Available online at www.sciencedirect.com
SCIENCE DIRECT®

Computer Communications 28 (2005) 1721–1730

computer communications

www.elsevier.com/locate/comcom

Packet scheduling in input-queued switches with a speedup of less than two and scheduling algorithms for switches with a configuration overhead

Claus Bauer

Dolby Laboratories, 100 Potrero Avenue, San Francisco, 94103 CA, USA

Received 3 December 2004; accepted 3 December 2004
Available online 13 January 2005

Abstract

Input-queued switches and the solution of input/output contention by scheduling algorithms have been widely investigated. Most research has focused on switches for fixed-size packets. This contrasts with the variable size packets of IP networks. Previous research on fixed packet length has only focused on maximum weight matching algorithms. As maximum weight matching algorithms are computationally very complex, it is of interest to investigate less complex, but stable algorithms that take into account the variable packet size of data packets. In this paper, we investigate how the class of maximal matching algorithms deployed in switches with a speedup of less than two can be modified to take into account the varying packet sizes. Using a novel model for the dynamics of maximal matching algorithms, we show that modified maximal matching algorithms guarantee stability of the switch and establish bounds on the average delay experienced by a packet.

In the second part of the paper, we apply the techniques developed for networks with variable packet size to the design of a scheduling algorithm for switches with large configuration overhead. We design a maximal weight matching algorithm for a switch with configuration overhead and determine the minimum speedup required to guarantee the stability of the switch.

Keywords: Switch architecture; Scheduling algorithm; Maximal matching; Lyapunov function

1. Introduction and motivation

The main performance features of switches are throughput and packet delay. These features are determined by the switch architecture which includes the queue structure, the scheduling algorithm and the speedup. Because output-queued switches are becoming less practical due to the high speedup required in the switching core, input-queued (IQ) and combined input/output queued (CIOQ) switches have gained major importance. A typical CIOQ $N \times N$ switch is shown in Fig. 1. For each input i, there are N virtual output queues $\mathrm{VOQ}_{i,j}$, $1 \leq j \leq N$. The packets arriving at input i and destined for output j are buffered in $\mathrm{VOQ}_{i,j}$. The switching core itself is modeled as a crossbar requiring that not more than one packet can be sent simultaneously from the same input or to the same output.

Two main classes of scheduling algorithms have been investigated in the literature. Maximum weight scheduling (MWM) algorithms have been proven to guarantee throughput and to provide bounds on average delay [9,13]. They do not require a speedup of the switching core, but are impractical due to their computational complexity of $O(N^3)$. A switch is said to work at a speedup of S, where S is a rational number $S \geq 1$, if it works at a speed S times faster than the speed of the input links. The less complex—and with regard to a chosen metric less optimal—maximal matching algorithms have been widely researched in [10,15]. A maximal matching algorithm is defined for a set of weights $Q_{i,j}$, $1 \leq i,j \leq N$, where $Q_{i,j}$ is the weight assigned to $\mathrm{VOQ}_{i,j}$, as follows:

1. Initially, all $\mathrm{VOQ}_{i,j}$ are considered potential choices for a cell transfer.
2. The VOQ with the largest weight, say $\mathrm{VOQ}_{a,b}$, is chosen for a cell transfer and ties are broken randomly.

E-mail address: cb@dolby.com.

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2004.12.016

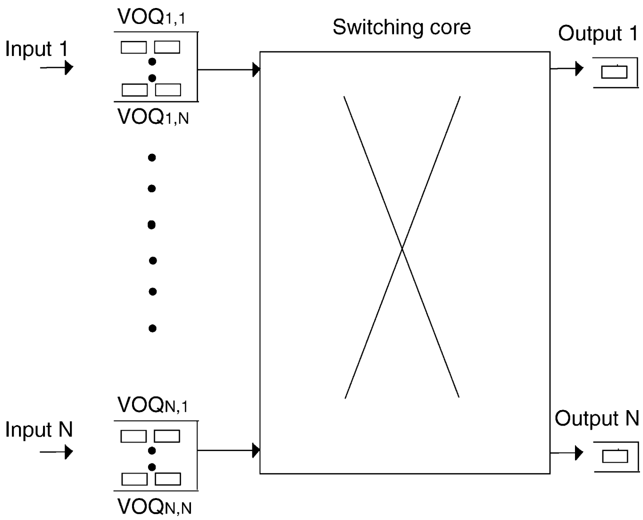

Fig. 1. Architecture of an input queued switch.

