Towards Demystifying Intra-Function Parallelism in Serverless Computing

7th International Workshop on Serverless Computing (WoSC7)
In conjunction with, ACM/IFIP Middleware 2021

**Michael Kiener**, Mohak Chadha, Michael Gerndt
michael.kiener@tum.de, mohak.chadha@tum.de, gerndt@in.tum.de
Technische University of Munich (TUM)
Chair of Computer Architecture and Parallel Systems
Garching (near Munich), Germany
7th of November, 2021
https://www.serverlesscomputing.org/wosc7/papers/p7
Outline

1. Motivation
2. Methodology
3. Results
   3.1. CPUs to vCPUs Mapping
   3.2. Performance Results
   3.3. Cost Analysis
4. Conclusion
Motivation

- Serverless services function like black-boxes
- Developer doesn’t have full insights about underlying hardware & infrastructure
- Underutilization of computing resources increases costs significantly
Methodology

• 3 Microbenchmarks
  • Atax
  • Go fast
  • MVT

• 2 Applications
  • Heat
  • Monte Carlo

<table>
<thead>
<tr>
<th>Language</th>
<th>Parallelization</th>
<th>Version</th>
<th>Compiler &amp; Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>C++</td>
<td>OpenMP</td>
<td>C++11</td>
<td>g++ -O3</td>
</tr>
<tr>
<td>Java</td>
<td>ExecutorService</td>
<td>Java 11</td>
<td>OpenJDK 11</td>
</tr>
<tr>
<td>Go</td>
<td>Goroutines</td>
<td>1.16</td>
<td>gc GOOS=linux</td>
</tr>
</tbody>
</table>
Results - Cores to vCPU Mapping

CPU Cores Allocation

<table>
<thead>
<tr>
<th>Memory in MB</th>
<th>CPU Cores</th>
</tr>
</thead>
<tbody>
<tr>
<td>512</td>
<td>2</td>
</tr>
<tr>
<td>2048</td>
<td>2</td>
</tr>
<tr>
<td>4096</td>
<td>3</td>
</tr>
<tr>
<td>8192</td>
<td>5</td>
</tr>
<tr>
<td>10240</td>
<td>6</td>
</tr>
</tbody>
</table>
Results - Performance

Parallel Speedups for AWS Lambda

Go fast

MVT

Heat

Monte Carlo
Results - Costs

Cost Comparison for AWS Lambda

<table>
<thead>
<tr>
<th>Language</th>
<th>Benchmark</th>
<th>Configuration</th>
<th>Maximum Cost Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>C++</td>
<td>Monte Carlo</td>
<td>10240MB</td>
<td>80,1 %</td>
</tr>
<tr>
<td>Java</td>
<td>Monte Carlo</td>
<td>10240MB</td>
<td>74,1 %</td>
</tr>
<tr>
<td>Go</td>
<td>Monte Carlo</td>
<td>10240MB</td>
<td>81,3 %</td>
</tr>
</tbody>
</table>

Michael Kiener | Towards Demystifying Intra-Function Parallelism in Serverless Computing | WoSC7
Cold starts

Impact of billable time difference of cold starts

![Bar chart showing billable cold start latency in ms for C++, Go, and Java on AWS](image-url)
Conclusion

• Allocated CPUs not always equal to vCPUs

• Parallelization efficiency heavily depends on service and configuration

• Maximum cost savings achieved by parallelizing
  • AWS: 81%
Thank you for your attention

Questions?