Milestone Report - Reproducing Network Research

CS244 Spring 2017

Authors:
Kate Stowell <kstowell@stanford.edu>
Travis Lanham <tlanham@stanford.edu>

Paper:
McKeown, Nick. "The iSLIP scheduling algorithm for input-queued switches."
IEEE/ACM transactions on networking 7.2 (1999): 188-201.

I. Introduction

Packet switches are the backbone of modern computer networks and make possible the local area networks used for the distributed applications that make up the modern Internet. The packet switching abstraction is integral to the architecture of the Internet and performant systems require high throughput switches with bounded latency that are simple to implement for cost efficiency.

Designing high speed switches requires an efficient scheduling algorithm to act as an arbiter between input requests and output availability. The throughput of input-queued switches is bounded by the efficiency of the scheduling algorithm, making it an important and rich area for research. The iSLIP algorithm that the paper introduces addresses issues with previous algorithms to achieve close to 100% throughput.

II. Goals: What problem was the original paper trying to solve?

The original paper sought to expand the family of algorithms that address that head of line blocking problem that input-queued switches face. It builds off previous work that demonstrated that almost 100% throughput was possible on input-queued switches.

The essence of the switch scheduling problem is that cells (packets) enter the switch on a given input and must exit on a certain output. For example, a typical switch might be 16x16 which means it has 16 inputs that it can receive cells on and 16 outputs that it can send cells out. The switch scheduler must decide when an input can use a certain output. A given input can forward at most one cell to an output at one time; there is contention for an output if more than one input is requesting to forward to that output.

The paper aims to resolve this bipartite matching problem in a way that can be implemented easily in hardware with bounded latency to maintain line rate. The paper sought to improve existing switching algorithms to reduce latency for high loads.

III. Motivation: Why is the Problem Important/Interesting?

The switch scheduling problem is important to the scalability of ethernet switches that form the backbone of local area networks that connect data centers, enterprises, universities, and more. These networks share the characteristic (particularly in the 1990s when the paper was written) of having high intra-network transfer requirements (ex. servers within a datacenter send information to each other for distributed applications, researchers at a university share results with each other).

In order to meet high network throughput requirements, switches must have low latency for packet forwarding. Without performant switches, network latency would be much higher, requiring additional hardware that costs more to realize network designs for the large-scale distributed systems that make the Internet useful. Additionally, multi-media applications that are sensitive to network latency would be prohibitively costly or impossible.

These difficulties are not theoretical, previous switch designs that used FIFO scheduling policies were limited to 58% throughput. The goal of the paper and its predecessors is to find better algorithms to solve the scheduling problem.

Although there has not been much research on switch scheduling in the past decade, it remains a critical part of networking architecture that should be understood and supported with a verified theoretical foundation.

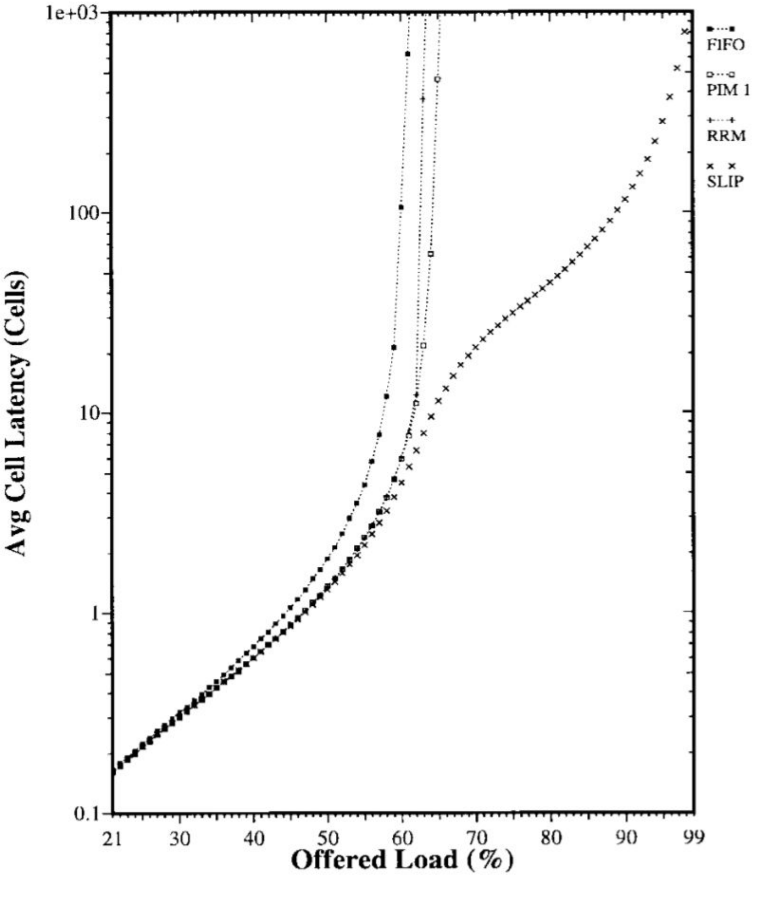
IV. Results: What Did the Original Author Find?

