Service Engineering Research Area
ICCLAB SPLAB
Zurich University of Applied Sciences (CH)

CONICET
ISISTAN
Instituto Superior de Ingeniería de Software Tandil
ISISTAN Research Institute CONICET - UNICEN (AR)

ITIC
National University of Cuyo (AR)

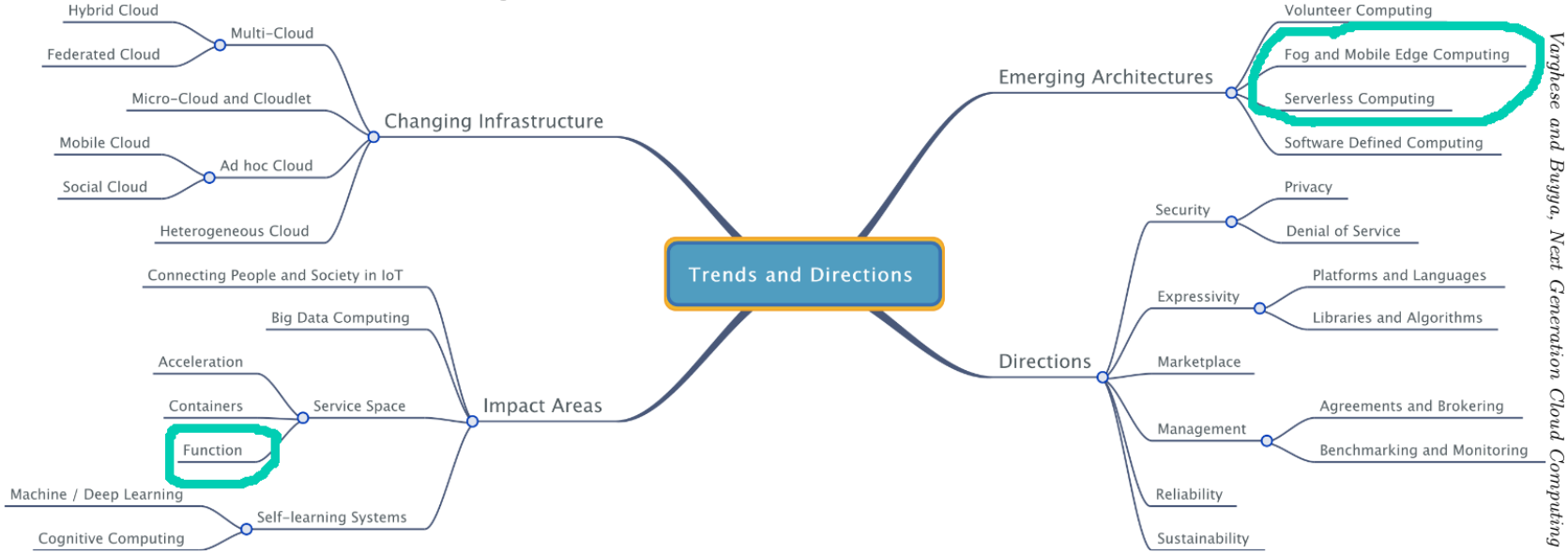
FaaSter, Better, Cheaper: The Prospect of Serverless Scientific Computing and HPC

Josef Spillner, Cristian Mateos, David A. Monge

September 21, 2017 | CARLA ‘17

Background: Meeting of »Cultures«

Serverless computing …



… vs. data centre servers for scientific and high-performance computing

nVidia DGX for healthcare

Sunway TaihuLight
#1 TOP500 Jun'17

Background: Serverless Computing

is a:

- **marketing term**
  - **for Function-as-a-Service ecosystems**
- **programming model**
  - **functions or methods in diverse programming languages**
  - **with specific signatures (parameters, return values)**
  - **sometimes, executable implementations, e.g. containers**
- **deployment model**
  - **upload of source files or compiled binaries**
  - **configuration of entrance handler, memory allocation, etc.**
- **execution model**
  - **time limit, e.g. 5 minutes**
  - **pay-per-use microbilling, e.g. per invocation + 100ms duration**

Specialised Function Instances?

