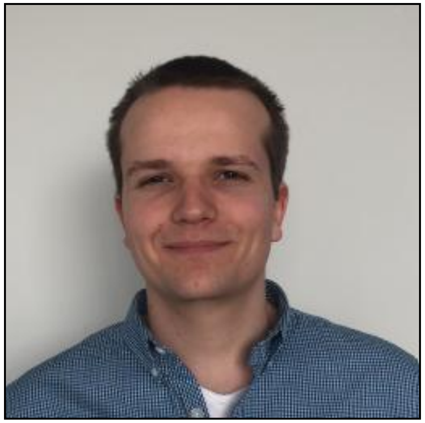
A Trace-Based Performance Study of Autoscaling Workloads of Workflows in Datacenters



Ir. Laurens Versluis
Massivizing Computer Systems
@Large research,
https://atlarge-research.com

Mihai Neacşu, BSc.

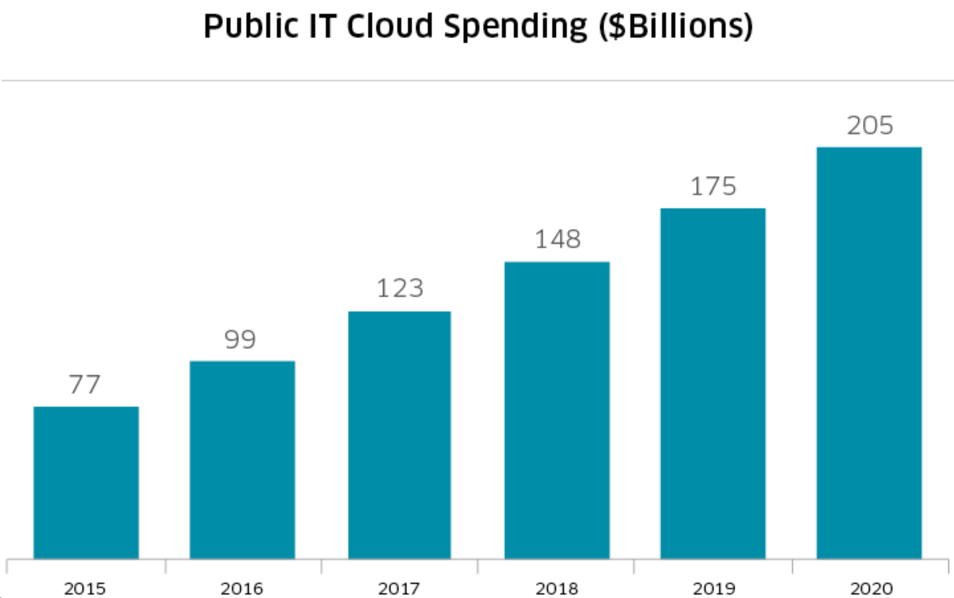
Prof.dr.eng. Alexandru Iosup

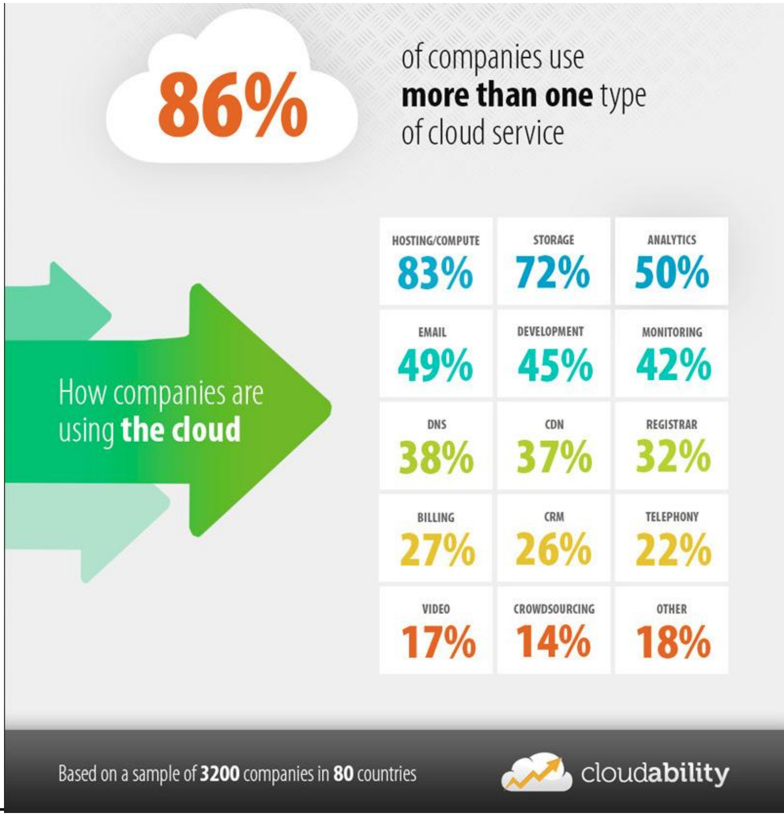
VRIJE UNIVERSITEIT AMSTERDAM

Cloud popularity and usage at all-time high

- Surveys: >$200B market by 2020, 86% of companies use >1 cloud service

- Resource utilization increasingly important
  - Competitive position for companies
  - Reduce costs for customer & provider





