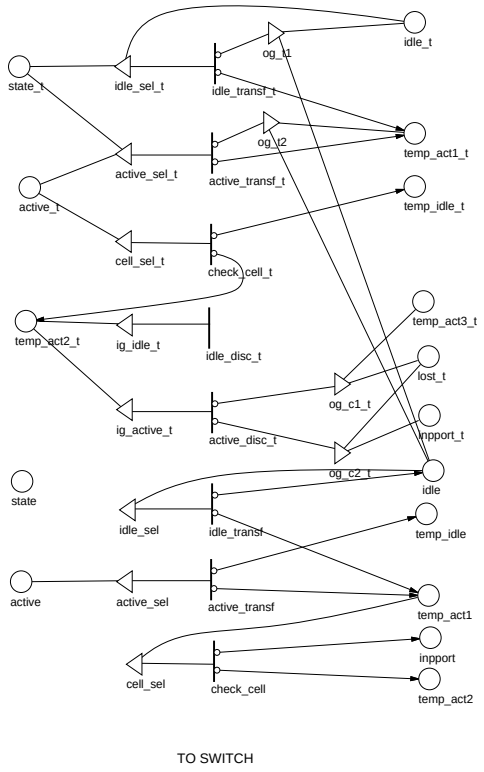

Figure 2: SAN model for bursty workload and a tagged port

types: (a) *Timed* and (b) *Instantaneous*. Timed activities (denoted by hollow vertical bars) have durations which impact the performance of the modeled system. For eg., the service of an ATM cell, represented by *service* in Figure 3. Instantaneous activities (denoted by a thick vertical bar) represent actions that complete in a negligible amount of time compared to other activities in a system. For eg., *admit_inp* in Figure 3 represents high speed hardware switching and routing within the FPS.

Case probabilities, represented as small circles on the right side of an activity, represent uncertainty associated with the completion of an activity, with each case representing a possible outcome. For eg., *idle_transf_t* in Figure 2 has two cases. *Input gates* are used to enable activities, while both input and output gates help change the state of the system. In Figure 2, *trig_inp* and *admit_inp* are examples of input and output gates, respectively.

In Figure 2, the upper portion represents the workload for a tagged input while the lower portion denotes the remaining (N-1) inputs. Places *idle* and *active* represent the silent and active spurts of the input. Activities *idle_transf* and *active_transf* together with their four cases represent the transition probabilities between these two states, i.e., p_{00}, p_{01}, p_{10} and p_{11}. If an input is active, *check_cell* is enabled to determine the presence

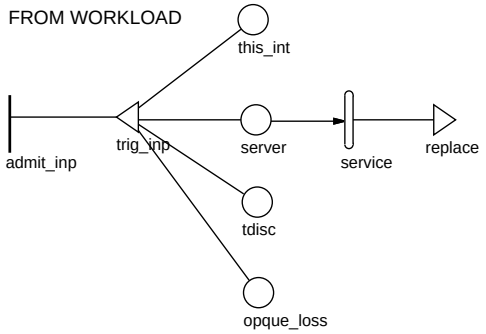

Figure 3: SAN model for switch

(with probability q) or absence (with probability $(1-q)$) of a cell. In the former case, *inpport* is incremented.

A similar explanation holds for the upper half of Figure 2, with the activity, place and gate names now having an "*_t*" to represent the tagged input. The tagged case however has a few extra SAN primitives. Activity *active_disc_t* with its two cases and enabled by *ig_active_t* represents the probability of either losing or not losing a tagged cell given that the tagged input had a cell over the current SI. The probability that the tagged cell is discarded is assumed to be equally likely with respect to all other cells. Output gates *og_c1* and *og_c2* then either increment or reset *lost_t*, which represents consecutive losses for the tagged input, and *inpport_t* is incremented if the tagged cell is not lost. If the tagged input had no cell over the SI, *ig_idle_t* enables *idle_disc_t* and resets *lost_t*.

