Flexible Snooping:
Adaptive Forwarding and Filtering of Snoops in Embedded-Ring Multiprocessors

Karin Strauss, Xiaowei Shen*, Josep Torrellas

University of Illinois at Urbana-Champaign
*IBM Research

http://iacoma.cs.uiuc.edu
Motivation

• CMPs are becoming standard components

• cheaper to build medium size machines
  – 32 to 128 cores (multi-CMP)

• shared memory, cache coherent
  – easier to program, easier to manage

• supporting cache coherence is difficult
# Cache coherence solutions

<table>
<thead>
<tr>
<th>strategy</th>
<th>ordered network?</th>
<th>pros</th>
<th>cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>snoopy broadcast bus</td>
<td>yes</td>
<td>simple</td>
<td>difficult to scale</td>
</tr>
<tr>
<td>directory based protocol</td>
<td>no</td>
<td>scalable</td>
<td>indirection, extra hardware</td>
</tr>
<tr>
<td>snoopy embedded ring</td>
<td>no</td>
<td>simple</td>
<td>long latencies</td>
</tr>
</tbody>
</table>

• other proposals (e.g. token coherence)
Contributions

• family of adaptive coherence protocols for rings
• two were chosen as best options

compared to fastest state-of-the-art scheme
Multi-CMP multiprocessor

- coherence protocol used: only one supplier if line is cached
Ring in action

Lazy

Eager

Oracle

cmp

supplier predictor

snoop

request

response

Karin Strauss
Flexile Snooping
Ring in action

- **Lazy**
- **Eager**
- **Oracle**

- **goal:** adaptive schemes that approximate Oracle’s behavior
Primitive snooping actions

- snoop and then forward
  + fewer messages

- forward and then snoop
  + shorter latency

- forward only
  + fewer snoops
  + shorter latency
  – false negative predictions not allowed
Predictors and algorithms

<table>
<thead>
<tr>
<th>predictor / algorithm</th>
<th>action on negative prediction</th>
<th>action on positive prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subset</td>
<td>forward then snoop</td>
<td>snoop</td>
</tr>
<tr>
<td>Super set</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Con</td>
<td>forward</td>
<td>snoop then forward</td>
</tr>
<tr>
<td>Agg</td>
<td>forward</td>
<td>forward then snoop</td>
</tr>
<tr>
<td>Exact</td>
<td>forward</td>
<td>snoop</td>
</tr>
</tbody>
</table>
### Algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Negative</th>
<th>Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subset</td>
<td>forward then snoop</td>
<td>snoop</td>
</tr>
<tr>
<td>Superset</td>
<td>snoop then forward</td>
<td>forward</td>
</tr>
<tr>
<td>SupersetAgg</td>
<td>forward then snoop</td>
<td></td>
</tr>
<tr>
<td>Exact</td>
<td>forward</td>
<td>snoop</td>
</tr>
</tbody>
</table>

**Per miss service:**

- Lazy
- Eager
- Oracle / Exact
- Subset
- SupersetCon
- SupersetAgg

---

Karin Strauss

Flexible Snooping
Predictor implementation

- **Subset**
  - associative table: subset of addresses that can be supplied by node

- **Superset**
  - bloom filter: superset of addresses that can be supplied by node
  - associative table (exclude cache): addresses that recently suffered false positives

- **Exact**
  - associative table: all addresses that can be supplied by node
  - downgrading: if address has to be evicted from predictor table, corresponding line in node has to be downgraded
Downgrading

Negative effects:

• writes by this node need to snoop other nodes

• reads and writes by other nodes need to fetch line from memory
Experiments

- 8 CMPs, 4,000 cores each = 32 cores
  - private L2 caches
- on-chip bus interconnect
- off-chip 2D torus interconnect with embedded unidirectional ring
- per node predictors: latency of 3 processor cycles
- sesc simulator (sesc.sourceforge.net)
- SPLASH-2, SPECjbb, SPECweb
• performance of most flexible snooping algorithms is similar to Eager

• the fastest of all algorithms is SupersetAgg
• algorithms that eagerly forward messages use more energy

• SupersetCon is least energy-hungry algorithm
Most cost-effective algorithms

- **high performance:** Superset Aggressive
  - faster than Eager at lower energy consumption

- **energy conscious:** Superset Conservative
  - slightly slower than Eager at much lower energy consumption
Most cost-effective algorithms compared to **fastest** state-of-the-art scheme (Eager)

**Superset Aggressive**
- high performance scheme

**Superset Conservative**
- energy conscious scheme

performance | energy consumption
--- | ---
↑ | ↓

can be combined by only changing forwarding policy
Conclusions

• proposed **flexible snooping**, a family of adaptive protocols for embedded rings

• two chosen protocols
  – high performance: **Superset Aggressive**
  – energy conservation: **Superset Conservative**
  – can be selected dynamically

• embedded-ring protocols more attractive
Arch map

http://iacoma.cs.uiuc.edu/students/archmap/archmap.html
Flexible Snooping:
Adaptive Forwarding and Filtering of Snoops in Embedded-Ring Multiprocessors

Karin Strauss, Xiaowei Shen*, Josep Torrellas

University of Illinois at Urbana-Champaign
*IBM Research

http://iacoma.cs.uiuc.edu