Carnegie Mellon University

Lecture #08

Special Topics:

Self-Driving Database Management Systems

Knob/Parameter Tuning III

@Ying_Jiang // 15-799 // Spring 2022

LAST CLASS

CDBTune from Tencent & HUST

End-to-end RL (offline training + online tuning)

63 input metrics, 266 (65) knobs of MySQL v5.6

TODAY'S AGENDA

Overview

Solving Constrained Optimization

Boosting Tuning Process with Meta-Learning

Evaluation

Parting Thoughts

TODAY'S AGENDA

Overview

Solving Constrained Optimization

Boosting Tuning Process with Meta-Learning

Evaluation

Parting Thoughts

TERMINOLOGY

SLA
→ Service Level Agreement

TCO
→ Total Cost of Ownership

Feasible configuration
→ Configuration that satisfies the performance constraints

Meta-learning
→ learning from meta-data to accomplish new tasks
→ Meta-data: the data describing previous learning tasks and learned models, including measurable properties of the task, also known as meta-features

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

MOTIVATION

Resource utilization is a necessity for cloud providers
→ TCO
→ Availability
→ Request rates seldom reaches processing capacity

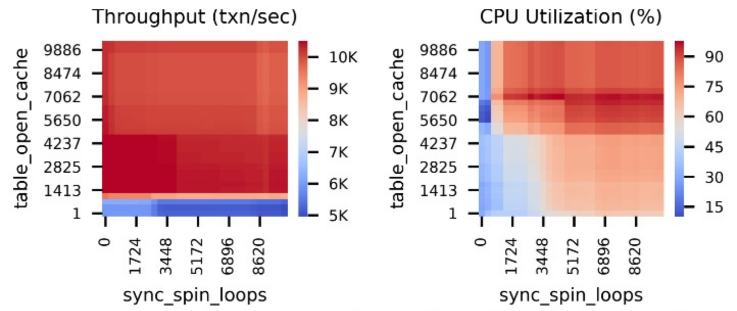


Figure 1: TPS and CPU Usage for Real Workload with 2 Knobs

Also important for users when choosing instances
→ Cost!

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

CHALLENGES

Reduce resource utilization while guaranteeing SLA

Satisfy constraints imposed by real applications
→ Tuning time is limited
→ Replaying, which is done repeatedly, dominates the tuning time
→ Real workloads even slower (> 5 mins/iteration)

Represent knowledge effectively from historical data
→ To accelerate the tuning process, ResTune utilizes data from tuning other tasks and transfer the experience

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

RELATED WORK

Mainly focus on optimizing performance

→ Search-based: cannot utilize past experience for new requests

→ BO based: adaptability (rely on a fixed hardware)

→ RL based: training overload is high

- QTune takes thousands of iterations to train SYSBENCH

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

ARCHITECTURE OVERVIEW

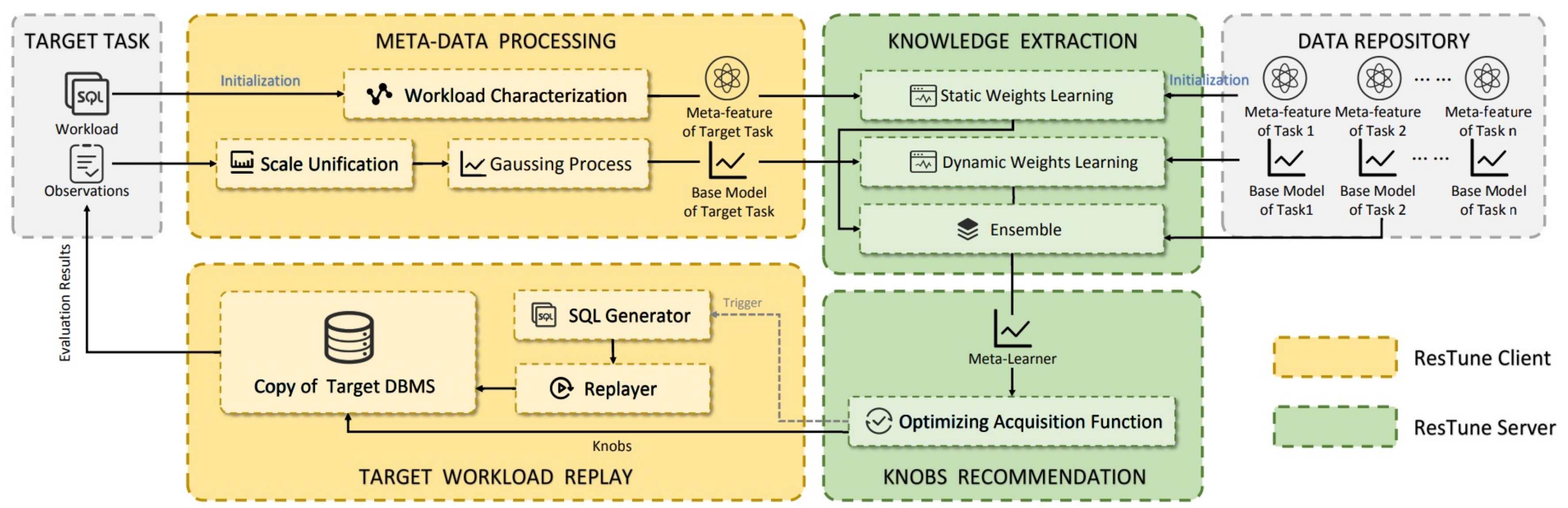


Figure 2: Overall Architecture of ResTune

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

APPROACH

1. Solving constrained optimization

→ To find configurations to minimize the resource usage without violating the throughput and latency restriction

→ Formulated as a constrained Bayesian Optimization problem

2. Boosting tuning process with meta-learning

→ To accelerate and improve knob tuning

→ Transfer experience from heterogeneous tasks with meta-learning

- Relative rankings, static /dynamic weights

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

TODAY'S AGENDA

Overview
Solving Constrained Optimization
Boosting Tuning Process with Meta-Learning
Evaluation
Parting Thoughts

PROBLEM STATEMENT

The resource-oriented tuning problem is formalized as an optimization problem with SLA constraints.

→ constraints are determined by the metrics under default configuration

$$\begin{aligned} &\arg\min_{\theta} f_{res}(\theta), \\ \text{s.t.} \quad & f_{tps}(\theta) \geq \lambda_{tps} \\ & f_{lat}(\theta) \leq \lambda_{lat} \end{aligned}$$

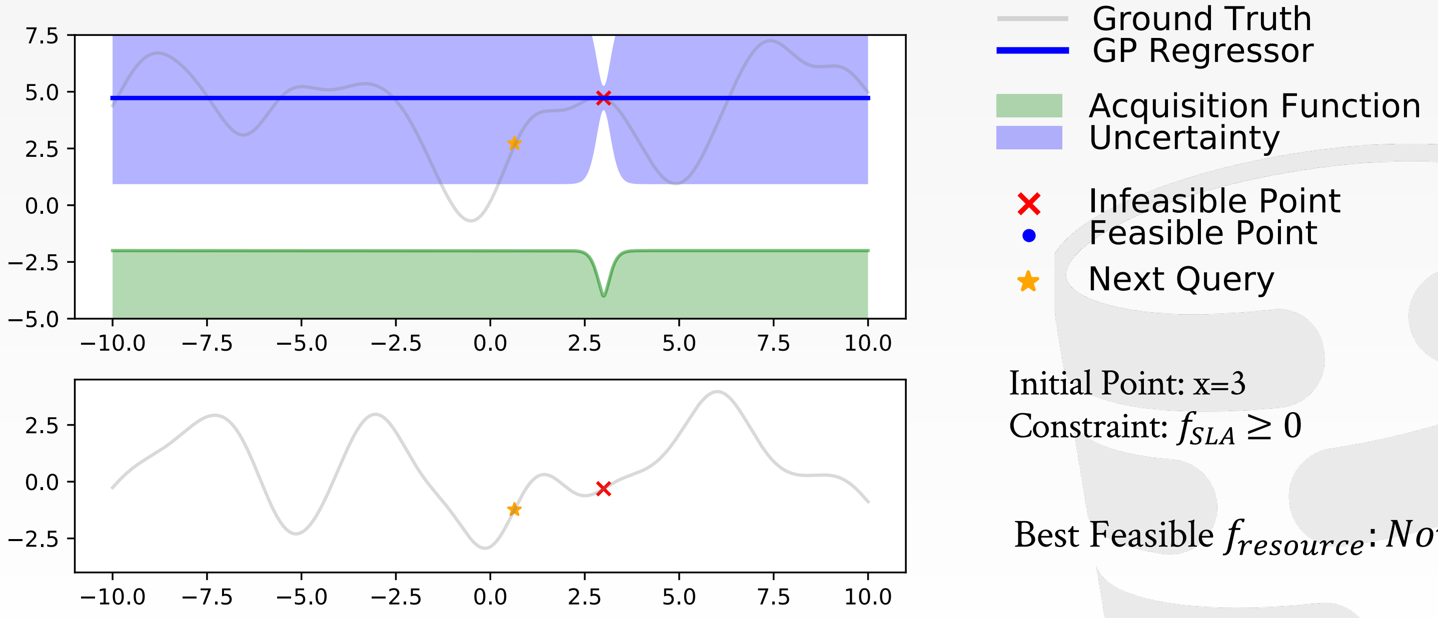
RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.

$f_{resource}$

f_{SLA}



Iteration 1

Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: *None*

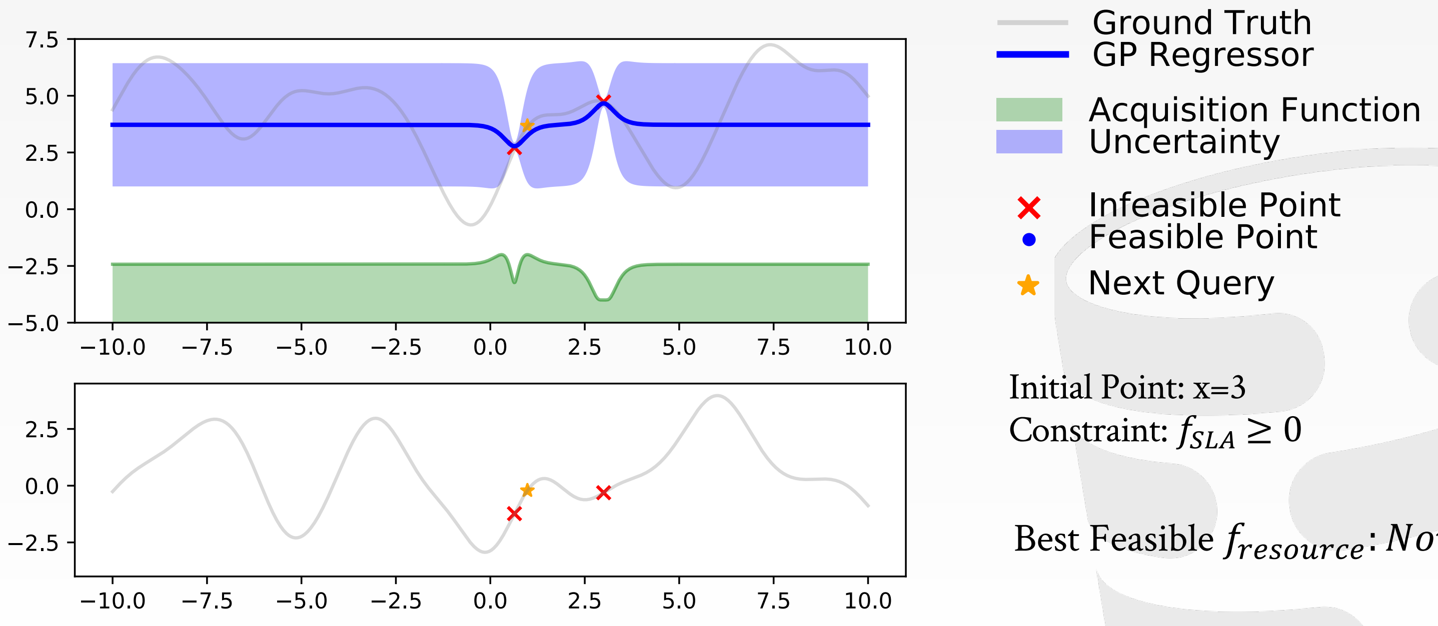
Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.