The iSLIP algorithm builds on work done with parallel iterative matching (PIM) for switch scheduling, extending the original PIM algorithm with a round-robin schedule. The paper presented both iterative and noniterative versions of the SLIP scheduling algorithm. The paper shows that SLIP can achieve 100% throughput for uniform traffic. The round-robin matching that SLIP uses prevents input queues from being starved and provides fair access to output queues. In the nonuniform case, SLIP performs similarly to round-robin policy among the busy queues.

The paper found that SLIP outperforms the PIM algorithm, allowing for higher load (throughput) while maintaining bounded latency. The author found that the desynchronization of output arbiters allows SLIP to outperform other algorithms.

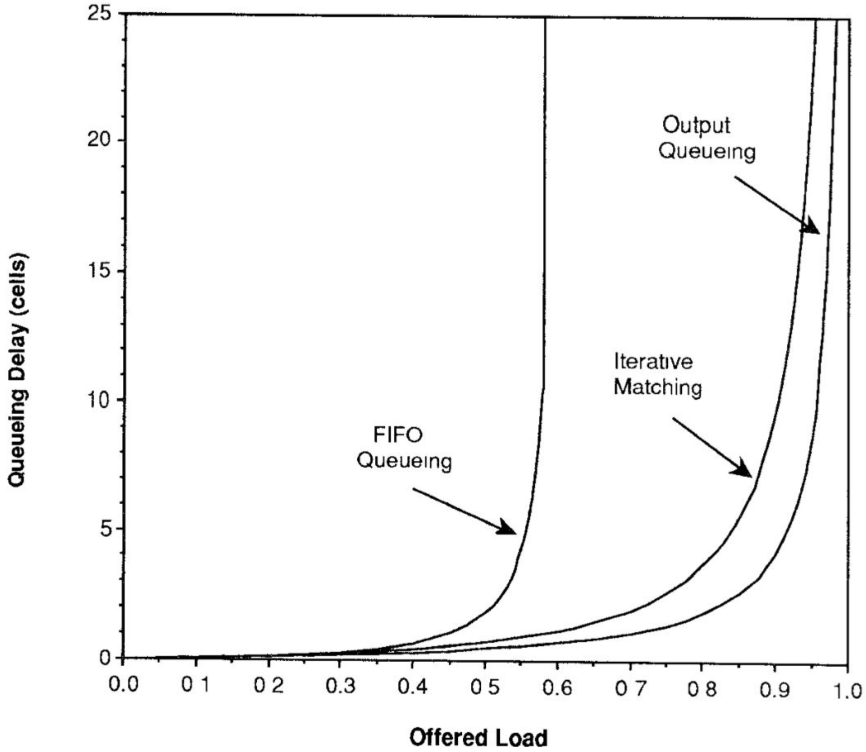
V. Subset Goal: What Subset of Results Did You Choose to Reproduce?

