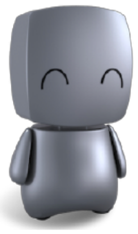


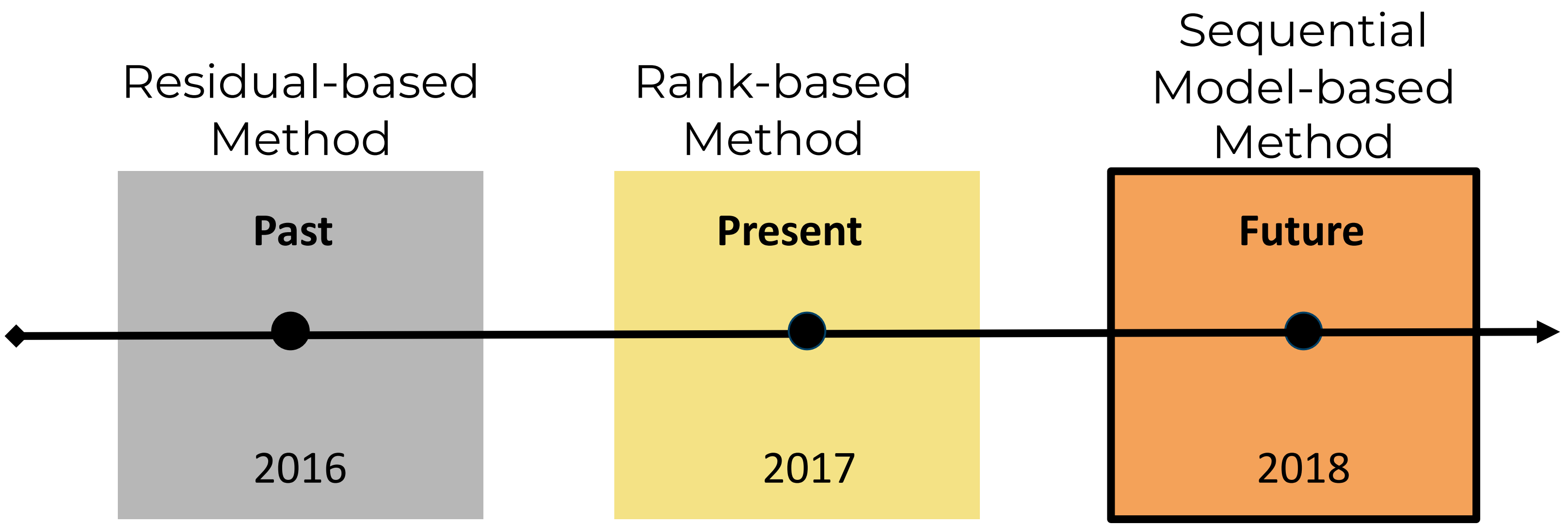
ai4se.net

Frugal Ways of Finding “Good” Configurations

Vivek Nair
Advisor: Dr. Tim Menzies

NC STATE UNIVERSITY

Flashback from last exam



Future Work: When will Flash win?

- Flash can reduce the cost of performance optimization.
- Flash can be adapted to solve multi-objective performance optimization.

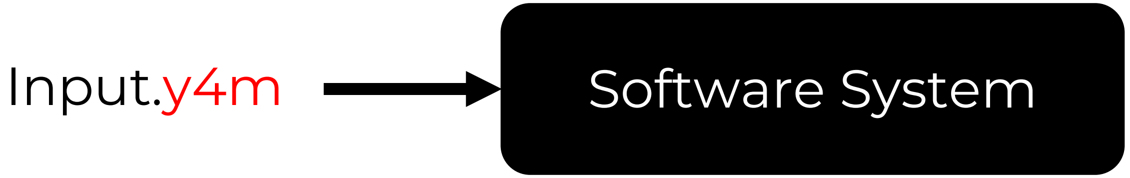
Statement of Thesis

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

What?

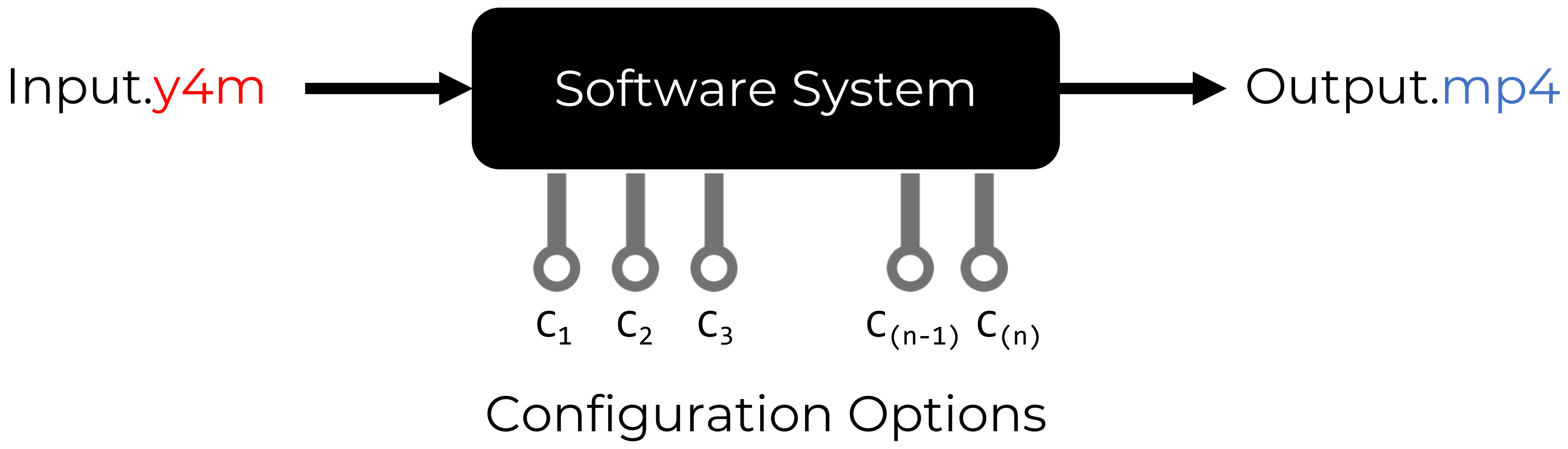
Software System

What?

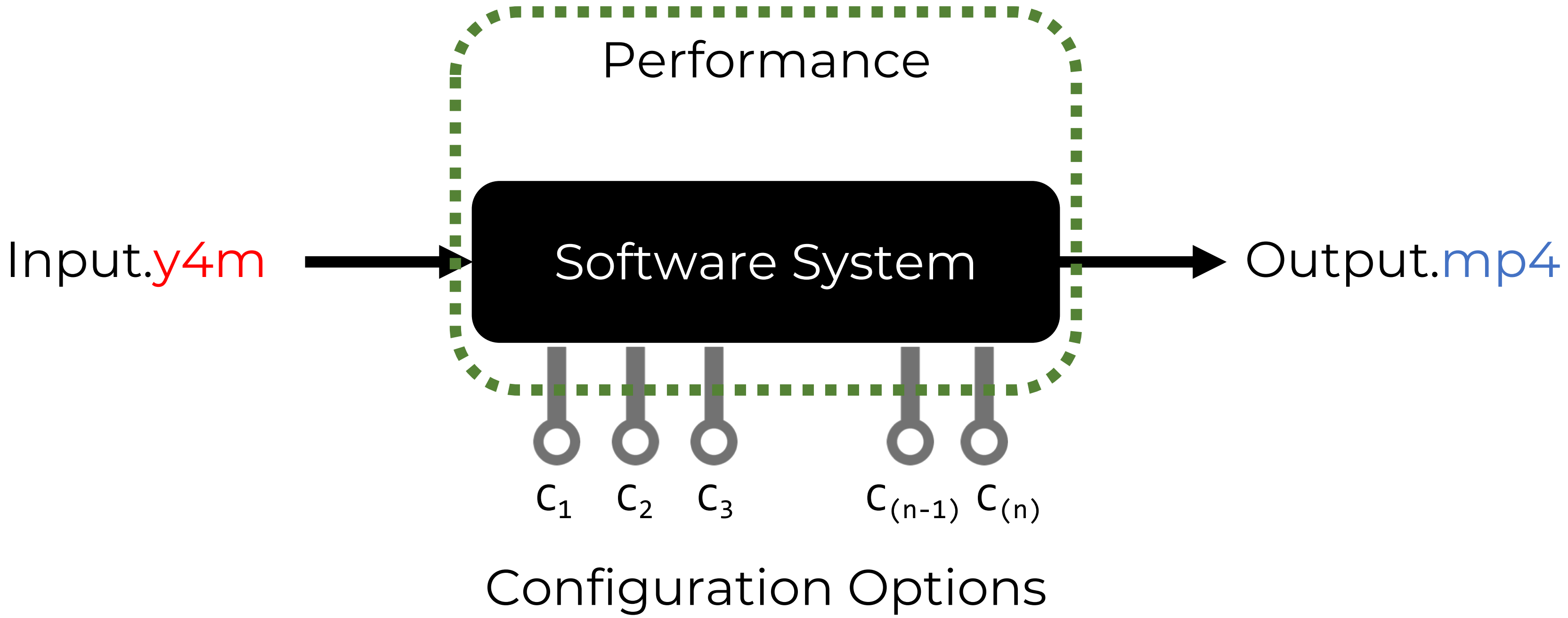


What?

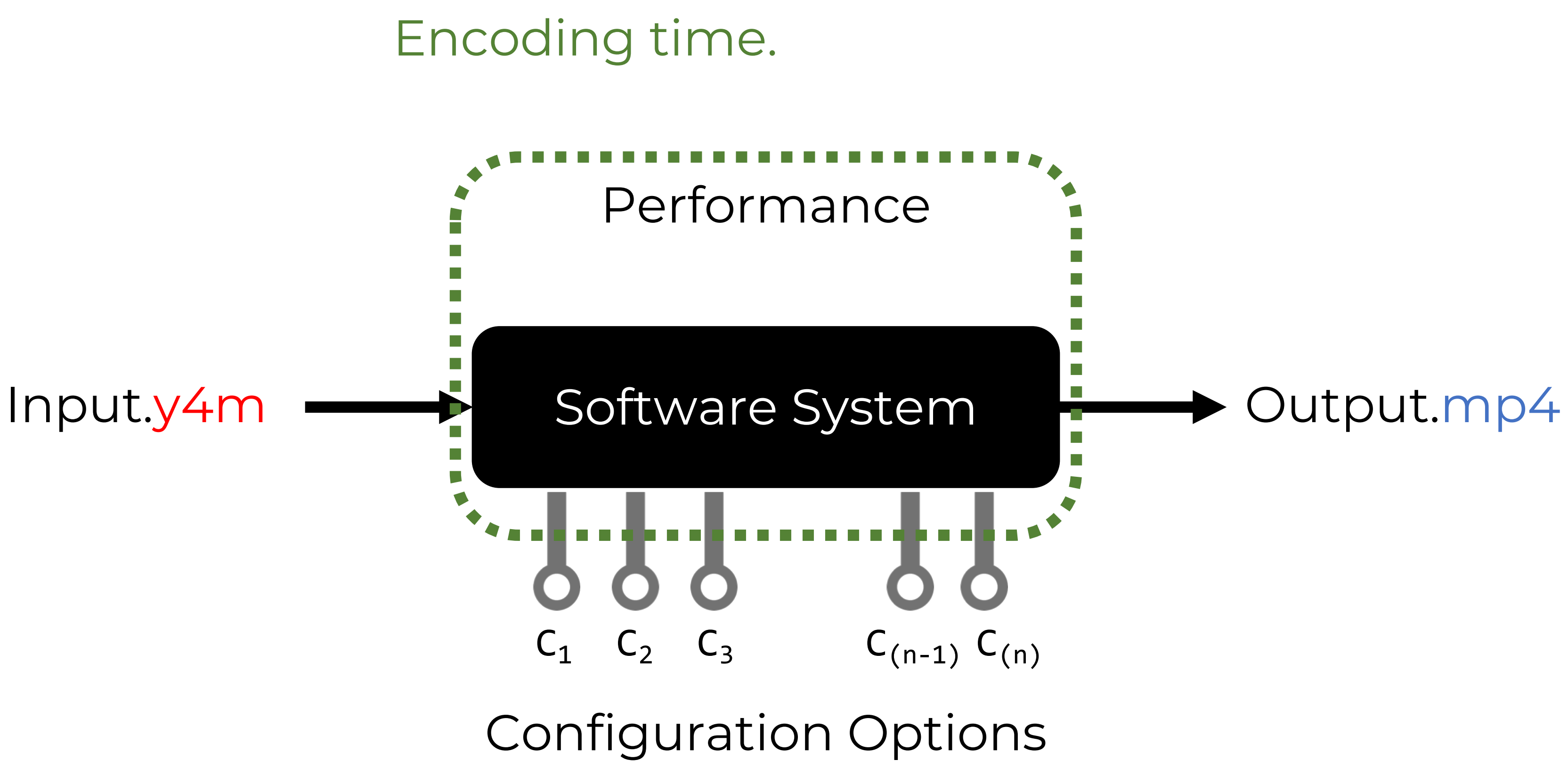
What?


Input.y4m
Software System
Output.mp4
C_1 C_2 C_3 $C_{(n-1)}$ $C_{(n)}$
Configuration Options

What?

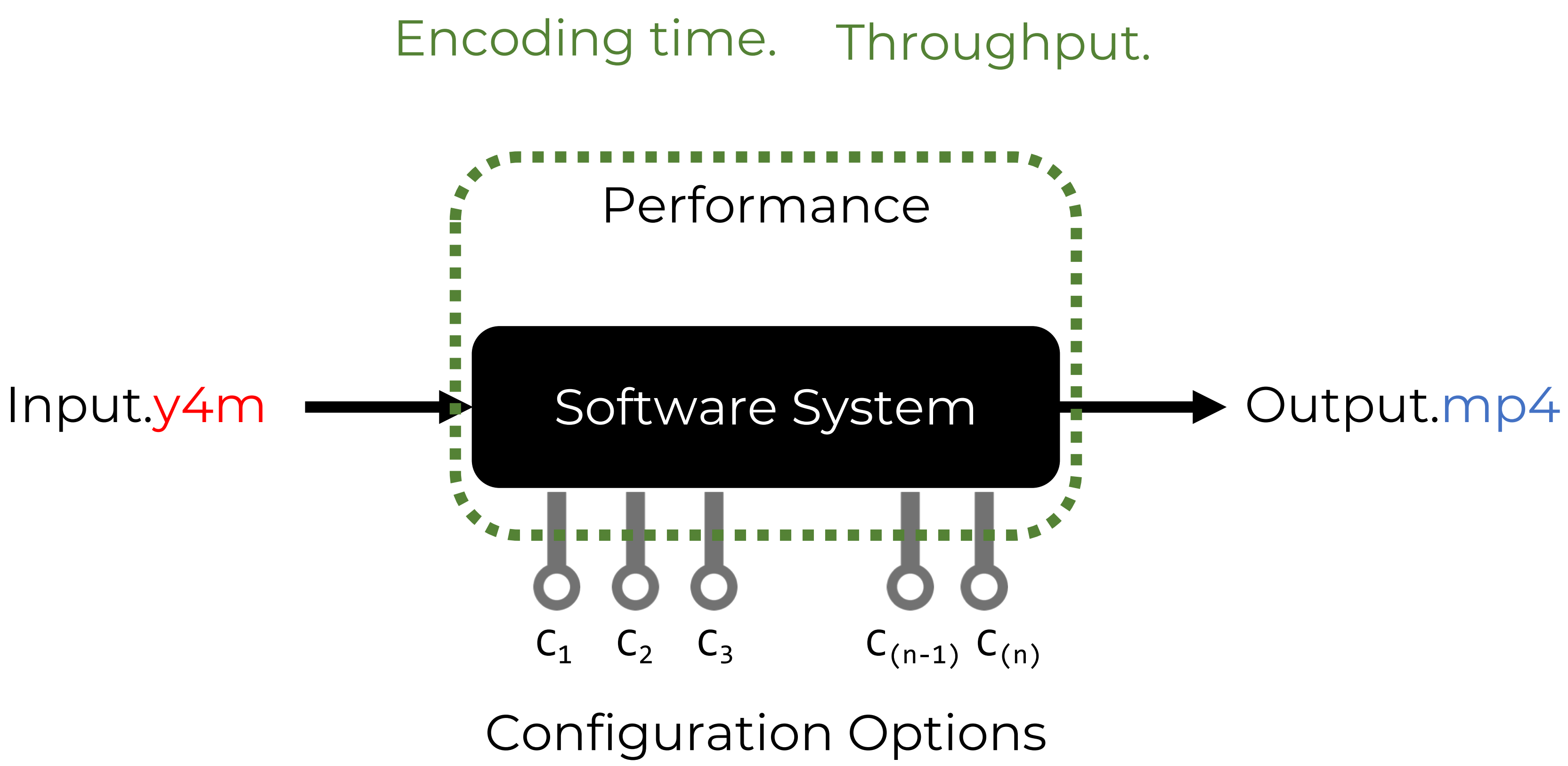

Performance
Input.y4m
Software System
Output.mp4
c_1 c_2 c_3 $c_{(n-1)}$ $c_{(n)}$
Configuration Options

What?



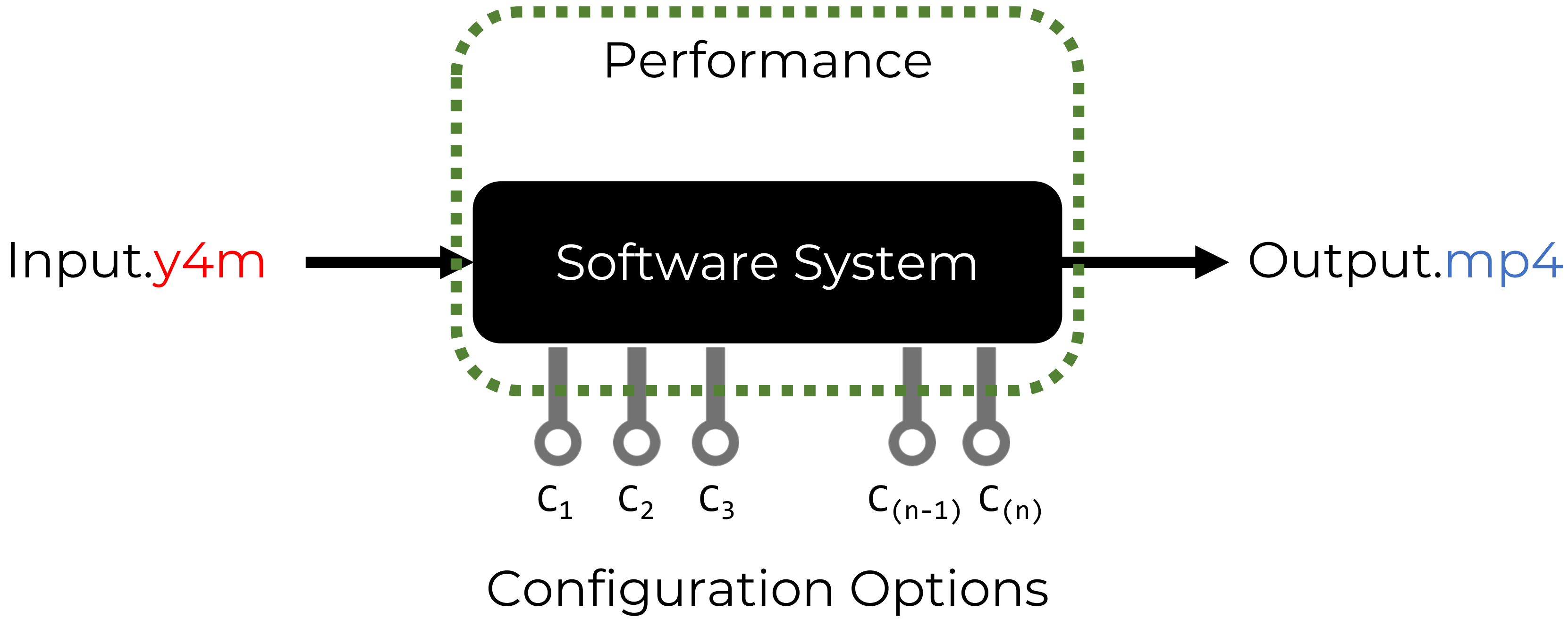
Encoding time.
Performance
Input.y4m
Software System
Output.mp4
c_1 c_2 c_3 $c_{(n-1)}$ $c_{(n)}$
Configuration Options

What?


Encoding time.
Throughput.
Performance
Input.y4m
Software System
Output.mp4
C1
C2
C3
C(n-1)
C(n)
Configuration Options

What?

Find (near) optimal configuration of a software system while minimizing measurements



What?

Features																Perf. (s)
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	y_i
1	1	0	1	1	1	1	0	1	0	0	1	1	0	0	1	651
1	1	1	1	1	1	0	1	1	1	0	0	1	0	1	0	536
1	1	1	1	0	0	0	0	1	1	0	0	1	0	0	1	581
1	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	381
1	1	0	1	0	0	0	1	1	1	0	0	1	0	1	0	424
1	1	0	0	1	0	1	1	1	1	0	0	1	0	0	1	615
1	0	1	0	1	1	1	0	1	1	0	0	1	0	1	0	477
1	0	1	0	0	0	0	1	1	0	0	1	1	1	0	0	263
1	0	0	0	0	0	1	1	1	0	0	1	1	1	0	0	272
1	1	1	1	0	0	0	1	1	0	0	1	1	1	0	0	247
1	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	612
1	0	1	1	1	0	0	0	1	0	0	1	1	0	1	0	510
1	1	1	1	0	1	1	0	1	0	1	0	1	0	0	1	555
1	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	264
1	0	1	0	0	1	1	1	1	0	0	1	1	0	0	1	576
1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	268

What?

Configuration Options

Features																Perf. (s)
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	y_i
1	1	0	1	1	1	1	0	1	0	0	1	1	0	0	1	651
1	1	1	1	1	1	0	1	1	1	0	0	1	0	1	0	536
1	1	1	1	0	0	0	0	1	1	0	0	1	0	0	1	581
1	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	381
1	1	0	1	0	0	0	1	1	1	0	0	1	0	1	0	424
1	1	0	0	1	0	1	1	1	1	0	0	1	0	0	1	615
1	0	1	0	1	1	1	0	1	1	0	0	1	0	1	0	477
1	0	1	0	0	0	0	1	1	0	0	1	1	1	0	0	263
1	0	0	0	0	0	1	1	1	0	0	1	1	1	0	0	272
1	1	1	1	0	0	0	1	1	0	0	1	1	1	0	0	247
1	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	612
1	0	1	1	1	0	0	0	1	0	0	1	1	0	1	0	510
1	1	1	1	0	1	1	0	1	0	1	0	1	0	0	1	555
1	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	264
1	0	1	0	0	1	1	1	1	0	0	1	1	0	0	1	576
1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	268

What?

Configuration Options

Configuration

Features																Perf. (s)
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	y_i
1	1	0	1	1	1	1	0	1	0	0	1	1	0	0	1	651
1	1	1	1	1	1	0	1	1	1	0	0	1	0	1	0	536
1	1	1	1	0	0	0	0	1	1	0	0	1	0	0	1	581
1	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	381
1	1	0	1	0	0	0	1	1	1	0	0	1	0	1	0	424
1	1	0	0	1	0	1	1	1	1	0	0	1	0	0	1	615
1	0	1	0	1	1	1	0	1	1	0	0	1	0	1	0	477
1	0	1	0	0	0	0	1	1	0	0	1	1	1	0	0	263
1	0	0	0	0	0	1	1	1	0	0	1	1	1	0	0	272
1	1	1	1	0	0	0	1	1	0	0	1	1	1	0	0	247
1	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	612
[illegible]	[illegible]	[illegible]	[illegible]	1	0	0	0	1	0	0	1	1	0	1	0	510
[illegible]	[illegible]	[illegible]	[illegible]	0	1	1	0	1	0	1	0	1	0	0	1	555
1	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	264
1	0	1	0	0	1	1	1	1	0	0	1	1	0	0	1	576
1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	268

What?

Configuration Options

							Features									Perf. (s)
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	y_i
1	1	0	1	1	1	1	0	1	0	0	1	1	0	0	1	651
1	1	1	1	1	1	0	1	1	1	0	0	1	0	1	0	536
1	1	1	1	0	0	0	0	1	1	0	0	1	0	0	1	581
1	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	381
1	1	0	1	0	0	0	1	1	1	0	0	1	0	1	0	424
1	1	0	0	1	0	1	1	1	1	0	0	1	0	0	1	615
1	0	1	0	1	1	1	0	1	1	0	0	1	0	1	0	477
1	0	1	0	0	0	0	1	1	0	0	1	1	1	0	0	263
1	0	0	0	0	0	1	1	1	0	0	1	1	1	0	0	272
1	1	1	1	0	0	0	1	1	0	0	1	1	1	0	0	247
1	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	612
[illegible]	[illegible]	[illegible]	[illegible]	1	0	0	0	1	0	0	1	1	0	1	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	0	1	1	0	1	0	1	0	1	0	0	[illegible]	[illegible]
1	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	264
1	0	1	0	0	1	1	1	1	0	0	1	1	0	0	1	576
1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	268

Configuration

Performance

What?

Configuration Options

Features																Perf. (s)
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	y_i
1	1	0	1	1	1	1	0	1	0	0	1	1	0	0	1	651
1	1	1	1	1	1	0	1	1	1	0	0	1	0	1	0	536
1	1	1	1	0	0	0	0	1	1	0	0	1	0	0	1	581
1	0	0	0	0	0	1	0	1	1	0	0	1	0	1	0	381
1	1	0	1	0	0	0	1	1	1	0	0	1	0	1	0	424
1	1	0	0	1	0	1	1	1	1	0	0	1	0	0	1	615
1	0	1	0	1	1	1	0	1	1	0	0	1	0	1	0	477
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0	0	1	1	0	0	1	1	1	0	0	263
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0	1	1	1	0	0	1	1	1	0	0	272
1	1	1	1	0	0	0	1	1	0	0	1	1	1	0	0	247
1	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	612
[illegible]	[illegible]	[illegible]	[illegible]	1	0	0	0	1	0	0	1	1	0	1	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	0	1	1	0	1	0	1	0	1	0	0	[illegible]	[illegible]
1	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	264
1	0	1	0	0	1	1	1	1	0	0	1	1	0	0	1	576
1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	268

Optimal Solution

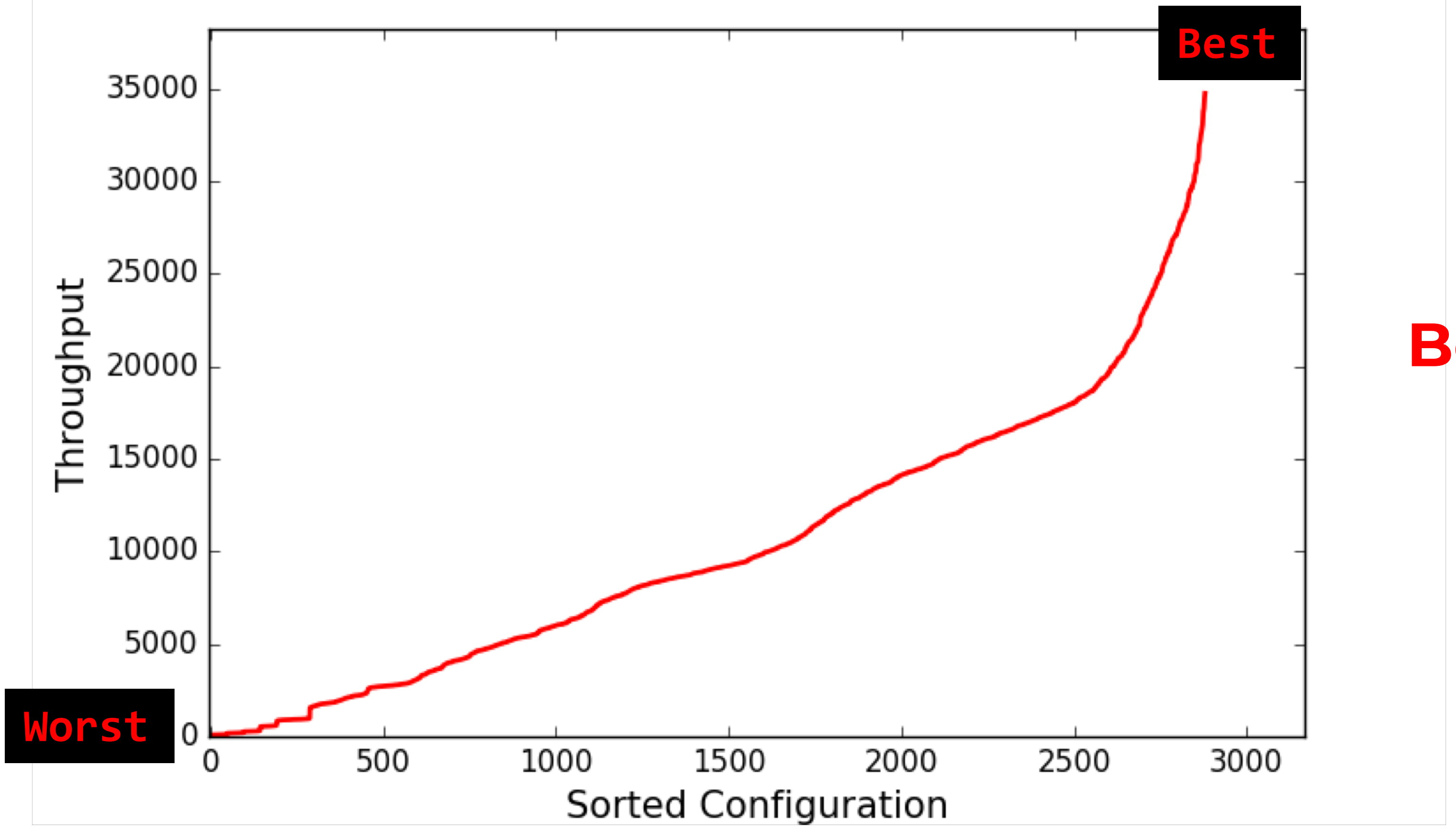
Configuration

Performance

Why is it important?

Why is it important?

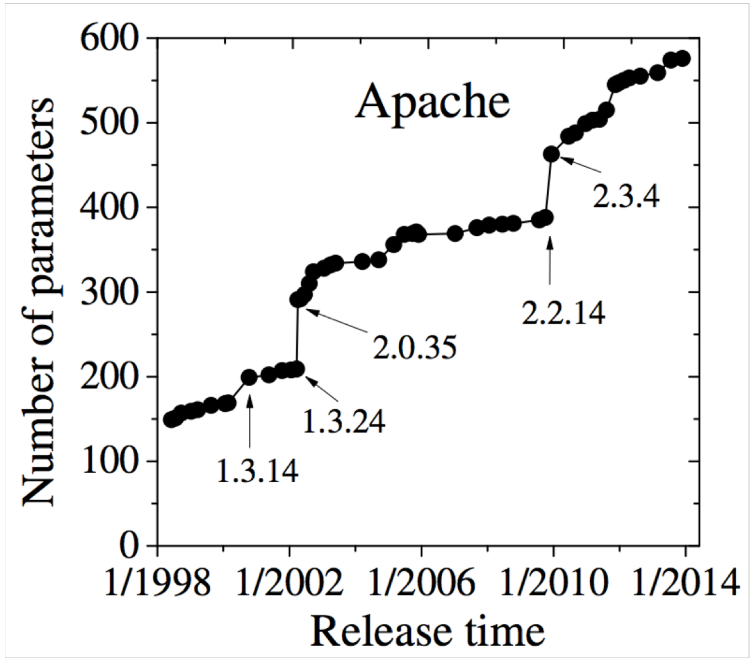
System: Apache Storm **Performance:** Throughput
Workload: Word Count **#Configuration options:** 6



Necessary

Best configuration is 480 times better than **Worst** configuration

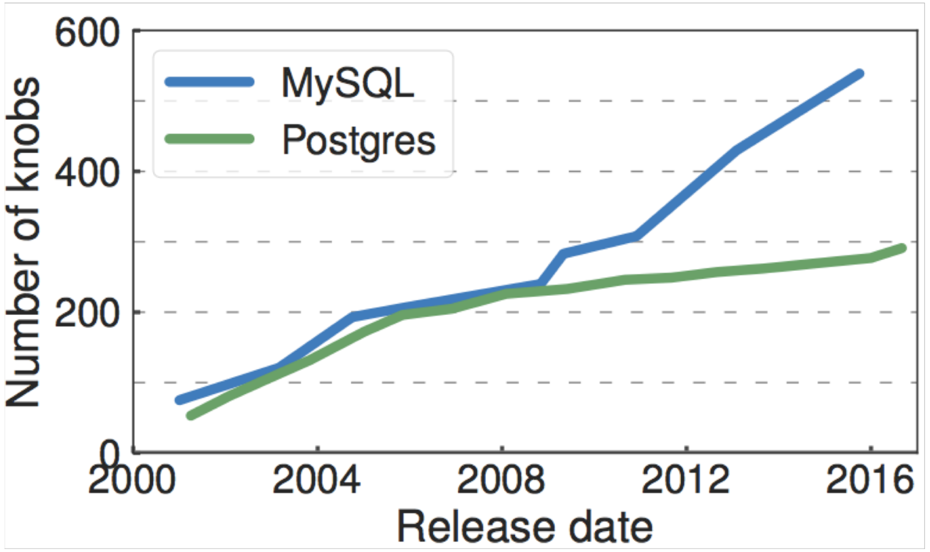
Why is it important?



200 new configuration options added to Apache HTTP server between 2010 and 2014[1]

Necessary

Complex



250 new configuration options added to MySQL between 2012 and 2016[2]

[1] Xu et. al.; Hey, you have given me too many knobs!: understanding and dealing with over-designed configuration in system software; FSE 2015

[2] Van Aken, Dana, et al. "Automatic Database Management System Tuning Through Large-scale Machine Learning." *ICMD* 2017.

Why is it important?

Default MySQL configuration in 2016 assumes that machine has only 160 MB of RAM[1]

Rule-of-thumb settings for WordCount (in Hadoop) gave one of its worst execution times[2]

Necessary

Complex

Default is not good

[1] Van Aken, Dana, et al. "Automatic Database Management System Tuning Through Large-scale Machine Learning." *International Conference on Management of Data*. ACM, 2017.
[2] Herodotou, Herodotos, et al. "Starfish: A Self-tuning System for Big Data Analytics." *CIDR*

Why is it important?

- Evaluation of single instance of software/hardware co-design problem can take weeks[1]
- Rolling Sort use-case required 21 days, within a total experimental time of about 2.5 months[2]
- Test suite generation using Evolutionary Algorithm can take weeks[3]

Necessary

Complex

Default is not good

Expensive

[1] Zuluaga, Marcela, et al. "Active learning for multi-objective optimization." *International Conference on Machine Learning*. 2013.
[2] Jamshidi, Pooyan, and Giuliano Casale. "An uncertainty-aware approach to optimal configuration of stream processing systems."*MASCOTS-2016*
[3] Wang, Tiantian, et al. "Searching for better configurations: a rigorous approach to clone evaluation." *FSE-2013*

Why is it important?

Cloud Computing

- Ernest
- Cherrypick
- PARIS

Database

- Otter-tune
- Ituned

Machine Learning

- Hyperparameter Tuning
- Random search
- SMBO
- Fabolas

Software Engineering

- Tuning or Default Values?
- Tuning for Software Analytics
- Tuning for Defect Prediction
- Topic Modelling

Necessary

Complex

Default is not good

Expensive

Ubiquitous

Why is it important?

Cloud Computing
- Ernest
- Cherrypick
- PARIS

Database
- Otter-tune
- Ituned

Machine Learning
- Hyperparameter Tuning
- Random search
- SMBO
- Fabolas

Software Engineering
- Tuning or Default Values?
- Tuning for Software Analytics
- Tuning for Defect Prediction
- Topic Modelling

Necessary

Complex

Default is not good

Expensive

Ubiquitous

Why is it important?

Performance Optimization

Necessary

Complex

Default is not good

Expensive

Ubiquitous

Why is it important?

CLOUDYN

SIGOPT

DataRobot

cloudability

uncountable

WHETLAB

Performance Optimization

- Necessary
- Complex
- Default is not good
- Expensive
- Ubiquitous

- Optimization is ubiquitous and expensive
- The Model-based optimization is a popular alternative

Claim: Better ways to build and use Models

Case Study: Configurable Software System Optimization

Potential future application: Any optimization problem

Previously on Performance Optimization [1][2]

Residual based Methods

[1] Guo et al.; "Variability-aware performance prediction: A statistical learning approach.”; ASE-2013
[2] Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t).”; ASE-2015

Previously…

Request

Response Time

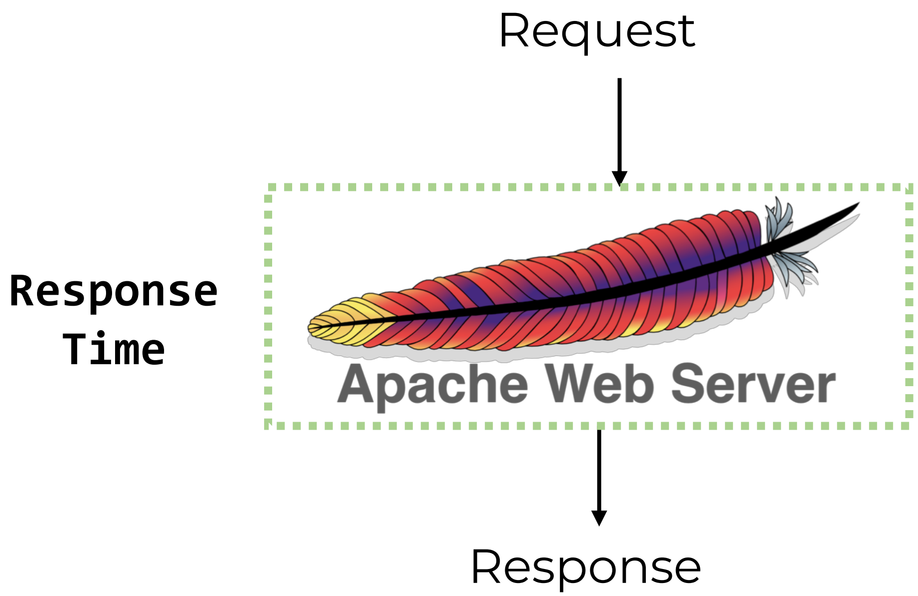
Apache Web Server

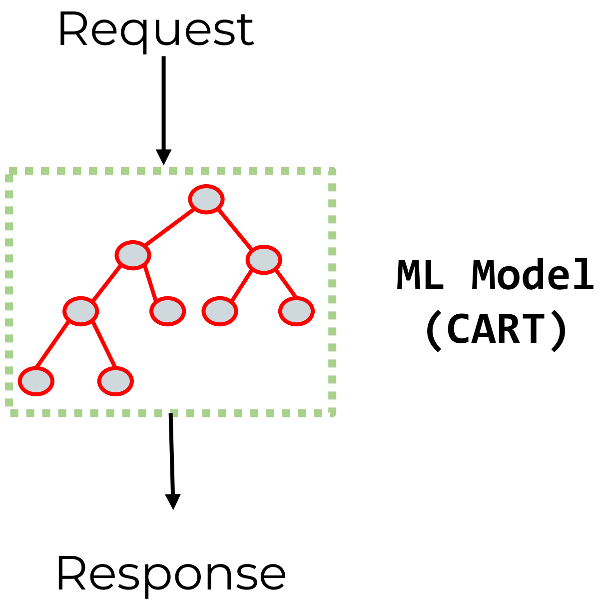
Response

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…



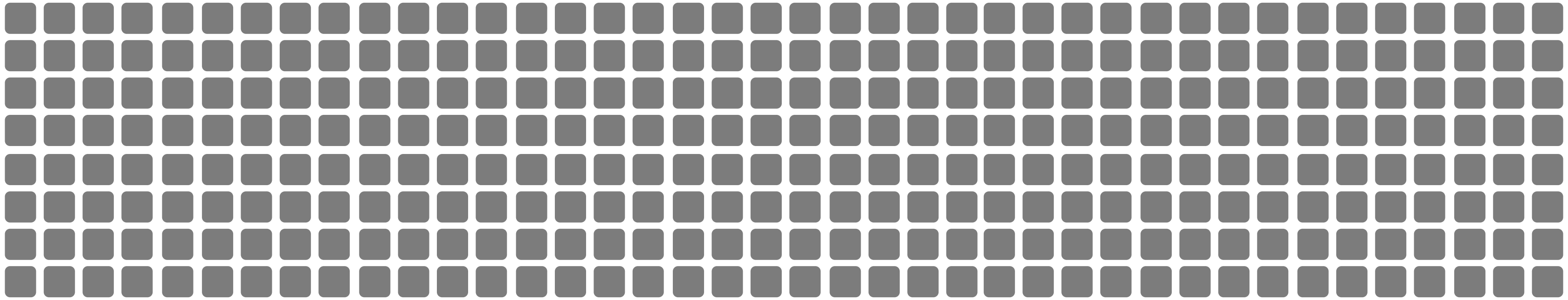


Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…

Configuration Space

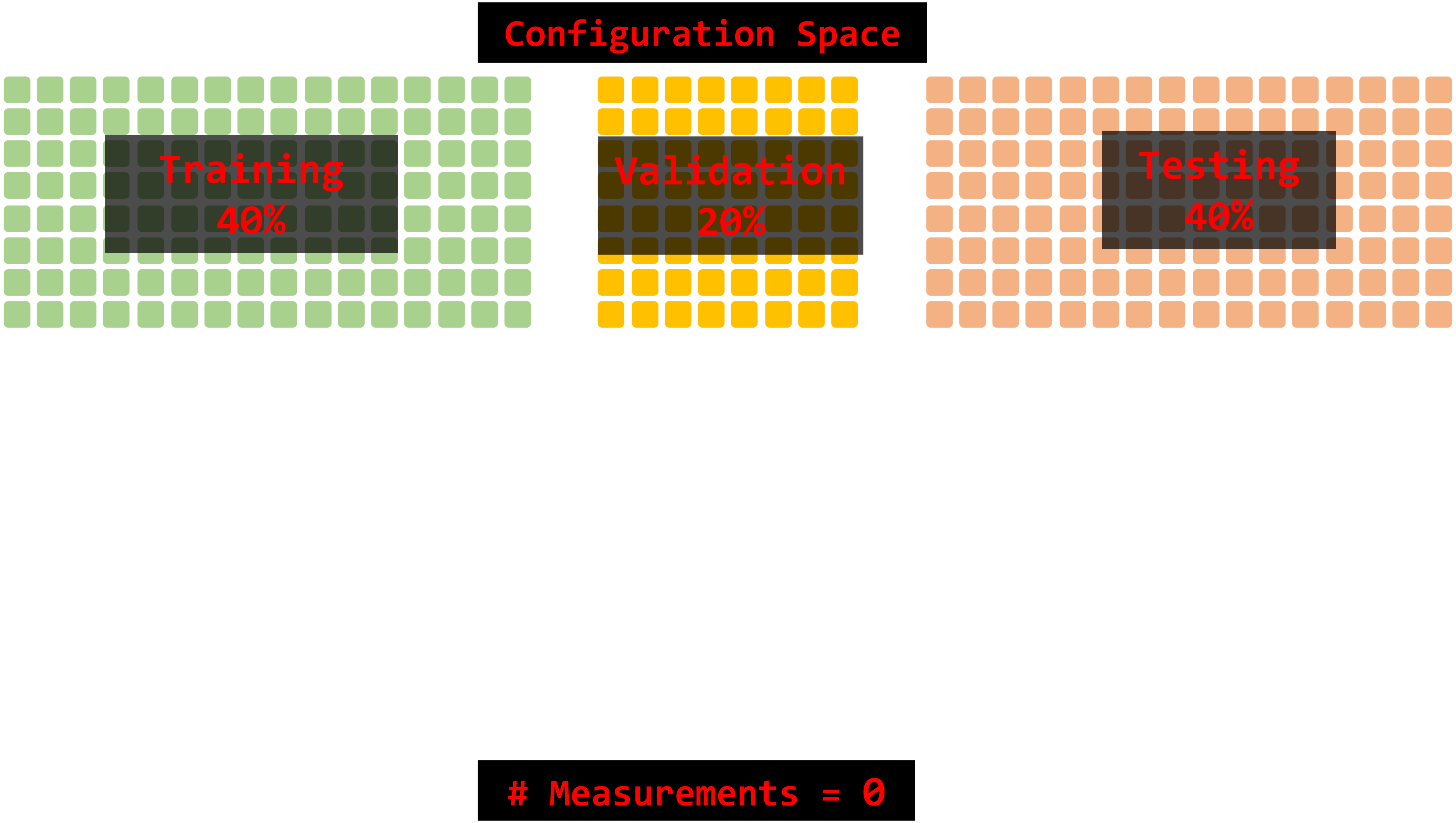


Measurements = 0

Guo et al.; "Variability-aware performance prediction: A statistical learning approach.”; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t).”; ASE-2015

Residual-based Method

Previously…

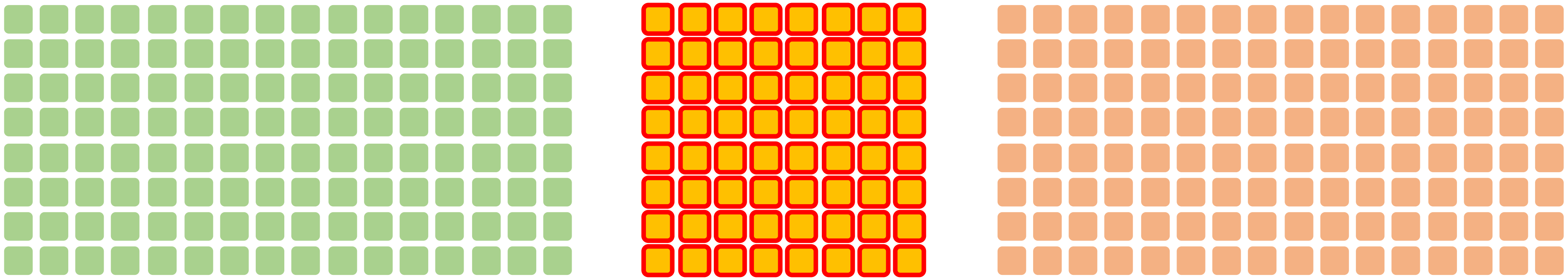


Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…

Configuration Space



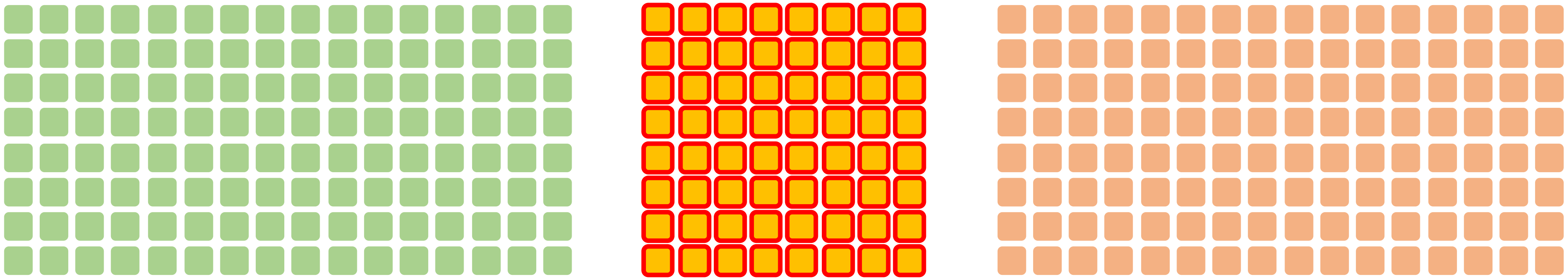
Measurements = 64

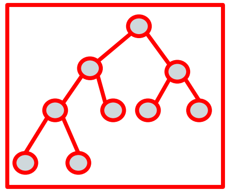
Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…

Configuration Space





CART

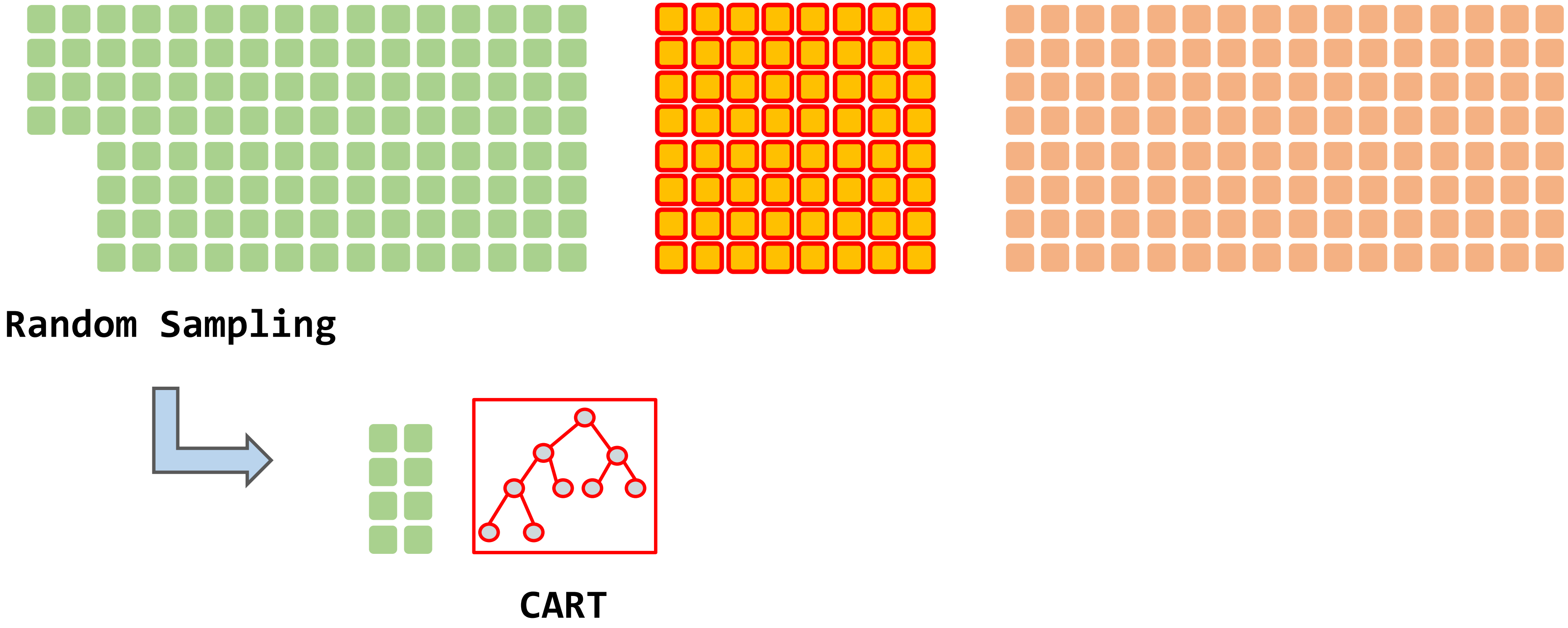
 # Measurements = 64

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously...

Configuration Space

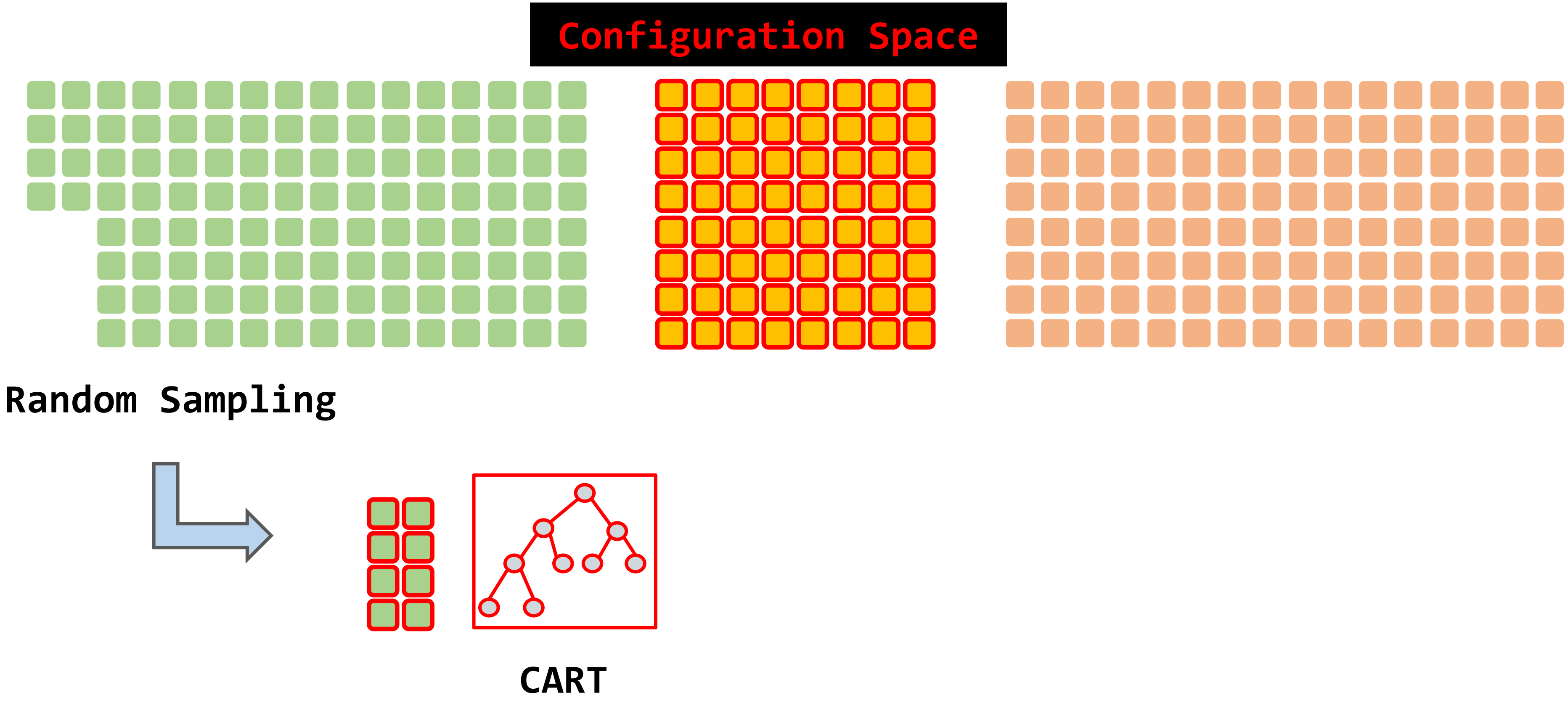


\# Measurements = 64

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…

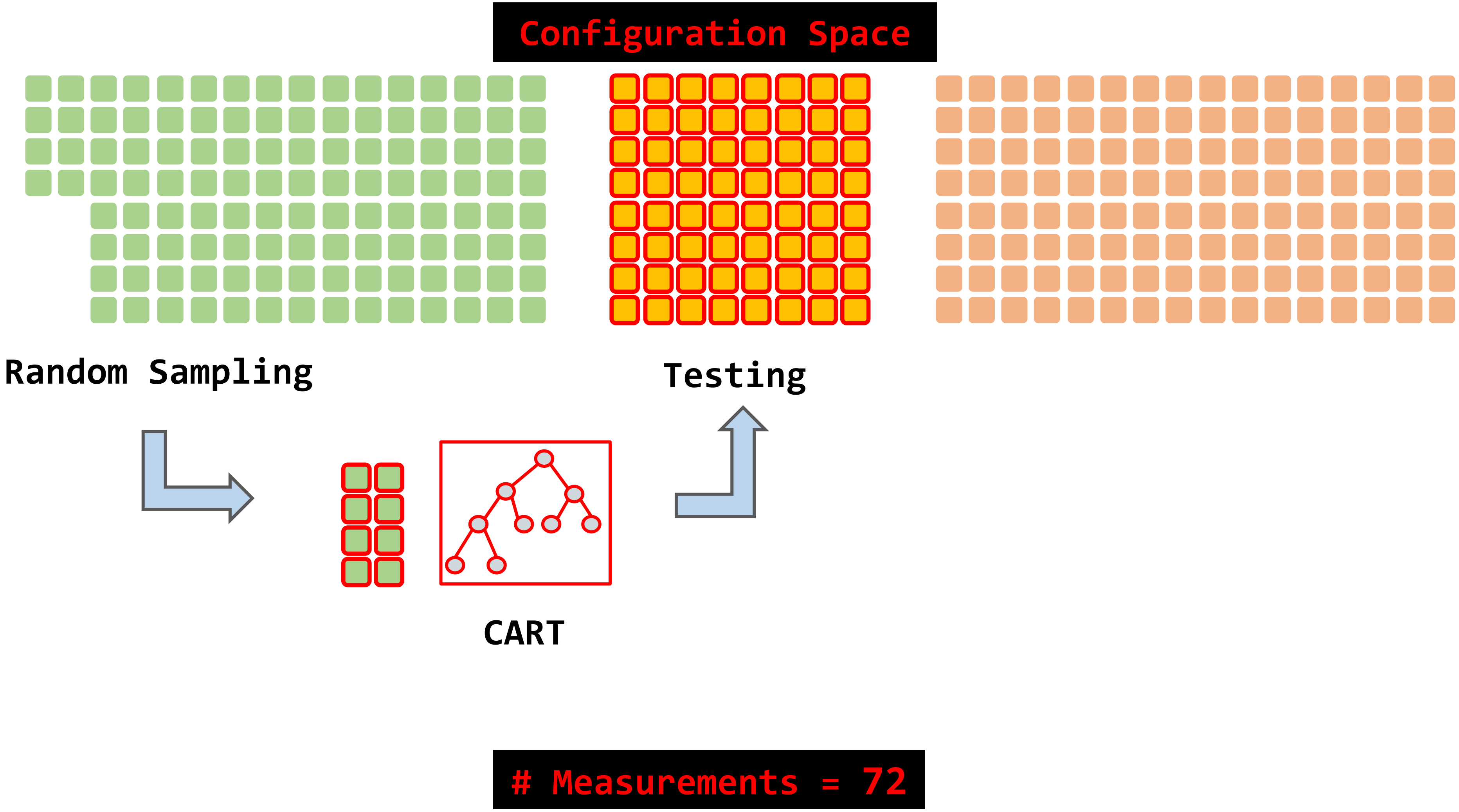


Measurements = 72

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

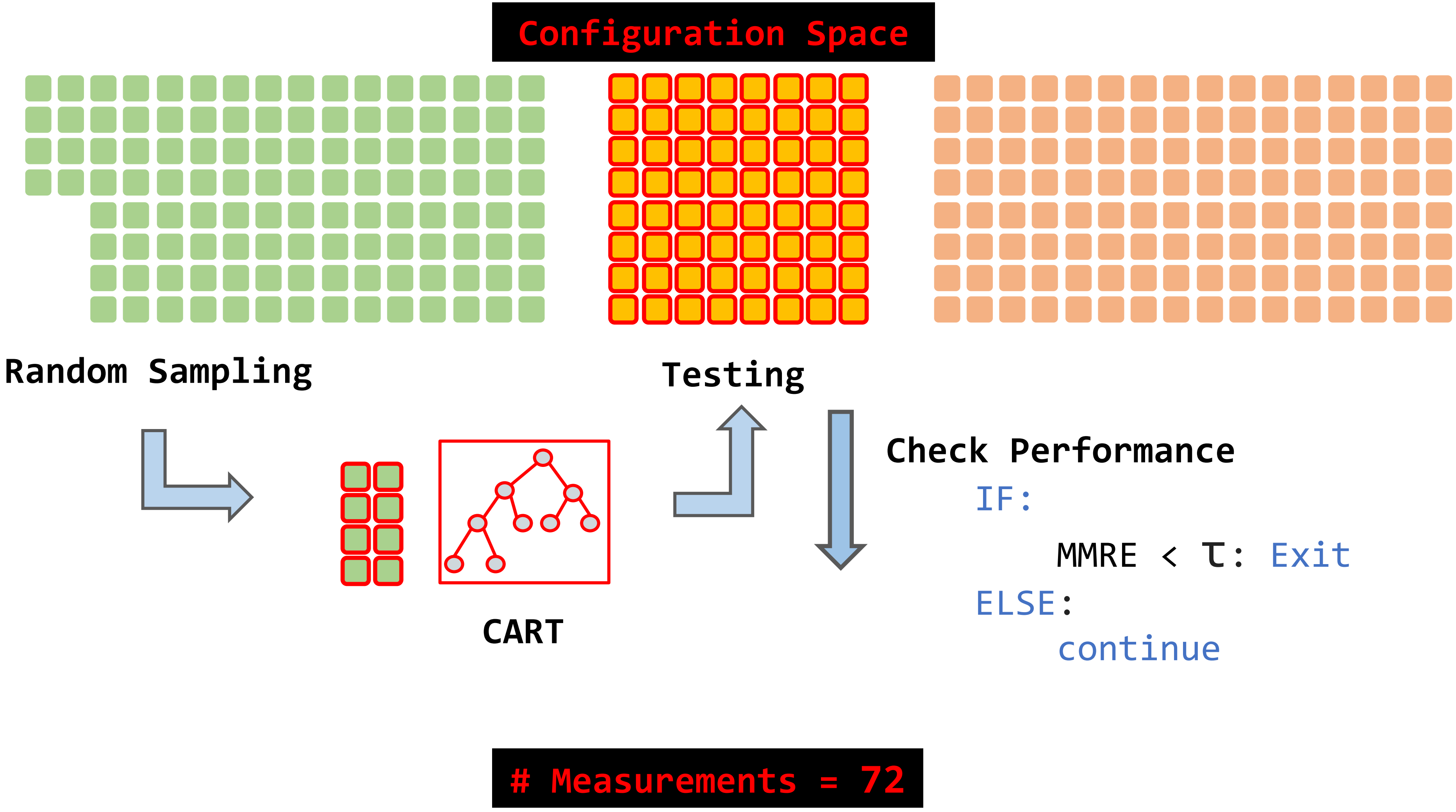
Previously...



Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

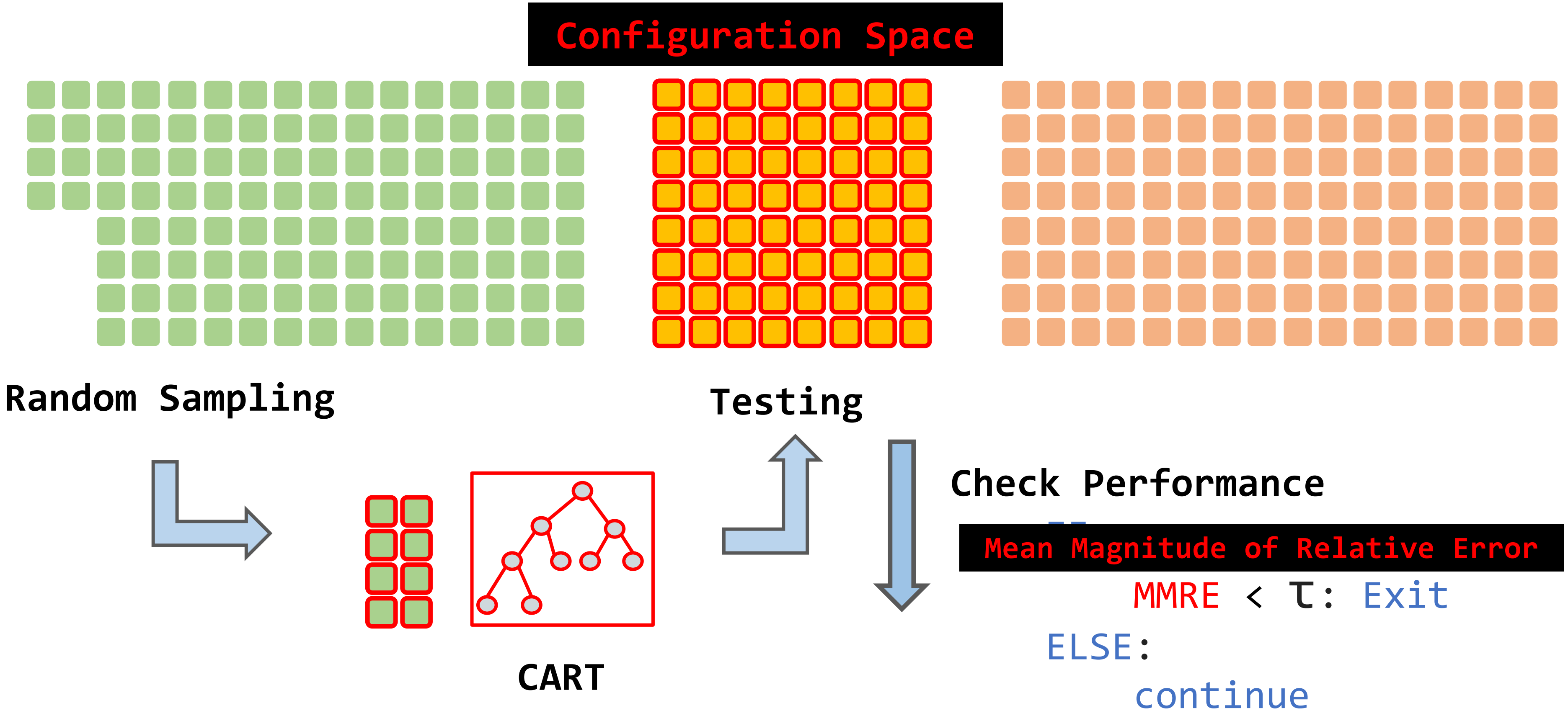
Previously...



Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously...

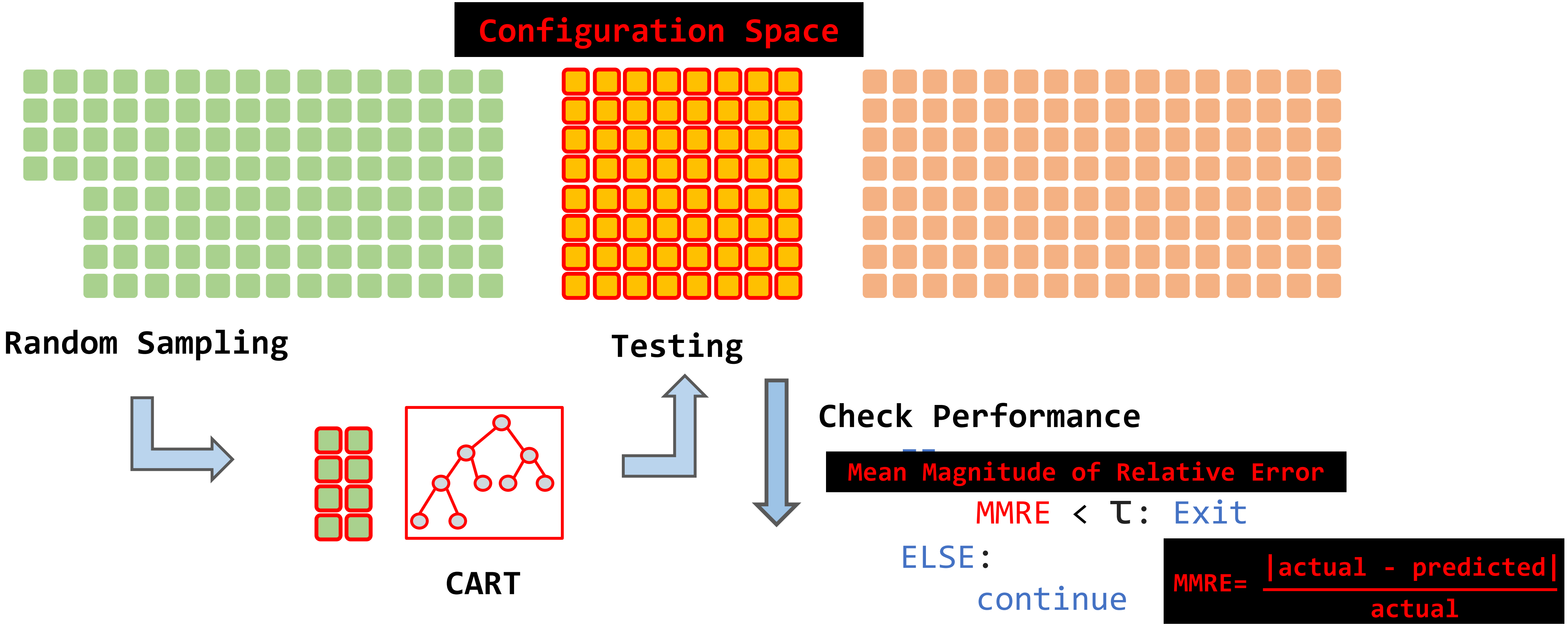


 # Measurements = 72

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…

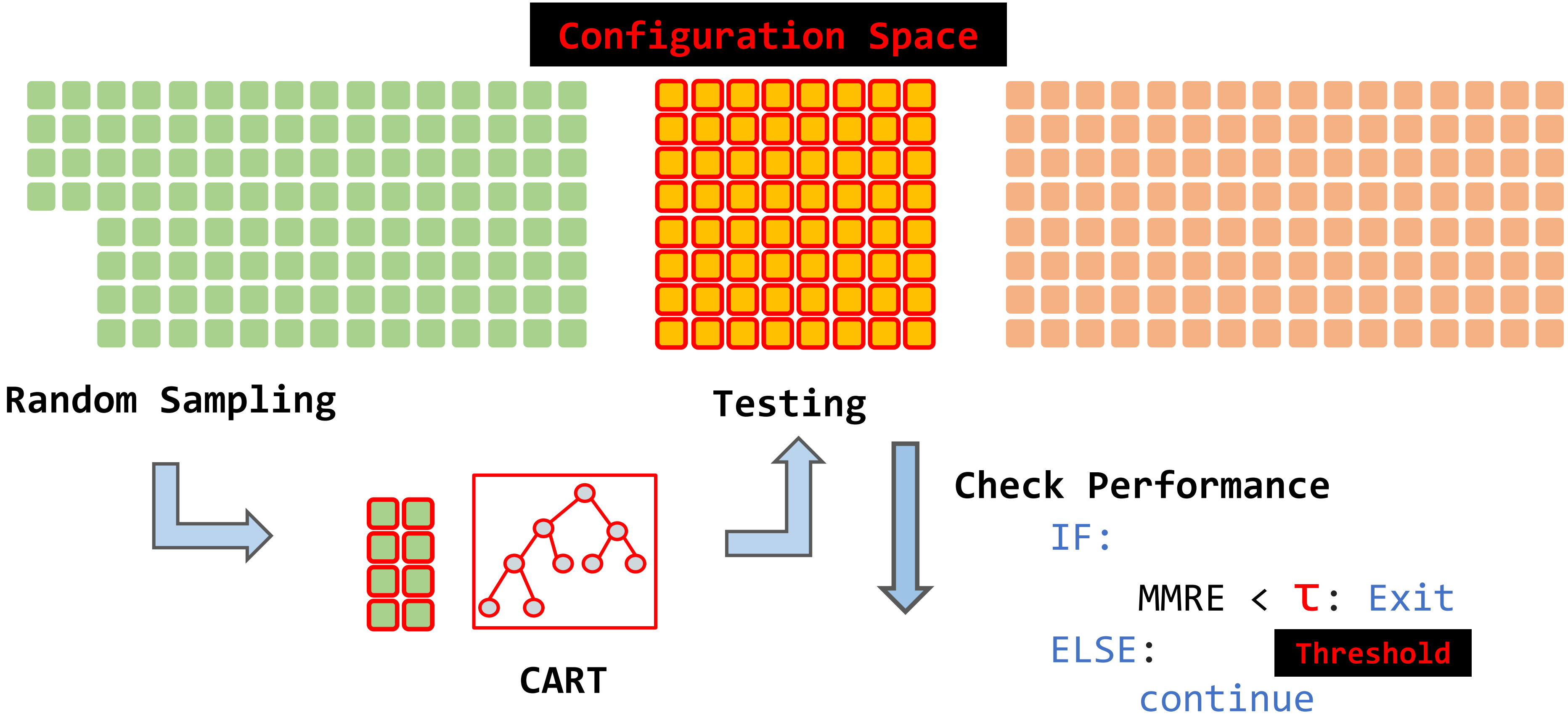


 # Measurements = 72

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…

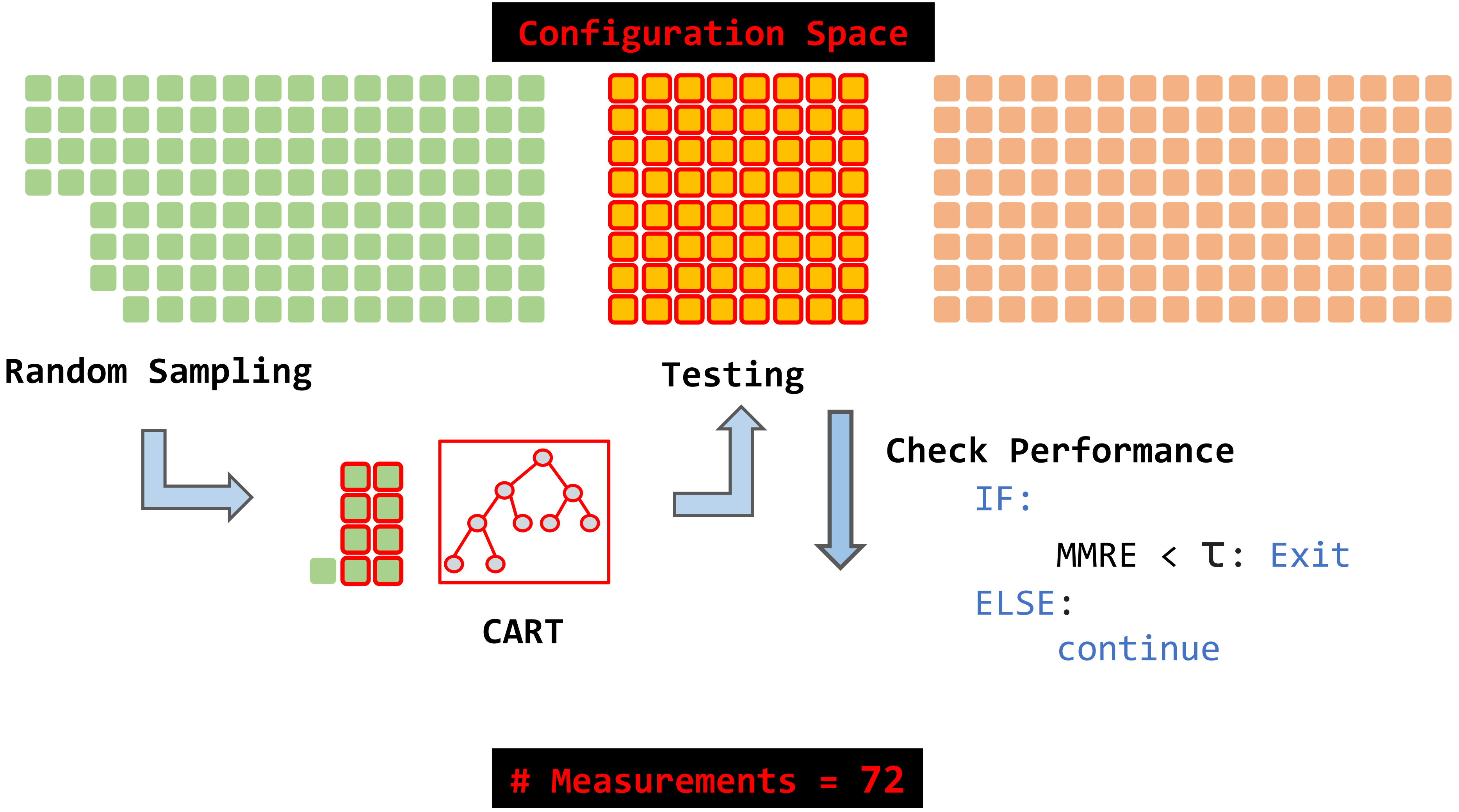


 # Measurements = 72

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

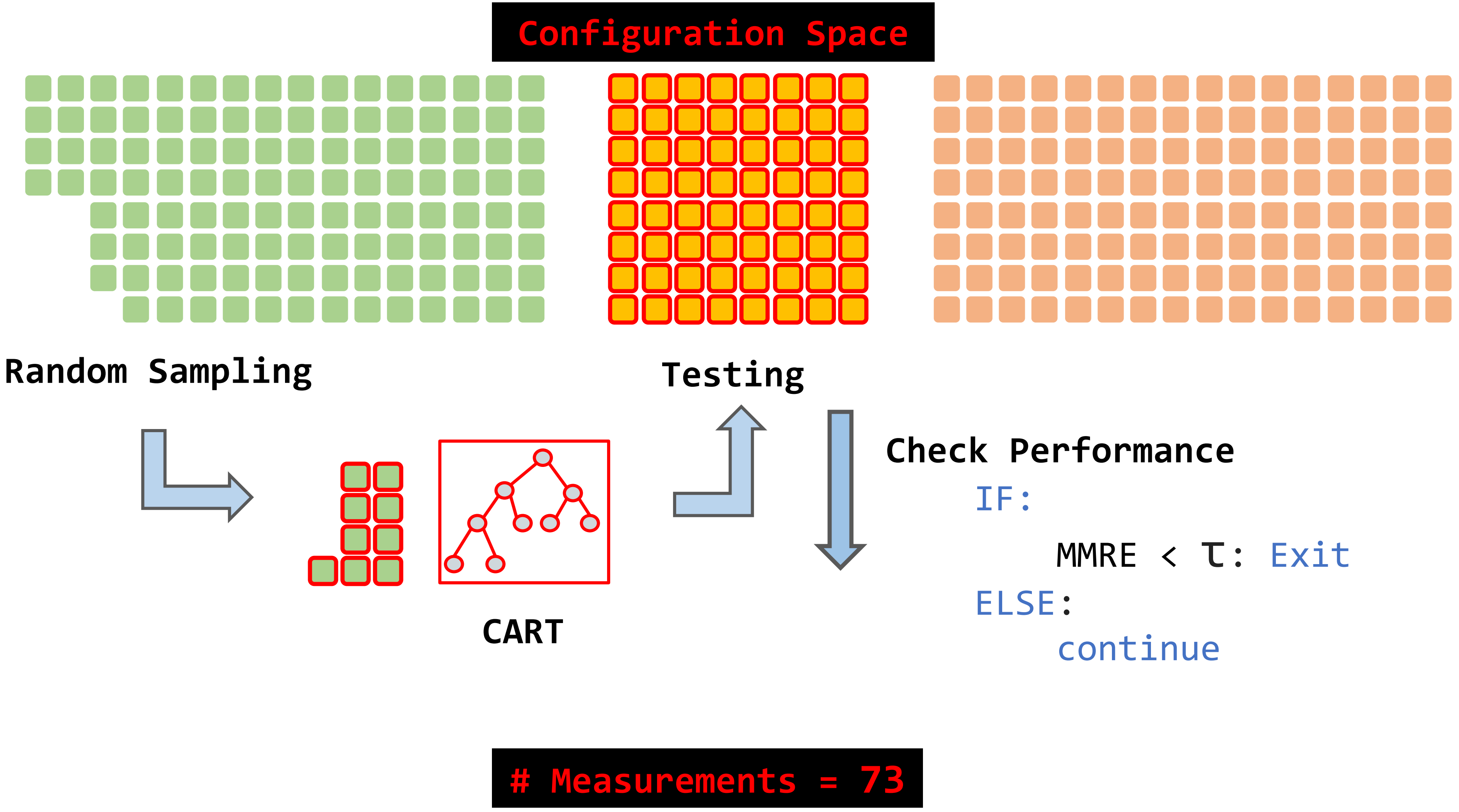
Previously...



Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

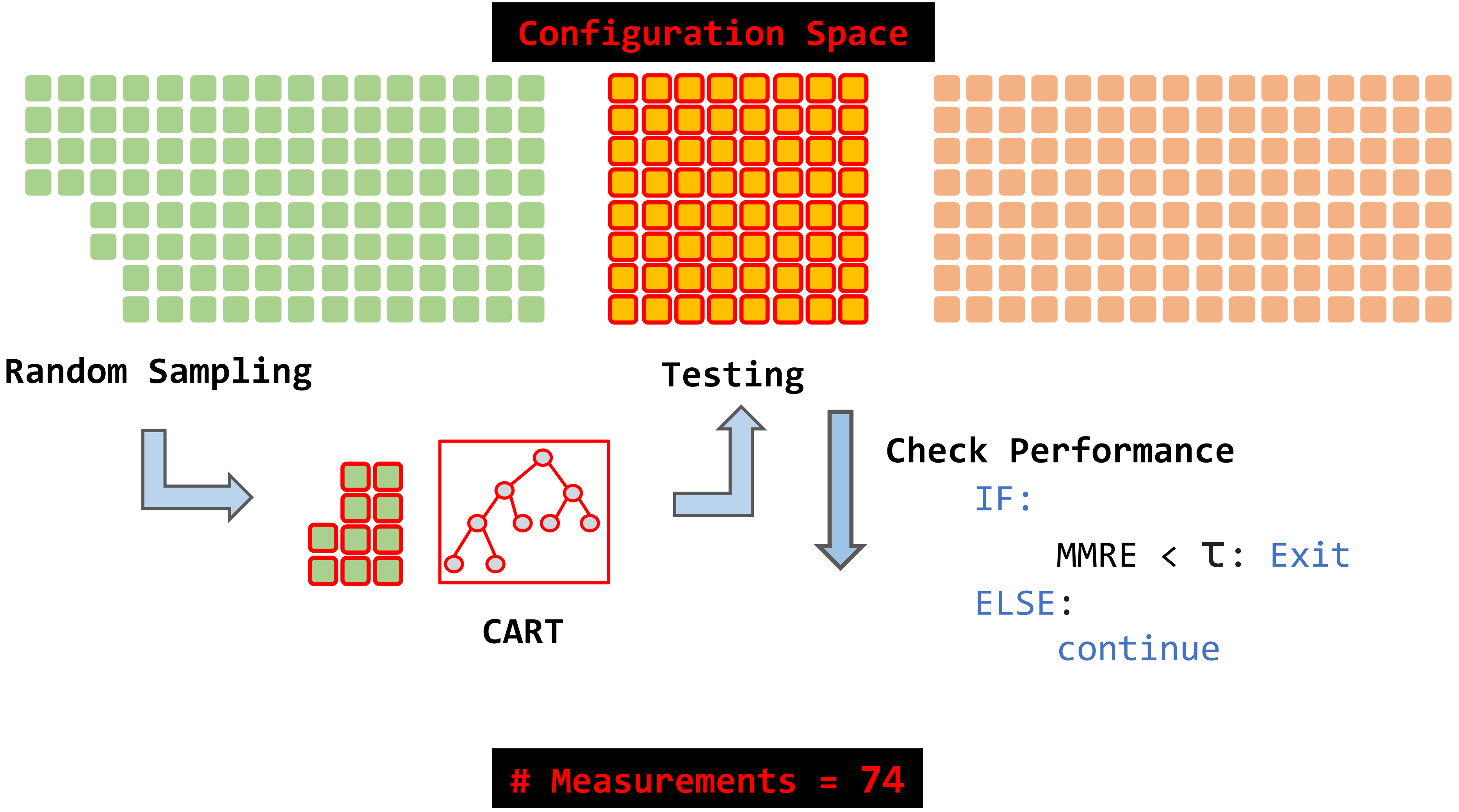
Previously...



Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

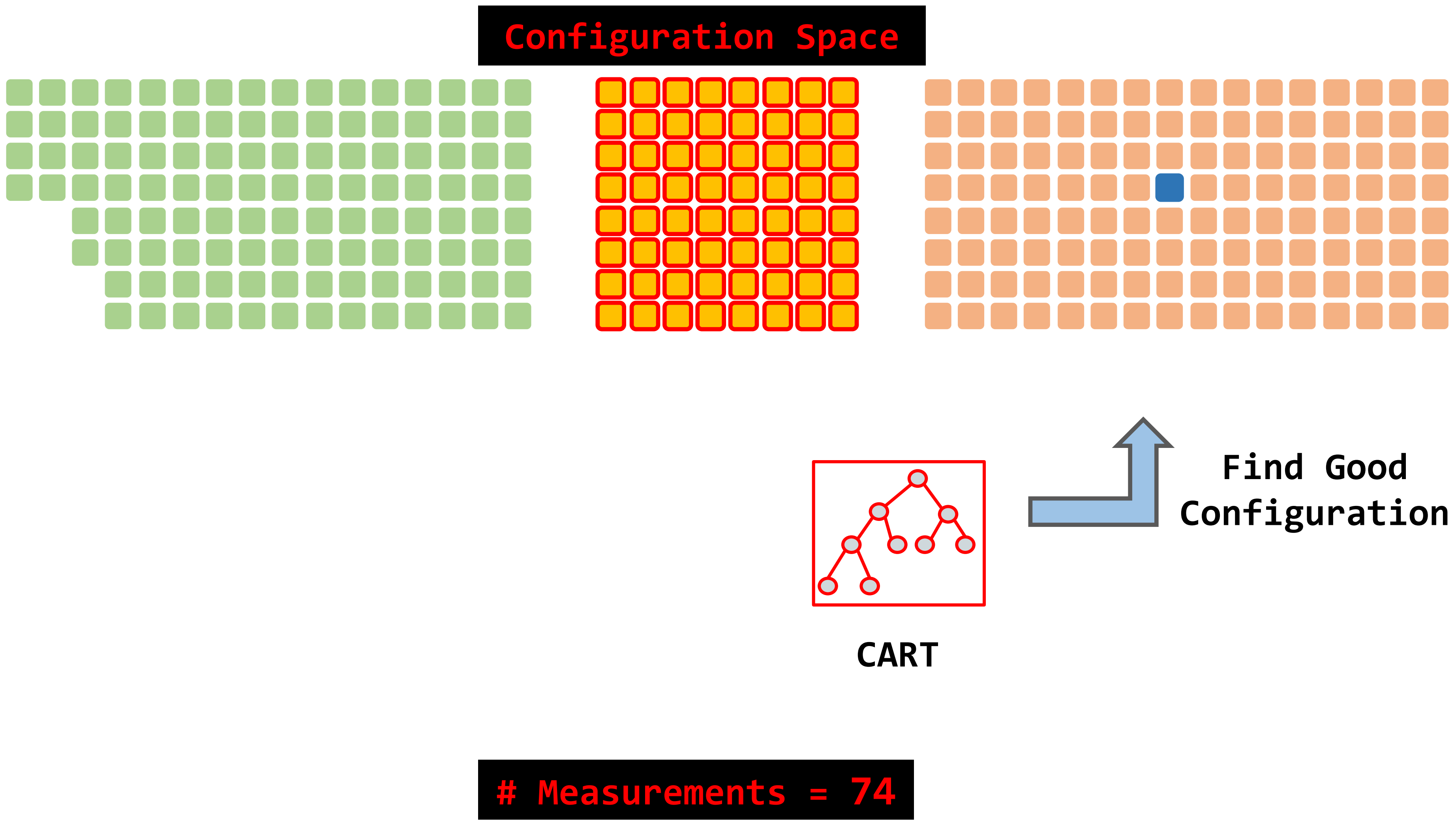
Residual-based Method

Previously…



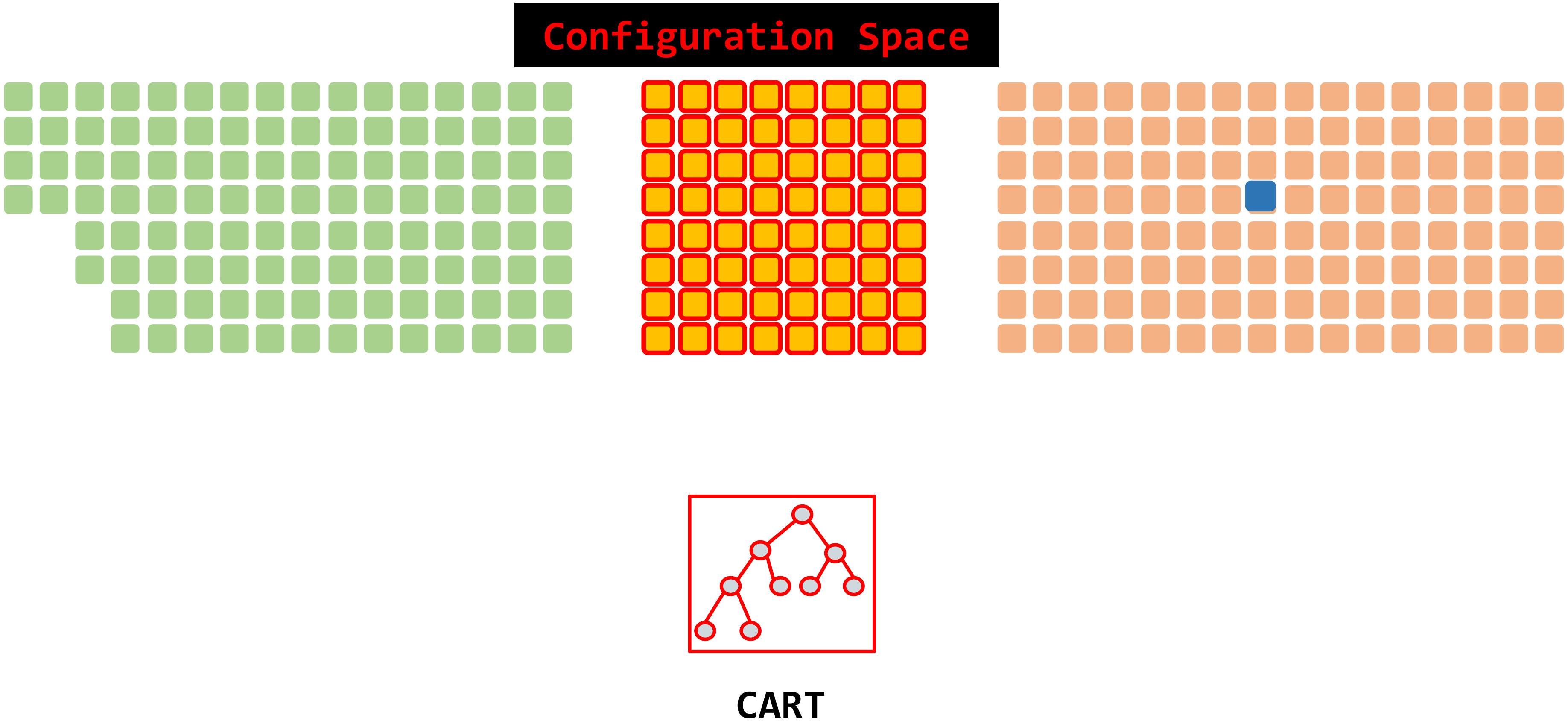
Guo et al.; "Variability-aware performance prediction: A statistical learning approach.”; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t).”; ASE-2015

Previously…



Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

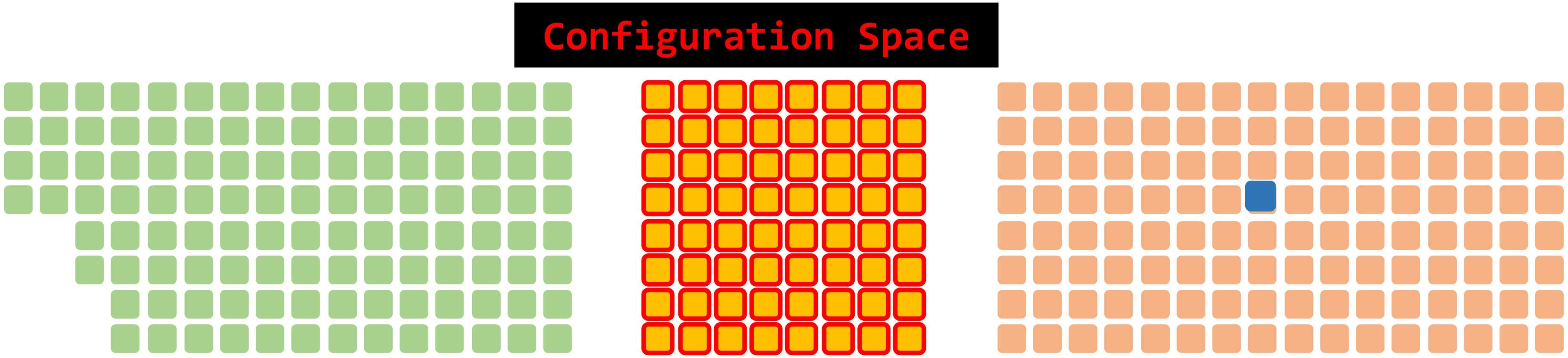
Previously...

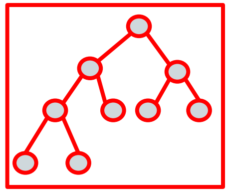


Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously…





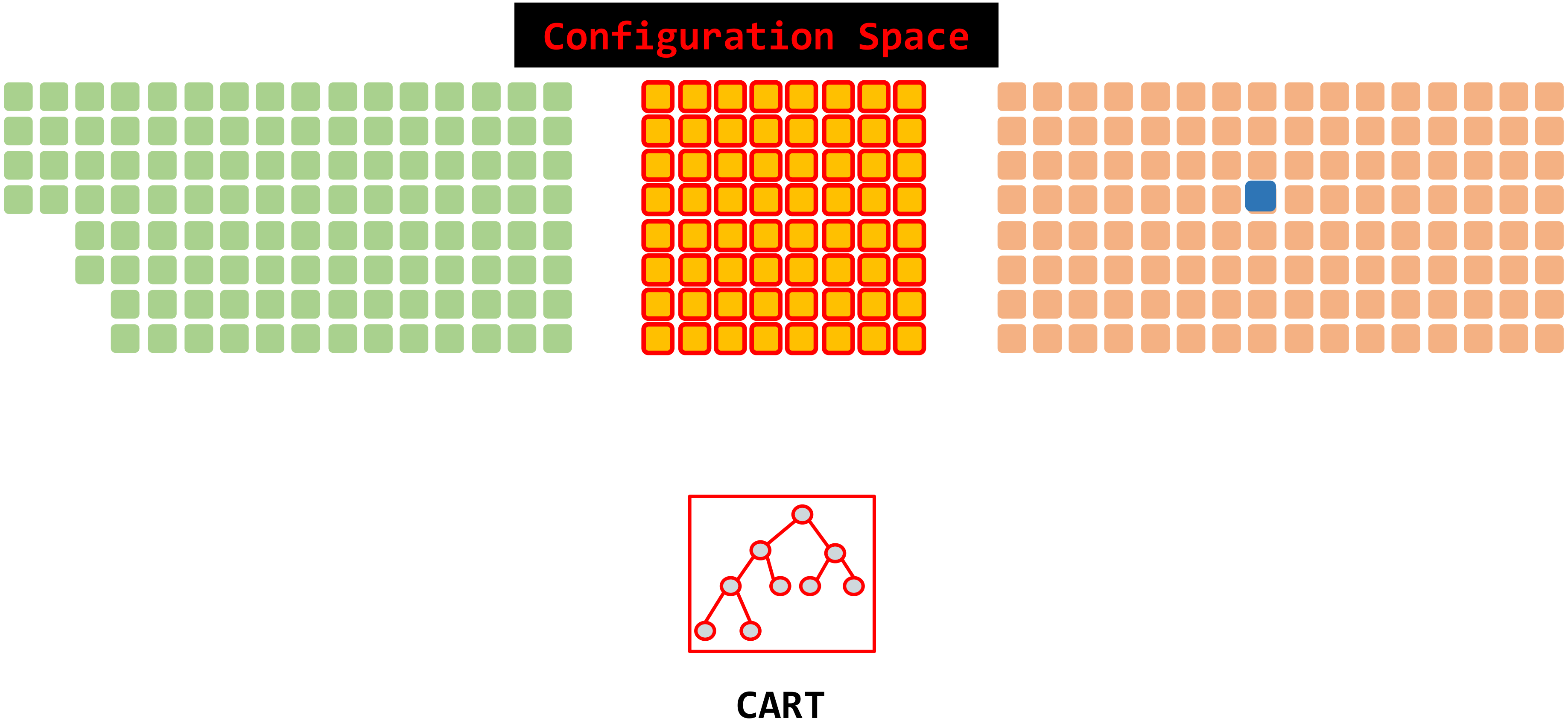
CART

How close is the predicted optimal from actual optimal?

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously...



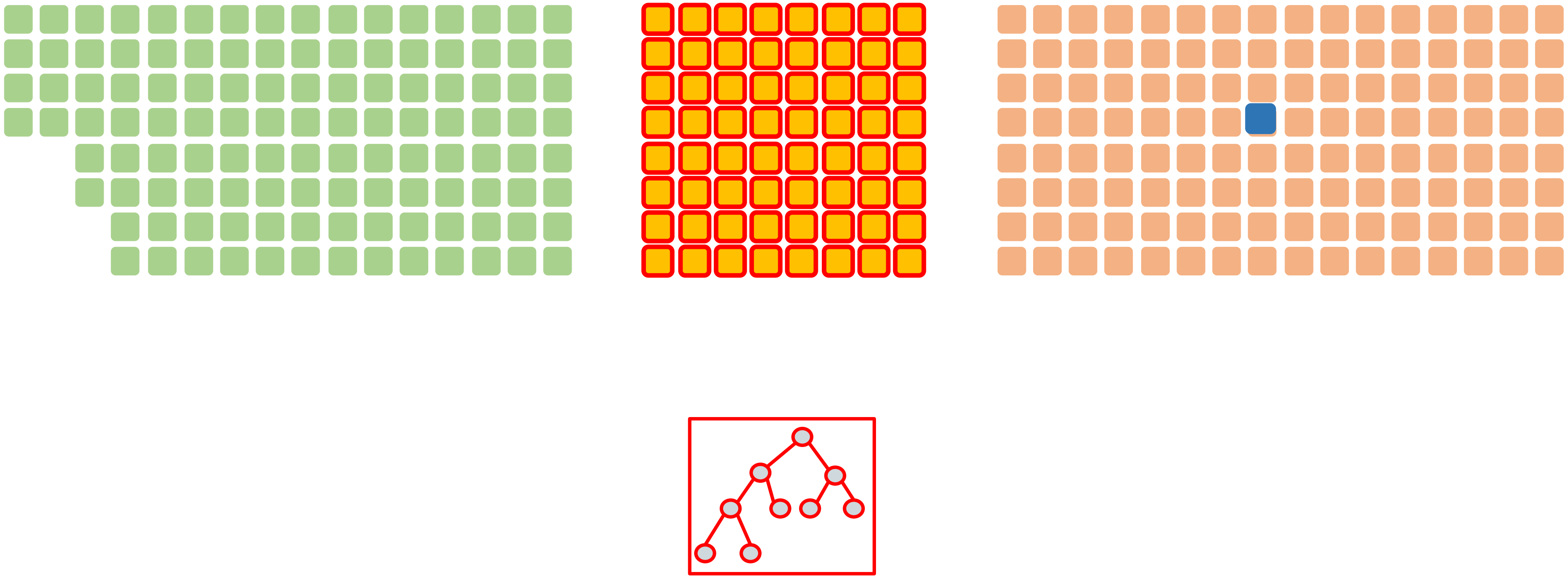
Quality How close is the predicted optimal from actual optimal?

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Residual-based Method

Previously...

Configuration Space



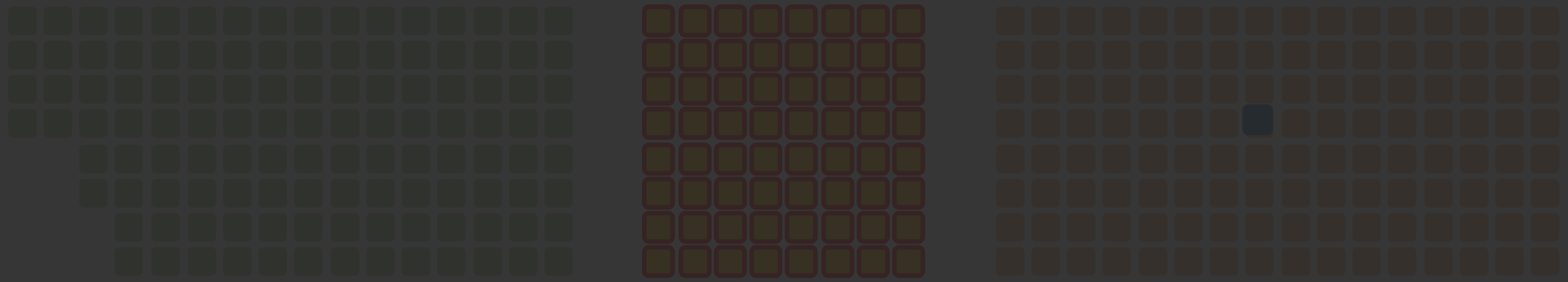
CART

Quality How close is the predicted optimal from actual optimal?

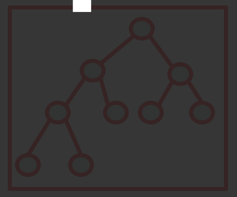
Cost $???

Previously...

Configuration Space



Expensive



Regression Tree

Quality How close is the predicted optimal from actual optimal?

Cost ???

Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

"..in real world scenarios, the cost of acquiring the optimal configuration is overly expensive and time consuming.."

- Gary M Weiss and Ye Tian

Gary M Weiss and Ye Tian. Maximizing classifier utility when there are data acquisition and modeling costs. Data Mining and Knowledge Discovery, 17(2):253–282, 2008.

"..in real world scenarios, the cost of acquiring the optimal configuration is overly expensive and time consuming.."

- Gary M Weiss and Ye Tian

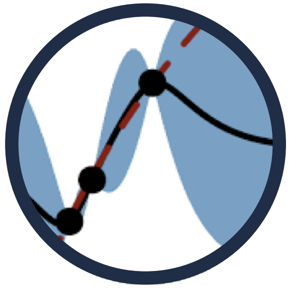
Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

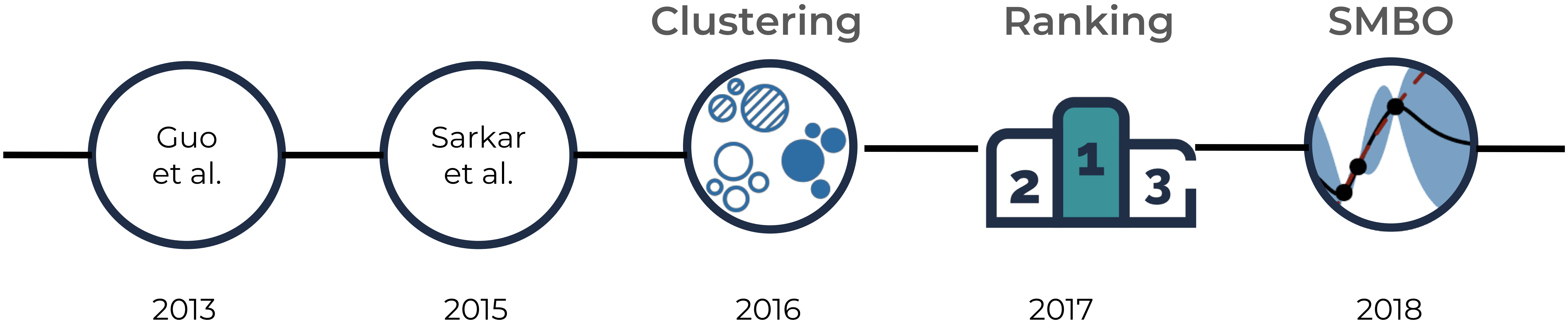
Clustering

Ranking

SMBO

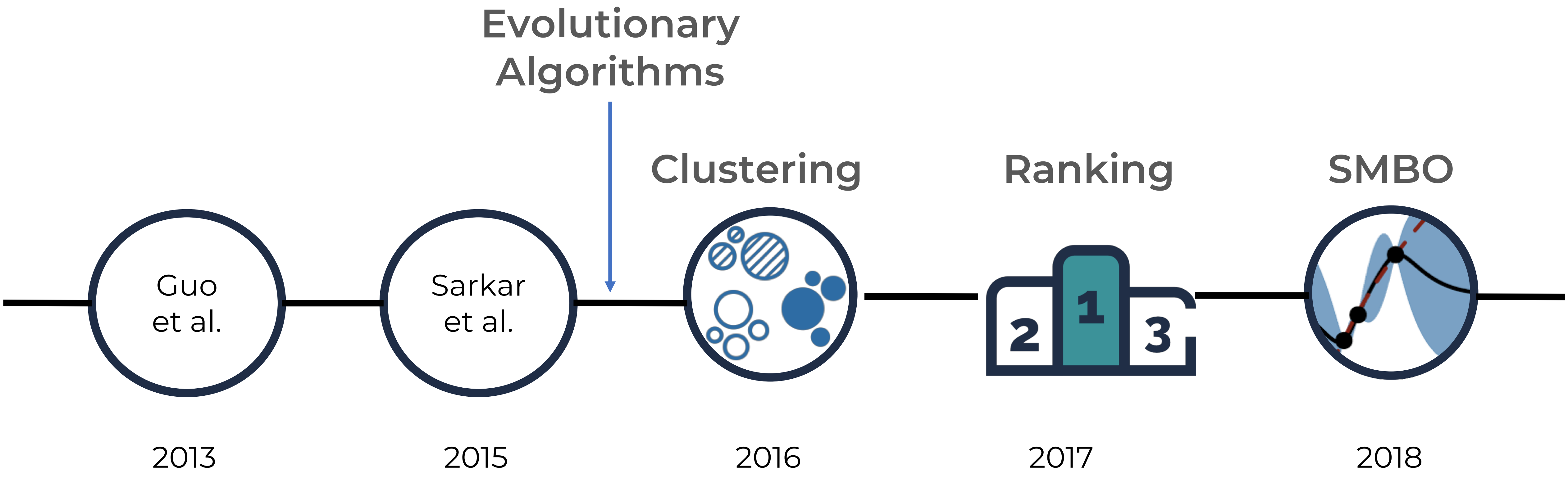


Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.



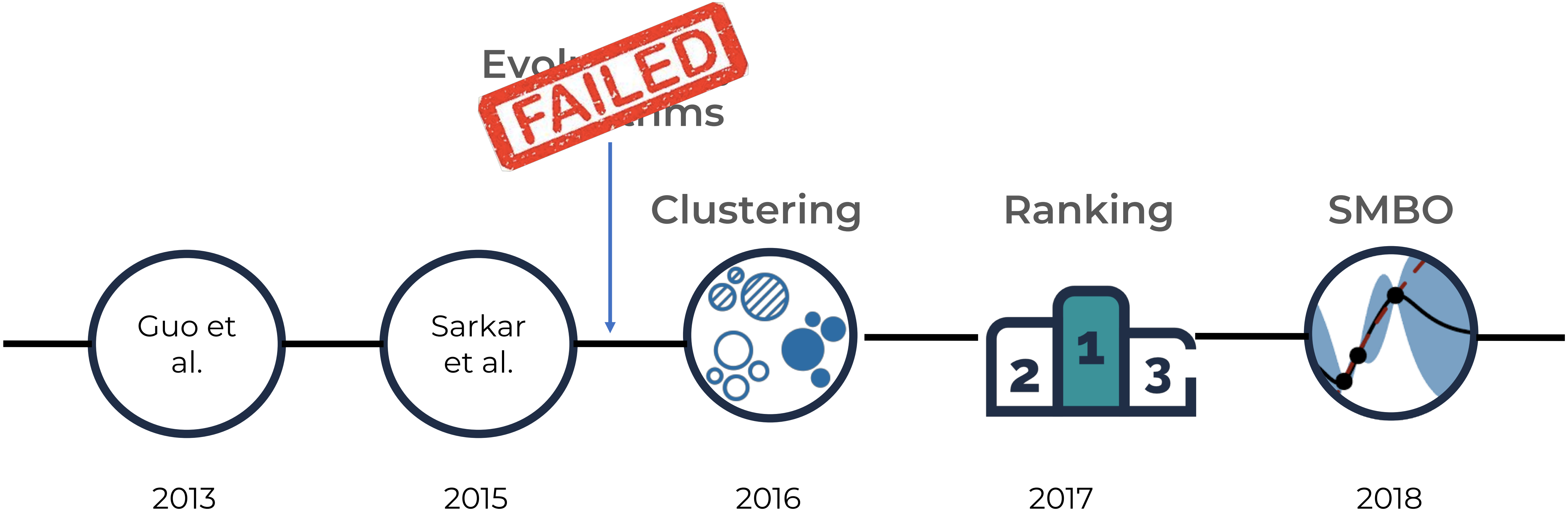
Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.



Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.



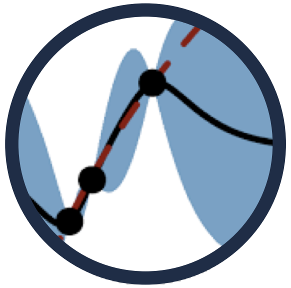
Guo et al.; "Variability-aware performance prediction: A statistical learning approach."; ASE-2013
Sarkar et al.;"Cost-efficient sampling for performance prediction of configurable systems (t)."; ASE-2015

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

Ranking

SMBO

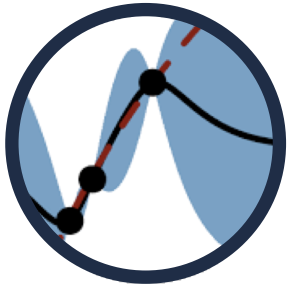


Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

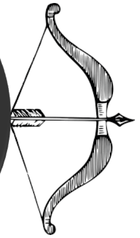
Ranking

SMBO



Cloud Computing

Arrow



ICDCS'18

Scout

*

Mickey

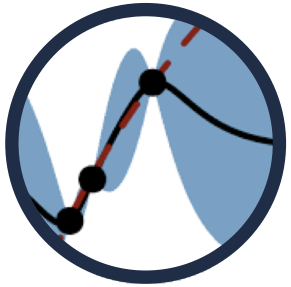
CLOUD'18

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

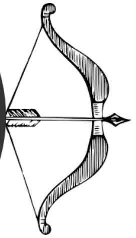
Ranking

SMBO



Cloud Computing

Arrow



ICDCS'18

Scout

*

Mickey

CLOUD'18

Transfer Learning

BEETLE

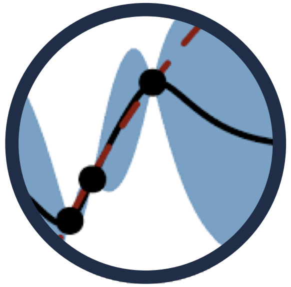
*

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

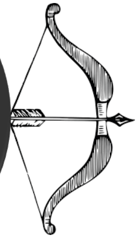
Ranking

SMBO



Cloud Computing

Arrow | Scout | Mickey



ICDCS'18 | * | CLOUD'18

Effort Estimation

ROME

*

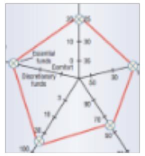
Transfer Learning

BEETLE

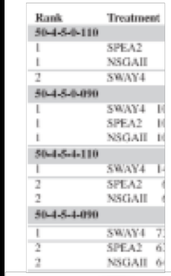
*

Have you published?

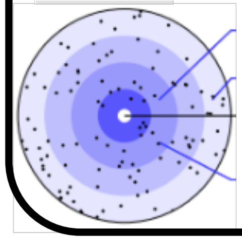
General Optimization



SSBSE’16



IST’16



TSE’18

Performance Optimization

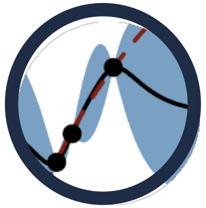
Clustering

Ranking

SMBO

ASEJ’17

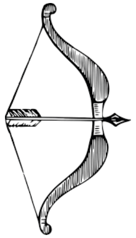
FSE’17



TSE’18

Cloud Computing

Arrow



ICDCS’18

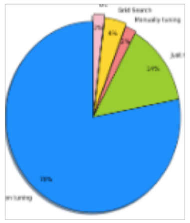
Scout

TPDS’19

Mickey

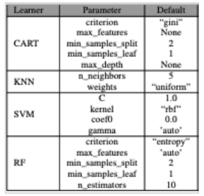
CLOUD’18

Software Analytics



JMSE’19

MSR’18



SWAN’18

Effort Estimation

ROME

EMSE’19

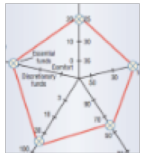
Transfer Learning

BEETLE

TSE’19

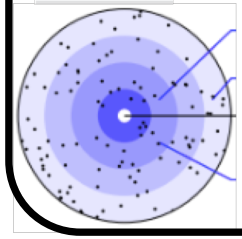
Have you published?

General Optimization



SSBSE'16

IST'16



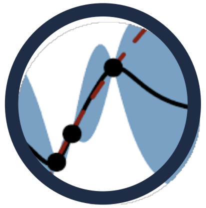
TSE'18

Performance Optimization

Clustering

Ranking

SMBO



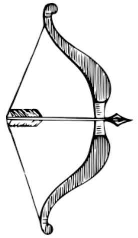
ASEJ'17

FSE'17

TSE'18

Cloud Computing

Arrow



Scout

Mickey

ICDCS'18

TPDS'19

CLOUD'18

Software Analytics

Learner	Parameter	Default
CART	criterion	"gini"
	max_features	None
	min_samples_split	2
	min_samples_leaf	1
	max_depth	None
KNN	n_neighbors	5
	weights	"uniform"
SVM	C	1.0
	kernel	"rbf"
	coef0	0.0
	gamma	'auto'
RF	criterion	"entropy"
	max_features	'auto'
	min_samples_split	2
	min_samples_leaf	1
	n_estimators	10

JMSE'19

MSR'18

SWAN'18

Effort Estimation

ROME

EMSE'19

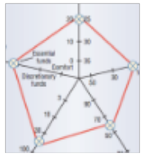
Transfer Learning

BEETLE

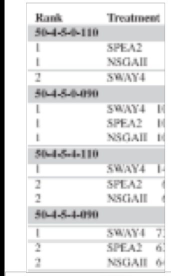
TSE'19

Have you published?

