



SimLess: Simulate Serverless Workflows and Their Twins and Siblings in Federated FaaS

Sashko Ristov, Mika Hautz, Christian Hollaus (UIBK, Austria)
Radu Prodan (UNI-KLU, Austria)

Agenda

- Federated FaaS challenges
- Motivation and Preliminary Observations
- *SimLess* FC simulation model
- *SimLess* parameter setup
- *SimLess* Evaluation
- Conclusion and future work

FaaS Heterogeneity



Alibaba



Google



AWS



IBM



Free at last?

We can compute everywhere!








But, Are We Really Free?



Hurdles in all directions

Federated FaaS Challenges

Such a **heterogeneous** environment. How to:

- **Deploy?** 
- **Model?** 
- **Orchestrate?** 
- **Optimize?** 
- **Run?** 

Federated FaaS Challenges

Such a **heterogeneous** environment. How to:

- **Deploy?**  Ristov et al. 22 (**GoDeploy** - IEEE Cloud Summit BEST PAPER)
- **Model?** 
- **Orchestrate?** 
- **Optimize?** 
- **Run?** 

Federated FaaS Challenges

Such a **heterogeneous** environment. How to:

- **Deploy?**  Ristov et al. 22 (**GoDeploy** - IEEE Cloud Summit BEST PAPER)
- **Model?** 
- **Orchestrate?**  Ristov et al. 2021 (**AFCL** - Elsev. FGCS)
- **Optimize?** 
- **Run?** 

Federated FaaS Challenges

Such a **heterogeneous** environment. How to:

- **Deploy?**  Ristov et al. 22 (**GoDeploy** - IEEE Cloud Summit BEST PAPER)
- **Model?** 
- **Orchestrate?**  Ristov et al. 2021 (**AFCL** - Elsev. FGCS)
- **Optimize?**  Ristov & Gritsch. 2022 (**FaaSSt** - IEEE Cluster '22)
- **Run?** 

Federated FaaS Challenges

Such a **heterogeneous** environment. How to:

- **Deploy?**  Ristov et al. 22 (**GoDeploy** - IEEE Cloud Summit BEST PAPER)
- **Model?** 
- **Orchestrate?**  Ristov et al. 2021 (**AFCL** - Elsev. FGCS)
- **Optimize?**  Ristov & Gritsch. 2022 (**FaaSSt** - IEEE Cluster '22)
- **Run?**  Ristov et al. 2021 (**xAFCL** - IEEE TSC),

Federated FaaS Challenges

Such a **heterogeneous** environment. How to:

- **Deploy?**  Ristov et al. 22 (**GoDeploy** - IEEE Cloud Summit BEST PAPER)
- **Model?** 
- **Orchestrate?**  Ristov et al. 2021 (**AFCL** - Elsev. FGCS)
- **Optimize?**  Ristov & Gritsch. 2022 (**FaaSSt** - IEEE Cluster '22)
- **Run?**  Ristov et al. 2021 (**xAFCL** - IEEE TSC), 2022 (**rAFCL** - IEEE TNSM)

Federated FaaS Challenges

Such a **heterogeneous** environment. How to:

- **Deploy?**  Ristov et al. 22 (**GoDeploy** - IEEE Cloud Summit BEST PAPER)
- **Model?**  *SimLess*
- **Orchestrate?**  Ristov et al. 2021 (**AFCL** - Elsev. FGCS)
- **Optimize?**  Ristov & Gritsch. 2022 (**FaaSSt** - IEEE Cluster '22)
- **Run?**  Ristov et al. 2021 (**xAFCL** - IEEE TSC), 2022 (**rAFCL** - IEEE TNSM)

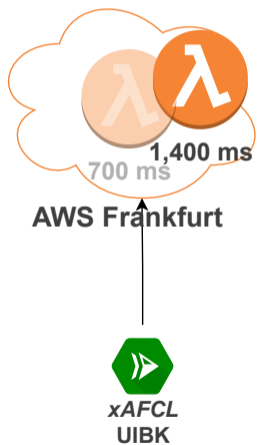
Agenda

- Federated FaaS challenges
- Motivation and Preliminary Observations
- *SimLess* FC simulation model
- *SimLess* parameter setup
- *SimLess* evaluation
- Conclusion and future work

