System-Level Interconnect Prediction 2014

Dynamic Routing of Hierarchical On Chip Network Traffic

Julian Kemmerer • jvk@drexel.edu
Baris Taskin • taskin@coe.drexel.edu
Electrical & Computer Engineering
Drexel University
1 June 2014

Drexel VLSI Laboratory
vlsi.ece.drexel.edu
Hierarchical On Chip Network

- Hierarchical NoC:
  - Designed to alleviate congestion issues common to flat topologies.
  - Accomplished using addition interconnect layers.
  - Homogeneous layers (e.g. mesh placed over another mesh).
  - Heterogeneous layers (e.g. pyramid shape).

* R. Manevich, I. Cidon, A. Kolodny, “Handling global traffic in future CMP NoCs,” Electrical and Electronics Engineers in Israel (IEEEI), Nov. 2012, pp. 1, 5, 14-17
• Known that hierarchical networks can suffer from an inability to balance network traffic across all layers.

• Long distance ('global') traffic tends to be a small portion of total traffic.
  • However, bandwidth allocated to global traffic is significant (%50+).

• Common metric for dividing local v.s. global traffic is the Manhattan distance between source and destination nodes on the lowest hierarchy level.
Local Router Range

- From previous work* that builds off of Manhattan distance as metric for routing decisions.
- Uses subdivided lowest hierarchy layer (subnets).
- Subnets that are not allowed to communicate (without routing through upper hierarchy) are said to be ‘hard-walled’.
- 'Leaky-walls' that allow subnets to communicate were proposed as solution to congestion on upper hierarchy levels.
- Local router range determines when to use 'leaky' subnet boundaries.

Local Router Range (LRR), defined as:

*Integer distance from the destination node within which the source node injects packets into the bottom-level network instead of the top-level network.*
Local Router Range (continued)

- Subnet Size: 2x2
- Manhattan Distance = 2
- LRR = 3

- Case 1)
  If the destination node is within the same sub-net as the source node, then the traffic is deemed local enough to be routed on the lowest hierarchy level.
- Subnet Size: 2x2
- Manhattan Distance = 3
- LRR = 3

- Case 2)
  If the destination node is not in the same sub-net as the source core, and the Manhattan distance between source and destination on the lowest hierarchy level is less than or equal to the LRR, then the lowest hierarchy level will be used.
Local Router Range (continued)

- Subnet Size: 2x2
- Manhattan Distance = 4
- LRR = 3

- Case 3)
  If the destination node is not in the same sub-net as the source core, and the number of hops is greater than the LRR, the packet is routed onto the higher hierarchy level.
Local Router Range (continued)

- Previous work:
  - Single static LRR value.
  - Statically balanced traffic across hierarchy levels.

- Presented technique(s) examine the effect of:
  - Various static LRR values.
  - Modification the LRR dynamically at run time.
Experimental Setup

- Custom cycle-accurate NoC simulator.
- Three topologies examined:
  - MeshX2
  - CMeshX2
  - MeshMesh
- Four traffic types examined:
  - Random, Localization = 25%, 50%, and 75%.
  - Random: Source and destination node chosen randomly.
  - Localization Percentage: Percentage of traffic has destination node within two hops of source node.
Effect of Static LRR Values

- Example case:
  - Fixed CMeshX2 topology.
  - Fixed injection rate and traffic pattern.

- Examined average throughput and delay.

- Difference between best performing and worst performing LRR values:
  - Throughput = 1.8X
  - Delay = 3.4X

- Overall maximum performance difference: 1.8X

- Issues:
  - Single LRR value is not optimal in both throughput and delay.
  - Fixed injection rate.
Performance Metric

- Combination of throughput and delay.
- Need a single 'optimal' LRR value.
- Performance defined as:

\[
\text{Performance} = \frac{\text{Throughput}}{\text{Delay}} \text{ (flits/cycle}^2)\]

- If throughput increases and/or delay decreases the ‘performance’ of the NoC has increased.
Static LRR and PIR Sweep

- Each (LRR, PIR) pair yields a single performance value.
  - Can now identify maximally performing (LRR, PIR) pair for each topology.
  - However, link exists between PIR and throughput.
  - Not looking to optimize PIR.

- Important Questions:
  - At which PIR does the sweep of LRR values yield the largest change in performance?
  - At that PIR, which LRR value yields the best performance?

<table>
<thead>
<tr>
<th>Topology</th>
<th>Static PIR (pkts/cycle)</th>
<th>Static LRR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MeshX2</td>
<td>0.18</td>
<td>3</td>
</tr>
<tr>
<td>CMeshX2</td>
<td>0.05</td>
<td>6</td>
</tr>
<tr>
<td>MeshMesh</td>
<td>0.06</td>
<td>8</td>
</tr>
</tbody>
</table>
Dynamic Local Router Range

• Ideally:
  • Instantaneous knowledge of network wide performance.
  • Use this as feedback for dynamic LRR control system.