General Optimization

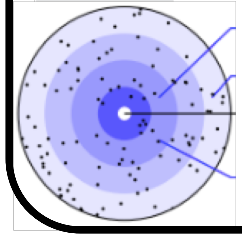


SSBSE'16



Rank	Treatment
50-4-5-0-110	
1	SPEA2
1	NSGAII
2	SWAY4
50-4-5-0-090	
1	SWAY4
1	SPEA2
1	NSGAII
50-4-5-4-110	
1	SWAY4
2	SPEA2
2	NSGAII
50-4-5-4-090	
1	SWAY4
2	SPEA2
2	NSGAII

IST'16



TSE'18

Performance Optimization

Clustering

Ranking

SMBO

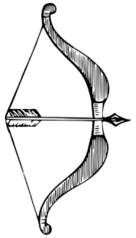
ASEJ'17

FSE'17

TSE'18

Cloud Computing

Arrow



ICDCS'18

Scout

TPDS'19

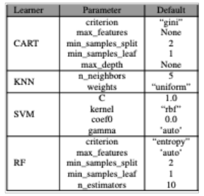
Mickey

CLOUD'18

Software Analytics

JMSE'19

MSR'18



Learner	Parameter	Default
CART	criterion	"gini"
	max_features	None
	min_samples_split	2
	min_samples_leaf	1
	max_depth	None
KNN	n_neighbors	5
	weights	"uniform"
SVM	C	1.0
	kernel	"rbf"
	coef0	0.0
	gamma	'auto'
RF	criterion	"entropy"
	max_features	'auto'
	min_samples_split	2
	min_samples_leaf	1
	n_estimators	10

SWAN'18

Effort Estimation

ROME

EMSE'19

Transfer Learning

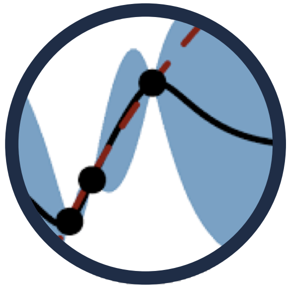
BEETLE

TSE'19

Clustering

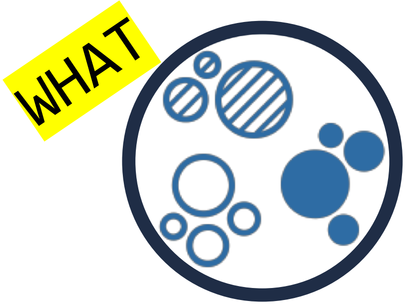
Ranking

SMBO



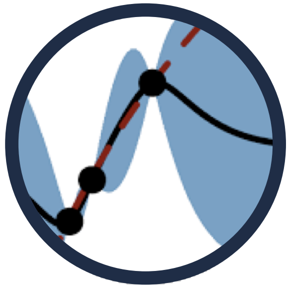
Presented during Written Prelims

Clustering



Ranking

SMBO

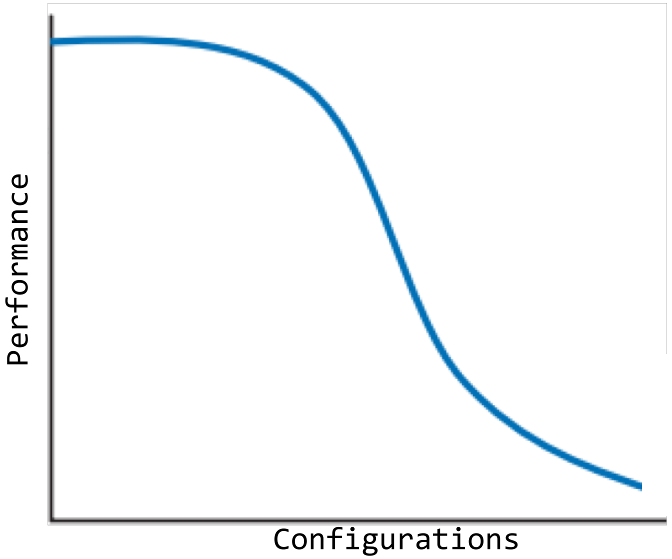


Nair et al.; Faster discovery of faster system configurations with spectral learning; ASEJ (2016)

Intuition

Intuition

Expectation

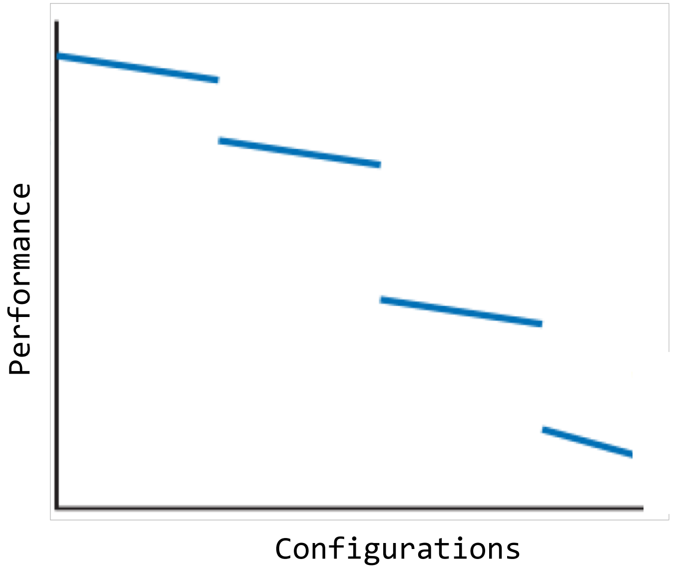


Oh, J., Batory, D., Myers, M., & Siegmund, N. (2017, August). Finding near-optimal configurations in product lines by random sampling. *Foundations of Software Engineering*(pp. 61-71). ACM.

WHAT (Clustering)

Intuition

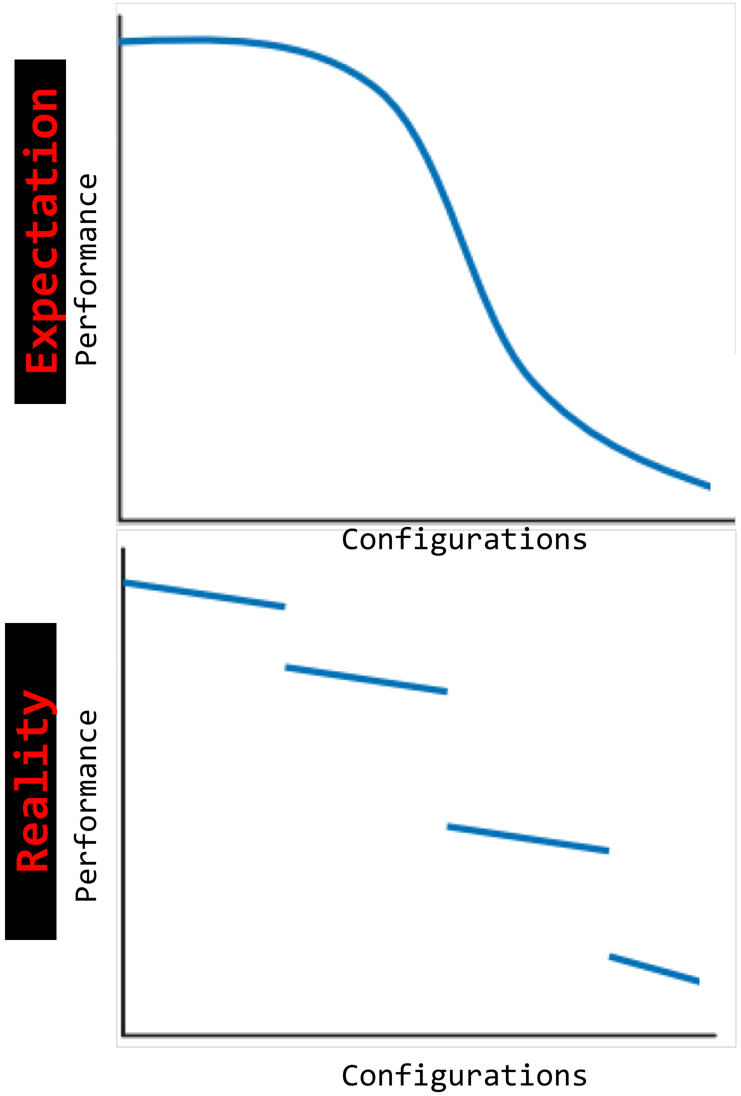
Reality



Oh, J., Batory, D., Myers, M., & Siegmund, N. (2017, August). Finding near-optimal configurations in product lines by random sampling. *Foundations of Software Engineering*(pp. 61-71). ACM.

WHAT (Clustering)

Intuition

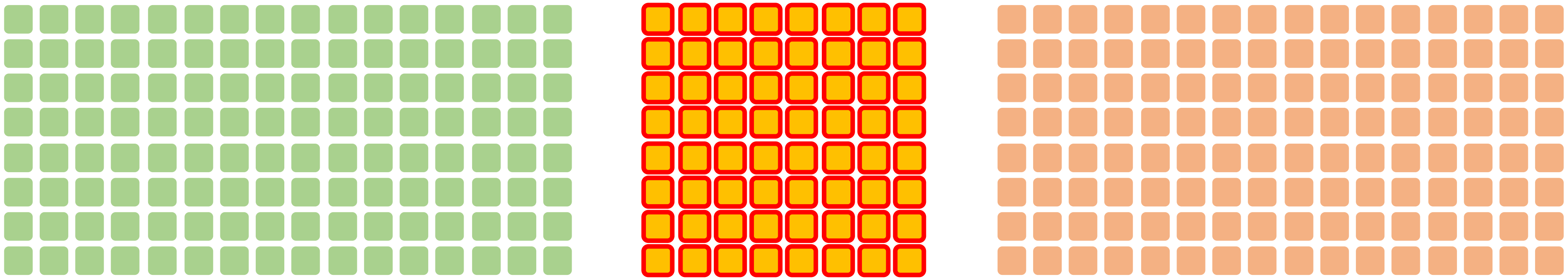


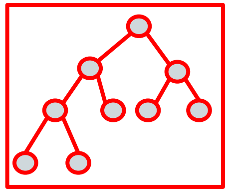
- Most of the configuration options does not affect the performance
- First Cluster then Sample

Oh, J., Batory, D., Myers, M., & Siegmund, N. (2017, August). Finding near-optimal configurations in product lines by random sampling. *Foundations of Software Engineering*(pp. 61-71). ACM.

WHAT (Clustering)

Configuration Space



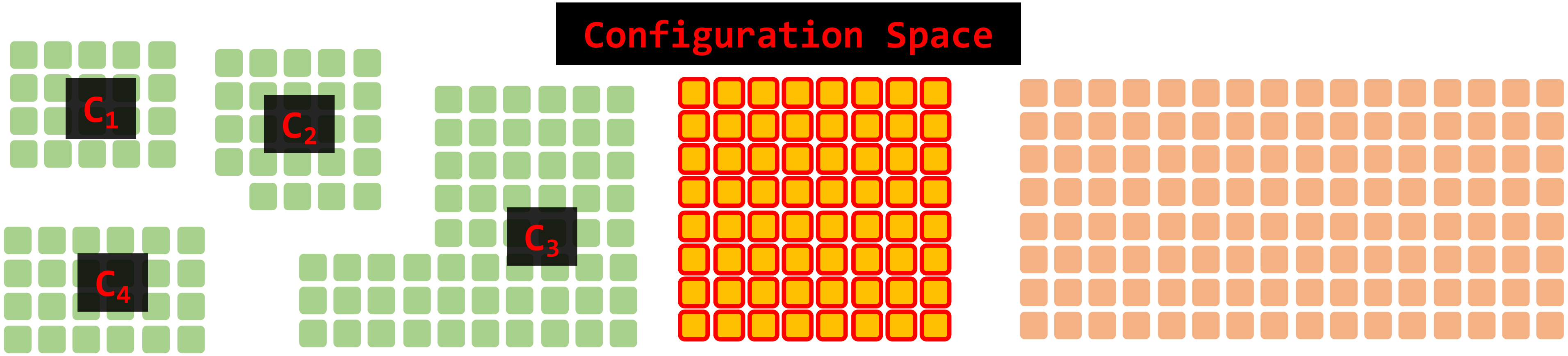


CART

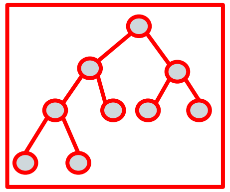
Measurements = 64

WHAT (Clustering)

WHAT



Clustering

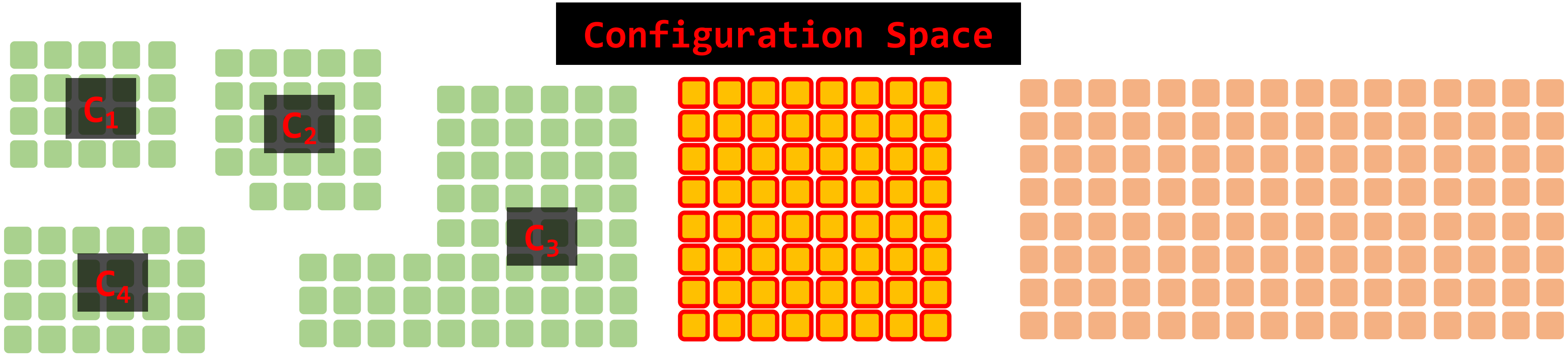


CART

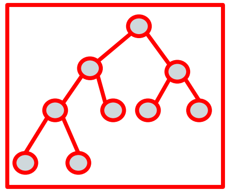
\# Measurements = 64

WHAT (Clustering)

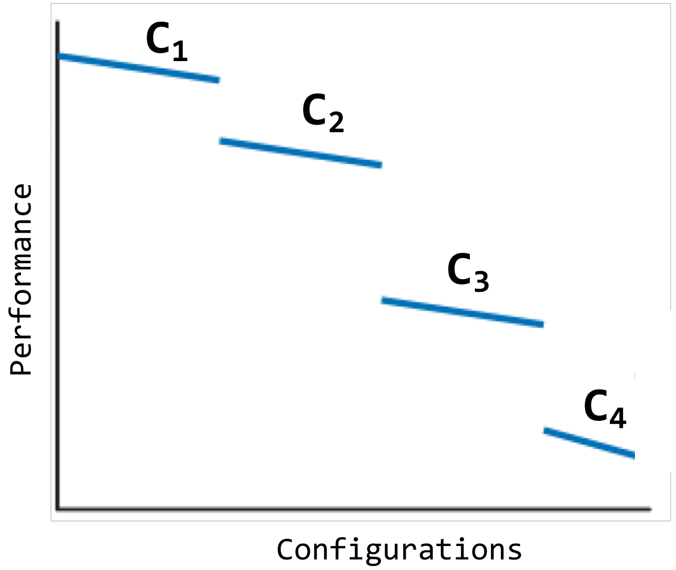
WHAT



Clustering



CART

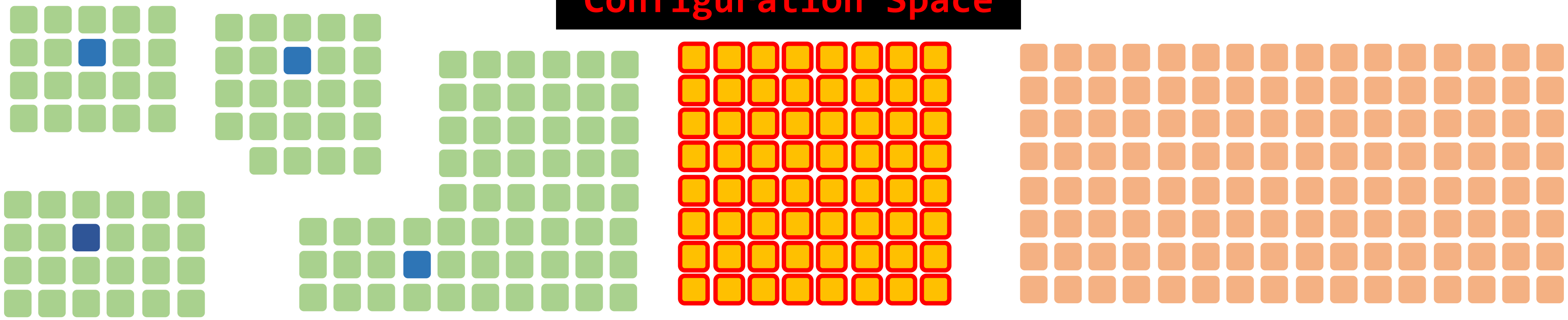


Measurements = 64

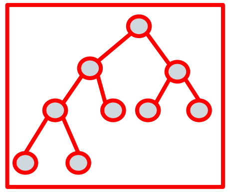
WHAT (Clustering)

WHAT

Configuration Space



Clustering + Sampling Policies

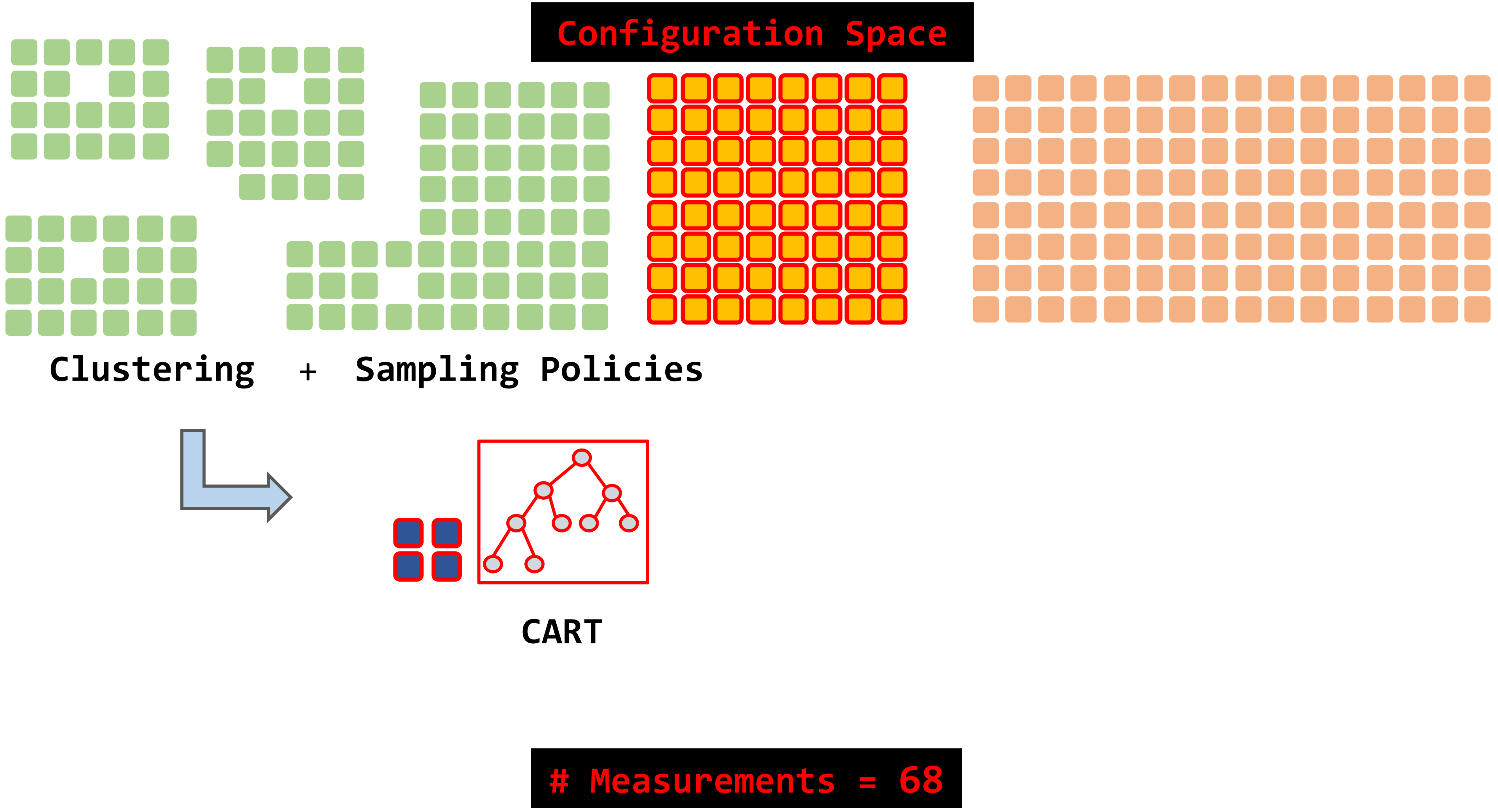


CART

Measurements = 64

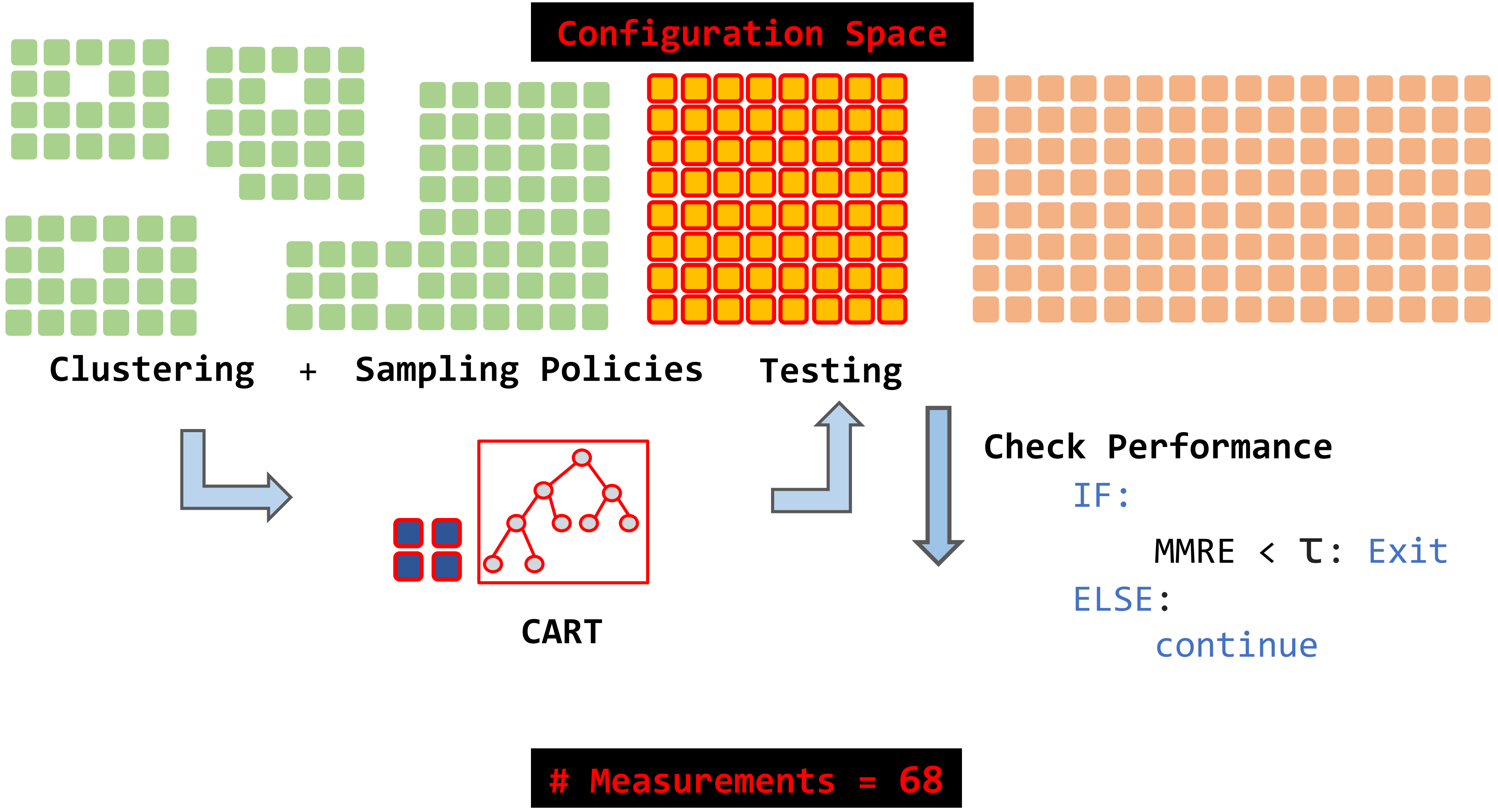
WHAT (Clustering)

WHAT



WHAT (Clustering)

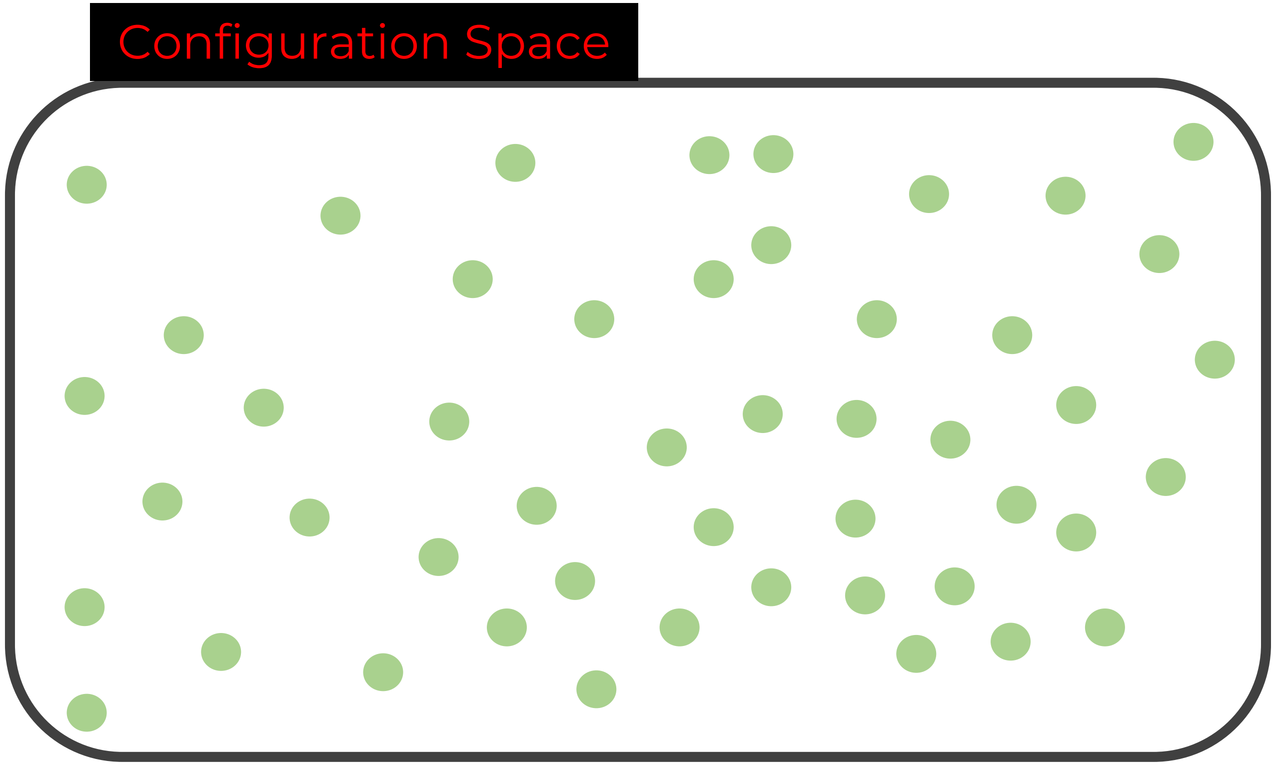
Previously?



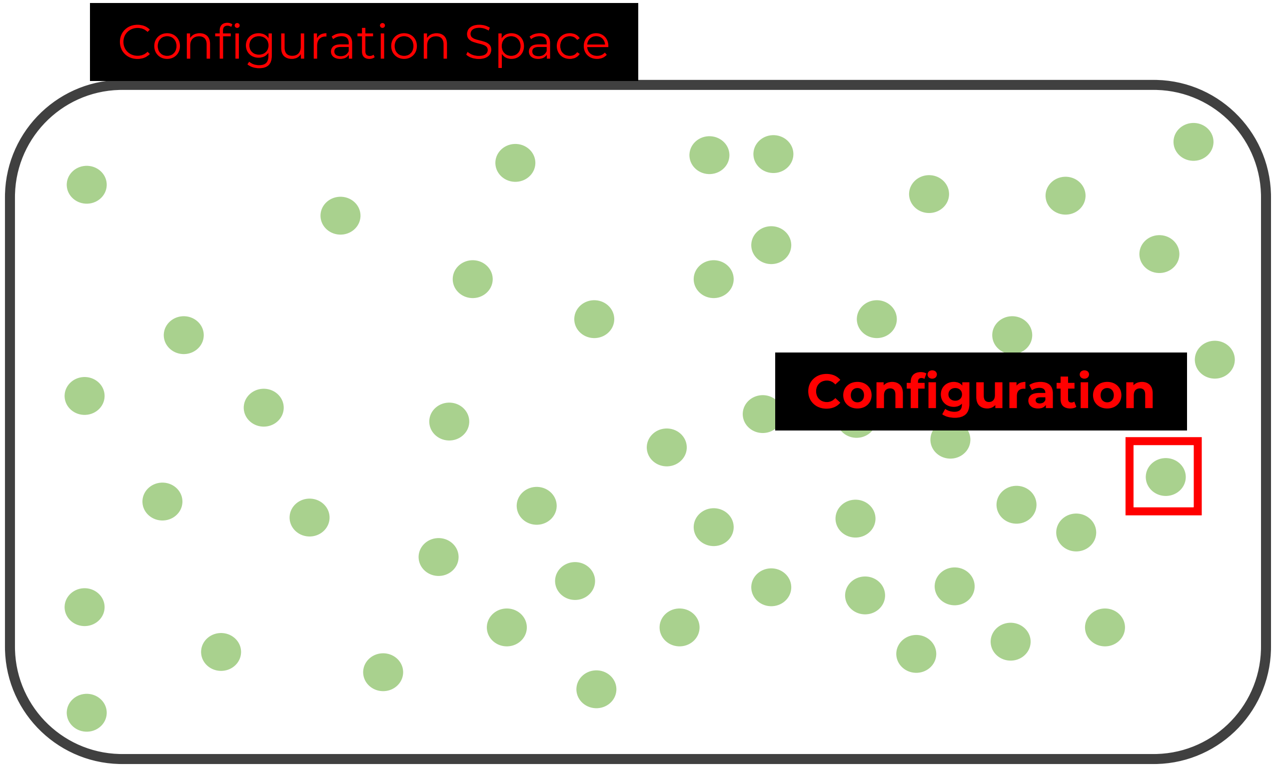
How to Cluster?

Configuration Space

How to Cluster?

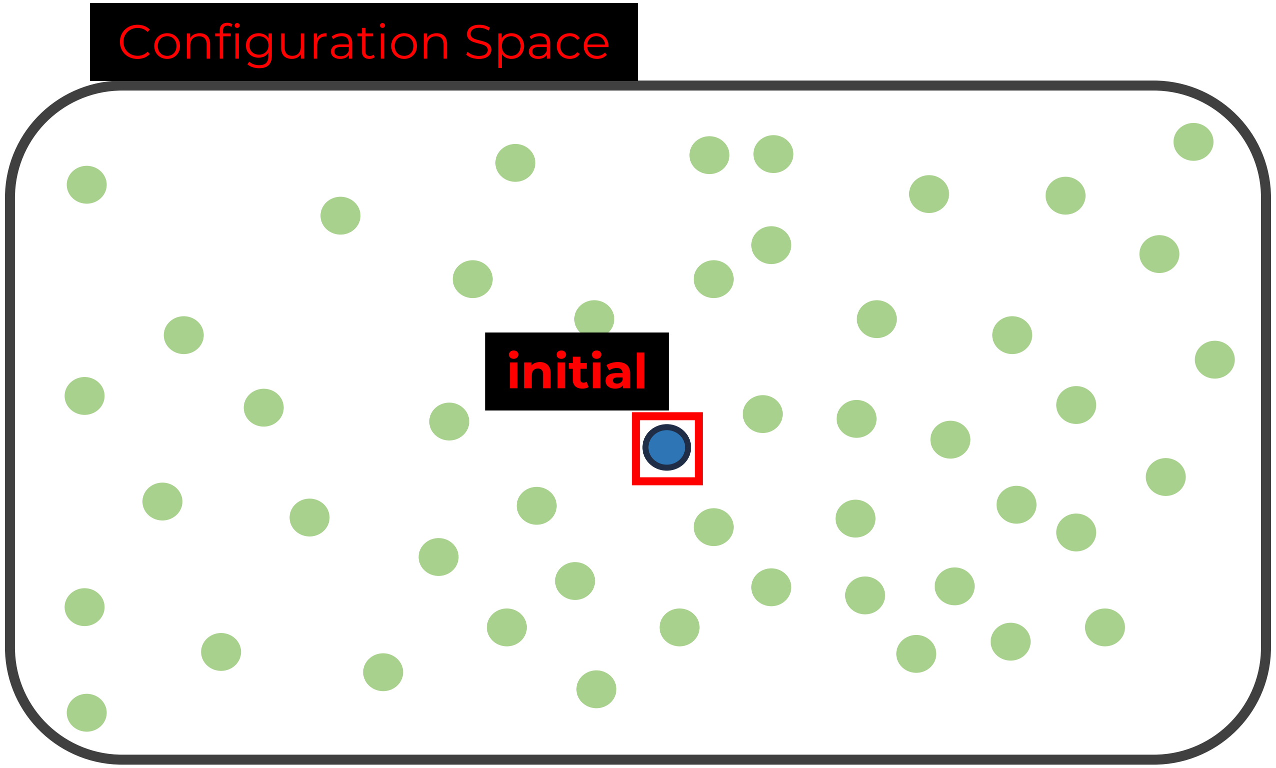


How to Cluster?



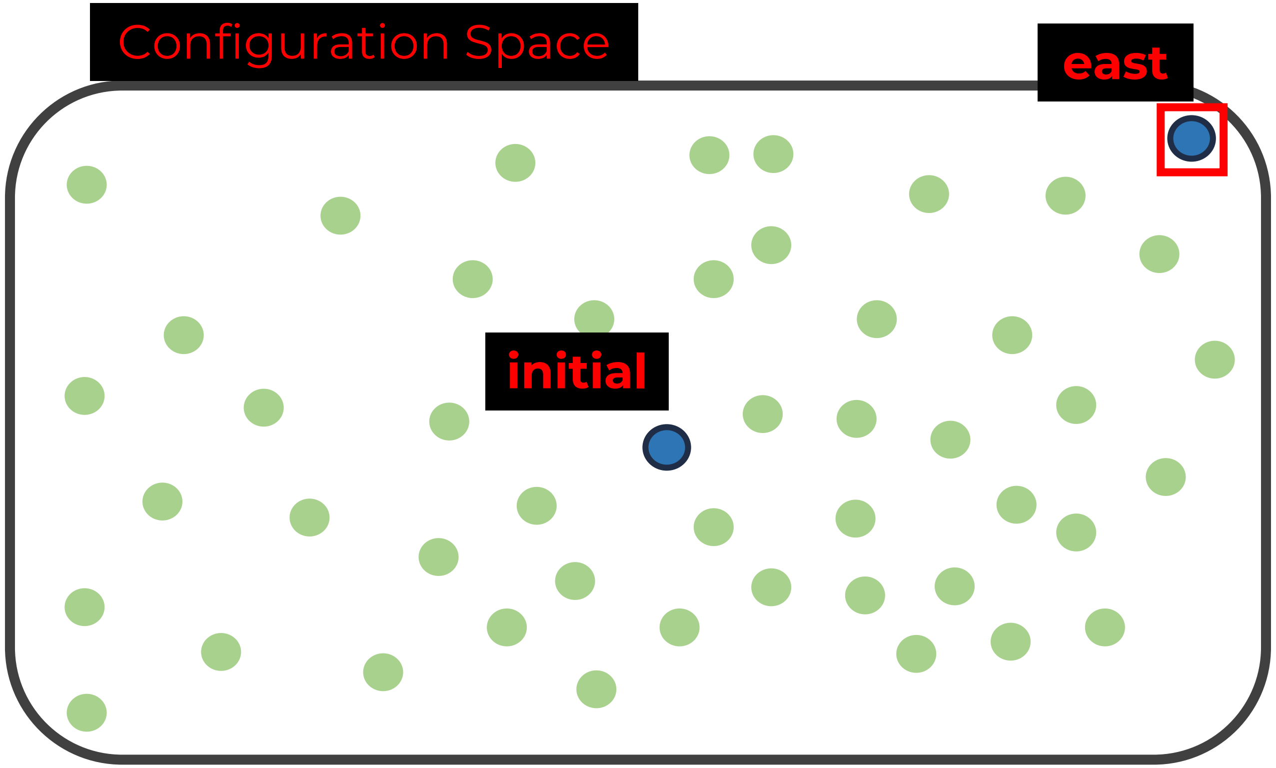
How to Cluster?

1. Select random configuration (initial)



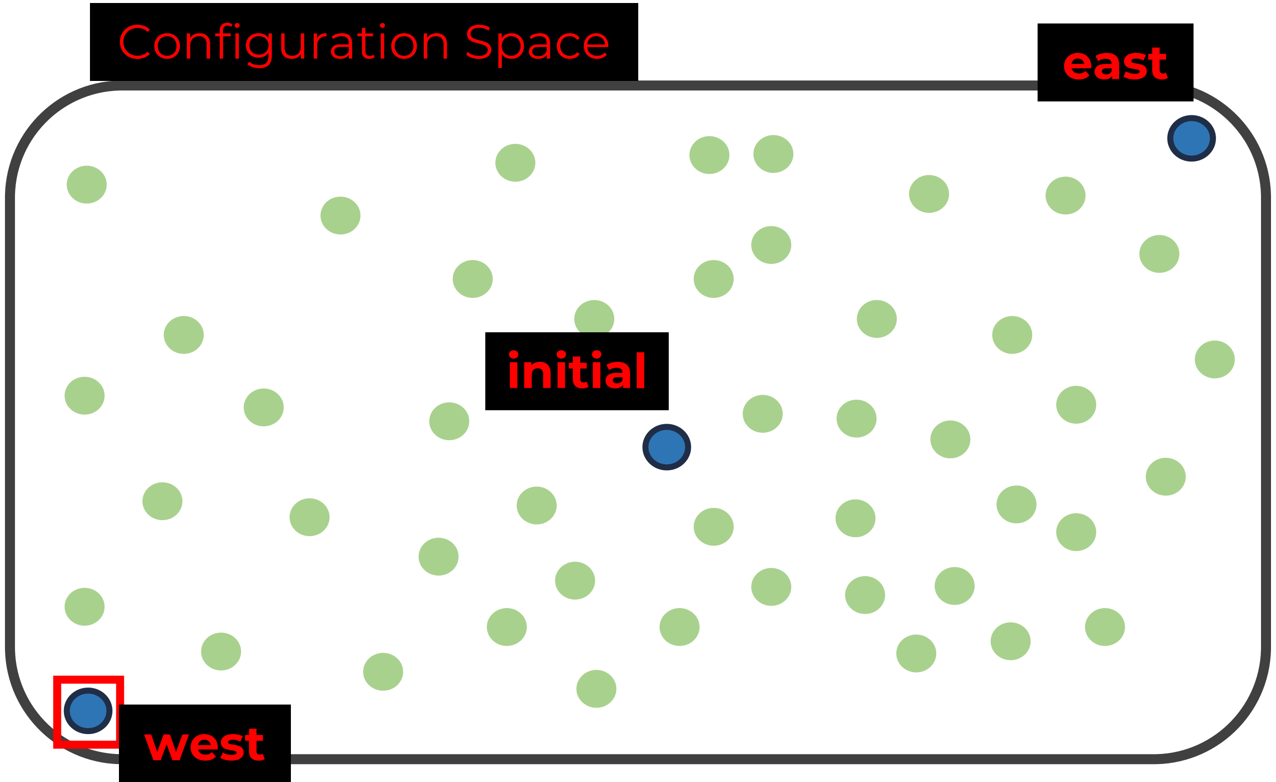
How to Cluster?

1. Select random configuration (initial)
2. Find furthest point (east)



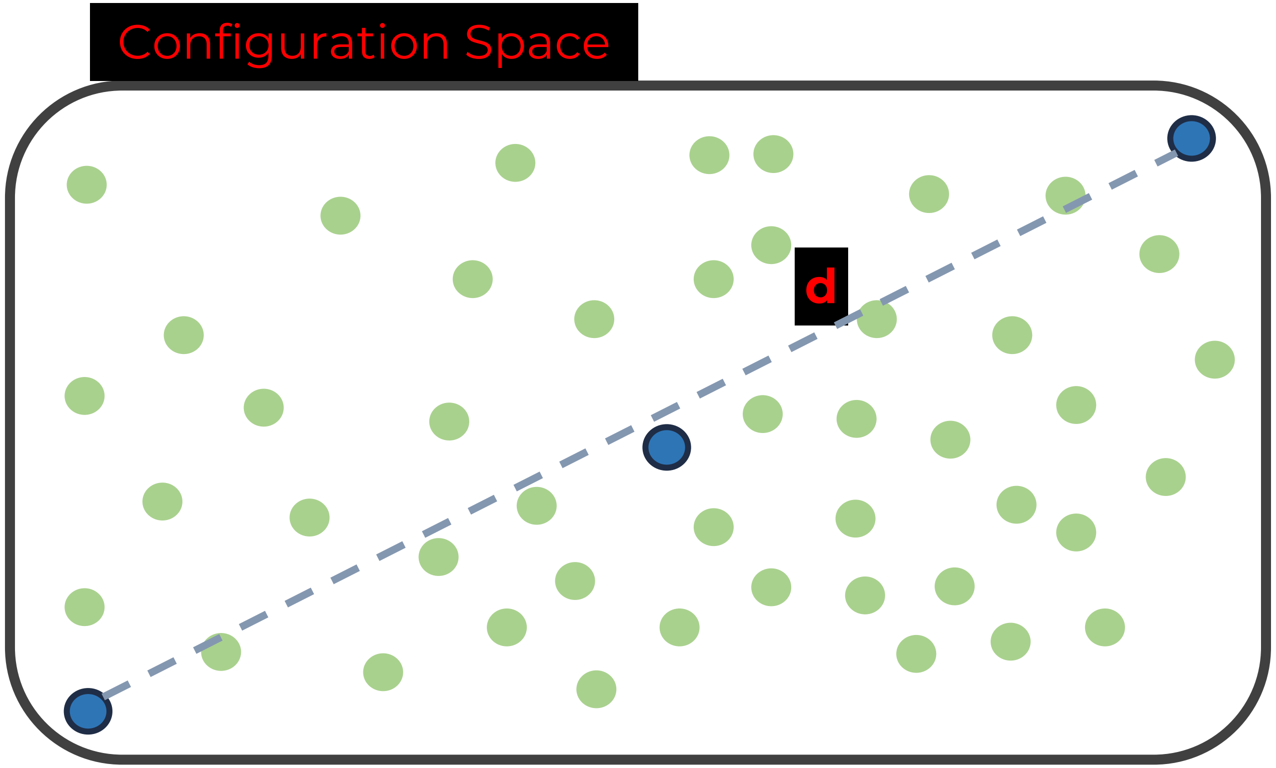
How to Cluster?

1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)



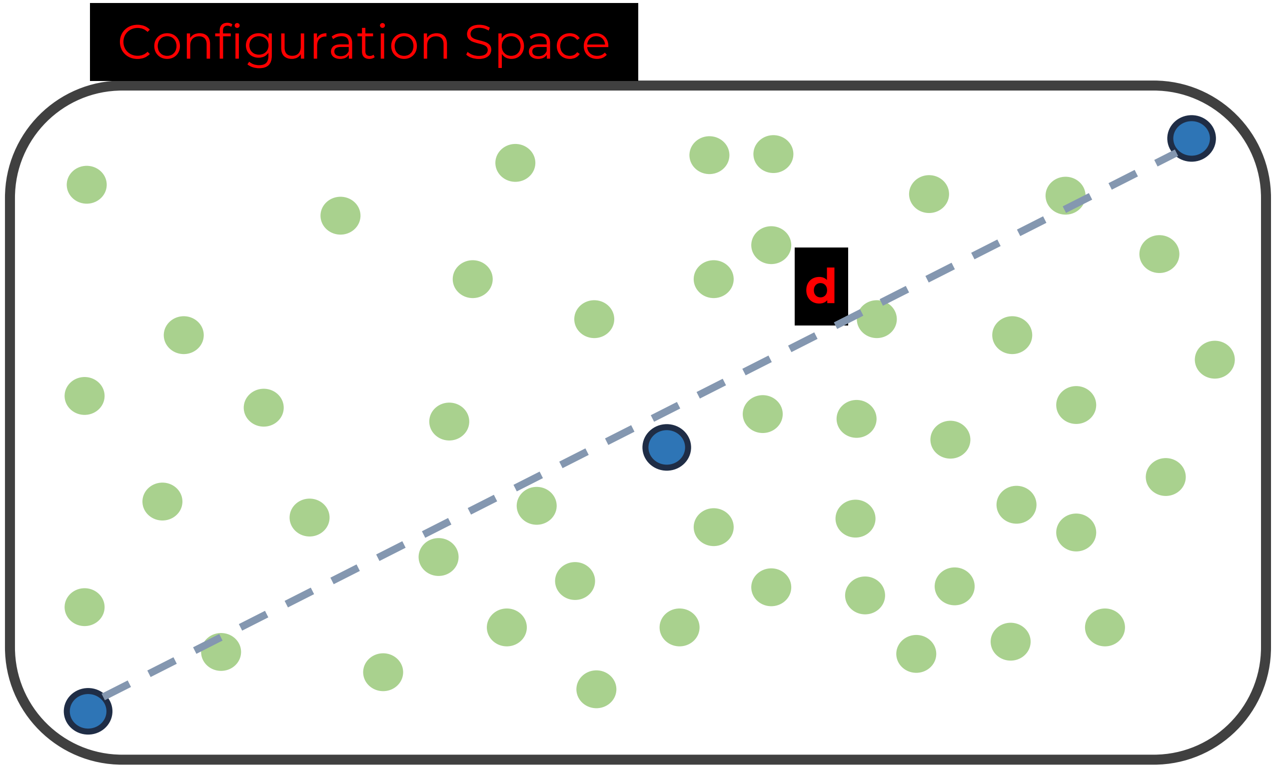
How to Cluster?

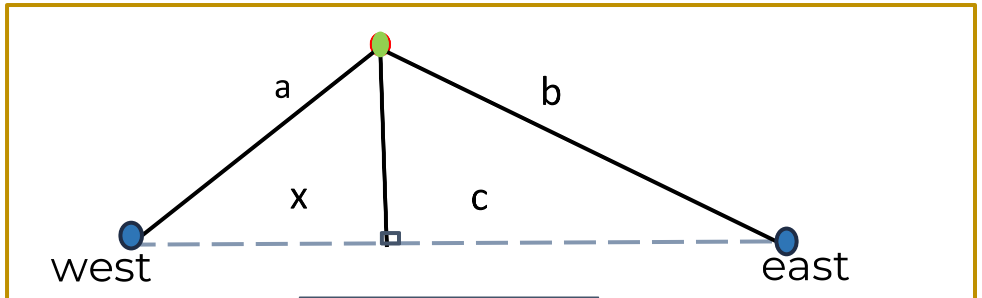
1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)



How to Cluster?

1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d

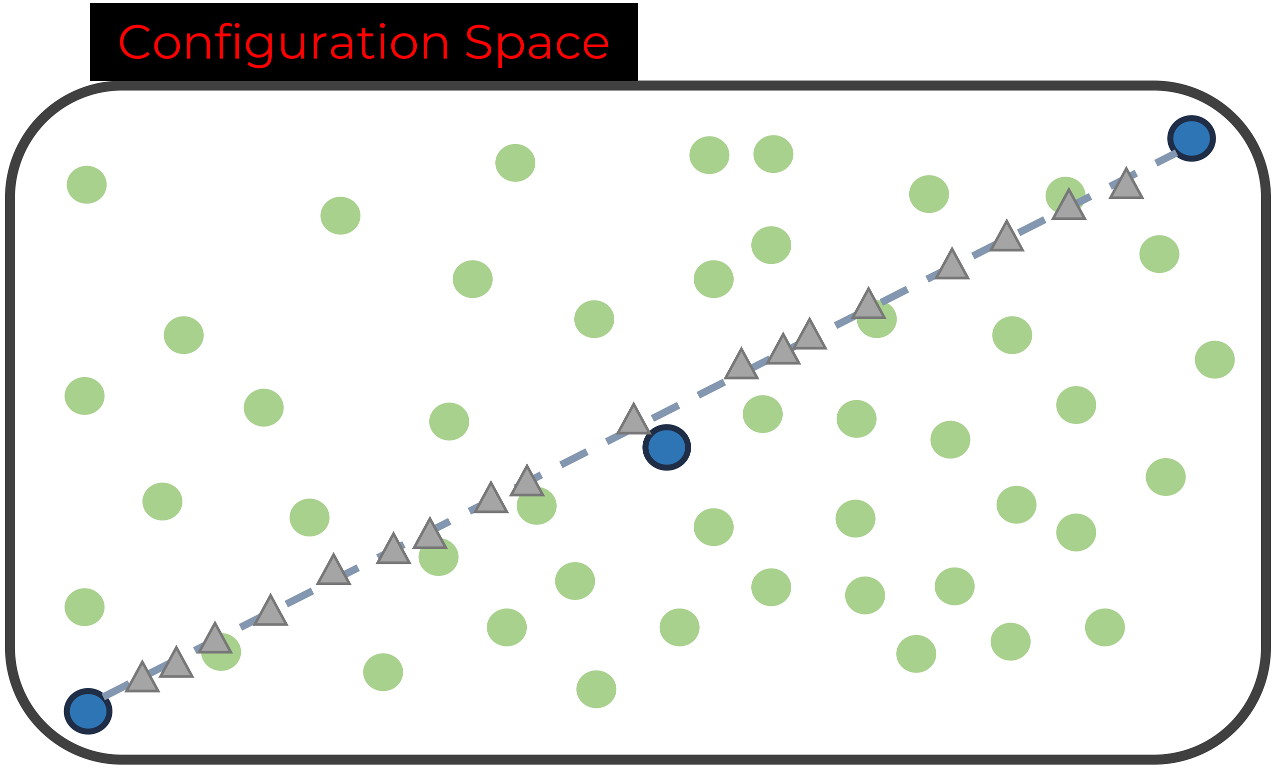




$$x = \frac{a^2 + c^2 - b^2}{2c}$$

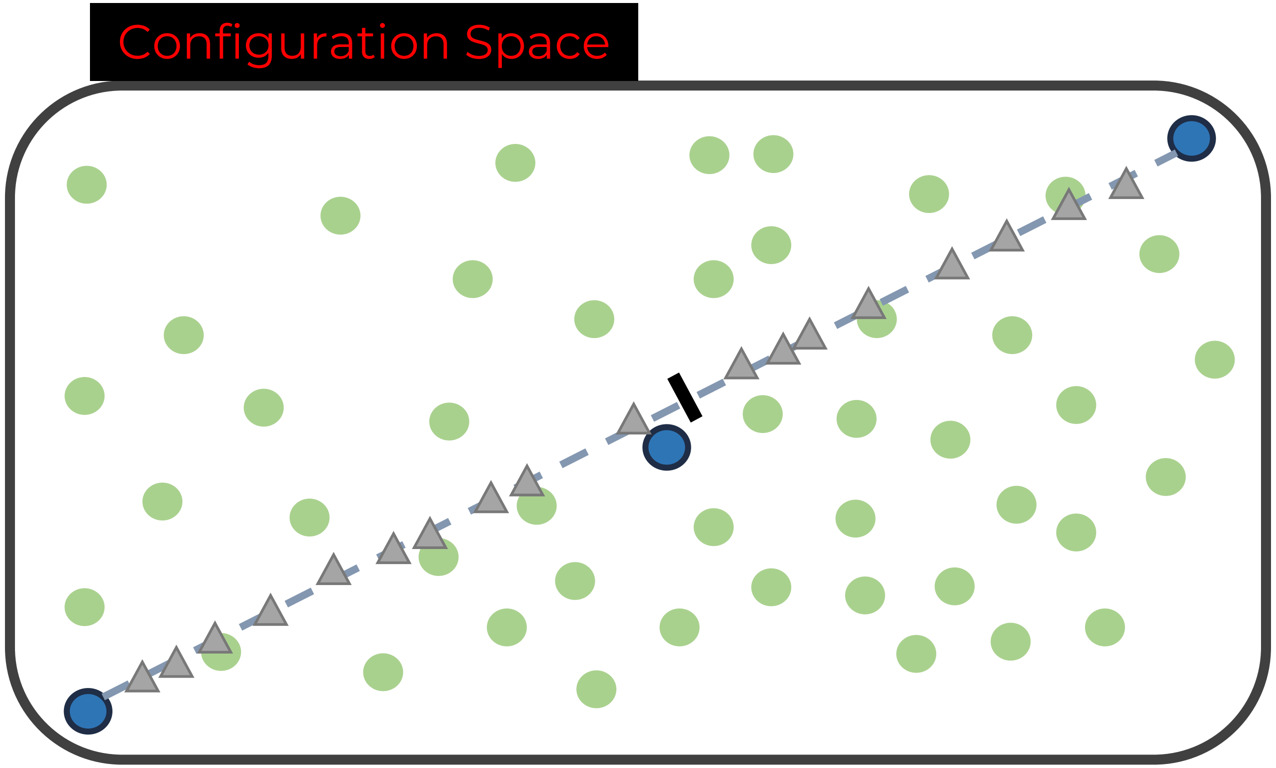
How to Cluster?

1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d



How to Cluster?

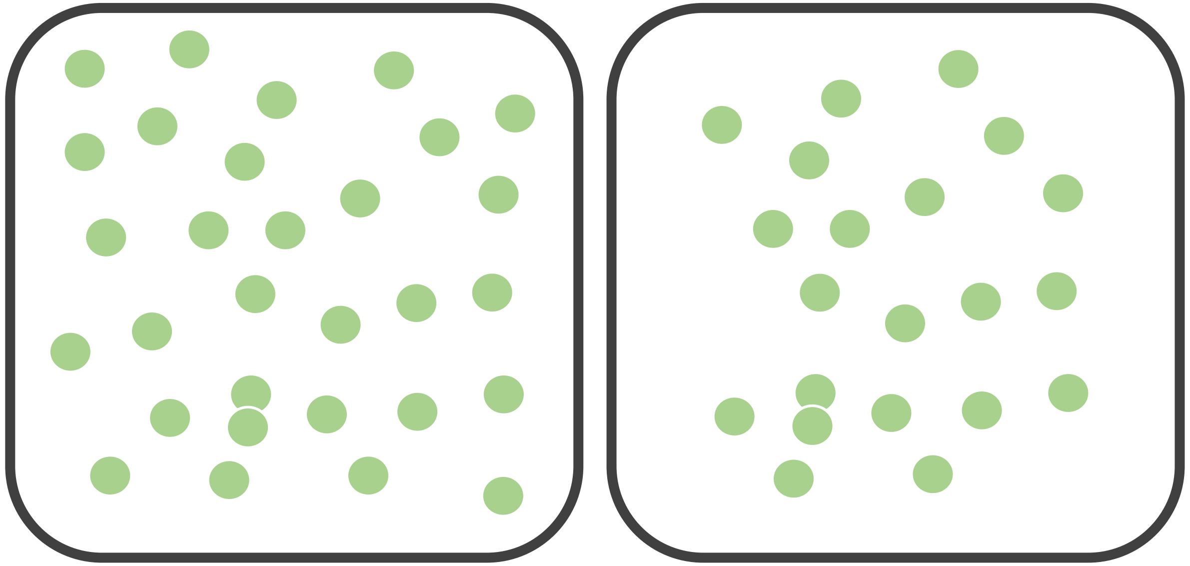
1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d
6. Split at median of d



How to Cluster?