Figure 3 represents the switch portion of the SAN. Input gate *trig_inp* and activity *admit_inp* implement the discarding decisions at the concentrator and output buffer. After the inputs have all been examined, *trig_inp* enables *admit_inp* and performs the knockout algorithm by doing the following: (a) If the number of arrivals stored in *this_int* exceeds L, *tdisc* which represents the cells lost at the concentrator is updated, else it is set to zero. (b) Following this, *opque_loss* is incremented if the cells passing the concentrator do not have space at the output queue, else is set to zero. Timed activity *service* represents the fixed time required to serve an ATM cell. Output gate *replace*, which executes upon every service completion, removes the cell that was in service, clears *this_int, tdisc* and *opque_loss* and triggers the workload for next SI by placing a token in *state_t*.

4 Model Solution and Discussion of Results

Model analysis with *UltraSAN* involves the construction and solution of the continuous time Markov process (CTMP) of the corresponding SAN. The switch model discussed in Sections 2 and 3 was represented by a DTMC. However, the steady-state solution of a DTMC with a single timed (deterministic) activity with time t and probability matrix $\mathbf{P}$ is equivalent to the so-

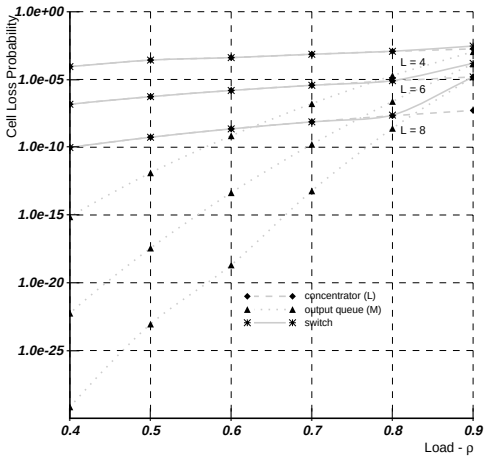

Figure 4: Effect of concentrator size on CLP with steady traffic

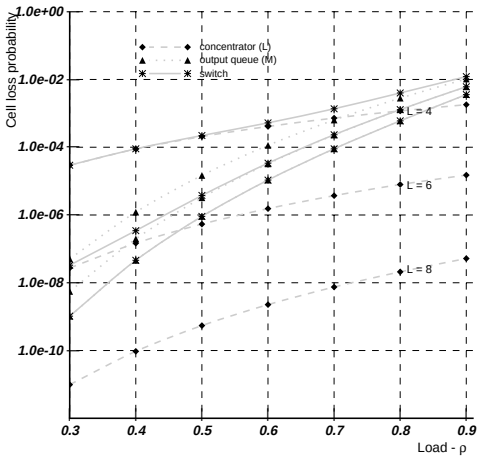

Figure 5: Effect of concentrator size on CLP with bursty traffic

lution of the CTMP with rate matrix $\mathbf{A}$ and a single exponential activity given by the relation $\mathbf{P} = \mathbf{A} \cdot t + \mathbf{I}$. This equivalence allows us to use *UltraSAN* to generate the corresponding Markov process. Once generated, the steady state probability distribution vector is determined numerically using successive over-relaxation (SOR), stopping the algorithm when the difference between the most recent and previous iteration is less than 10^{-9}. This is then used to determine the CLP and distribution of consecutive cell losses.

Based on the relative mix of (a) the number and type of applications multiplexed (b) the port speed, and (c) the aggregate load as seen at a port, the overall traffic to a port may have AFs ranging from very low (peaked) to large (smoothed) values. A similar phenomenon with respect to the load as seen on a LAN is described in [5, 6], where the authors discuss LAN workload characterization. Hence a variety of studies are conducted with varying burstiness (AF, BS and q). We present our results in two subsections. Subsection 4.1 discusses switch-based measures for a 16x16 switch. Subsection 4.2 examines CLP and distribution of consecutive cell losses for a tagged port with homogeneous sources. We use an 8-input switch for this analysis due to the increased complexity when tagging a specific input.

