Architecture and Performance Analysis of the Multicast Balanced Gamma Switch for Broadband Communications[1]

Cheng Li, *Member, IEEE,* R. Venkatesan, *Senior Member, IEEE,* and H. M. Heys, *Member, IEEE*
Faculty of Engineering and Applied Science
Memorial University of newfoundland
St. John's, NL, A1B 3X5, Canada

***Abstract*— This paper presents the architecture design as well as the performance analysis of a new cell-based multicast switch for broadband communications. Using distributed control and a modular design, the Balanced Gamma (BG) switch features a high performance for unicast, multicast and combined traffic under both random and bursty conditions. Although it has buffers on input and output ports, the multicast BG switch follows predominantly an output-buffered architecture. The performance is studied under uniform and non-uniform multicast traffic in terms of cell loss ratio and cell delay. The results are compared with those from an ideal pure output-buffered multicast switch to demonstrate how close its performance is to that of the ideal but impractical switch. Comparisons with other published switches reveals the superior of the BG switch and the tradeoffs between complexity and performance in a packet switch design. It is shown that the multicast BG switch achieves a performance close to the ideal switch while keeping hardware complexity reasonable.**

***Index Terms*— Multicast, Balanced Gamma (BG) switch, performance analysis, multistage interconnection network (MIN), self-routing, self-replication, complexity, cell loss ratio, cell delay.**

I. Introduction

Communication network capacity and applications have been changing at an enormous rate for the past two decades driven by the Internet and multimedia applications. The trend still remains the same. Besides very high speed, many real-time applications, such as videoconferencing, music on demand, and video on demand require messages to be sent to more than one destination. As a result, in addition to high throughput and low delay, multicast has become a necessary feature for any switch designed for future broadband communication networks.

Many possible multicast switch architectures have been explored since the late 1980s, such as Lee's multicast switch [1], Turner's broadcast switch [2], the PINIUM switch [3], the ABACUS switch [4], [5], and the input-queued (virtual output queued) switch [6]. Due to the many desirable features such as self-routing, distributed control, modularity, constant delay for all input-output pairs and suitability for VLSI implementation, multistage interconnection network (MIN) design becomes attractive future solution for broadband switch architecture. ATM-like fixed-size packet switching attracts much interest because of its application in high-speed Internet routers and switches. The switch fabric internally operates on the fixed-size packets called cells. The incoming variable-size IP packet (datagram) is internally segmented into cells that are transmitted to the output port, where they are reassembled into the IP datagram. In this paper, we use the term cell to identify the fixed-size packet used in the switch, which can be ATM cells, or any other convenient data format [7].

In this paper, we study the architecture design as well as the performance analysis of a new cell-based multicast switch that has input and output buffers, a backpressure mechanism and a very high throughput. The switch is called the multicast Balanced Gamma (BG) switch, which utilizes a multi-path MIN design. Cell replication to achieve multicast is integrated into the functionality of each switch element and is performed in a distributed fashion along with the routing. This paper is organized as follows. Section II introduces the design considerations and the architecture of the multicast BG switch. Section III presents the multicast traffic models. Section IV describes the performance analysis of the BG switch under various traffic conditions. The performance measures are compared to those from the ideal multicast switch, the PINIUM switch, and the Abacus switch. Conclusions are presented in Section V.

II. Design Considerations and Switch Architecture

A. Approaches to Construct Multicast Switch

Using MIN design, the multicast function can be achieved either by placing a copy network at the front of the routing network (*cascade approach*) or by integrating the cell replication function into the switch element (SE) of the MIN (*integrated approach*).

The cascade approach is an intuitive approach. The copy network replicates cells according to the fanout number specified in the header. The routing network uses the output of the copy network as its input and routes each copy to its destination. Many of proposed multicast switches follow this approach [1], [2], [8]. A typical example is Lee's multicast switch [1]. The basic structure of the copy network of Lee's multicast switch consists of a running adder network, dummy address encoders, a broadcast banyan network and trunk

1. This work is supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

number translators. Theoretically, any routing network can be used to route cells from an output port of the copy network to an output port of the multicast switch. However, Lee's multicast switch suffers from two problems. One is overflow, resulting when the total requested number of copies exceeds the available number of output ports of the copy network. In this situation, any cell whose fanout is larger than the remaining free output ports will be dropped [1], [2]. This will eventually decrease system performance and throughput. The other problem is the output port conflict problem in the routing network when multiple cells request the same output port simultaneously. Besides these two problems, the memory size of the trunk number translation tables will increase significantly as the fanout and the switch size increase. Some modifications are suggested to improve the design [1], [2]. However, they only mitigate the situation at best while increasing the hardware complexity.

The integrated approach combines the routing and replication functions into a single unified network. To minimize the load that multicast cells bring to the switch, cell replication will be performed only when necessary within the switch fabric as the cell is routed through. This kind of design will inherit most of the attractive features of the MIN design. Typical switch examples include the MOBAS switch [9], the Abacus switch [4], [5], and the PINIUM switch [3]. The problems encountered by the cascade approach no longer exist in the integrated approach. Even though each individual SE must be enhanced to handle both functions, which will increase its hardware complexity slightly, the overall complexity of the switch fabric is normally less than the sum of the copy and routing network because many resources originally required by both networks are now shared, such as the memory components which are used to store the routing and replication tags. Besides the advantage of reduced hardware complexity, the characteristics of reliability, scalability, and fault tolerance in the single unified network solution are also easier to improve. All these benefits make the integrated solution attractive for new architectures of the next-generation multicast switches. The multicast BG network proposed in this paper utilizes the integrated approach.