1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d
6. Split at median of d

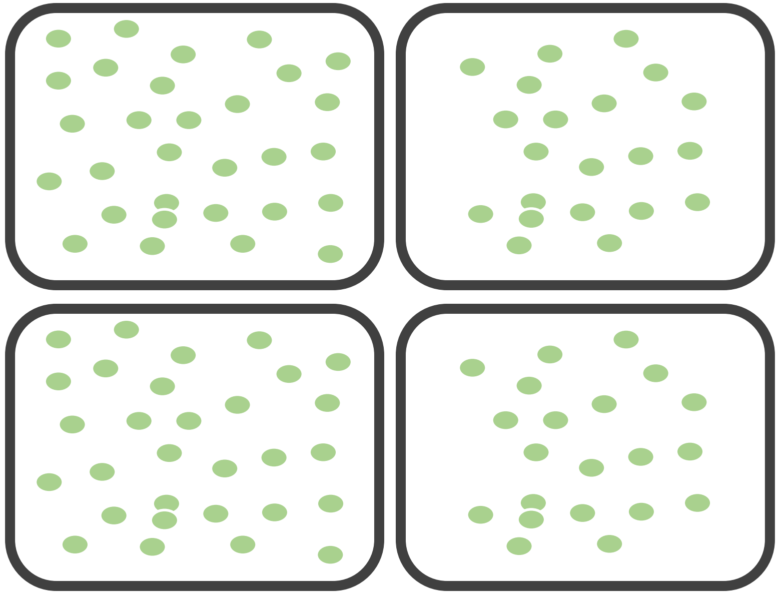
Configuration Space



How to Cluster?

1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d
6. Split at median of d
7. Recurse

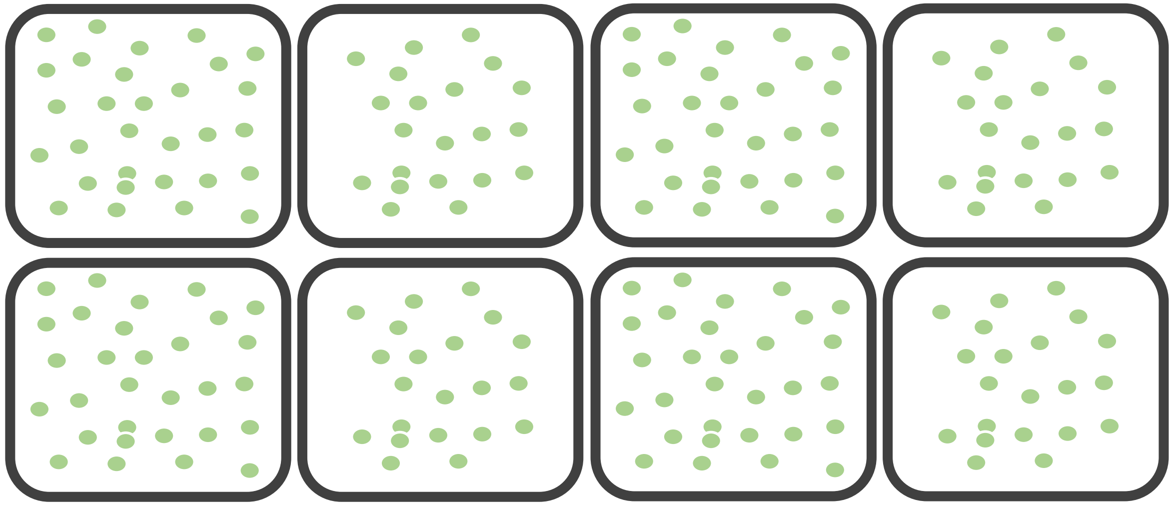
Configuration Space



How to Cluster?

1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d
6. Split at median of d
7. Recurse
8. Stop when **|n| < sqrt(N)**

Configuration Space



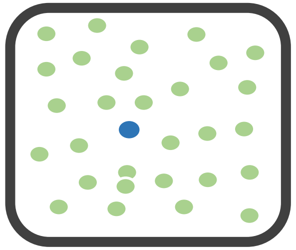
WHAT (Clustering)

How to Cluster?

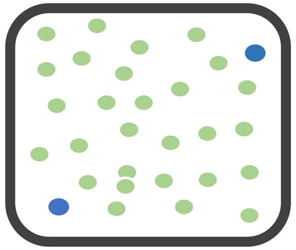
1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d
6. Split at median of d
7. Recurse
8. Stop when |n| < sqrt(N)

Sampling Policies

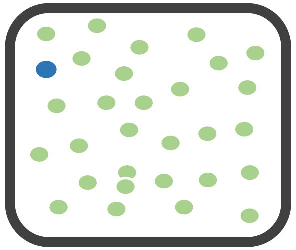
- Random
  - Choose a candidate at random
  - Number of evaluations/Cluster = 1
  - Point selected/Cluster = 1



- East-West
  - Choose extreme points in dimension of maximum variance
  - Number of evaluations/Cluster = 2
  - Point selected/Cluster = 2



- Exemplar
  - Choose the best candidate from the cluster
  - Number of evaluations/Cluster = n
  - Point selected/Cluster = 1



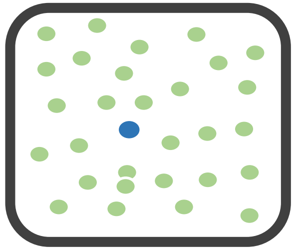
WHAT (Clustering)

How to Cluster?

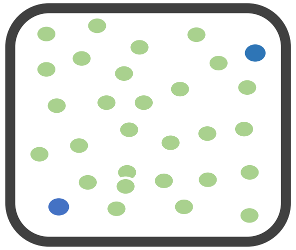
1. Select random configuration (initial)
2. Find furthest point (east)
3. Find furthest point from east (west)
4. Connect east and west (d)
5. Projects configurations to d
   a) For all points
      - ❑ Choose a point (candidate)
      - ❑ Calculate position on d
6. Split at median of d
7. Recurse
8. Stop when |n| < sqrt(N)

Sampling Policies

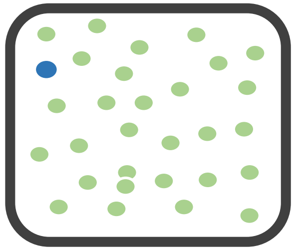
- Random
  - Choose a candidate at random
  - Number of evaluations/Cluster = 1
  - Point selected/Cluster = 1



- East-West
  - Choose extreme points in dimension of maximum variance
  - Number of evaluations/Cluster = 2
  - Point selected/Cluster = 2



- Exemplar
  - Choose the best candidate from the cluster
  - Number of evaluations/Cluster = n
  - Point selected/Cluster = 1



Summary

- WHAT can generate good predictions using only a small number of configurations
- WHAT can build “good” models which can be used in optimizers
- WHAT is comparable to the state of the art predictors

Summary

- WHAT can generate good predictions using only a small number of configurations
- WHAT can build “good” models which can be used in optimizers
- WHAT is comparable to the state of the art predictors

Quality WHAT is close to the actual optimal

Cost Cheaper than the state of the art

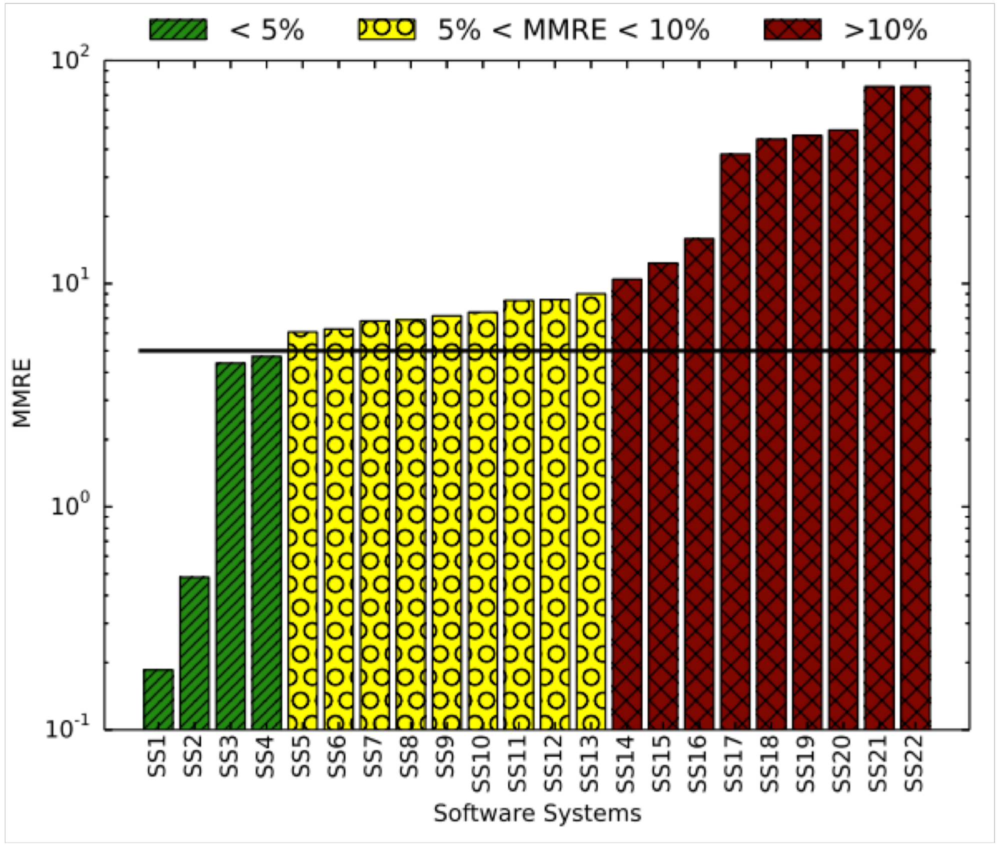
Unsupervised clustering does not work in all cases

Limitations

- Only works if WHAT can generate meaningful clusters.
- Only works when an accurate model can be built
- The stopping condition or threshold (τ) is arbitrary

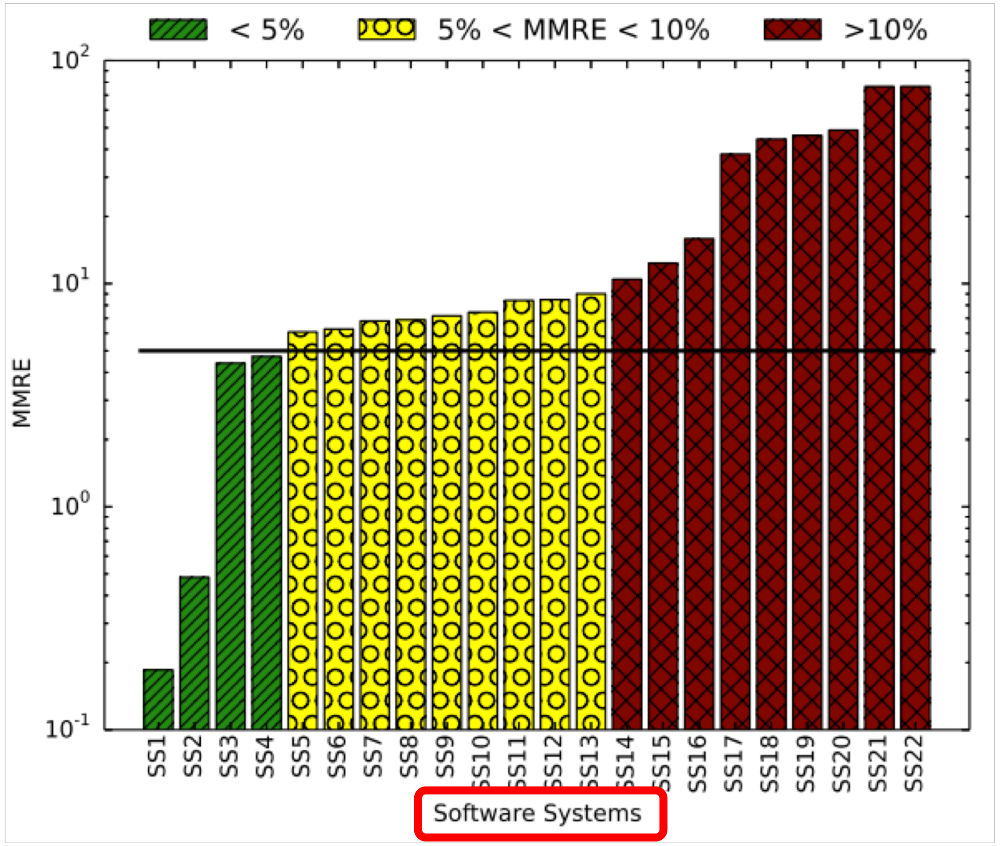
Limitations

- Only works if WHAT can generate meaningful clusters.
- Only works when an accurate model can be built
- The stopping condition or threshold (**τ**) is arbitrary



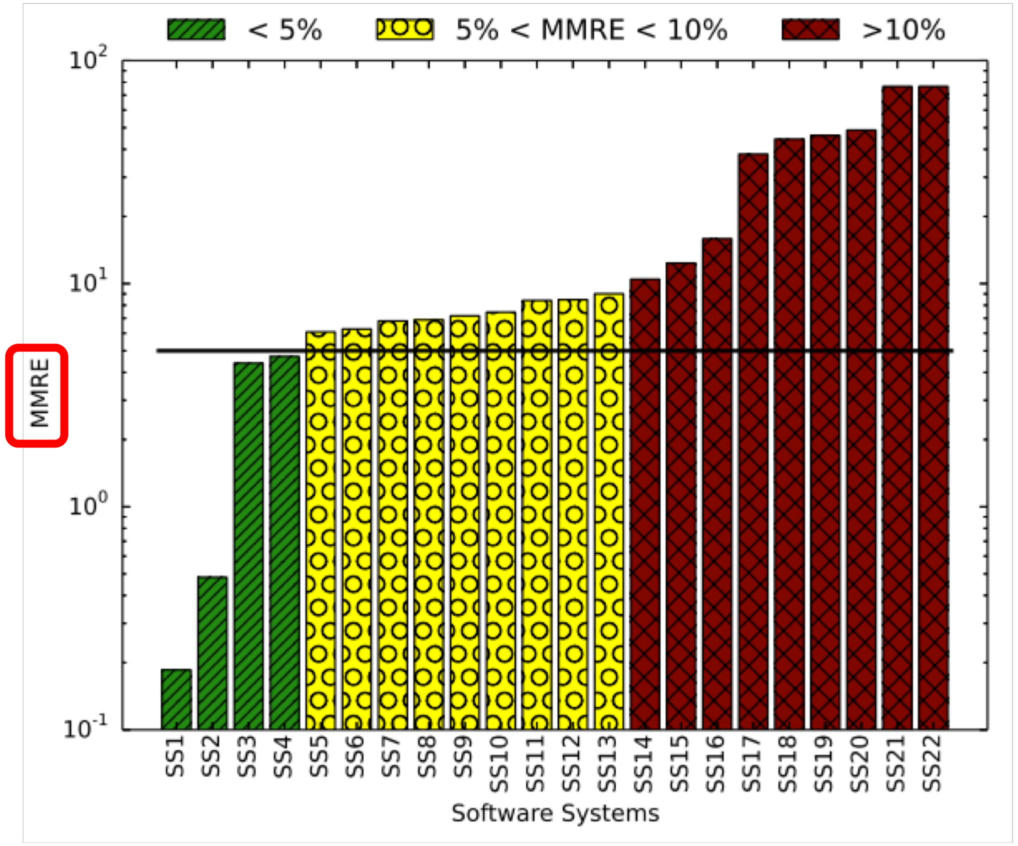
Limitations

- Only works if WHAT can generate meaningful clusters.
- Only works when an accurate model can be built
- The stopping condition or threshold (τ) is arbitrary



Limitations

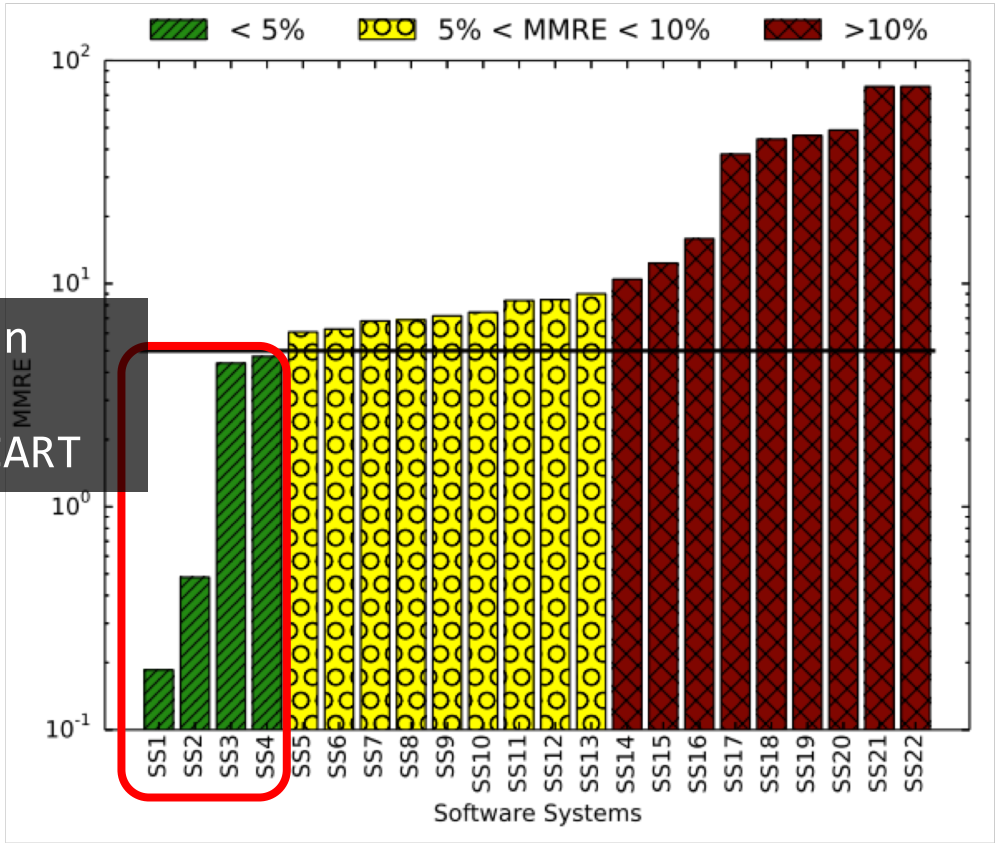
- Only works if WHAT can generate meaningful clusters.
- Only works when an accurate model can be built
- The stopping condition or threshold (τ) is arbitrary



Limitations

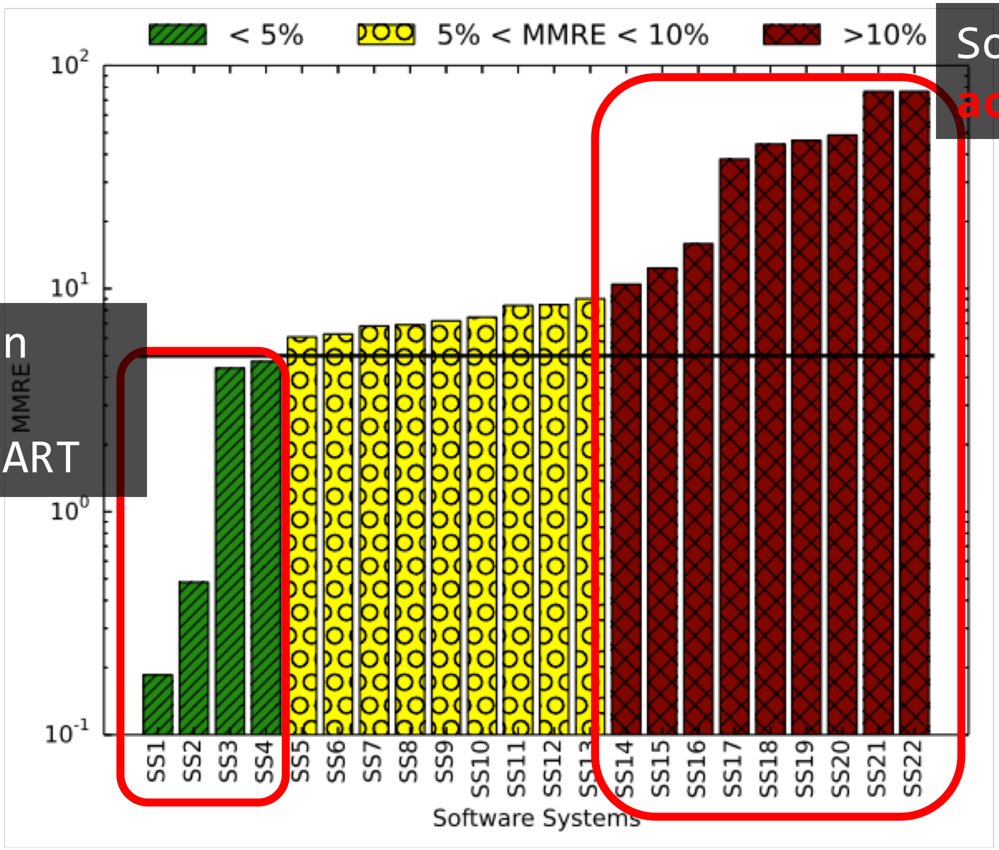
- Only works if WHAT can generate meaningful clusters.
- Only works when an accurate model can be built
- The stopping condition or threshold (**τ**) is arbitrary

Software Systems used in prior works can be accurately modeled by CART



Limitations

- Only works if WHAT can generate meaningful clusters.
- Only works when an accurate model can be built
- The stopping condition or threshold (**τ**) is arbitrary

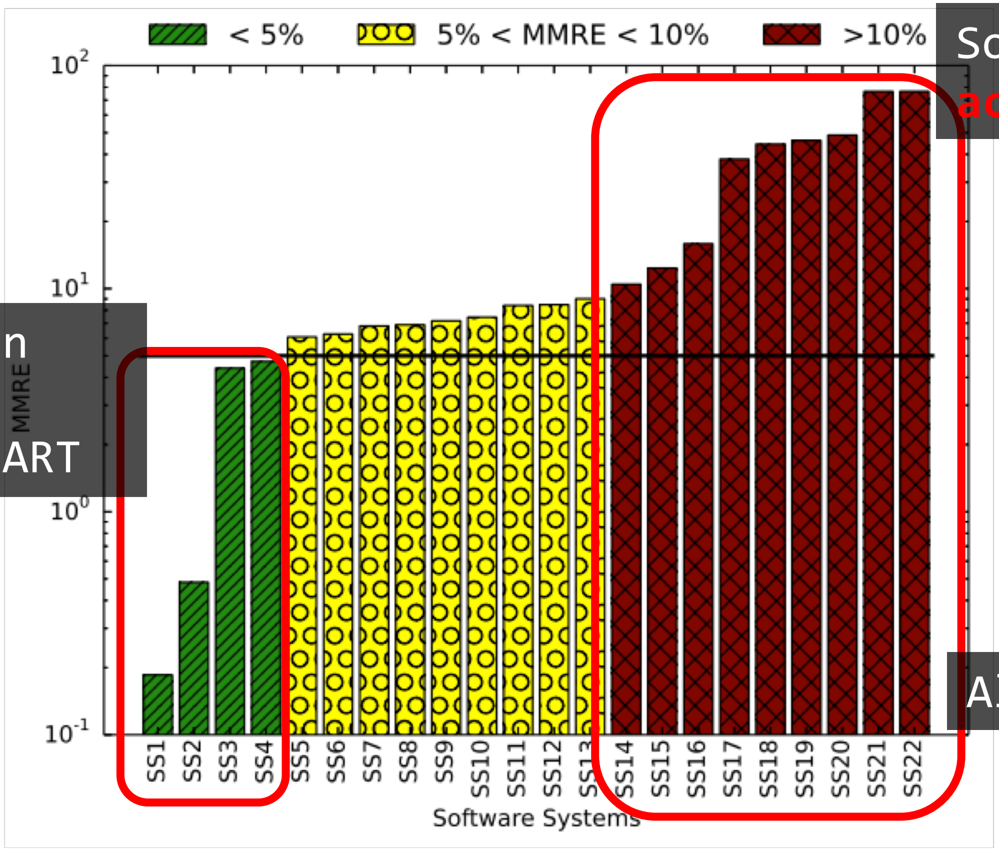


Software Systems cannot be accurately modeled by CART

Software Systems used in prior works can be accurately modeled by CART

Limitations

- Only works if WHAT can generate meaningful clusters.
- Only works when an accurate model can be built
- The stopping condition or threshold (τ) is arbitrary



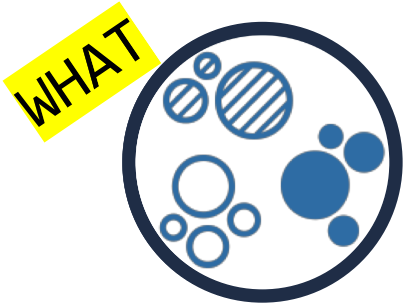
Software Systems used in prior works can be accurately modeled by CART

Software Systems cannot be accurately modeled by CART

All prior work would FAIL

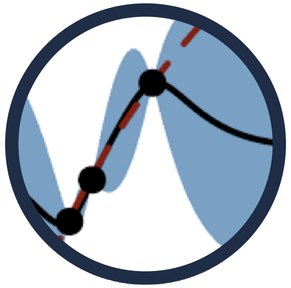
Presented during Oral Prelims

Clustering



Ranking

SMBO



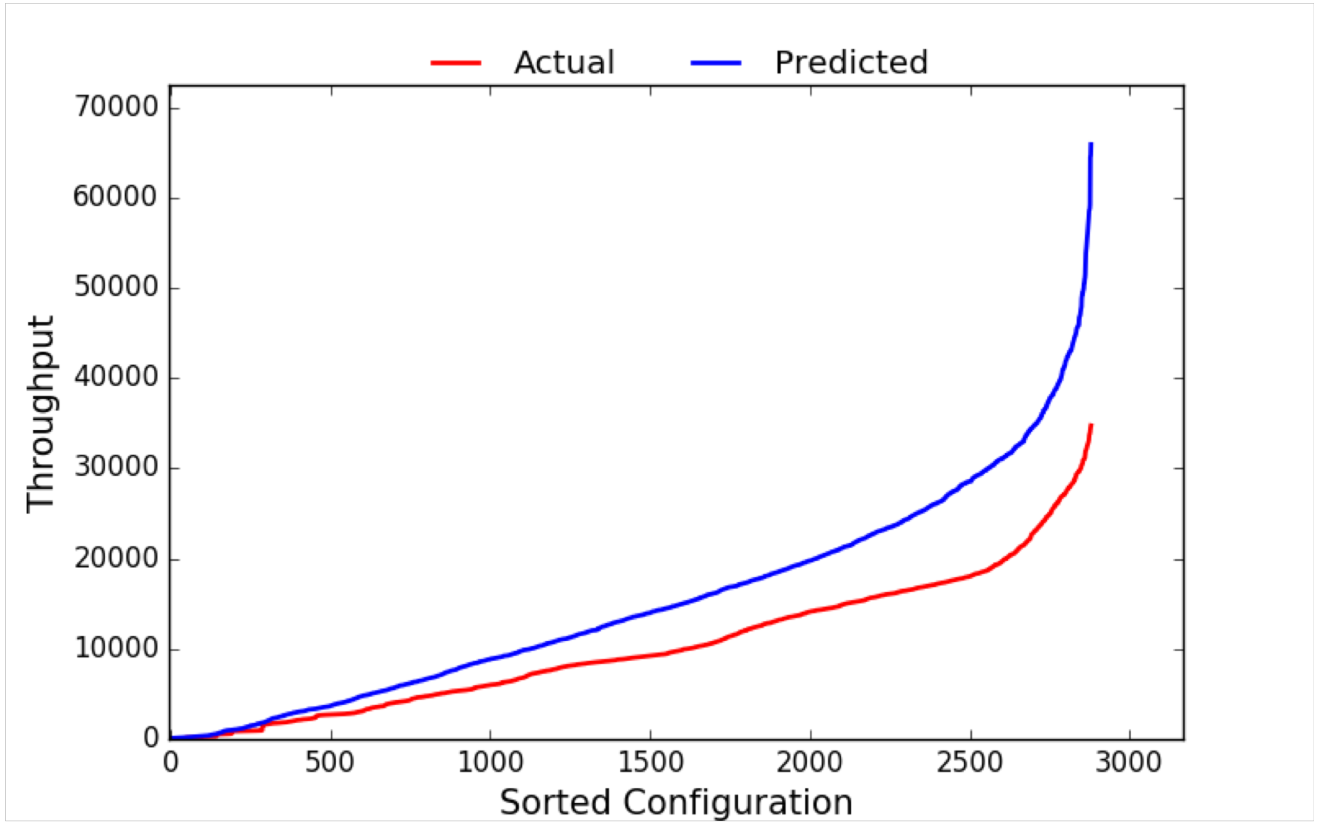
Nair et al.; Using Bad Learners to find Good Configurations; FSE (2017)

Intuition

Rank-preserving model rather than **highly accurate model**

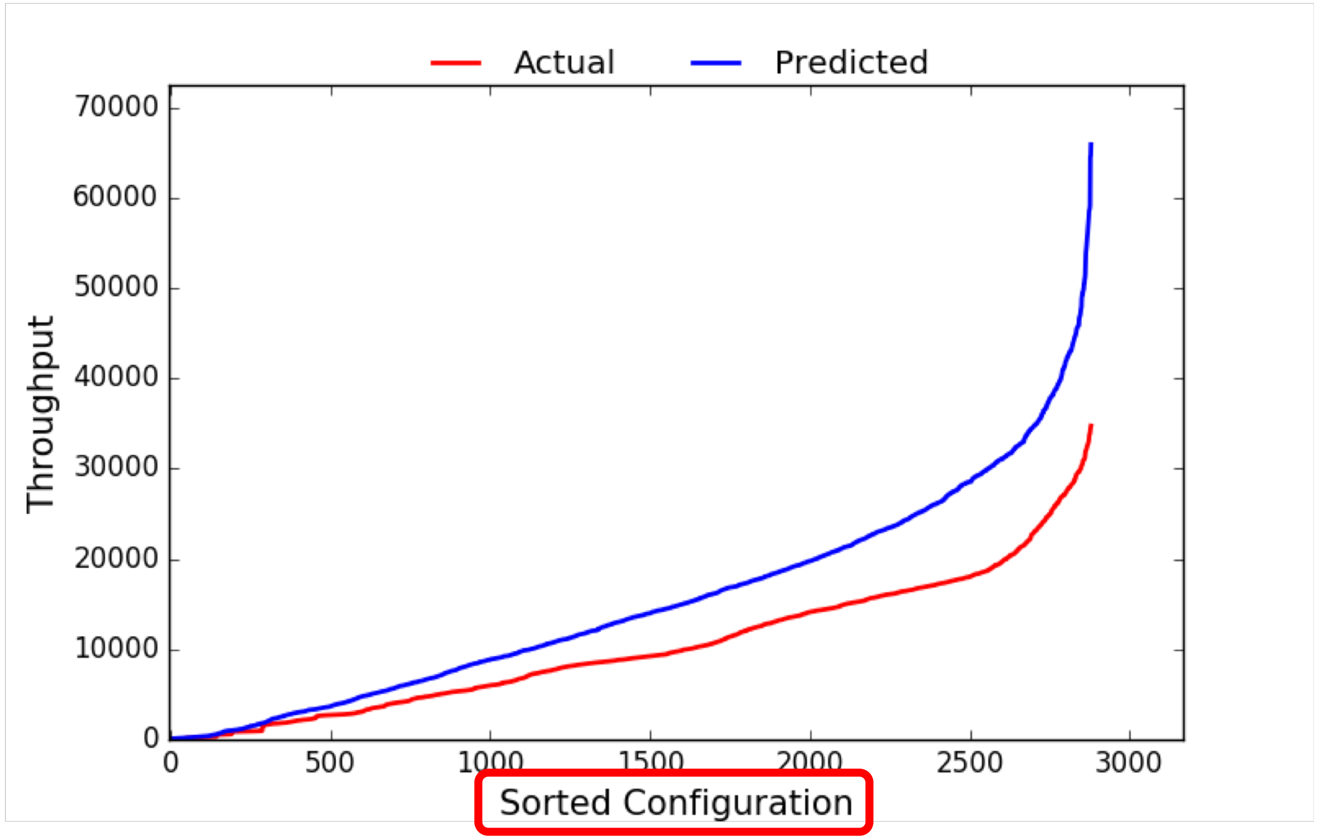
Intuition

Rank-preserving model rather than **highly accurate model**



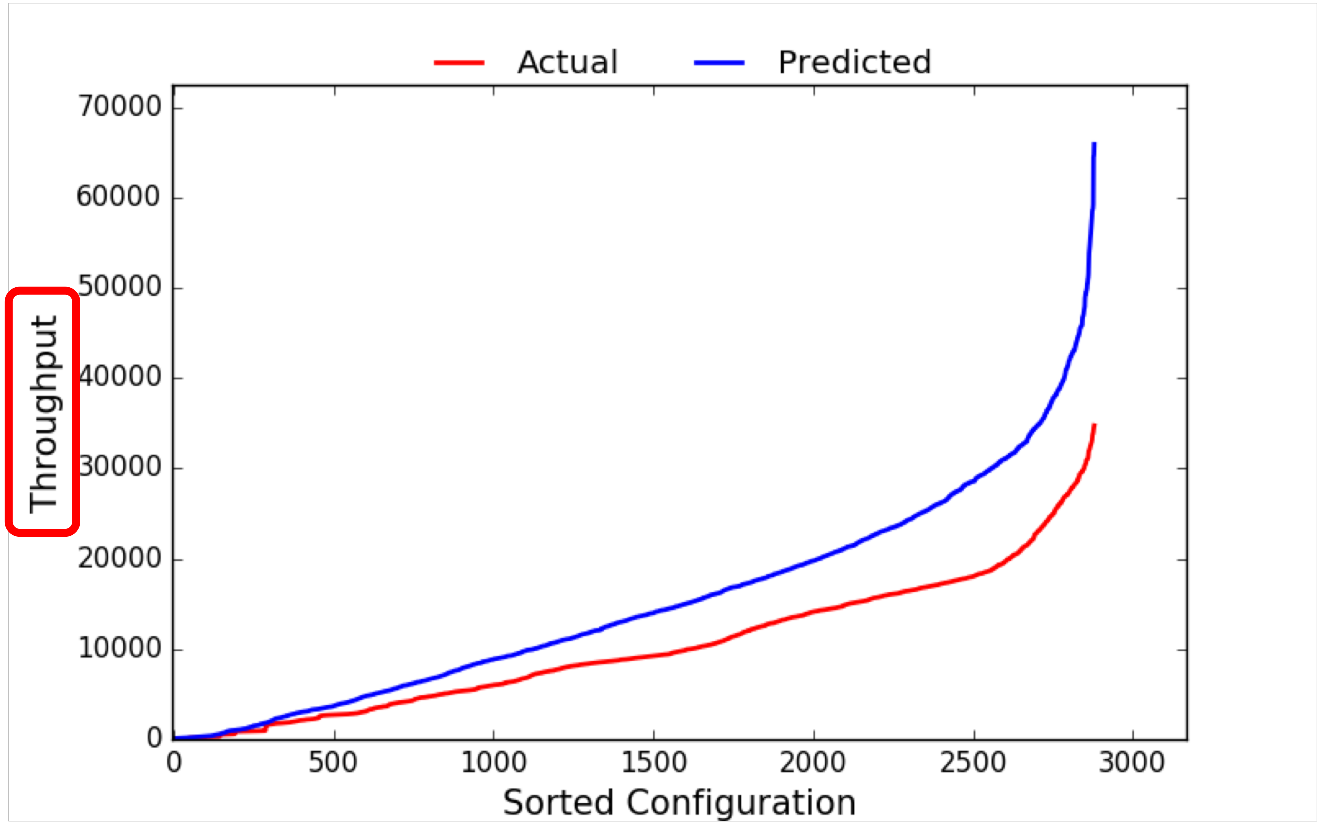
Intuition

Rank-preserving model rather than **highly accurate model**



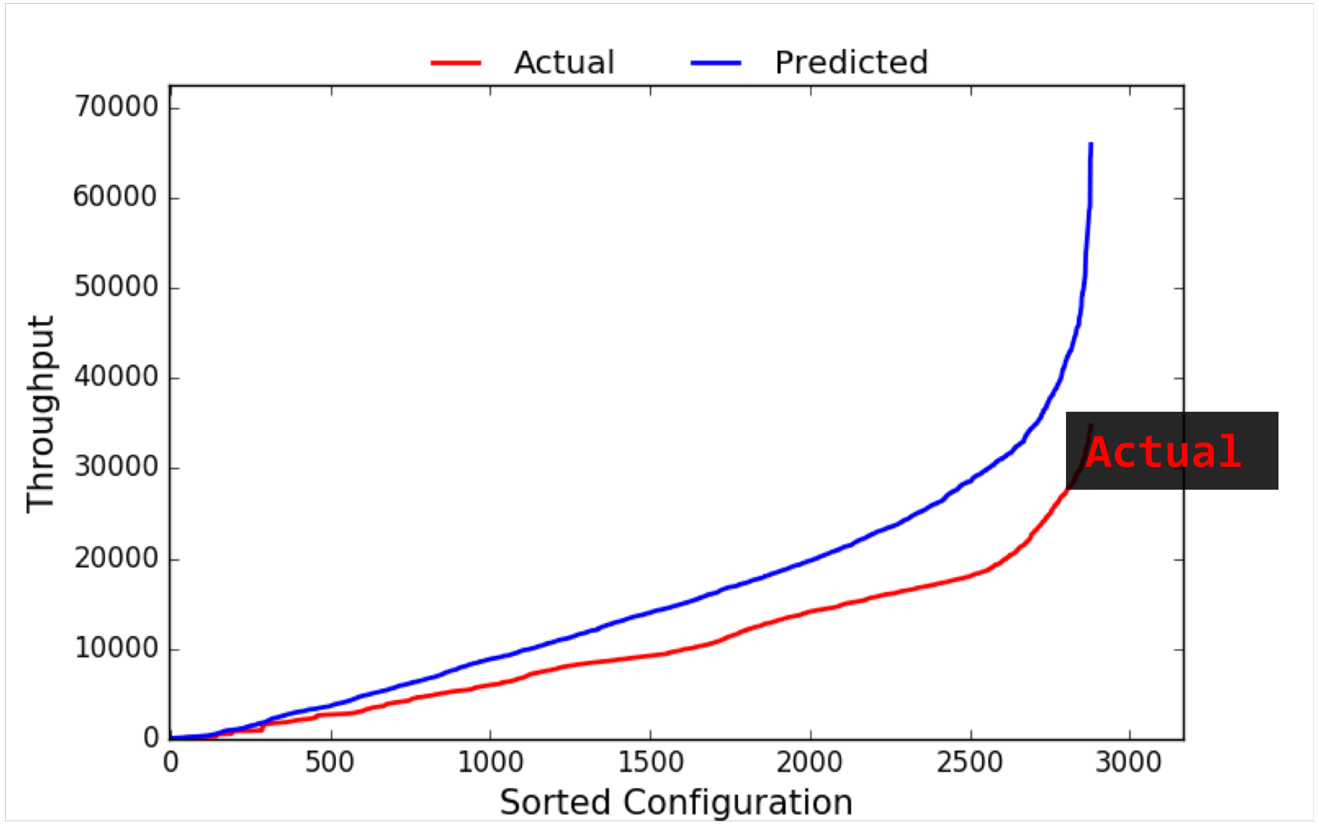
Intuition

Rank-preserving model rather than **highly accurate model**



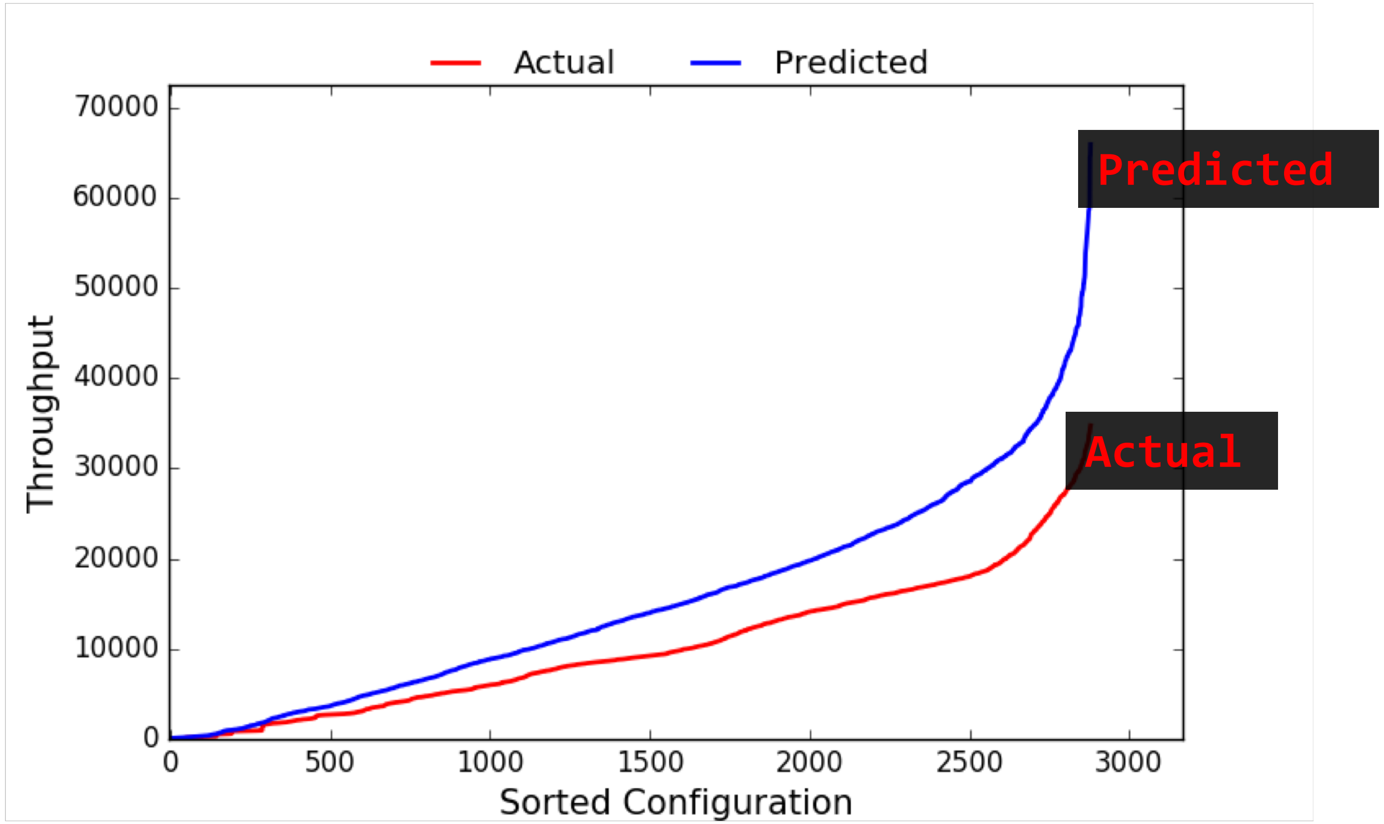
Intuition

Rank-preserving model rather than **highly accurate model**



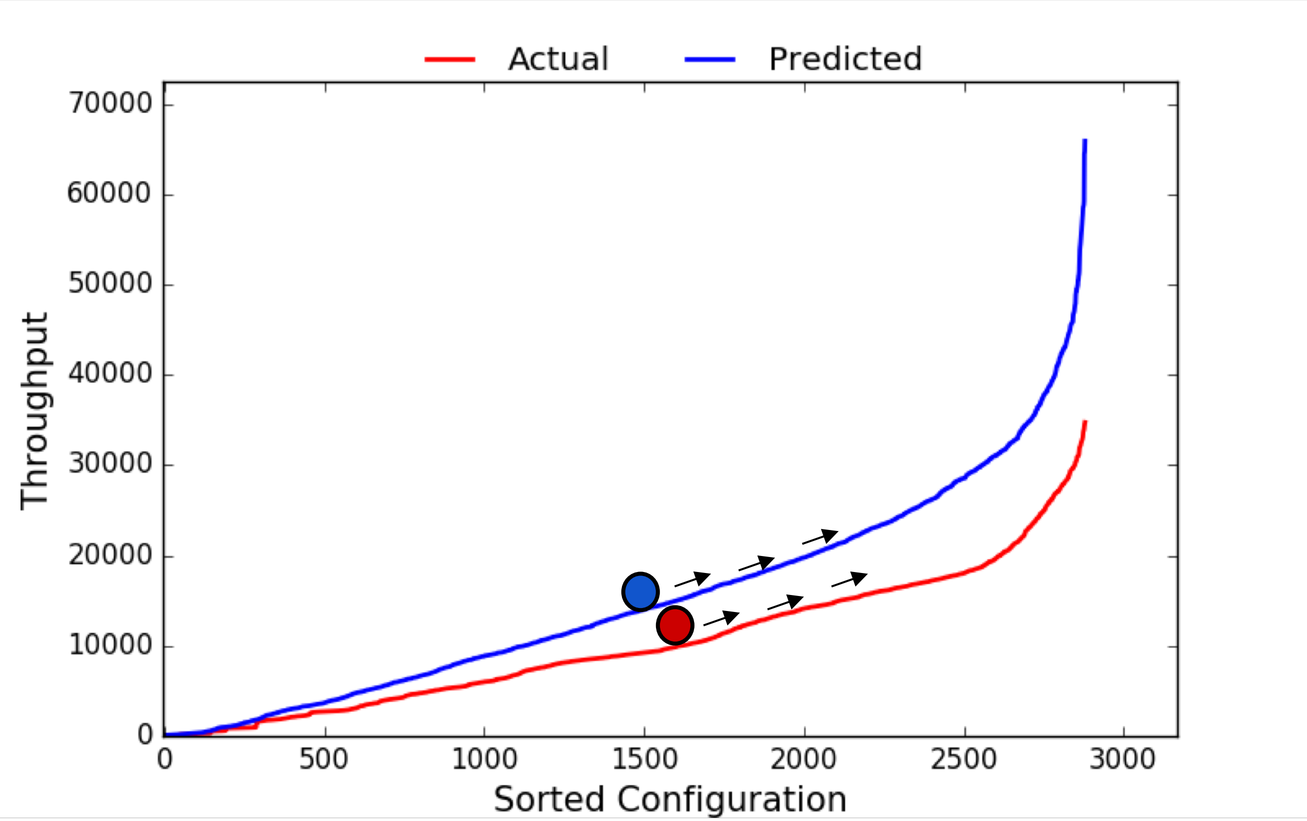
Intuition

Rank-preserving model rather than **highly accurate model**



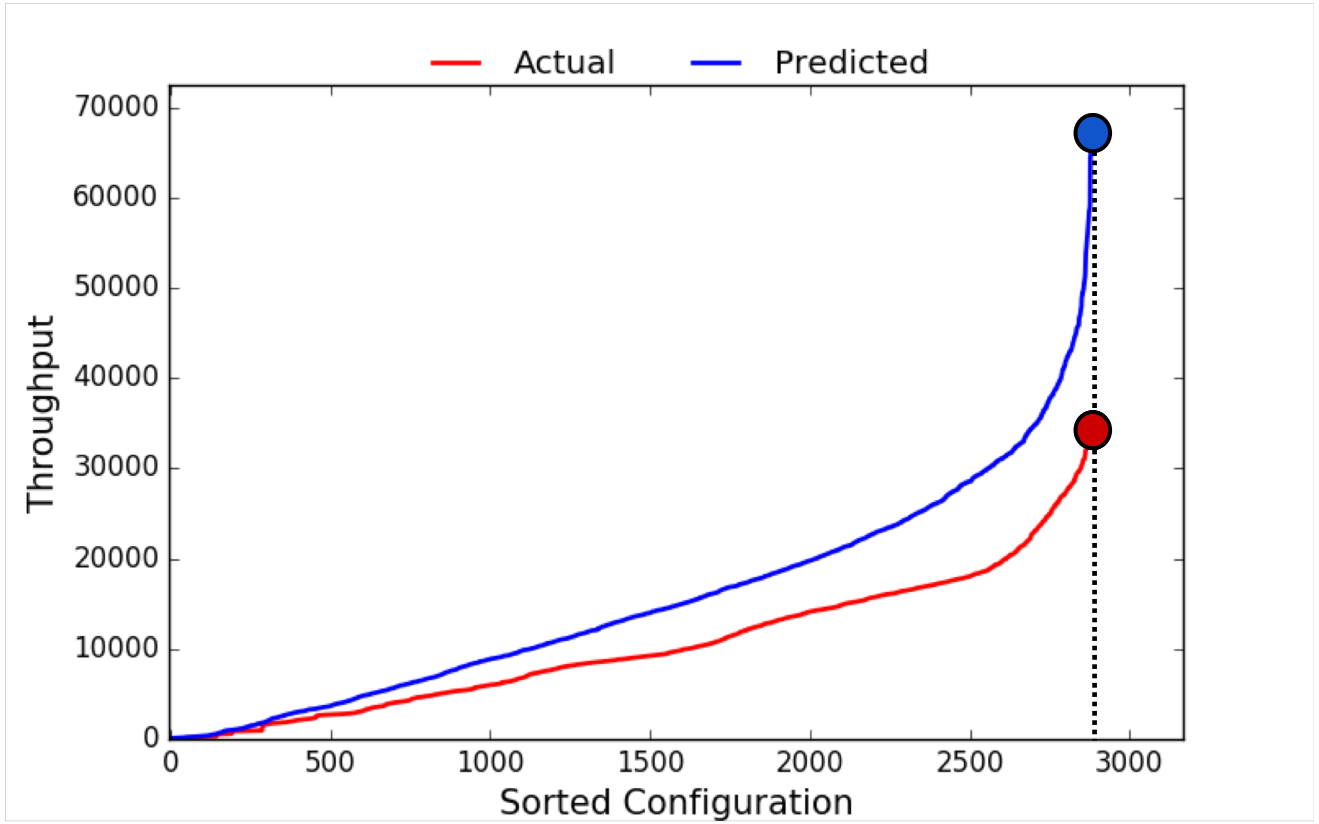
Intuition

Rank-preserving model rather than **highly accurate model**



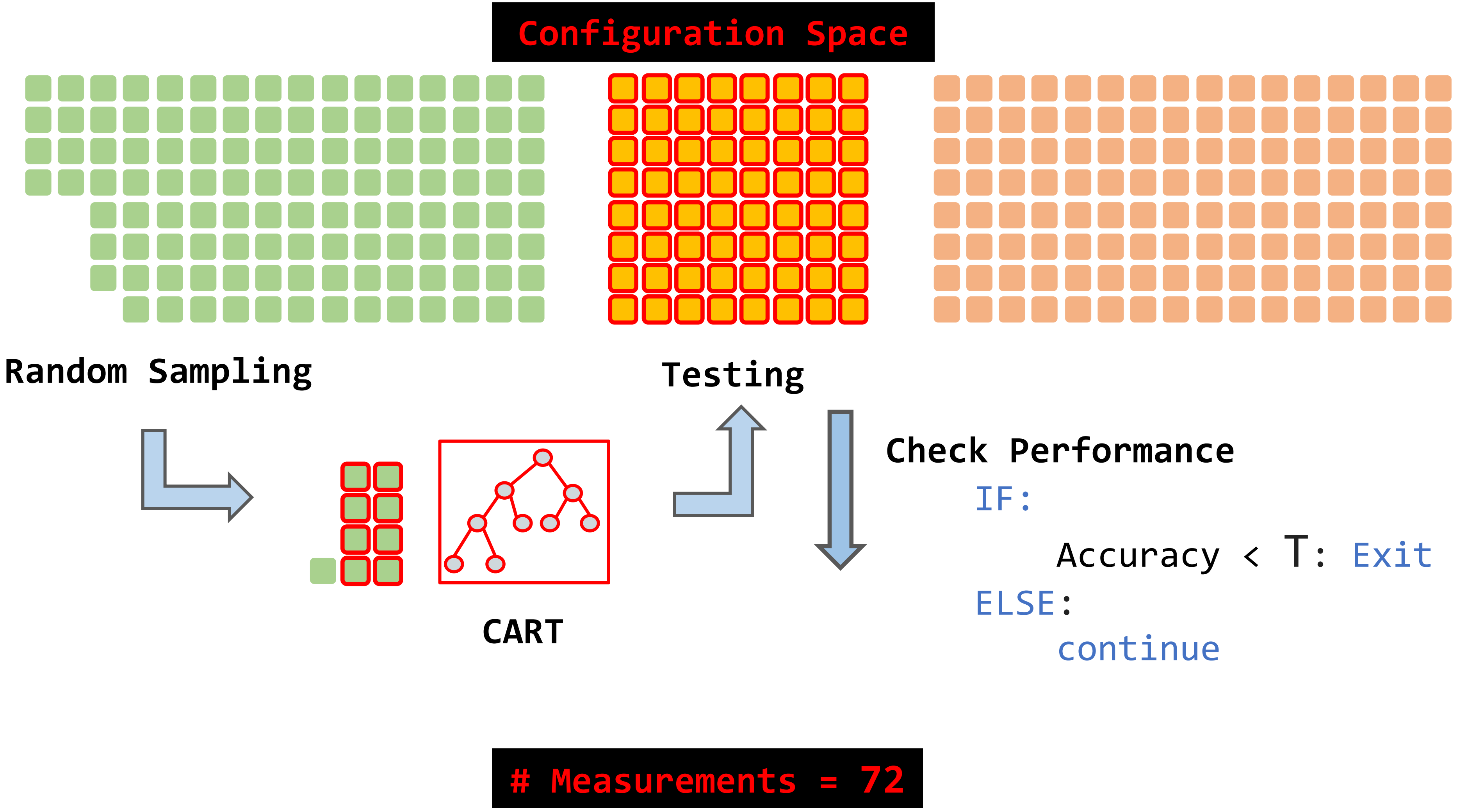
Intuition

Rank-preserving model rather than **highly accurate model**



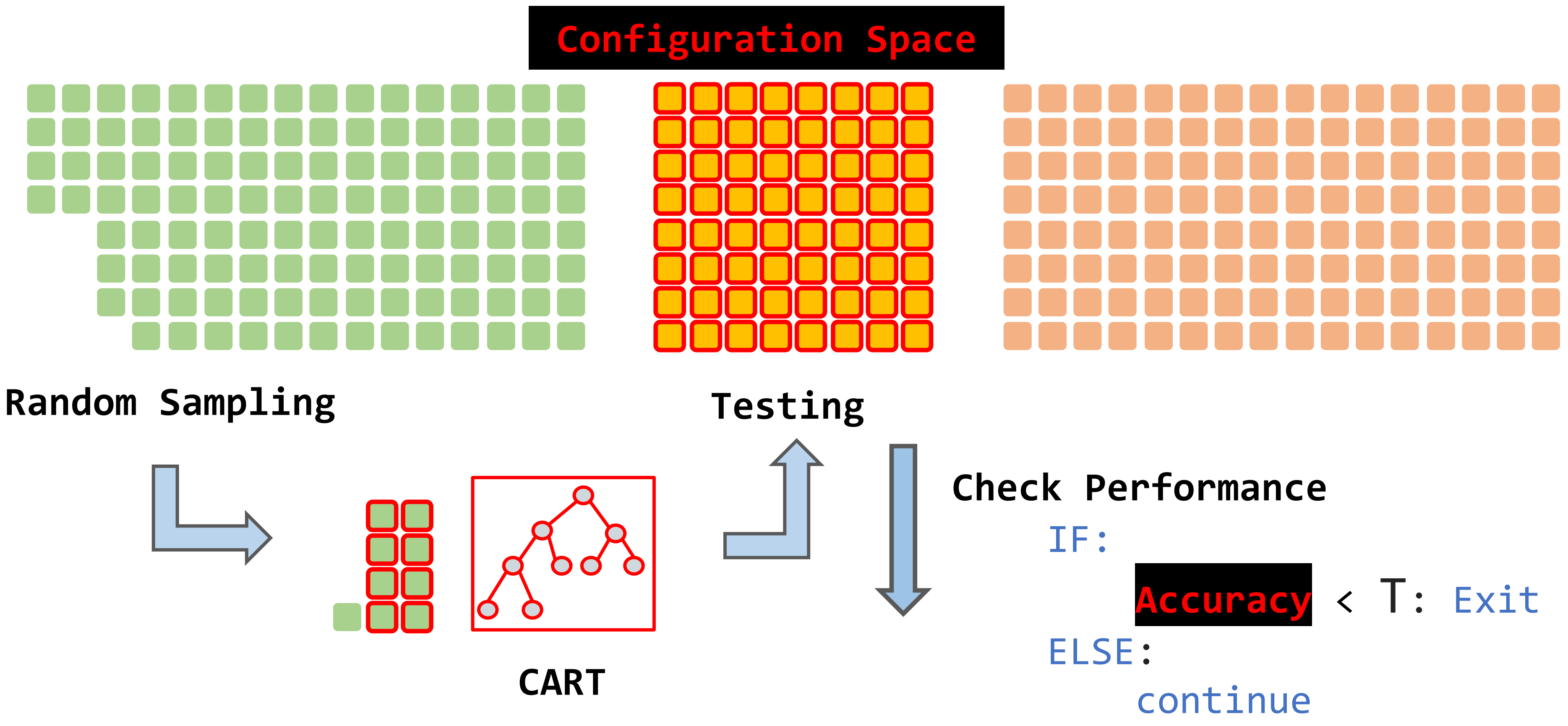
Best Configuration obtained using actual and the predicted values is the same

Ranking



Configuration Space
Random Sampling
Testing
CART
Check Performance
IF:
Accuracy < T: Exit
ELSE:
continue
Measurements = 72

Ranking



Measurements = 72

$$accuracy = \frac{1}{n} \cdot \sum_{i=1}^{n} \left| rank(y_i) - rank(f(x_i)) \right|$$

Summary

- A rank-based method can be used to find (near) optimal configurations
- A rank-based approach requires fewer measurements

Conclusion

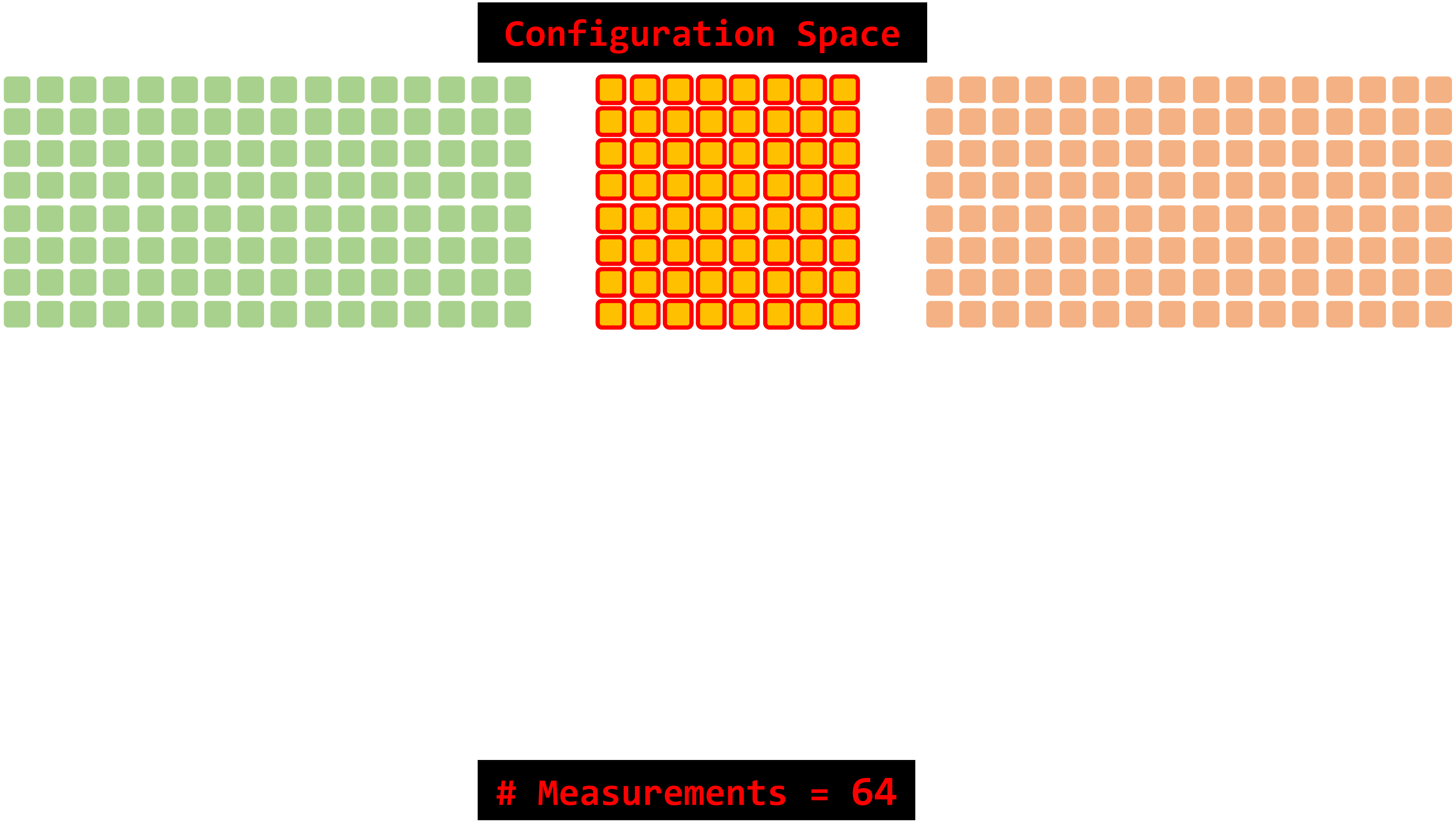
- A rank-based method can be used to find (near) optimal configurations
- A rank-based approach requires fewer measurements

Quality Rank based approaches finds configurations close to the actual optimal

Cost Cheaper than the state of the art

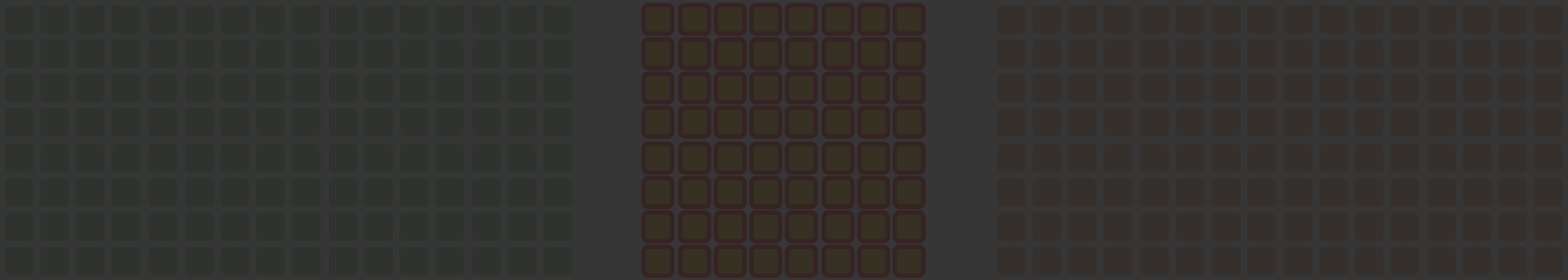
Ranking is a useful paradigm

Limitations



Previously?