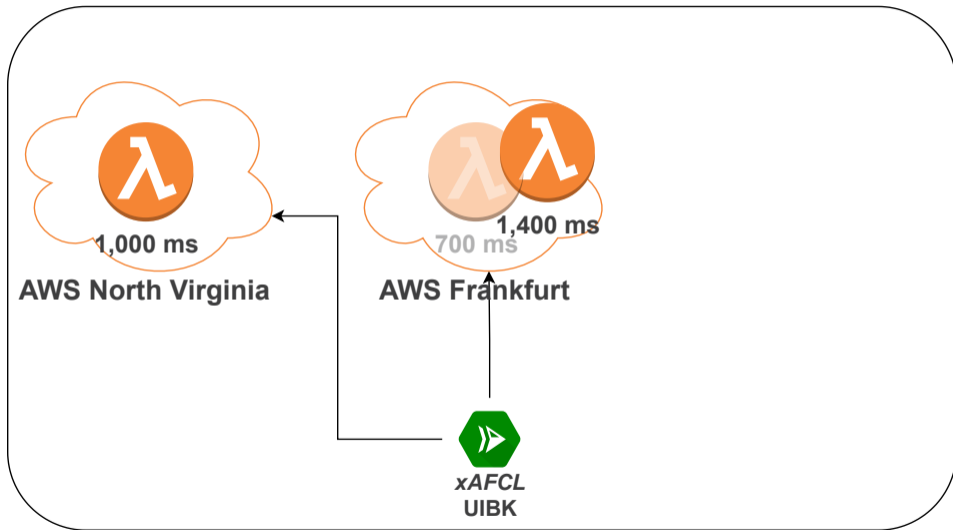
Motivation



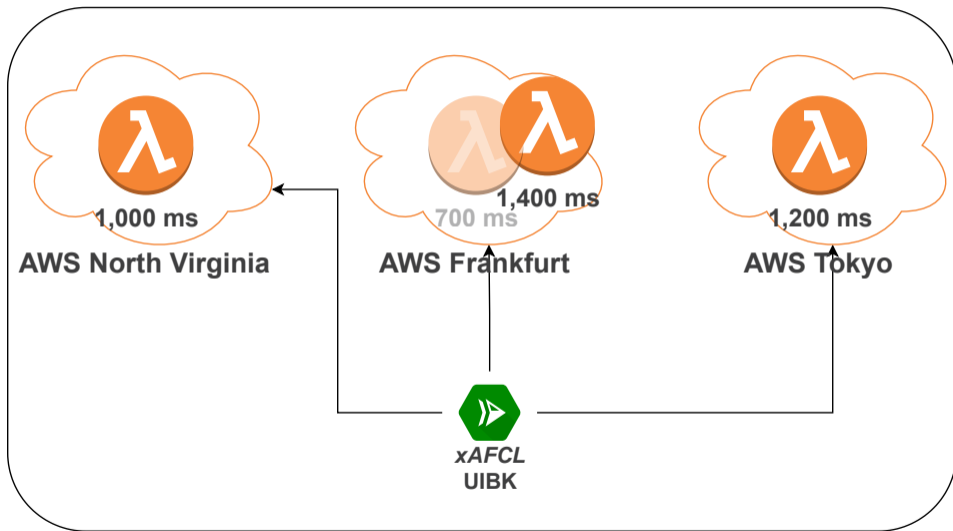
Motivation



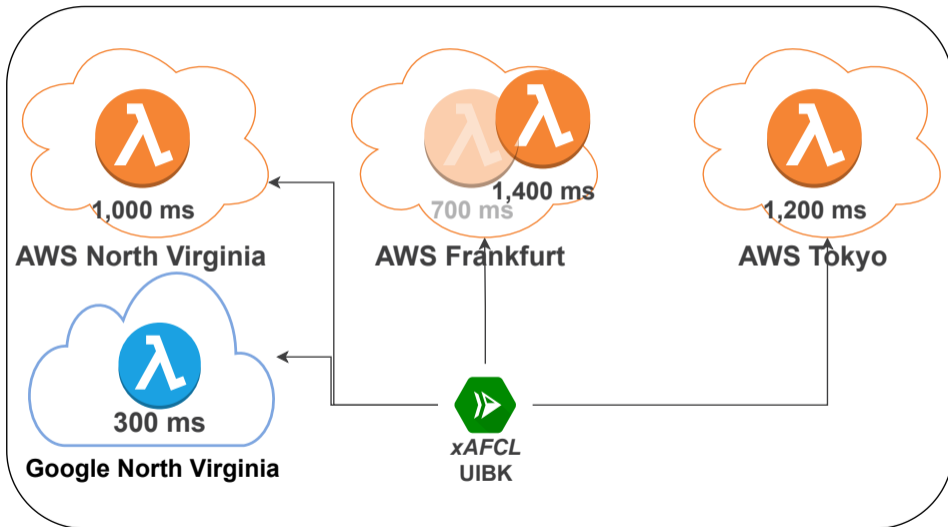
Motivation



Motivation



Motivation



Research question - Simulation and Modeling

Research question:

- Which **parameters** of federated clouds should be formalized to estimate the performance of serverless functions, with **minimum effort** to learn them?

Research question - Simulation and Modeling

Research question:

- Which **parameters** of federated clouds should be formalized to estimate the performance of serverless functions, with **minimum effort** to learn them?