B. Justification for Buffering Choices

Blocking is a problem with which every switch design must deal. Blocking can happen either internally when cells contend for the same internal links, or at the output port when multiple cells request the same output port in one switching cycle. To improve the throughput and mitigate blocking, many solutions have been proposed. The effects of internal blocking can be minimized by providing internal buffers or by providing input buffers and incorporating a backpressure mechanism. The former solution is expensive and leads to out-of-sequence receipt of cells in multi-path networks. The second solution is capable of handling incoming traffic efficiently and helps to maintain a high throughput of the switch, and thus is preferred in broadband communication networks. Output blocking can be controlled by choosing a MIN architecture capable of accepting multiple cells at each output line in each switching cycle. Such MINs, along with input buffers and a backpressure mechanism, would be capable of overcoming the drop in performance due to internal blocking as well as output blocking. Any input-buffered switch architecture may suffer from head-of-line (HOL) blocking, in which the temporarily un-transmissible cell at the head of the input buffers impedes the transmission of cells behind it and thus reduces the switch throughput. However, if the MIN possesses very high throughput, e.g., better than 99%, the HOL blocking does not significantly degrade performance because most of the cells can be delivered immediately without being buffered and delayed [10], [11].

Output buffered switches have been shown to provide the best delay and throughput performance [4]. It is costly to achieve a pure output buffered switch because the output lines have to operate N times as fast as a normal link, where N is the size of the switch. However, with a moderate speed-up for the output queue and a high throughput switch fabric, most of the cells will be switched to the output buffer, and therefore, we need to provide only small amounts of input buffering to temporarily store the cells that lose contention to an internal link or output port. The input-output buffering approach, when armed with a suitable backpressure mechanism, can provide satisfactory performance and reduce the speed requirements of the output buffer.

C. Switch Architecture

The Balanced Gamma network is a fault-tolerant and reliable MIN that was first reported in [12] as a broadband switch architecture. The performance, fault tolerance and reliability properties of the BG network have been extensively studied and it has been found that the BG network is superior to other well-known MINs that have a comparable hardware complexity, such as the 2-dilated 2-replicated (2D2R) banyan network. As well, it performs much better than the crossbar network, which has a higher hardware complexity [11]. To support multicast traffic, we have developed a new switch architecture that can support implicit multicasting efficiently while preserving other attractive features of the BG network [13].

Unlike the banyan network, which utilizes a 2×2 switch element (SE), the multicast BG switch utilizes a 4×4 SE. Figure 1 shows the architecture of an 8×8 multicast BG switch. The basic architecture of an $N \times N$ BG multicast switch consists of N input port controllers (IPCs), an $N \times N$ multistage interconnected switch fabric that supports self-routing, self-replication and delivery acknowledgement, and N output port controllers (OPCs). No dedicated copy network to support the replication functionality is required. The IPC terminates the input signals from the network, strips the information contained in the cell header, and uses a lookup table to determine the destinations. The switch fabric is the core of the multicast BG switch. An $N \times N$ BG switch fabric consists of $n + 1$ stages, where $n = log_2 N$, with each stage consisting of N SEs numbered from 0 to $N - 1$. In stage 0, 1×2 SEs are used and in stage 1, 2×4 SEs are used. Each of the following $n - 2$ stages is comprised of 4×4 SEs.

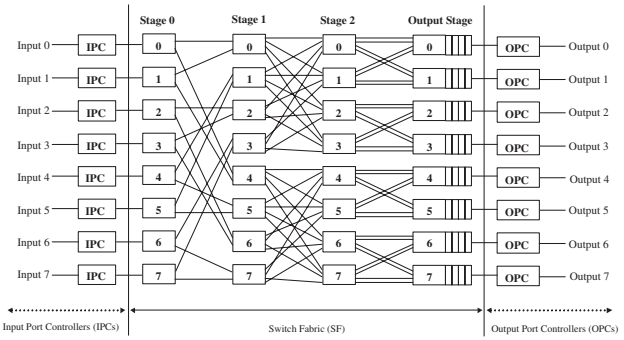

Fig. 1. The architecture of an 8×8 multicast BG switch

The last stage is the output buffer stage, which can accept up to 4 cells per output port in one switching cycle. Network bandwidth is expanded through the first two stages and then remains the same for all subsequent stages. Through internal bandwidth expansion, the multicast BG switch can achieve better performance while keeping the hardware complexity reasonable. The OPC includes regulator and scheduler; it updates each arrived cell with a new cell header and sends onto the output link. Detailed discussion on the justification for the architecture choices of the multicast BG switch can be found in [13], [14].