• Realistically:
  • LRR Control → LRR → NoC
  • Estimated Performance
  • Performance Estimation
  • LRR → NoC
  • ?
Dynamic LRR (continued)

- Issues:
  - Centralized – difficult to implement in hardware.
  - '?' - What to pull from NoC that indicates performance?
  - 'Performance Estimation' – How to use that information to estimate performance?
  - 'LRR Control' – What control system will govern LRR changes?
Circumventing Centralized Control Systems

- Standard NoC Tile:
  - Router + Processing Element

![Diagram of a standard NoC tile consisting of a Router and a Processing Element with RX/TX ports.]
Circumventing Centralized Control Systems (continued)

- Distributed LRR control:
LRR Control
System Parameters

• **AFS** - Average Free Slots (buffer space) available in neighboring routers.
  • Requires connection to neighboring routers.
  • Value is average buffer space available across all connected neighbors.

• **APQ** - Average Packets in processing element Queue.
  • Value is average over all virtual channel queues.

• These values should indicate, to some degree, network congestion at and around a specific NoC tile.
Circumventing Centralized Control Systems (continued)

- Actual System Implementation:
  - Dynamic and distributed control.
  - FS: Free Slots
  - APQ: Average packets in processing element queue
Dynamic LRR Control System
(continued)

- Remaining Issues:
  - 'Performance Estimation' – How to use AFS/APQ information to estimate performance?
  - 'LRR Control' – What control system will govern LRR changes?
Performance Estimation Using AFS and APQ

- Perf(AFS) = ?, Perf(APQ) = ?

- Each LRR CTRL block should be identical.

- Use data from NoC simulator.

- Record run time values (AFS, APQ, Performance).

- Repeat for all four traffic conditions.
Performance as function of APQ for MeshX2 Topology.

Performance as function of AFS for MeshX2 Topology.

- These Perf(APQ) and Perf(AFS) relationships are required for a LRR control system.
Performance Estimation Using AFS and APQ (continued)

- Data from all traffic patterns combined.
- Regression lines shown.

Performance as function of APQ for MeshX2 Topology.

Performance as function of AFS for MeshX2 Topology.
LRR Control

- Estimated performance now available to LRR Control blocks.
- Control Algorithm: How and how often to adjust LRR values?

![LRR Control Diagram](image)
LRR Control Algorithm

#Inputs: LRR, prev_LRR, perf, prev_perf, counter

# Initialization
# LRR = 3
# prev_LRR = LRR - 1

if counter % n == 0:
    # Update
    delta = LRR - prev_LRR
    prev_LRR = LRR

if (perf - prev_perf) > 0:
    # Performance increase
    LRR += delta
else if (perf - prev_perf) < 0:
    # Performance decrease,
    # 2 steps back to make up for mistake
    LRR -= 2*delta
    # But keep old local router range
    # at just one step back
    prev_LRR = LRR + delta
Algorithm Evaluation
Frequency – 'n' Value Sweep

• 'n' : How often the adjust the LRR value.

Performance as function of 'n' parameter for MeshMesh topology.
Comparison to Static LRR

- Dynamic LRR vs. optimal static LRR.
- Ratio of dynamic performance / static performance.

Dynamic LRR / static LRR performance ratio for MeshX2 topology.
Comparison to Static LRR (continued)

Dynamic LRR / static LRR performance ratio for CMeshX2 topology.
Comparison to Static LRR (continued)

Dynamic LRR / static LRR performance ratio for MeshMesh topology.
Summary and Conclusions

- Local router range as a mechanism for hierarchical routing decisions
- Optimal static LRR
  - Non-optimal v.s. optimal static LRR performance: 1.8X
- Dynamic routing without being a dynamic routing algorithm
  - Deadlock free
- Decentralized control
  - Minimal design overhead
- Dynamic LRR performance > Static LRR
  - MeshX2 and CMeshX2 topologies
  - At all packet injection rates
  - For all traffic types
  - Maximum dynamic v.s. optimal static LRR performance difference: 1.37X
- Maximum performance increase over non-optimal static LRR:
  - $1.8 \times 1.37 = 2.47X$

<table>
<thead>
<tr>
<th>Topology</th>
<th>Maximum Perf. Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>MeshX2</td>
<td>1.37</td>
</tr>
<tr>
<td>CMeshX2</td>
<td>1.22</td>
</tr>
<tr>
<td>MeshMesh</td>
<td>1.07</td>
</tr>
</tbody>
</table>