- Can we simulate AWS North Virginia and AWS Tokyo **without running** them?

Research question - Simulation and Modeling

Research question:

- Which **parameters** of federated clouds should be formalized to estimate the performance of serverless functions, with **minimum effort** to learn them?
- Can we simulate AWS North Virginia and AWS Tokyo **without running** them?

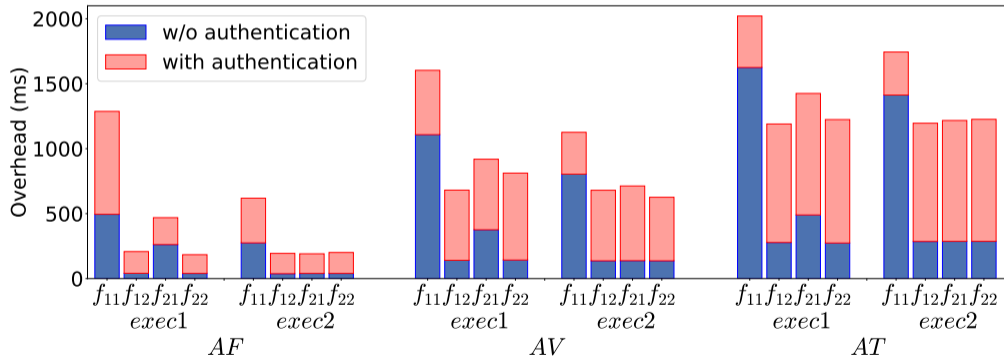
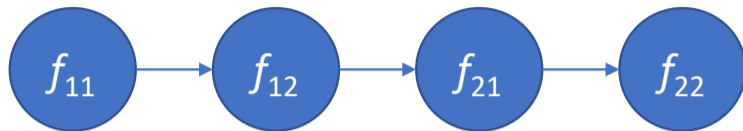
Research goal: Reuse parameters

- Can we correlate cloud regions and reuse value 700 ms of AWS Frankfurt and derive
 - 1,200 ms for AWS North Virginia and
 - 1,400 ms for AWS Tokyo?

