Architectures and Packet Scheduling of Optical Interconnects

Yuanyuan Yang

High Performance Computing and Networking Research Lab
Stony Brook University
www.ece.sunysb.edu/~yang

Research supported by U.S. National Science Foundation
Outline

- Background
  - Interconnect architectures and packet scheduling
  - Current optical technology
- Scheduling in all-optical interconnects
  - Unbuffered interconnects
  - Interconnects with output buffers
  - Interconnects with input buffers
- Scheduling in hybrid interconnects
- Summary
Interconnects

- Consider a packet interconnect/switch which forwards packets arrived at its input to its output.
- Interconnects are used at all levels: intra-chip, inter-chip, LAN, WAN,…
Basic Interconnect Architectures

- Buffering is the major approach to resolving output contention in interconnects.
- Based on where buffering takes place, interconnects can be divided into:
  - Output-queued
  - Input-queued
  - Combined-input-output-queued
  - Buffered crossbar
Scheduling in Interconnects

- A core component of an interconnect: packet scheduler, which takes care of scheduling of packets from input to output

- Goals of a scheduling algorithm include
  - Maximize throughput
  - Minimize average packet delay
  - Minimize packet loss (in a lossy interconnect)
  - Meet additional requirements such as fairness or quality-of-service (QoS)
Outline

- Background
  - Interconnect architectures and packet scheduling
  - Current optical technology
- Scheduling in all-optical interconnects
  - Unbuffered interconnects
  - Interconnects with output buffers
  - Interconnects with input buffers
- Scheduling in hybrid interconnects
- Summary
Optical Networking

- Optical fibers dominate as transmission media
  - High bandwidth (50THz)
  - Low error rate
  - Low interference

- All-optical systems (data are carried in optics from source to destination) are desirable
  - Data transparency
  - Low power consumption
  - Electronics is approaching its cap

- However, there are many challenges …
Wavelength Division Multiplexing (WDM)

- WDM (wavelength-division-multiplexing) provides a platform to exploit the huge capacity of optical fibers.
- In a WDM network, the multiplexing of optical signals on a single fiber is achieved by carrying each signal on a separate wavelength channel.
Wavelength Conversion

- A unique dimension to resolve contentions in WDM optical networks
  - When two signals compete for one wavelength channel, one of them can be converted to a different wavelength

- Wavelength conversion models
  - Full-range: convert a wavelength to any other wavelength -- expensive and difficult to implement
  - Limited-range: convert a wavelength to several adjacent wavelengths (our focus)
Buffering in Optics

- No optical random access memory (RAM)
- Fiber delay lines (FDLs)
  - Buffering by letting the signal go through extra fibers
  - Inflexible and can only provide limited delay

Buffering time: \( t = \frac{L}{v} \)

- Slow light: slow down optical signal
  - Provide continuous delay
  - Still in its early stage
Outline

- Background
  - Interconnect architectures and packet scheduling
  - Current optical technology
- Scheduling in all-optical interconnects
  - Unbuffered interconnects
  - Interconnects with output buffers
  - Interconnects with input buffers
- Scheduling in hybrid interconnects
- Summary
Unbuffered Wavelength Convertible Interconnect

- The interconnect has no buffers but has wavelength conversion ability.
- The scheduler determines how the wavelength converters and SOA gates should be configured.
- To maximize throughput is to find a maximum matching in the request graph.
First Available Algorithm

- Finding a maximum matching in an arbitrary bipartite graph needs $O(n^{2.5})$ time.
- First Available Algorithm finds a maximum matching in linear time $O(n)$ by exploiting some properties of the interconnect architecture.

```
First Available Algorithm
for i := 1 to n do
    Find $b_j$ which is the right side vertex adjacent to $a_i$ with the smallest index and is not matched to any other vertex yet.
    if such $b_j$ exists
        match $a_i$ to $b_j$
    else
        $a_i$ is not matched
    end if
end for
```
Outline

- Background
  - Interconnect architectures and packet scheduling
  - Current optical technology
- Scheduling in all-optical interconnects
  - Unbuffered interconnects
  - Interconnects with output buffers
  - Interconnects with input buffers
- Scheduling in hybrid interconnects
- Summary
Optical Interconnects with Dedicated Output Buffer

- Packets on the same wavelength and destined for the same output fiber can be sent to different delay lines of that fiber in the same time slot.

