Scheduling Cells in an Input-Queued Switch*

Published in Electronics Letters, December 9th, 1993

Nick McKeown   Pravin Varaiya
Jean Walrand
Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
Berkeley, CA 94720
USA

March 10, 1994

Abstract

We present two algorithms, IRRM and SLIP-IRRM, for scheduling cells in an input-queued cell switch. Both algorithms exhibit asymptotically 100% utilisation under high load; SLIP-IRRM within a single iteration.

1 Introduction

The major problem with input-queued switches is head-of-line blocking which can limit the maximum throughput asymptotically to 58% [1].

Previous algorithms have improved upon the 58% utilisation [2, 3, 4, 5, 6] using non-FIFO input queues and state-dependent scheduling. In each case, if the traffic is not uniformly distributed over outputs, or if the arrival process is not Bernoulli then the asymptotic utilisation decreases from its maximum. In particular, Parallel Iterative Matching (PIM) [6] will tend towards 100% utilisation if a sufficiently large number of iterations are used.

The proposed scheduling algorithm called Iterative Round Robin Matching (IRRM) is a simplified version of PIM [6]. The maximum achievable utilisation of IRRM is always asymptotic to 100% in just two iterations. It is also simpler to implement than PIM and will allocate bandwidth fairly under low and high load.

2 Iterative Round Robin Matching

Similarly to PIM, the following three steps are iterated for an M input, N output switch:

1. Each unmatched input sends a request to *every* output for which it has a queued cell.
2. If an unmatched output receives any requests, it chooses the one that appears next in a round-robin schedule starting from the highest priority element. The output notifies each input whether or not its request was granted. The pointer g_i to the highest priority element of the round-robin schedule is incremented (modulo M) to one location beyond the granted input.

*This work is supported by Pacific Bell, Bellcore and California State MICRO Program Grant 93-152.

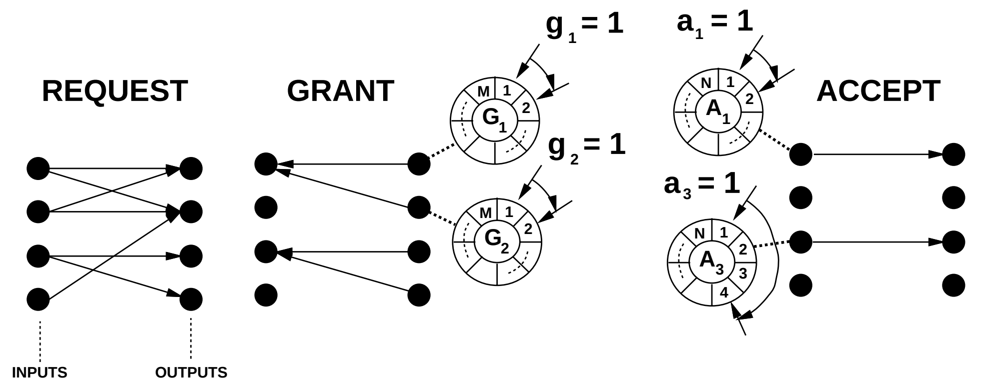

Figure 1: One iteration of the three phase Iterative Round-Robin Scheduling Algorithm.

3. If an input receives a grant, it accepts the one that appears next in a round-robin schedule starting from the highest priority element. The pointer a_i to the highest priority element of the round-robin schedule is incremented (modulo N) to one location beyond the accepted output.

An example of one iteration of the three phases is illustrated in figure 1. In the example, input 1 has one or more cells for outputs {1,2}, input 2 has one or more cells for outputs {1,2} and so on. The grant schedules are shown for outputs 1 and 2. The accept schedules are shown for inputs 1 and 3. At the end of the iteration, g_1, g_2 and a_1 are incremented to 2 and a_3 is incremented to 4.

2.1 How many Iterations?

Under high load when all input queues are occupied, it can take $\min(M, N)$ iterations to successfully schedule the switch if the schedulers are initially aligned synchronised. However, in successive cycles, a perfect match will occur in a single iteration provided that all input FIFO queues are still occupied. This is the reason for the asymptotic 100% utilisation under high load independent of the traffic statistics.

We select a fixed number of iterations, T, where $1 < T << \min(M, N)$. Under high load the schedulers will become misaligned over several time slots. Once they have become misaligned, matches will continue to be perfect so long as the input queues are occupied.

The basic IRRM algorithm will not perform well for a single iteration, because the basic algorithm does not allow the schedulers to become misaligned. But if we make the following small change to the basic algorithm, the maximum achievable utilisation becomes asymptotic to 100%. The SLIP-IRRM algorithm is identical to IRRM with the following condition added to phase 2 above:

2. ...Pointer g_i is incremented if and only if the grant was accepted.

