Mage: Online and Interference-Aware Scheduling for Multi-Scale Heterogeneous Systems

Francisco Romero\textsuperscript{1} and Christina Delimitrou\textsuperscript{2}
\textsuperscript{1}Stanford University, \textsuperscript{2}Cornell University

PACT – Session 4a – November 2, 2018
Motivation

• Heterogeneity is becoming more prevalent
  • Different server generations
  • Advanced management features, e.g., power management

• Allows for systems to better match applications to the underlying hardware

• **Challenge**: How do we maximize application performance *and* maintain high resource utilization?
## Prior Work

<table>
<thead>
<tr>
<th>System</th>
<th>Heterogeneous Clusters</th>
<th>Heterogeneous CMPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paragon</td>
<td>✓</td>
<td>❌</td>
</tr>
<tr>
<td>Whare-map</td>
<td>✓</td>
<td>❌</td>
</tr>
<tr>
<td>Bubble-flux</td>
<td>✓</td>
<td>❌</td>
</tr>
<tr>
<td>Composite cores</td>
<td>❌</td>
<td>✓</td>
</tr>
<tr>
<td>Hass</td>
<td>❌</td>
<td>✓</td>
</tr>
<tr>
<td>PIE</td>
<td>❌</td>
<td>✓</td>
</tr>
</tbody>
</table>
The Problem with “Sum of Schedulers”

- Suboptimal performance
- Revisit several scheduling decisions

Need a data-driven approach to avoid exhaustive search

Exhaustive Search
- High overhead
- Not scalable
Mage

- Tiered runtime scheduler that considers inter- and intra-server heterogeneity jointly
- Leverages fast and online data mining to quickly explore the space of application placements
- Lightweight application monitoring and rescheduling
- Heterogeneous CMPs: 38% average improvement compared to a greedy scheduler
- Heterogeneous Cluster: 30% average improvement compared to a greedy scheduler and 11% average improvement compared to a heterogeneity- and interference-aware scheduler
Mage Master and Mage Agents

**Mage Master**
- Runs inference
- Makes optimal application-to-resource scheduling decision
- Decides when applications should be migrated/rescheduled

**Mage Agent**
- Monitor the performance of all scheduled applications
- Notify the master when QoS violations occur
Application Arrival and Initial Scheduling
What we want

<table>
<thead>
<tr>
<th>Applications</th>
<th>Resource 1</th>
<th>Resource 2</th>
<th>Resource 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>App 1</td>
<td>Core 1</td>
<td>Core 1</td>
<td>Core 3</td>
</tr>
<tr>
<td></td>
<td>Core 1</td>
<td>Core 3</td>
<td>Core 2</td>
</tr>
</tbody>
</table>

✓ Heterogeneous resources that benefit an application
✓ Performance impact of co-scheduling applications

How can Mage quickly and accurately generate this matrix?
Collaborative Filtering

• Use Single Value Decomposition (SVD) with PQ-Reconstruction (SGD) to uncover:
  • Heterogeneous resources that benefit individual applications
  • Interference that can be tolerated between applications
Contentious Kernel Profiling

Common reference point for the sensitivity of new applications to interference of shared resources
Co-Scheduling Sensitivity
## Co-Scheduling Sensitivity

<table>
<thead>
<tr>
<th>App1</th>
<th>App2:Core1</th>
<th>App2:Core2</th>
<th>App2:Core3</th>
<th>App3:Core1</th>
<th>App3:Core2</th>
<th>App3:Core3</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIPS(_{1,1})</td>
<td>MIPS(_{1,2})</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>MIPS(_{2,1})</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>MIPS(_{2,6})</td>
</tr>
<tr>
<td>MIPS(_{3,1})</td>
<td>?</td>
<td>MIPS(_{3,3})</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
# Co-Scheduling Sensitivity

Profile of the impact of co-scheduling applications on all combinations of resources