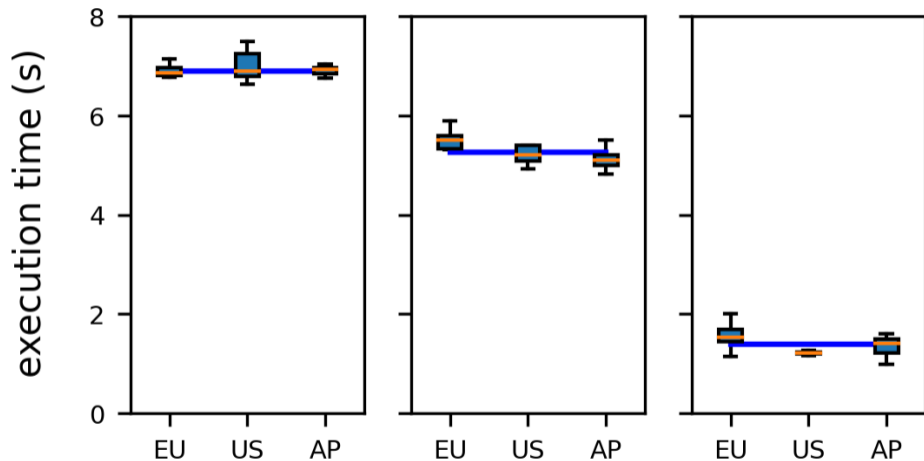
Preliminary Observations 01 - **Regions Similarity**

- CloudPing
- similar internal latency within AWS regions (1.25 ms to 5.25 ms, 3.44 ms)
- distributing a function to different regions causes various latency from University of Innsbruck
 - 18 ms to the EU
 - 122 ms to the US, and
 - 190 ms to the Asia Pacific.

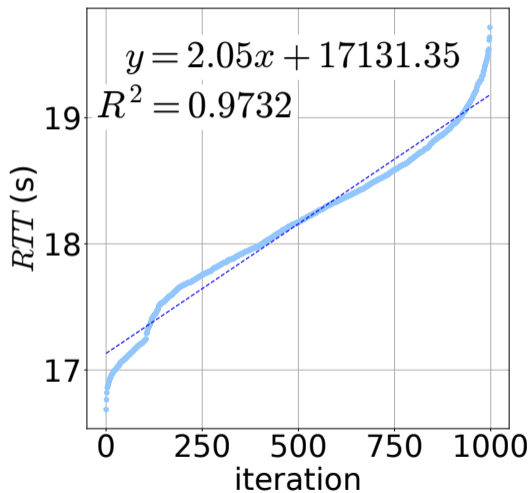
Preliminary Observations 02 - **Overheads** *NO*, *AO*, *SO*



Preliminary Observations 03 - *ET* Similarity (A, G, I)



Preliminary Observations 04 - **Overheads** - CO



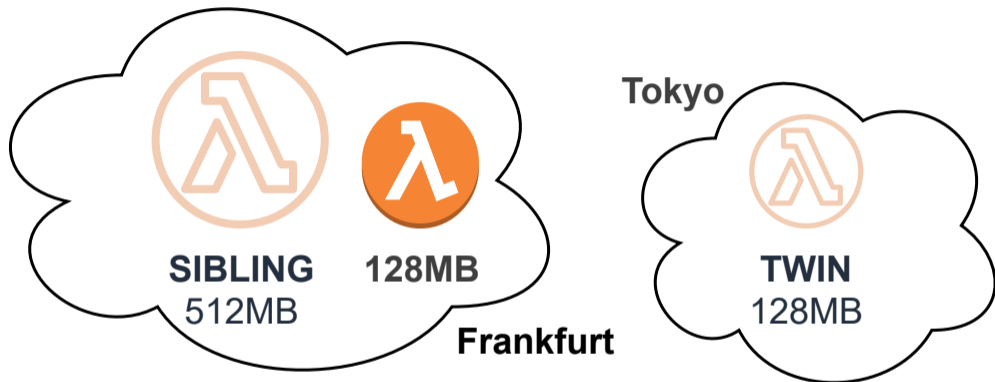
SOTA Limitations

- Focused on *Execution Time ET* rather than total *Round Trip Time RTT*
- Focused on a single function
- Lack of serverless workflow or *function choreography (FC)* simulation models in federated FaaS
- Serverfull workflow simulation model not applicable to FCs