D. Self-Routing and Self-Replication Algorithm

A three-phase switching operation is performed inside the multicast BG switch [15], [16]. First is the reservation phase during which the tag of the HOL cell is routed through the SF. Tag information is used internally by the switch fabric for the connection setup and is generated and attached to each cell by the IPC. Multicast cell replication is performed implicitly by the SEs along with the routing operation only when necessary. Next comes the acknowledgement phase, during which cell delivery information is reported to the IPC by use of backpressure mechanism. Based on that information, the IPC decides whether the HOL cell should be transmitted or kept in the input buffer for the next cycle, or possibly both in the case of a multicast cell. In the third phase, the payload is transmitted via the established path just as in a circuit switch. Because of the memoryless design of the SF, a cell will either reach the desired output port(s) or be kept in the input buffer. Cell sequence can be easily maintained. Cell loss occurs only when the input buffer is full and a new cell arrives.

In the multicast BG switch, there are three types of SEs: the 1×2 and 2×4 SEs are used for the first two expanding stages while the 4×4 SEs are used for all subsequent stages. Functionally, the first two types of SEs can be treated as a special (simpler) case of the 4×4 SE. Therefore, in the following discussion, the more general 4×4 SE is used, which is shown in Figure 2. The four output links are numbered 0 to 3 from top to bottom. Among the four links, link 0 and link 2 are called upper and lower regular links, respectively, while link 1 and link 3 are called upper and lower alternate links, respectively. Both the regular link and its alternate link have the same capability of reaching the same destination.

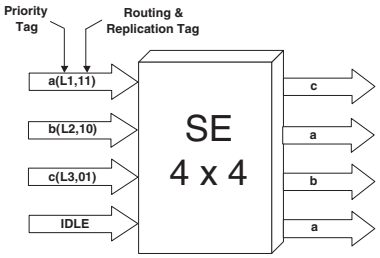

Fig. 2. Self-routing and cell replication in the 4×4 SE.

Bit 1	Bit 0	Routing Action	Replication Action
0	0	Idle (no action)	Idle (no action)
0	1	Lower link	No replication
1	0	Upper link	No replication
1	1	Both links	Replication

TABLE I
ROUTING AND REPLICATION ACTIONS BASED ON TAG PAIR INFORMATION.

Upon switching, the regular links are always used first. The alternate link is used only when the regular link has already been assigned to a connection.

In the multicast environment, tag design becomes more challenging because not only the routing information should be carried but also the cell replication information, and the tag length should be minimized to minimize the delay in the reservation phase. In the BG switch, for each SE to make the right routing and replication decision, a 2-bit tag is used by each SE for each input link. Four different actions can be taken by the SE and these are summarized in Table I.

Priority switching is a feature considered in the multicast BG switch, with up to 8 priority levels currently supported. The SE will make its decision in two steps. Firstly, the SE decides the processing order of incoming cells based on the priority level associated with each cell. Secondly, incoming cells are switched following the order determined in the first step. Cells with higher priority are always processed first until all incoming cells are processed or all the sources are used up. In the latter case, the remaining low priority cells will be blocked. An example is provided in Figure 2 in which cells are coming in from the top three input links. By sorting on the priority tag, the process order is $c \rightarrow b \rightarrow a$. Following the routing and replication table, cell c is a unicast cell which requests an upper output link, it is switched to output 0 and similarly cell b is switched to output 2. For cell a, the tag bit pair '11' indicates that replication is required. The available outputs are checked and cell a is replicated and sent to both upper and lower alternative output links, links 1 and 3, respectively.

VLSI design of the multicast BG switch fabric has been conducted using the $0.18\mu m$ CMOS technology. Detailed information and results about the VLSI design and implementation of the BG switch can be found in [13], [14]

III. Multicast Traffic Model

An important part of any performance analysis is an accurate traffic model which will be used to generate traffic for both simulation and analytical purposes. The multicast traffic model can be described by three random processes: the arrival process, the fanout process, and the destination selection process. The arrival process specifies the correlation among the successive cells. The fanout process determines the number of destinations associated with a cell when it is generated. The destination selection process describes how cell destination will be selected. In this paper, cell destinations are considered to be uniformly distributed.

A. Arrival Process

Two types of patterns are considered for the arrival process: random and bursty. For random traffic, the cell arrival is randomly selected based on the link load and is independent of cell arrival during the previous switching cycle. For bursty traffic, the ON-OFF model is used [3], [17], [18]. The ON-OFF model is the least complex and the most widely used model to simulate bursty sources. It can describe most of the existing sources with a reasonable accuracy [19]. The source generates cells in a bursty manner: one active period (ON period) followed by an idle period (OFF period). During the ON period, the traffic source continues sending cells in every switching cycle to the same destination. The durations of ON and OFF periods are independently evaluated from two geometric distributions with the period length L in cells derived from

$$L = 1 + \left\lceil \frac{\ln(1-R)}{\ln(1-p)} - 1 \right\rceil, \tag{1}$$

where $R,\ 0 \leq R < 1$, is the random number generated, and $p,\ 0 < p < 1$, is the reciprocal of the average period length in cells. The cells arriving at each input line in a burst have the same fanout number and are destined to the same output ports.

