Dynamically Reconfiguring Multi-Core Architectures using Task Graph based Analysis

Sriseshan Srikanth, Brian P. Railing, Thomas M. Conte

School of Computer Science, Georgia Institute of Technology

There is significant variation in resource usage and requirements across generic parallel workloads, across threads within a workload as well as across various phases within a thread.

Significant performance and energy benefits result from dynamically reconfiguring multi-core architectures to achieve optimal resource utilization.

... $A[i] = B[i] * C[i]$

A hardware agnostic dynamic trace of a parallel program complete with basic block information, memory address and synchronization traces.

An analysis that abstracts coherence misses into a stochastic variable.

A hardware agnostic dynamic trace of a parallel program complete with basic block information, memory address and synchronization traces.

... $i = add i32 \%i, i32 \%4$ store $i32 \%6, i32 * \%i$

An analysis that computes critical path of a Task Graph using the recorded task times and speedup estimates.

The plot below depicts the accuracy of the predicted bottleneck resource vis-à-vis an oracle dynamic configuration obtained by a brute force configuration sweep.

<table>
<thead>
<tr>
<th>Function Unit</th>
<th>Baseline Count</th>
<th>Function Unit</th>
<th>Baseline Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int Add</td>
<td>3</td>
<td>FP Add</td>
<td>1</td>
</tr>
<tr>
<td>Int Mul</td>
<td>1</td>
<td>FP Mul</td>
<td>1</td>
</tr>
<tr>
<td>Int Div</td>
<td>1</td>
<td>FP Div</td>
<td>1</td>
</tr>
</tbody>
</table>

Average normalized energy-delay product reduction of 32%