Configuration Space

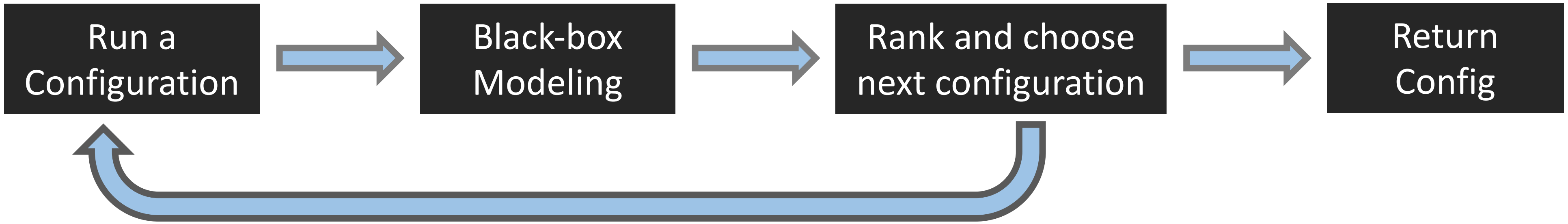


Expensive

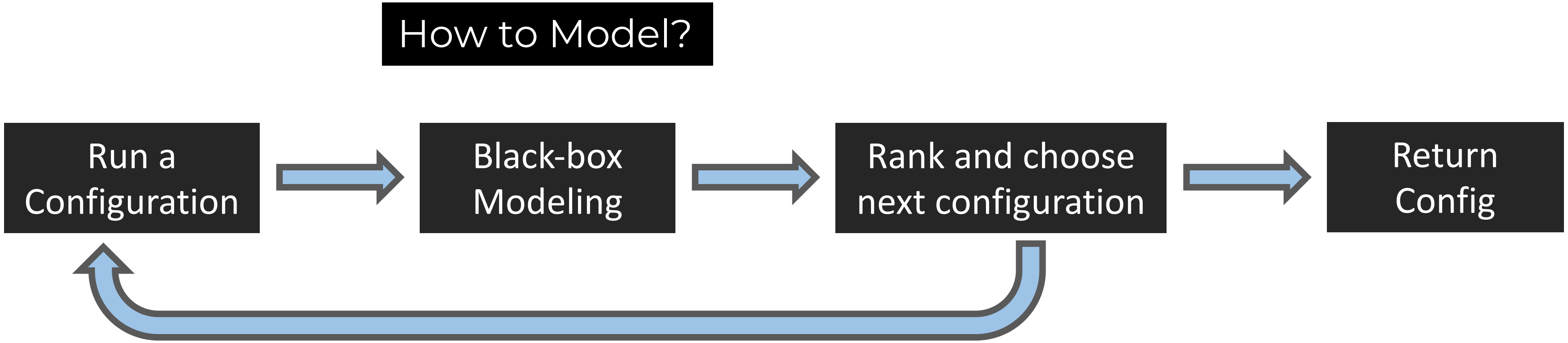
Measurements = 64

Nair et al.; Finding faster configurations using Flash; TSE (2018)

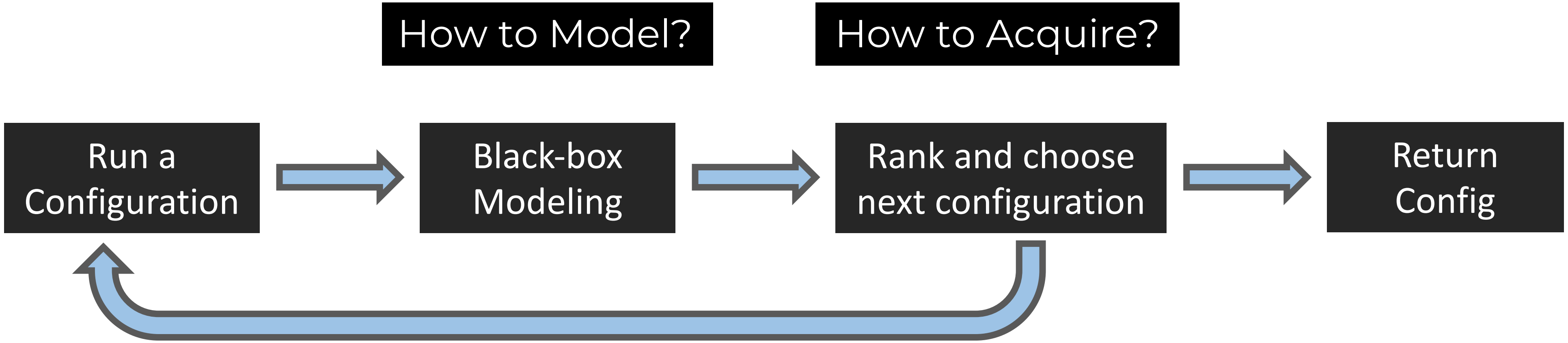
Workflow of SMBO



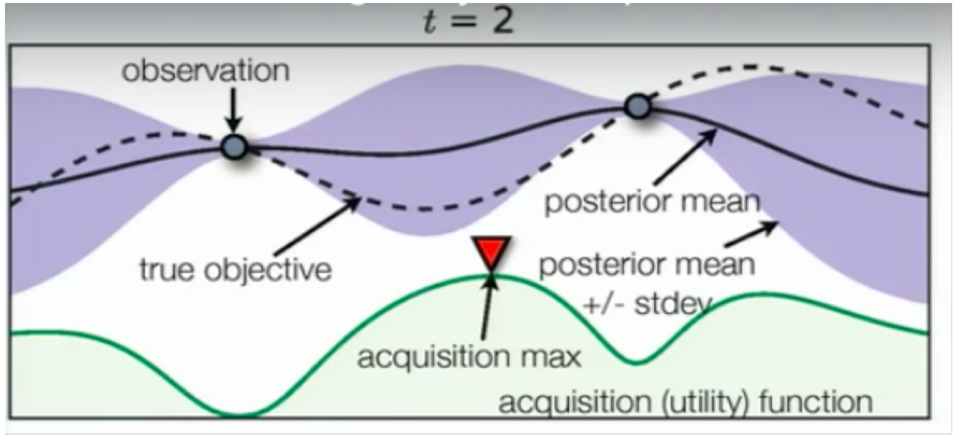
Workflow of SMBO



Workflow of SMBO

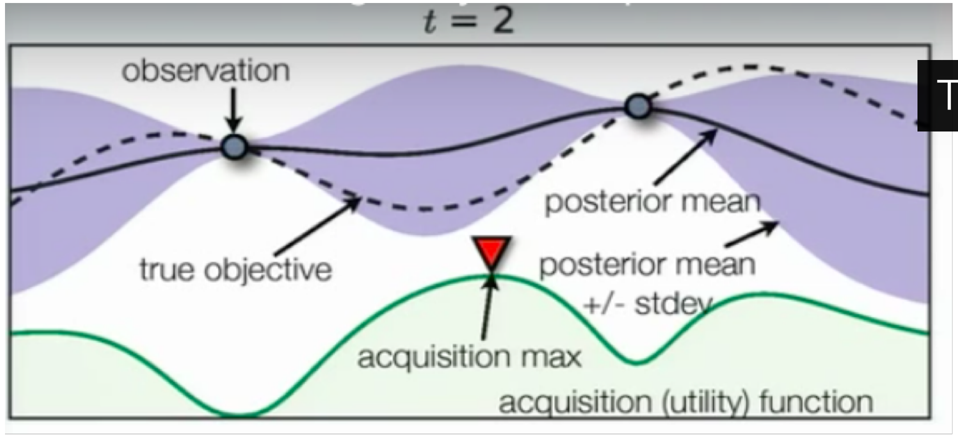


Workflow of SMBO



Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

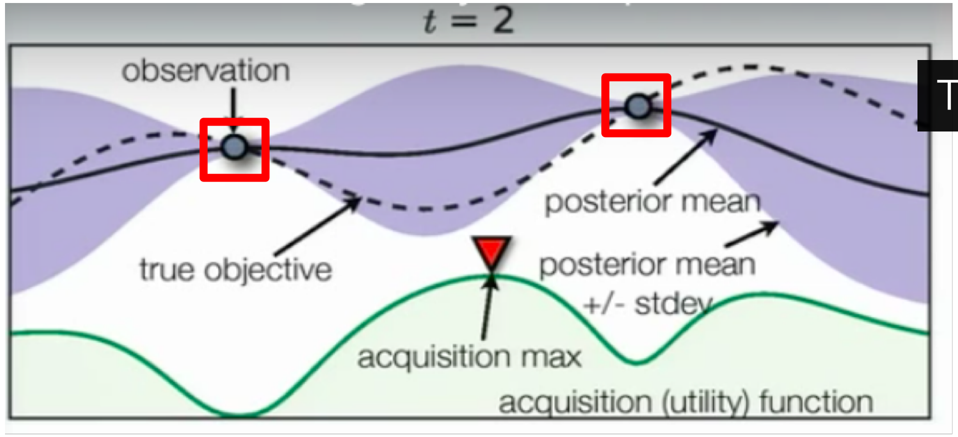
Workflow of SMBO



True Performance Distribution

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

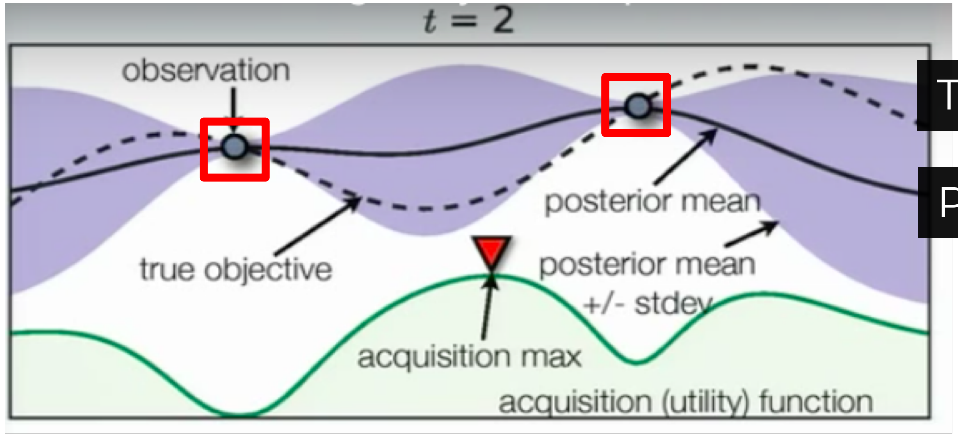
Workflow of SMBO



True Performance Distribution

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO

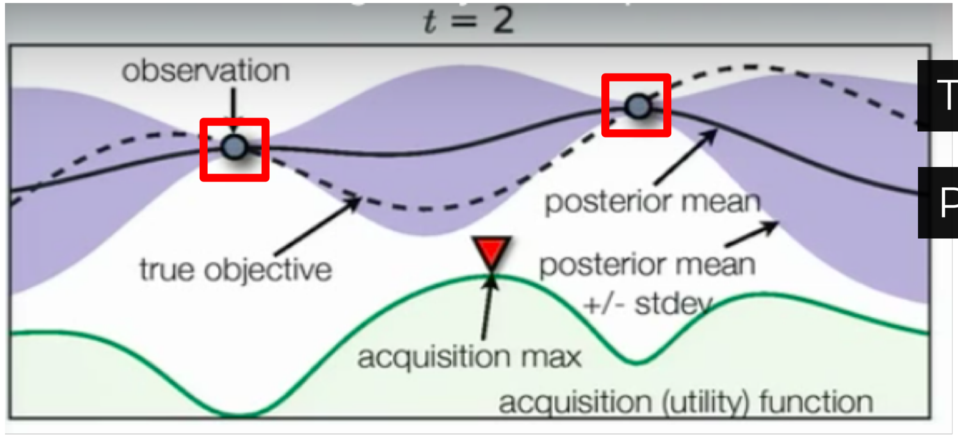


True Performance Distribution

Predicted Performance Distribution

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



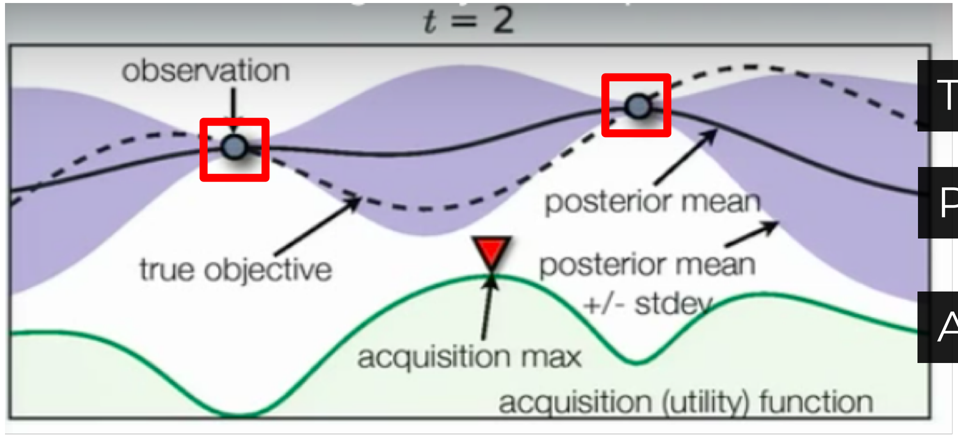
True Performance Distribution

Predicted Performance Distribution

Surrogate of choice:
Gaussian Processes (GP)

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



True Performance Distribution

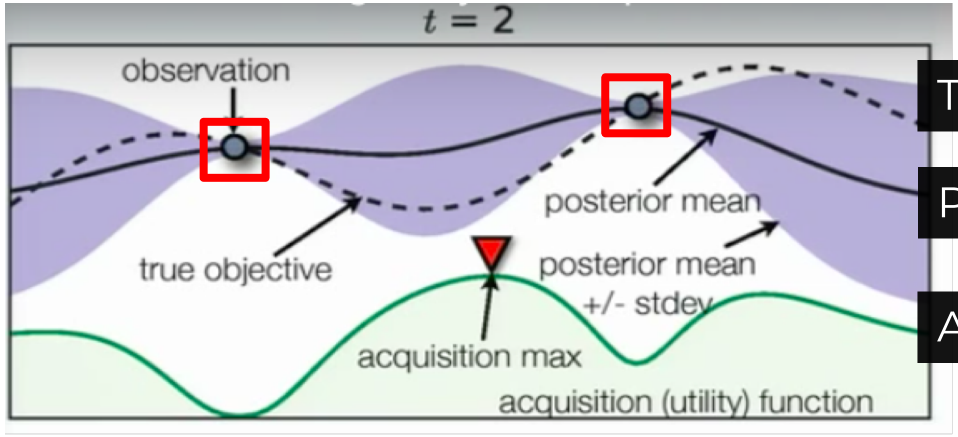
Predicted Performance Distribution

Acquisition Function

Surrogate of choice:
Gaussian Processes (GP)

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



True Performance Distribution

Predicted Performance Distribution

Surrogate of choice:
Gaussian Processes (GP)

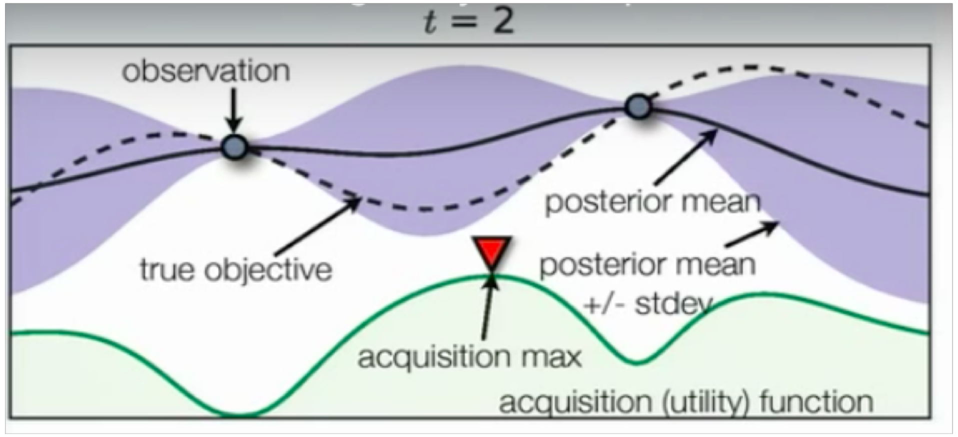
Acquisition Function

$\mu(x) + \kappa . \sigma(x)$

Tradeoff between
Exploration vs Exploitation

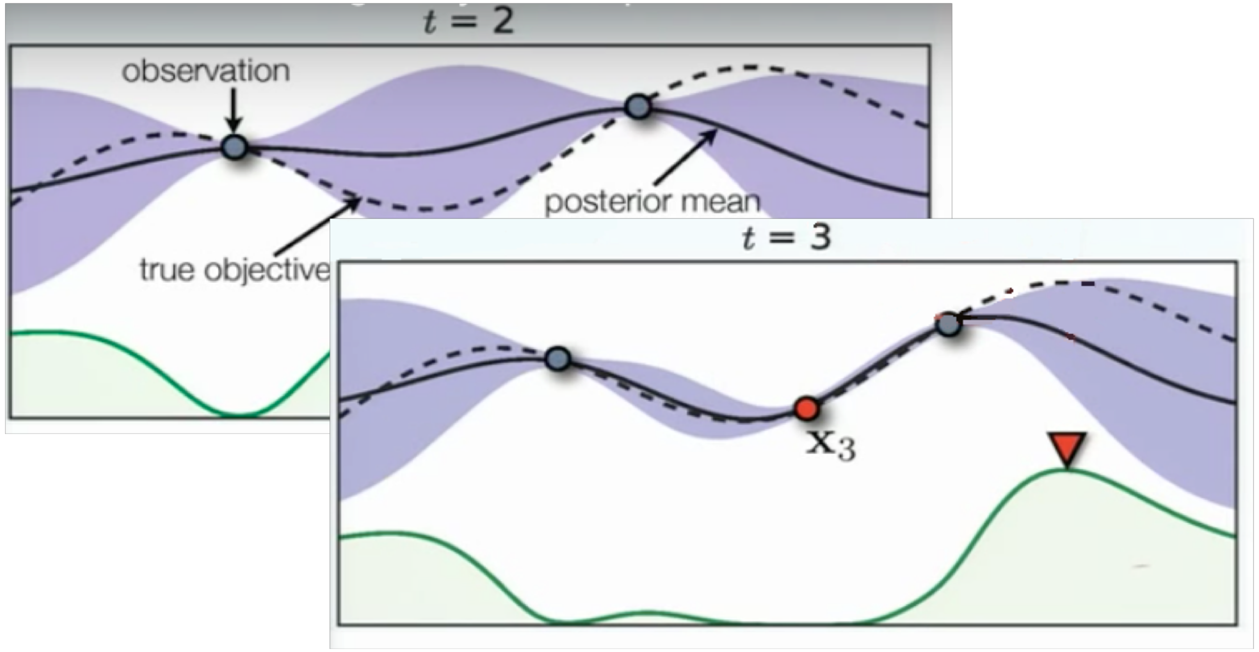
Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



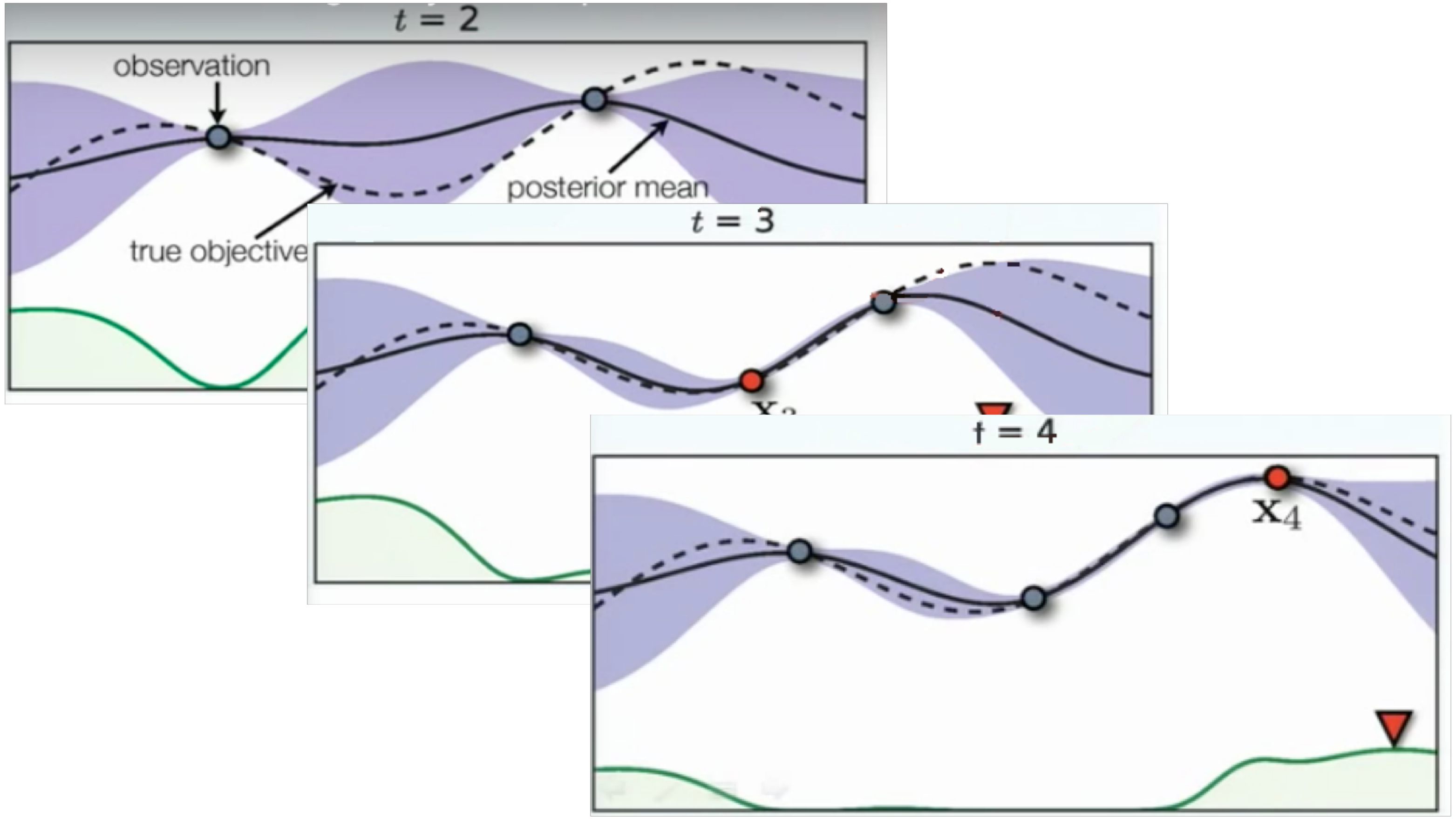
Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



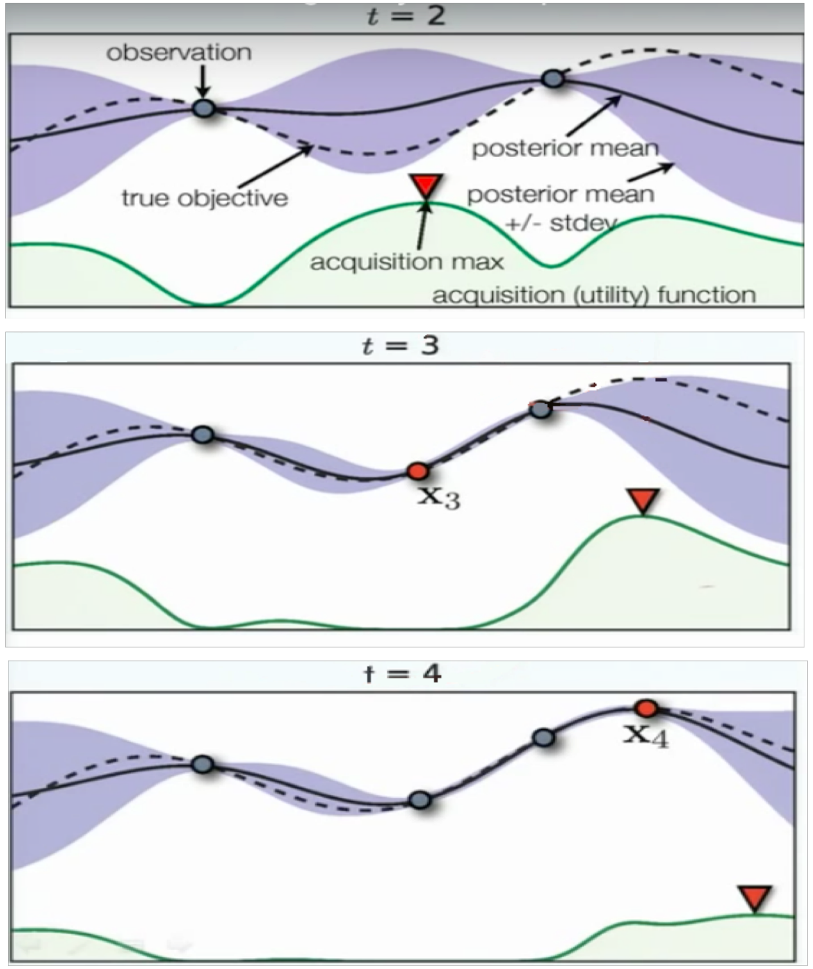
Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



Input: $f, \mathcal{X}, S, \mathcal{M}$

$\mathcal{D} \leftarrow \textsc{InitSamples}(f, \mathcal{X})$

for $i \leftarrow |\mathcal{D}|$ **to** T **do**

  $p(y \mid \mathbf{x}, \mathcal{D}) \leftarrow \textsc{FitModel}(\mathcal{M}, \mathcal{D})$

  $\mathbf{x}_i \leftarrow \arg\max_{\mathbf{x} \in \mathcal{X}} S(\mathbf{x},\ p(y \mid \mathbf{x}, \mathcal{D}))$

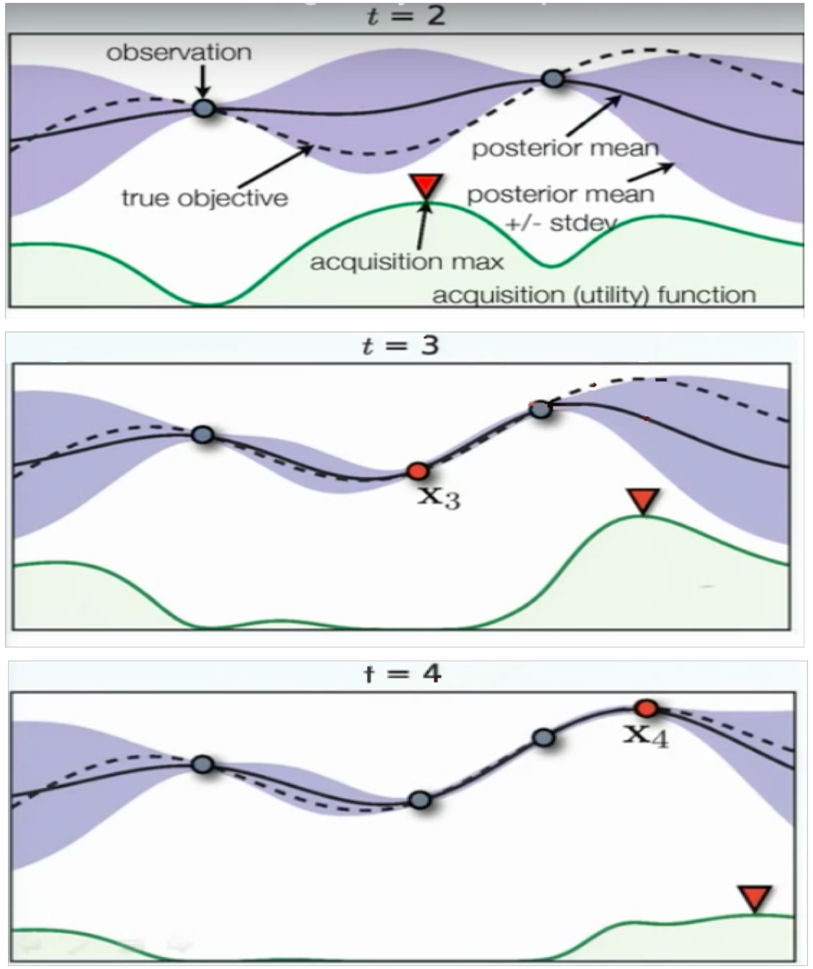
  $y_i \leftarrow f(\mathbf{x}_i)$ $\triangleright$ Expensive step

  $\mathcal{D} \leftarrow \mathcal{D} \cup (\mathbf{x}_i, y_i)$

end for

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of SMBO



Input: $f, \mathcal{X}, S, \mathcal{M}$

$\mathcal{D} \leftarrow \textsc{InitSamples}(f, \mathcal{X})$

for $i \leftarrow |\mathcal{D}|$ **to** T **do**

  $p(y \mid \mathbf{x}, \mathcal{D}) \leftarrow \textsc{FitModel}(\mathcal{M}, \mathcal{D})$

  $\mathbf{x}_i \leftarrow \arg\max_{\mathbf{x} \in \mathcal{X}} S(\mathbf{x},\ p(y \mid \mathbf{x}, \mathcal{D}))$

  $y_i \leftarrow f(\mathbf{x}_i)$ $\triangleright$ Expensive step

  $\mathcal{D} \leftarrow \mathcal{D} \cup (\mathbf{x}_i, y_i)$

end for

Hyperopt

MOE

SMAC

Spearmint

Google Vizier

ePAL

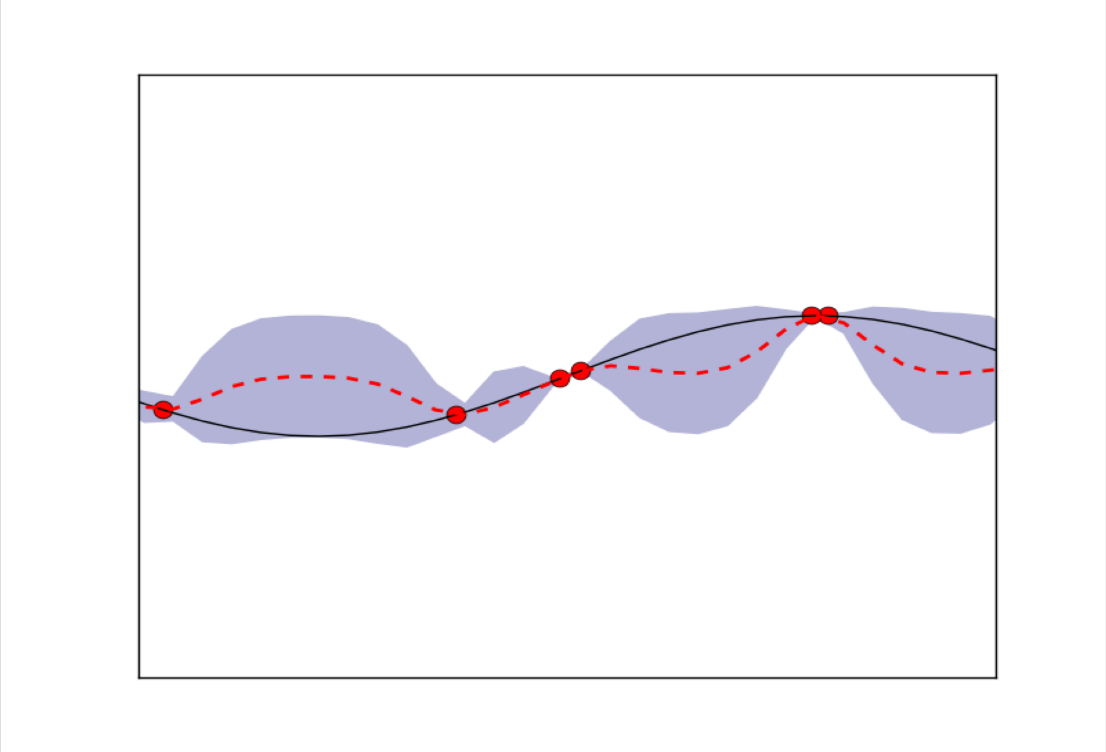
Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Limitations of SMBO

- GPMs can be very fragile, that is, very sensitive to the parameters of GPMs[1]
- GPMs do not scale to high dimensional data as well as a large dataset[2]
- GPMs for optimization was limited to models with around ten decisions[3]

[1] Brochu et al.; “A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning,”; ArXiv, p. 49, 2010.
[2] Shen et al.; Fast gaussian process regression using kd-trees. In Advances in neural information processing systems; 2006.
[3] Wang et al.; Bayesian optimization in a billion dimensions via random embeddings; Journal of Artificial Intelligence Research, 2016.

Workflow of Flash



Gaussian Processes

$\mu(x) + \kappa . \sigma(x)$

Run a Configuration

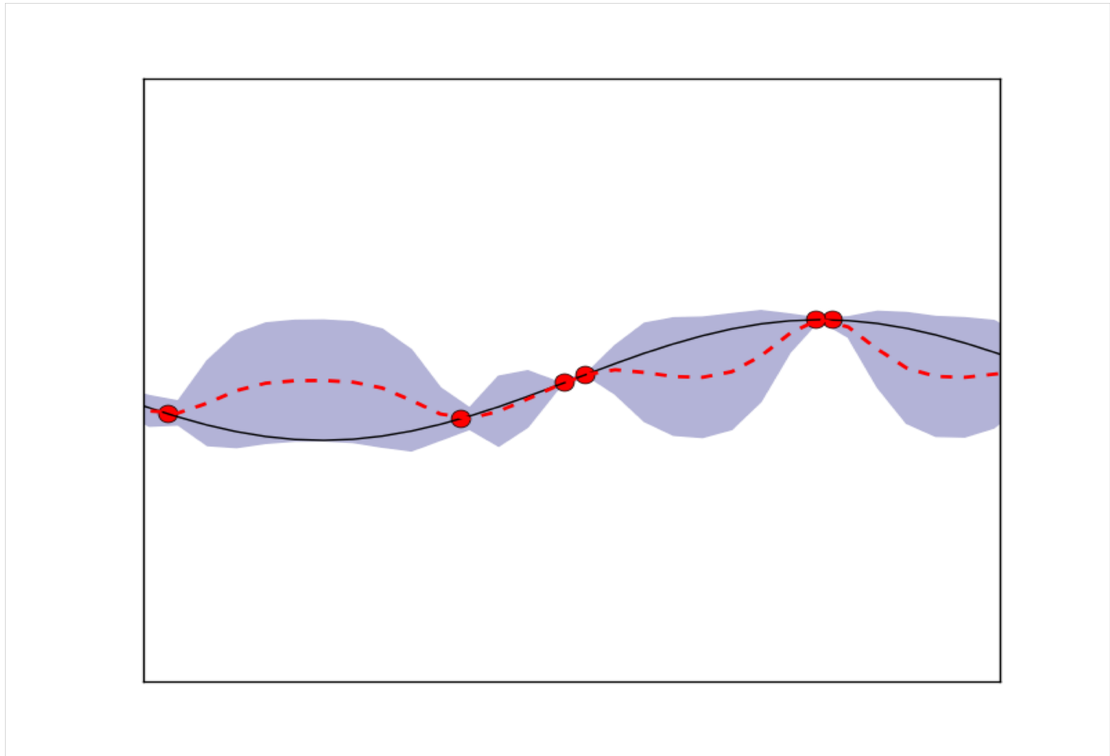
Black-box Modeling

Rank and choose next configuration

Return Config

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of Flash



Gaussian Processes

$\mu(x) \cancel{+ \kappa.\sigma(x)}$

Run a Configuration

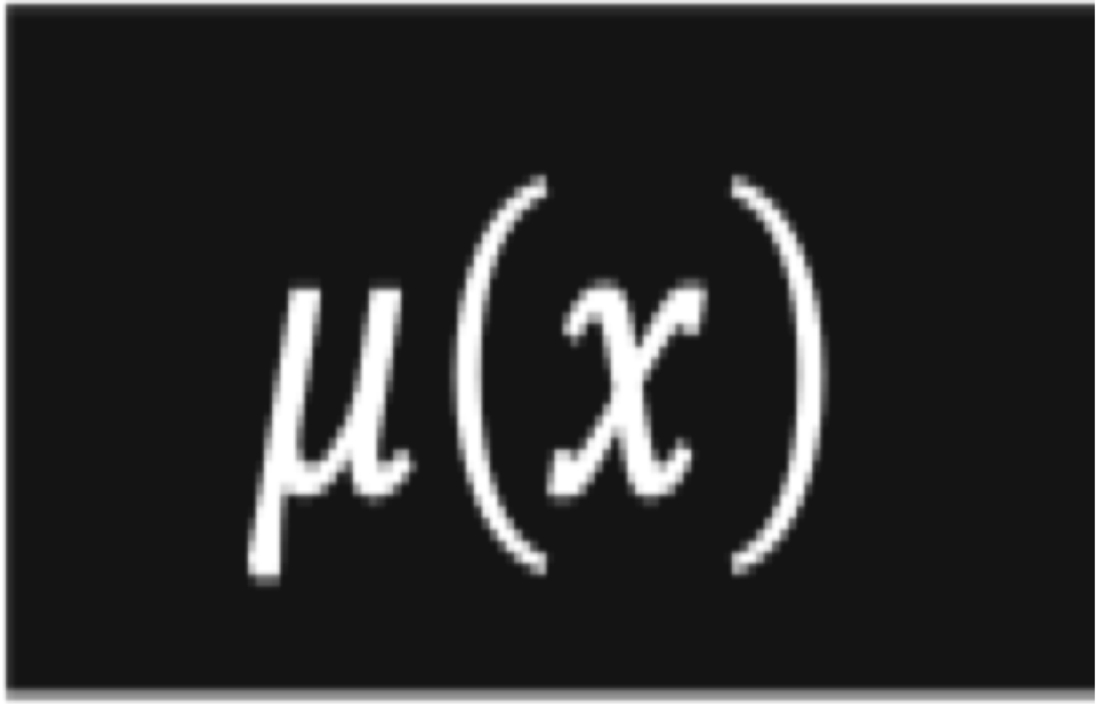
Black-box Modeling

Rank and choose next configuration

Return Config

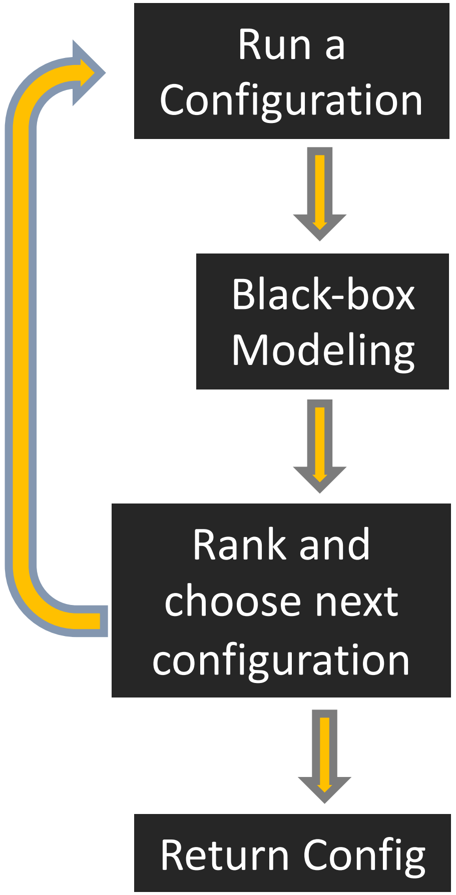
Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of Flash



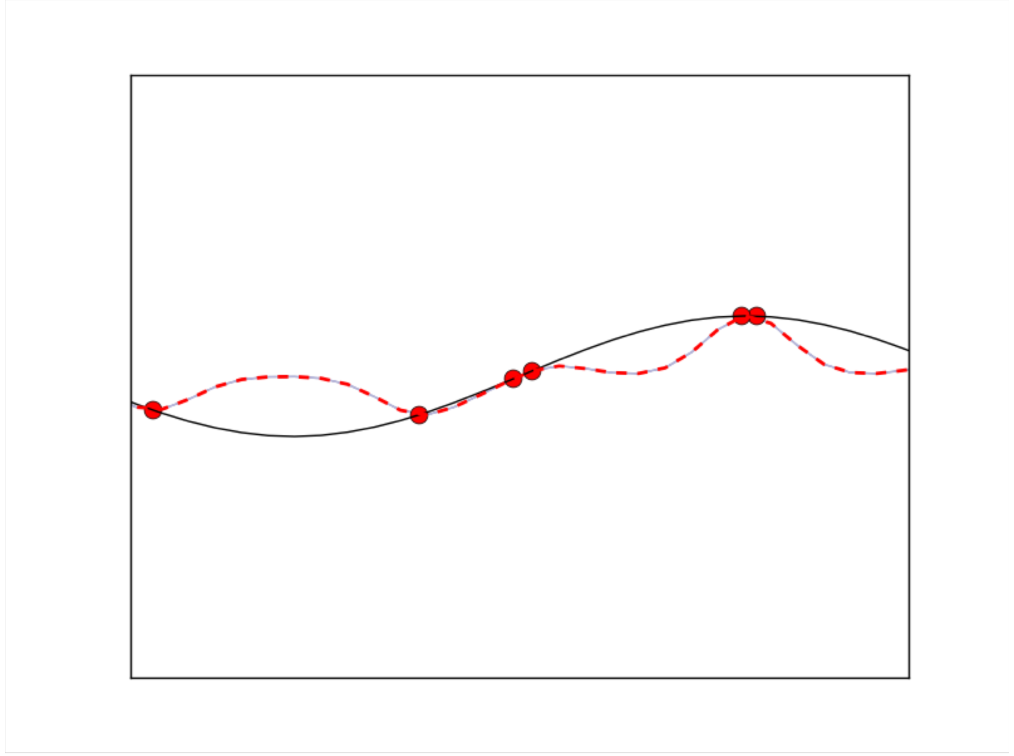
Gaussian Processes

$\mu(x)$



Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of Flash



CART

$\mu(x)$

Run a Configuration

Black-box Modeling

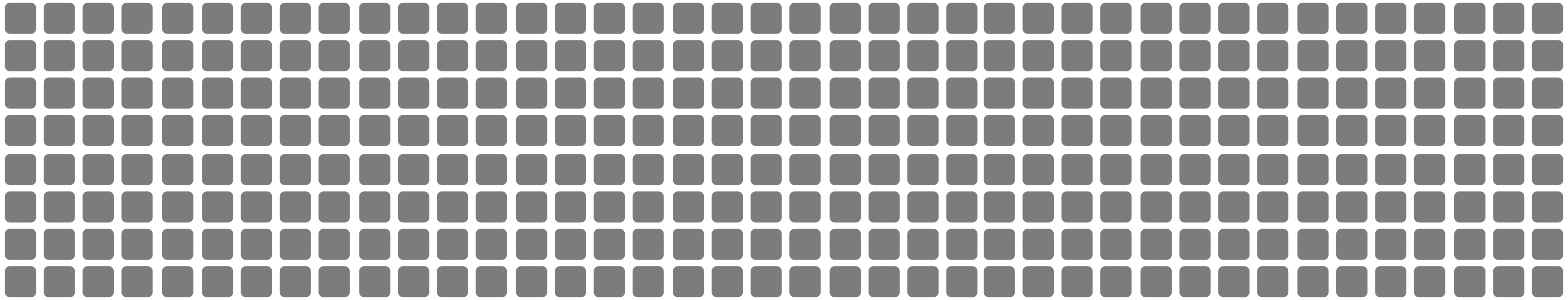
Rank and choose next configuration

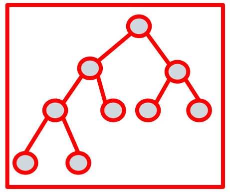
Return Config

Taken from Dr. Nando de Freitas (tiny.cc/4tgeny)

Workflow of Flash

Configuration Space



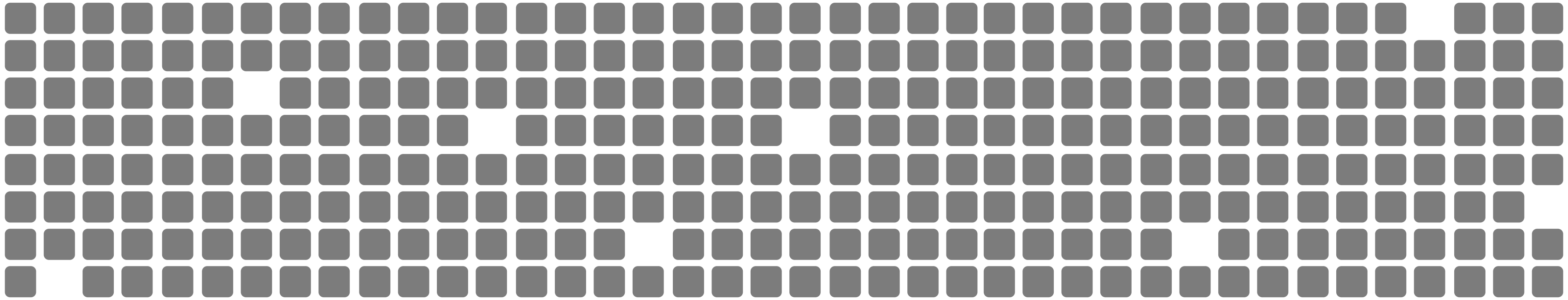


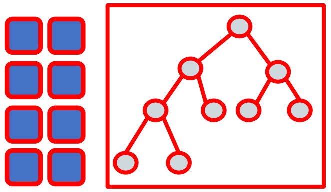
CART

Measurements = 0

Workflow of Flash

Configuration Space

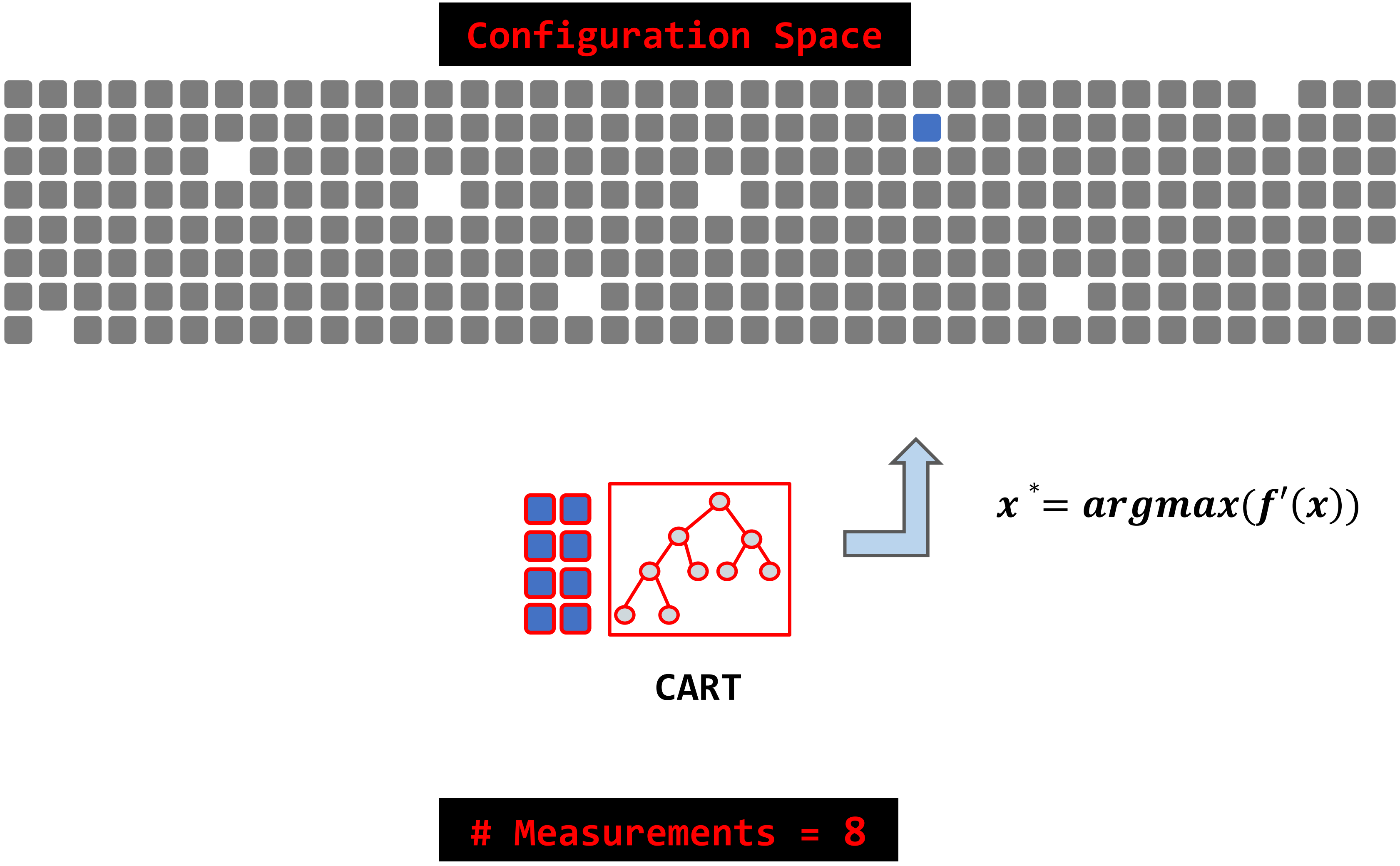




CART

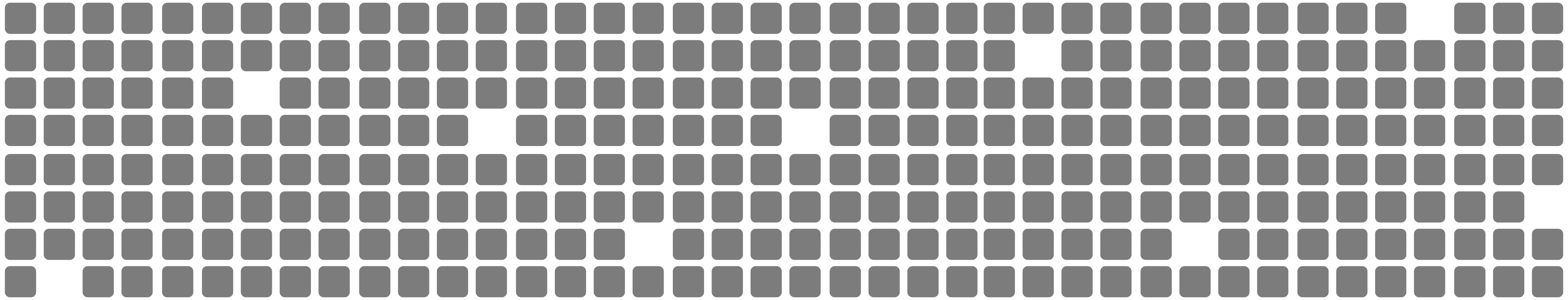
Measurements = 8

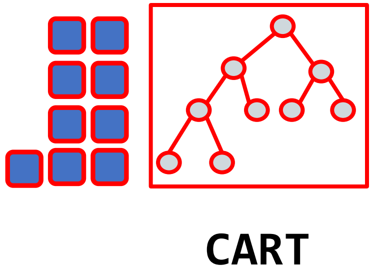
Workflow of Flash



Workflow of Flash

Configuration Space



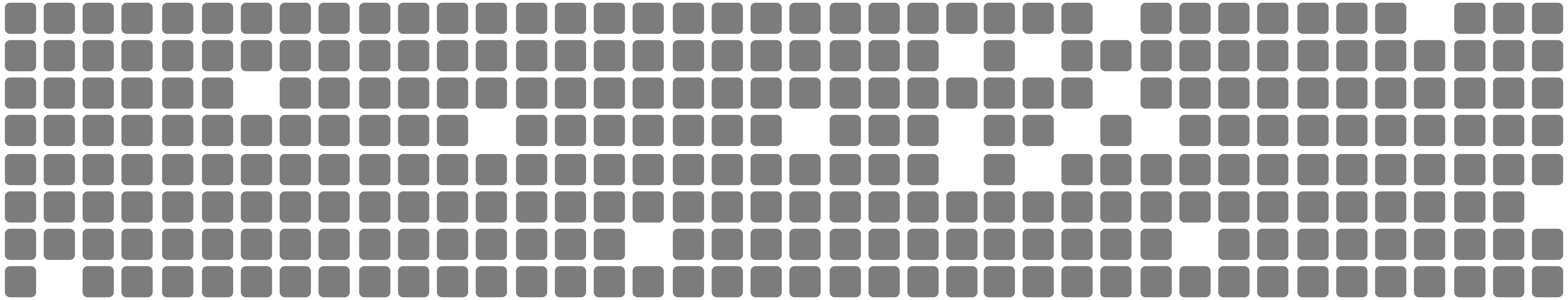


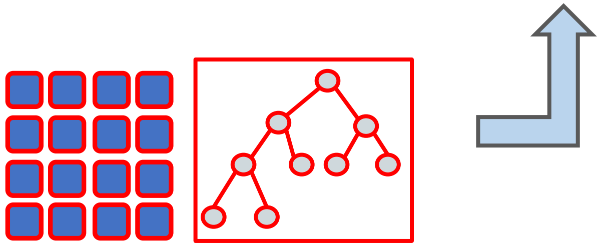
$$x^{*} = argmax(f'(x))$$

 # Measurements = 9

Workflow of Flash

Configuration Space





CART

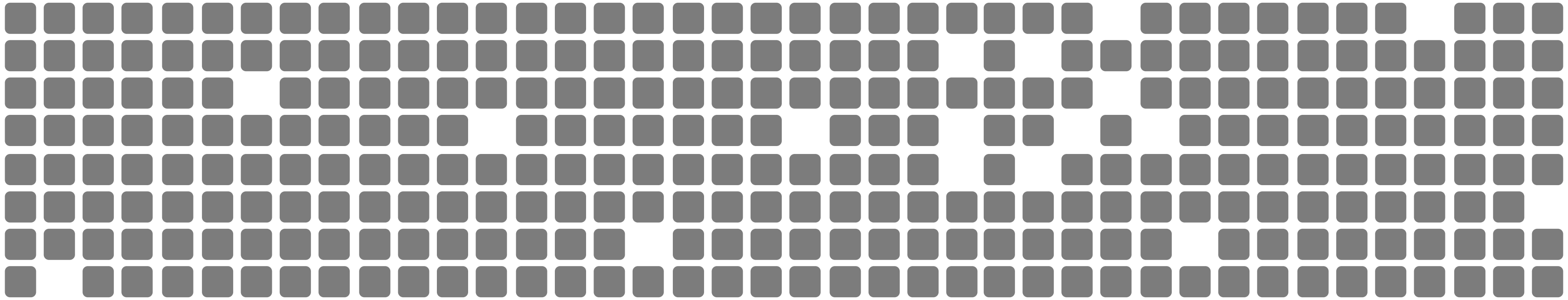
$$x^* = argmax(f'(x))$$

Measurements = 16

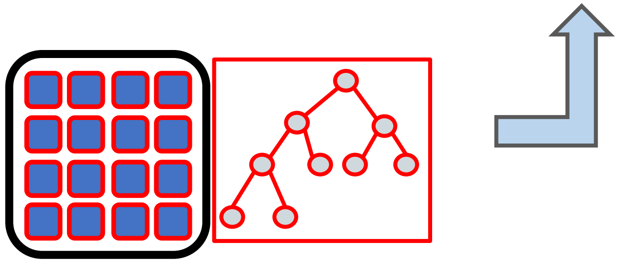
Flash (SMBO)

Workflow of Flash

Configuration Space



Budget



CART

$$x^{*} = argmax(f'(x))$$

Measurements = 16

Research Questions

RQ1 Can FLASH find the good configuration?