We choose to reproduce Figure 5 in the paper (shown below), which compares the FIFO, PIM, RRM, and iSLIP scheduling algorithms. The comparison in the paper was done with Bernoulli arrivals and with destinations uniformly distributed over all outputs on a 16x16 switch. We will measure the average delay per cell and compare this to the cell delay described in the paper. We intend to use a switch scheduling simulator to verify the performance of the algorithms. We will test the SLIP algorithm implementation that the author used and write our own implementation as well to verify the results.



VI. Subset Motivation: Why that particular choice?

We chose Figure 5 since it is the key result of the paper, depicting the relative performance of iSLIP against the other algorithms that it seeks to improve upon. Further, we noticed some discrepancy between the performance of PIM in the PIM paper and the iSLIP paper and wanted to investigate further. The PIM paper also uses a 16x16 switch to evaluate performance and has destinations uniformly distributed over all outputs but shows PIM being capable of higher load before unbounded latency, while the iSLIP paper shows a smaller load for PIM.



VII. Progress so Far

So far we have obtained the switch simulator used by the author of the paper (as well as other authors for scheduling research), updated it and fixed bugs, and done preliminary runs for validation. The simulator, SIM, simulates switching behavior in software for ATM switches.

We have done a simulation run for an 8x8 switch with the iSLIP algorithm (Appendix B) and a 16x16 switch with iSLIP (Appendix C). Additionally, we did a run for a 16x16 switch with FIFO and compared it to iSLIP (Appendix A) to illustrate the head of line blocking issue that FIFO schedulers face. We also did a preliminary run with PIM but need to do further validation since our results diverged significantly from the author's. The full results can be seen in Appendix D which shows the input latency for loads (throughput) starting at 19% and going to 99% in 2% intervals for FIFO as well as PIM and iSLIP (each with 4 iterations).

The initial measurements from the simulator are similar to those reported in *The iSLIP Scheduling Algorithm for Input-Queued Switches*; however, the standard deviation is very high and there are some minor differences in the data. We intend to explore these inconsistencies.

SIM website:
https://web.archive.org/web/20100719163607/http://klamath.stanford.edu/tools/SIM/