Buffers dedicated to each output link

- wavelength converter
- splitter
- SOA gate
- FDL buffers
Optimal Packet Scheduling

- In each time slot, find a scheduling such that
  - A maximum number of packets can be transmitted to the output buffer
  - A minimum average buffering delay is introduced
Network Flow Approach for Finding Optimal Scheduling

- An optimal scheduling corresponds to a maximum flow with minimum cost in the flow graph.
  - The connection between input wavelengths and output wavelengths is determined by the wavelength conversion pattern;
  - The connection between output wavelengths and buffers is determined by the packet queue length at each output wavelength.
- Adopting an existing network flow algorithm incurs high complexity.
The New Scheduling Algorithm

- **Step 1: Augment to Full Algorithm**
  - Determine the number of packets on each input wavelength to be transmitted in current time slot, $I$, and the number of buffer cells on each output wavelength to be used, $O$, in an optimal scheduling.

- **Step 2: Scheduling construction algorithm**
  - Construct an optimal scheduling from $I$ and $O$.

- **Time complexity asymptotically matches the lower bound of the scheduling problem (linear to the network size)**
Outline

☐ Background
  ■ Interconnect architectures and packet scheduling
  ■ Current optical technology

☐ Scheduling in all-optical interconnects
  ■ Unbuffered interconnects
  ■ Interconnects with output buffers
  ■ Interconnects with input buffers

☐ Scheduling in hybrid interconnects

☐ Summary
Input-Buffered Optical Interconnects

- Input-queued interconnects rely on "virtual output queue" (VOQ) to eliminate head-of-line blocking to achieve 100% throughput, while VOQ is difficult to implement in optics due to lack of optical RAM.

- Question:
  Can 100% throughput be achieved in an input-buffered all-optical interconnect with a proper scheduling algorithm?
Controllable Fiber-Delay-Line (FDL) Buffer

A controllable FDL with multiple "exits."

- Consists of cascaded FDLs connected by 9:1 couplers.
- A signal can be sent to the next FDL through the 90% output or be sent out of the FDL through the 10% output.
- Provides more flexible delay than a regular FDL.
Most-Packet Wavelength-Fiber Pair First Algorithm (MPWFPF)

- Basic idea: to favor input wavelength channel/output fiber pairs that are more congested than others.
- Each pair of input wavelength/output fiber is assigned a weight $Z_{i,j}^w(n)$: number of packets destined to output fiber $j$ and buffered on wavelength $w$ of input fiber $i$ in time slot $n$. 
Formalizing the Scheduling Problem into a Network Flow Problem

MPWFPPF finds an integral flow with maximum weight in the flow graph in each time slot.
Theorem 1. At speedup 1, MPWFPF achieves 100% throughput for an input-buffered WDM packet interconnect which uses the controllable FDL as input buffer under any admissible traffic and wavelength conversion pattern.

Proof is based on controllable FDL buffer but the conclusion does not limited to it.

Valid for any optical VOQs if become feasible in future.
Outline

- Background
  - Interconnect architectures and packet scheduling
  - Current optical technology
- Scheduling in all-optical interconnects
  - Unbuffered interconnects
  - Interconnects with output buffers
  - Interconnects with input buffers
- Scheduling in hybrid interconnects
- Summary
Hybrid Optical Interconnects

- All-optical interconnects are promising but the absence of optical RAM is one of the major problems.

- A compromise:
  - Transmission/switching in optical domain, buffering in electronic domain.
  - Packets go through O/E/O conversions at intermediate nodes.
OpCut Switch – Basic Idea

- OpCut: optically cut-through the switch
- Equipped with recirculating electronic buffers
  - Let packet cut-through whenever possible.
  - Minimum O/E/O conversion.
  - Low average packet latency
- Use currently available optical technology for all other components
Implementation of OpCut Switch – Single Wavelength

- A flow is defined between each input/output port pair.
- A buffer may contain packets from different flows.
- No queue is maintained in a buffer.

