Scheduling Algorithms for Input-Queued Cell Switches

by

Nicholas William McKeown

B.Eng (University of Leeds) 1986

M.S. (University of California at Berkeley) 1992

A thesis submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy
in

Engineering-Electrical Engineering
and Computer Sciences

in the
GRADUATE DIVISION
of the
UNIVERSITY of CALIFORNIA at BERKELEY

Committee in charge:

Professor Jean Walrand, Chair
Professor Pravin P. Varaiya
Professor Ronald W. Wolff

1995

This thesis of Nicholas William McKeown is approved:

Chair | Date

Date

Date

University of California at Berkeley

1995

Scheduling Algorithms for Input-Queued Cell Switches

by

Nicholas William McKeown

Abstract

Scheduling Algorithms for Input-Queued Cell Switches

by

Nicholas William McKeown

Doctor of Philosophy in Engineering-Electrical Engineering and Computer Sciences

University of California at Berkeley

Professor Jean Walrand, Chair

The algorithms described in this thesis are designed to schedule cells in a very high-speed, parallel, input-queued crossbar switch. We present several novel scheduling algorithms that we have devised, each aims to match the set of inputs of an input-queued switch to the set of outputs more efficiently, fairly and quickly than existing techniques.

In Chapter 2 we present the simplest and fastest of these algorithms: SLIP — a parallel algorithm that uses rotating priority ("round-robin") arbitration. SLIP is simple: it is readily implemented in hardware and can operate at high speed. SLIP has high performance: for uniform i.i.d. Bernoulli arrivals, SLIP is stable for any admissible load, because the arbiters tend to *desynchronize*. We present analytical results to model this behavior. However, SLIP is not always stable and is not always monotonic: adding more traffic can actually make the algorithm operate more efficiently. We present an approximate analytical model of this behavior. SLIP prevents starvation: all contending inputs are eventually served. We present simulation results, indicating SLIP's performance. We argue that SLIP can be readily implemented for a 32x32 switch on a single chip.

In Chapter 3 we present *i*-SLIP, an iterative algorithm that improves upon SLIP by converging on a maximal size match. The performance of *i*-SLIP improves with up to $\log_2 N$ iterations. We show that although it has a longer running time than SLIP, an *i*-SLIP scheduler is little more complex to implement.

In Chapter 4 we describe maximum or maximal *weight* matching algorithms based on the occupancy of queues, or waiting times of cells. These algorithms are stable over a wider range of traffic loads. We describe two algorithms, *longest queue first* (LQF) and *oldest cell first* (OCF) and consider their performance. We prove that LQF, although too complex to implement in hardware, is stable under all admissible i.i.d. offered loads. We consider two implementable, iterative algorithms *i*-LQF and *i*-OCF which converge on a maximal weight matching. Finally, we present two interesting implementations of the Gale-Shapley algorithm, designed to solve the *stable marriage problem*.

To my parents,

and

My Wife,
My Love,
My Le.

Table of Contents

Acknowledgements

For their continued guidance, support and encouragement throughout my time at Berkeley, I would like to thank my adviser Jean Walrand and Pravin Varaiya. I greatly appreciate the freedom and collegial respect you have given me and your other students.

Numerous discussions with Richard Edell lead to the design of the datapath that provided the main motivation for this thesis. Richard, you are a truly gifted engineer and it has been a pleasure to be your colleague.

I am grateful to Professor Tom Anderson for discussions about the iterative properties of the SLIP algorithm, and to Professor Venkat Anantharam for suggesting the proof in Appendix 4. I also wish to acknowledge the helpful feedback and suggestions of Chuck Thacker (DEC SRC), the inventor of parallel iterative matching. I thank Dana Randall for introducing me to the stable marriage problem, and Matthew J. Salzman (CMU) for kindly donating his code to implement the maximum match algorithms used in many of my simulations.

I am extremely grateful to John Limb, for whom I worked at Hewlett-Packard Labs in Bristol, England. John, you have been a constant source of inspiration to me; and without your encouragement I would not have gone back to school to pursue my Ph.D.

I would like to give thanks to the numerous other people at Hewlett-Packard Labs who supported me over the years, in particular John Taylor, Daniel Pitt, Steve Wright and Gwenda Ward.

Last, and definitely most, I want to thank my family. Words cannot express my thanks to my wife and parents for all your love and encouragement. I dedicate this thesis to you.