VRIJE UNIVERSITEIT AMSTERDAM

Source: http://business.nasdaq.com/marketinsite/2017/Cloud-Computing-Industry-Report-and-Investment-Case.html

Workflow execution is gaining popularity

- Workflows = set of tasks with precedence constraints
  - Usually represented as a Directed Acyclic Graph (DAG)
  - Used to model applications in many domains

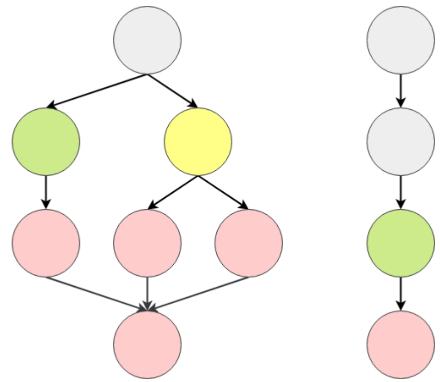
- Today: thousands of applications in use

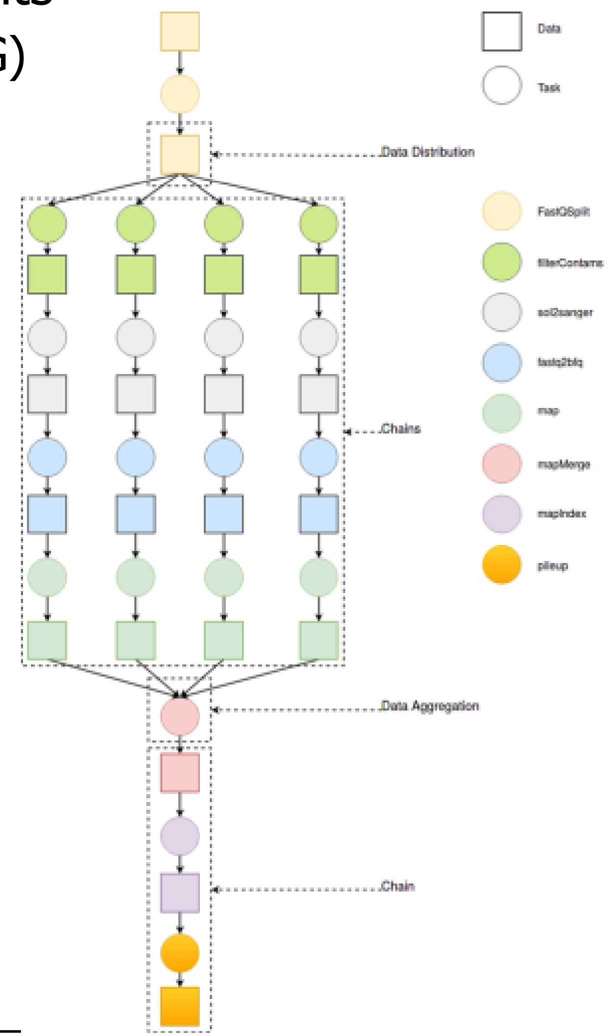
Protein BLAST
protein ▸ protein

finding cancer early
epigenomics

Montage
An Astronomical Image Mosaic Service for the National Virtual Observatory
http://montage.ipac.caltech.edu