$f_{resource}$

f_{SLA}



Iteration 2

Initial Point: x=3

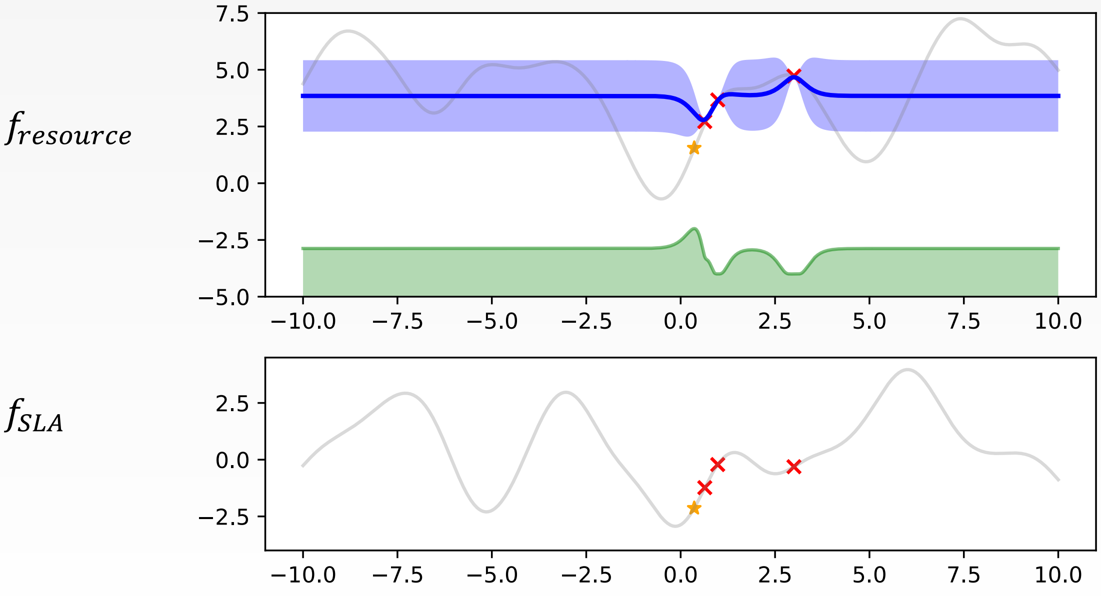
Constraint: $f_{SLA} \geq 0$

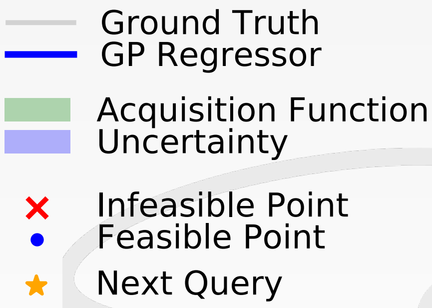
Best Feasible $f_{resource}$: *None*

Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.





Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

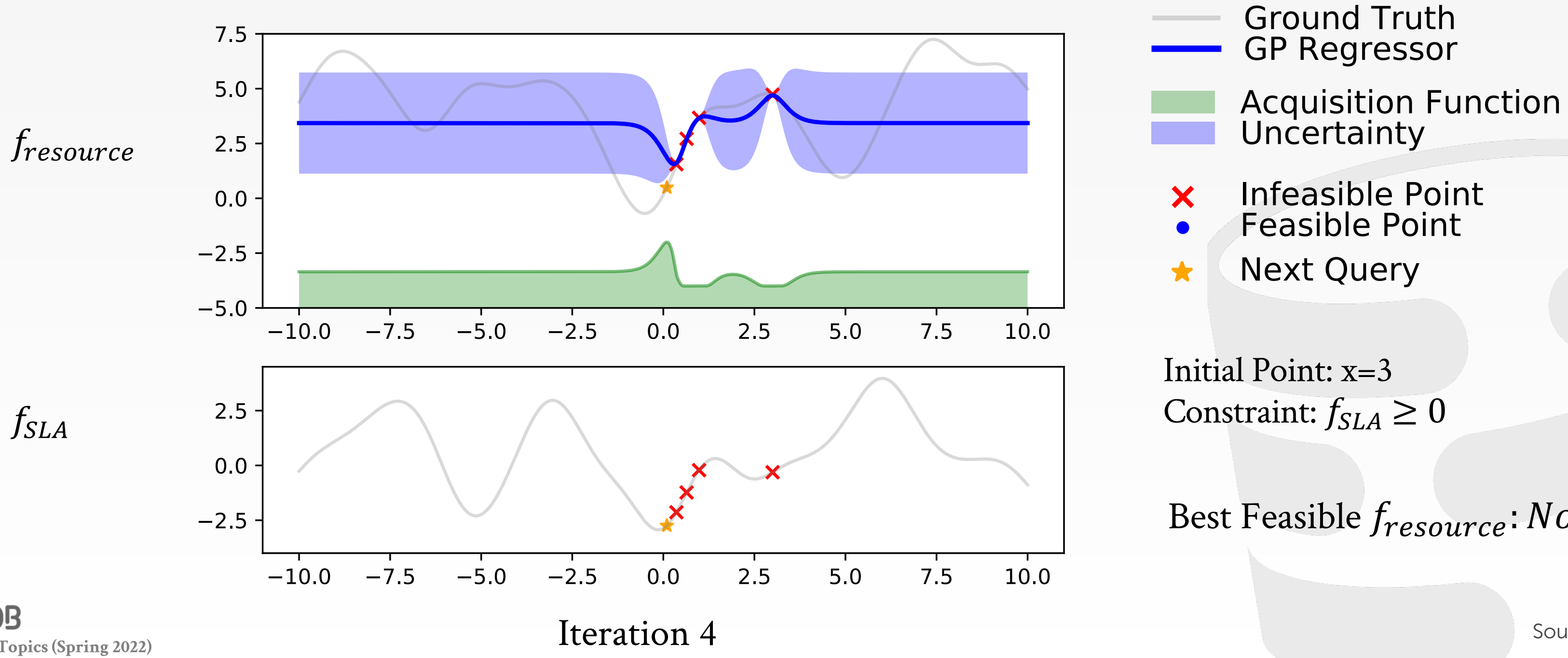
Best Feasible $f_{resource}$: $None$

Iteration 3

Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.



Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: *None*

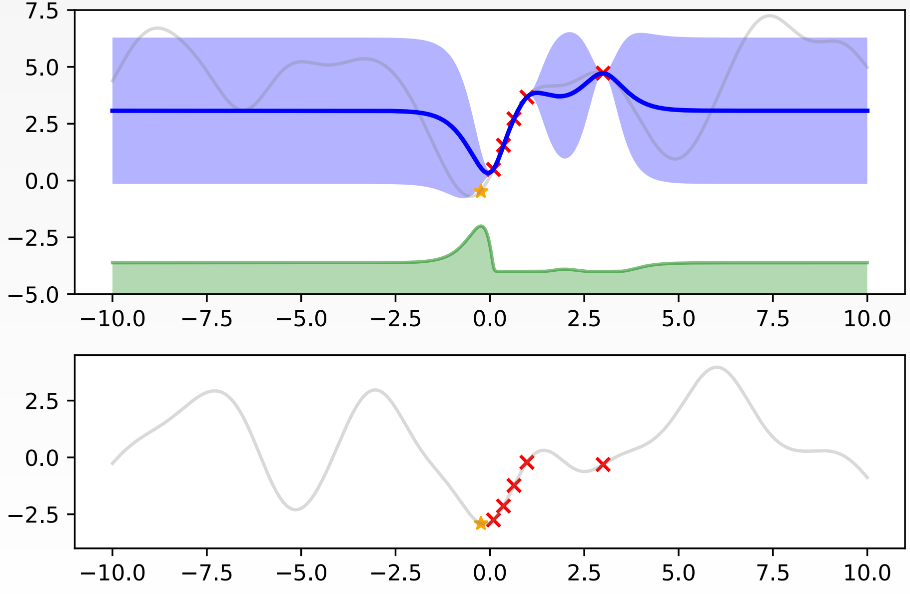
Source: Xinyi Zhang

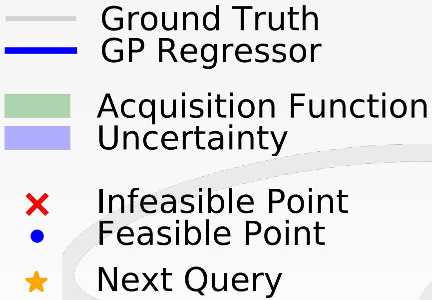
SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.

$f_{resource}$

f_{SLA}





Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

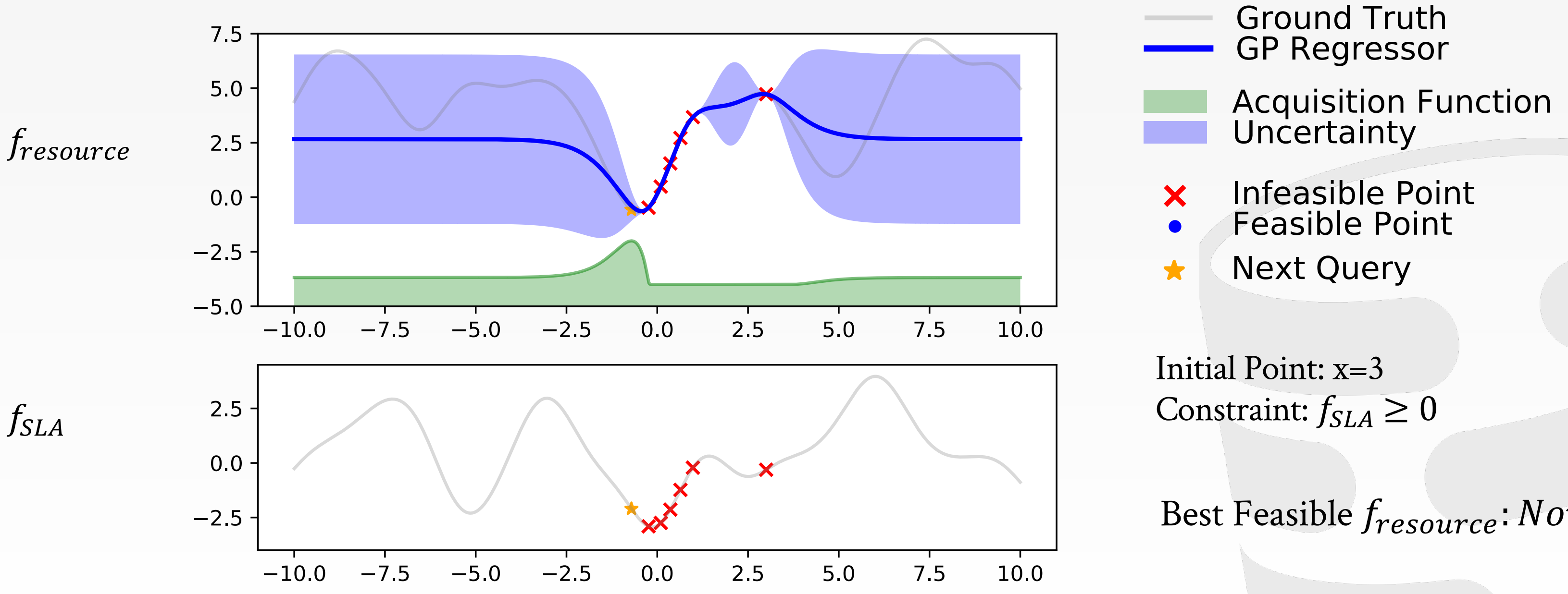
Best Feasible $f_{resource}$: *None*

Iteration 5

Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.



Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

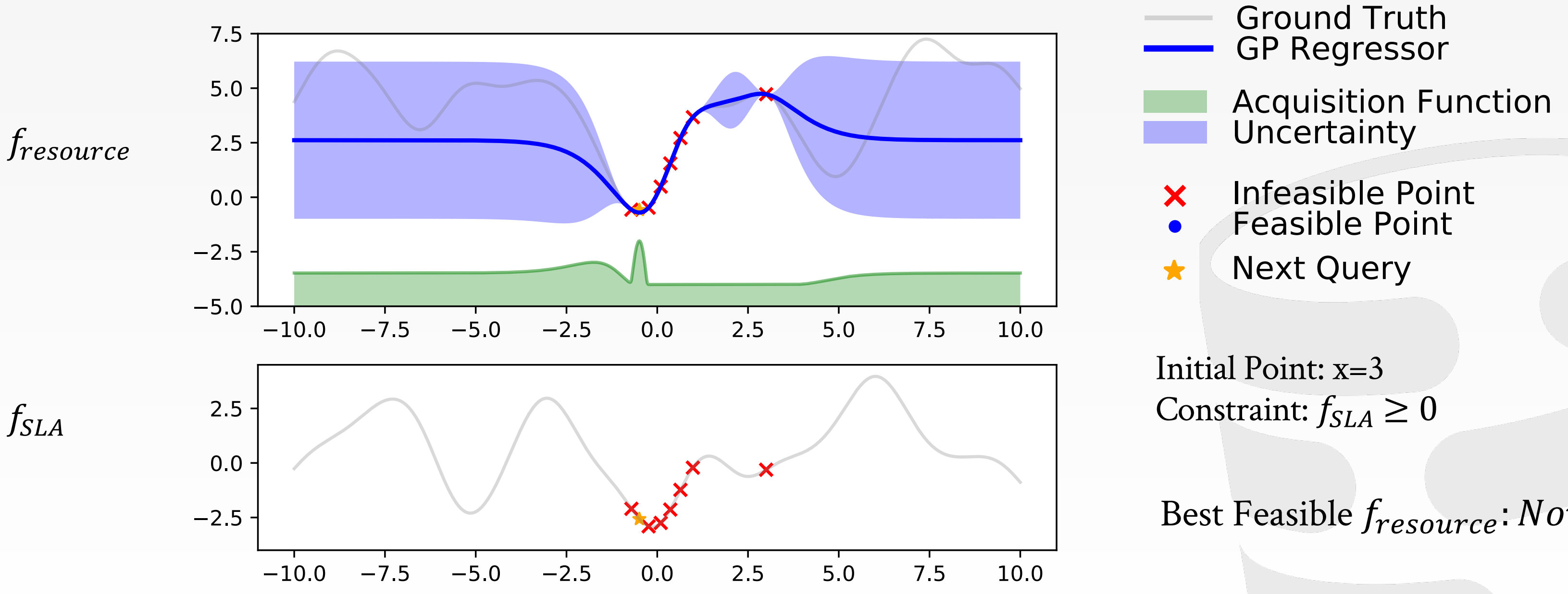
Best Feasible $f_{resource}$: *None*

Iteration 6

Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.



Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

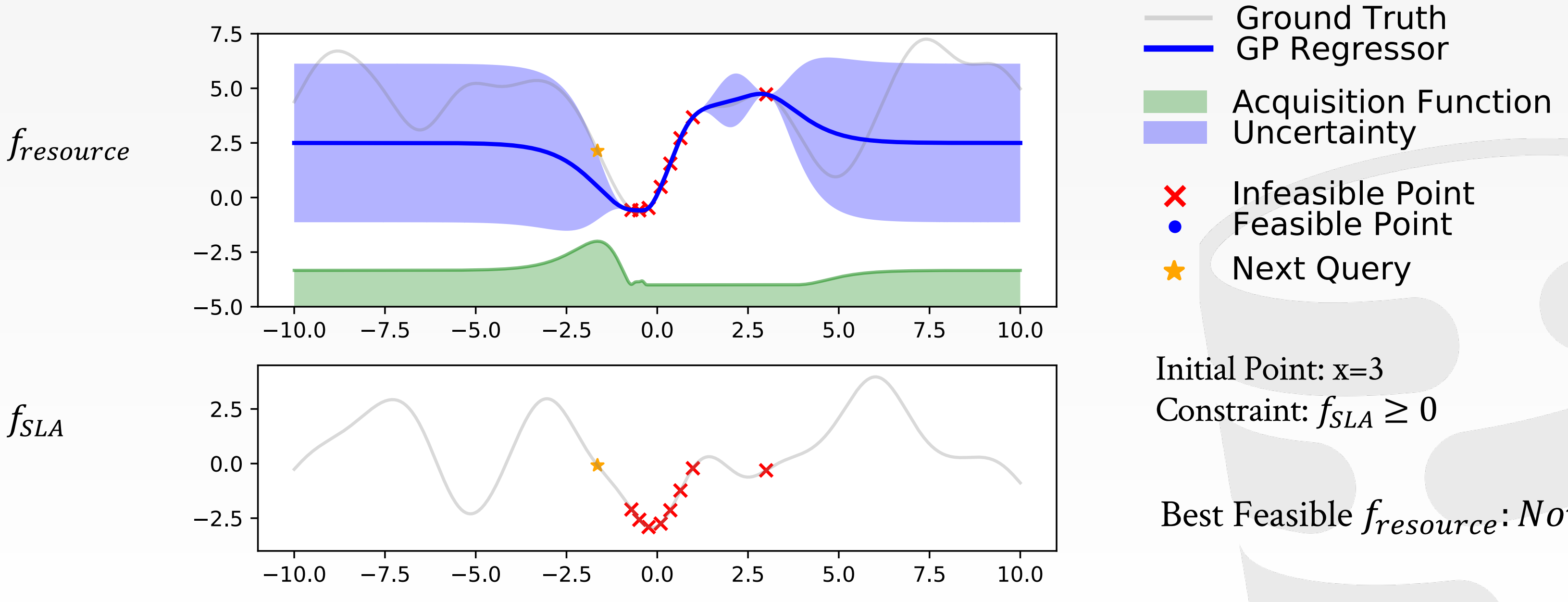
Best Feasible $f_{resource}$: *None*

Iteration 7

Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.



Iteration 8

Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: *None*

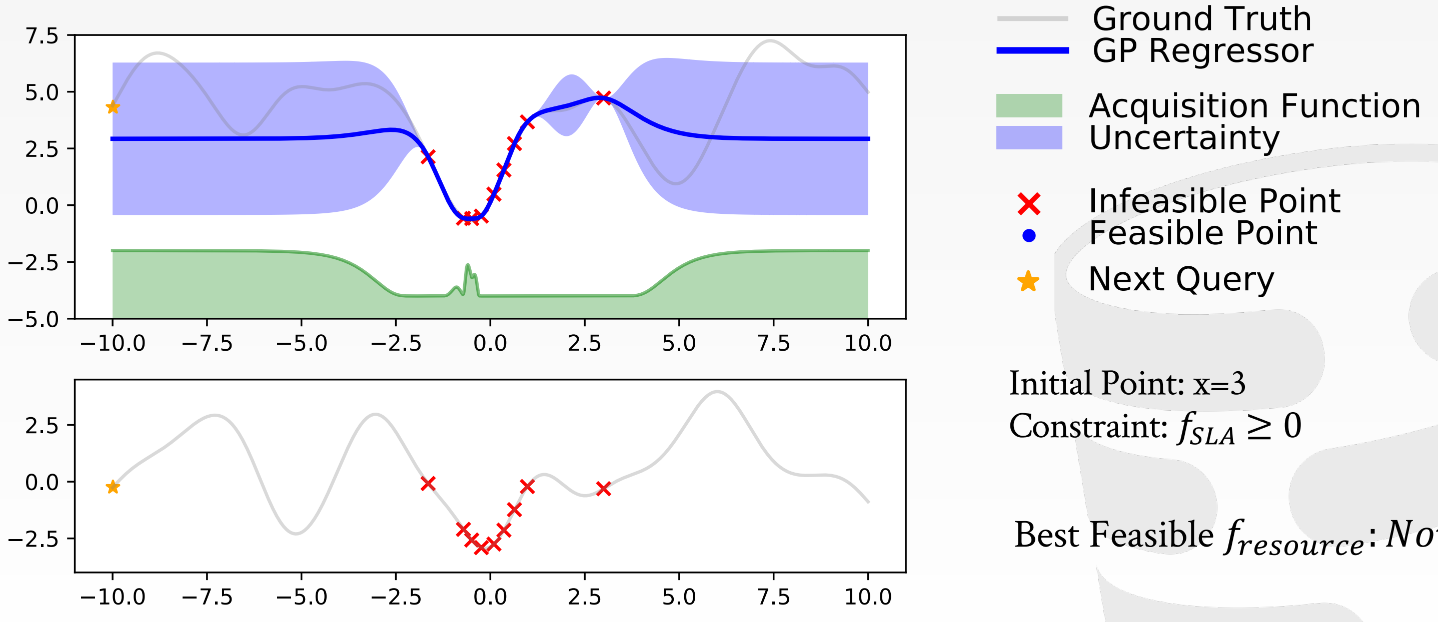
Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.

$f_{resource}$

f_{SLA}



Initial Point: x=3
Constraint: $f_{SLA} \geq 0$

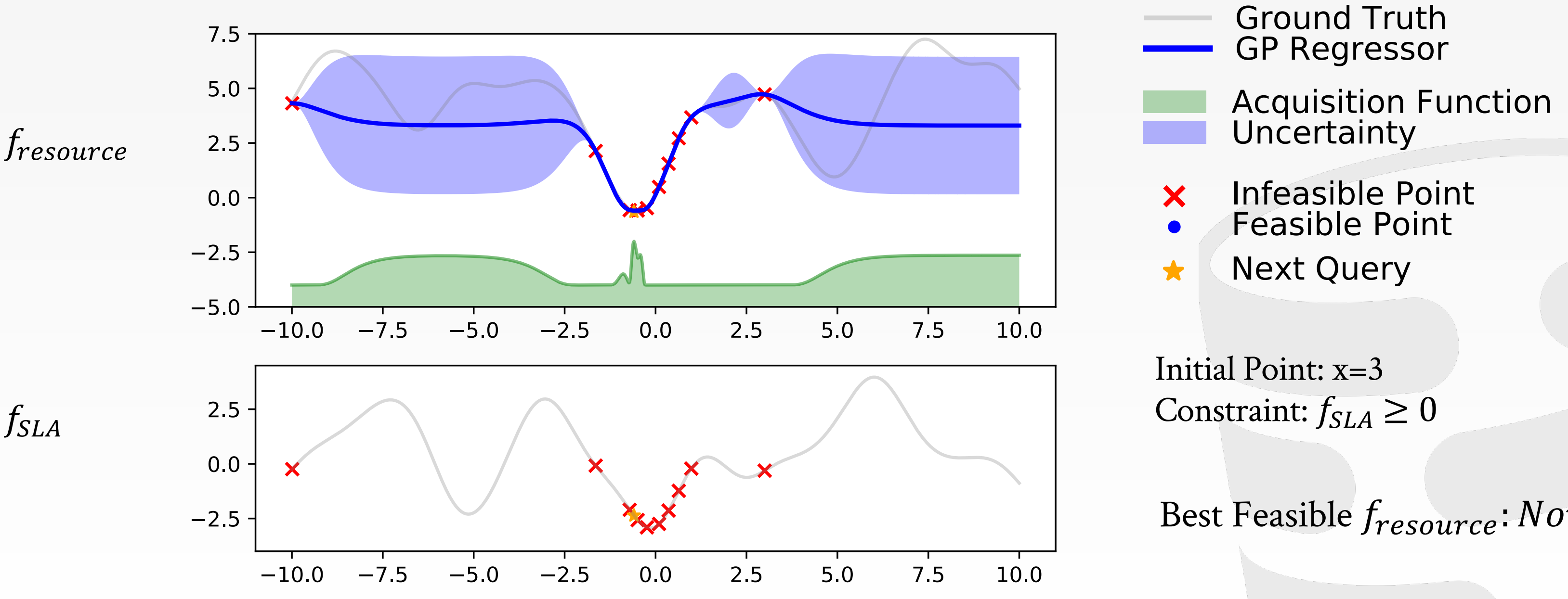
Best Feasible $f_{resource}$: *None*

Iteration 9

Source: Xinyi Zhang

SOLVING TRADITIONAL B.O.