Comparison:
Amazon EC2 (virtual machines) and Lambda (functions)

Amazon EC2

accelerated
memory-optimised
storage-optimised

Model	vCPU	CPU Credits / hour	Mem (GiB)	Storage
t2.nano	1	3	0.5	EBS-Only
t2.micro	1	6	1	EBS-Only
t2.small	1	12	2	EBS-Only
t2.medium	2	24	4	EBS-Only
t2.large	2	36	8	EBS-Only
t2.xlarge	4	54	16	EBS-Only

general-purpose

Model	vCPU	Mem (GiB)	Storage	Dedicated EBS Bandwidth (Mbps)
c4.large	2	3.75	EBS-Only	500
c4.xlarge	4	7.5	EBS-Only	750
c4.2xlarge	8	15	EBS-Only	1,000
c4.4xlarge	16	30	EBS-Only	2,000

compute-optimised

AWS Lambda

Lambda @ Edge

Memory (MB)
128
192
256
320
384
448
512
576

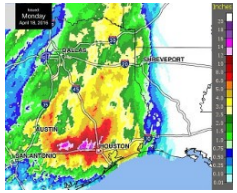
(CPU performance proportional to memory allocation)

The Need for Specialised Instances

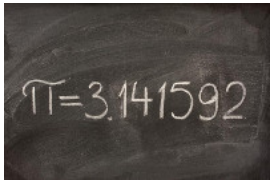
4 scientific computing experiments ...

