Prediction Model for Evaluation of Reconfigurable Interconnects in Distributed Shared-Memory Systems

Wim Heirman
Ghent University, Belgium
Outline

• Introduction
• Reconfigurable Optical Networks
• Prediction Model
• Results
• Future work & conclusions
Architecture of a distributed shared-memory system

- Nodes:
  - Processor
  - Caches
  - Main memory
  - Network interface

- Interconnection network
  - Packet switched
Architecture of a distributed shared-memory system

'Remote' memory access: handled by the network interfaces, requires use of the interconnection network.
Interconnect requirements

• Network latency is a major bottleneck:
  - Instruction (.5 ns)
  - Local memory access (50 ns)
  - Remote memory access (500 ns)
Interconnect requirements

Non-uniform network traffic in space and time

=> Reconfigurable network?
Outline

• Introduction
• Reconfigurable Optical Networks
• Prediction Model
• Results
• Future work & conclusions
Reconfigurable Optical Networks

- **WDM (wavelength division multiplexing)**
  - Tunable lasers / detectors
  - Passive star coupler (PSC)

![Diagram of reconfigurable optical network with nodes A, B, and C connected through a passive star coupler (PSC). Connections indicate A -> B, B -> C, and C -> A.]
Reconfigurable Optical Networks

- **WDM (wavelength division multiplexing)**
  - Tunable lasers / detectors
  - Passive star coupler (PSC)
Reconfigurable Optical Networks

- Photonic Crystal components (crossbar)

Source: D. Prather, University of Delaware
Reconfiguration in shared-memory machines

- Reconfiguration speed: up to 1 ms
- One memory access: < 1 μs
- Locality needed in address streams!

(Traffic Temporal Analysis for Reconfigurable Interconnects in Shared-Memory Systems, W. Heirman et. al., Reconfigurable Architectures Workshop, April 4-5, 2005, Denver, CO)
Reconfiguration in shared-memory machines

Traffic burst durations

# Bursts

0 20 40 60 80 100

Time (ms)

5 10 15
Reconfiguration in shared-memory machines

Base network (fixed)
Extra links (reconfigurable)
Reconfiguration in shared-memory machines

- Requirement:
  Reconfiguration time $\ll$ reconfiguration interval
  $\ll$ burst duration
Evaluating network performance

• Full-system simulations are needed:
  - Current statistical traffic models don't exhibit the 'bursty behavior' exploited here
  - 'Application speedup' cannot be derived from network performance alone
• The simulation needs to model tens of processors, caches, and the interconnection network
• Different benchmarks
Evaluating network performance

Evaluating just one set of network parameters takes hours of simulations...

How can we do this faster?

Derive performance for several sets of network parameters from one simulation!
Outline

- Introduction
- Reconfigurable Optical Networks
- Prediction Model
- Results
- Future work & conclusions
Predicting network performance

One full-system simulation

Estimate extra link placements

Estimate memory access times

Predict speedup

Estimate extra link placements

Estimate memory access times

Predict speedup

network packets
memory accesses

Our prediction model

for each parameter set
Predicting network performance

- Estimate extra link placement:

Parameters: reconfiguration interval (\( \delta t \)), number of extra links (\( n \)), link placement algorithm

\( \delta t = 1 \), \( n = 2 \)

\( \delta t = 2 \), \( n = 4 \)
Predicting network performance

• Estimate new memory access latency for each transaction:

  No change  No change(!)  Reduced access time
Predicting network performance

• Predict application speedup:

Original execution time ➔ New execution time

Application speedup: 2.13

- computation time (constant)
- unchanged fraction of memory latency
- reduced fraction of memory latency
Outline

• Introduction
• Reconfigurable Optical Networks
• Prediction Model
• Results
• Future work & conclusions
Results

Application speedup

Predicted speedup

Measured speedup

barnes, cholesky, fft, fmm, lu, ocean, cont, ocean, ncont, radiosity, radix, raytrace, volrend, water, nsq, water, sp
Assumptions

- Access latency is not hidden by out-of-order execution
- Average reduction factor is used for all improved memory accesses (2.13 for 4x4 torus network)
- Memory accesses require only 2 nodes
- Computation time remains constant
- Congestion is not modeled
- Any combination of extra links can be made
- Extra links are not used as part of a path
Results: application variability

- Correlation between computation time variability and prediction error is high, this could explain larger errors in some benchmarks.
Results: different parameters

FFT benchmark, results for different reconfiguration intervals and # extra links: good relative prediction
Outline

• Introduction
• Reconfigurable Optical Networks
• Prediction Model
• Results
• Future work & conclusions
Future work

• **Access latency is not hidden by out-of-order execution**
• **Average reduction factor is used for all improved memory accesses** (2.13 for 4x4 torus network)
• **Memory accesses require only 2 nodes**
• **Computation time remains constant**

- Congestion is not modeled
- Any combination of extra links can be made
- Extra links are not used as part of a path
Conclusions

- Using our technique, good predictions can be made using much less time-consuming simulations.
- Good relative accuracy over a range of parameters allows for quick design-space exploration.
- Further refinements can be made by including application variability and congestion.