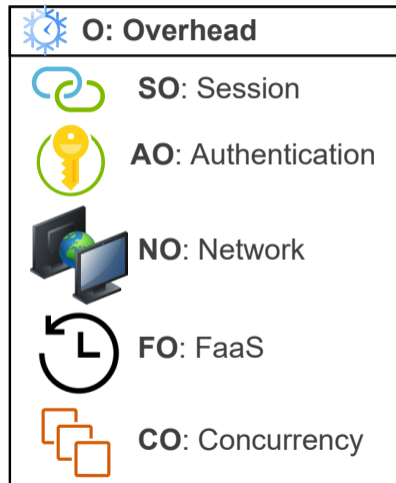
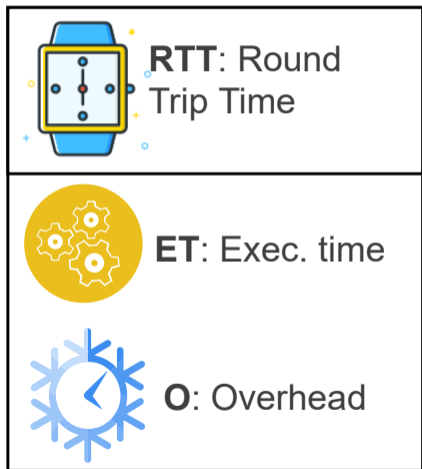
Agenda

- Federated FaaS challenges
- Motivation and Preliminary Observations
- *SimLess* FC simulation model
- *SimLess* parameter setup
- *SimLess* evaluation
- Conclusion and future work

SimLess Method: **Twins** and **Siblings**



SimLess Innovative Approach: $RTT = ET + O$



SimLess FC Simulation Model

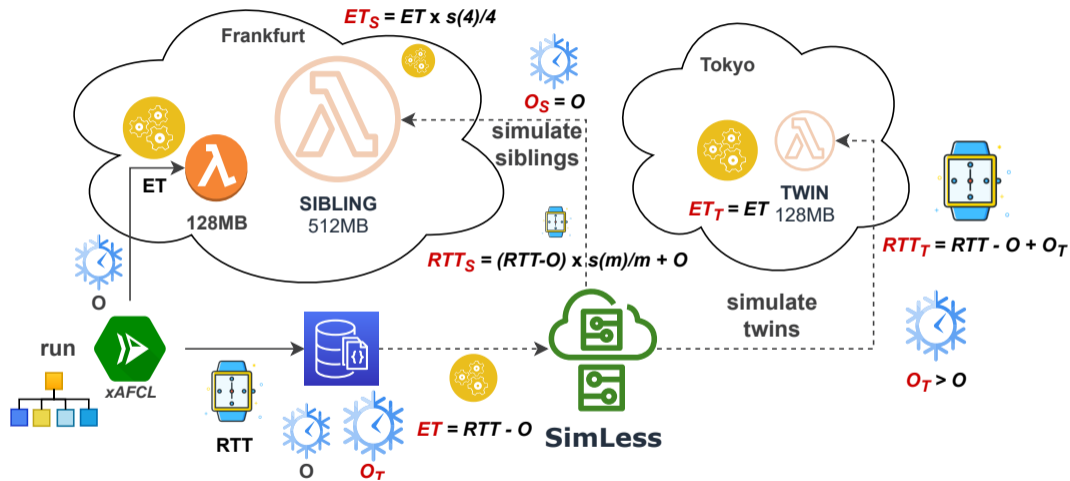
$$\overline{RTT} = \overline{O} + \overline{ET} \quad (1)$$

$$\overline{O} = \overline{SO} + \overline{NO} + \overline{AO} + \overline{FO} + \overline{CO} \quad (2)$$

$$\overline{AO} = \begin{cases} \overline{cr} + 3 \cdot \overline{NO}, & \text{2-way authentication;} \\ \overline{cr} + 2 \cdot \overline{NO}, & \text{1-way authentication.} \end{cases} \quad (3)$$

$$\overline{CO} = (k - 1) \cdot \overline{d} \quad (4)$$

SimLess FC Simulation Model: Twins and Siblings

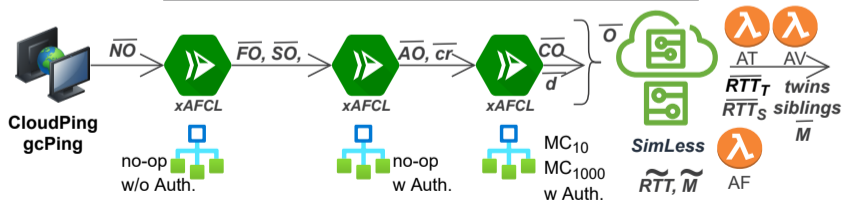


Agenda

- Federated FaaS challenges
- Motivation and Preliminary Observations
- *SimLess* FC simulation model
- *SimLess* parameter setup
- *SimLess* evaluation
- Conclusion and future (past) work