4.1 Switch based measures

Roles of concentrator and output queue with steady and bursty input traffic Figures 4 and 5 display CLPs at the concentrator, output queue and switch with steady and bursty traffic, respectively. In the bursty input case, the AFs are varied from 0.27 to 0.81 to vary ρ. BS was set to 8 and q to 0.07. Observe that the concentrator rather than the output queue contributes significantly to the CLP up to $\rho = 0.8$ in Figure 4, while the roles reverse based on AF and L in Figure 5 (with bursty traffic). In the bursty case, with small AFs, the concentrator knocks out the clusters before they reach the output queue, becoming the bottleneck (in Figure 5 for L=4, AF ≤ 0.55 and $\rho < 0.7$). However with higher AFs, the burstiness decreases, and hence the same concentrator size no longer knocks off a significant number of cells causing the output queue to become the bottleneck (L=4, $AF > 0.55$ and $\rho \geq 0.7$). With a larger concentrator size (L $\geq$ 6), the concentrator itself knocks out few cells at all AFs, thus not contributing significantly to CLP.

Effect of AF on CLP Figure 6 demonstrates the effect of AF on CLP. The AF is varied from 0.2-0.8 with BS fixed (8) and three ρ's (0.3, 0.5 and 0.8) to provide the effect of multiplexing different types and numbers of sources. The figure indicates a rapid deterioration in CLP with increasing burstiness, despite a fixed average load. However, observe that low AFs do not always imply poor CLP, since low AFs with low loads are not as detrimental as low AFs and high loads. This suggests the need for efficient policing and CAC techniques, which take into account differing AF-load pairs and react appropriately, rather than police and admit on AF or load alone. (A similar result is observed when the mean ON times were varied from 48 to 2048 cells with a fixed average load ρ, to study the effect of varying congestion levels. The figures have not been included due to space limitations.)

Effect of buffer size Figure 7 displays the effect of buffer sizing on CLP illustrating the significance of AFs while investing in buffers to improve CLP and QoS. ρ is fixed at 0.8 and BS at 8, and three AFs (0.8, 0.5 and 0.3) are used. While CLP improvement by increasing buffer size is fairly good for high AFs (0.8), it is not significant with lower AFs (0.3) which exhibit a fairly steep slope, despite the same average load. (The trends at lower average loads, which are not included

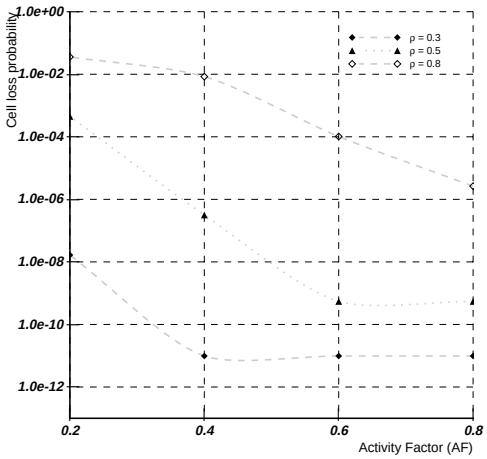

Figure 6: Effect of increasing AF on CLP

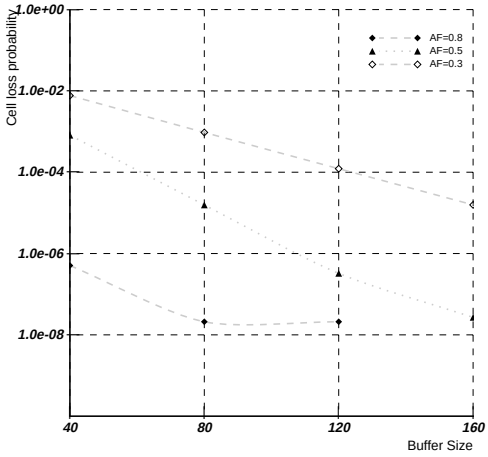

Figure 7: Effect of buffer size on CLP with bursty traffic

due to space constraints, follow their higher load counterparts.) This indicates the necessity for combining prioritized discarding and congestion control strategies along with buffer sizing to guarantee acceptable QoS with bursty traffic, especially critical with low AFs and medium to high loads.