2.2 Performance of Iterative Round Robin Matching

Figure 2 shows the utilisation versus delay for both IRRM and SLIP-IRRM for a 16-input 16-output switch. IRRM was simulated for 1, 2 and 4 iterations.

For SLIP-IRRM we can see that the utilisation is once again asymptotic to 100% because of the forced misalignment of the schedulers introduced by the modification.

In addition to the high utilisation of this algorithm, IRRM and SLIP-IRRM will always allocate bandwidth fairly between contending inputs. This is because of the deterministic nature of round-robin scheduling.

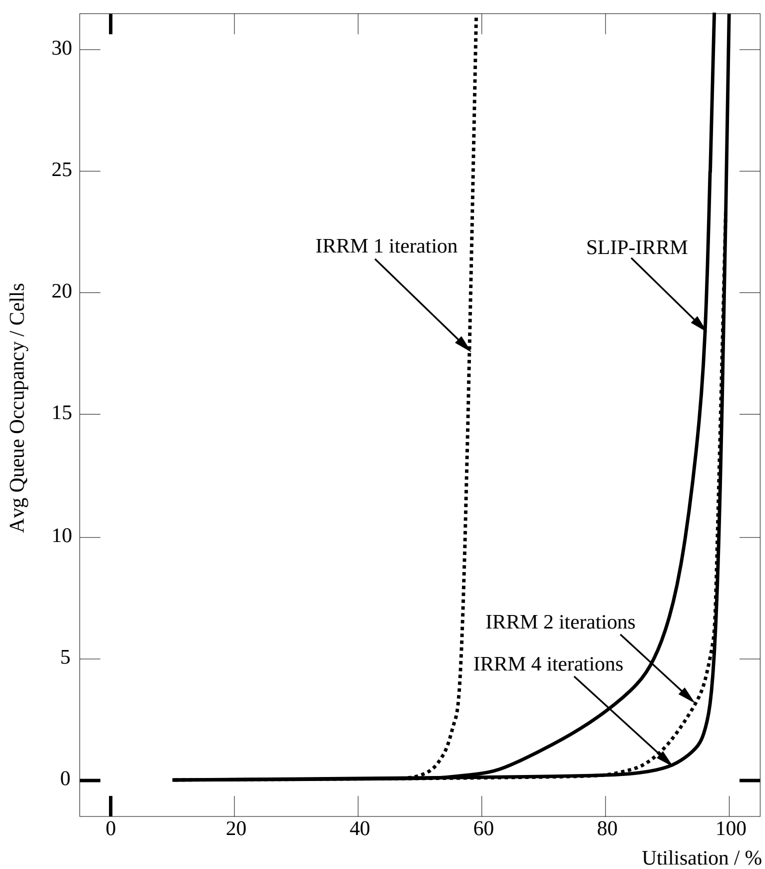

Figure 2: Average Buffer Occupancy in cells versus percentage utilisation for the Iterative Round-Robin Matching algorithm. Switch has 16 inputs and 16 outputs. Traffic model is independent Bernoulli arrivals with destinations uniformly distributed over all outputs.

3 Conclusion

We have introduced a new algorithm called IRRM for scheduling cells in an input-queued cell switch. The algorithm exhibits the interesting property of asymptotically 100% utilisation under high load when all input queues become occupied. The algorithm performs efficiently and fairly under low and high load and is simple to implement. One drawback of the algorithm is that it requires more than one iteration in any slot time. A modification to the IRRM algorithm called SLIP-IRRM provides a utilisation asymptotic to 100% in a single iteration.

4 Acknowledgements

We would like to thank Richard Edell and Tom Anderson for their helpful advice and comments.

References

[1] Karol, M., Hluchyj, M., and Morgan, S. "Input Versus Output Queueing on a Space Division Switch," *IEEE Trans. Comm*, 35(12) pp.1347-1356

[2] Obara, H. "An Efficient Contention Resolution Algorithm for Input Queueing ATM Switches," *Intl. Jour. of Digital & Analog Cabled Systems,* vol. 2, no. 4, Oct-Dec 1989, pp. 261-267.

[3] Obara, H. "Optimum Architecture For Input Queueing ATM Switches," *Elect. Letters*, 28th March 1991, pp.555-557

[4] Obara, H., Okamoto, S., and Hamazumi, Y. "Input and Output Queueing ATM Switch Architecture with Spatial and Temporal Slot Reservation Control" *Elect. Letters*, 2nd Jan 1992, pp.22-24

[5] Karol, M., Eng, K., Obara, H. "Improving the Performance of Input-Queued ATM Packet Switches," *INFOCOM '92*, pp.110-115

[6] Anderson, T., Owicki, S., Saxe, J., and Thacker, C. "High Speed Switch Scheduling for Local Area Networks," *Proc. Fifth International Conference on Architectural Support for Programming Languages and Operating Systems* Oct 1992, pp. 98–110.