- Tradition Bayesian Optimization uses acquisition function (e.g, the Expected Improvement α_{EI}) to guide the search of the optimal.



Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: $None$

Iteration 10

Source: Xinyi Zhang

SOLVING CONSTRAINED OPTIMIZATION

Current Expected Improvement function:

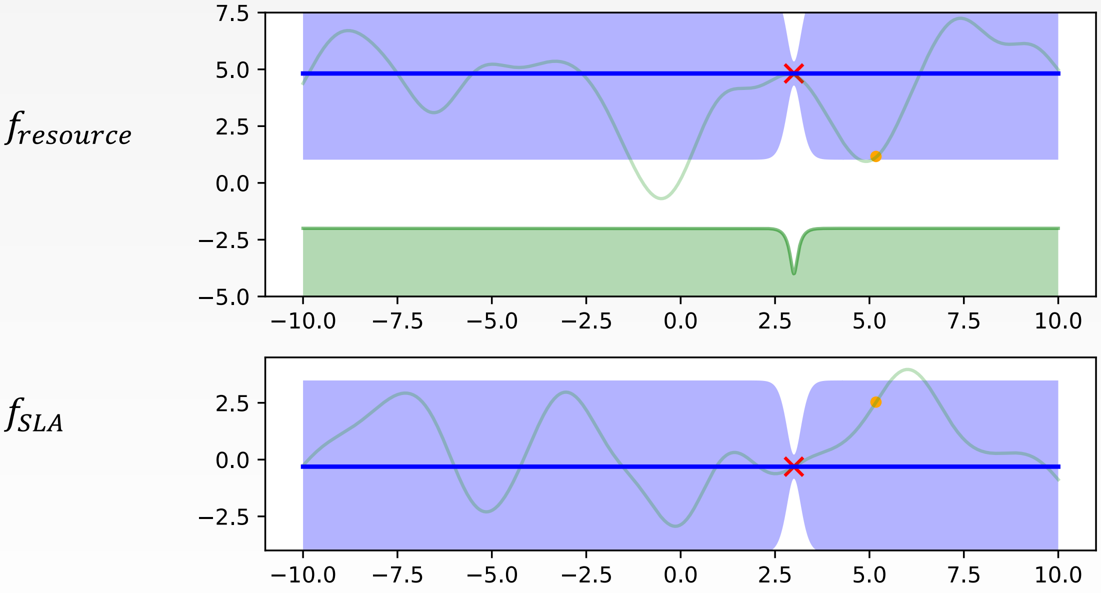
$$\alpha_{EI}(\theta) = \mathbb{E}\left[max(0, f_{res}(\theta_{best}) - f_{res}(\theta)\right]$$

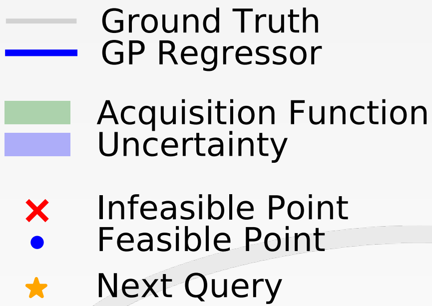
New acquisition function to guide searching with feasible information:

→ Maintain three independent Gaussian Processing models for f_{res}, f_{tps}, f_{lat}

$$\alpha_{CEI}(\theta) = \mathbb{E}\left[\tilde{I}_C(\theta)|\theta\right] = \mathbb{E}\left[\tilde{\Delta}(\theta)\tilde{I}(\theta)|\theta\right] = \mathbb{E}\left[\tilde{\Delta}(\theta)|\theta\right]\mathbb{E}\left[\tilde{I}(\theta)|\theta\right]$$
$$= Pr[\tilde{f}_{tps}(\theta) \geq \lambda_{tps}] \cdot Pr[\tilde{f}_{lat}(\theta) \leq \lambda_{lat}] \cdot \alpha_{EI}(\theta)$$

GUIDING SEARCH IN FEASIBLE REGION





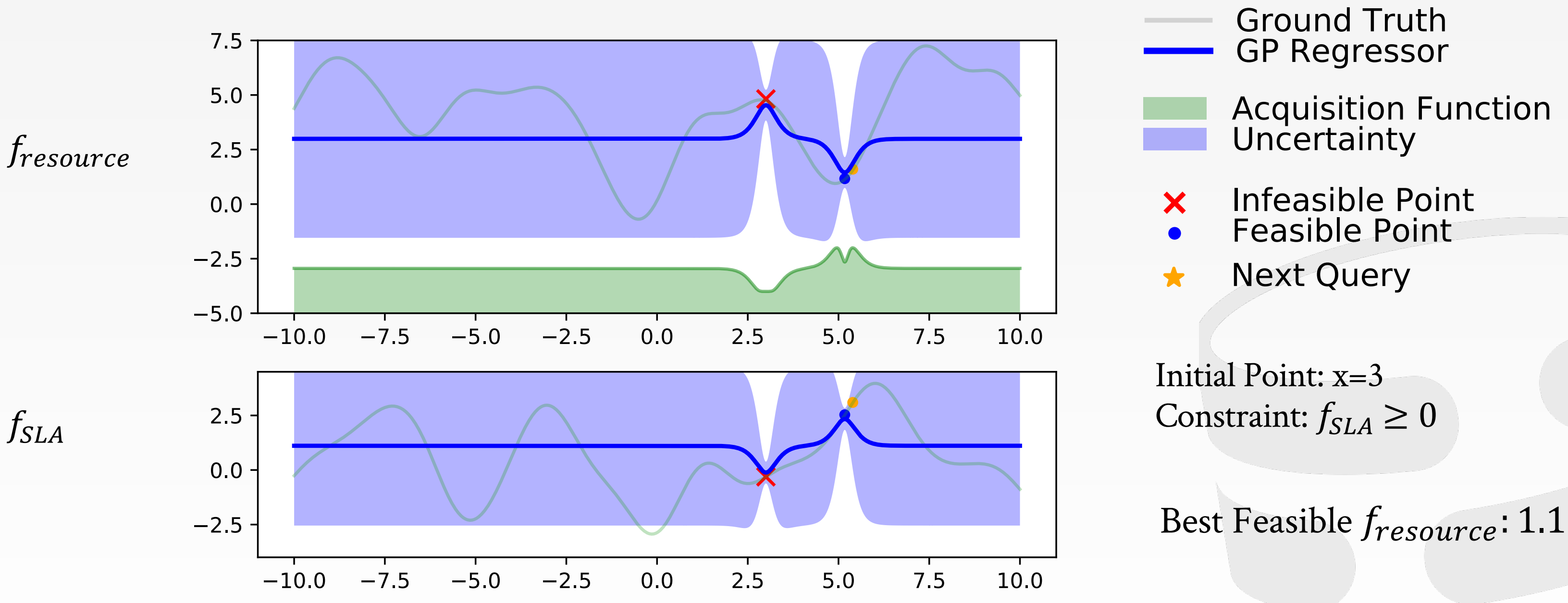
Iteration 1

Initial Point: x=3
Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



Iteration 2

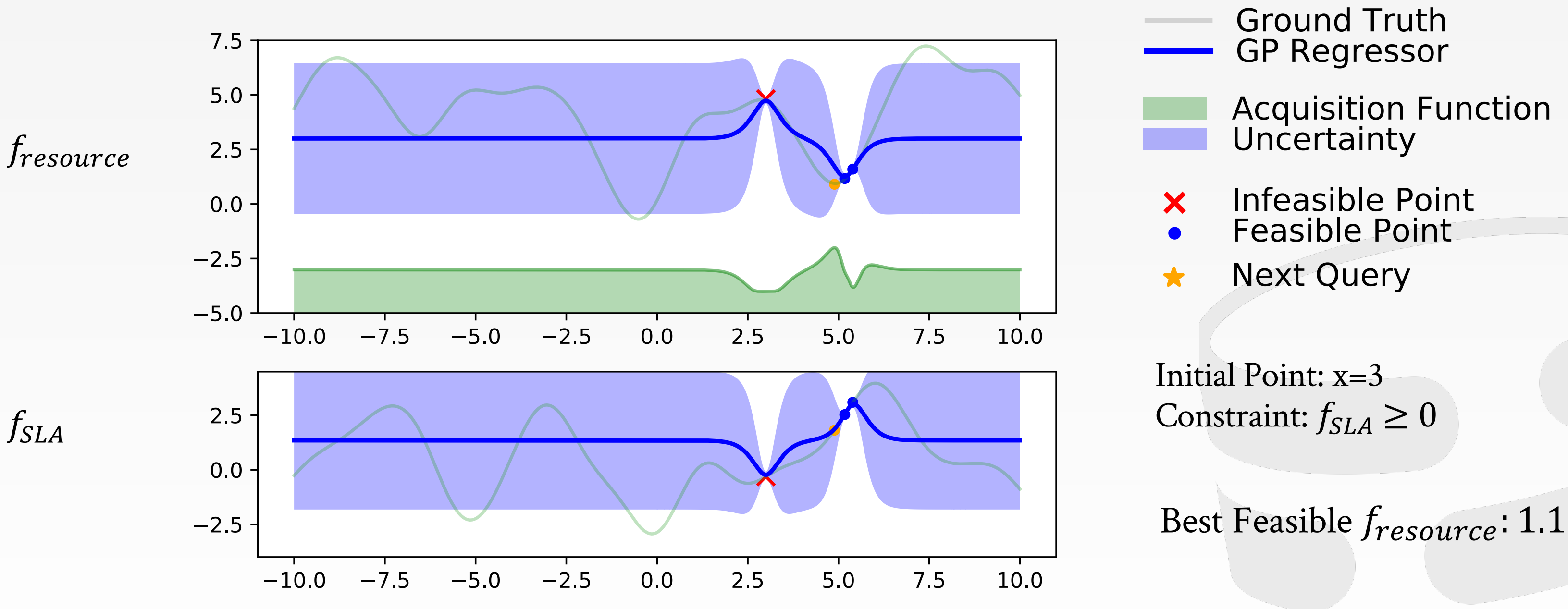
Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



Iteration 3

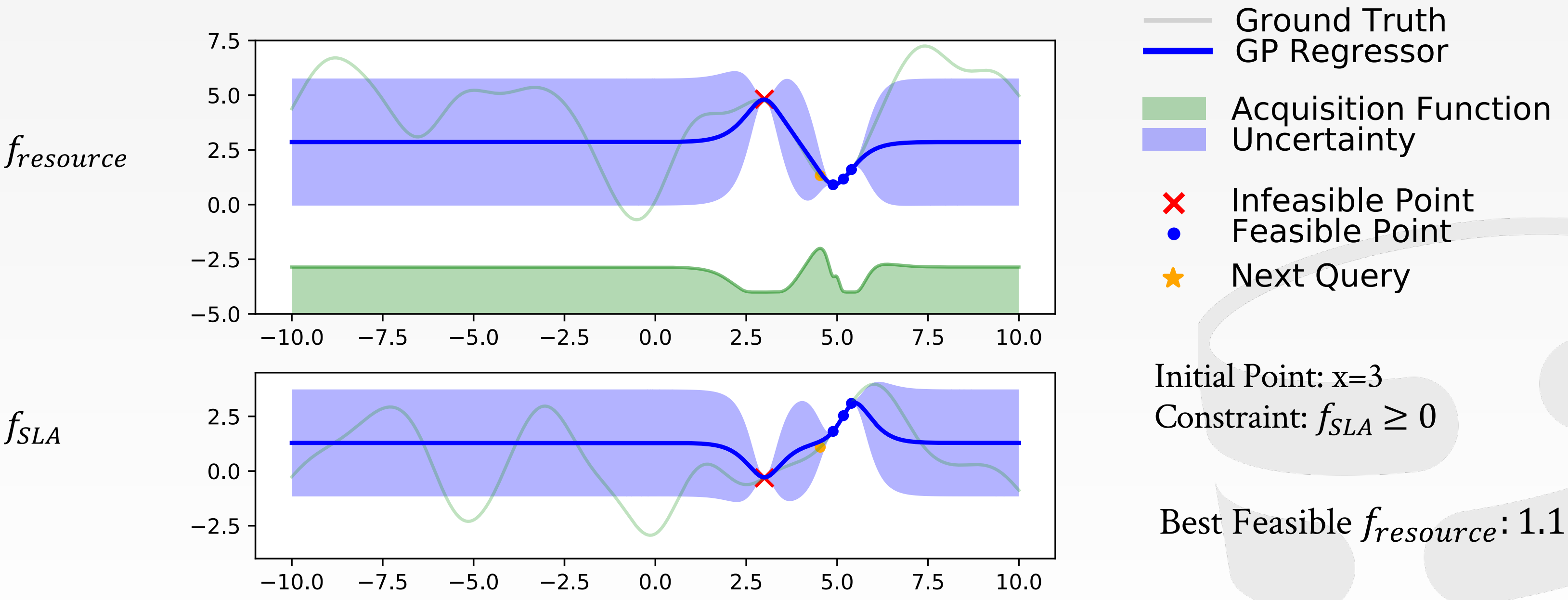
Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