Cloud Regions for Learning and Validation

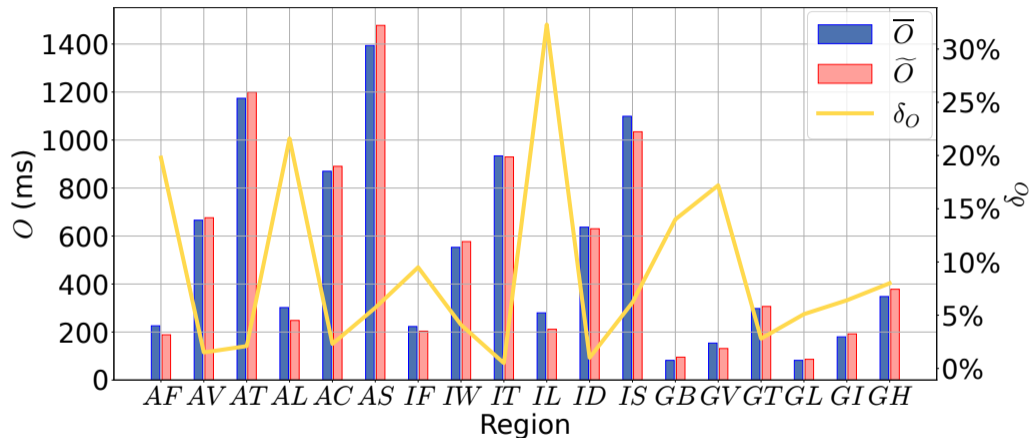
	<i>Goal</i>	<i>Cont.</i>	<i>AWS</i>	<i>IBM</i>	<i>Google</i>
<i>Learn</i>		EU ₁	AF	IF	GB
		US ₁	AV	IW	GV
		AP ₁	AT	IT	GT
<i>Validate</i>		EU ₂	AL	IL	GL
		US ₂	AC	ID	GI
		AP ₂	AS	IS	GH



SimLess Parameter Setup

Region	\overline{SO}	\overline{NO}	\overline{cr}	\overline{AO}	\overline{FO}	\overline{O}	\tilde{O}	$\delta_o[\%]$
AF		30		166		226	189	19.6
AV	550	140	76	496	30	666	676	1.48
AT		267		877		1174	1200	2.17
IF		30		136		223	204	9.31
IW	152	140	76	356	57	553	577	4.16
IT		267		610		934	930	0.43
GB		59				82	95	13.7
GV	112	131	-	-	23	154	131	17.6
GT		275				298	307	2.93

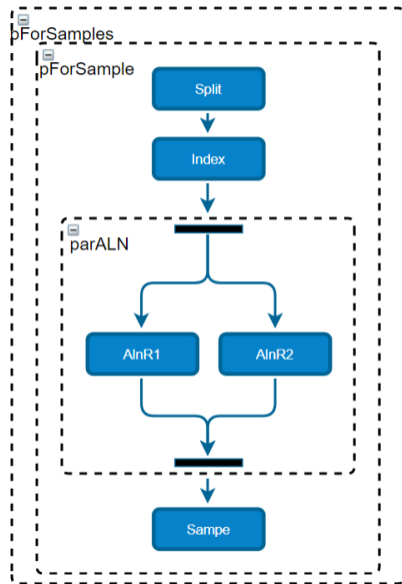
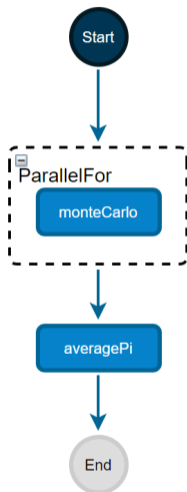
SimLess Overhead Accuracy