cryptology:
password cracking

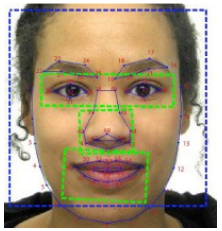
meteorology:
precipit. forecast



mathematics:
π approximation



computer graphics:
face detection



... in a competitive serverless setting: local + cloud execution

AWS Lambda

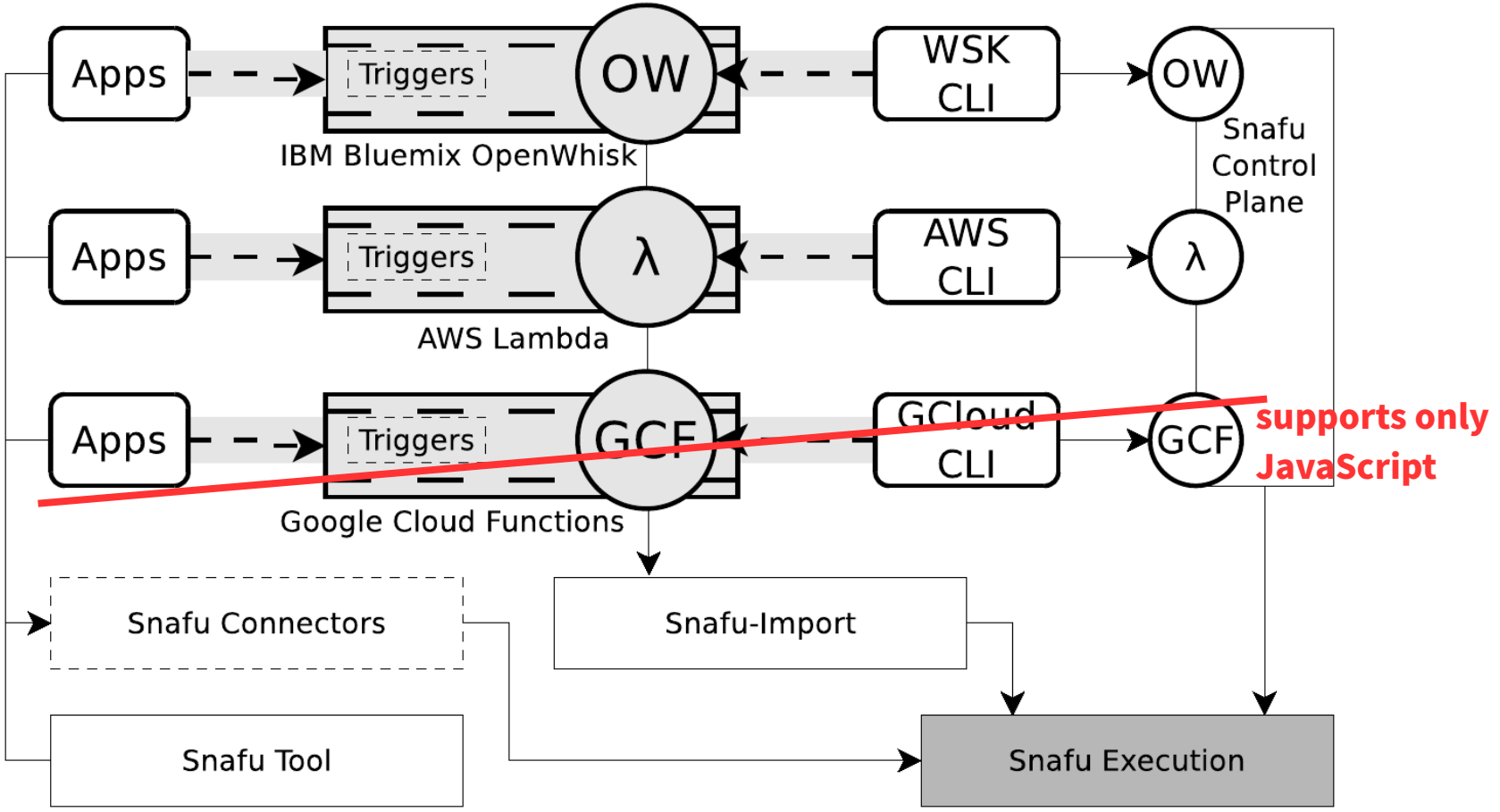
IBM Bluemix OpenWhisk