4.2 Tagged input based measures

Effect of AF and E_{active} on the distribution of consecutive cell losses The effect of burstiness (both varying AFs and mean ON times) with a fixed average load on the distribution of consecutive cell losses is shown in Figures 8 and 9, respectively. In Figure 8, BS is fixed at 8 and the AF in Figure 9 is fixed at 0.3. ρ in both cases is 0.8.

While increasing burstiness in both cases produces longer consecutive loss lengths, observe that the effect of decreasing AFs is more significant than increasing

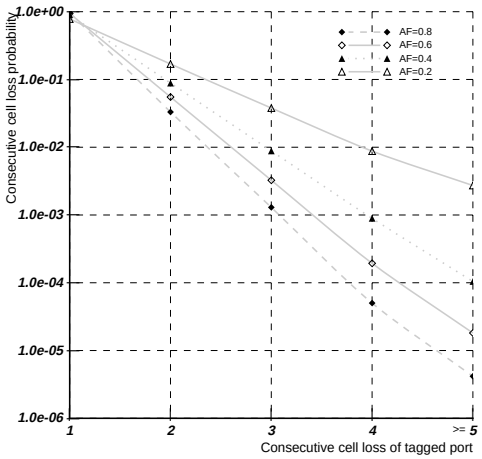

Figure 8: Effect of AF on distribution of consecutive cell losses of a tagged port

BS. Further, the slopes with lower AFs become much steeper than those with higher AFs, leading to widely differing values for the consecutive loss lengths in Figure 8. While consecutive cell losses of length ≥ 5 have fairly small probabilities ($\leq 10^{-4}$ for $AFs > 0.4$), they change significantly for low AFs (≤ 0.4). These curves therefore highlight the severity of burstiness arising from small AFs despite a fixed average load.

Effect of buffer size on the distribution of consecutive cell losses Figure 10 displays the effect of larger buffers with a fixed ρ (0.8), fixed BS (8) and three AFs (0.8,0.5 and 0.3). Observe that the size of the buffers does not influence the distribution of consecutive cell losses. This is since once the buffer is full, losses occur until the buffer is no longer full, regardless of its absolute size. What bigger buffers imply is that the onset of congestion, so to speak, is delayed, thus emphasizing on efficient discarding strategies rather than buffer sizing alone to improve loss behavior.

5 Conclusion

In conclusion, this paper makes two contributions. First, we illustrate the appropriateness of stochastic activity network models for analytically predicting the performance ATM switch designs. By representing the offered workload and switch architectures as a stochastic activity network, we were able to accurately characterize its functioning and automatically generate a Markov process representation of its behavior. The size of the resulting Markov process makes them unreasonable to generate by hand, but simple to generate and solve, numerically, using existing SAN-based evaluation software. The results show that it is indeed possible to model realistic switch architectures and workloads as SANs, and solve them, numerically, in reasonable time.

Second, we examine the distribution of consecutive

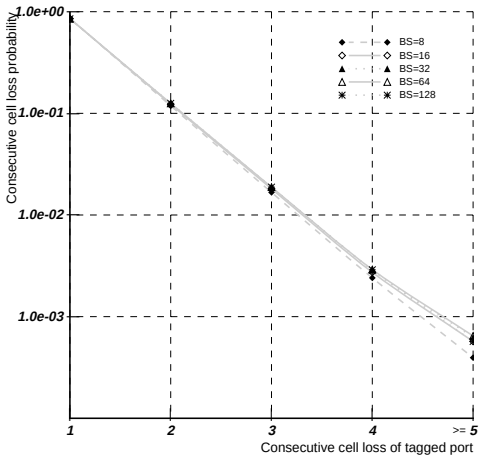

Figure 9: Effect of E_{active} on distribution of consecutive cell losses of a tagged port