Shell



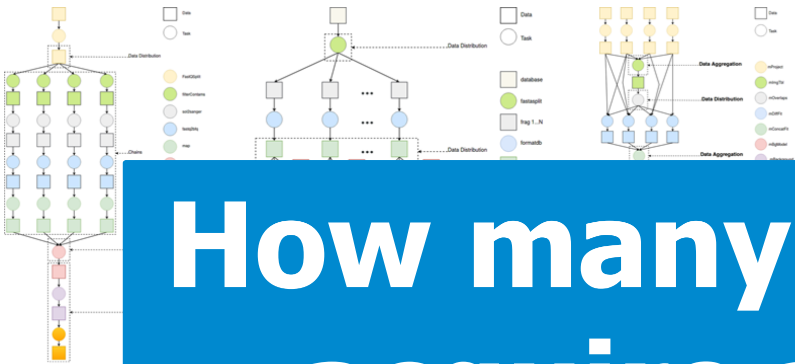


VRIJE UNIVERSITEIT AMSTERDAM

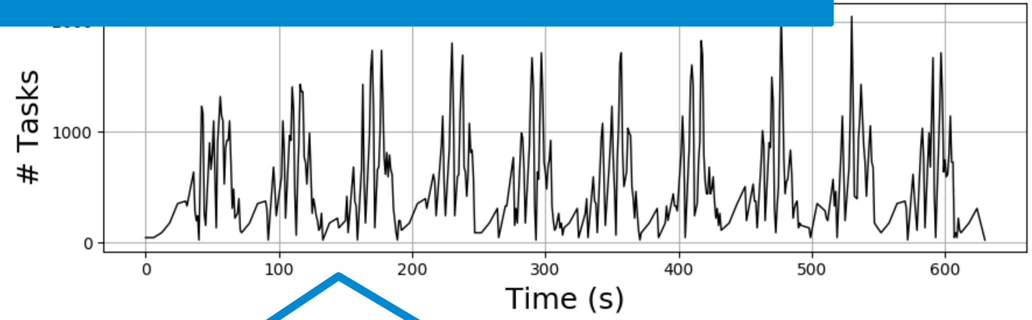
Executing workflows in the cloud

1. Workflows are submitted to the cloud, executed in datacenters

2. Workflow resource demand changes over time due to their complex structures



How many resources to acquire and when?