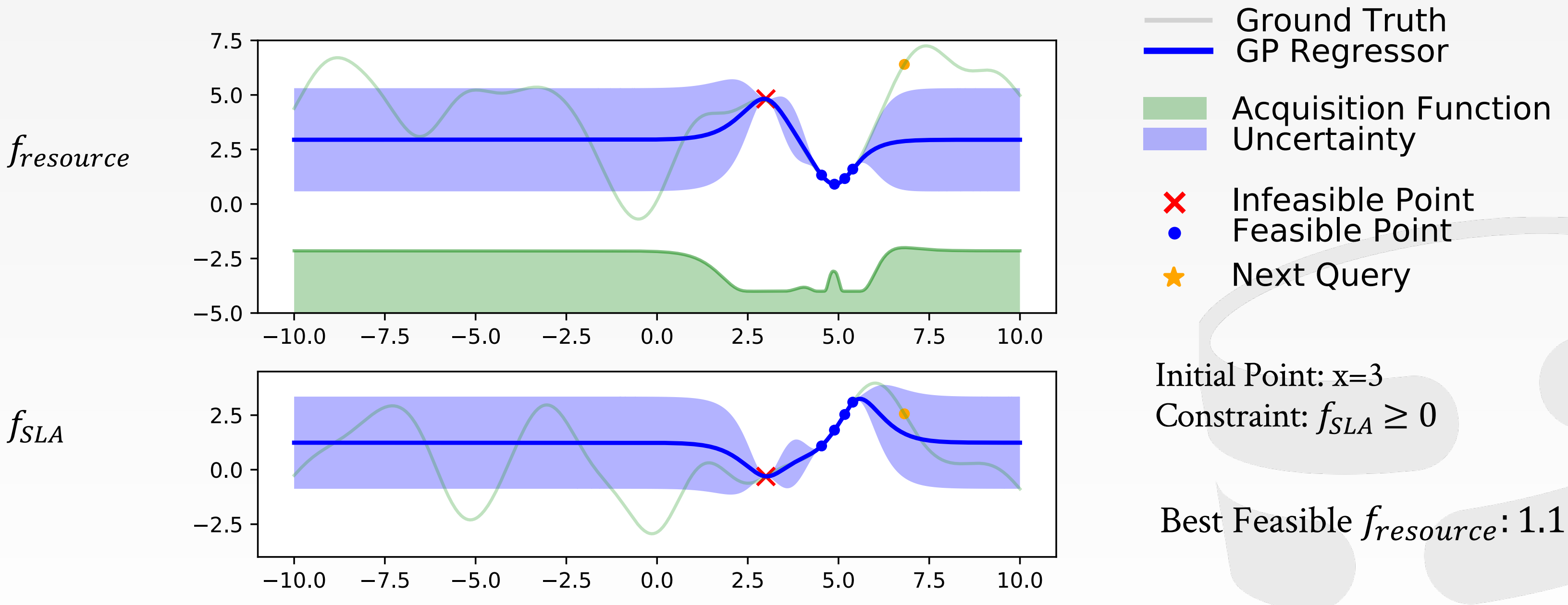
Iteration 4

Initial Point: x=3
Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



Iteration 5

Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

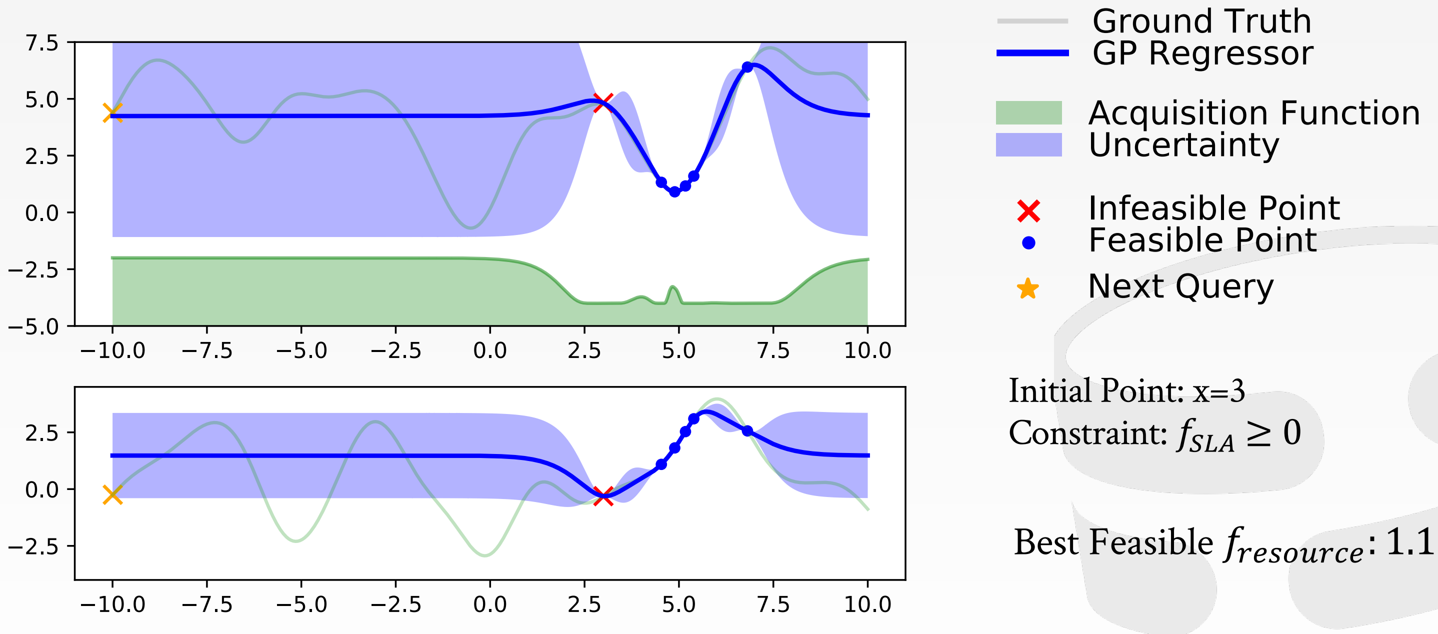
Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION

$f_{resource}$

f_{SLA}



Iteration 6

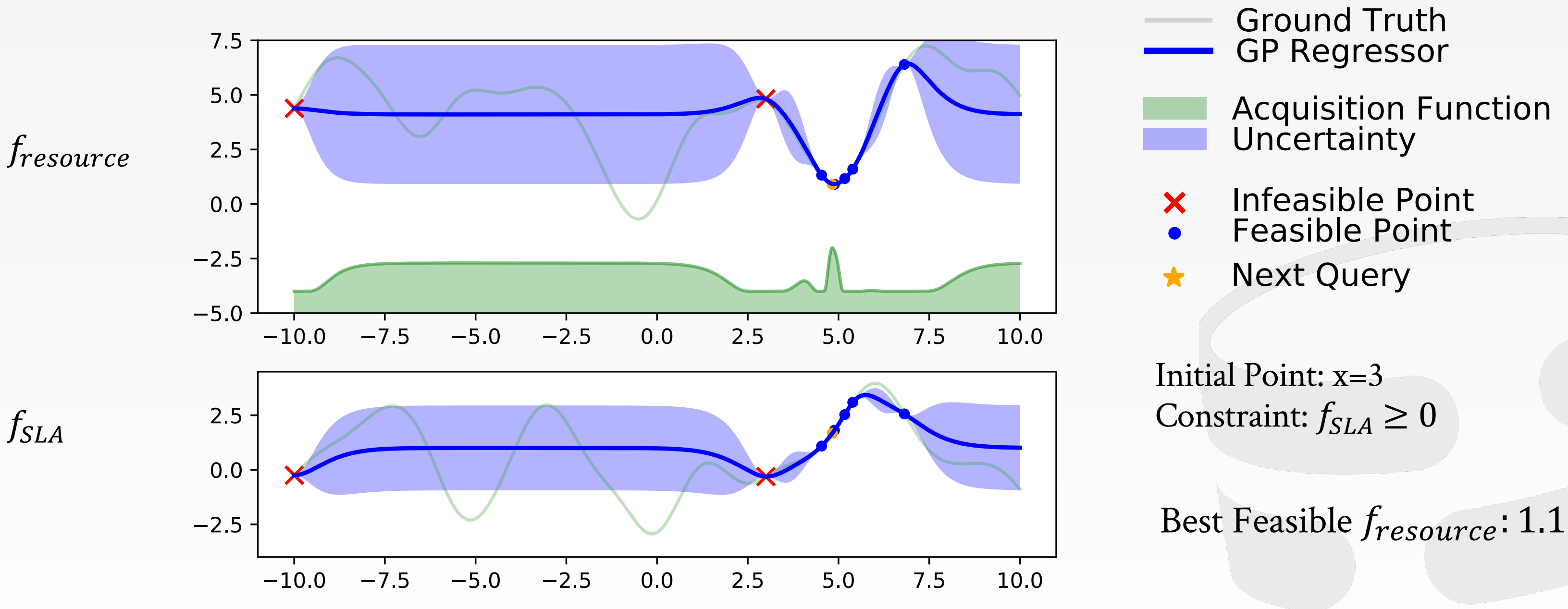
Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