B. Fanout Distribution

The fanout process describes the fanout distribution of a multicast cell, i.e., the distribution of the number of copies of an incoming cell. A multicast cell with a fanout of one is a unicast cell. The traffic model provides a mix of unicast and multicast traffic with the level determined by the fanout distribution. The truncated geometric distribution is used to model the fanout distribution of the multicast cells [3], [18]. Given a switch size N, parameter q can be calculated numerically for any given mean fanout $\overline{F}$ following the equation

$$\overline{F} = \sum_{i=1}^{N} i \times \frac{(1-q) \times q^{i-1}}{1-q^N} = \frac{1}{1-q} - \frac{N \times q^N}{1-q^N}. \tag{2}$$

With parameter q, the probability of having a fanout value f, denoted by $P_{tg}(f)$, can be calculated by using

$$P_{tg}(f) = \begin{cases} \frac{(1-q)\times q^{f-1}}{1-q^N} & \text{for } 1 \leq f \leq N \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Under uniform destination selection, all output ports are equally likely to be requested, therefore, the input and output load of the switch can be represented by the load of each of the input and output links, denoted by ρ_{in} and ρ_{out}. Given an ideal strict-sense non-blocking switch fabric, for unicast traffic, ρ_{out} is given by

$$\rho_{out} = \rho_{in}. \tag{4}$$

However, for multicast traffic, when cell replication occurs inside the switch fabric, the load at the input port is different from that at the output port. The offered load ρ_{out} can be associated with ρ_{in} via the mean traffic fanout $\overline{F}$ by using

$$\rho_{out} = \overline{F} \times \rho_{in}. \tag{5}$$

From basic queuing theory, we know that a queue will become unstable when the data arrival rate is greater than the departure rate. For each output queue, the departure rate is assumed to be one cell per switching cycle. To avoid overflow, the offered load ρ_{out} should normally be kept below one. Even though the average load is kept below one, due to the statistical nature of the traffic, it is possible that the load momentarily exceeds one. To accommodate different fanout situations and carry out reasonable comparison in multicast traffic, the offered load to the switch is defined at the switch output. The load is converted to the input load via the mean fanout $\overline{F}$. As long as the load used does not cause the output queue to overflow, it is guaranteed that there is no overflow problem at the BG switch input. Unless otherwise stated, the traffic load reported in the paper refers to the offered output load.

IV. Performance Analysis

Due to congestion and bottleneck nodes in the network, traffic in high-speed networks tends to be bursty. Bursty traffic is a traffic type in which the switch inputs receive sudden bursts of packets destined to one output. In this section, multicast bursty traffic and simulation results are used for the following discussion. All performance measures are obtained through a simulation period of switching 10^9 cells across the switch fabric. The loss performance and delay performance are studied in this paper. Other performance measures, such as maximum/average input/output buffer requirements, can be found in [14]. Performance results are compared to those of an ideal multicast switch and other two published high performance switches. The results from an ideal switch represent the best performance that a switch fabric can achieve. However, it would be too costly to build such a switch in terms of hardware complexity, especially when the switch size becomes very large.

A. Performance Comparison with Ideal Multicast Switch

Because the ideal multicast switch can transfer all incoming cells to their requested outputs during the same switching cycle, zero cell loss will always be achieved as long as adequate output buffering is provided. Therefore, the loss performance study focus only on the BG switch.

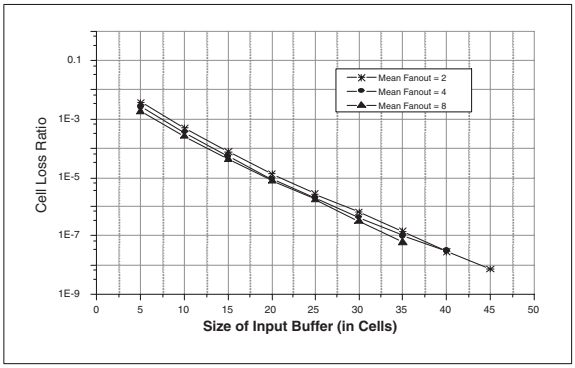

Fig. 3. Loss performance for 128×128 BG switch under different mean fanout for 90% multicast bursty traffic

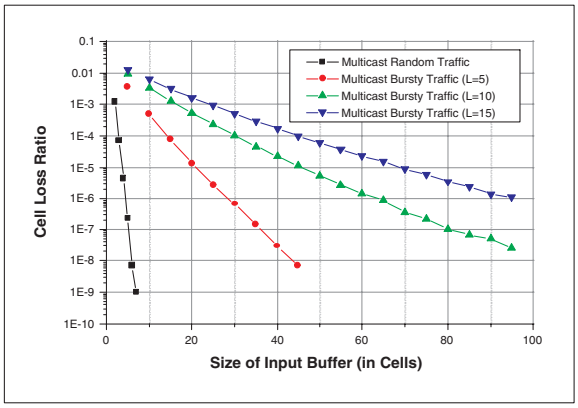

Fig. 4. Loss Performance under Multicast Bursty Traffic for 128×128 BG switch

Switch Size	Traffic Load	Delay of Multicast BG Switch			Ideal Switch
		Total	Input	Output	Total Delay
16×16	0.9	45.0546	0.2246	44.7828	44.7995
32×32	0.9	47.3710	0.3569	46.9396	47.1070
64×64	0.9	48.7309	0.5015	48.1258	48.4956
128×128	0.9	49.5026	0.6429	48.7291	48.8855
256×256	0.9	49.5963	0.7844	48.6546	49.0639

TABLE II
AVERAGE DELAY BREAKDOWN UNDER 90% MULTICAST BURST TRAFFIC WITH MEAN FANOUT 2 AND AVERAGE BURST LENGTH 5

1) Loss Performance: Because of the backpressure algorithm, a blocked cell is buffered in the input queue for further switching. Cell loss occurs only when the input queue is full and a new cell arrived. In that case, all copies implicitly contained in the new cell will be dropped. In the analysis, adequate output buffering resource is assumed so that any cell that manages to reach any output queue will be accepted. Cell loss probability is measured over the size of the input queue. Therefore, the only reason for the HOL cell to be kept in the input queue is the internal blocking of the switch fabric.