VIII. Plan For the Next Weeks

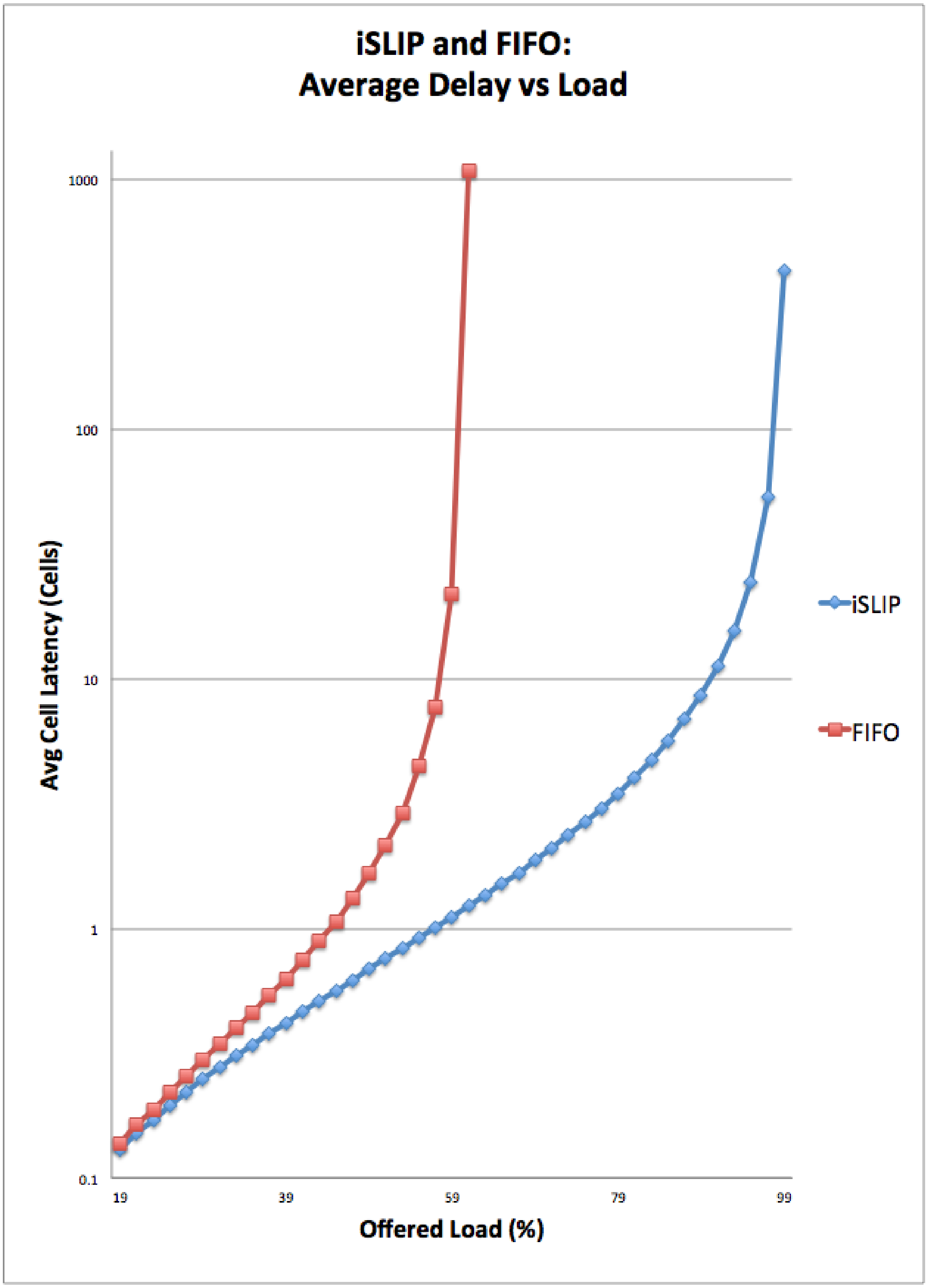
We will use the simulator to evaluate iSLIP, PIM, RRM, and FIFO algorithms under the conditions from figure 5 (16x16 switch using bernoulli arrivals and with traffic uniformly distributed across all outputs). The results for the PIM algorithm will be compared to the results from *High-Speed Switch Scheduling for Local-Area Networks* by Anderson et. al. We will also explore other figures presented in the paper and validate their performance with the simulator and evaluate the four algorithms with non-Bernoulli arrivals.

Additionally, we will write our own implementation of the iSLIP algorithm for the simulator to verify the author's results. For reproducibility, we will create a script that can be used to run simulations of the four algorithms compared in the paper and produce a graph of the results.

Lastly, we will explore extending the simulator and validating with other papers to offer further evidence of the simulator's integrity.

Appendices:

Appendix A: iSLIP vs FIFO Scheduling Performance



Appendix B: Data from 8x8 iSLIP with Bernoulli Arrivals and Uniform Destinations

```
iSLIP sim results:
=============

Traffic Information
  -------------------
  I/P  model  Util
  0  bernoulli_iid_uniform  IN: 0.940000 OUT: 0.940000
  (0,0)    0      0.150000
  1  bernoulli_iid_uniform  IN: 0.920000 OUT: 0.920000
  2  bernoulli_iid_uniform  IN: 0.950000 OUT: 0.950000
  3  bernoulli_iid_uniform  IN: 0.920000 OUT: 0.920000
  4  bernoulli_iid_uniform  IN: 0.930000 OUT: 0.930000
  5  bernoulli_iid_uniform  IN: 0.910000 OUT: 0.910000
  6  bernoulli_iid_uniform  IN: 0.930000 OUT: 0.930000
  7  bernoulli_iid_uniform  IN: 1.000000 OUT: 1.000000


  INPUT BUFFER LATENCY STATS
  --------------------------
  I/P O/P   Pri  Avg   SD
  ----------------------
    0   1   0   4.75000 1.92029 (4)
    1   1   0   4.75000 3.03109 (4)
    2   1   0   2.40000 1.35647 (5)
    3   1   0   4.42857 2.82120 (7)
    4   1   0   3.80000 3.31059 (5)
    5   1   0   0.00000 0.00000 (5)
    6   1   0   2.25000 2.77263 (4)
    7   1   0   3.16667 2.19216 (6)

  INPUT BUFFER TIME AVG OCCUPANCY STATS
  -------------------------------------
  I/P O/P   Pri  Avg   SD
  ----------------------
    1   0   0   0.22449 0.46358 (49)
```

```
   1   1   0   0.48980 0.57579 (49)
   1   2   0   0.04082 0.19786 (49)
   1   3   0   1.18367 0.74706 (49)
   1   4   0   0.30612 0.46088 (49)
   1   5   0   2.30612 0.70578 (49)
   1   6   0   0.16327 0.36961 (49)
   1   7   0   1.53061 1.08988 (49)
   1   X 6.260 1.128 (50)

  Latency statistics
  ------------------
                     Avg       SD     Number
                   -------------------------
    Input Latency:  7.7931034 7.400121    (348)
   Fabric Latency:         0        0    (348)
   Output Latency:         0        0    (348)
   Switch Latency:  7.7931034 7.400121    (348)

Total Latency over all cells:  7.7931034 7.400121    (348)
```

Appendix C: Data from 16x16 iSLIP with Bernoulli Arrivals and Uniform Destinations

```
Traffic Information
-------------------
I/P  model  Util
0  bernoulli_iid_uniform  IN: 0.931000 OUT: 0.931000
(0,0)    0      0.069000
1  bernoulli_iid_uniform  IN: 0.919000 OUT: 0.919000
2  bernoulli_iid_uniform  IN: 0.923000 OUT: 0.923000
3  bernoulli_iid_uniform  IN: 0.924000 OUT: 0.924000
4  bernoulli_iid_uniform  IN: 0.927000 OUT: 0.927000
5  bernoulli_iid_uniform  IN: 0.910000 OUT: 0.910000
6  bernoulli_iid_uniform  IN: 0.909000 OUT: 0.909000
7  bernoulli_iid_uniform  IN: 0.922000 OUT: 0.922000
8  bernoulli_iid_uniform  IN: 0.915000 OUT: 0.915000
9  bernoulli_iid_uniform  IN: 0.926000 OUT: 0.926000
10  bernoulli_iid_uniform  IN: 0.922000 OUT: 0.922000
11  bernoulli_iid_uniform  IN: 0.925000 OUT: 0.925000
12  bernoulli_iid_uniform  IN: 0.918000 OUT: 0.918000
13  bernoulli_iid_uniform  IN: 0.914000 OUT: 0.914000
14  bernoulli_iid_uniform  IN: 0.919000 OUT: 0.919000
15  bernoulli_iid_uniform  IN: 0.990000 OUT: 0.990000

INPUT BUFFER LATENCY STATS
--------------------------
I/P O/P   Pri  Avg   SD
-----------------------
  0   1   0   33.40625 19.65302 (32)
  1   1   0   18.07692 12.80301 (26)
  2   1   0   98.54286 52.61686 (35)
  3   1   0   27.88889 17.03011 (27)
  4   1   0   25.83333 19.34612 (30)
  5   1   0   12.40000 10.14298 (25)
  6   1   0   20.46875 13.72949 (32)
  7   1   0   20.64516 9.16356 (31)
  8   1   0   38.31250 19.55671 (32)
  9   1   0   20.66667 10.89342 (27)
```

```
 10   1   0   115.94118 38.76546 (34)
 11   1   0   8.10000 6.40234 (20)
 12   1   0   19.92857 11.35759 (28)
 13   1   0   42.27273 29.10719 (33)
 14   1   0   62.55882 39.10333 (34)
 15   1   0   56.83871 40.25090 (31)

INPUT BUFFER TIME AVG OCCUPANCY STATS
-------------------------------------
I/P O/P   Pri  Avg   SD
-----------------------
  1   0   0   1.50902 1.48250 (499)
  1   1   0   0.95792 0.92192 (499)
  1   2   0   1.00802 1.15178 (499)
  1   3   0   2.89980 1.37905 (499)
  1   4   0   0.60922 0.80678 (499)
  1   5   0   0.75752 0.96602 (499)
  1   6   0   1.37074 1.31159 (499)
  1   7   0   0.78156 0.93388 (499)
  1   8   0   0.83768 0.79811 (499)
  1   9   0   0.74148 1.01655 (499)
  1  10   0   0.54108 0.69914 (499)
  1  11   0   2.01202 1.67136 (499)
  1  12   0   1.04409 1.15790 (499)
  1  13   0   2.02004 1.28954 (499)
  1  14   0   2.94188 1.80476 (499)
  1  15   0   1.03006 0.91803 (499)
  1   X 21.062 1.598 (500)

Latency statistics
------------------
                  Avg       SD     Number
                -------------------------
   Input Latency:  30.472802 30.773891    (7335)
  Fabric Latency:          0         0    (7335)
  Output Latency:          0         0    (7335)
  Switch Latency:  30.472802 30.773891    (7335)
```

```
Total Latency over all cells:  30.472802 30.773891    (7335)
```