Azure Functions

Serverless Computing Snafu

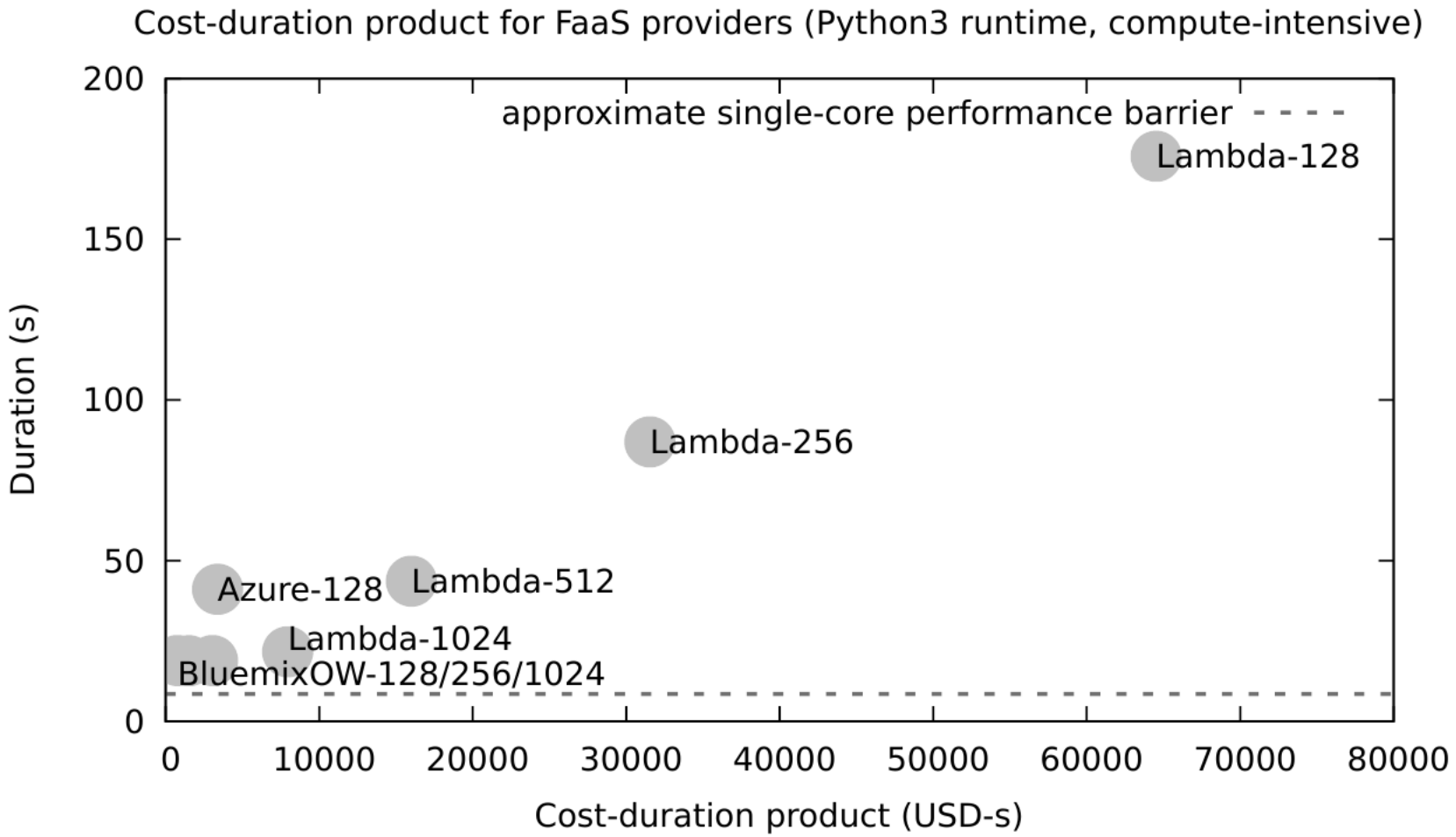
Experiment Setup

Python function generation (by source code decomposition), deployment and execution



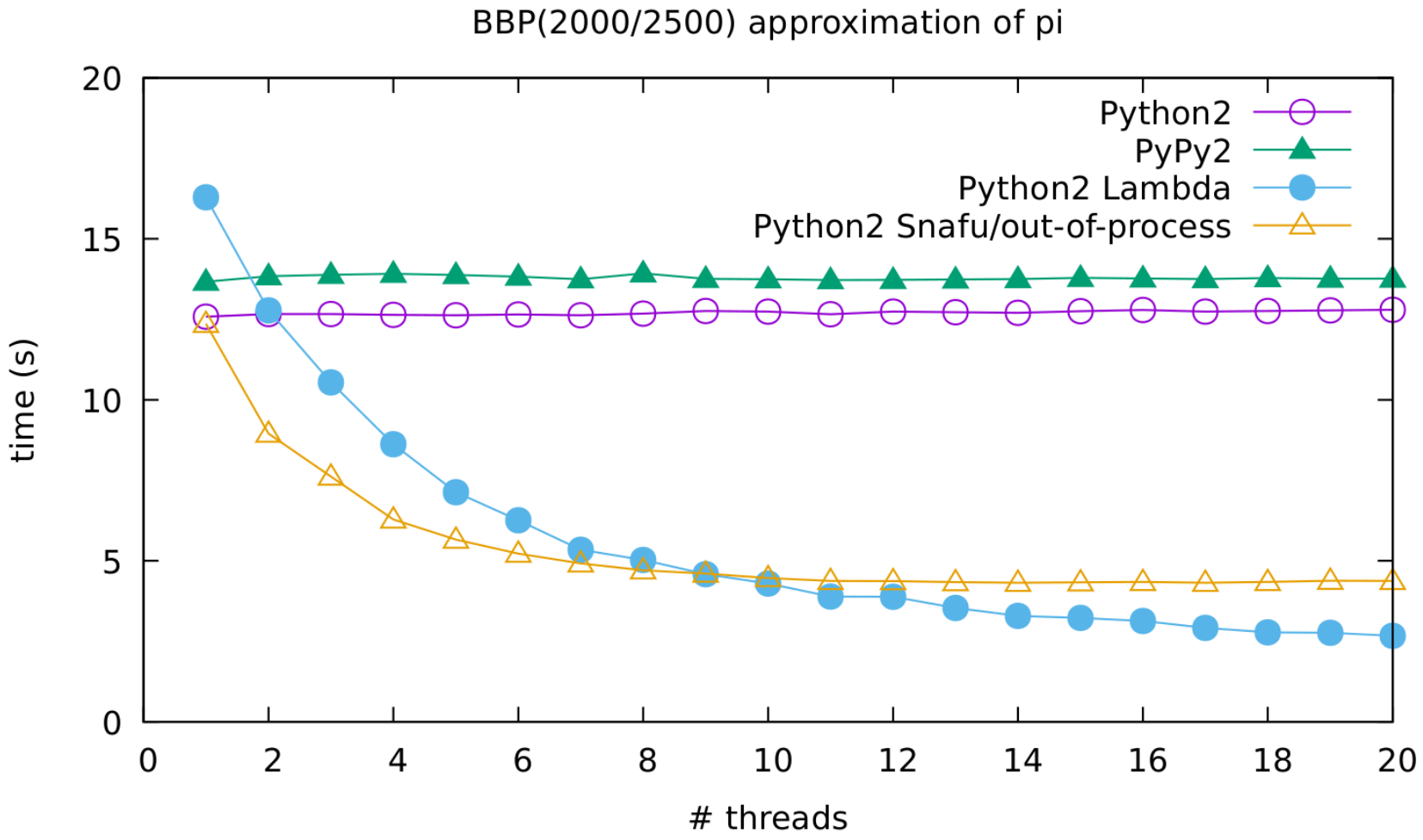
Experiment Results

Cheaper? The Cost-Duration Product (CPD) comparison.