Iteration 7

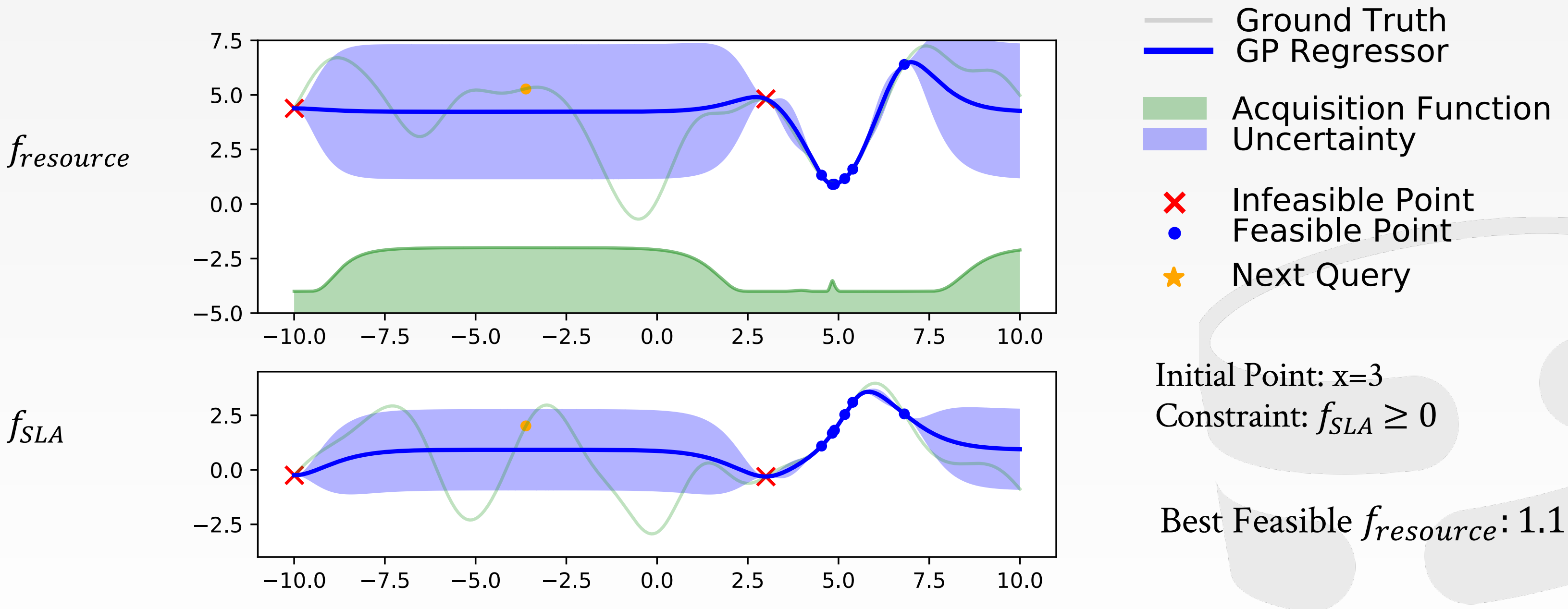
Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



Iteration 8

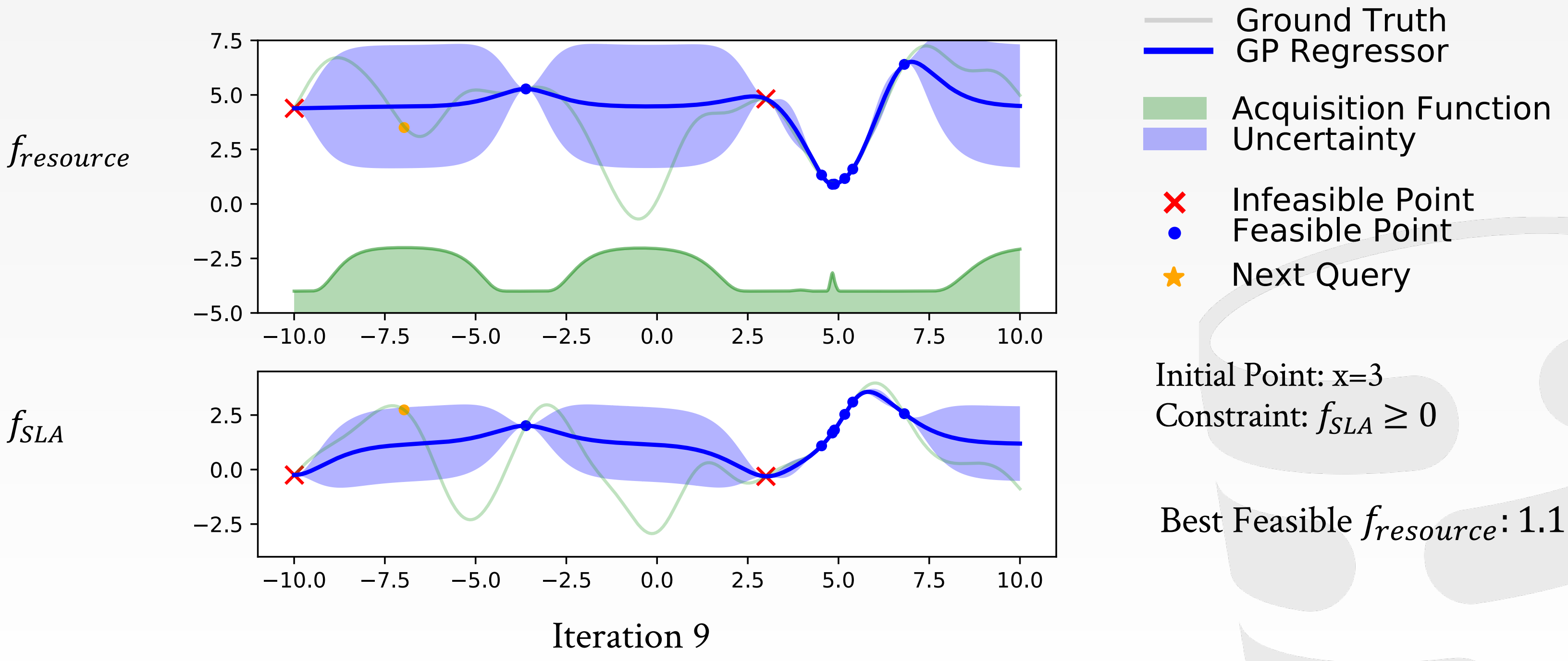
Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



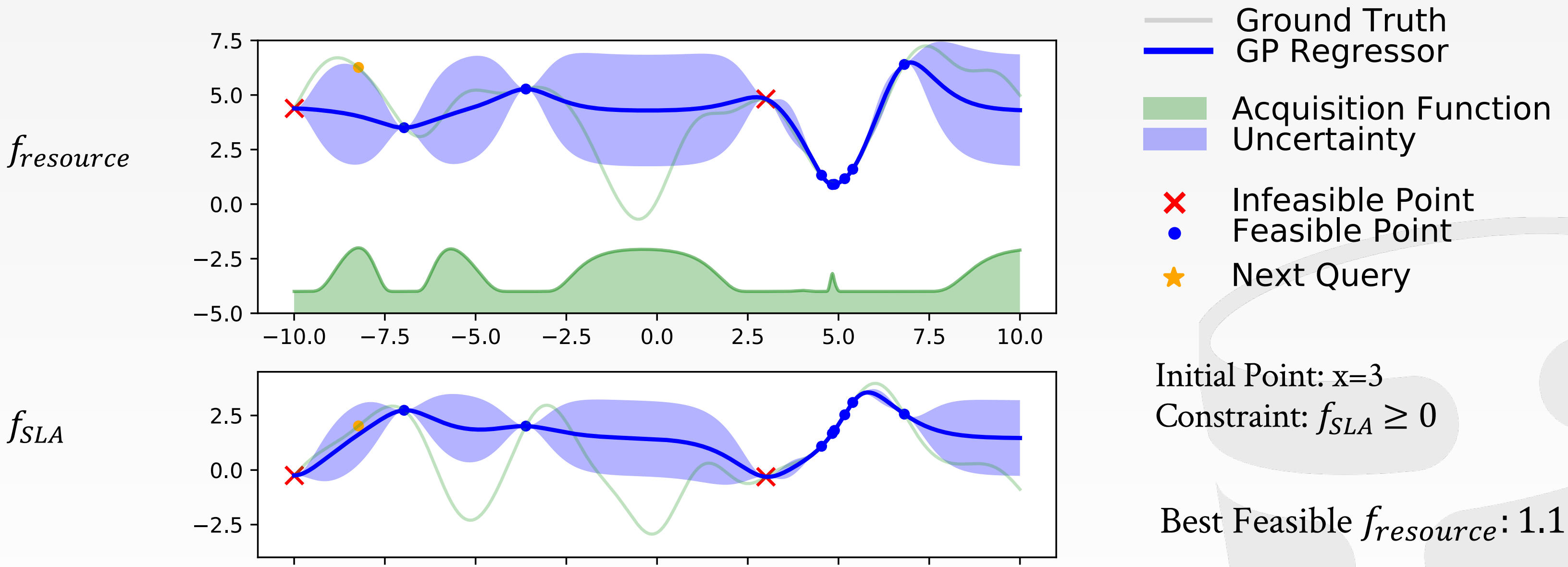
Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

GUIDING SEARCH IN FEASIBLE REGION



Iteration 10

Initial Point: x=3

Constraint: $f_{SLA} \geq 0$

Best Feasible $f_{resource}$: 1.1656

Source: Xinyi Zhang

TODAY'S AGENDA

Overview
Solving Constrained Optimization
Boosting Tuning Process with Meta-Learning
Evaluation
Parting Thoughts

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

BOOSTING: MOTIVATION

Intuitively, the same workloads running on different hardware share information for tuning knobs.

Even for different tasks, the relationship between hidden features can lead to knowledge sharing.

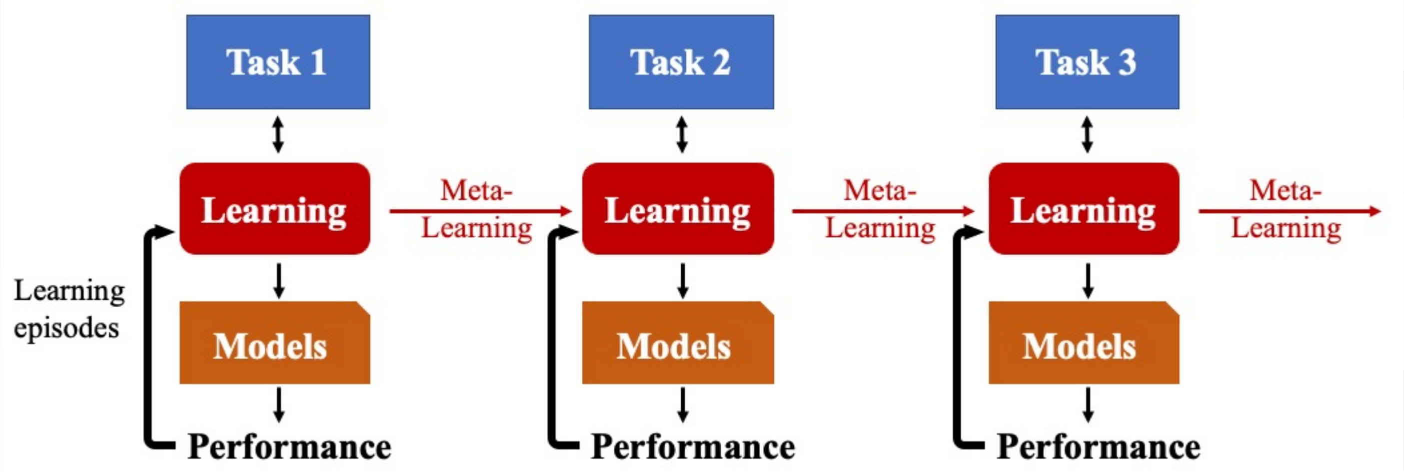
Cloud providers can collect abundant tuning data from numerous tasks and further accelerate the tuning process of new tasks.

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

BOOSTING TUNING: META-LEARNING

Human learns across tasks.