Workload of workflows

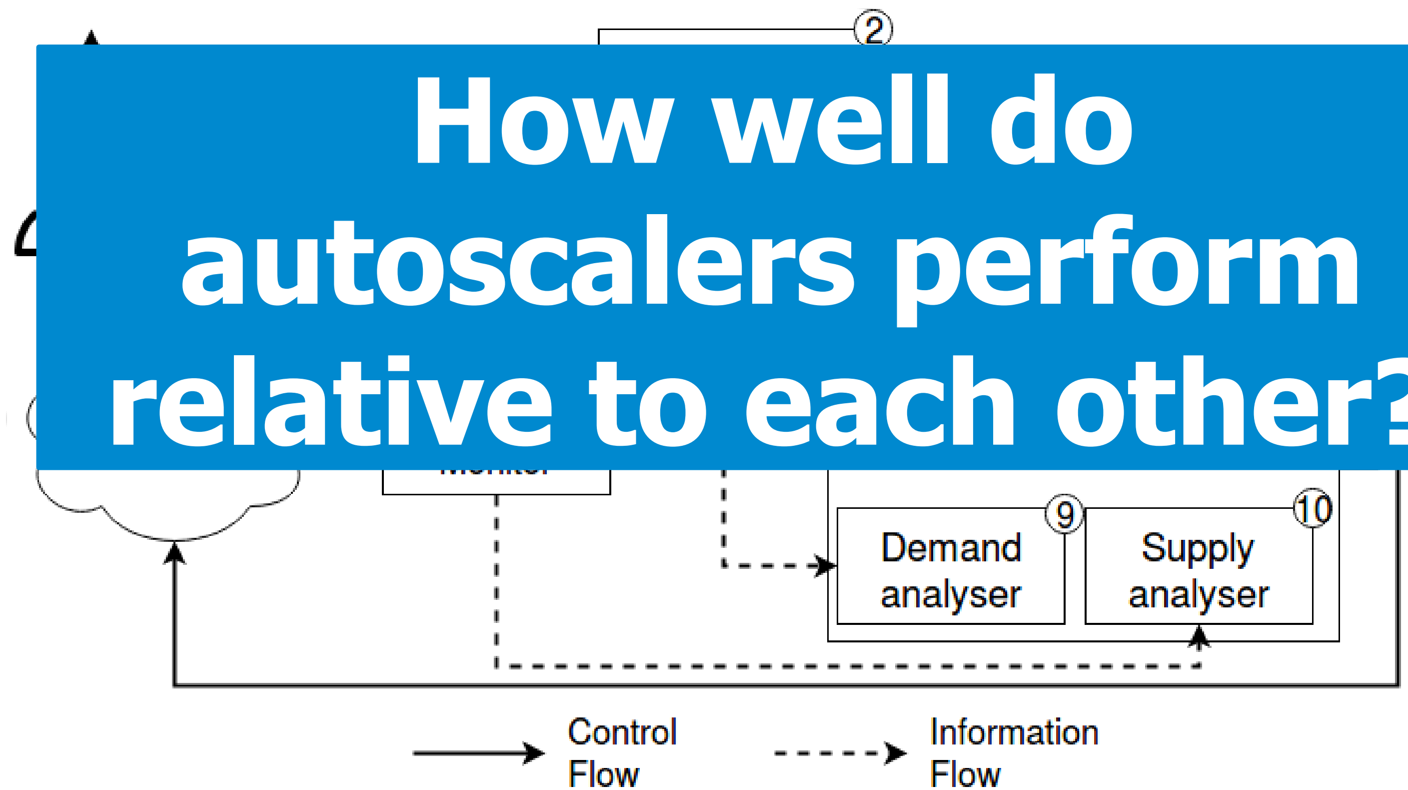
A problem for cloud providers

- Minimize overprovisioning (allocating too many resources)
  - Reduces costs
- Adhere to the Quality-of-Service (QoS) requirements of the client
  - Minimize underprovisioning
- Automate this process
  - Poor user estimates of resource requirements

- Solution: Autoscalers
  - Scale resources on demand
  - Forecast resource demand in the near-future
  - Deal with sudden flash-crowds/peaks

Autoscaler operations

1. Monitor current workload (9) and available resources (10)
2. Attempt to predict future utilization (8)
3. Scale resources up or down according to prediction (4)



How well do autoscalers perform relative to each other?

In this work: Compare autoscalers in simulation

- Four distinct workload traces
  - A **workload** is a set of workflows (applications)
- Use a rich set of metrics
  - 10+ forms of **elasticity**
- Four experiments
  1. Different workload **domains** **(new)**
  2. Bursty workloads (deeper understanding)
  3. Impact of the allocation policy **(new)**
  4. Different resource environments **(new)**

Medium scale

ID	Source	Domain	Workflows	# Tasks
T1	SPEC Cloud Group	Scientific	200	13,876
T2	Chronos	Industrial	1,024	3,072
T3	Askalon EE	Engineering	757	45,786
T4	Askalon EE2	Engineering	3,551	122,105

Experiment: different workload domains

- Investigate the performance of autoscalers while processing workloads of different domains

- Three workloads:
  - Scientific (T1)
  - Industrial (T2)
  - Engineering (T3)

- Use the Chronos (Shell) industrial resource setup
  - 70 resources per cluster
  - At start, the number of clusters is scaled to reach 70% resource utilization

ID	Source	Domain	Workflows	# Tasks
T1	SPEC Cloud Group	Scientific	200	13,876
T2	Chronos	Industrial	1,024	3,072
T3	Askalon EE	Engineering	757	45,786
T4	Askalon EE2	Engineering	3,551	122,105

