PERFORMANCE ANALYSIS OF MULTICAST REPLICATION MECHANISM IN SHARED-MEMORY SWITCH WITH SPEEDUP

Wang Weizhang  Ge Ning  Feng Chongxi
(Dept of Electronic Engineering, Tsinghua University, Beijing 100084)

Abstract   A multicast replication algorithm is proposed for shared memory switches. It uses a dedicated FIFO to multicast by replicating cells at receiver and the FIFO is operating with shared memory in parallel. Speedup is used to promote loss and delay performance. A new queueing analytical model is developed based on a sub-timeslot approach. The system performance in terms of cell loss and delay is analyzed and verified by simulation.

Key words   Switch; Shared memory switch; Multicast; Cell loss

I. Introduction

Broadband data networks require fast switches to move the packet from input ports to the output ports with the appropriate route. Shared memory switches provide optimal performance in terms of throughput and delay under different unicast traffic patterns, and also known to provide the best buffer utilization and can easily handle service classes. An N × N shared-memory switch consists of an N : 1 multiplier, Serial to Parallel (S/P) converter, Parallel to Serial (P/S) converter, Shared Memory (SM), switch controller, and a 1 : N de-multiplier. The function of switch is performing N write and N read operations in a round (frame), writing the N cell from N input ports to the tail of linked-queues in SM organized by outputs, and reading cells from the head of queues to the output port.

Depending on the switch architecture, multicast traffic may degrade the performance dramatically. Multicast traffic constitutes a high portion of traffic and requires special facilities to process. Most of current schemes route the unicast effectively, but do multicast poorly.

There are two approaches to implement multicasting capabilities in SM switches: (1) Replication-At-Receiving (RAR), a multicast cell is replicated in front of the SM, the multiple copies of the cell are stored in the SM according to its destined port; (2) Replication-At-Sending (RAS), a multicast cell is directly written to the dedicated multicast queue in the SM and it is linked to all its corresponding destined queues. Cell replication takes place only at dequeue time when it is sent to the outputs.

The RAR scheme is relatively simple because each copy of the cell can be treated in the same way as unicast cell after being replicated. Both the control and queue structure are the same as the unicast switch. The queue structure of RAS is more complex than RAR because a cell must be linked to several queues simultaneously according to its fanout.

It is a common knowledge that the bandwidth of SM must be adequate to sustain the aggregate flow of cells in and out of the memory. With unicast traffic in an N × N switch, the input memory bandwidth requirement is N × R, where R is the input or output line rate. In RAR, there may be N^2 replicating requests per cell time. The worst case occurs
when multicast cells arriving at all inputs are destined to all outputs simultaneously. In this case the required input bandwidth of the SM is $N^2R$. Clearly, the bandwidth may become a limiting factor.

To alleviate the bandwidth requirements, SM+RFIFO\textsuperscript{[1,2]} configuration has been used. In the configuration, the bandwidth of the replicating is not sufficient to sustain the worst case, and a Replication FIFO (RFIFO) is used before the SM to temporarily store cells while they are replicated. Arriving cells are first stored in the RFIFO. In every sub-timeslot, cells are served from the RFIFO. Once a cell has been replicated for the desired number of times and stored in the SM, it is removed from the RFIFO. In this case the required input bandwidth of the SM is $NR$. For better performance, the serving rate of the RFIFO is increased by assigning more sub-timeslots. So the input bandwidth of SM is $(N + S)R$, while $S$ is the speedup ratio for replication.

To promote the performance of replication mechanism in RAR further, we propose another RAR scheme, named SM||RFIFO, different from the sequential operation of SM+RFIFO. In SM||RFIFO, the SM and RFIFO are working in parallel. Unicast cells can be directly transported to SM for decreasing the load of RFIFO. Speedup also is used to promote the replicating performance in this algorithm.

The performance of unicast SM switch has been analyzed in Refs.\textsuperscript{[3-7]}, and in our switch architecture, the main memory is operating just as a standard unicast switch, so we will skip the analysis of the main memory. This paper is organized as follows. In Section II, we describe our replication mechanism for SM switch. In Section III, we develop the queueing model of the SM||RFIFO and analyze the cell loss and cell queueing delay of RFIFO. In Section IV, we use simulation to validate our analysis and comparing its performance with SM+RFIFO. Finally, Section V concludes the paper.

II. SM||RFIFO’s Architecture

We consider an $N \times N$ switch system, as shown in Fig.1. The switch architecture consists of a RFIFO and an SM. The RFIFO is used to temporarily store multicast cells before finishing replicating. The main memory contains $N$ queues according to each output and stores cells. The switch performs $(N + S)$ writing and $N$ reading operation in a round. The writing controller assigns $N$ writing sub-timeslots to $N$ input port, which can be used by RFIFO when the accessed input port is idle; and assigns $S$ writing sub-timeslots to RFIFO. The assignment of sub-timeslots is shown in Fig.2. For an $N \times N$ switch with input/output

---

Fig.1 Switch system architecture