Why? Require less trial-and-error, less data



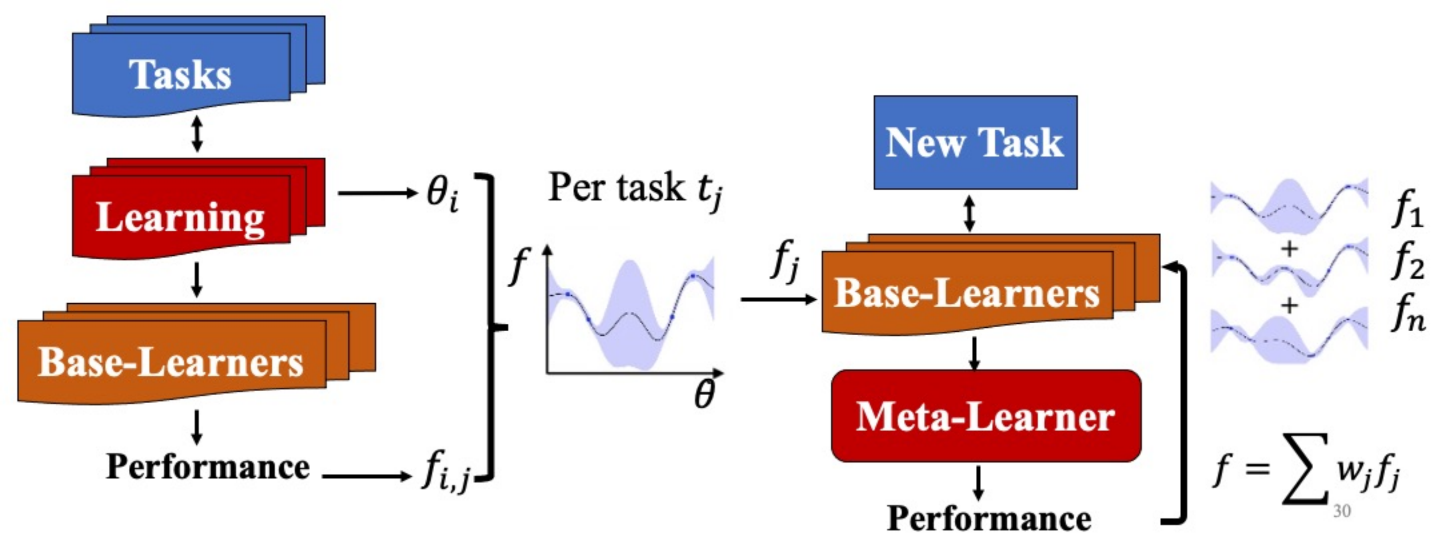
Source: Xinyi Zhang

KNOWLEDGE EXTRACTION

Knowledge is extracted from historical tuning tasks by ensemble

→ Weighted average of predictions (relative values)

→ Using the target base-learner's uncertainty



Source: Xinyi Zhang

HOW TO DETERMINE THE WEIGHTS?

Learning from Meta-Feature → **Learning from Model Predictions**

- Static
- Good initialization

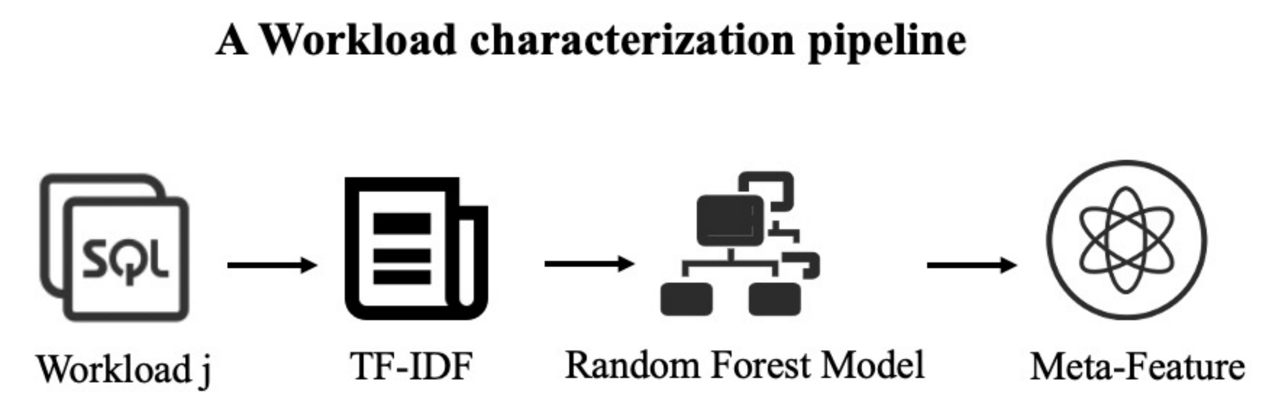
- Dynamic
- Avoid "over-fitting"

Source: Xinyi Zhang

LEARNING FROM META-FEATURE

Meta-features: measurable properties of tasks

ResTune learns meta-features by workload characterization, averaging predicted resource levels over all queries.



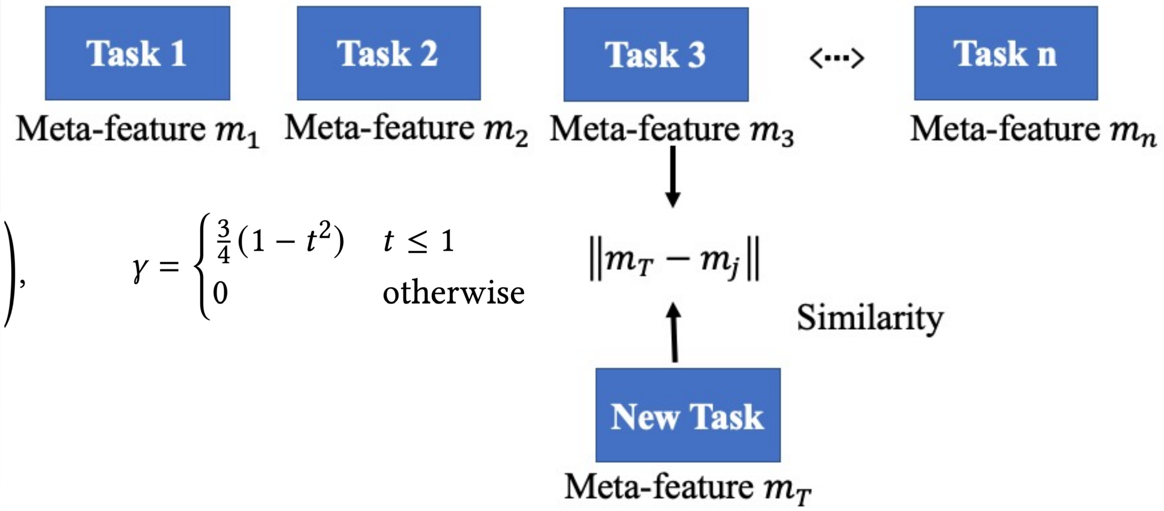
Source: Xinyi Zhang

LEARNING FROM META-FEATURE

The static weight is calculated by the distance between meta-features.

$$g_i = \gamma\left(\frac{\|m_i - m_{T+1}\|_2}{\rho}\right),$$

$$\gamma = \begin{cases} \frac{3}{4}(1 - t^2) & t \le 1 \\ 0 & \text{otherwise} \end{cases}$$

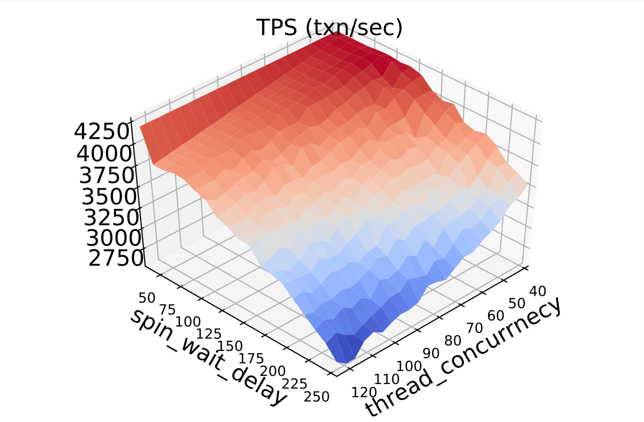


Source: Xinyi Zhang

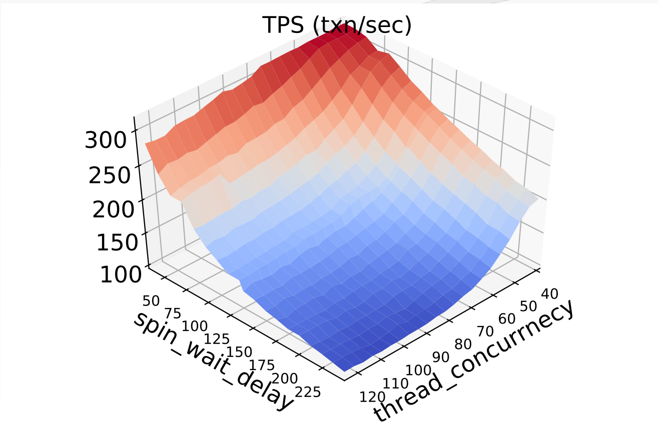
LEARNING FROM MODEL PREDICTIONS

A base-learner's similarity is defined as how accurate it can predict the performance of the target task.

Challenge: The performances can differ in scale significantly among various hardware environments in the cloud.



Instance A



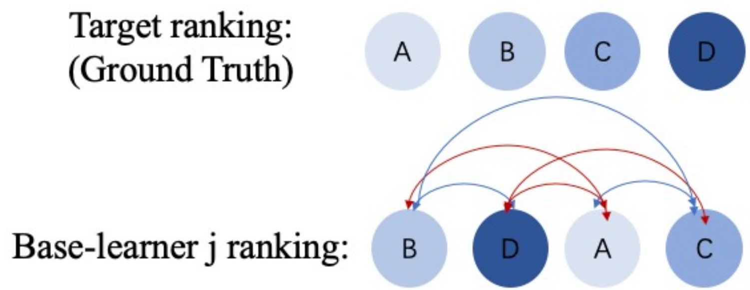
Instance B

Source: [Xinyi Zhang]

LEARNING FROM MODEL PREDICTIONS

Observation: the actual values of the predictions do not matter, since we only need to identify the location of the optimum.

ResTune weight base learners by calculating the ranking loss against target observations



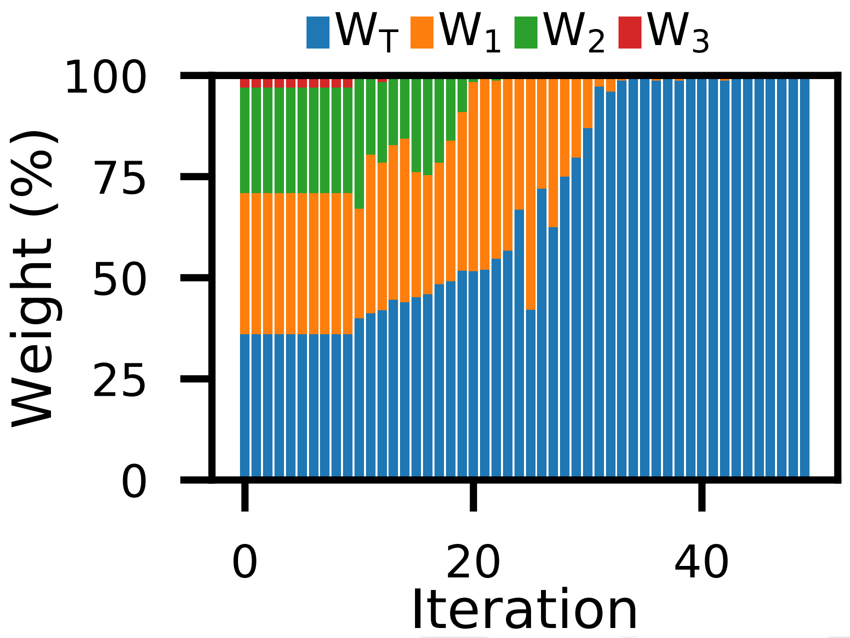
$$\text{Ranking Loss for j} = \frac{\#\,Misranking\ pairs}{\#Pairs} = \frac{6}{12}$$

Source: Xinyi Zhang

ADAPTIVE WEIGHT SCHEMA

- Static weight assignment:
  - Meta-features gives a coarse-grained abstraction about task properties
  - Suggest knobs that are promising according to similar historical tasks

- Dynamic weight assignment:
  - Use rankings to measure the similarity of tasks
  - Avoid over-fitting by shrinking historical base learners' weight