This is no doubt that better loss performance can be achieved under a lighter load. In this paper, we focus on the main characteristics of the multicast bursty traffic: fanout and burstiness. Figure 3 plots the cell loss probability for an 128×128 BG switch under 90% bursty traffic with a mean fanout of 2, 4, and 8. Although the difference is minor, we observe that under the same offered load, the larger mean fanout, the better loss performance. For example, with 30-cell input queue, the multicast BG switch achieves a cell loss ratio of 6.36×10^{-7}, 4.13×10^{-7}, and 3.06×10^{-7} for the mean fanout of 2, 4, and 8, respectively. This is because traffic load is defined at the switch output and is converted to input load via the mean fanout, as shown in Equation 5. With the same offered load, traffic with larger mean fanout will have a lower input load. Because cell replication is performed inside the SF, the load on the interconnection links increases gradually as cells approach the output port. Before reaching the output buffer stage, the average traffic load on the link with larger fanout is always less than that from traffic with a smaller fanout.

Figure 4 plots the cell loss ratio for the 128×128 BG switch under 90% multicast bursty traffic with the average burst length of 5, 10, and 15. Random traffic, which can be viewed as a special case of bursty traffic where the burstiness is constant one, is plotted in the same figure for comparison. A mean fanout of 2 is used. It is clear that as the level of traffic correlation increases, internal blocking inside the switch fabric becomes larger. As a result, the demand of resource at the input queue increases in order to keep the same level of performance. For example, only 6 cell spaces for each input queue are required to achieve a better than 10^{-8} loss performance in random traffic. This number increases to 45 and over 95 when the burst length becomes 5 and 10, respectively. As the burst length gets greater than 15, more than 100 cell space must be equipped in order to keep the same cell loss ratio.

2) Delay Performance: Because of the bufferless switch fabric design, cells are delayed either at the input queue or at the output queue. The delay associated with the overhead transfer during the reservation phase, which is a constant value and applies to every cell, is not included. At the input queue, only the master cell is stored. Multiple destination requests are contained in the cell header. When reaching the output queue, each copy becomes an independent cell. Therefore, in delay performance analysis, the input queueing delay is measured in terms of the master cell while the output queueing delay and total delay are calculated based on an individual copy. In the delay performance analysis, enough input and output buffering are assumed so as to provide zero cell loss.

Table II lists the delay performance breakdown for various switch sizes under 90% multicast bursty traffic with a mean fanout of 2 and an average burst length of 5. Total delay for the ideal switch is also provided in the table for comparison. Because the ideal switch fabric can switch any incoming cell to the requested output ports in the same switching cycle, its input queueing delay is zero. Because signal propagation delay inside the switch fabric is neglected, the output queueing delay for the ideal switch is in fact its total delay.

From the table, it is clear that the output queueing delay is the dominant part for the BG switch, and its total delay is

Burst Length	Mean Fanout	Delay of Multicast BG Switch			Ideal Switch
		Total	Input	Output	Total Delay
5	2	49.5026	0.6428	48.7291	48.8855
	4	49.3910	0.7787	48.3376	48.7632
	8	49.3592	0.9459	48.0373	48.3810
10	2	94.1450	1.2324	92.5919	93.0086
	4	94.3925	1.5435	92.1378	93.9522
	8	94.6992	1.9188	91.7577	93.6238
15	2	139.1580	1.8302	136.8090	137.3470
	4	139.3400	2.3427	135.8240	136.8180
	8	138.8480	2.9285	134.1840	137.2500

TABLE III

AVERAGE DELAY PERFORMANCE BREAKDOWN FOR 128×128 BG AND IDEAL MULTICAST SWITCH UNDER 90% MULTICAST BURST TRAFFIC

always only slightly larger than that of the ideal switch. It is not difficult to explain this observation. The high throughput of the multicast BG SF ensures that many of the cells are transferred to the output queue within one switching cycle, thus resulting in the same delay as that in the ideal switch. Only the very few cells that are left behind due to internal blocking contribute to the input queueing delay and make the overall cell delay slightly larger than in the ideal case.

It is also observed that as the switch size grows, the increase of total cell delay for the BG switch becomes slightly larger than that of the ideal switch. This is because as the switch size grows, more stages are added, the chance of cells being blocked inside the SF increases. As a result, the total cell delay increases as the switch size grows and the difference between the BG switch and the ideal switch becomes larger.

In Table III, the delay breakdown is presented for the 128×128 BG switch under 90% traffic load and various fanout and burstiness conditions. It is obvious that the change in burtiness affects the delay performance significantly while the impact of fanout is trivial. The burstiness increase means the correlation between successive cells increases because all cells belonging to the same burst go to the same destinations. Even though in the long run, destination selection is uniformly distributed, on a cycle by cycle basis, the possibility of having more cells come to the same output increases with longer bursts. Having more than two arrivals to the same output causes output queue buildup. As a result, each cell will experience longer output queueing delay. At the same time, traffic correlation will increase the chance of internal blocking because more cells are competing for the links toward the same output port, which in turn causes more blocked cells to be retained at the input queue, and thus increases the input queueing delay. Even though the internal blocking becomes worse as the traffic burstiness increases, the switching capability of the multicast BG switch ensures that most of the cells manage to reach their destinations. Therefore, although the input queueing delay increases along with the traffic burst length, it is always a small fraction of the output queueing delay. In general, average cell delay in the BG switch is very close to that of the ideal switch.