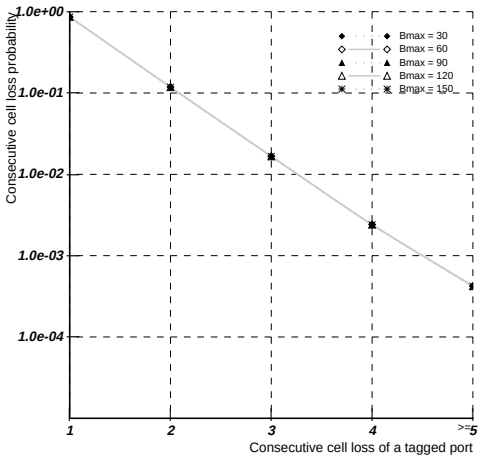

Figure 10: Effect of buffer size on distribution of consecutive cell losses of a tagged port

cell losses for a specific port through the Knockout switch, as well as the robustness of the Knockout mechanism for a wide range of burst parameters. This provides useful information to switch designers, as well as about performance at a tagged input.

References

[1] H. Ahamadi and W. E. Denzel, "A Survey of Modern High-Performance Switching Techniques," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 1091-1103, 1989.

[2] A. B. Bondi, "On the bunching of cell losses in ATM-based networks," *GLOBECOM'91*, 14.1.1-14.1.4, pp. 444-447, 1991.

[3] K. Y. Eng, M. G. Hluchyj, and Y. S. Yeh, "A Knockout Switch for variable-length packets," *IEEE Journal on Selected Areas in Communications*, vol. 5, pp. 1426-1435, 1987.

[4] J. B. Evans, E. Duron, and Y. Wang, "Analysis and Implementation of a Priority Knockout Switch," *IEEE INFOCOM*, pp. 1090-1106, 1993.

[5] K. M. Khalil, K. Q. Lue, and D. V. Wilson, "LAN Traffic analysis and workload characterization," *Proceedings. 15th conference on Local Computer Networks*, pp. 112-122, 1991.

[6] K. M. Khalil and Y. S. Sun "The effect of bursty traffic on the performance of local area networks," *GLOBECOM'92*, pp. 597-603, 1992.

[7] S.-Q. Li and J. W. Mark, "Traffic Characterization for Integrated Services Networks," *IEEE Transactions on Communications*, no. 38, pp. 1231-1243, 1990.

[8] J. F. Meyer, A. Movaghar, and W. H. Sanders, "Stochastic activity networks: Structure, behavior, and, application," *Proc. International Workshop on Timed Petri nets*, pp. 106-115, 1985.

[9] H. Ohta and T. Kitami, "Simulation study of the cell discard process and the effect of cell loss compensation in ATM networks," *The transactions of the IECE*, vol. E73, no. 10, pp. 1704-1710, 1990.

[10] O. Osterbo, "Duration of heavy load states in an ATM network," *Queueing, Performance and Control in ATM, ITC-13*, pp. 91-95, 1991.

[11] J. M. Pitts, Z. Sun, J. P. Cosmas, E. M. Scharf, "Burst-Level Teletraffic Modelling: Applications in Broadband Network Studies," *Third IEE Conference on Telecommunications*, pp. 348-352, 1991.

[12] W. H. Sanders, W. D. Obal, M. A. Qureshi, and F. K. Widjanarko, "The *UltraSAN* Modeling Environment," accepted for publication in *Performance Evaluation Journal, special issue on Performance Modeling Tools*, 1995.

[13] H. Schulzrinne, J. F. Kurose and D. F. Towsley, "Loss correlation for queues with bursty input streams," *SUPERCOM/ICC'92*, pp. 219-224, 1992.

[14] T. Takine, T. Suda, and T. Hasegawa, "Cell loss and output process analyses of a finite-buffer discrete-time ATM queueing system with correlated arrivals," *IEEE INFOCOM*, 10c.3.1-10c.3.11, pp. 1259-1269, 1993.

[15] 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 Communications*, no. 5, pp. 1274-1283, 1987.

[16] H. Yoon, M. T. Liu, and K. Y. Lee, "The Knockout Switch Under Nonuniform Traffic," *GLOBECOM'88*, pp. 1628-1634, 1988.