<table>
<thead>
<tr>
<th></th>
<th>App1:Core1</th>
<th>App1:Core1</th>
<th>App1:Core2</th>
<th>App1:Core2</th>
<th>App1:Core3</th>
<th>App1:Core3</th>
</tr>
</thead>
<tbody>
<tr>
<td>App1</td>
<td>MIPS_{1,1}</td>
<td>MIPS_{1,2}</td>
<td>MIPS_{1,3}</td>
<td>MIPS_{1,4}</td>
<td>MIPS_{1,5}</td>
<td>MIPS_{1,6}</td>
</tr>
<tr>
<td>App2</td>
<td>MIPS_{2,1}</td>
<td>MIPS_{2,2}</td>
<td>MIPS_{2,3}</td>
<td>MIPS_{2,4}</td>
<td>MIPS_{2,5}</td>
<td>MIPS_{2,6}</td>
</tr>
<tr>
<td>App3</td>
<td>MIPS_{3,1}</td>
<td>MIPS_{3,2}</td>
<td>MIPS_{3,3}</td>
<td>MIPS_{3,4}</td>
<td>MIPS_{3,5}</td>
<td>MIPS_{3,6}</td>
</tr>
</tbody>
</table>
Initial Application Placement
Runtime Monitoring and Rescheduling

- Increase resources locally
- Migrate from smaller core to bigger core
- Migrate across servers

Least invasive

Most invasive
Evaluation

- **Workloads**
  - Single- and multi-threaded benchmark suites
  - Latency-critical, interactive services

- **Execution scenarios**
  - Simulated heterogeneous 16-core CMP
  - Real 40-server heterogeneous cluster
  - Real cluster with core-level heterogeneity using power management (DVFS)

- **Comparison schedulers**
  - Greedy, Smallest-First, Mage-Static, PIE [ISCA’12], Paragon [ASPLOS’13]
Mage has low initial scheduling overhead and low estimation error

- Reduces the need to adjust scheduling decisions frequently during application lifetime
Mage outperforms the Greedy scheduler by only allocating the necessary resources to meet an application’s QoS
Mage outperforms the Smallest-First scheduler by not exacerbating contention in shared resources
Versus Mage-Static

Mage outperforms *Mage-Static* by rescheduling applications that were mispredicted or that exhibit diurnal patterns.
Mage outperforms Paragon+PIE and Paragon+Paragon by having a global view of resource availability and per-application resource requirements.
Sensitivity to Heterogeneity Increase

- As degree of heterogeneity increases, the benefits of using Mage also increases
  - Results are also consistent for heterogeneous CMPs
- Minimal scheduling overhead as degree of heterogeneity increases
Conclusion

- Heterogeneity is becoming more prevalent; need a scheduler that can match applications to their resource needs
- Mage is a tiered scheduler that bridges the gap between CMP- and cluster-level heterogeneous scheduling
- Mage leverages a novel staged, parallel SGD algorithm to quickly and accurately classify applications
- Mage is lightweight and scalable
- Mage outperforms heterogeneity-agnostic and the sum of CMP- and cluster-level schedulers
Thank you!

Questions?

faromero@stanford.edu
Backup
Versus Paragon
Versus PIE

![Graph showing Speedup Gmean versus Application Mix for Heterogeneous CMP](image-url)
Partial Interference Sensitivity – SGD Step 2

Solution: Run SGD *without* those columns, and add them in afterwards
Partial Interference Sensitivity – SGD Step 2

Solution: Run SGD *without* those columns, and add them in afterwards.
Partial Interference Sensitivity – SGD Step 2

Solution: Run SGD *without* those columns, and add them in afterwards
Complete Placements – SGD Step 3

\[
\begin{bmatrix}
A_{SGD2}
\end{bmatrix}
\]

Populate remaining columns with results from Partial Placements

<table>
<thead>
<tr>
<th></th>
<th>App1:Core2</th>
<th>App1:Core3</th>
<th>App2:Core3</th>
<th>App2:Core1</th>
<th>App3:Core1</th>
<th>App3:Core2</th>
</tr>
</thead>
<tbody>
<tr>
<td>App1</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td></td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td></td>
</tr>
<tr>
<td>App2</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td></td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td></td>
</tr>
<tr>
<td>App3</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td></td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td>[min\textsuperscript{\textit{ASGD2}}, max\textsuperscript{\textit{ASGD2}}]</td>
<td></td>
</tr>
</tbody>
</table>
Complete Placements – SGD Step 3

Select column from $A_{SGD3}$ with highest geometric mean for scheduling