RQ2 How expensive is FLASH?

Research Questions

Quality	RQ1	Can FLASH find the good configuration?
Cost	RQ2	How expensive is FLASH?

Baselines

Residual based Method

Sequentially (randomly) sample configuration to build a decision tree till threshold accuracy is reached

Rank based Method

Sequentially (randomly) sample configurations to build a decision tree which preserves relative ordering

Baselines

Guo et al., 2013

Residual based Method

Sequentially (randomly) sample configuration to build a decision tree till threshold accuracy is reached

Nair et al., 2017

Rank based Method

Sequentially (randomly) sample configurations to build a decision tree which preserves relative ordering

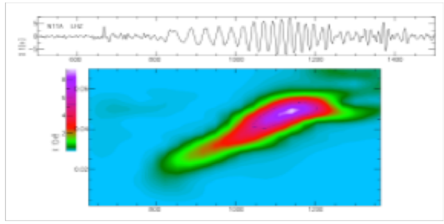
Subject Systems

Data Processing

Mesh Solver

FPGA

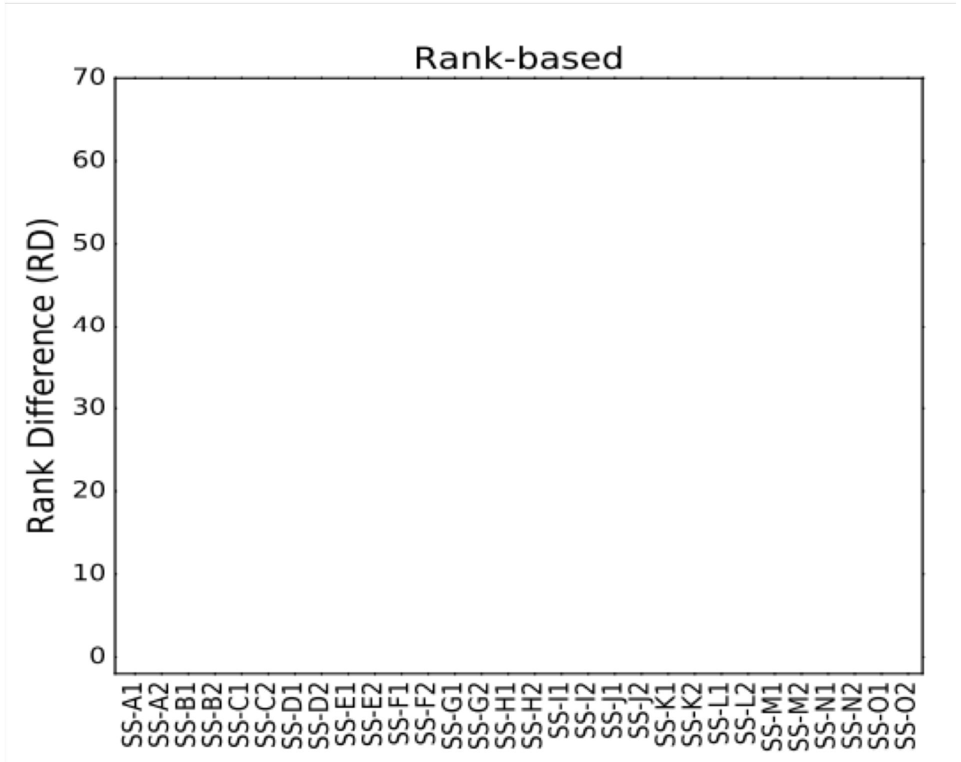
Seismic Analysis



Compiler

Video Encoder

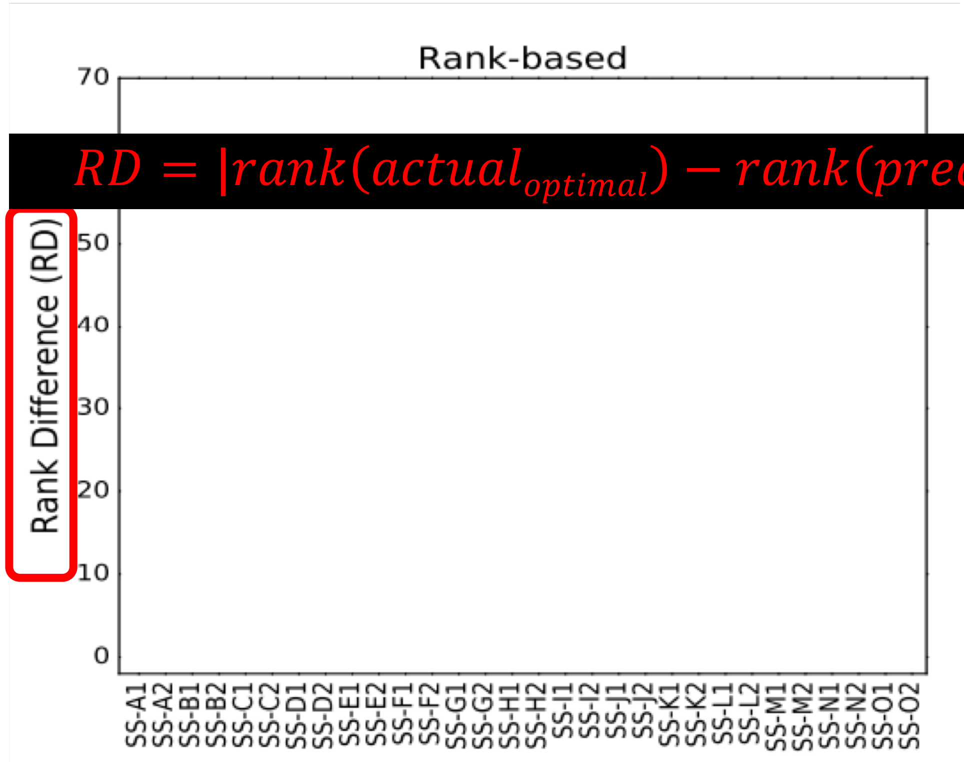
RQ1: Can FLASH find the good configuration?



Flash (SMBO)

RQ1: Can FLASH find the good configuration?

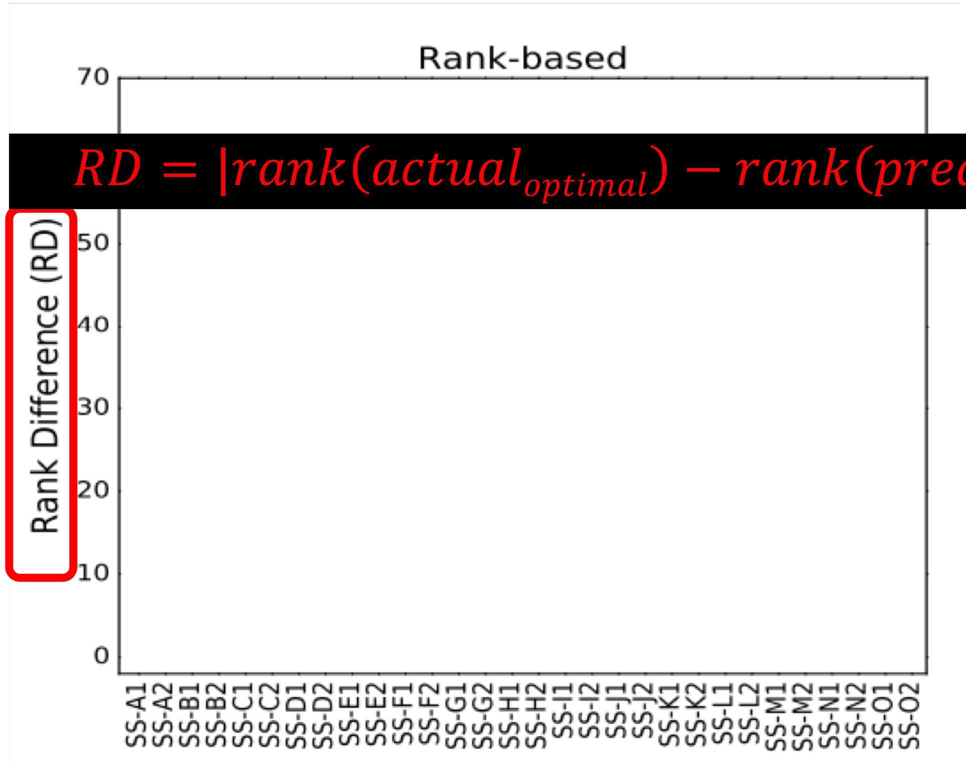
$$RD = |rank(actual_{optimal}) - rank(predicted_{optimal})|$$

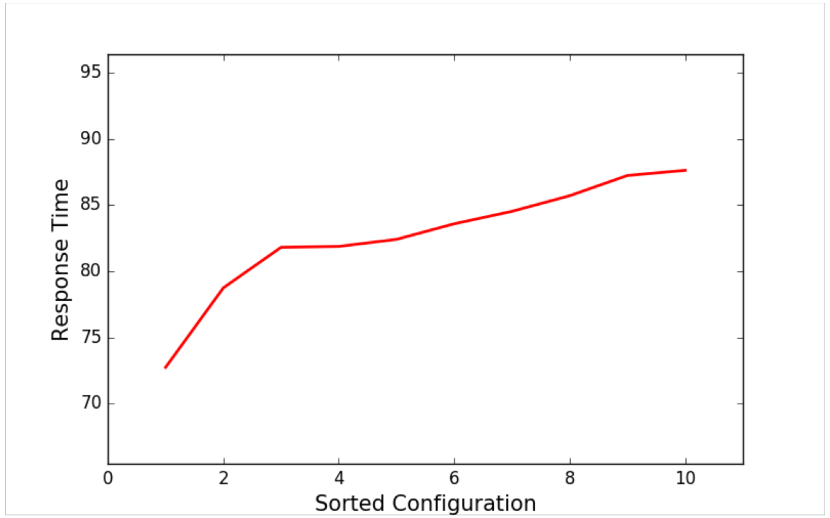


Flash (SMBO)

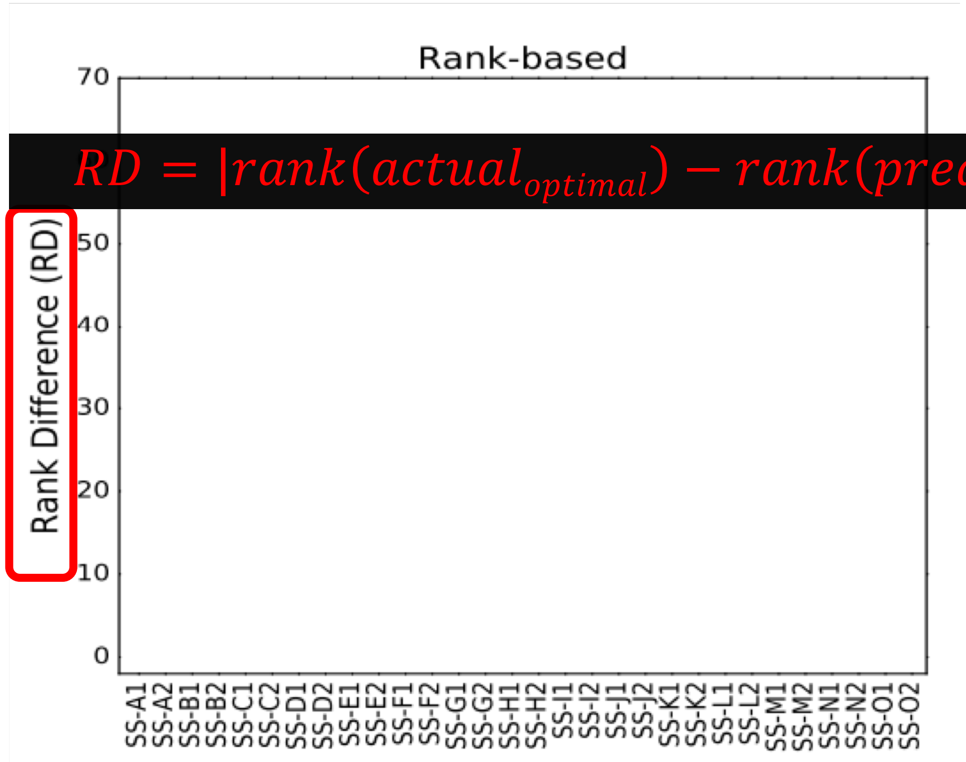
RQ1: Can FLASH find the good configuration?

$$RD = |rank(actual_{optimal}) - rank(predicted_{optimal})|$$

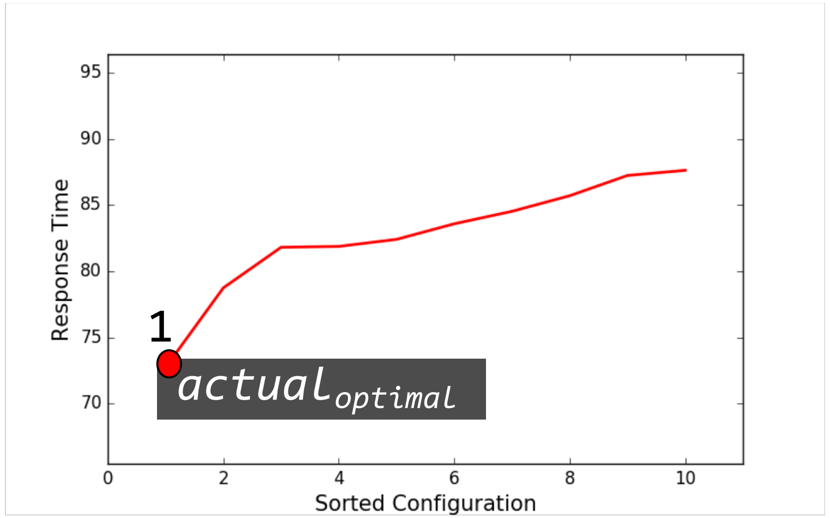




RQ1: Can FLASH find the good configuration?

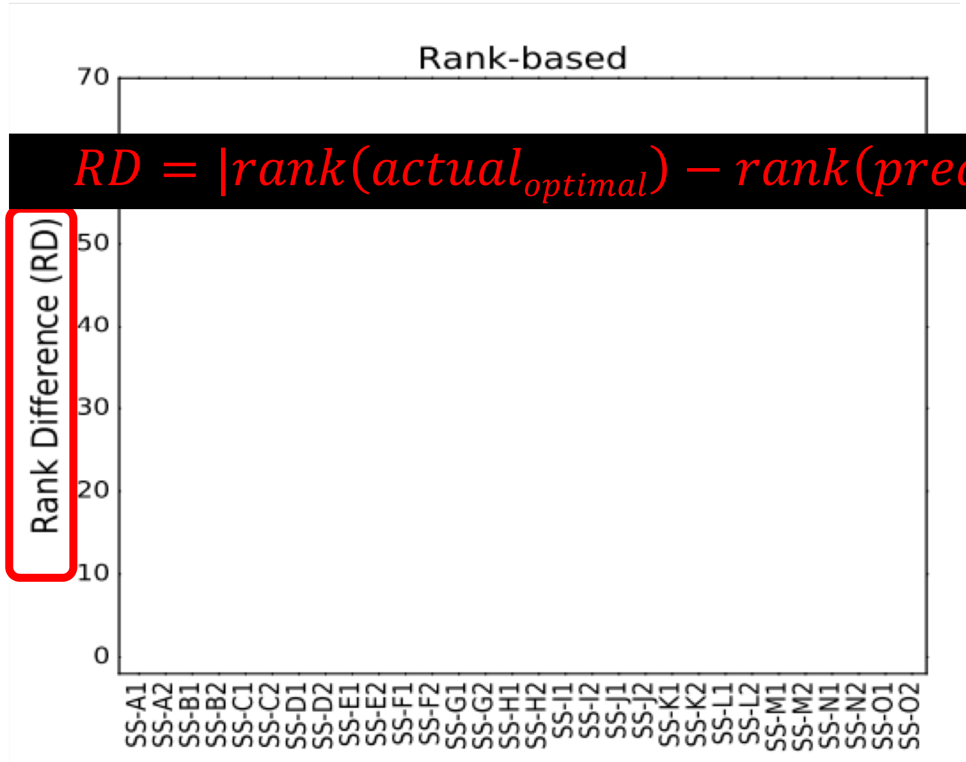


$$RD = |rank(actual_{optimal}) - rank(predicted_{optimal})|$$

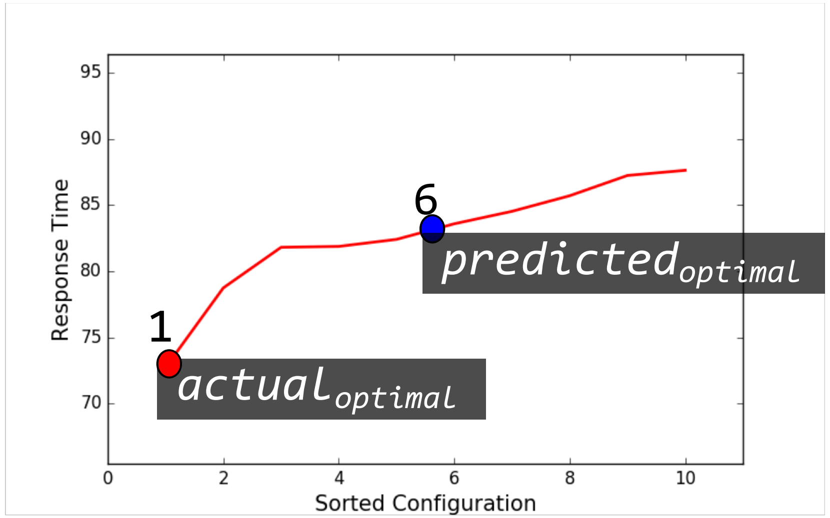


Flash (SMBO)

RQ1: Can FLASH find the good configuration?

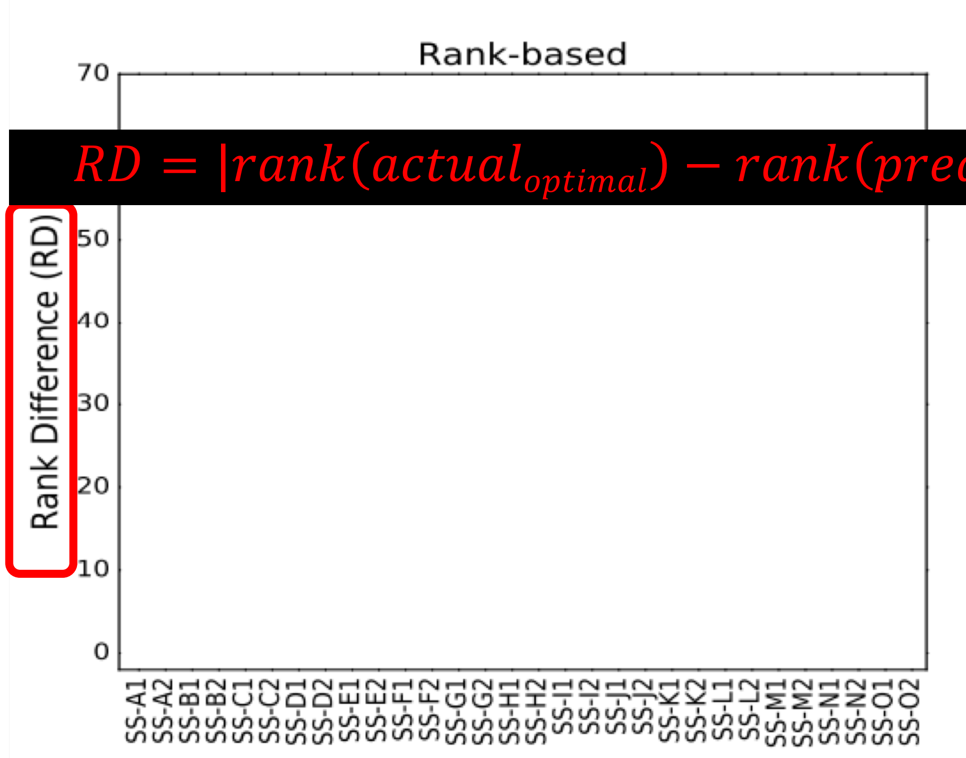


$$RD = |rank(actual_{optimal}) - rank(predicted_{optimal})|$$

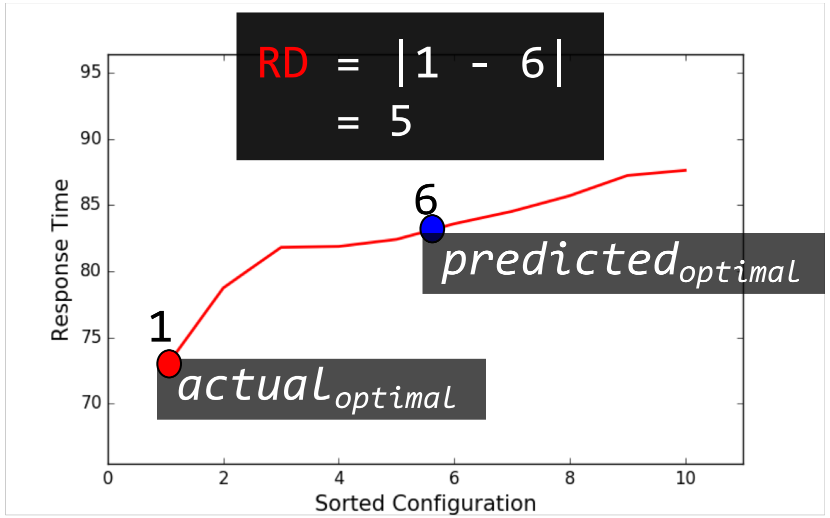


Flash (SMBO)

RQ1: Can FLASH find the good configuration?

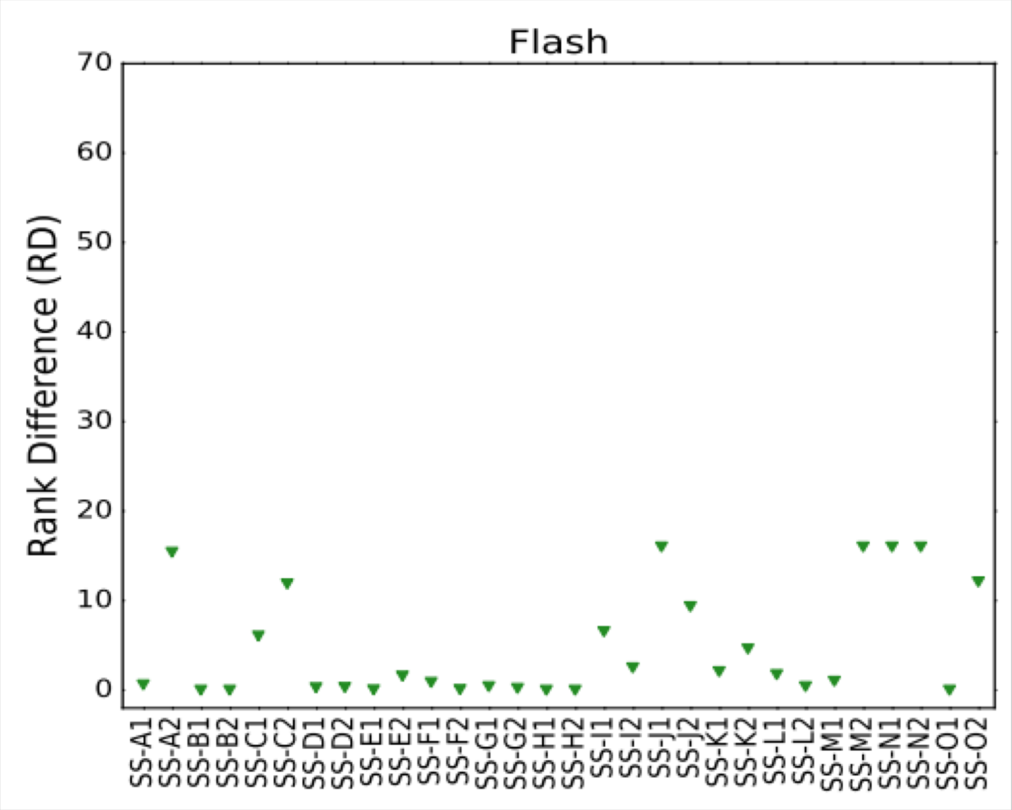


$$RD = |rank(actual_{optimal}) - rank(predicted_{optimal})|$$

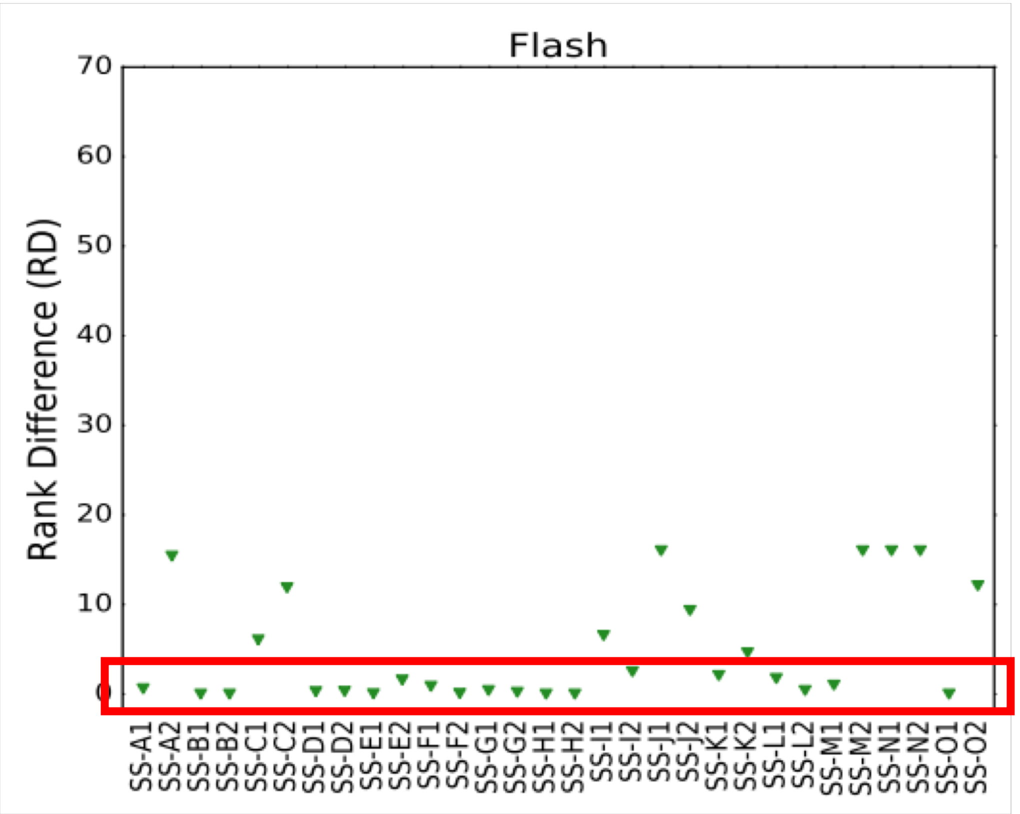


Flash (SMBO)

RQ1: Can FLASH find the good configuration?

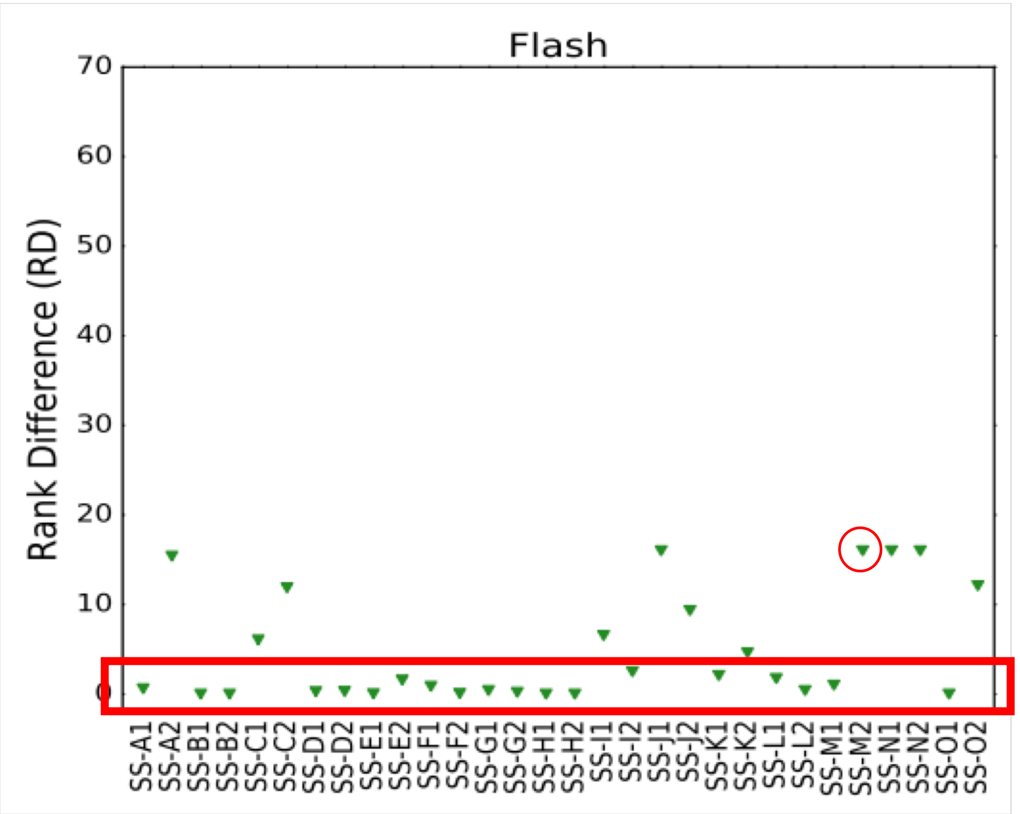


RQ1: Can FLASH find the good configuration?



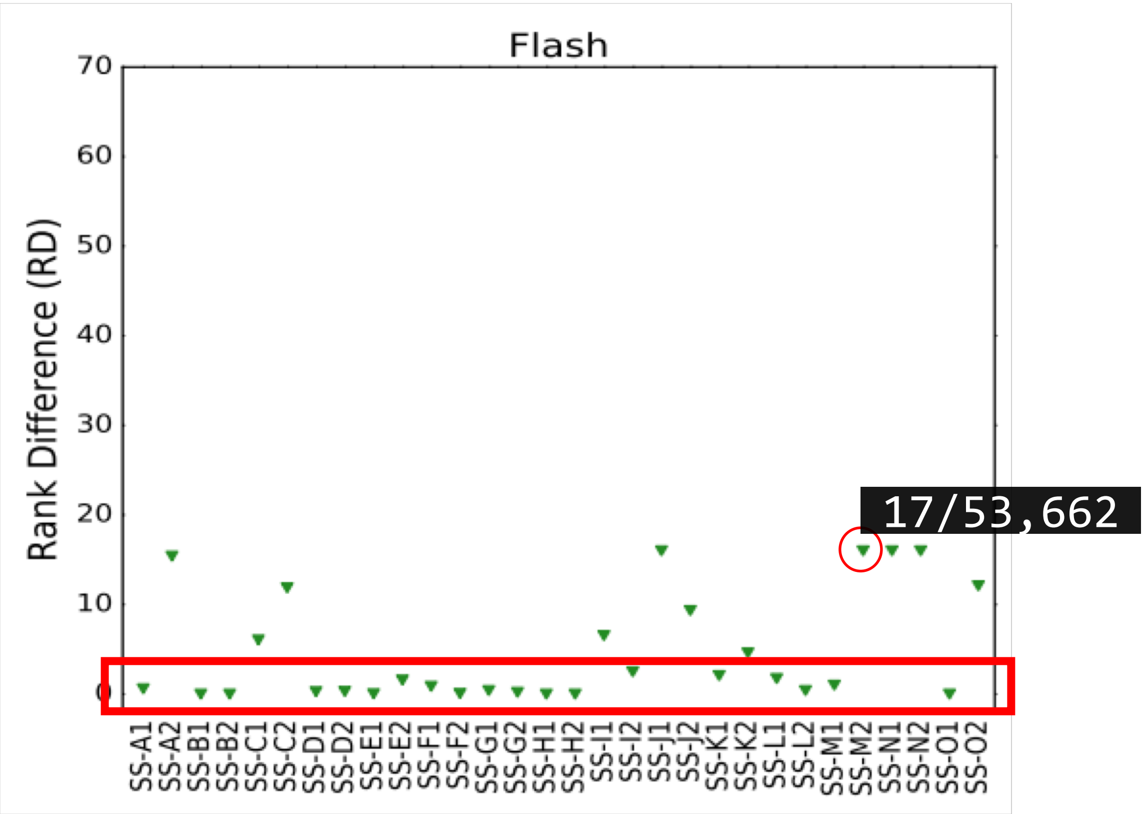
Flash (SMBO)

RQ1: Can FLASH find the good configuration?



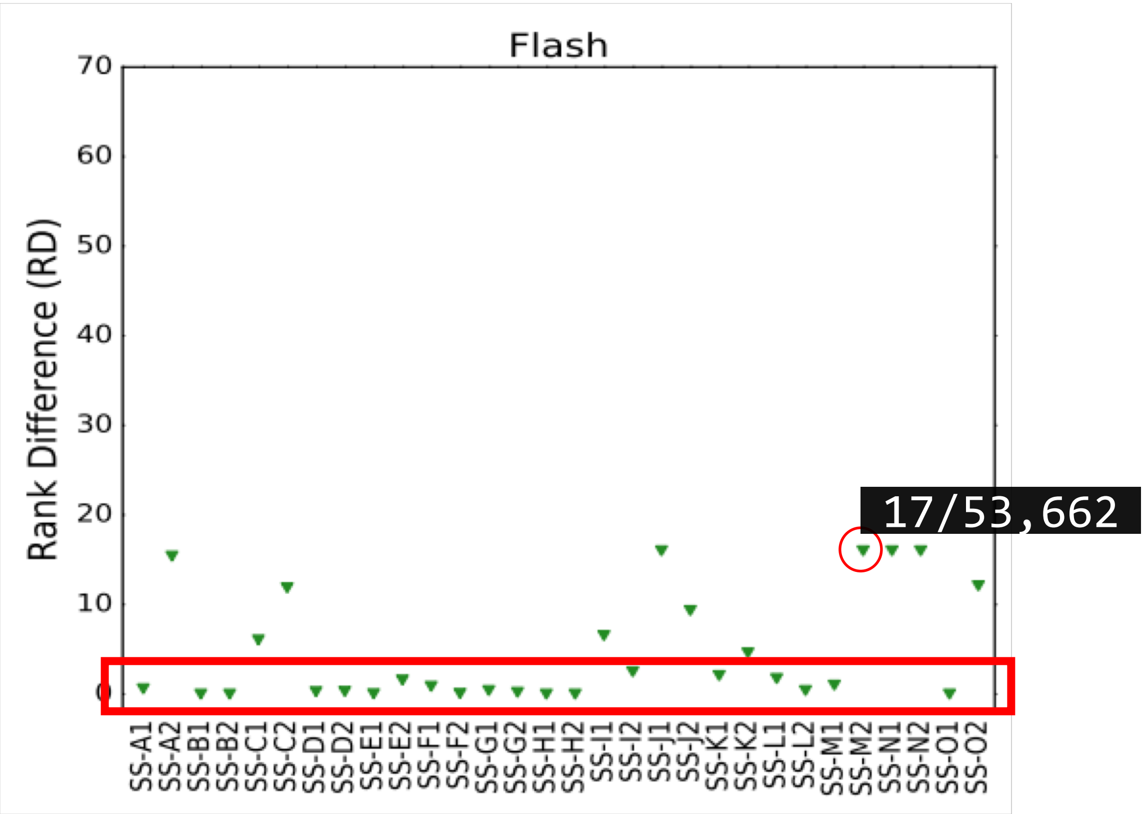
Flash (SMBO)

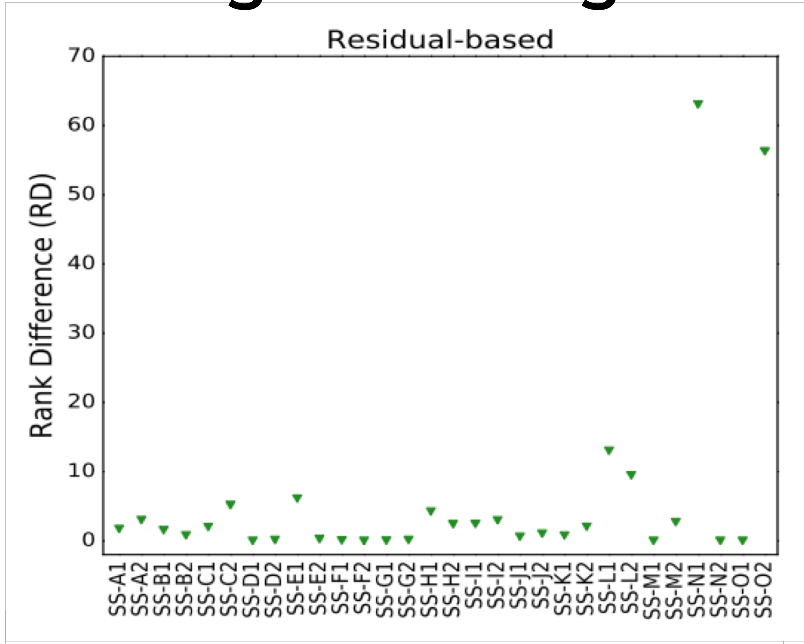
RQ1: Can FLASH find the good configuration?

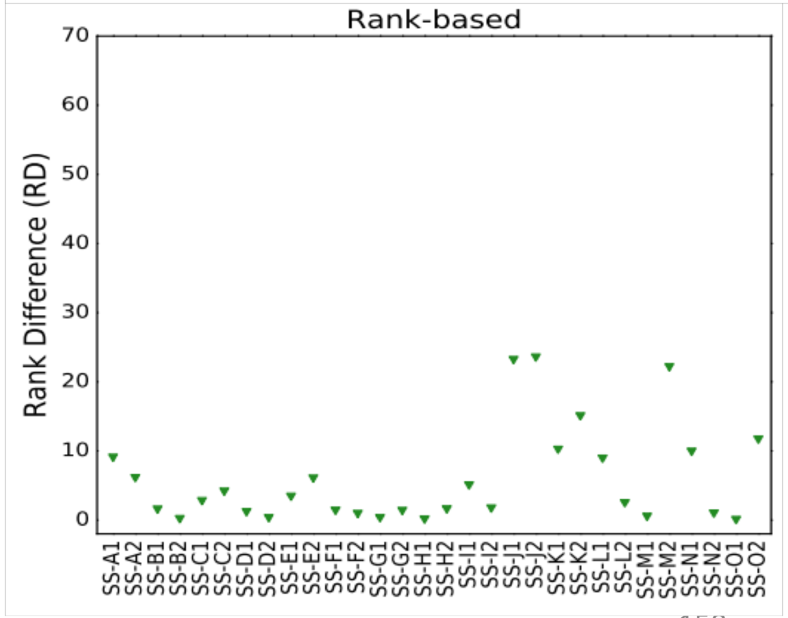


Flash (SMBO)

RQ1: Can FLASH find the good configuration?



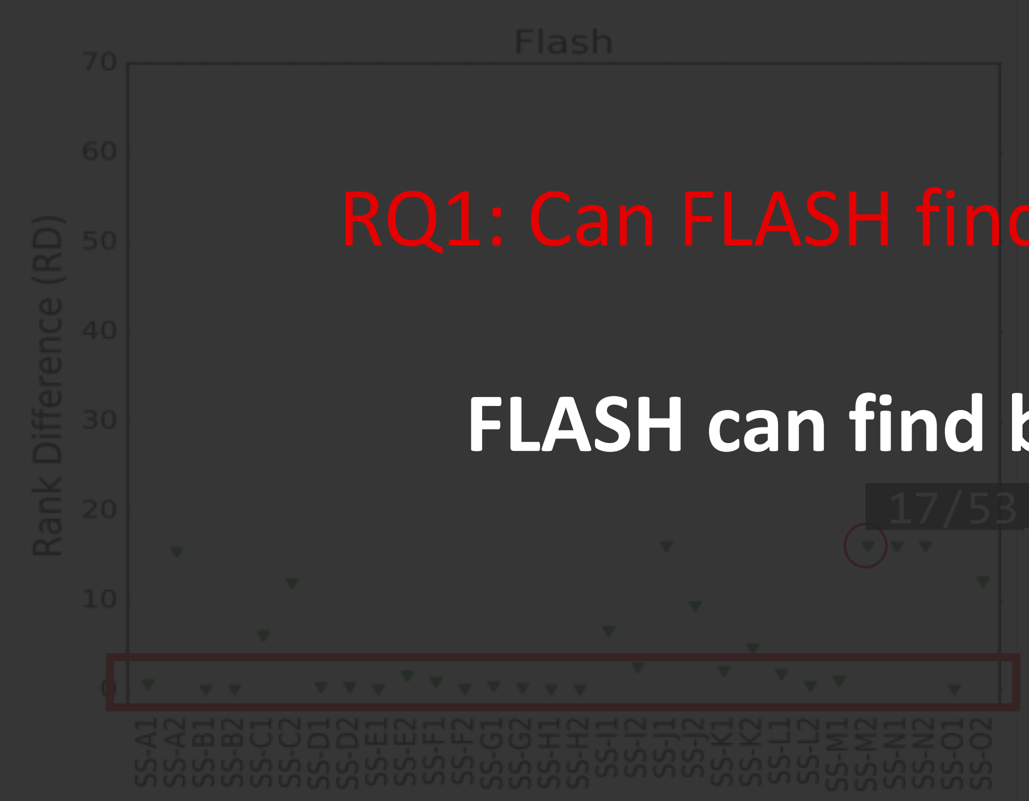


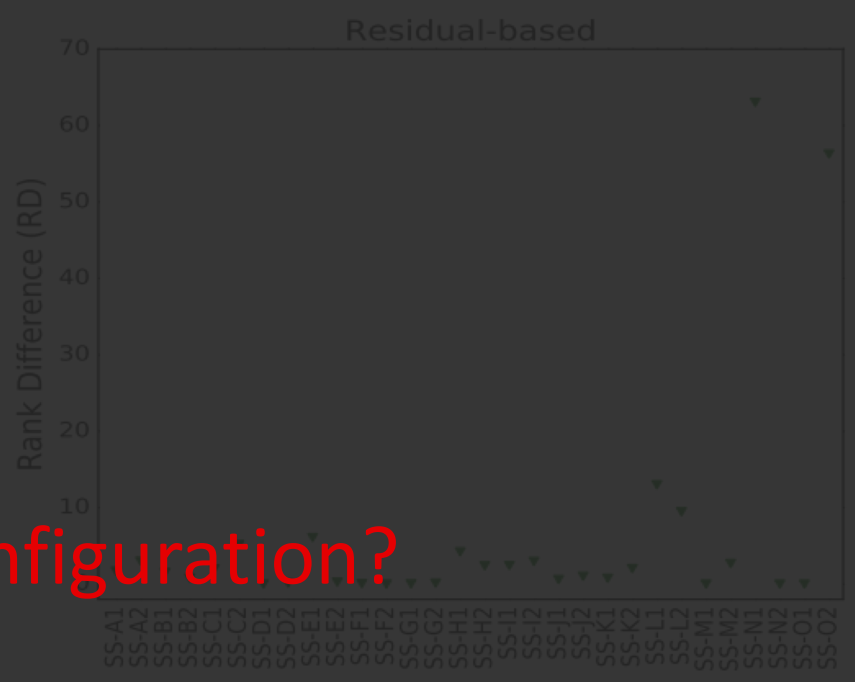


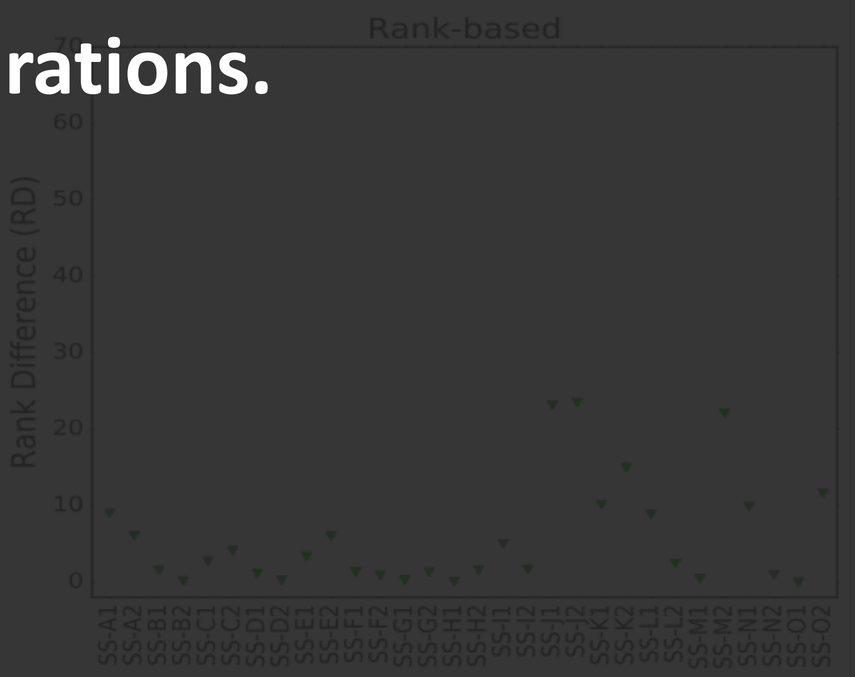
RQ1: Can FLASH find the good configuration?

FLASH can find better configurations.

RQ1: Can FLASH find the good configuration?



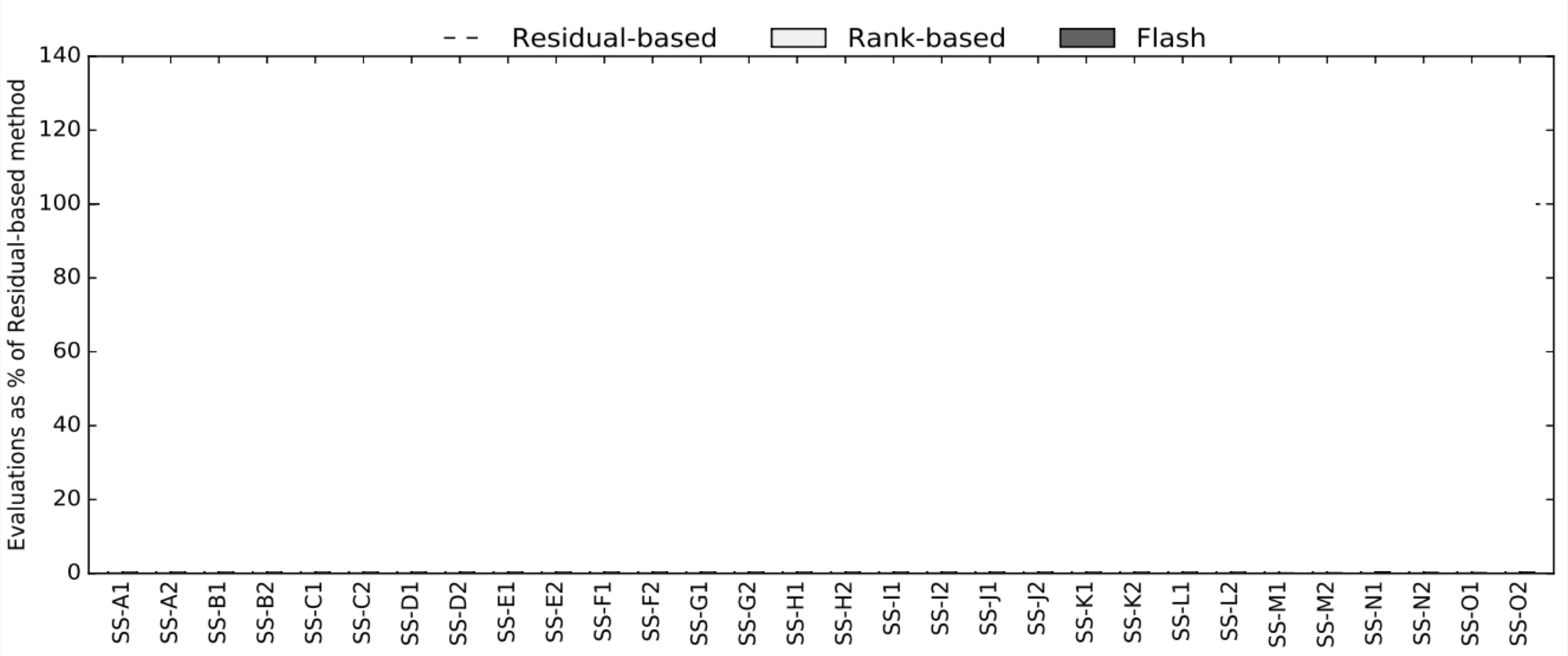




RQ2: How expensive is FLASH?

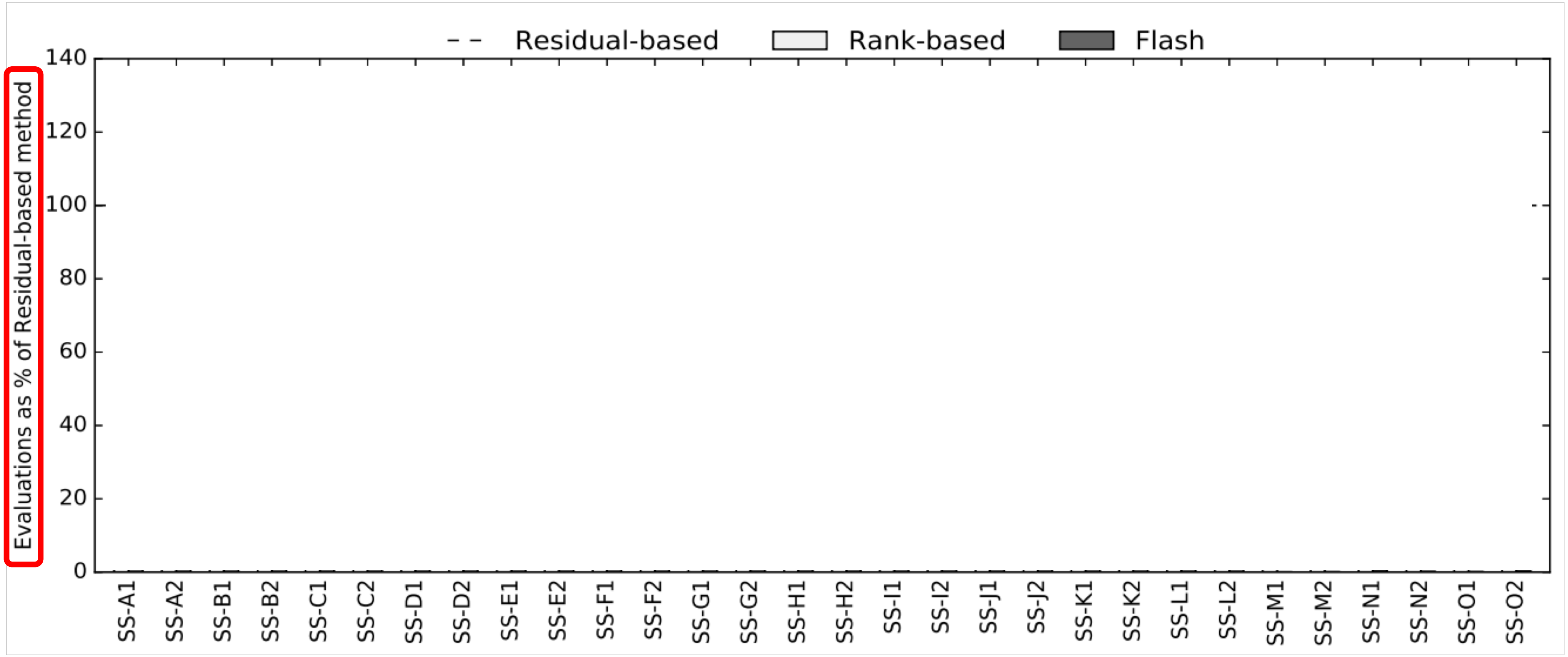
Flash (SMBO)

RQ2: How expensive is FLASH?



