Tell Me When You Are Idle and What May Wake You Up !!!

Djob Mvondo¹, Antonio Barbalace¹, Alain Tchana², Gilles Muller³

University of Edinburgh¹, ENS Lyon², Inria³

04 November 2021
Serverless cloud model is gaining a lot of traction.

~ 22 Billion $ estimated by 2025\(^1\)

Amazon Lambda  Google Functions  Azure Functions

...
Developers send the code and configures the events/trigger.

Event sources -> Gateway -> microVMs/containers

Billed based on execution time and memory used.

Focus on your code and leave the rest to the provider.
The cloud scheduler will direct the request to a server to launch the isolation units.

OS scheduler must ensure fair sharing of **CPU time** for every isolation unit.
PROBLEM: WASTED CPU TIME ON IDLE ISOLATION UNITS

However, some isolation units may be idle.

- Keep alive policy to reduce function start up time.

Diagram:
- microVMs/containers
  - Waiting
  - New func
- OS
- Serverless framework
- Hardware
- NIC
PROBLEM: WASTED CPU TIME ON IDLE ISOLATION UNITS

However, some isolation units may be idle.

- Keep alive policy to reduce function start up time.
- Functions awaiting inputs from other functions.
PROBLEM: WASTED CPU TIME ON IDLE ISOLATION UNITS

For a pipeline of 3 functions, \textbf{Func\{1,2,3\}. Func\{2,3\} isolation units are initialized but awaits func1 completion}

\textbf{Figure 1. Illustration of micro-VMs idle times. Micro-VMs b and c running Func2 and Func3 respectively, are scheduled even though they await Func1 output which has not finished running. This results in wasted CPU time.}
PROBLEM: WASTED CPU TIME ON IDLE ISOLATION UNITS

We analyzed the wasted CPU time on idle isolation units.

- In-lab setup and ec2 a1.metal with Firecracker
- Triggering up to 50 pipelines image processing functions
- Inputs and outputs images stored in AWS S3
- We compute isolation units idle CPU usage

PROBLEM: WASTED CPU TIME ON IDLE ISOLATION UNITS

Avg Pipeline execution time
- 28.3s to 83.41s --- inlab
- 20s to 78.52s --- aws a1

Idle time ratio
- 20.18% to 75.31% --- inlab
- 16.25% to 69% -- aws a1

CPU time is wasted, smarter use could improve overall execution time
Possible ideas: Scheduling semantic gap

Well known problem in the context of virtualization.

The host scheduler should:

- Understand when an isolation unit is idle
- Understand the events that will affect idle isolation units

How do you detect?
- What's the penalty of a false positive?
- Intrusiveness?

Monitor events
Dynamically update scheduling policy?
Possible ideas: Scheduling semantic gap

Approaches worth exploring

Understand when an isolation unit is idle

- How do you detect?
- What's the penalty of a false positive?
- Intrusiveness?

- Trusted source
- Collaborative
- Learning
Possible ideas: Scheduling semantic gap

Approaches worth exploring

Understand the events that will affect idle isolation units

Monitor events

Dynamically update scheduling policy?

eBPF³ (Extended Berkeley Packet Filter)

Conclusion

To provide more packing with FaaS, scheduling needs to be improved.

Understand when an isolation unit is idle

Monitor the events that will affect idle isolation units and update

- Trusted source
- Collaborative
- Learning

- eBPF (Extended Berkeley Packet Filter)
Thanks

Questions ?