Development and evaluation of a Kubernetes cluster simulator based on Batsim

Presented by: Théo Larue
Supervised by: Olivier Richard & Michael Mercier

Université Grenoble Alpes

August 31, 2020

Table of contents

Computer infrastructures

Distributed systems, many domains: Grid, Edge, HPC, Cloud, P2P, Volunteer, Cluster.
These systems are very complex. (some graph to illustrate the increase in size of supercomputers)

Studying distributed systems

Why studying these infra? To test a system performances under varying loads, applications, scheduling policies, system size and topology. Or to develop new RJMS or research new scheduling algorithms.

Studying distributed systems

How to study these infra? Too many elements and interactions to consider, so no theoretical study.
Real experiments are too costly (both in time and resources) and not reproducible.

Studying distributed systems

Emulation resolves the issue of reproducibility.
Simulation resolves both reproducibility and scalability issues.

Domain specific simulators

refs on domain specific simulators, summed up in a table. Explain briefly the concept behind some of them.

Software specific simulators

YARNSim, SLURM simulator

Publication specific simulators

"Publish and perish" - Milian Poquet

SimGrid

SimGrid: Grid simulator (so general purpose). Versatile, scalable, accurate.
Cpu = a computation speed.
Disk = a capacity, a latency, a service rate.
Network = a flow model.
Simple models but thoroughly validated.

Batsim

Aimed at studying RJMS.
Strong decoupling decision process / simulator.

Batsim - related work

Alea: modular, extensible.
Accasim: supports additional information (temperature, power consuption). Very efficient in terms of simulation time and memory usage.
Both outperform Batsim in terms of scalability. However it is not fair to compare them on this point because Batsim relies on well thought models, when these two only implement delay jobs. Also Batsim has the decoupling with the decision process.

Kubernetes

Explain containers real quick.
Container orchestration software, description based.

Kubernetes cluster simulation

k8s-cluster-simulator: open source, student project, delay jobs. Schedulers provided via a Go interface.

joySim: closed-source, fully fledged kubernetes cluster simulator, service oriented (mock nodes).

Batsim concepts

[schema échanges batsim - scheduler]
Batsim events and protocol.
User defined workloads.

Kubernetes concepts

[schema architecture kube]
Cluster, Nodes, Pods.
Present the api-server, and how everything revolves around it.
Several schedulers can be used at once.

Batkube integration with Kubernetes

[schema custom api]

Time interception

[schema lib go custom]

batsky-go

Explication fonctionnement batsky-go. (diagramme séquence)

Time synchronization

Schéma lignes de temps.

Studied workloads and platforms

Minimum delay

TODO for future work: study min delay effect on makespan and mwt

Timeout

Maximum simulation timestep

the max timestep onyl experiment did not how much. Need to couple it with a backoff multiplier experiment also.

Experimentation on a real cluster

[schema cluster emulé]

Deviation with reality

[meme table que dans le raport] Reprendre les analyses du rapport.

Capabilities of Batkube

- delay jobs
- cpu and memory requests
- can patch any kubernetes scheduler written in Go
- the api only supports the default scheduler

Limitations

- memory hungry (in fact, the scheduler is memory hungry)
- some problems with the scheduler
- not scalable

Perspectives for future work

- parallel jobs
- storage
- more complete api: support for more schedulers but also tools (monitoring tools)

References I