With larger mean fanout, each cell will contain more copies. It will take longer for all copies contained in the cell header to be delivered, which in turn will make cell delay in the input queue longer. But larger mean fanout also means less traffic load at the switch input which helps to reduce internal blocking, especially in the early stages. Therefore, as traffic mean fanout grows, the input queueing delay increases only slightly.

B. Performance Comparison with Other Switches

In this section, the delay and loss performance of the multicast BG switch are compared with two high-performance switches published in the literature, the Abacus switch [4], [5] and the PINIUM switch [3]. Multicast cell replication is handled implicitly by both switches. The Abacus switch is an input-output buffered switch, which is very similar to BG switch, while the PINIUM switch is a purely output-buffered switch.

1) PINIUM Switch: The basic architecture of the PINIUM switch consists of a distribution section and a concentration section. The distribution section provides the routing and multicasting functions and is made up of a stack of multicast radix-r trees. The concentration section uses the knockout principle and is made up of a row of N-to-L priority concentrating sorters.

The knockout parameter L, used in the multicast switch to describe the concentrators, decides how many cells are accepted by the output buffer in each output line in a given cycle. When L = 4, this situation is somewhat similar to the four-cell acceptance at each output line of the BG switch. Clearly, if a larger value was used, better performance would be obtained, but the hardware complexity of the output buffers would also increase correspondingly, and the buffer speed becomes a constraint. As the PINIUM switch does not have any input buffers, much larger values of L are required. It has been found that L should be above 8 so that cell loss rates of better than 1×10^{-6} can be obtained, and a figure of 16 is recommended. As the BG switch employs input buffers, the output buffers can be much simpler than those in the PINIUM switch.

Due to the above, as well as to other architectural differences between the PINIUM switch and the BG switch, comparing their performances is not straightforward [15]. For example, a 64×64 PINIUM switch with L = 16 gives a cell loss probability of better than 1×10^{-6} under 85% uniform random traffic when 37 output buffers are provided. When all other variables remain the same, providing 50 or more output buffers virtually eliminates cell loss. The 64×64 BG network under 85% uniform random traffic requires 39 output buffers as well as 5 input buffers to ensure that there is virtually no cell loss. These figures indicate that the two switches have similar performance under random traffic with similar buffer sizes. However, the BG switch is less complex. A knockout factor of 16 in the PINIUM switch would require output buffers that operate at sixteen times the speed of the link. Given that a 64×64 PINIUM switch and the 155.52 Mbps OC-3 link are used, the required minimum memory speed is 2.48 Gbps. As switch sizes become larger and link speeds grow higher, it is not practical to build the output queue for such a switch using the current memory storage technologies.

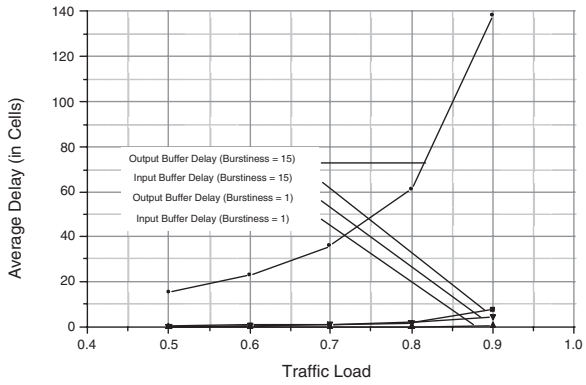

Fig. 5. Average input and output queueing delay of the multicast BG switch (to compare with Abacus switch)

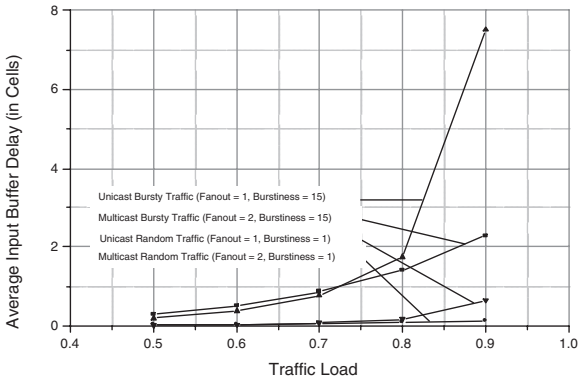

Fig. 6. Average input buffer delay vs. traffic load of the multicast BG switch (to compare with Abacus switch)

2) Abacus Switch: The Abacus [4], [5] switch is modified from the purely output-buffered MOBAS switch [9]. Input buffers are equipped to temporarily store cells that have lost contention inside the multicast grouping network (MGN). A total of K feedback lines are added. Each feedback line is actually a broadcast bus, which is used to report blocking messages to all IPCs. Multicast translation tables (MTTs) are used between the MGN and small switch modules (SSMs) to generate the routing and replication tag that will be used for switching inside SSMs for cells that managed to depart from the MGN. The results for the Abacus switch, which will be compared with the BG switch, can be found from page 204 to 205 of [5].