Size: $N \times 2N + N \times N$
Packet Scheduling in OpCut Switch

- **Goals**
  - Maximize throughput
  - Maintain packet order

- **Packet cutting through**
  - A newly arrived packet may cut through the switch *iff* no packet from the same flow is being buffered.

- **Packet buffering**
  - If not cutting through, a packet is picked up by a receiver.

- **Buffered packet scheduling**
  - If there are still outputs available, schedule packets from the buffers to the outputs.
Packet Scheduling in Single-Wavelength OpCut Switch

- Find three matchings between
  - Inputs/Outputs
  - Inputs/Receivers
  - Transmitters/Outputs
Matching Inputs to Outputs

- In each time slot, an input may send a scheduling request
  - If a packet arrives and is eligible for cut-through.
- All requests can be handled by a single iteration of iSLIP.
Matching Inputs to Receivers

- Inputs are connected to receivers in a round-robin fashion.
- The main purpose of round-robin fashioned connection, instead of fixed connection, is to provide better load-balancing among receivers.
Matching Transmitters to Outputs

☐ Step 1: Announce
  ■ Each output announces the status of buffers.

☐ Step 2: Request
  ■ Each receiver checks for packets with matched time stamp, and sends a request for corresponding output if it finds any.

☐ Step 3: Grant
  ■ Each output grants one of the requests, if it receives any.

☐ Step 4: Accept
  ■ Each receiver accepts one of the grants, if it receives any.
Pipelining Packet Scheduling Process

- Relax the time constraint, especially for optical interconnects
  - Simpler implementation of existing scheduling algorithms
- Allow more sophisticated scheduling algorithms
  - Further improve system performance

(a) without pipelining

(b) with pipelining
Pipelined Scheduling in OpCut Switch

- k sub-schedulers
  - The \textit{i-th} sub-scheduler responsible for the scheduling of \textit{i-th} oldest flows. A schedule is calculated in k time slots.

- Intermediate results are relayed from one sub-scheduler to another.

- At any time instant
  - Different sub-schedulers work on different schedules.
  - Multiple schedules are being calculated.

- FDL attached at the switch input to provide necessary packet delay for pipelining.
Two Sub-schedulers (k = 2)

<table>
<thead>
<tr>
<th>time slot</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>...</th>
<th>t</th>
<th>t+1</th>
<th>t+2</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>ss(_1)</td>
<td>S(_1)^2</td>
<td>S(_1)^3</td>
<td>S(_1)^4</td>
<td>S(_1)^5</td>
<td>...</td>
<td>S(_1)^t+2</td>
<td>S(_1)^t+3</td>
<td>S(_1)^t+4</td>
<td>...</td>
</tr>
<tr>
<td>ss(_2)</td>
<td>S(_2)^2</td>
<td>S(_2)^3</td>
<td>S(_2)^4</td>
<td>...</td>
<td>S(_2)^t+1</td>
<td>S(_2)^t+2</td>
<td>S(_2)^t+3</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>schedule</td>
<td>S(_1)^2</td>
<td>S(_1)^3</td>
<td>...</td>
<td>S(_1)^t</td>
<td>S(_1)^t+1</td>
<td>S(_1)^t+2</td>
<td>...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- It can be proved that there is not no duplicate scheduling of packets when k = 2.
- Duplicate scheduling: the same packet is included in multiple schedules (waste bandwidth).
More Sub-schedulers (k > 2)