Source: Xinyi Zhang

ARCHITECTURE OVERVIEW

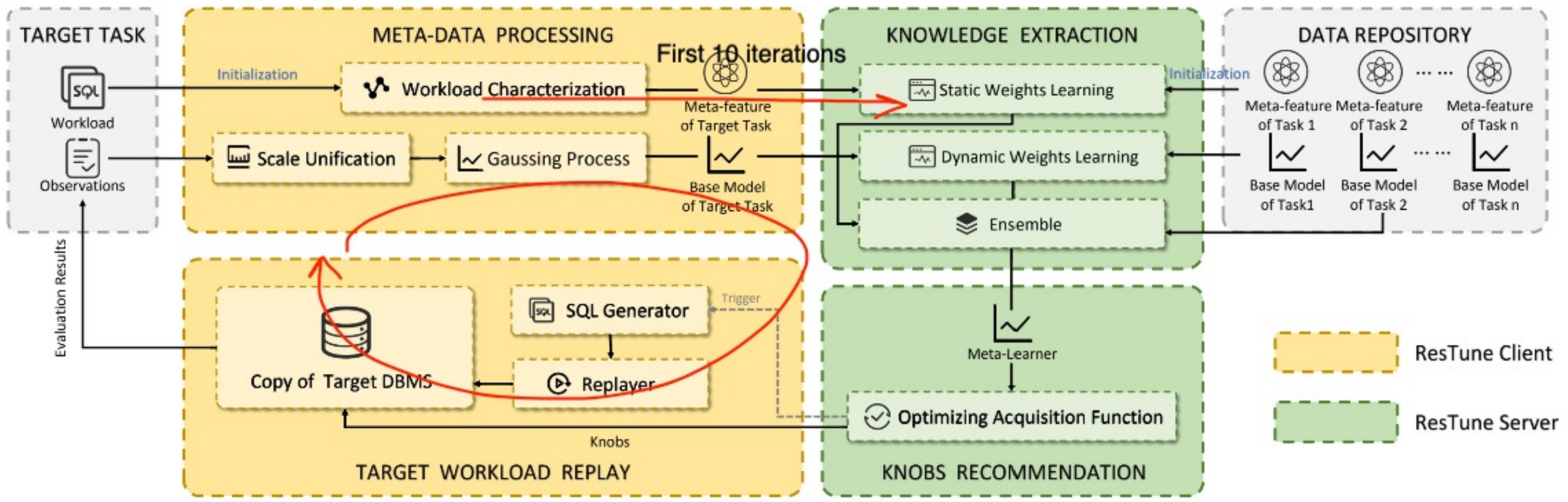


Figure 2: Overall Architecture of ResTune

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

TODAY'S AGENDA

Overview

Solving Constrained Optimization

Boosting Tuning Process with Meta-Learning

Evaluation

Parting Thoughts

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

EVALUATIONS

DBMS: version 5.7 of MySQL RDS

Pre-selected 40 knobs (14 CPU, 6 memory, 20 I/O) 😊

Table 1: Hardware Configurations for Database Instances

	A	B	C	D	E	F
CPU	48 cores	8 cores	4 cores	16 cores	32 cores	64 cores
RAM	12GB	12GB	8GB	32GB	64GB	128GB

Workloads

→ SYSBENCH, TPC-C, Twitter, Hotel Booking, Sales

Data Repository

→ 34 past tuning tasks → base learners

→ 17 different workloads and 2 HW environments (A, B)

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

EVALUATIONS

Methods:

→ **Default**: The default knobs provided by experienced DBA

→ **iTuned**: Objective changed to minimizing resource utilization, w/o SLA

→ **OtterTune-w-Con**: Acquisition function replaced with CEI

→ **CDBTune-w-Con**: Replaced latency in reward function to resource utilization; max (r, 0) when SLA satisfied, min(r, 0) otherwise

→ **ResTune-w/o-ML**: ResTune w/o data repository, learned from scratch

→ **ResTune**: Proposed approach

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

EFFICIENCY

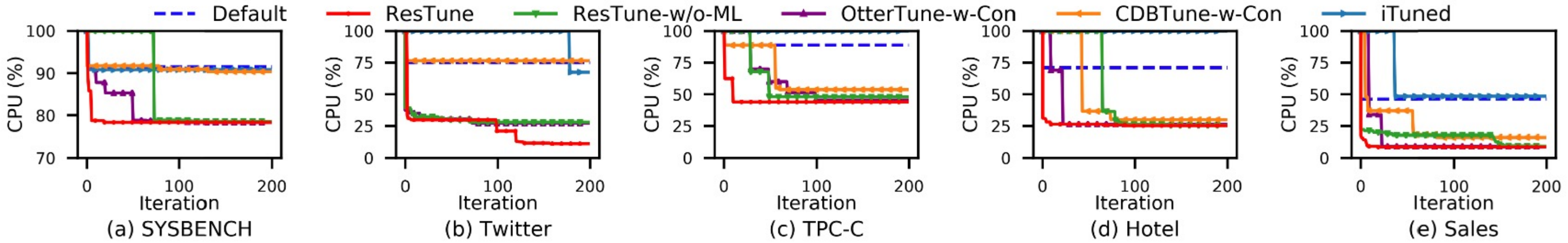


Figure 3: Efficiency Comparison

Original setting without hold-out 🤔

Observations

→ ResTune-w/o-ML better than iTuned, CDBTune-w-Con

- Both: no historical tuning data
- iTuned: no SLA constraints
- CDBTune: RL model, but optimal solution not related to action

→ ResTune achieves 7.38x speedup over OtterTune-w-Con

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

GENERALIZATION: HW

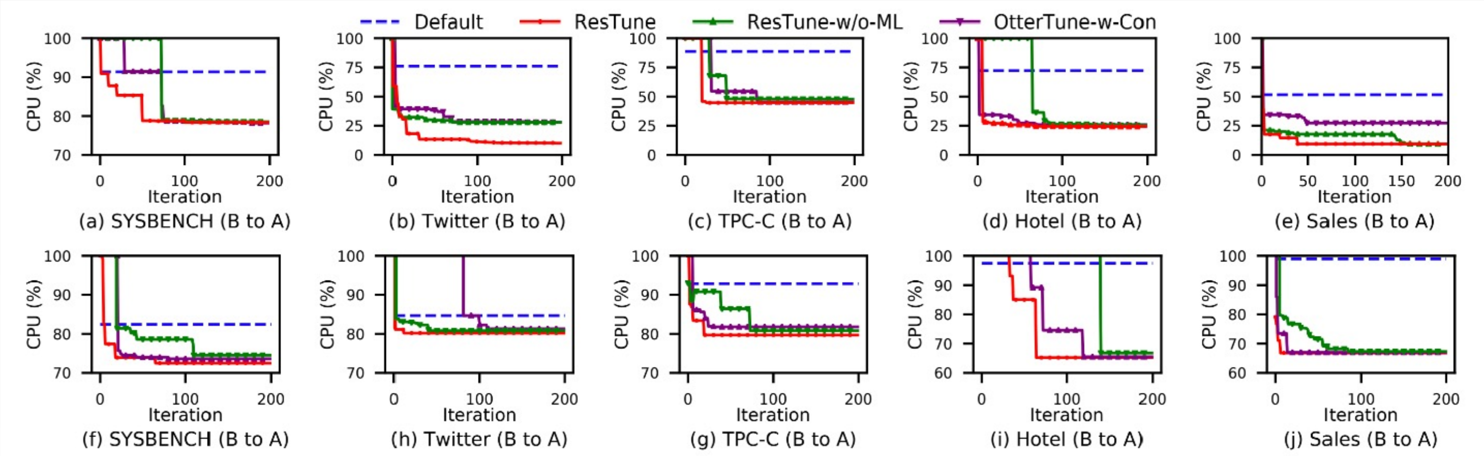


Figure 4: Performance Adapting to Different Hardware Environments

Table 4: Workload Adaptation on More Instances

Instance			C	D	E	F
SYSBENCH	Improvement	Restune	5.02%	8.13%	17.16%	20.38%
		Restune-w/o-ML	3.34%	7.58%	16.76%	19.96%
	Iteration	Restune	37	64	100	35
		Restune-w/o-ML	57	80	115	53
		Speed Up	35%	20%	14%	34%
TPC-C	Improvement	Restune	4.96%	19.22%	33.26%	47.60%
		Restune-w/o-ML	2.78%	18.28%	33.09%	42.62%
	Iteration	Restune	12	25	45	18
		Restune-w/o-ML	99	47	79	25
		Speed Up	87.87%	46.80%	43.03%	28%

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

GENERALIZATION: WORKLOAD

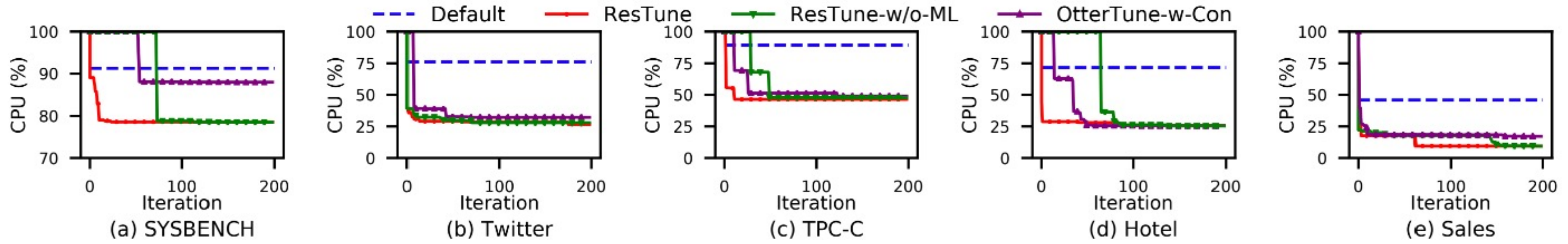


Figure 5: Performance Adapting to Different Workloads

- ResTune's ensemble + adaptive weights
  - Utilize more experience from multiple workloads than OtterTune
  - Getting a quick start
  - Stops matching when there is no similar workload
- Workload mapping: ranking loss (surface measurement) better than absolute similarity

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

TUNING OTHER RESOURCES

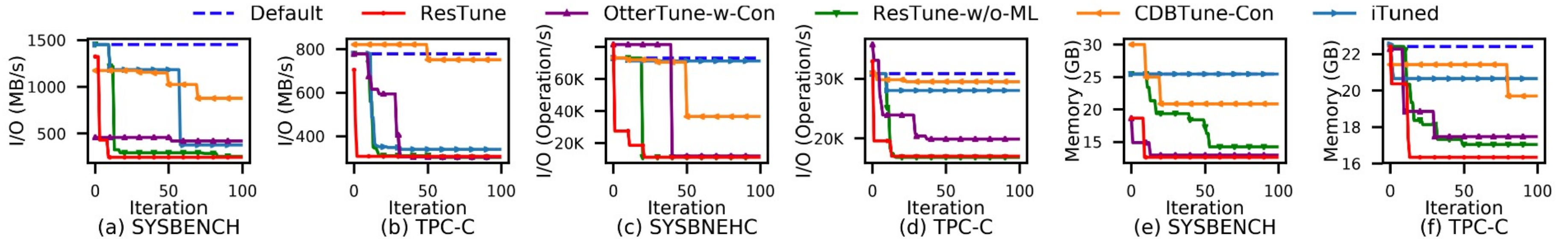


Figure 9: Tuning Other Types of Resources

- I/O heavy scenario for both: buffer pool size ~ ½ data size
- I/O objective: BPS (60%-80%), IOPS (84%-90%)
- Memory objective: buffer pool size (39%)

RESTUNE: RESOURCE ORIENTED TUNING BOOSTED BY META-LEARNING FOR CLOUD DATABASES
SIGMOD 2021

TODAY'S AGENDA

Overview
Solving Constrained Optimization
Boosting Tuning Process with Meta-Learning
Evaluation
Parting Thoughts

PARTING THOUGHTS

Industry's view of knob tuning

- Resource-oriented, SLA restrictions

Ensemble learning to transfer historical knowledge

SOTA: an **offline** automated pipeline with iterative replaying

- Assuming workload pattern has not changed
  - Should start again if changed
- Online tuning is harder
  - Online service availability; Faster & less iterations; Stable improvements …

NEXT CLASS

Knob/Parameter Tuning IV

REFERENCES

[1] Zhang Xinyi, Hong Wu, Zhuo Chang, Shuowei Jin, Jian Tan, Feifei Li, Tieying Zhang, and Bin Cui. "Restune: Resource oriented tuning boosted by meta-learning for cloud databases." In *Proceedings of the 2021 International Conference on Management of Data*, pp. 2102-2114. 2021.

Credit and thanks to Xinyi Zhang's method demonstration in the SIGMOD talk!