In the Abacus switch, cell replication is achieved by broadcasting incoming cells to all routing modules of the multicast grouping network. The group expansion ratio L and the group size M can be engineered to meet the performance requirement. In Figure 5, the average input and output queueing delay is plotted for 256×256 BG switch under both unicast uniform random traffic and bursty traffic with an average burst length of 15. It has been observed that the input queueing delay is almost negligible when compared to output queueing delay for both switches. Both switches perform much better under random traffic than bursty traffic. The Abacus switch performs only slightly better than the BG switch. For example, under 70% bursty traffic, the output queueing delay for Abacus switch is around 35 switching cycles while that for the BG switch is around 36. Under 80% bursty traffic, the output queueing delays are 58 and 62 respectively for the two switches.

In Figure 6, the input queueing delay for unicast, multicast, random, and bursty traffic conditions for the BG switch are plotted. The trends are consistent for both switches. Under the same offered load, both switches perform better under multicast traffic than unicast traffic. Again, the Abacus switch performs slightly better than the BG switch. For example, under 80% unicast and multicast bursty traffic, the average input buffer delays are around 0.6 and 0.5 switching cycles for the Abacus switch, while those numbers become 1.7 and 1.4 for the BG switch.

In Figure 7, input buffer overflow probability in terms of input buffer size is measured under 90% unicast bursty traffic with an average burst length of 1, 10, and 15, respectively. Although the amount of input buffering resource required by the BG switch is always only a small fraction of the output buffering, it is higher than that for Abacus switch. This implies that the internal blocking for the BG switch is slightly higher than that for the Abacus switch, especially when highly bursty traffic is used.

Through the above comparison and discussion, it seems that the BG switch is inferior to the Abacus switch in performance. However, it has been noticed that there is a big difference in hardware to construct the two switches. For the 256×256 Abacus switch, group size $M = 16$ and expansion ratio $L = 1.25$ is used. That means there are a total of 16 RMs, MTTs, and SMMs used to construct the switch. Each RM is a $256 : 20$ knockout switch and each SMM is a $20 : 16$ knockout switch. The hardware complexity is estimated in terms of the number of crosspoints for a switch. Then the complexity for the Abacus switch is $16 \times 256 \times 20 + 16 \times 20 \times 16 = 87,040$, which is even higher than the crossbar switch ($256 \times 256 = 65,536$). The complexity for an 256×256 BG switch is given by $1 \times 256 \times 1 \times 2 + 1 \times 256 \times 2 \times 4 + 6 \times 256 \times 4 \times 4 = 27,136$. Therefore, the complexity for the BG switch is less than one third of the complexity of the Abacus switch. This number does not include the complexity associated with the translation tables and the feedback buses, which are required by the Abacus switch but not for the BG switch. As the switch size grows larger or when the Abacus switch needs to be reconfigured to achieve higher performance, the hardware complexity will become even higher.

It appears that the higher performance of the Abacus switch is actually due to its very high hardware complexity inside the fabric. In addition, high-speed memory and complicated control are also required by the Abacus switch because the output buffer is shared by all the channels within the same group. If the same 256×256 switch is used as the example, then a group size $M = 16$ means that the output buffer has to run at least 16 times the link speed. There is no doubt that the control function has to be fast enough to coordinate and achieve full sharing among the buffers. As higher link speed and larger group size are adopted, the buffer control unit will soon reach its bottleneck. Even though such sharing

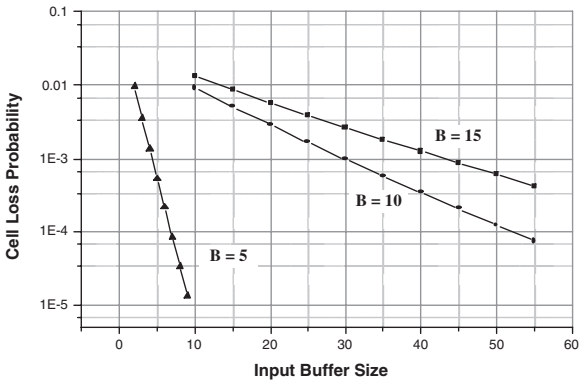

Fig. 7. Input buffer overflow probability vs. input buffer size of the multicast BG switch (to compare with Abacus switch)

can improve buffer utilization, it is very costly to build for large switches. Besides this, there is no feedback mechanism to report blocking that occurs inside the SSM or due to output buffer overflow. As a result, cell loss will happen when such a blocking situation occurs. Therefore, SSM and output buffer have to be configured large enough to accommodate any kind of incoming traffic, which will result in increased cost.

Compared with the sophisticated buffer used in the Abacus switch, all building blocks used in the BG switch, including the buffer design, are simple and will not become bottlenecks for the switch. As well, it has been discovered in the study of the pipeline structure of the BG switch [20] that a single-plane BG switch is enough to achieve a satisfactory performance in most traffic conditions. With two-plane-pipelining, the performance becomes very close to the ideal switch. Therefore, using the simple method of replicating the whole switch plane, the BG switch can be easily adjusted to much higher performance with less or comparable hardware complexity than the Abacus switch. Furthermore, such pipelining could improve the fault tolerance, robustness and reliability of the switch. It is clear that the BG switch has superior performance in comparison to the Abacus switch in relation to the level of hardware complexity.