Results: significant differences per domain

AS	Workload	A_U	A_O	$\bar{A}_U$	$\bar{A}_O$	T_U	T_O	k	k'	M_U	$\bar{V}$	$\bar{h}$	$\bar{C}$
React	Chronos	70.7	17.6	27							2.3	3,467.9	21.0
	Askalon EE	211.5	$\boxed{16.1}$	32							9.9	2,121.3	58.9
	SPEC	54.8	38.9	15							8.8	3,538.9	784.0
ConPaaS	Chronos	70.7	17.6	27							2.3	3,467.9	21.0
	Askalon EE	211.5	$\boxed{50.4}$	31							8.1	3,355.9	93.2
	SPEC	52.9	40.2								0.0	3,597.6	797.0
Hist	Chronos	70.7	17.6								2.3	3,467.9	21.0
	Askalon EE	211.5	$\boxed{56.2}$	31							8.3	3,565.2	99.0
	SPEC	53.9	40.2	14							0.0	3,597.6	797.0
Adapt	Chronos	70.7	17.6	27.8	17.6	45.9	50.5	0.0	0.0	36.0	202.3	3,467.9	21.0
	Askalon EE	211.5	$\boxed{17.6}$	32.4	47.5	40.0	56.0	4.0	1.0	18.2	1,353.6	2,175.4	60.4
	SPEC	54.9	38.8	15.7	38.8	33.6	64.6	1.6	0.8	70.9	68.8	3,538.9	784.0
Plan	Chronos	70.7	17.7	27.8	17.7	45.9	50.5	0.0	0.0	36.0	202.3	3,467.9	21.0
	Askalon EE	210.0	$\boxed{17.8}$	33.1						8.4	1,350.0	2,169.6	60.7
	SPEC	54.2	38.1	15.7						0.0	68.2	3,507.3	777.0
Reg	Chronos	70.7	17.6	27.8						6.0	202.3	3,467.9	21.0
	Askalon EE	210.0	$\boxed{16.1}$	33.1						6.7	1,312.4	2,109.2	59.0
	SPEC	53.9	39.7							1.6	69.4	3,570.5	791.0
Token	Chronos	70.7	17.6							6.0	202.3	3,467.9	21.0
	Askalon EE	211.5	$\boxed{16.8}$	31.4						7.4	1,335.3	2,146.1	59.6
	SPEC	53.4	40.3	14.8	40.3	32.7	66.1	2.6	0.0	72.5	70.0	3,597.6	797.0

For some metrics such as overprovisioning, ConPaaS and Hist perform worse for the EE workload

