Towards Dynamic and Safe Configuration Tuning for Cloud Databases

Xinyi Zhang
Hong Wu, Li Yang, Jian Tan, Feifei Li, Bin Cui
Background

Throughput as a function of configurations

$25/75$ read/write workload

read-only workload

$75/25$ read/write workload

Figure 1: Two-knob example

$k_1$ denotes sort_buffer_pool_size and $k_2$ denotes max_heap_table_size.
Background

Offline Methods

![Workflow of offline methods.](image)

Figure 2: Workflow of offline methods.
Background

Offline methods fail to adapt to dynamic environment.

Figure 3: Dynamic environment in the cloud.
Background

Offline Methods

Figure 2: Workflow of offline methods.
Background

Online Tuning

Figure 4: Online Tuning.
Preliminaries
Challenges for Online Tuning

**Dynamicy**
The tuner is capable of responding to the dynamic environment (e.g., workload and its underlying data) adaptively.

**Safety**
The tuner should recommend configurations that do not downgrade the database performance during the tuning process.
Problem Statement

**Dynamicity**
At each iteration $t$, the tuner receives context $c_t$ and outputs a configuration $\theta_t$ to maximize the database performance $f$.

**Safety**
We additionally need to ensure that, for each tuning iteration $t$, $f(\theta_t, c_t) \geq \tau$ holds, where $\tau \in \mathbb{R}$ is a specific safety threshold.

$$\arg \max_{\theta_t} f(\theta_t, c_t)$$
$$\text{s.t. } f(\theta_t, c_t) \geq \tau$$
Methodology
Methodology

OnlineTune: A Safe and Dynamic Online Tuner

➢ Contextual performance modeling

➢ Safe configuration recommendation
OnlineTune: A Safe and Dynamic Online Tuner

Figure 5: OnlineTune Workflow
Methodology

Context Featurization

➢ Workload
  • Query arrival rate
  • Query composition

➢ Data
  • Estimate of rows to be examined by queries
  • The percentage of rows filtered by table conditions in queries
  • Whether an index is used.

Figure 6: LSTM auto-encoder network
Methodology

Safe Configuration Recommendation

➢ Inspired by the trust region optimization, OnlineTune reduce the optimization over the whole configuration space into a sequence of subspace optimizations.

➢ OnlineTune maintains a subspace for each surrogate model, restricts its optimization in the subspace, and gradually adapts the subspace.
Methodology

Subspace Adaptation

Figure 6: Visualization of subspace adaptation.
Methodology

Safety Assessment

➢ Black-box knowledge
  • \( \mu(\theta, c) - \beta \sigma(\theta, c) > \tau \)

➢ White-box knowledge (heuristics rules)
  • Examples
    • The total buffer size can not exceed the physical memory capacity of the deployed machine.
    • Increasing the join buffer size if #joins without indexes per day is larger than 250.
    • The value of maximum thread concurrency should be larger than half of the number of virtual CPUs.
Methodology

Candidate Selection

➢ We adopt Upper Confidence Bound (UCB) constrained to the safety set as a sampling criterion.

➢ To expand the safe subspace explicitly, OnlineTune also selects the safe configurations at the boundary of the safety set.
Methodology

More in our paper...

Performance Modeling with Contexts
➢ Extends the Gaussian Process to support dynamic environments.

Bounding The Complexity of Gaussian Process
➢ Propose a clustering and model selection strategy.
Evaluation
Evaluation

Setting

Setup
➢ Version 5.7 of MySQL RDS on a cloud instance with 8 vCPU and 16GB RAM.
➢ We tune 40 dynamic configuration knobs.
➢ We use the DBA default configuration as the initial safety set and its performance as the safety threshold.

Metrics
➢ Cumulative performance during tuning
➢ Safety: the number of unsafe configuration recommendations (#Unsafe) and the number of system failures (#Failure).
Evaluation

Baselines

- **DBA Default** is the configuration provided by experienced DBAs.
- **BO** is a Bayesian Optimization approach, widely used in database configuration tuning.
- **DDPG** is a reinforcement learning agent which is used to tune the database configuration.
- **QTune** is a query-aware tuner that supports workload-level tuning.
- **ResTune** adopts constrained Bayesian Optimization to maximize the performance with safety constraints.
- **MysqlTuner** is a white-box tuning tool that examines DBMS metrics and uses static heuristics to suggest configurations.
Takeaway:

➢ OnlineTune finds the workload-specific configuration
  - OnlineTune achieves $16.2\% - 21.9\%$ improvement on cumulative performance than the DBA default.
  - OnlineTune achieves $14.4\% - 165.3\%$ improvement on cumulative performance than existing offline approaches.

➢ OnlineTune reliably respects the safety requirement when tuning the online database.
  - OnlineTune reduces $91.0\% - 99.5\%$ unsafe recommendations, compared to the offline methods.

Figure 7: Cumulative performance and safety statistics when tuning dynamic workloads
Evaluation

Iterative Performance on OLTP-OLAP circle

Figure 8: Iterative Performance on OLTP-OLAP circle
Evaluation

Ablation Study on Safe Exploration

Figure 8: Ablation study on safe exploration.
We introduce OnlineTune, an online tuning system that is aware of the dynamic environments and optimizes the database safely.

OnlineTune featurizes the dynamic environmental factors as context feature and leverages Contextual Bayesian Optimization to optimize the context-configuration joint space.

We propose a safe exploration strategy, greatly enhancing the safety of online tuning.

Compared with the state-of-the-art methods, OnlineTune achieves 14.4%~165.3% improvement on cumulative performance while reducing 91.0%~99.5% unsafe configuration recommendations.
Thanks for Listening!

zhang_xinyi@pku.edu.cn