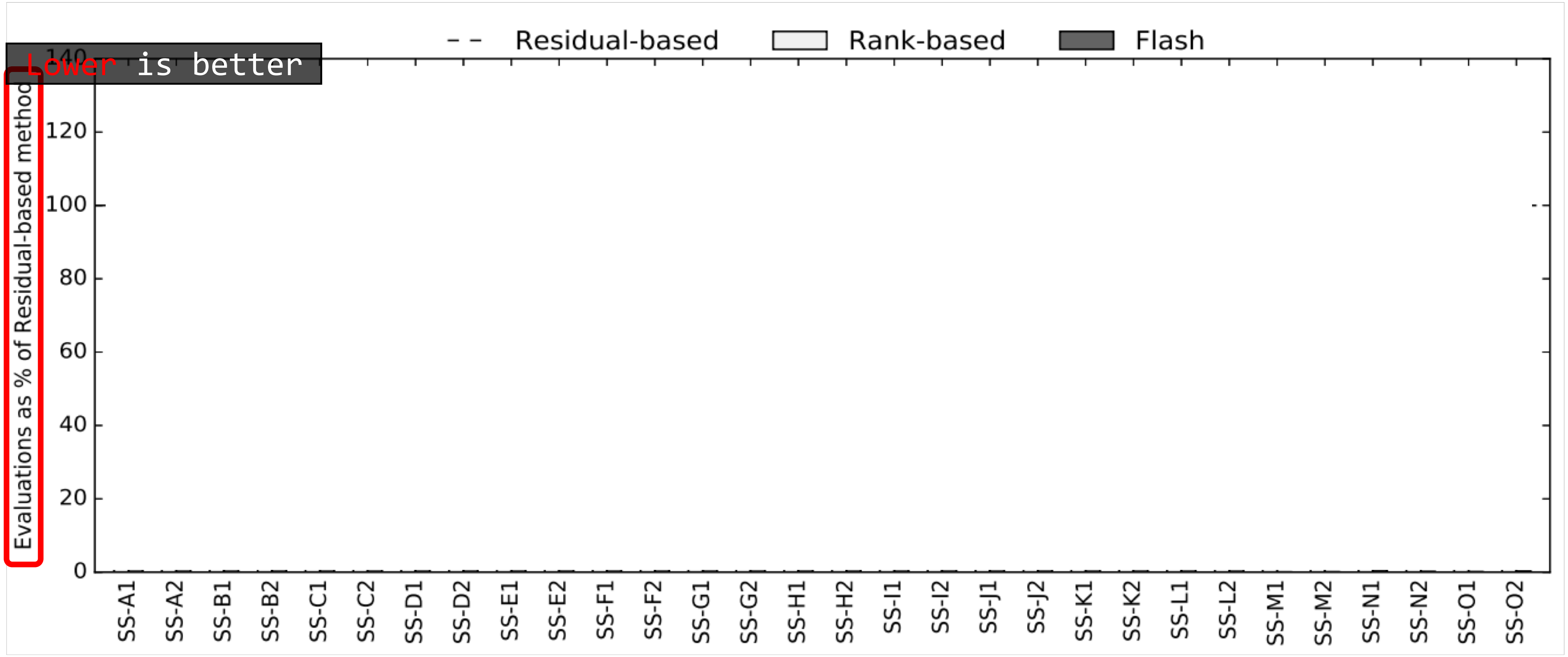
Flash (SMBO)

RQ2: How expensive is FLASH?

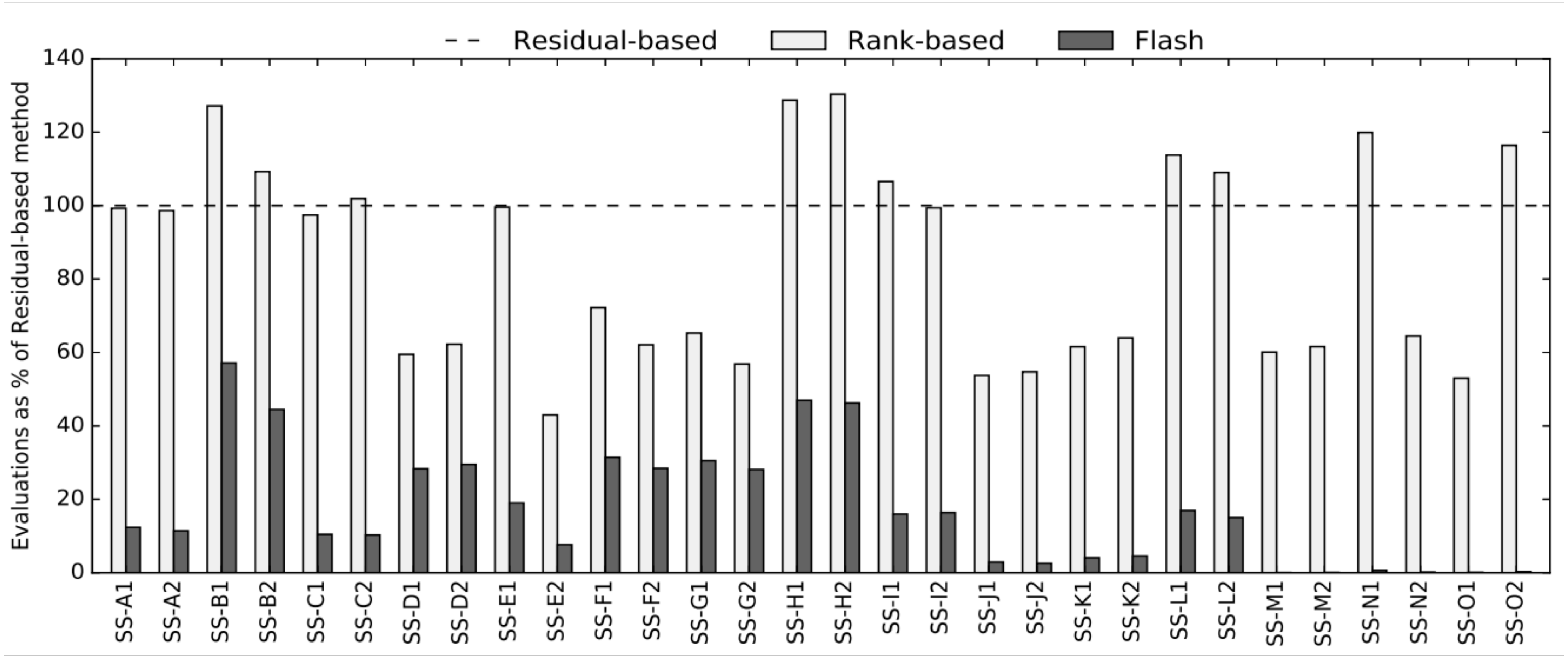


Flash (SMBO)

RQ2: How expensive is FLASH?

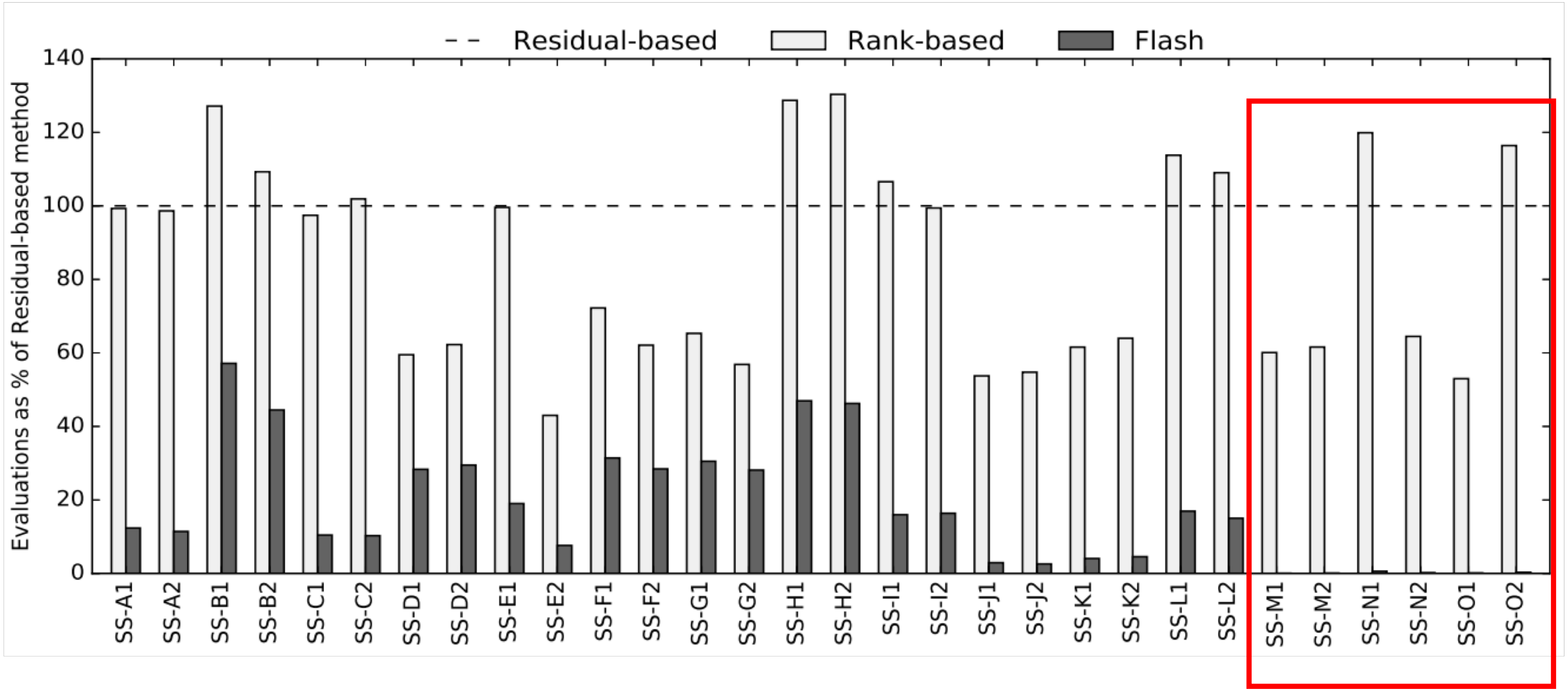


RQ2: How expensive is FLASH?

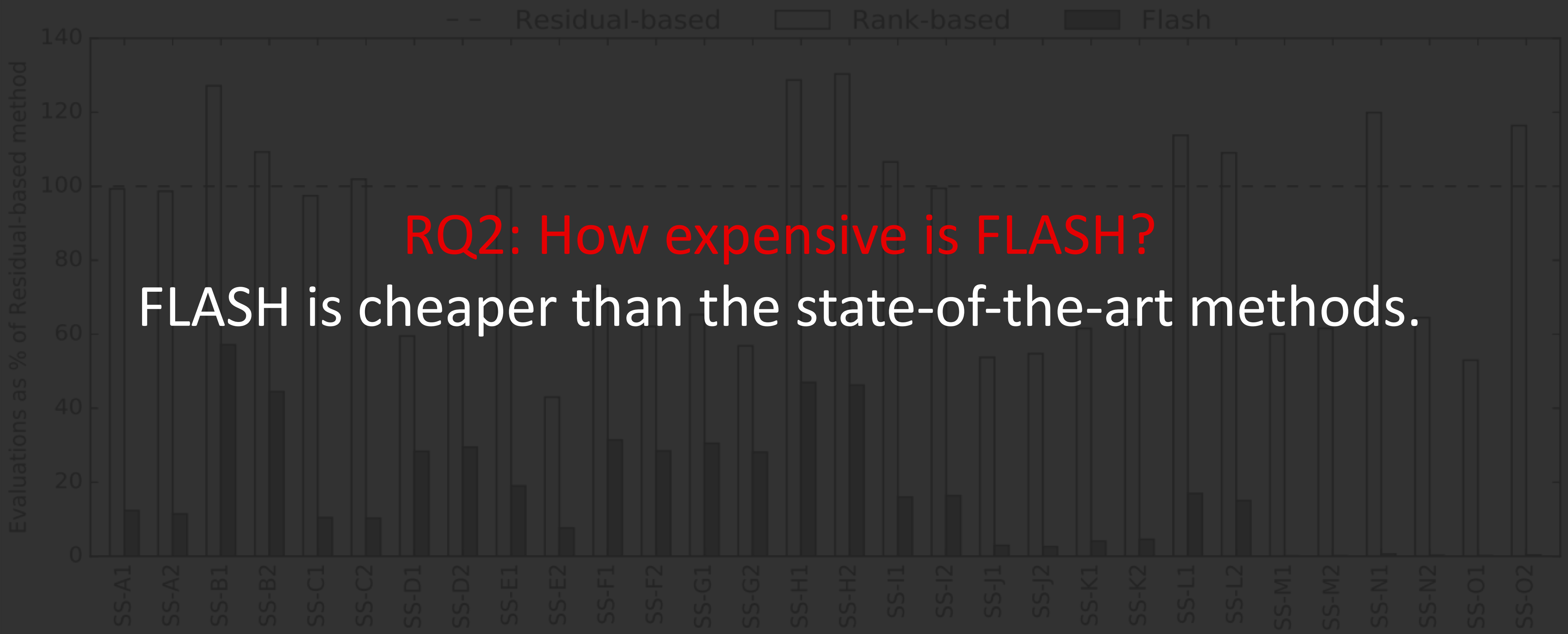


Flash (SMBO)

RQ2: How expensive is FLASH?



RQ2: How expensive is FLASH?



RQ2: How expensive is FLASH?

FLASH is cheaper than the state-of-the-art methods.

FLASH can answer

FLASH can answer

Q. Given a software system, which configuration maximizes the throughput (performance measure) for a given benchmark?

FLASH can answer

Single Objective problem

Q. Given a software system, which configuration maximizes the throughput (performance measure) for a given benchmark?

FLASH can answer

Single Objective problem

Q. Given a software system, which configuration maximizes the throughput (performance measure) for a given benchmark?

FLASH cannot answer

Q. Given a software system, which configuration maximizes the throughput while minimizing latency for a given benchmark?

FLASH can answer

Single Objective problem

Q. Given a software system, which configuration maximizes the throughput (performance measure) for a given benchmark?

FLASH cannot answer

Multi-Objective problem

Q. Given a software system, which configuration maximizes the throughput while minimizing latency for a given benchmark?

Flash (SMBO)

FLASH can answer

Single Objective problem

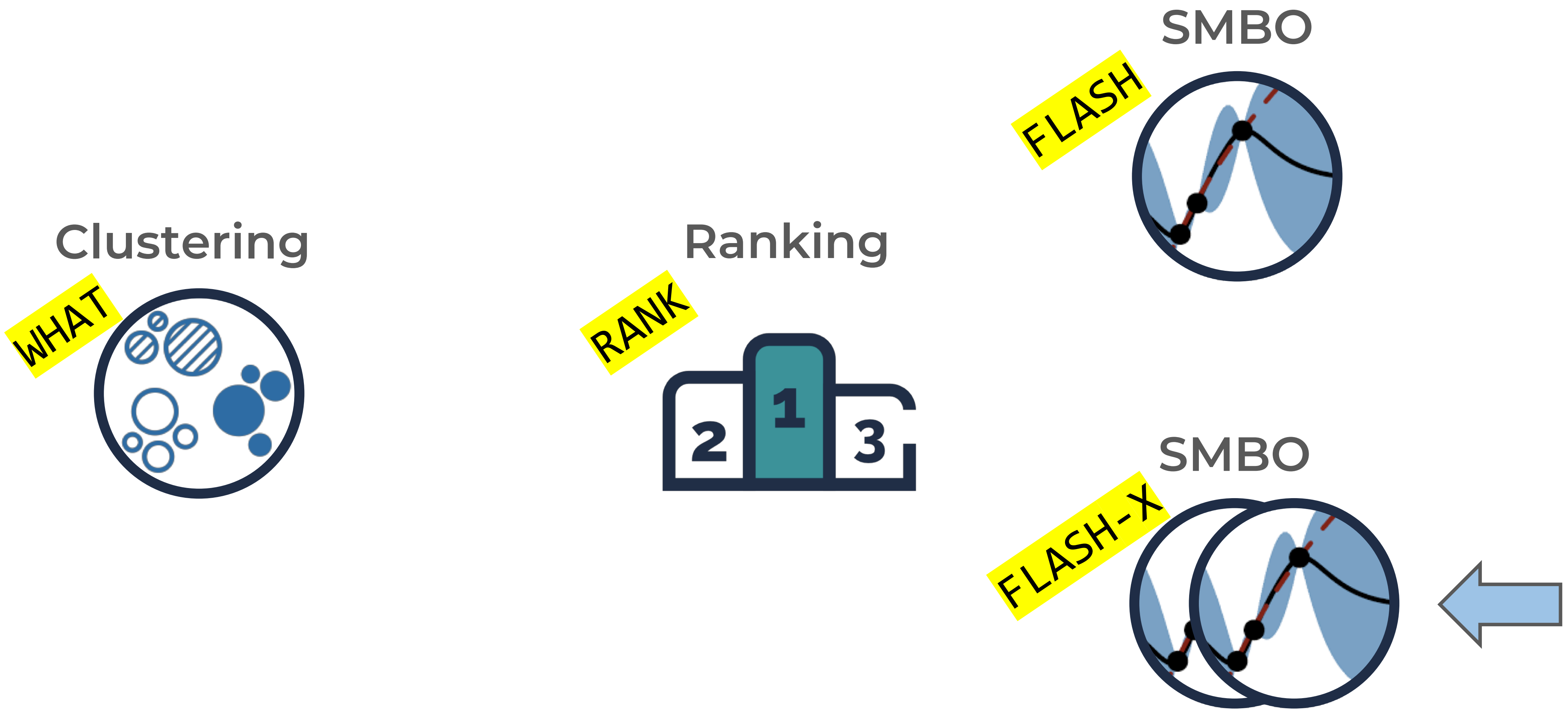
Q. Given a software system, which configuration maximizes the throughput (performance measure) for a given benchmark?

FLASH cannot answer

Multi-Objective problem

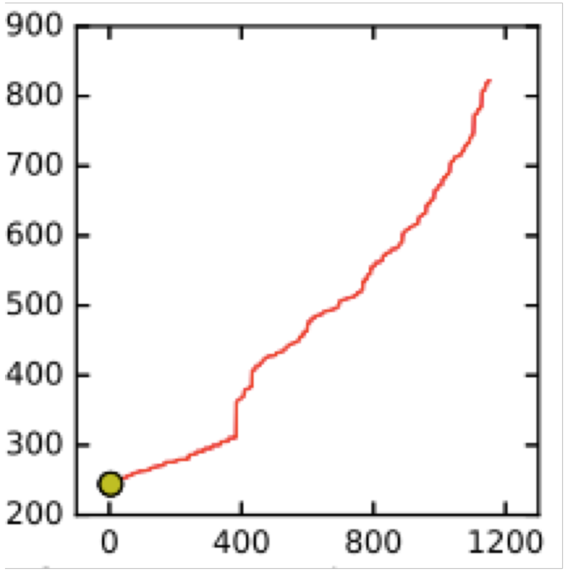
Q. Given a software system, which configuration maximizes the throughput while minimizing latency for a given benchmark?

How can FLASH be modified to solve multi objective (MO) problems?



Nair et al.; Finding faster configurations using Flash; TSE (2018)

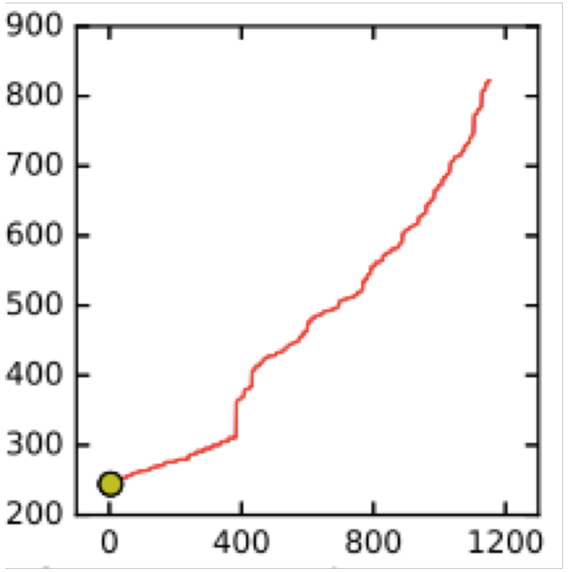
What is a MO Problem?



Single Objective Problems

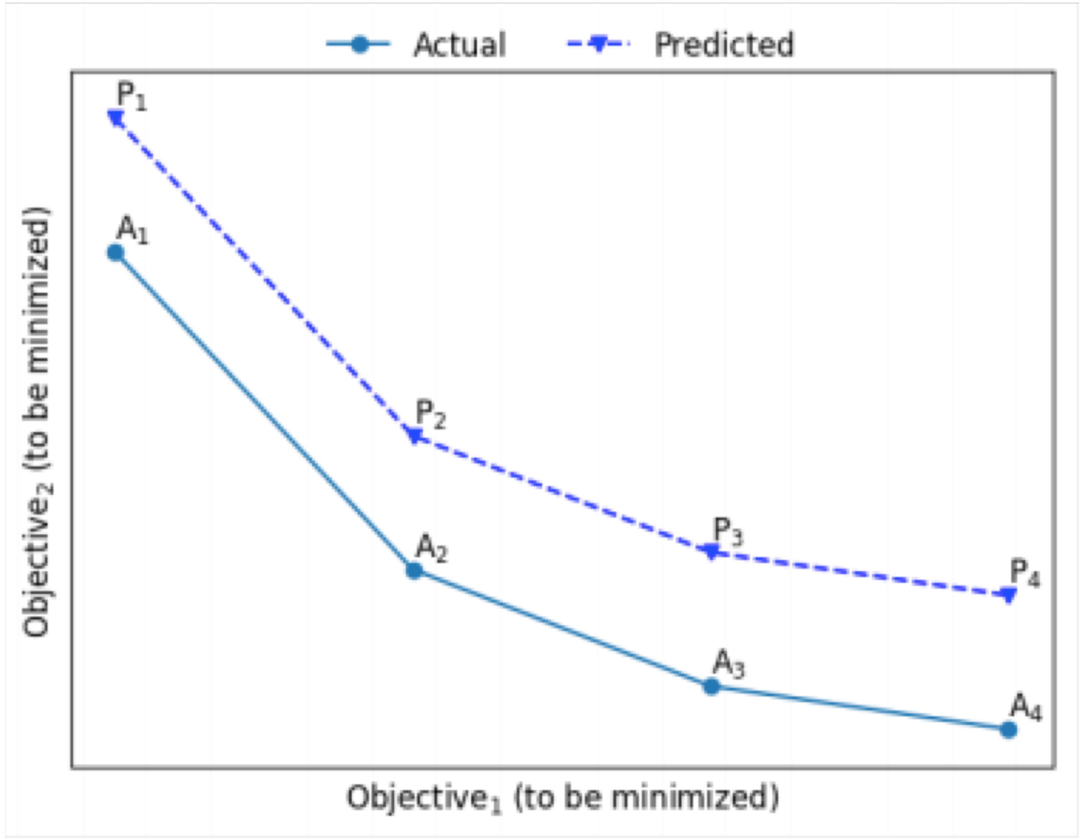
Single ‘best’ solution

What is a MO Problem?



Single Objective Problems

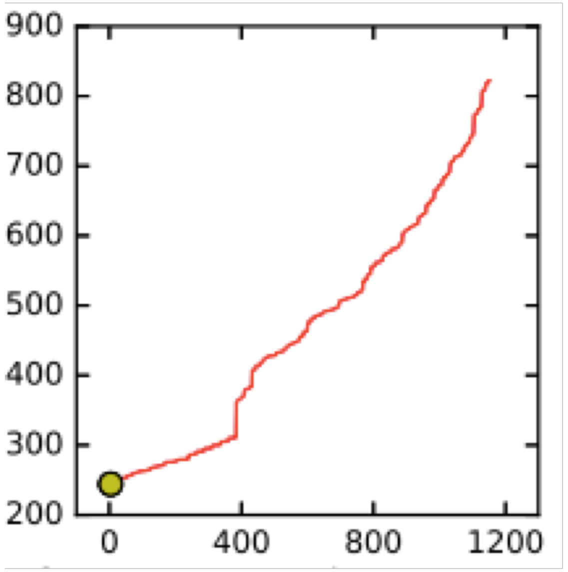
Single ‘best’ solution



Multi-Objective Problems

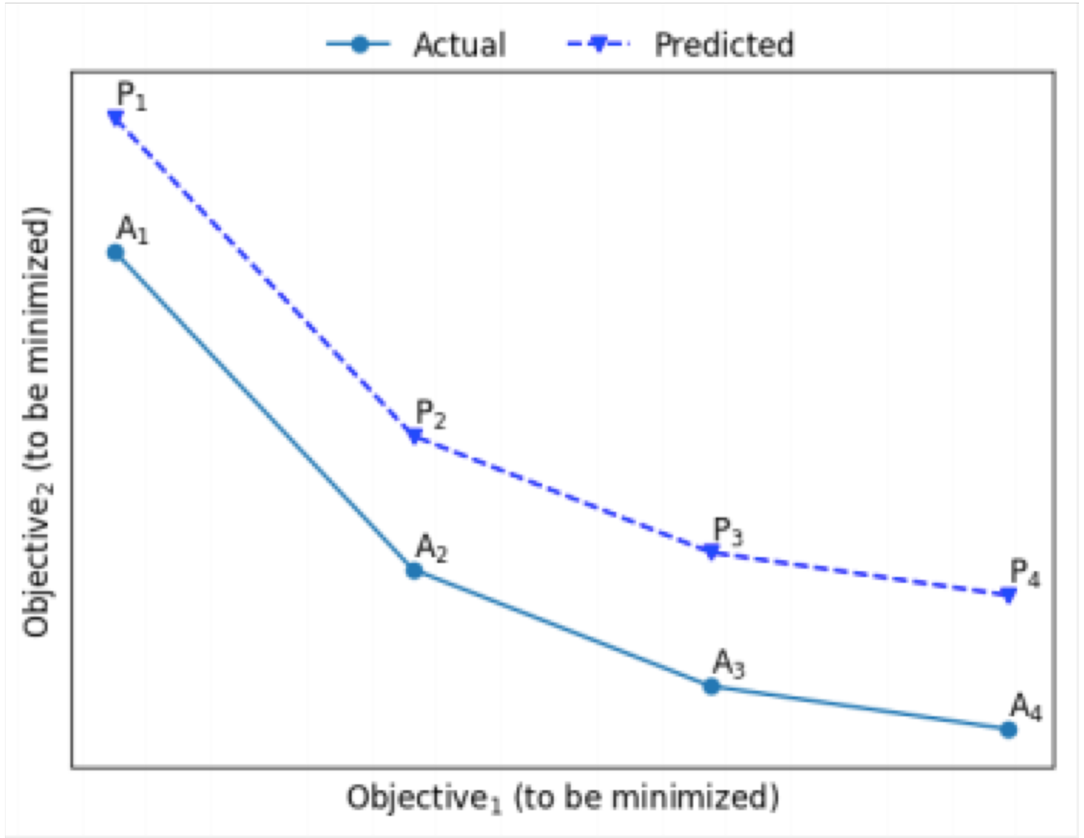
- No single ‘best’ solution
- Number of ‘best’ solutions

What is a MO Problem?



Single Objective Problems

Single 'best' solution

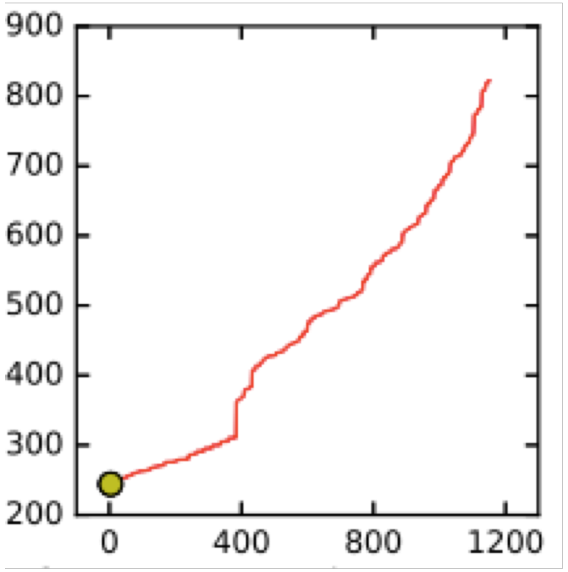


Multi-Objective Problems

- No single 'best' solution
- Number of 'best' solutions

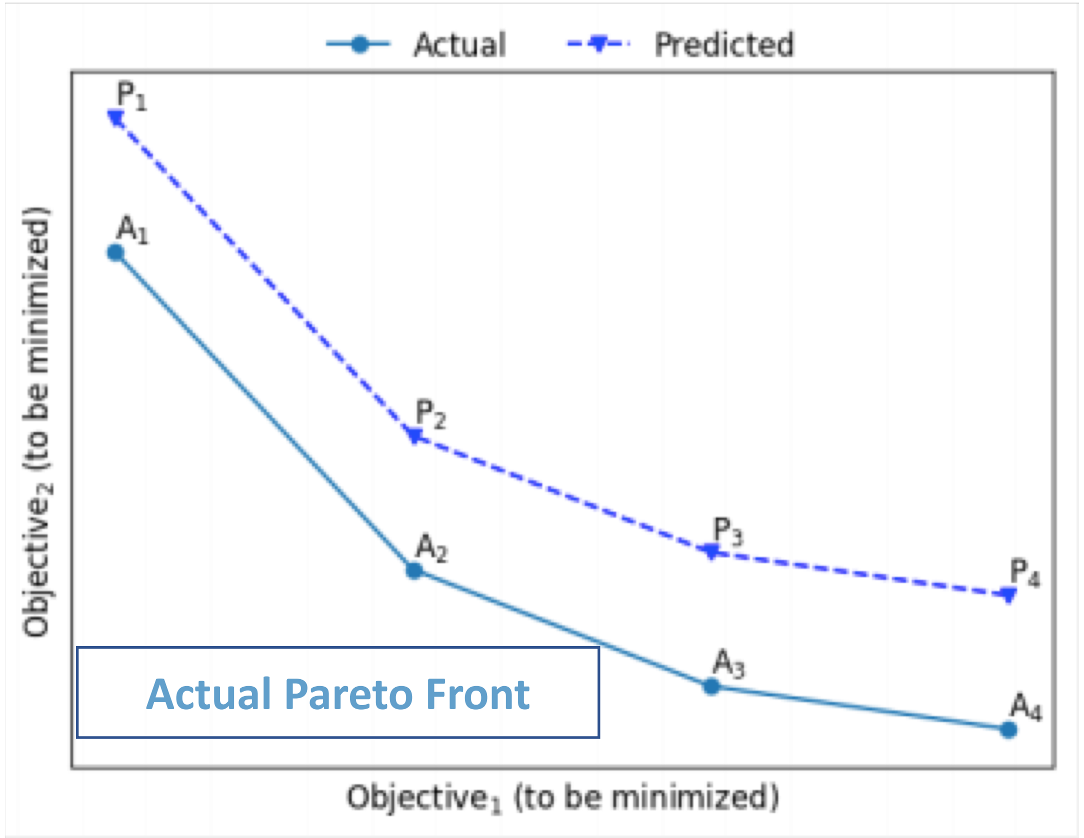
Pareto Front

What is a MO Problem?



Single Objective Problems

Single 'best' solution

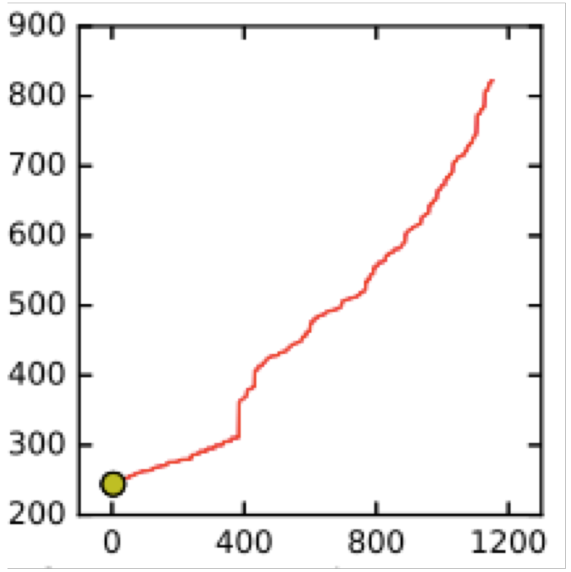


Multi-Objective Problems

- No single 'best' solution
- Number of 'best' solutions

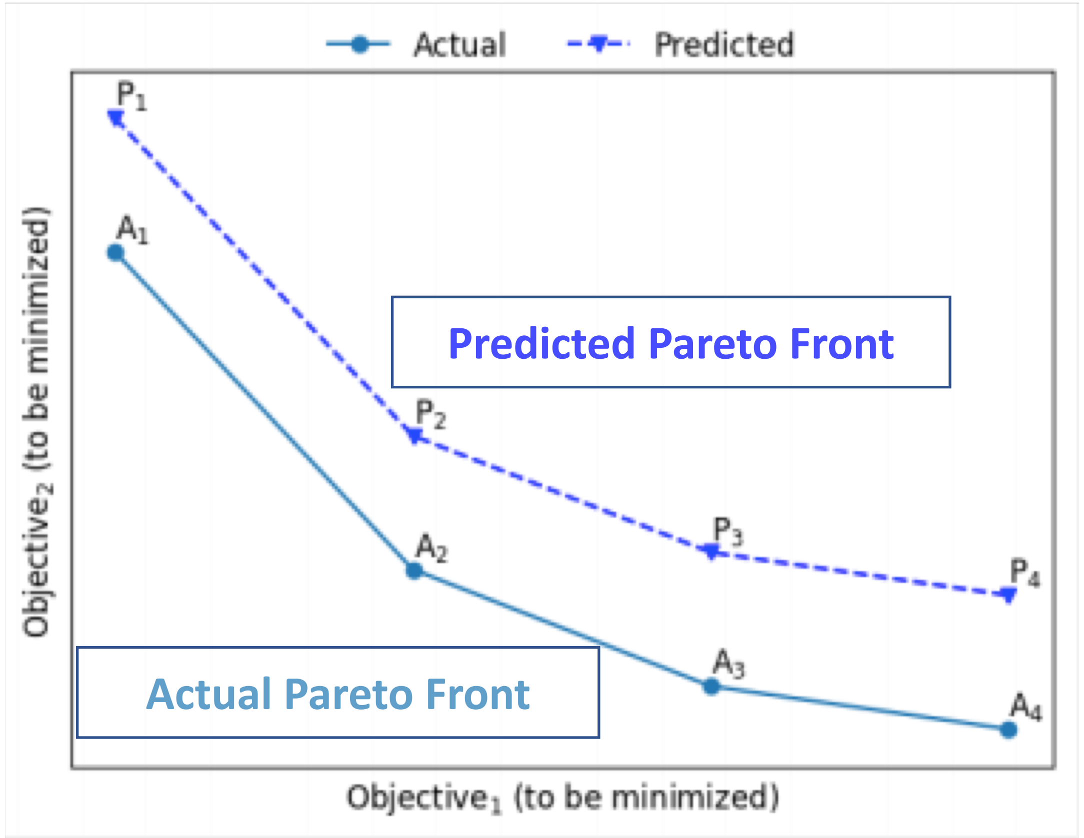
Pareto Front

What is a MO Problem?



Single Objective Problems

Single 'best' solution

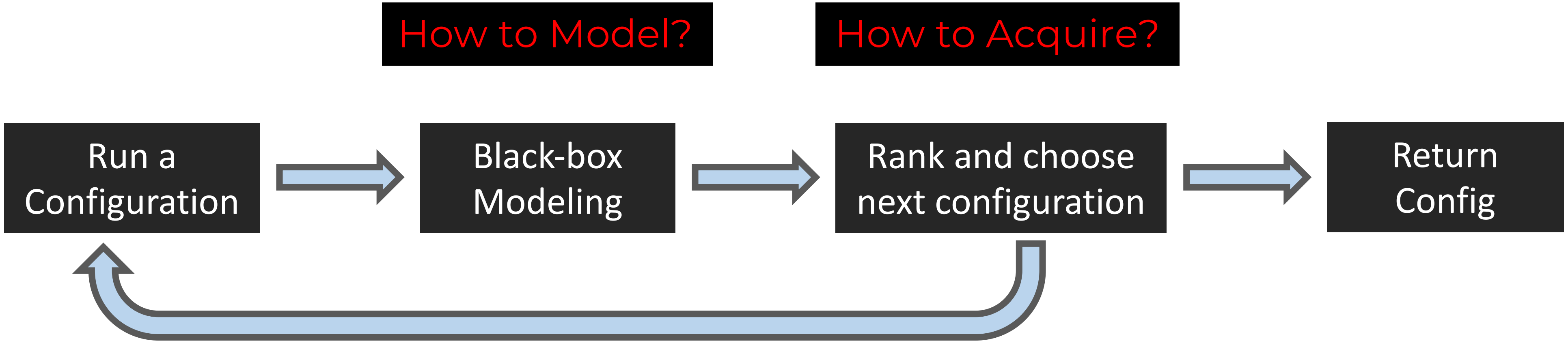


Multi-Objective Problems

- No single 'best' solution
- Number of 'best' solutions

Pareto Front

Workflow of Flash



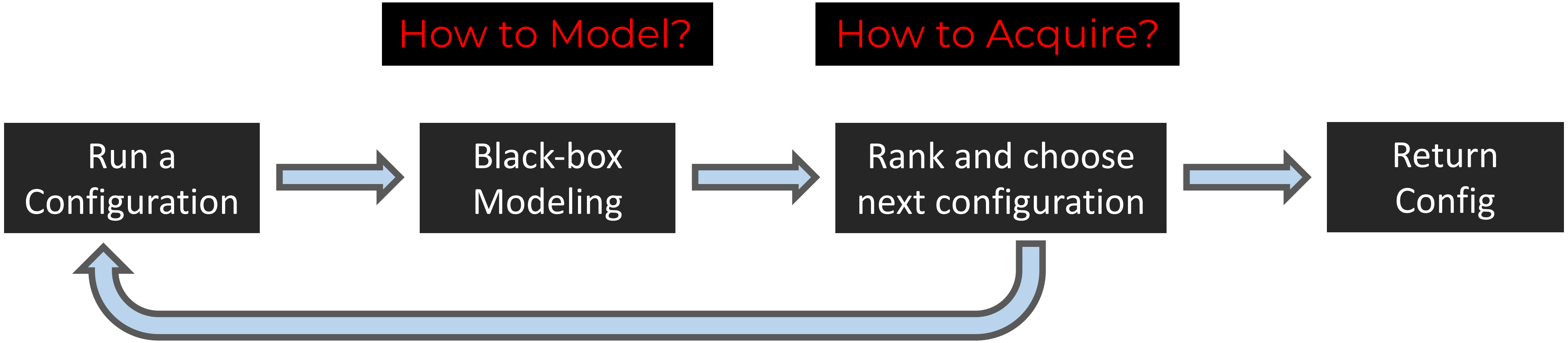
Single Objective

CART

$\mu(x)$

Flash-X (SMBO)

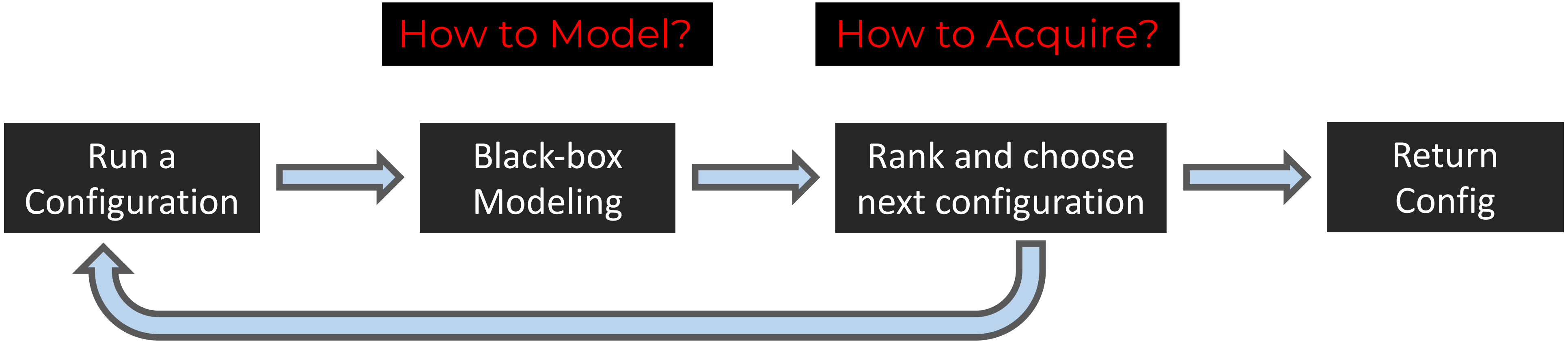
Workflow of Flash



Single Objective `CART` $\mu(x)$

Multi Objective `Multiple CARTs`

Workflow of Flash



Single Objective	`CART`	$\mu(x)$
Multi Objective	`Multiple CARTs`	**?**

How to acquire new configurations? Multi-objective

How to acquire new configurations? Multi-objective

- Need for a fitness assignment scheme to quantify relative fitness value.

How to acquire new configurations? Multi-objective

- Need for a fitness assignment scheme to quantify relative fitness value.
- Decomposition based scheme: Divide a problem into sub-problems.[1]

[1] Zhang, Qingfu, and Hui Li. "MOEA/D: A multiobjective evolutionary algorithm based on decomposition." IEEE Transactions on evolutionary computation 11.6 (2007): 712-731.

How to acquire new configurations? Multi-objective

- Need for a fitness assignment scheme to quantify relative fitness value.
- Decomposition based scheme: Divide a problem into sub-problems.[1]

Weight Vector

$$\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$$

$$\lambda_i \geq 0 \text{ for all } i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \lambda_i = 1$$

[1] Zhang, Qingfu, and Hui Li. "MOEA/D: A multiobjective evolutionary algorithm based on decomposition." IEEE Transactions on evolutionary computation 11.6 (2007): 712-731.

How to acquire new configurations? Multi-objective

- Need for a fitness assignment scheme to quantify relative fitness value.
- Decomposition based scheme: Divide a problem into sub-problems.[1]

Weight Vector

$$\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$$

$$\lambda_i \geq 0 \text{ for all } i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \lambda_i = 1$$

$$\boxed{\text{maximize } \; g^{ws}(x|\boldsymbol{\lambda}) = \sum_{i=1}^{m} \lambda_i f_i(x)}$$

[1] Zhang, Qingfu, and Hui Li. "MOEA/D: A multiobjective evolutionary algorithm based on decomposition." IEEE Transactions on evolutionary computation 11.6 (2007): 712-731.

How to acquire new configurations? Multi-objective

- Need for a fitness assignment scheme to quantify relative fitness value.
- Decomposition based scheme: Divide a problem into sub-problems.[1]

Weight Vector

$$\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$$

$$\lambda_i \geq 0 \text{ for all } i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \lambda_i = 1$$

$$\boxed{\text{maximize } \; g^{ws}(x|\boldsymbol{\lambda}) = \sum_{i=1}^{m} \lambda_i f_i(x)}$$

- N weight vectors can be used to find multiple pareto optimal solutions.

[1] Zhang, Qingfu, and Hui Li. "MOEA/D: A multiobjective evolutionary algorithm based on decomposition." IEEE Transactions on evolutionary computation 11.6 (2007): 712-731.

How to acquire new configurations? Multi-objective

- Need for a fitness assignment scheme to quantify relative fitness value.
- Decomposition based scheme: Divide a problem into sub-problems.[1]

Weight Vector

$$\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$$

$$\lambda_i \geq 0 \text{ for all } i = 1, \ldots, m$$

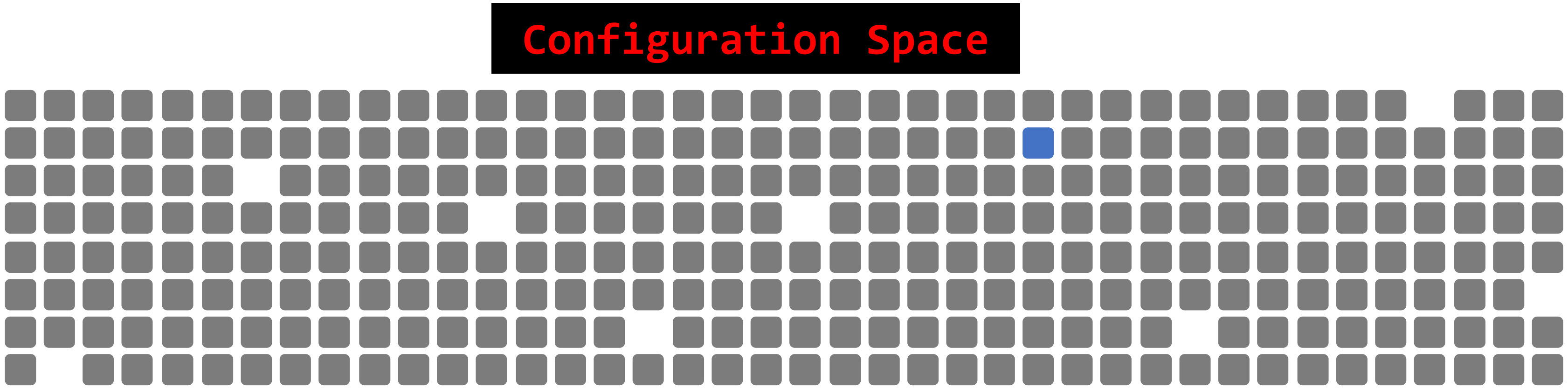
$$\sum_{i=1}^{m} \lambda_i = 1$$

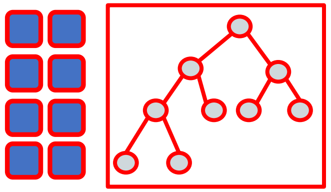
$$\boxed{\text{maximize } \; g^{ws}(x|\boldsymbol{\lambda}) = \sum_{i=1}^{m} \lambda_i f_i(x)}$$

- N weight vectors can be used to find multiple pareto optimal solutions.

[1] Zhang, Qingfu, and Hui Li. "MOEA/D: A multiobjective evolutionary algorithm based on decomposition." IEEE Transactions on evolutionary computation 11.6 (2007): 712-731.

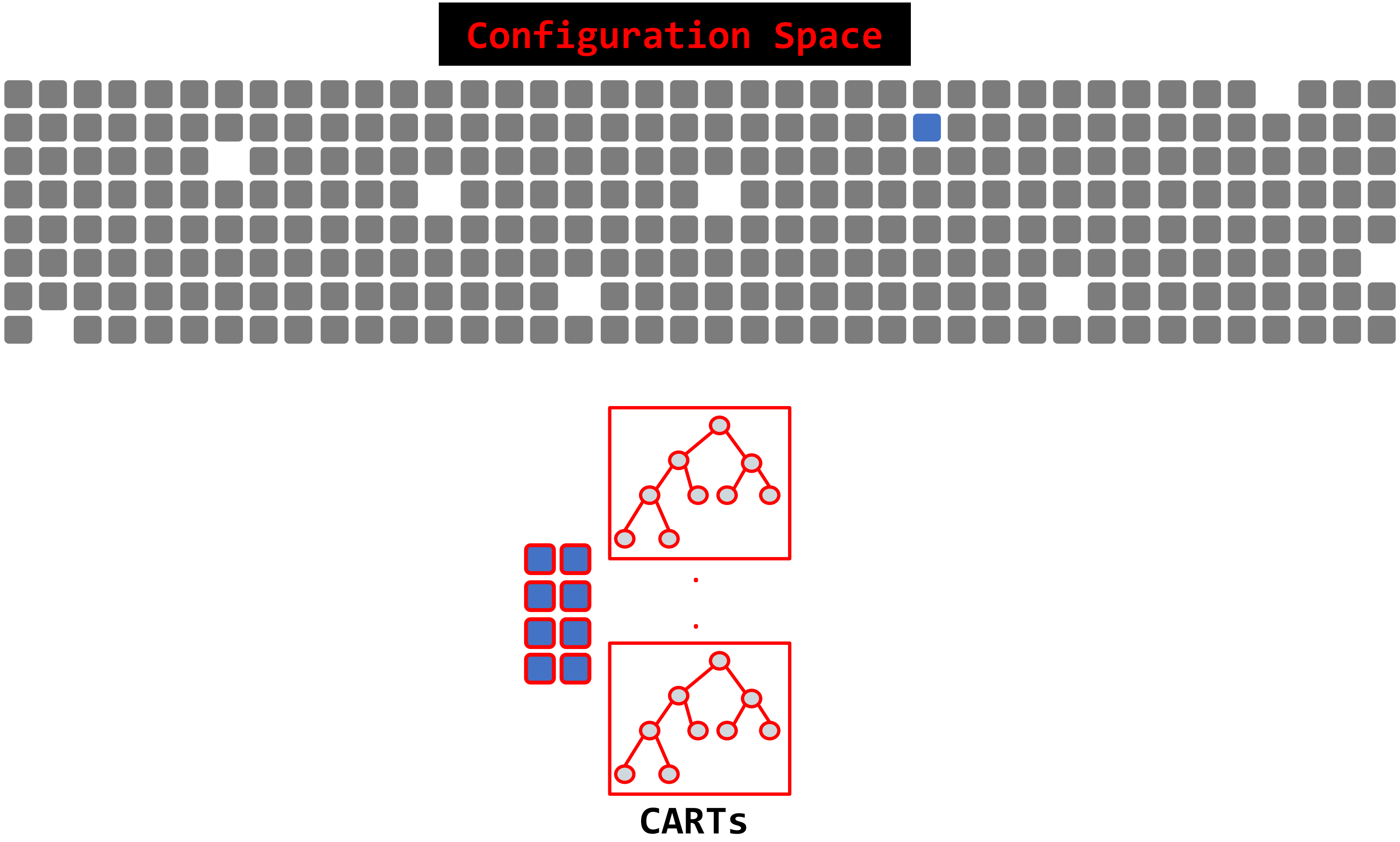
Flash-X (SMBO)



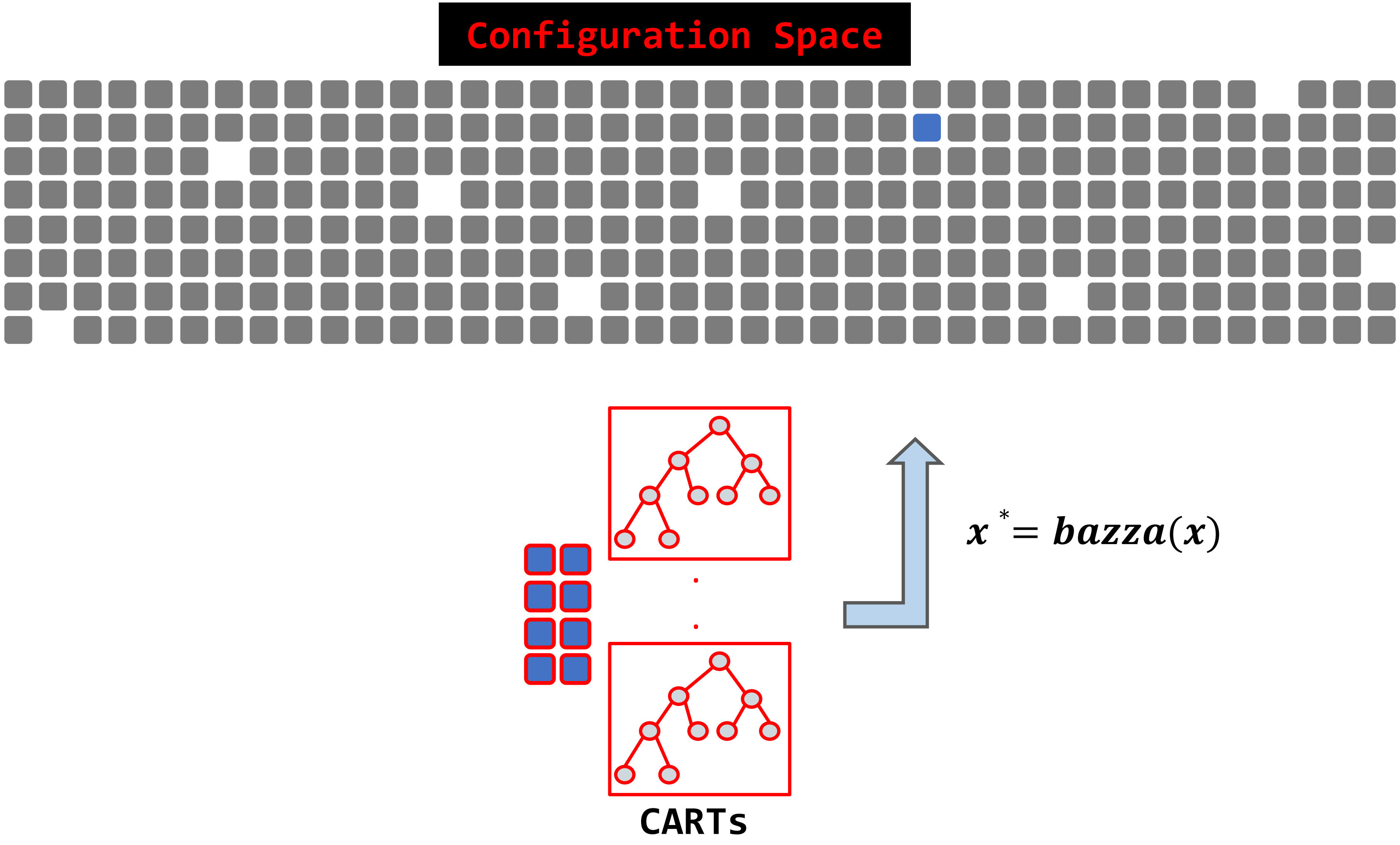


CARTs

Flash-X (SMBO)

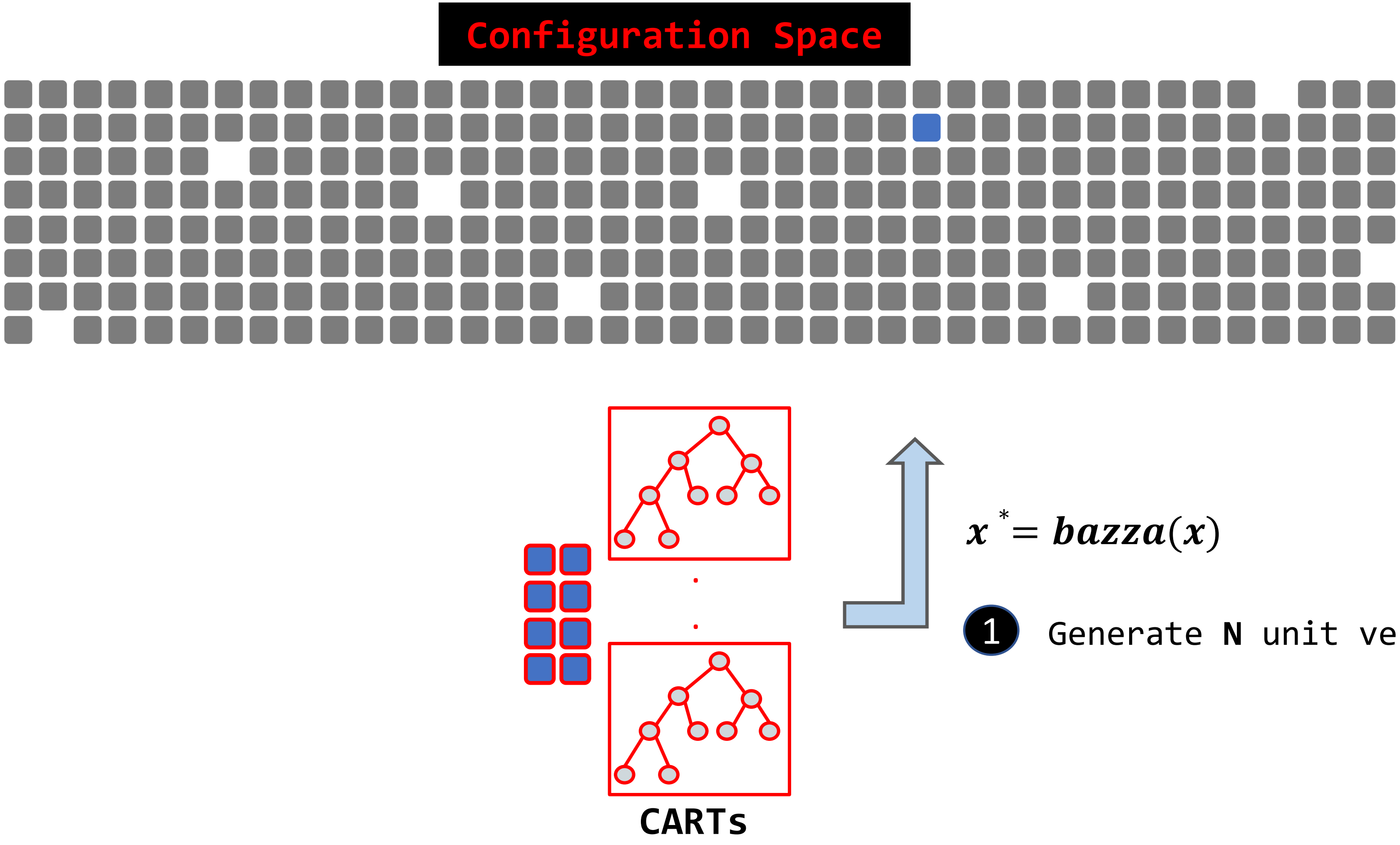


Flash-X (SMBO)



$$x^* = bazza(x)$$

Flash-X (SMBO)