- Same basic idea

<table>
<thead>
<tr>
<th>time slot</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>...</th>
<th>k-1</th>
<th>k</th>
<th>...</th>
<th>t</th>
<th>t+1</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>ss_1</td>
<td>S_1^k</td>
<td>S_1^{k+1}</td>
<td>S_1^{k+2}</td>
<td>...</td>
<td>S_1^{2k-1}</td>
<td>S_1^{2k}</td>
<td>...</td>
<td>S_1^{t+k}</td>
<td>S_1^{t+k+1}</td>
<td>...</td>
</tr>
<tr>
<td>ss_2</td>
<td>S_2^k</td>
<td>S_2^{k+1}</td>
<td>S_2^{k+2}</td>
<td>...</td>
<td>S_2^{2k-2}</td>
<td>S_2^{2k-1}</td>
<td>...</td>
<td>S_2^{t+k-1}</td>
<td>S_2^{t+k}</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>ss_k</td>
<td>...</td>
<td>S_k^k</td>
<td>S_k^{k+1}</td>
<td>...</td>
<td>S_k+t</td>
<td>S_k</td>
<td>...</td>
<td>S_k+t+1</td>
<td>S_k</td>
<td>...</td>
</tr>
</tbody>
</table>

- Can no longer eliminate duplicate scheduling of packets but we found a way to reduce it.
Adaptive Pipelining

- Pipelining introduces extra delay in generating a schedule
  - With $k$ sub-schedulers, the delay is $k$ time slots.
- Minimize delay by adaptive pipelining
  - Adjust the number of sub-schedulers according to traffic intensity (under heavier traffic it needs more time to generate a schedule)
  - Under low traffic intensity, no need to pipeline or use fewer sub-schedulers.
  - Need to combine FDL with switches to provide flexible delay at the OpCut switch input.
Adaptive Pipelining – An Example

Traffic intensity versus time

Sampling of average packet delay over time

- adaptive pipeline
- non-pipeline
- 2-scheduler pipeline
- 3-scheduler pipeline
Each fibers has multiple wavelength channels.

Need $N \times W$ receivers and transmitters ($W$ is the number of wavelengths per fiber).
WDM OpCut Switch

- Cut through can still be handled by a single iteration of iSLIP.
- Packets that fail to cut through can be picked up by receivers in a similar way to that in the single wavelength case.
- However, scheduling packets from electronic buffers to interconnect outputs becomes more challenging.
WDM OpCut Switch - Scheduling packets from buffers to outputs

- Can be formalized into the Maximum-Coverage Prefixes (MCP) Problem
  - Input: a collection of sequences of elements; No element is repeated in a single sequence.
  - Output: a prefix of each sequence, such that the maximum number of elements are covered by all the prefixes, and no element appears in more than one prefix.

- NP-hard and hard to approximate: no constant-factor approximation algorithm that runs in polynomial time.
Longest-or-Heads (LOH) Approximation Algorithm

- Basic idea as the name suggests
  - Take the longest sequence, or consider only heads of sequences, whichever results in more packets being scheduled.

- Approximation bound
  - $\sqrt{OPT}$: if the optimal solution schedules $OPT$ packets, LOH schedules at least $\sqrt{OPT}$ packets.

- Achieve good performance in practice.
Outline

- Background
  - Interconnect architectures and packet scheduling
  - Current optical technology
- Scheduling in all-optical interconnects
  - Unbuffered interconnects
  - Interconnects with output buffers
  - Interconnects with input buffers
- Scheduling in hybrid interconnects
- Summary
Optical interconnects are widely considered as the most promising candidate to provide ultra-high speed interconnections for future applications.

Studied packet scheduling in WDM interconnects with limited range wavelength conversion

- **First Available Algorithm** for unbuffered WDM interconnects
- **Augment to Full Algorithm** for output-buffered WDM interconnects
- **Most-Packet Wavelength-Fiber Pair First Algorithm** for input-buffered WDM interconnects
Summary

- Proposed the OpCut switch
  - Built with existing techniques and features low packet latency
  - Simple architecture and implementation-friendly
  - Studied packet scheduling in both single-wavelength and WDM OpCut switches
    - Single wavelength: presented a basic scheduling algorithm and an adaptive pipelining mechanism
    - WDM: proved NP-hardness and inapproximability within constant factor, presented a bounded approximation algorithm, **Longest-or-Heads Approximation Algorithm**
Related Publications


X. Qin and Y. Yang, ``Optical crossconnect architectures for wavelength routed WDM networks,'' *Optical Networks Magazine*, vol. 4, no. 4, pp. 50-63, July/August 2003.


Thank you!

Questions?