V. Conclusion

In this paper, we have presented the architecture and performance evaluation of a new scalable multicast switch called the multicast BG switch. The switch adopts a multipath MIN design. A distributed control and a modular architecture are used in the design to fulfill the high-speed requirement. No dedicated copy network is needed to support multicast switching. Performance under various multicast traffic was compared with that of an ideal pure output-buffered multicast switch and other published high performance multicast switches. It was shown that the BG switch maintains high performance for unicast and multicast traffic under both random and bursty traffic conditions. It was also shown that the BG switch achieves a performance close to an ideal pure output-buffered architecture while keeping hardware complexity reasonable.

References

[1] T. T. Lee, "Nonblocking Copy Networks for Multicast Packet Switching," *IEEE J. Select. Areas Commun*, vol. 6, pp. 1445–1467, December 1988.

[2] J. Turner, "A Practical Version of Lee's Multicast Architecture," *IEEE Transaction on Communications*, vol. 41, pp. 1166–1169, August 1993.

[3] K. L. E. Law and A. Leon-Garcia, "A Large Scalable ATM Multicast Switch," *IEEE Journal on Selected Area in Communications*, vol. 15, pp. 844–854, July 1997.

[4] H. J. Chao, B.-S. Choe, J.-S. Park, and N. Uzun, "Design and Implementation of Abacus Switch: A Scalable Multicast ATM Switch," *IEEE Journal on Selected Area in Communications*, vol. 15, pp. 830–843, June 1997.

[5] H. J. Chao, C. Lam, and E. Oki, *Broadband Packet Switching Technologies - A Practical Guide to ATM Switches and IP Routers*. New York: John Wiley & Sons, 2001.

[6] M. A. Marsan, A. Bianco, P. Giaccone, E. Leonardi and F. Neri, "Multicast traffic in input-queued switches: optimal scheduling and maximum throughput," *IEEE/ACM Transactions on Networking*, vol. 11, Issue 3, pp. 465–477, June 2003.

[7] M. A. Marsan, A. Bianco, P. Giaccone, E. Leonardi and F. Neri, "Packet Scheduling in Input-Queued Cell-Based Switches," in *Proceedings of IEEE INFOCOM'2001*, vol. 2, pp. 1085-1094, Alaska, USA, April, 2001.

[8] W. D. Zhong and Y. Onozato, "A Copy Network With Shared Buffers For Large Scale Multicast ATM Switching," *IEEE Transactions on Networking*, vol. 1, pp. 157–165, April 1993.

[9] H. J. Chao and B. S. Choe, "Design and Analysis of a Large Scale Multicast Output Buffered ATM Switch," *IEEE/ACM Trans. Networking*, vol. 3, pp. 126–138, April 1995.

[10] Y. El-Sayed and R. Venkatesan, "Modeling and Simulation of the Pipelined Balanced Gamma Network," in *Proceeding of the Fourth IEEE International Conference on Electronics, Circuits, & Systems (ICECS'97)*, vol. 1, pp. 97–101, 1997.

[11] R. Venkatesan, Y. El-Sayed, R. Thuppal, and H. Sivakumar, "Performance Analysis of Pipelined Multistage Interconnection Networks," *Informatica – International Journal of Computing and Informatics*, vol. 23, Spetember 1999.

[12] R. Venkatesan and H. Mouftah, "Balanced Gamma Network - A New Candidate for Broadband Packet Switching Architectures," in *Proceedings of the IEEE INFOCOM'92*, vol. 3, pp. 2482–2488, 1992.

[13] C. Li, H. M. Heys and R. Venkatesan, "Design and Scalability of the Multicast Balanced Gamma (BG) Switch," *Proceedings of the Eleventh International Conference On Computer Communications and Networks (IEEE ICCCN'2002)*, p.p. 518-521, Miami, Florida, USA, October 2002.

[14] C. Li, *Design, modelling, and Analysis of the Balanced Gamma Multicast Switch for Broadband Communications*. PhD Dissertation, Memorial University of Newfoundland, October, 2004.

[15] C. Li, R. Venkatesan, and H. M. Heys, "Performance Evaluation of the Multicast Balanced Gamma (BG) Switch," in *the 2002 International Symposium on Performance Evaluation of Computer and Telecommunication System (SPECTS 2002)*, pp. 118–125, July 2002.

[16] A. Pattavina, *Switching Theory: Architecture and Performance in Broadband ATM Networks*. New York: Wiley, 1998.

[17] I. Elhanany and V. Tabatabaee, "Benchmarking Next-Generation Switch Fabrics," *IEEE Computer Magazine*, pp. 109–110, October 2003.

[18] S. H. Dyun and D. K. Sung, "A General Expansion Architecture for Large-Scale Multicast ATM Switches," *IEICE Transactions on Communications*, vol. E80-B, pp. 1671–1679, November 1997.

[19] G. D. Stamoulis, M. Anagnostou, and A. Georgantas, "Traffic Source Models for ATM Networks: A Survey," *Computer Communications*, vol. 17, no. 6, pp. 428–438, 1994.

[20] Yaser El-Sayed, *Performance Analysis, Design and Reliability of the Balanced Gamma Network*. PhD thesis, Memorial University of Newfoundland, 2000.