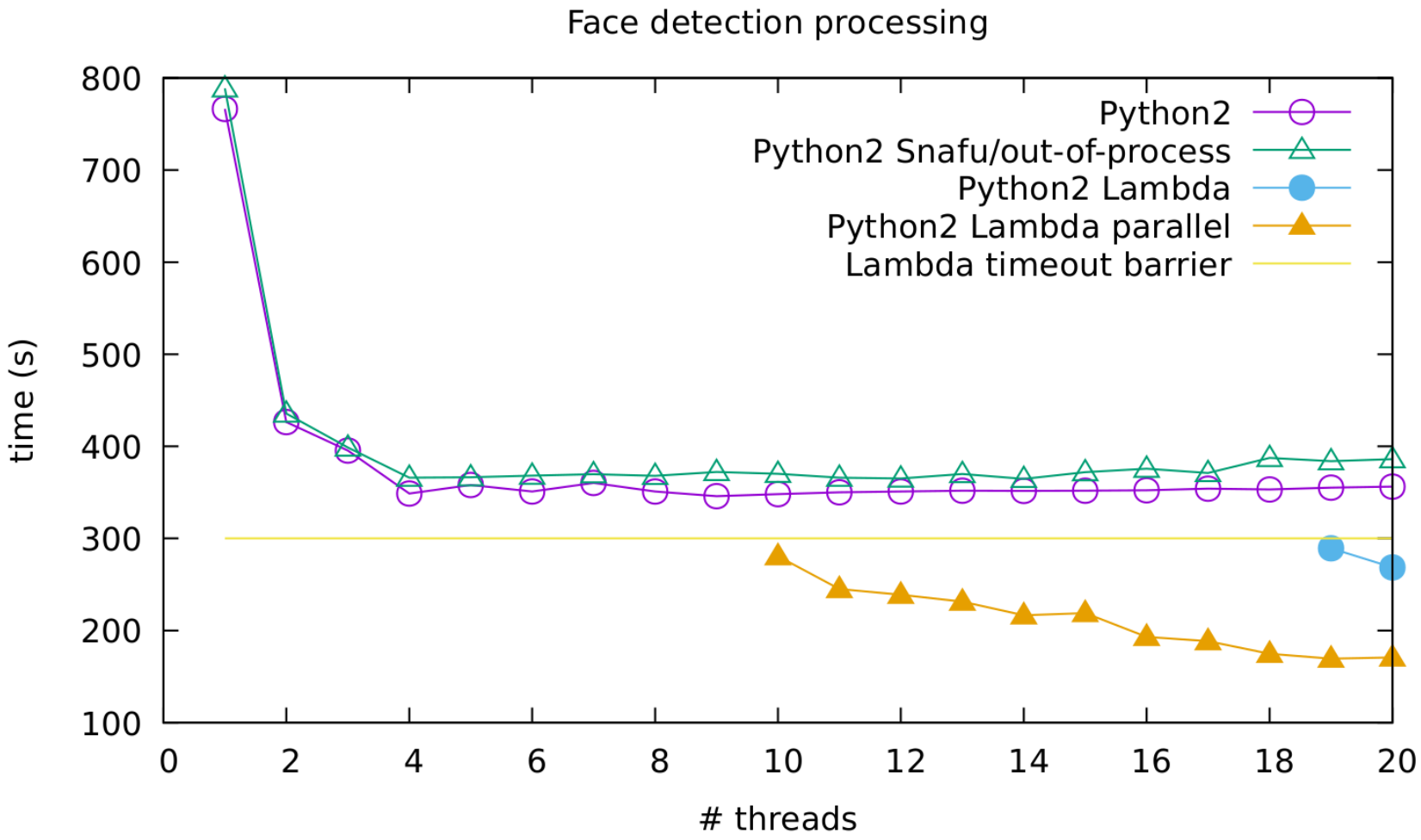
Experiment Results

Faster? The comparison of π approximation.