Autoscalers underprovision roughly equally on all workloads

VRIJE UNIVERSITEIT AMSTERDAM

Experiment: different allocation policies

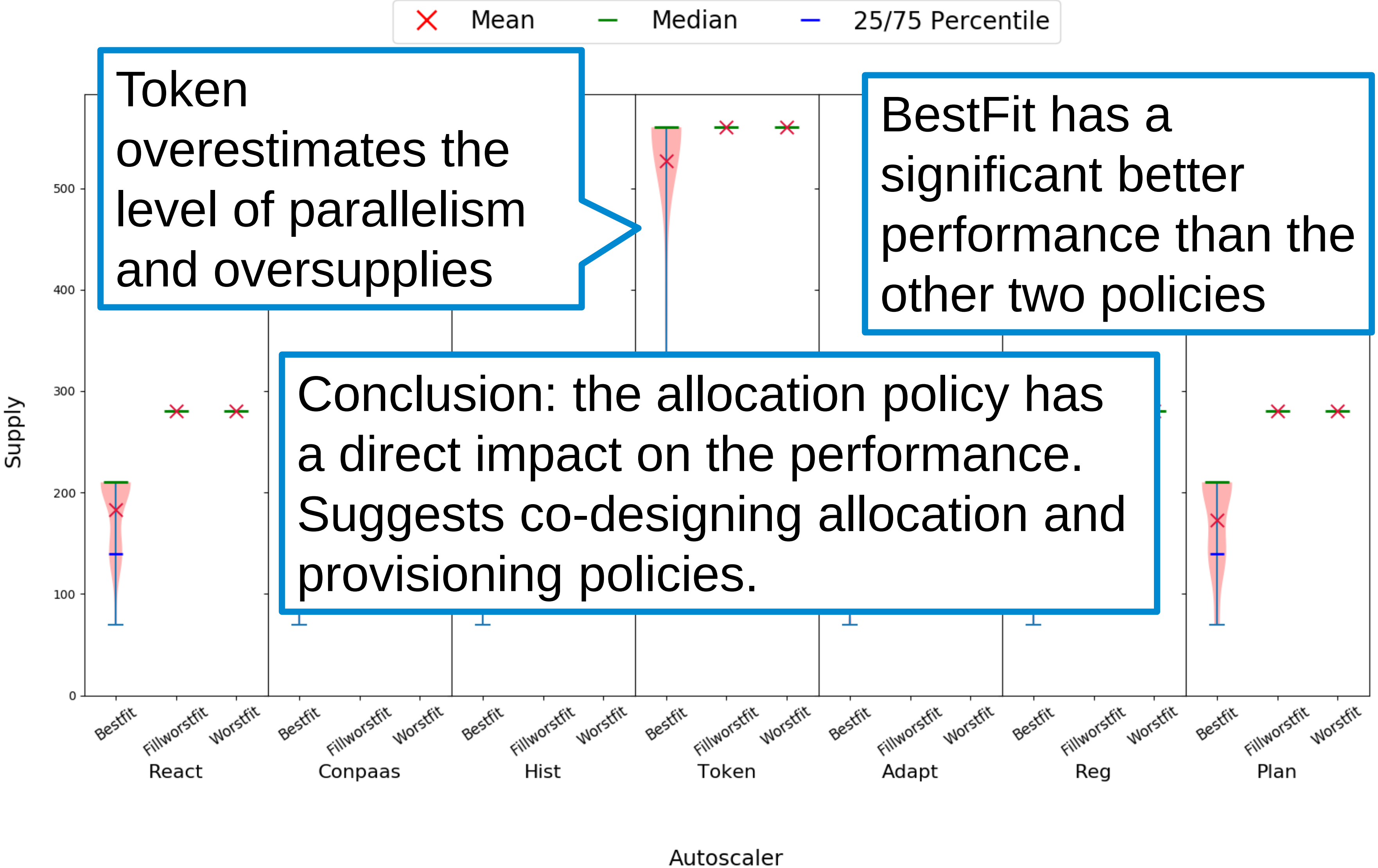
- Inspect the performance of autoscalers using different allocation policies
- One workload:
  - Engineering (T4)

- Three allocation policies
  - WorstFit
  - FillWorstFit
  - BestFit

- Use the Chronos (Shell) industrial setup
  - 70 resources per cluster
  - At start, the number of clusters is scaled to reach 70% resource utilization

- Metrics
  - Supplied resources

Results



Token overestimates the level of parallelism and oversupplies

BestFit has a significant better performance than the other two policies

Conclusion: the allocation policy has a direct impact on the performance. Suggests co-designing allocation and provisioning policies.

VRIJE UNIVERSITEIT AMSTERDAM

Conclusion and future work

- We compared seven autoscalers using four distinct traces

In this work four experiments:

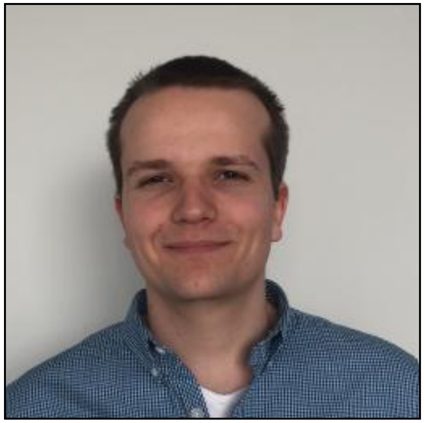
1. The performance differs significantly per application domain
2. The allocation policy has a direct impact on performance
3. *All autoscalers perform similar on bursty workloads in terms of NSL*
4. *Some autoscalers overprovision more while yielding no better NSL*

Future work:

- Study the impact of heterogeneity
- Apply several cost metrics by using e.g. cost models
- Experiment with job and task migrations
- Improve our simulations by including resource boot-up times

Interested in the other two experiments or the paper? Let me know!

A Trace-Based Performance Study of Autoscaling Workloads of Workflows in Datacenters



Mihai Neacşu, BSc.

Prof. Dr. Ir. Alexandru Iosup

Ir. Laurens Versluis
Massivizing Computer Systems
@Large research,
https://atlarge-research.com