Appendix D: Data from 16x16 Switch with Bernoulli Arrivals and Uniform Destinations

Offered Load (%)	iSLIP Average Delay	PIM Average Delay	FIFO Average Delay
19	0.13002966	0.12935235	0.13747345
21	0.1505621	0.15024124	0.16306983
23	0.16976339	0.16878633	0.18724183
25	0.19600828	0.19423429	0.22109398
27	0.22183144	0.22044314	0.25683386
29	0.24860593	0.24772358	0.29933248
31	0.27719037	0.27513968	0.34501927
33	0.30821537	0.3060611	0.39854691
35	0.34171044	0.34085872	0.4566108
37	0.38153055	0.378062	0.5409031
39	0.41865222	0.41651893	0.62410354
41	0.46546341	0.45830628	0.74803582
43	0.51020818	0.50669544	0.88588793
45	0.56239461	0.55709555	1.060785
47	0.61789009	0.61160461	1.3145359
49	0.6892558	0.68374956	1.6639336
51	0.7554669	0.7457077	2.1575111
53	0.83050363	0.82141744	2.9188094
55	0.91977256	0.90471804	4.4612984
57	1.0092236	0.99962263	7.7315072
59	1.110198	1.0948473	21.691853

61	1.2347142	1.2215403	1080.302
63	1.3608158	1.3420907	3396.9623
65	1.5113615	1.4954242	5584.0283
67	1.6738519	1.6510386	7764.4728
69	1.8693744	1.8419183	8643.0828
71	2.0961229	2.0759654	8944.4506
73	2.3517823	2.3487255	9463.2291
75	2.659788	2.6334853	9891.0201
77	3.0182403	3.0145661	4798.0218
79	3.4502314	3.43008	4788.9104
81	4.0018977	3.9877868	4628.2683
83	4.6994997	4.6723097	4403.5119
85	5.6333764	5.6177245	4389.2353
87	6.887782	6.8831723	4323.7887
89	8.627052	8.7033423	4204.6825
91	11.180338	11.264693	4062.0105
93	15.595423	15.310075	4217.4537
95	24.418425	23.884576	4017.7956
97	53.378934	44.509355	3780.6833
99	428.62	161.01979	3909.4984