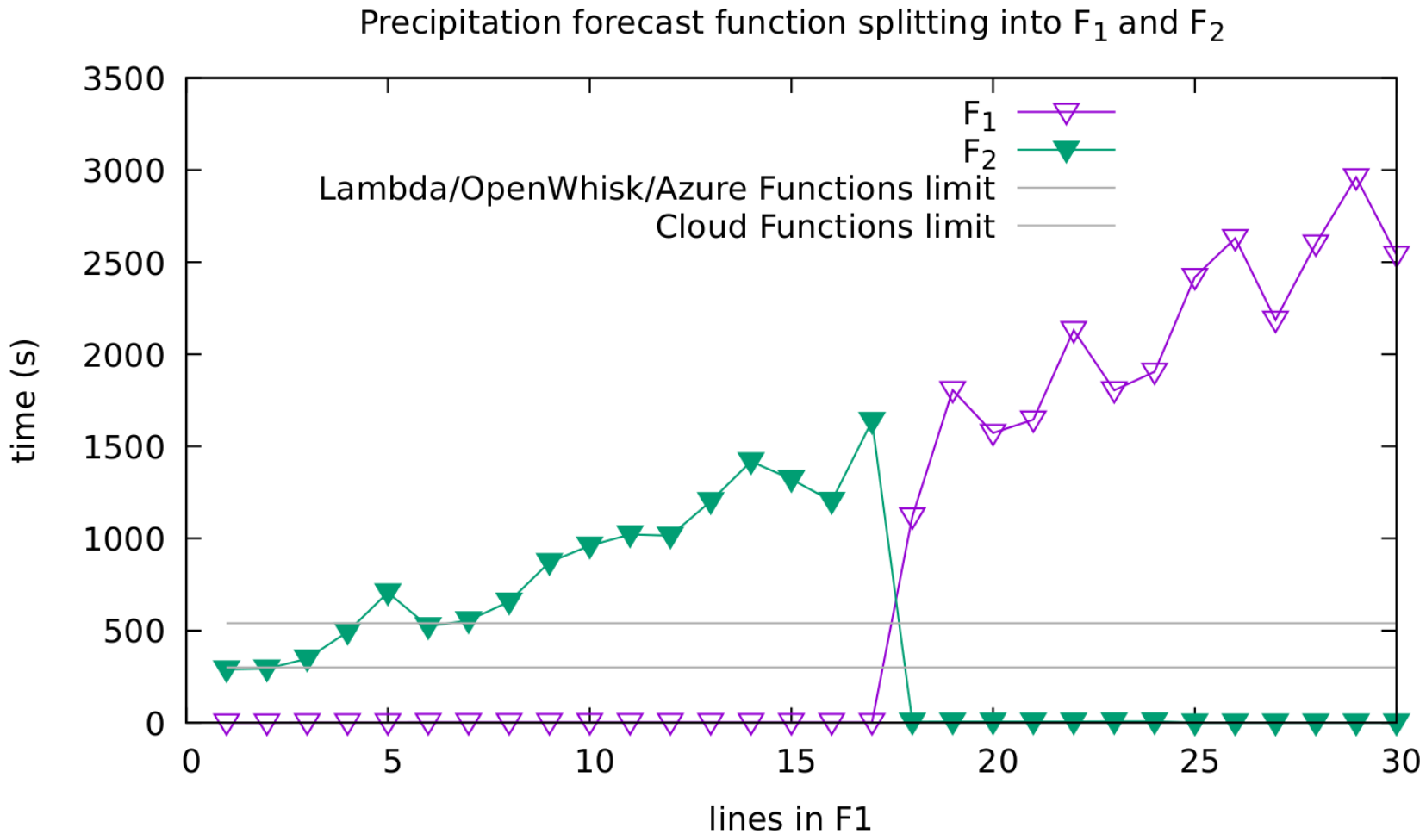
Experiment Results

Faster? Another look, comparison of face detection.



Experimental Results

Better? Towards deep FaaSification to support legacy code.



Conclusion

Is serverless better, cheaper, fa(a)ster?
- **depends, we have shown that it can be**

Contributions:
- **refined FaaSification process**
  - **shallow, medium, deep**
- **code: function futures for Python programmers**
  - **complements multi-threading and multi-processing**
- **concept: worm functions for serverless developers**
  - **works around the execution time limit in public clouds**

Download Snafu:
- **git clone https://github.com/serviceprototypinglab/snafu**
- **pip install snafu**
- **docker run jszhaw/snafu**