Towards Simulation-Driven Optimization of Container Orchestration Mechanisms

14th Symposium on Software Performance (2023)  
Session 4: Quality

Timo Dittus, Martin Sträßer, Samuel Kounev – Descartes Research Group

08.11.2023

https://se.informatik.uni-wuerzburg.de
Introduction

Typical container orchestration tasks (definition by Google):

➢ Provisioning and deployment
➢ Scaling containers up or down and load balancing
➢ Allocating resources between containers
➢ Moving containers to another host to ensure availability if there’s a shortage of resources or an unexpected outage
➢ Performance and health monitoring of the application
➢ Service discovery
Container orchestration (CO) frameworks significantly impact performance
Motivation

- Testing configurations for CO mechanisms is costly and time-consuming, because:
- CO mechanisms have a (very) large variety of configuration parameters
- … where one mechanism’s configuration can exert influence on another’s

More pods of ai-service please!

Provision more nodes with a GPU please!

No machines with a GPU left!

Automatically optimizing configurations via a simulation strongly preferrable!
Recap: MiSim

MiSim: A Simulator for Resilience Assessment of Microservice-based Architectures

Sebastian Frank$^{1,2}$, Lion Wagner$^2$, Alireza Hakamian$^2$, Martin Straesser$^3$, André van Hoorn$^1$

$^1$University of Hamburg, Hamburg, Germany
$^2$University of Stuttgart, Stuttgart, Germany
$^3$University of Würzburg, Würzburg, Germany

{sebastian.frank, andre.van.hoorn}@uni-hamburg.de, mir-alireza.hakamian@iste.uni-stuttgart.de,
martin.straesser@uni-wuerzburg.de

- State-of-the-Art microservice simulator
- Based on discrete event simulation, uses CPU performance model
- Focus on resilience mechanisms
Recap: MiSim-Orchestration

- Extend MiSim by simulating a Kubernetes environment
- Let MiSim communicate with real Kubernetes components, pretending to be in an actual cluster
- … to enable realistic behavior
Recap: MiSim and MiSim-Orchestration

Microservice architecture and load profile definitions

MiSim

MiSim-Orchestration

Kubernetes Adapter

Kube-scheduler

Cluster-autoscaler

Statistics

configure

output

output

configure

communicate

communicate

communicate

communicate
Simulation-driven Optimization

Key concept: Run simulation iteratively in an optimization loop
Towards Simulation-Driven Optimization of Container Orchestration Mechanisms

Timo Dittus, Martin Sträßer, Samuel Kounev
Towards Simulation-Driven Optimization of Container Orchestration Mechanisms

Timo Dittus, Martin Sträßer, Samuel Kounev
Characterizing the Optimization Problem

- Objectives can be opposing to each other - e.g., costs vs. service quality
- Derivative-free optimization a.k.a. black-box optimization
- Unfeasible to try all configurations
  - Suppose we have 15 configuration parameters, each restricted to 5 possible values and an average simulation runtime of 10s (very optimistic!)
  - Can test 0.0102% of all possible combinations in a full year of running 24/7
  - Improvements in simulation time possible, however, unlikely to improve by many magnitudes
- Therefore, a highly efficient and systematic optimization method is required
## Potential Optimization Methods

<table>
<thead>
<tr>
<th>Description</th>
<th>Probablistic Model-based</th>
<th>Evolutionary Algorithms</th>
<th>Bandit-based Methods</th>
<th>Combined Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Predict good solutions with learned probabilistic model</td>
<td>Mimic mechanisms from natural selection to optimize solutions iteratively</td>
<td>Dynamically allocate resources, apply early stops to unpromising trials</td>
<td>Combine strengths of two or more techniques</td>
</tr>
<tr>
<td>Strength / Weakness</td>
<td>Can be bad for very high-dimensional search space</td>
<td>Great for very high-dimensional search space</td>
<td>Highly efficient, but may lead to inaccurate optimization</td>
<td>May also transfer weaknesses of techniques</td>
</tr>
<tr>
<td>Examples</td>
<td>Bayesian Optimization, Tree-structured Parzen Estimators</td>
<td>Genetic Algorithm, Differential Evolution</td>
<td>Successive Halving, Hyperband</td>
<td>DEHB: Differential Evolution w/ Hyperband, BOHB: Bayesian Optimization w/ Hyperband</td>
</tr>
</tbody>
</table>
Evaluation

- Ideally, real and fully known microservice architecture to replicate in simulation with recorded traces
- ... then deploy in real cloud to measure unoptimized vs. optimized metrics
- ... and see if they improve in similar magnitude to the simulated unoptimized vs. optimized metrics

Analysis and Interpretability

- Automatically find correlations
- ...maybe by simplifying with some constraints (e.g., binning values)
- Visualize results with techniques from hyperparameter tuning

<table>
<thead>
<tr>
<th>HPA: minReplicas</th>
<th>HPA: maxReplicas</th>
<th>Node affinity: required label for &quot;clock-speed&quot;</th>
<th>Node affinity: preferred label for &quot;location&quot;</th>
<th>SLO violations</th>
<th>Operating costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>9</td>
<td>high</td>
<td>eu-west</td>
<td>0-20</td>
<td>233</td>
</tr>
<tr>
<td>5</td>
<td>12</td>
<td>high</td>
<td>eu-west</td>
<td>0-20</td>
<td>400</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>high</td>
<td>eu-west</td>
<td>0-20</td>
<td>418</td>
</tr>
<tr>
<td>9</td>
<td>24</td>
<td>medium</td>
<td>eu-west</td>
<td>0-20</td>
<td>190</td>
</tr>
<tr>
<td>5</td>
<td>12</td>
<td>high</td>
<td>eu-west</td>
<td>0-20</td>
<td>392</td>
</tr>
<tr>
<td>5</td>
<td>16</td>
<td>high</td>
<td>eu-east</td>
<td>21-40</td>
<td>456</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>high</td>
<td>eu-east</td>
<td>41-60</td>
<td>444</td>
</tr>
</tbody>
</table>
Summary

Problem  Configuration of CO environments is complex, testing is time-consuming and costly

Idea  Automate process of selecting and testing configurations, leverage simulation

Benefit  Effortlessly and efficiently find good configurations

Action  Build optimizer feature for simulation, perform lots of testing and analysis
Thank you for listening!

Any questions?