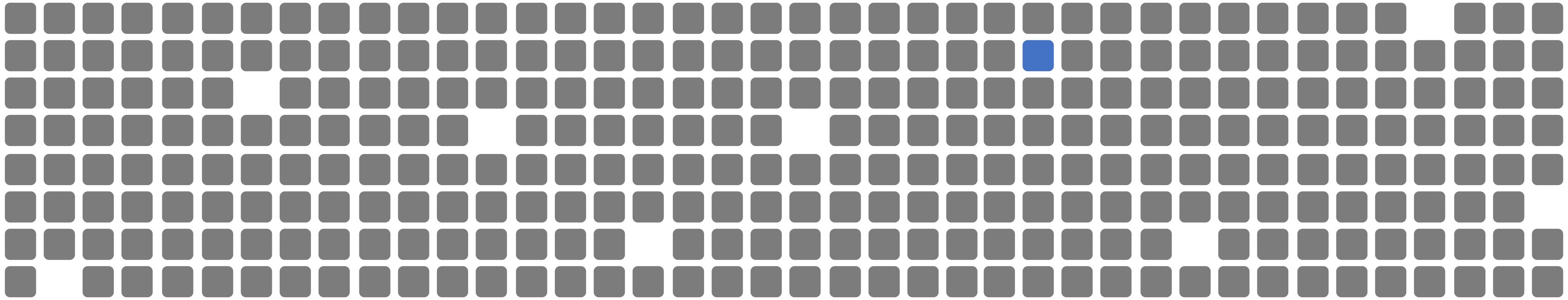


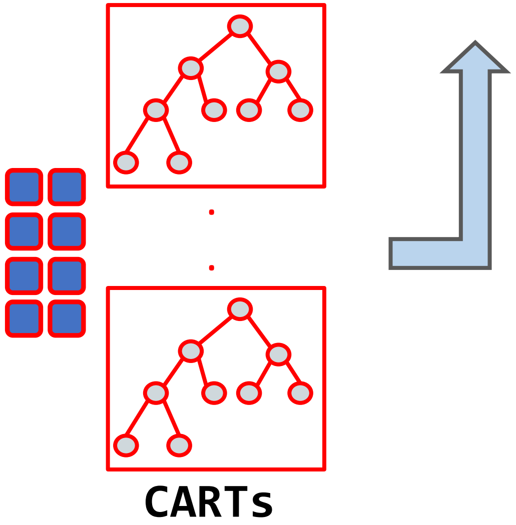
$$x^* = bazza(x)$$

1 Generate **N** unit vectors (V) of length M

Flash-X (SMBO)

Configuration Space



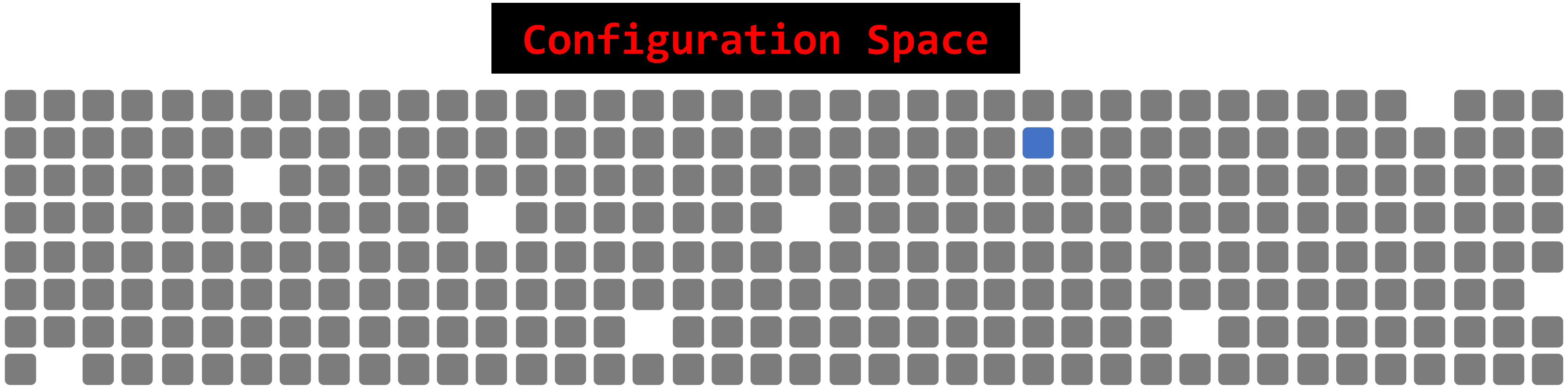


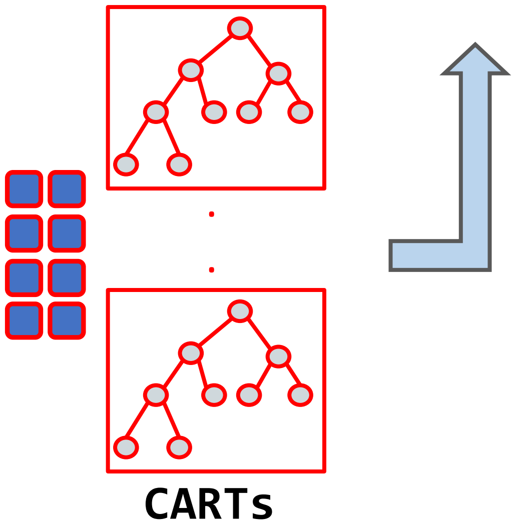
$$x^* = bazza(x)$$

1 Generate N unit vectors (V) of length M

User Defined

Flash-X (SMBO)



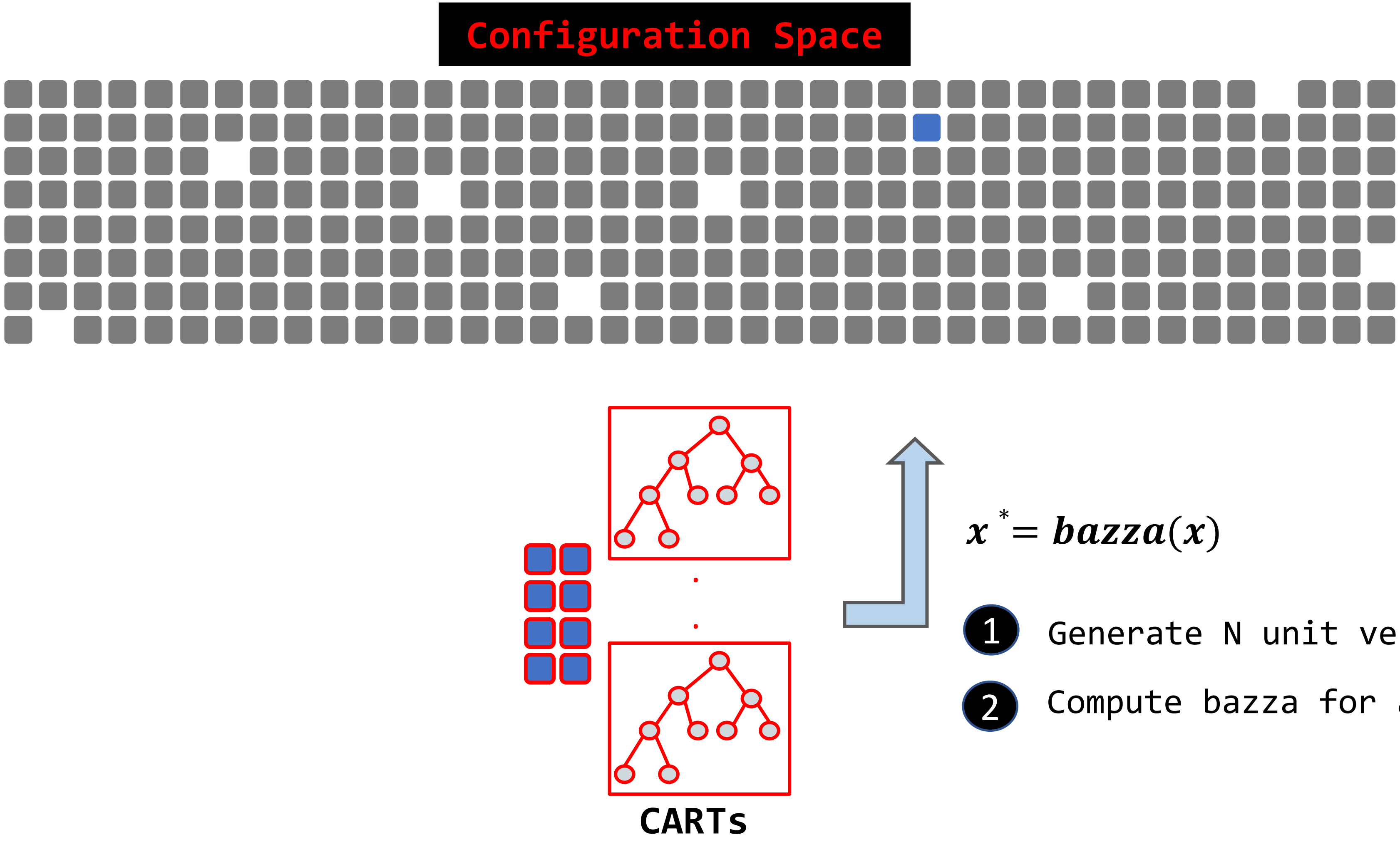


$$x^* = bazza(x)$$

1 Generate N unit vectors (V) of length M

Objectives

Flash-X (SMBO)

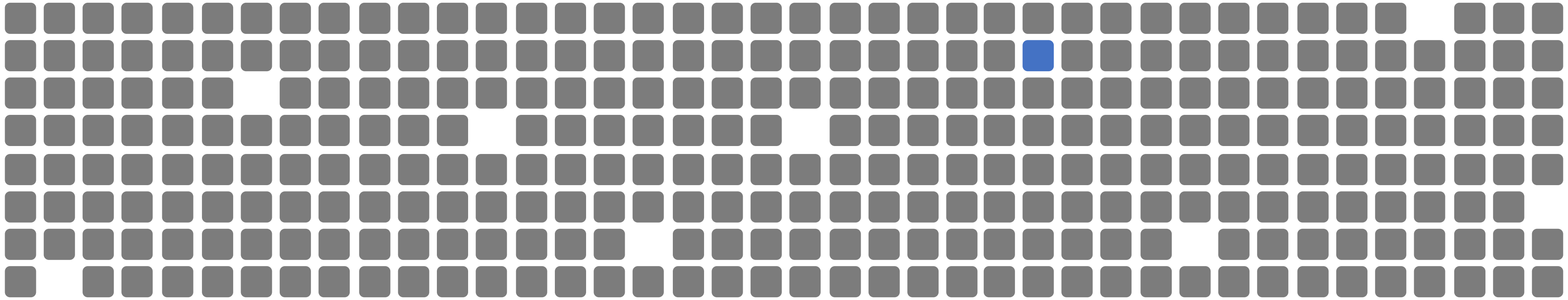


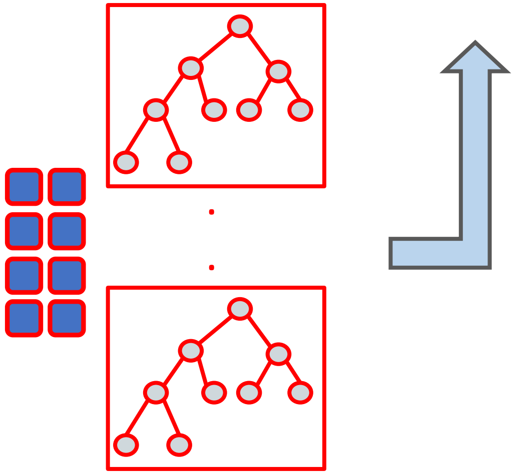
$$x^* = bazza(x)$$

1. Generate N unit vectors (V) of length M
2. Compute bazza for all configurations

Flash-X (SMBO)

Configuration Space





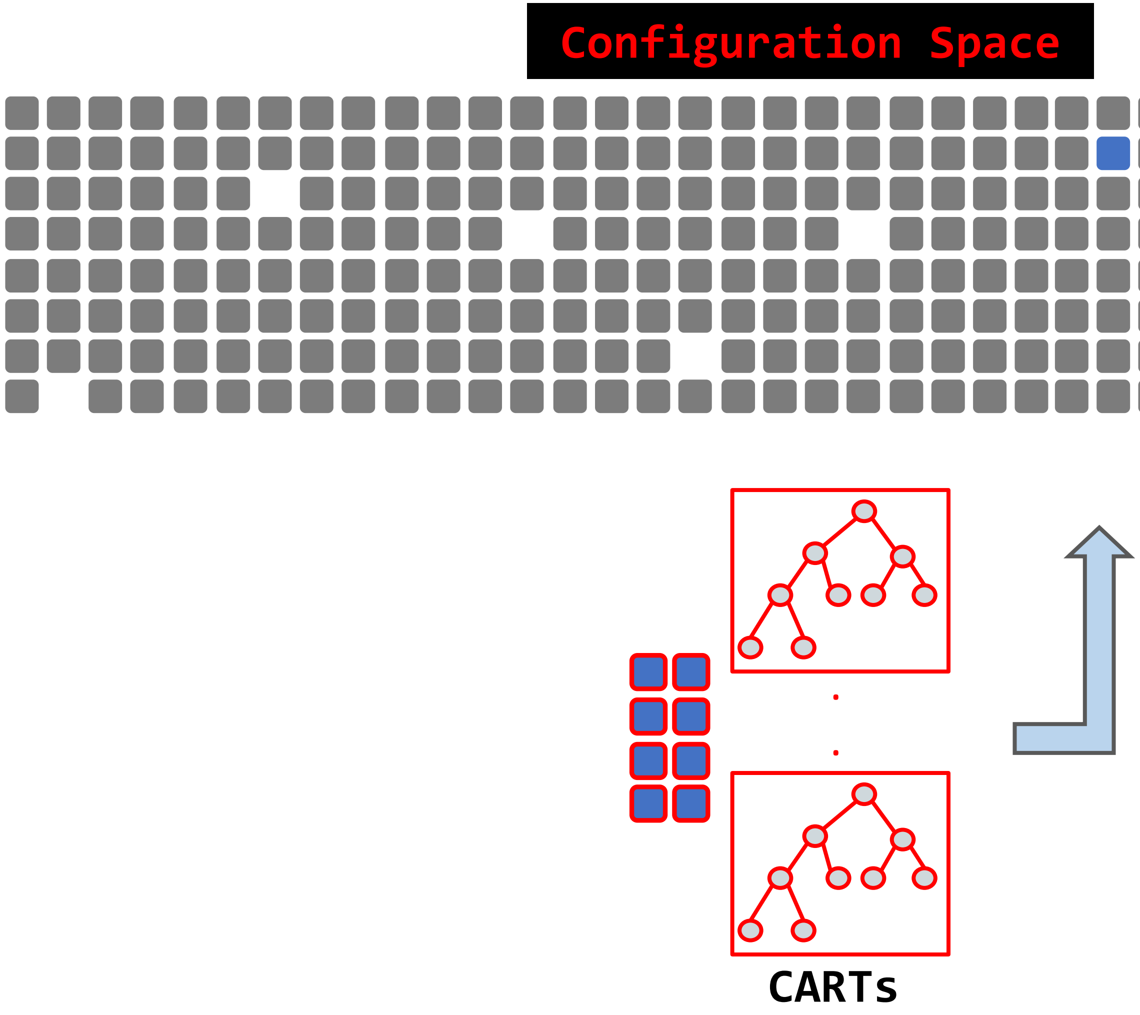
CARTs

$$x^* = bazza(x)$$

1. Generate N unit vectors (V) of length M
2. Compute bazza for all configurations

$$\boxed{bazza_i = \frac{1}{N}\sum_{n}^{N}\sum_{j}^{m}\left(V_{n,j} \cdot f_j(x_i)\right)}$$

Flash-X (SMBO)



$$x^* = bazza(x)$$

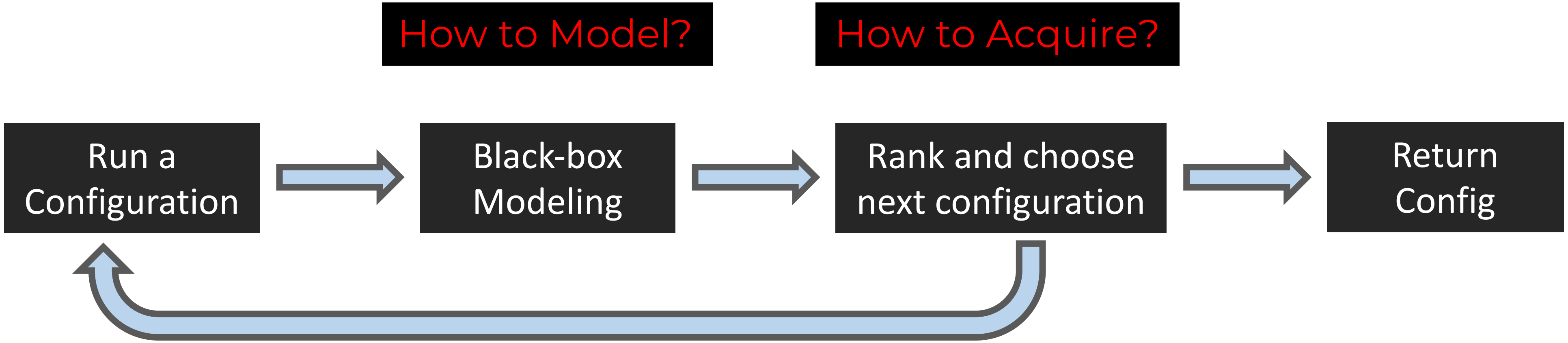
1. Generate N unit vectors (V) of length M
2. Compute bazza for all configurations

$$\boxed{bazza_i = \frac{1}{N}\sum_{n}^{N}\sum_{j}^{m}(V_{n,j} \cdot f_j(x_i))}$$

3. Return argmax($bazza_i$)

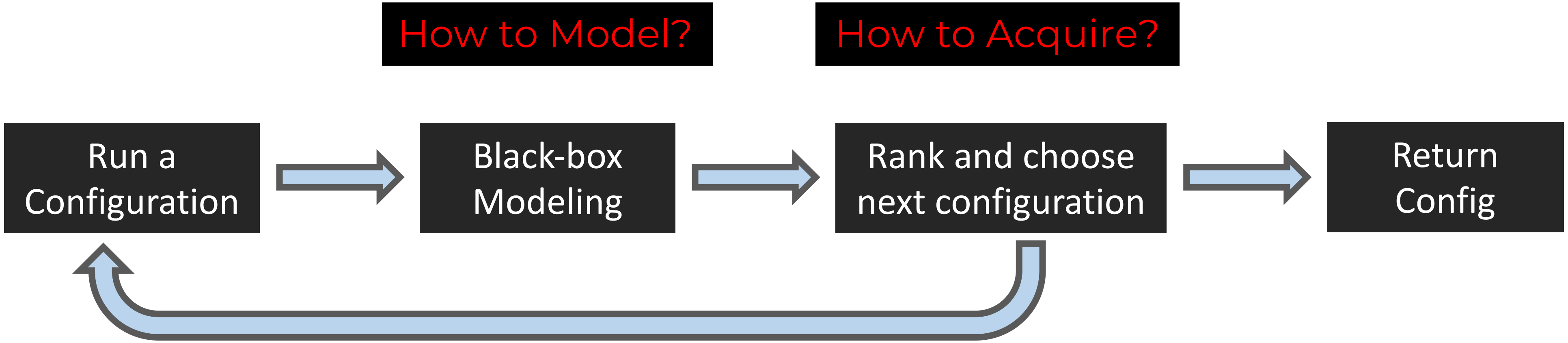
Flash-X (SMBO)

Workflow of Flash



Single Objective	`CART`	$\mu(x)$
Multi Objective	`Multiple CARTs`	**?**

Workflow of Flash



Single Objective	`CART`	$\mu(x)$
Multi Objective	`Multiple CARTs`	**Bazza**

Baseline

ePAL[1]

Reflects on the evaluated configurations to decide the next best configuration to measure using Maximum Variance (predictive uncertainty) as an acquisition function.

[1] Zuluaga et al.; "ε-pal: an active learning approach to the multi-objective optimization problem.”; The Journal of Machine Learning Research 17.1 (2016)

Baseline

ePAL[1]

Reflects on the evaluated configurations to decide the next best configuration to measure using Maximum Variance (predictive uncertainty) as an acquisition function.

We use two versions of ePAL:

- ePAL with ∈ = 0.01 (ePAL_0.01)
- ePAL with ∈ = 0.3 (ePAL_0.3)

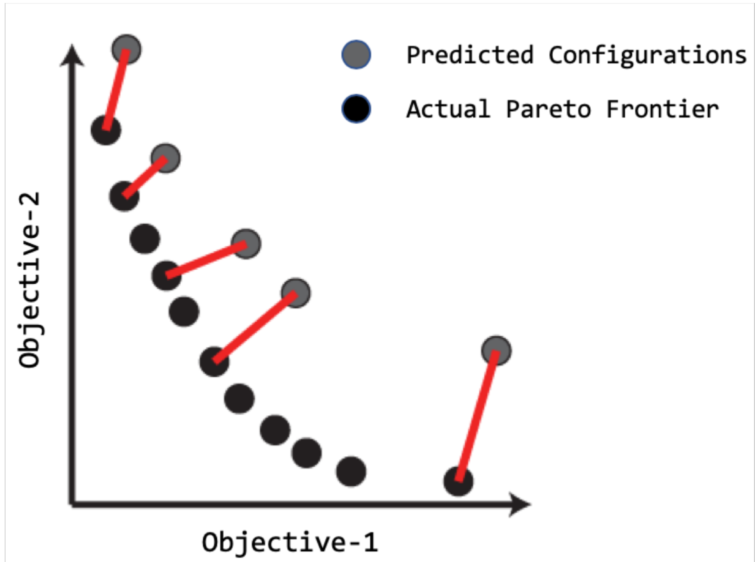
[1] Zuluaga et al.; "ε-pal: an active learning approach to the multi-objective optimization problem.”; The Journal of Machine Learning Research 17.1 (2016)

Evaluation Metrics

Evaluation Metrics

Generational Distance (GD)

Measures the closeness of the solutions from by the optimizers to the Pareto frontier that is, the actual set of non-dominated solutions.

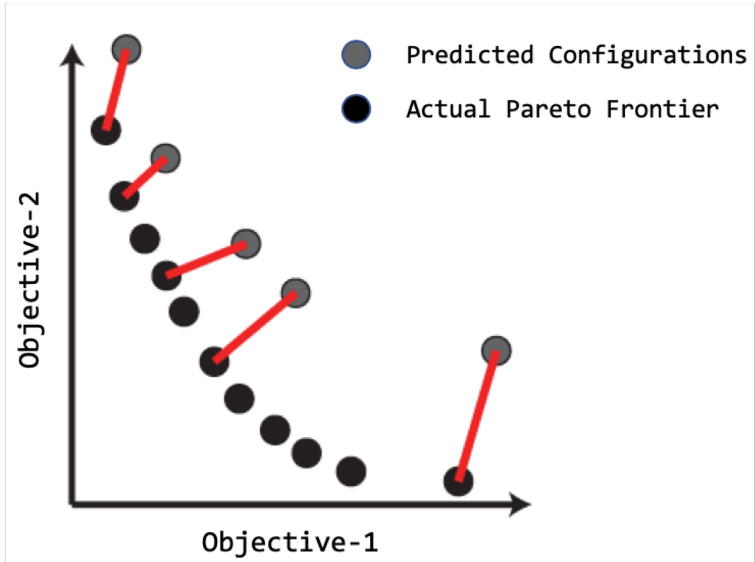


Flash-X (SMBO)

Evaluation Metrics

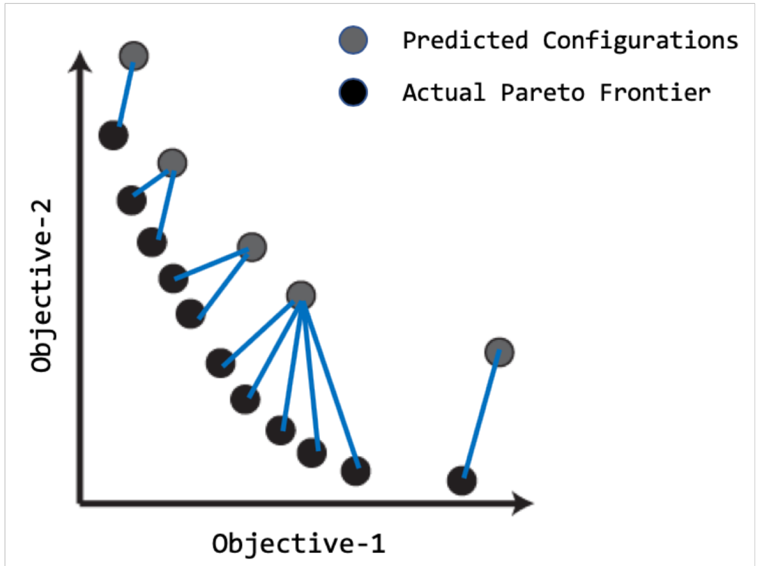
Generational Distance (GD)

Measures the closeness of the solutions from by the optimizers to the Pareto frontier that is, the actual set of non-dominated solutions.



Inverted Generational Distance (IGD)

Mean distance from points on the true Pareto-optimal solutions to its nearest point in solutions returned by the optimizer.



Research Questions

RQ3 How effective is FLASH-X for MO performance optimization?

RQ4 Can FLASH-X reduce the effort of MO performance optimization?

RQ5 Does FLASH-X save time for MO performance optimization?

Research Questions

Quality	RQ3	How effective is FLASH-X for MO performance optimization?
Cost	RQ4	Can FLASH-X reduce the effort of MO performance optimization?
	RQ5	Does FLASH-X save time for MO performance optimization?

Flash-X (SMBO)

RQ3: How effective is FLASH-X for MO performance optimization?

RQ3: How effective is FLASH-X for MO performance optimization?

Software	GD			IGD		
	epal_0.01	epal_0.3	FLASH-X	epal_0.01	epal_0.3	FLASH-X
SS-A	**0.002**	**0.002**	**0**	**0.002**	0.002	**0**
SS-B	**0**	**0**	**0.005**	**0**	0.003	0.001
SS-C	**0.001**	**0.001**	**0.003**	**0.004**	**0.004**	**0**
SS-D	**0**	**0.004**	**0.014**	**0.002**	0.007	0.009
SS-E	**0.001**	**0.001**	**0.012**	**0.004**	0.008	0.002
SS-F	**0**	0.016	0.008	**0**	0.006	0.016
SS-G	**0**	**0**	**0.023**	**0.003**	0.006	0.004
SS-H	**0**	**0**	**0**	**0**	**0**	**0**
SS-I	0.008	0.018	**0**	0.008	0.018	**0**
SS-J	**0**	**0**	**0.002**	**0.002**	**0.002**	**0**
SS-K	**0.001**	**0.001**	**0.003**	**0.001**	**0.002**	**0.001**
SS-L	**0.01**	**0.028**	**0.006**	**0.007**	**0.008**	**0.009**
SS-M	X	X	**0**	X	X	**0**
SS-N	X	X	**0.065**	X	X	**0.015**
SS-O	X	X	**3.01E-07**	X	X	**3.20E-06**
Win (%)	73	67	**93**	**67**	33	**67**

RQ3: How effective is FLASH-X for MO performance optimization?

Software	GD			IGD		
	epal_0.01	epal_0.3	FLASH-X	epal_0.01	epal_0.3	FLASH-X
SS-A	0.002	0.002	0	0.002	0.002	0
SS-B	0	0	0.005	0	0.003	0.001
SS-C	0.001	0.001	0.003	0.004	0.004	0
SS-D	0	0.004	0.014	0.002	0.007	0.009
SS-E	0.001	0.001	0.012	0.004	0.008	0.002
SS-F	0	0.016	0.008	0	0.006	0.016
SS-G	0	0	0.023	0.003	0.006	0.004
SS-H	0	0	0	0	0	0
SS-I	0.008	0.018	0	0.008	0.018	0
SS-J	0	0	0.002	0.002	0.002	0
SS-K	0.001	0.001	0.003	0.001	0.002	0.001
SS-L	0.01	0.028	0.006	0.007	0.008	0.009
SS-M	X	X	0	X	X	0
SS-N	X	X	0.065	X	X	0.015
SS-O	X	X	3.01E-07	X	X	3.20E-06
Win (%)	73	67	93	67	33	67

RQ3: How effective is FLASH for MO performance optimization?

RQ3: How effective is FLASH-X for MO performance optimization?

FLASH-X is very effective for MO performance configuration optimization.

Software	GD			IGD		
	epal_0.01	epal_0.3	FLASH	epal_0.01	epal_0.3	FLASH
SS-A	0.002	0.002	0	0.002	0.002	0
SS-B	0	0	0.005	0	0.003	0.001
SS-C	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0
SS-D	0	0.004	0.014	0.002	0.007	0.009
SS-E	0.001	0.001	0.012	0.004	0.008	0.002
SS-F	0	0.016	0.008	0	0.006	0.016
SS-G	0	0	0.023	0.003	0.006	0.004
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
SS-I	0.008	0.018	0	0.008	0.018	0
SS-J	0	0	0.002	0.002	0.002	0
SS-K	0.001	0.001	0.003	0.001	0.002	0.001
SS-L	0.01	0.028	0.006	0.007	0.008	0.009
SS-M	X	X	0	X	X	0
SS-N	X	X	0.065	X	X	0.015
SS-O	X	X	3.01E-07	X	X	3.20E-06
Win (%)	73	67	93	67	33	67

Flash-X (SMBO)

RQ4: Can FLASH-X reduce the effort of MO performance optimization?

Flash-X (SMBO)

RQ4: Can FLASH-X reduce the effort of MO performance optimization?

Software	Evals		
	epal_0.01	**epal_0.3**	FLASH-X
SS-A	109.5	73.5	**50**
SS-B	84.5	**20**	**50**
SS-C	247	101	**50**
SS-D	119.5	67	**50**
SS-E	208	54.5	**50**
SS-F	138	71	**50**
SS-G	131	69	**50**
SS-H	52	**28**	**50**
SS-I	48	**30**	**50**
SS-J	186	**30**	**50**
SS-K	209	140	**50**
SS-L	68.5	**35**	**50**
SS-M	X	X	**50**
SS-N	X	X	**50**
SS-O	X	X	**50**
Win (%)	0	33	**80**

Flash-X (SMBO)

RQ4: Can FLASH-X reduce the effort of MO performance optimization?

Software	Evals		
	epal_0.01	**epal_0.3**	FLASH-X
SS-A	109.5	73.5	**50**
SS-B	84.5	**20**	**50**
SS-C	247	101	**50**
SS-D	119.5	67	**50**
SS-E	208	54.5	**50**
SS-F	138	71	**50**
SS-G	131	69	**50**
SS-H	52	**28**	**50**
SS-I	48	**30**	**50**
SS-J	186	**30**	**50**
SS-K	209	140	**50**
SS-L	68.5	**35**	**50**
SS-M	X	X	**50**
SS-N	X	X	**50**
SS-O	X	X	**50**
Win (%)	0	33	**80**

RQ4: Can FLASH reduce the effort of MO performance optimization?

Software	Evals		
	epal_0.01	epal_0.3	FLASH
SS-A	109.5	73.5	50
SS-B	[illegible]	20	50
SS-C	[illegible]	[illegible]	[illegible]
SS-D	119.5	67	50
SS-E	208	54.5	50
SS-F	138	71	50
SS-G	131	69	50
SS-H	[illegible]	[illegible]	[illegible]
SS-I	48	30	50
SS-J	186	30	50
SS-K	209	140	50
SS-L	68.5	35	50
SS-M	X	X	50
SS-N	X	X	50
SS-O	X	X	50
Win (%)	0	33	80

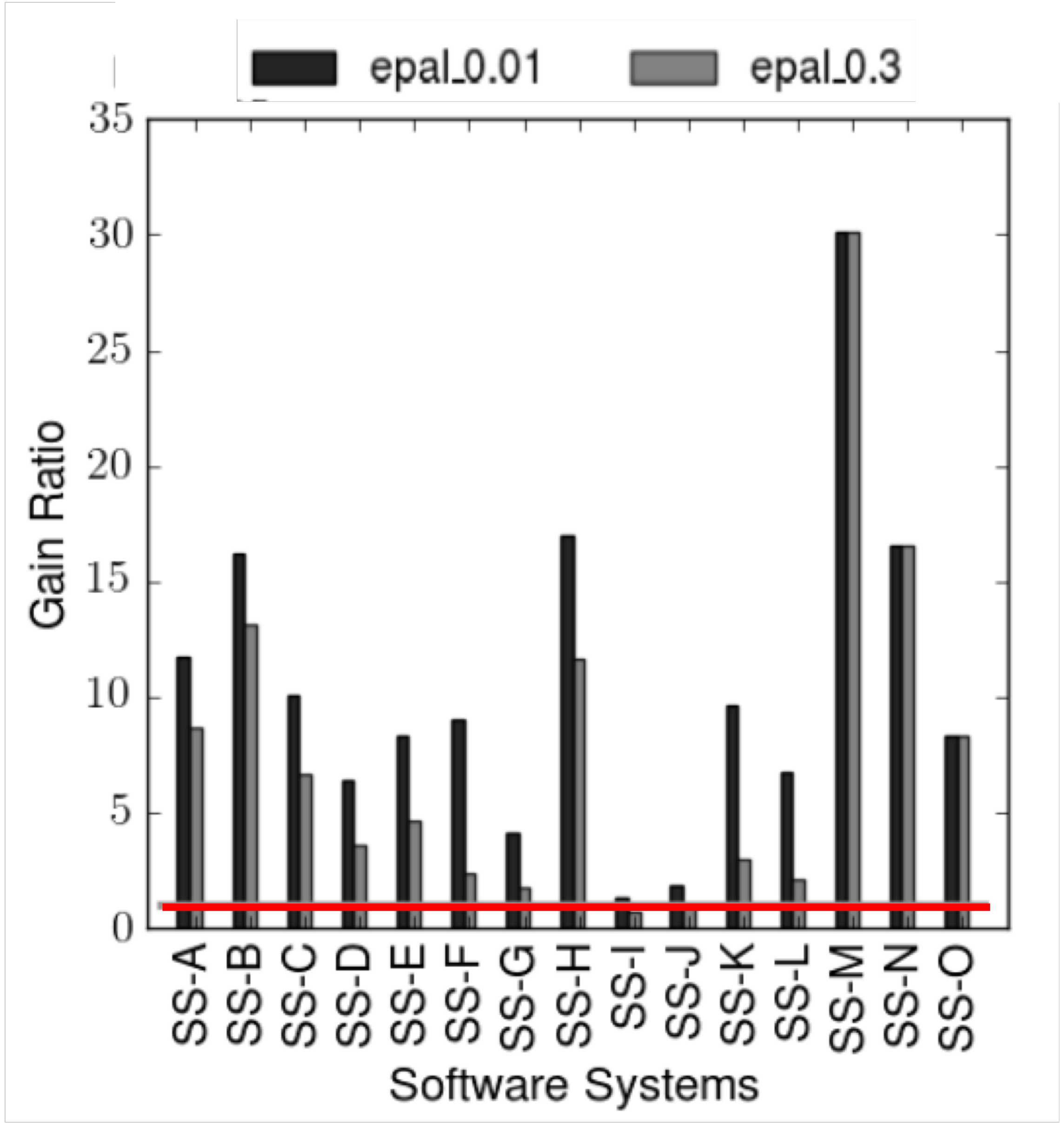
RQ4:Can FLASH-X reduce the effort of MO performance optimization?

FLASH-X requires fewer measurements than ePAL.

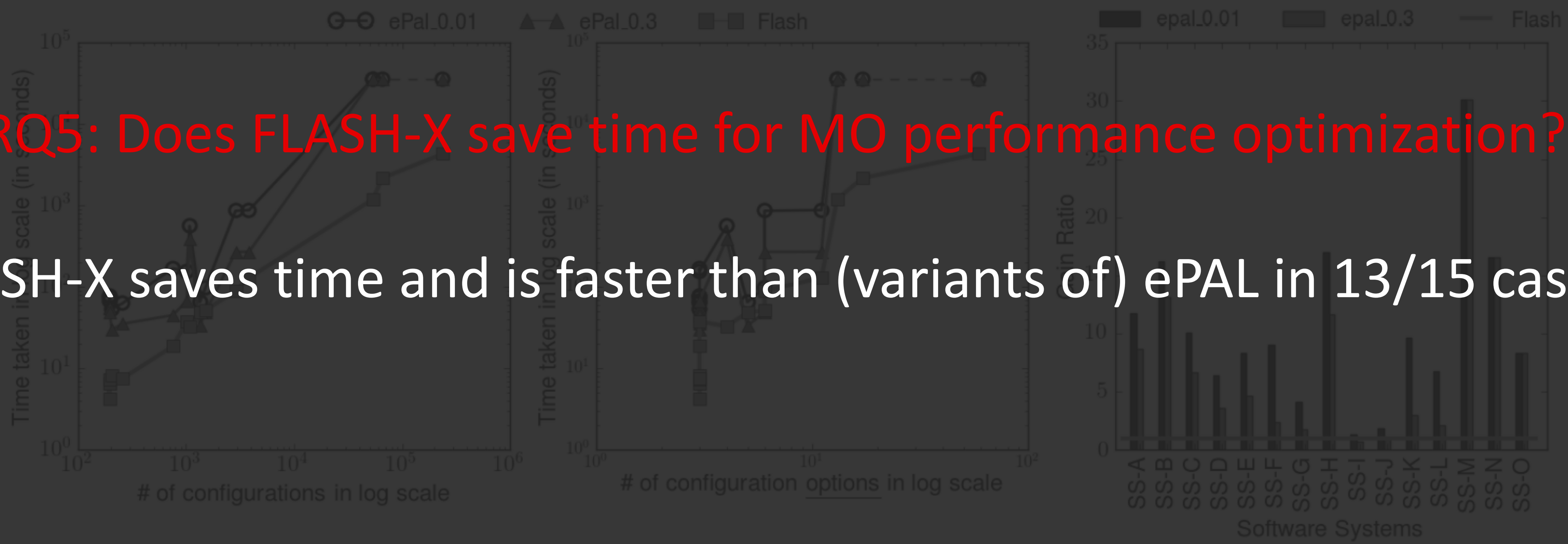
RQ5: Does FLASH-X save time for MO performance optimization?

Flash-X (SMBO)

RQ5: Does FLASH save time for MO performance optimization?



RQ5: Does FLASH save time for MO performance optimization?



RQ5: Does FLASH-X save time for MO performance optimization?

FLASH-X saves time and is faster than (variants of) ePAL in 13/15 cases.

Use the model to sample

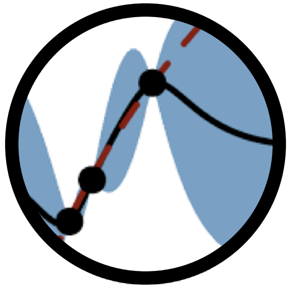
Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

Ranking

SMBO



Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

- First **Cluster and then Sample** to avoid redundant samples
- Did not perform well in External Validation Studies

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

- First **Cluster and then Sample** to avoid redundant samples
- Did not perform well in External Validation Studies

Unsupervised clustering does not work in all cases

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

- First **Cluster and then Sample** to avoid redundant samples
- Did not perform well in External Validation Studies

Unsupervised clustering does not work in all cases

Ranking

- The **Ranking** is a useful paradigm
  - Ranking is extremely robust to errors or outliers
  - reduces the number of training samples to train models
- Requires use of holdout set

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

- First **Cluster and then Sample** to avoid redundant samples
- Did not perform well in External Validation Studies

Unsupervised clustering does not work in all cases

Ranking

- The **Ranking** is a useful paradigm
  - Ranking is extremely robust to errors or outliers
  - reduces the number of training samples to train models
- Requires use of holdout set

Evaluating holdout set can be expensive, hence not suitable in practice

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

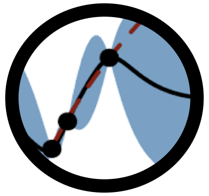
- First **Cluster and then Sample** to avoid redundant samples
- Did not perform well in External Validation Studies

Unsupervised clustering does not work in all cases

Ranking

- The **Ranking** is a useful paradigm
  - Ranking is extremely robust to errors or outliers
  - reduces the number of training samples to train models
- Requires use of holdout set

Evaluating holdout set can be expensive, hence not suitable in practice



SMBO

- "**Given what one knows about the problem, what can be done next?**" is a powerful idea

Effective performance optimization of configurable software systems only requires **approximate**, **cheap** and **easy to build** models.

Clustering

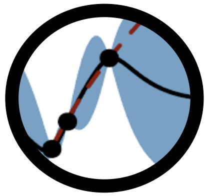
- First **Cluster and then Sample** to avoid redundant samples
- Did not perform well in External Validation Studies

Unsupervised clustering does not work in all cases

Ranking

- The **Ranking** is a useful paradigm
  - Ranking is extremely robust to errors or outliers
  - reduces the number of training samples to train models
- Requires use of holdout set

Evaluating holdout set can be expensive, hence not suitable in practice



SMBO

- "**Given what one knows about the problem, what can be done next?**" is a powerful idea

Use model to sample

Future Work

Future Work

Can expert knowledge be used to increase the rate of convergence?

Future Work

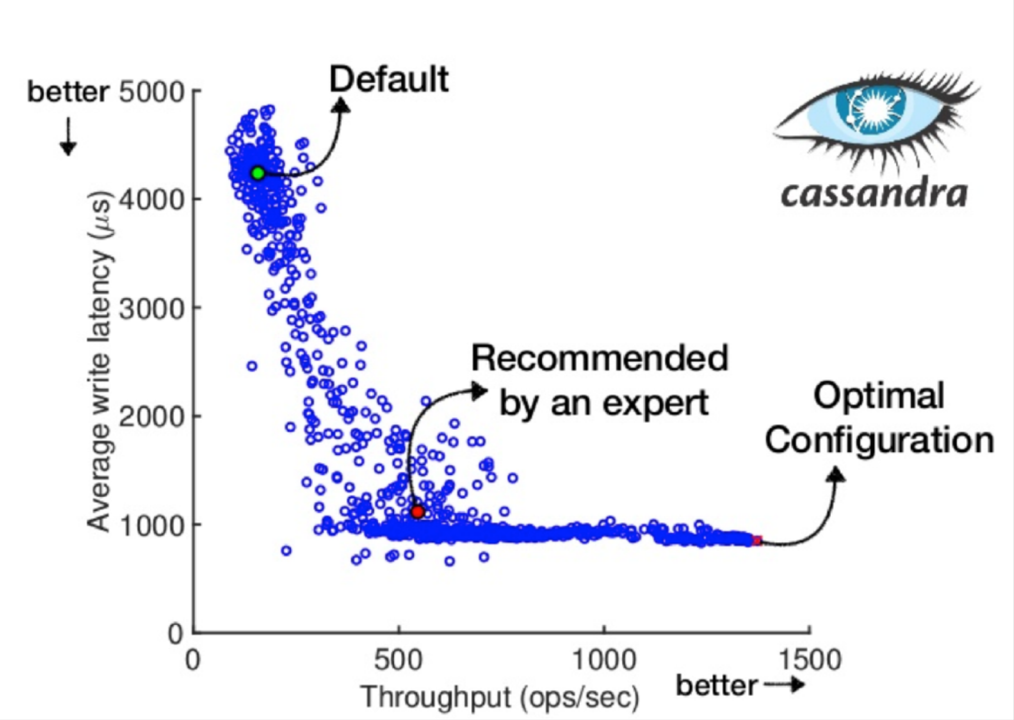
Human in the loop

Can expert knowledge be used to increase the rate of convergence?

Future Work

Human in the loop

Can expert knowledge be used to increase the rate of convergence?



Future Work

Human in the loop

Can expert knowledge be used to increase the rate of convergence?

Can we learn from our experience to increase the rate of convergence or decrease the cost?

Future Work

Human in the loop

Can expert knowledge be used to increase the rate of convergence?

Can we learn from our experience to increase the rate of convergence or decrease the cost?

FLASH has to be repeated if ever the **software** is updated on the **workload** of the system changes abruptly or **environment** changes.

Future Work

Human in the loop

Can expert knowledge be used to increase the rate of convergence?

Transfer Learning

Can we learn from our experience to increase the rate of convergence or decrease the cost?

Future Work

Human in the loop

Can expert knowledge be used to increase the rate of convergence?

Transfer Learning

Can we learn from our experience to increase the rate of convergence or decrease the cost?

Can these ideas be applied to other domains?

Future Work

Human in the loop

Can expert knowledge be used to increase the rate of convergence?

Transfer Learning

Can we learn from our experience to increase the rate of convergence or decrease the cost?

External Validity

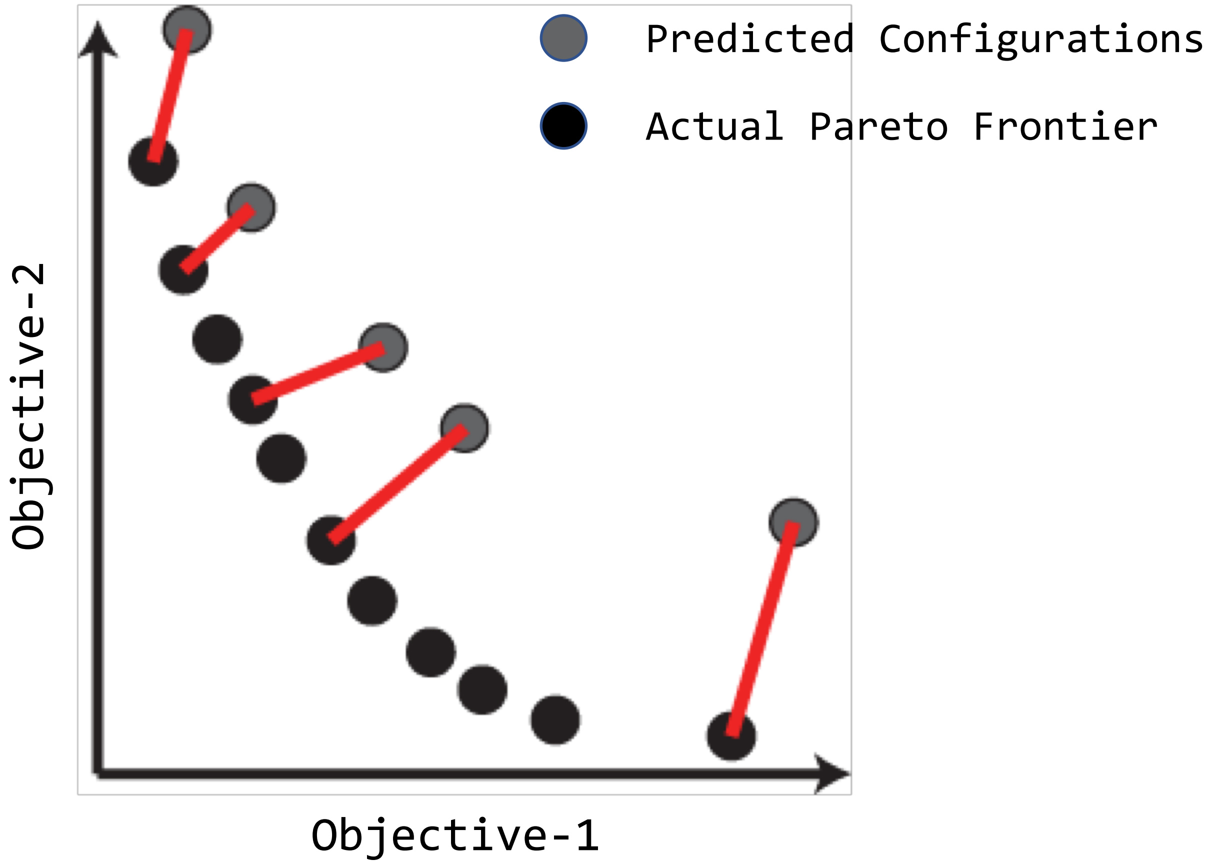
Can these ideas be applied to other domains?

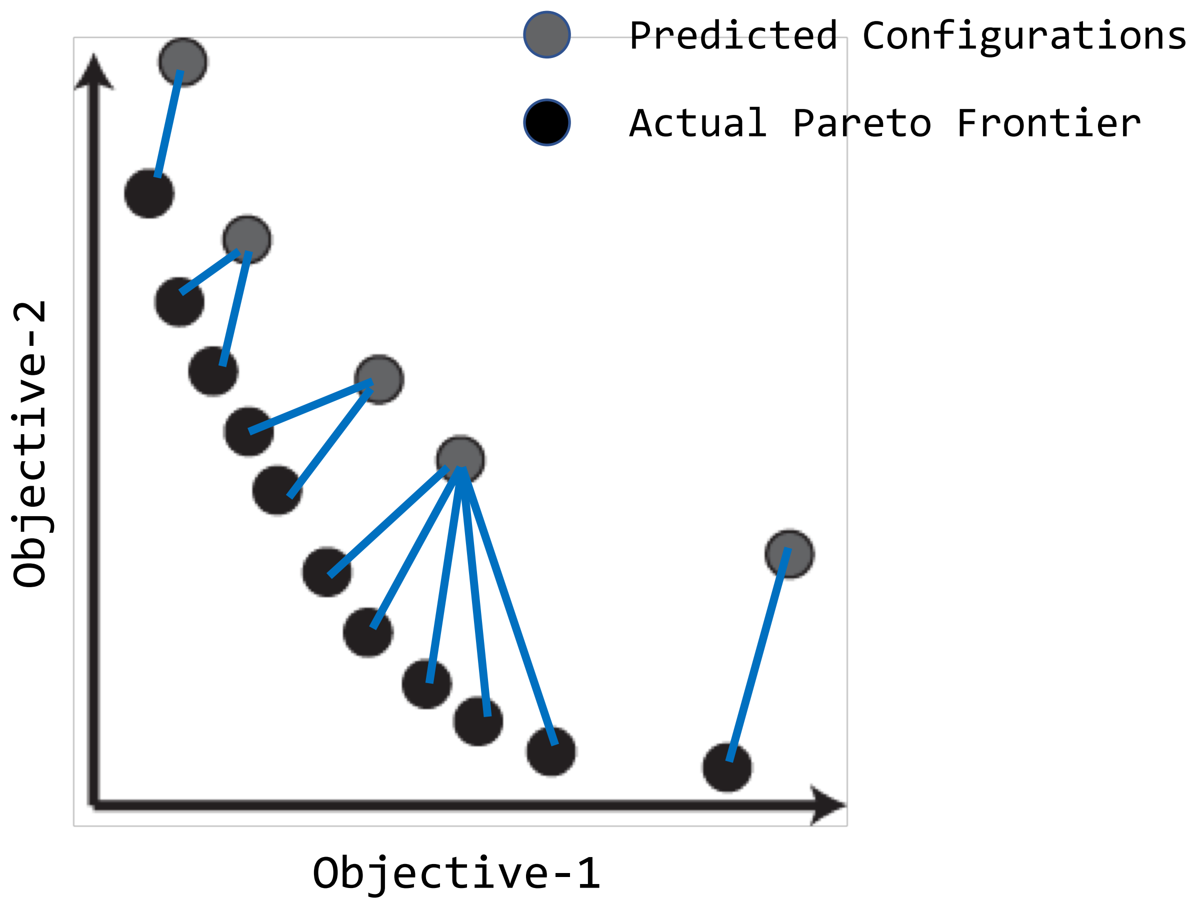
Effective performance optimization of configurable software systems only requires approximate, cheap and easy to build models.

Effective performance optimization of configurable software systems only requires approximate, cheap and easy to build models.

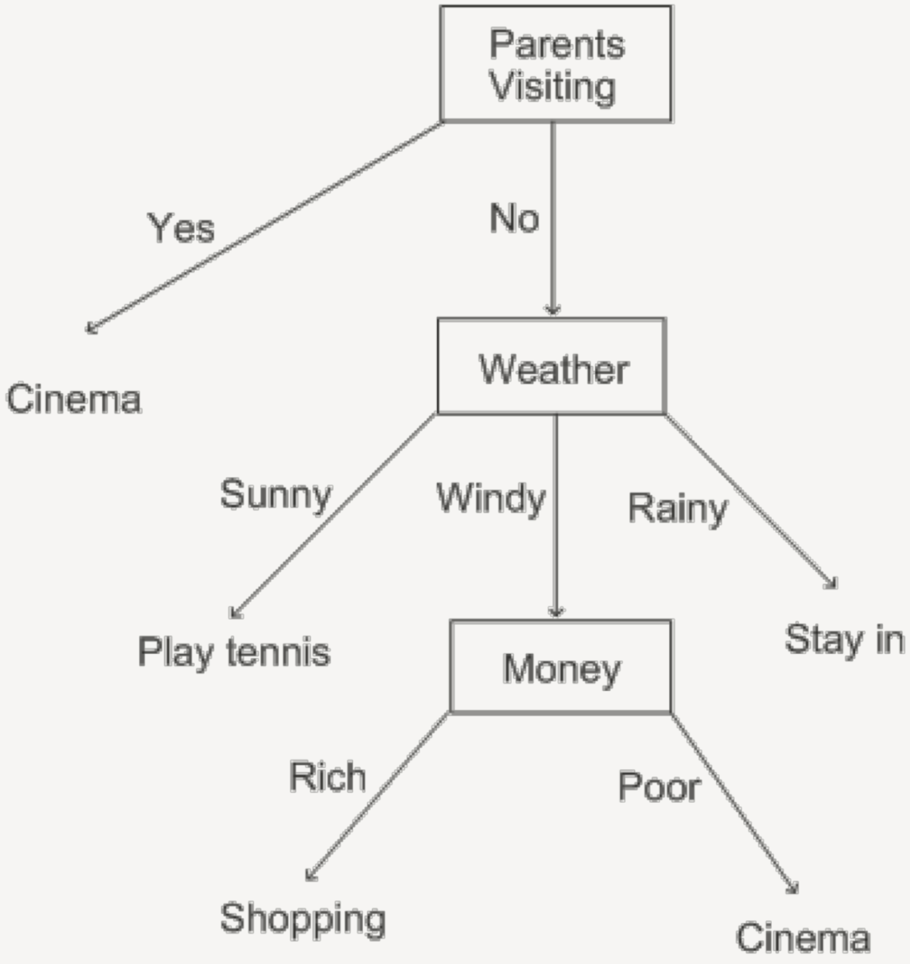
Thank you

“There’s no sense in being precise when you don’t even know what you’re talking about.”
—John von Neumann


Predicted Configurations
Actual Pareto Frontier
Objective-2
Objective-1


Predicted Configurations
Actual Pareto Frontier
Objective-2
Objective-1

Decision Tree



- It worked!
- Prior work* used CART
- Scalable
- More comprehensible

[Guo'13] Guo, Jienmei, et al. "Variability-aware performance prediction: A statistical learning approach". ASE-2013
[Sarkar'15] Sarkar, Atri, et al. "Cost-efficient sampling for performance prediction of configurable systems (t)." ASE-2015
[Nair'16] Nair, Vivek, et al. "Faster discovery of faster system configurations with spectral learning." ASE Journal-2017

Software	Regr. Model	Acq. Function
Spearmint	Gaussian Process	Exp. Improv
MOE	Gaussian Process	Exp. Improv
Hyperopt	Tree Parzen Est.	Exp. Improv
SMAC	Random Forest	Exp. Improv