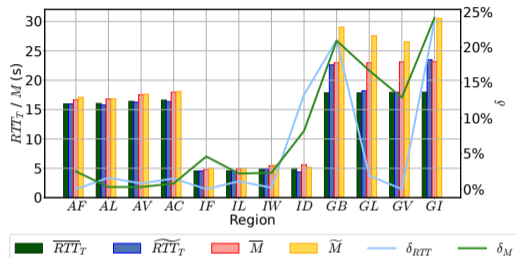
Agenda

- Federated FaaS challenges
- Motivation and Preliminary Observations
- *SimLess* FC simulation model
- *SimLess* parameter setup
- *SimLess* evaluation
- Conclusion and future (past) work

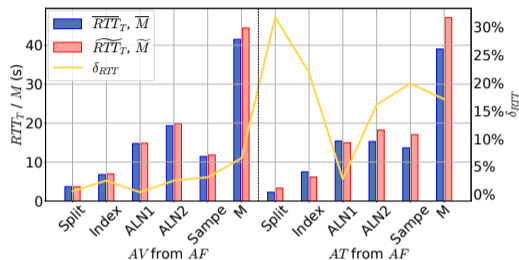
Evaluated Tools



Low Concurrency MC₁₀ and BWA₂₀

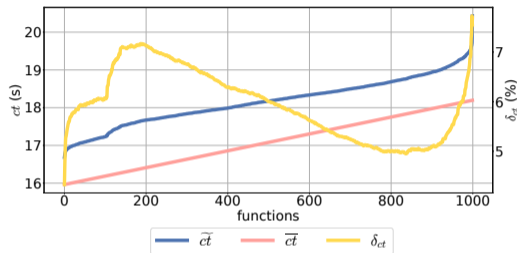


Simulated versus measured RTT and makespan of monteCarlo twins, and their inaccuracy.

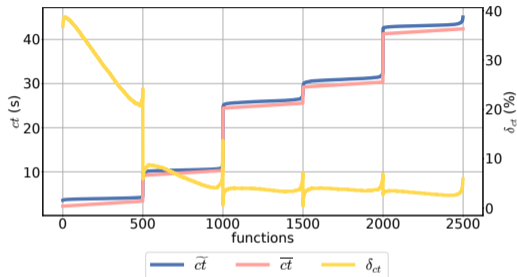


Simulated versus measured RTT of BWA₂₀ twins and their inaccuracy

High Concurrency MC_{1,000} and BWA_{2,500}



Simulated versus measured completion time of MC_{1,000} functions on AF and their inaccuracy.



Simulated vs. measured completion time ct of BWA_{2,500} functions on AF and their inaccuracy.

Agenda

- Federated FaaS challenges
- Motivation and Preliminary Observations
- *SimLess* FC simulation model
- *SimLess* parameter setup
- *SimLess* evaluation
- Conclusion and future (current) work

Conclusion

Validation

Self-configurable parameters defined by the *SimLess* FC simulation model are provider-dependent and FC-agnostic.

Novelty

SimLess is the first FC simulation framework that models important overheads, including *provider-specific* (i.e., \overline{SO} , \overline{FO} , concurrency invocation \overline{d} , cryptography \overline{cr}) and *region-specific* (i.e., \overline{NO} , \overline{AO}).

Evaluation

Running FCs with a low concurrency of 20 functions in a single region is sufficient to simulate a highly concurrent FC with 2,500 functions with 77.23% lower costs.

Further Investigation in 2022

Interoperability - Storage

- `copy(srcURL, destURL)` (Java, Python, Go¹)
- Dynamically select storage during runtime, without redeployment

DeployLess Model and Scheduler

Functions	Size	$\mathcal{A}_{\mathcal{F}}$	$\mathcal{A}_{\mathcal{N}}$	$\mathcal{A}_{\mathcal{T}}$
Split, Index, AlnR1, AlnR2	0.17 MB	1.8 s	2.7 s	4.5 s
Sampe, Merge, Sort	1.7 MB	2.0 s	3.1 s	5.6 s

¹<https://github.com/FaaSTools/GoStorage>

QUESTIONS?



Email: sashko.ristov@uibk.ac.at