3. All $VOQ_{i,j}$ with either $i=a$ or $j=b$ are removed.
4. If all $VOQ_{i,j}$ are removed, the algorithm terminates. Else go to step 2.

The algorithm chooses one input-output connection in each iteration. Thus, due to the crossbar constraint, the algorithm terminates after N iterations. It has been shown that maximal matching algorithms provide stability for a speedup of $S=2$ [3] or even less [2].

Most existing research has focused on cells of fixed size. This contrasts with the fact that the size of packets in IP networks varies. In order to forward packets with varying size, a cell-based scheduling algorithm requires an input module that segments packets into fixed size cells. Cells are then forwarded independently. This leads to bandwidth inefficiencies as described in [16]. Moreover, cells belonging to the same packet do not necessarily arrive contiguously at the output. Thus, at the output, cells belonging to various packets must be stored and re-assembled to a packet when all cells belonging to a packet have been received. In order to overcome these disadvantages of cell based switching, cell-based switch architectures that take the actual packet size into account have been investigated in [11] and [16]. In both papers, cell-based scheduling MWM algorithms are modified for packet scheduling. In [16], two procedures to derive a packet based scheduling algorithm from a cell based scheduling algorithm ALGO are described. With both approaches, when an input-output connection is selected for the transfer of the first cell of a packet by the ALGO algorithm, then the connection is maintained at least until all cells of the packet are forwarded. With the first approach, the matching is maintained until no port is transmitting an unfinished packet. Then, a new matching is calculated. We call this algorithm PB_1-ALGO. With the second approach, at each timeslot, all input and output ports are divided in busy ports, which are busy sending unfinished packets, and free ports, which either have no packets to send or just finished sending a packet. At each timeslot the considered algorithm is only applied to the free ports. This algorithm is denoted as PB_2-ALGO. It is shown in [11] that PB_1-MWM and PB_2-MWM provide stability.

It is desirable to understand if the less complex maximal matching algorithms MM can be modified into packet scheduling algorithms PB_1-MM and PB_2-MM such that they provide stability. This paper investigates this question for switches with a speedup of $S>R$, where R is defined as follows. Let $\lambda_{i,j}$ define the arrival rate expressed in cells for $VOQ_{i,j}$ and set $\lambda_{\min}=\min_{\substack{i,j\\ \lambda_{i,j}>0}} \lambda_{i,j}$. In the sequel, we will assume that traffic is admissible, i.e. $\sum_{j=1}^{N}\lambda_{i,j}\le 1$, $\sum_{i=1}^{N}\lambda_{i,j}\le 1$, $\forall i,j,\ 1\le i,j\le N$. We set

$$E_{i,j}=\{(a,b),1\le a,b\le N:|a=i\vee b=j.\}$$

$$R_{i,j}=\sum_{(a,b)\in E_{i,j}}\lambda_{i,j},\quad R=\max_{1\le i,j\le N} R_{i,j}. \tag{1}$$

We see that $R<2$ In this paper, it is shown that PB_1-MM and PB_2-MM algorithms with a speedup of $S>R$ are stable and an upper bound for the average delay experienced by a packet is established. In contrast to earlier work that proved the stability of MM-algorithms with a fluid technique [3], this paper is based on a discrete model. The application of the discrete model bears two advantages compared to the application of the fluid model. First, the discrete model allows to show that the queue length of individual VOQs is bounded. In contrast, the fluid model only allows to show that the queue length does at most grow at a sublinear rate-when the queue length is considered as a function of timeslots that have passed since the switch was initialized. A sublinear growth, however, still allows the queue length to grow to infinity (if the queue buffer was infinite). Second, in consequence, the discrete model allows to establish bounds on the average delay experienced by a packet at the VOQs, where as the fluid model does not allow to derive these bounds. On the other hand, the fluid methodology only requires the arriving traffic to be admissible, whereas the discrete model also requires the expected arrival rates $\lambda_{i,j}$ to be i.i.d.

We develop a new model to describe the dynamics of maximal matching algorithms is used to establish results on stability and delay for the $PB_1-MM-LQF$ algorithm, the PB_1-MM algorithm whose weights are proportional to the lengths of the VOQs. A more general characterization of the forwarding behavior of all inputs and outputs competing for specific resources allows us then to establish stability and delay bounds for all PB_1-MM and PB_2-MM algorithms.

In the second part of this paper, we apply our methods developed for switches that take into account variable packet sizes to the design of scheduling algorithms for