An Inquiry into Machine Learning-based Automatic Configuration Tuning Services on Real-World Database Management Systems

Dana Van Aken , Dongsheng Yang, Sebastien Brillard, Ari Fiorino Bohan Zhang , Christian Bilien, Andrew Pavlo

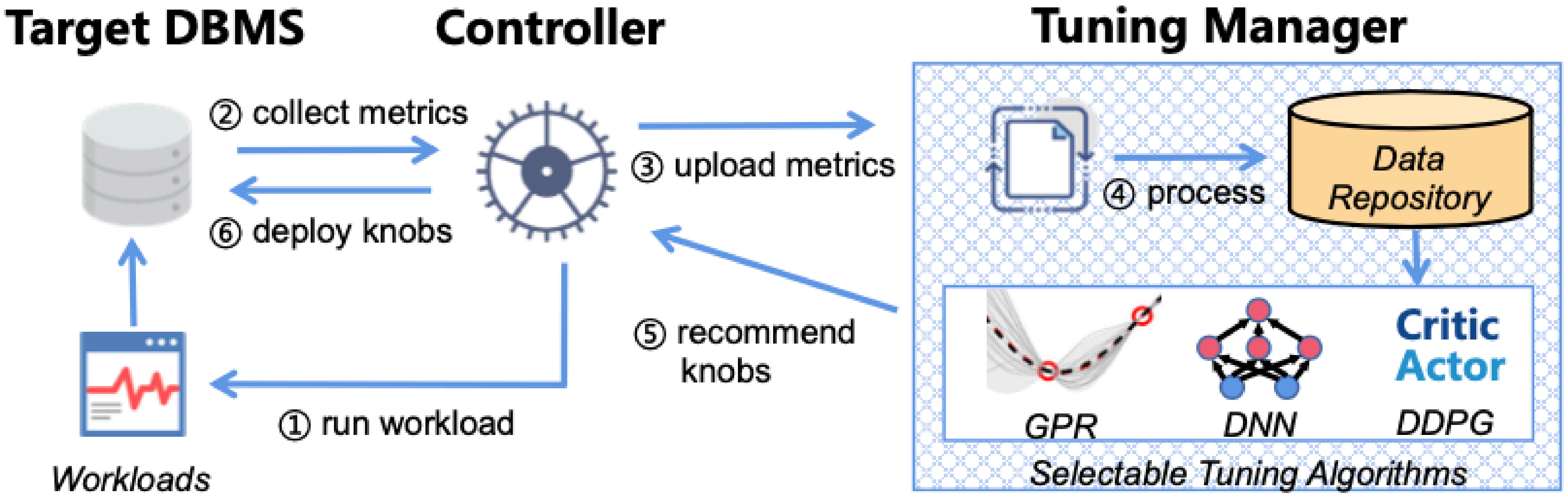
Background & Related Work

- Self-adaptive (Physical Design)
  - Automatic Index Selection
  - Automatic partitioning
- Self-tuning (Autotune Knob Configuration)
  - Heuristics
    - Only target subset of knobs
    - Static rules does not capture relationship between knobs
    - Example: BestConfig
  - ML
    - Ability to consider more knobs
    - Able to handle dependencies between knobs
    - Example: iTuned (BO), CDBTune (RL), iBTune (DNN)

Motivation

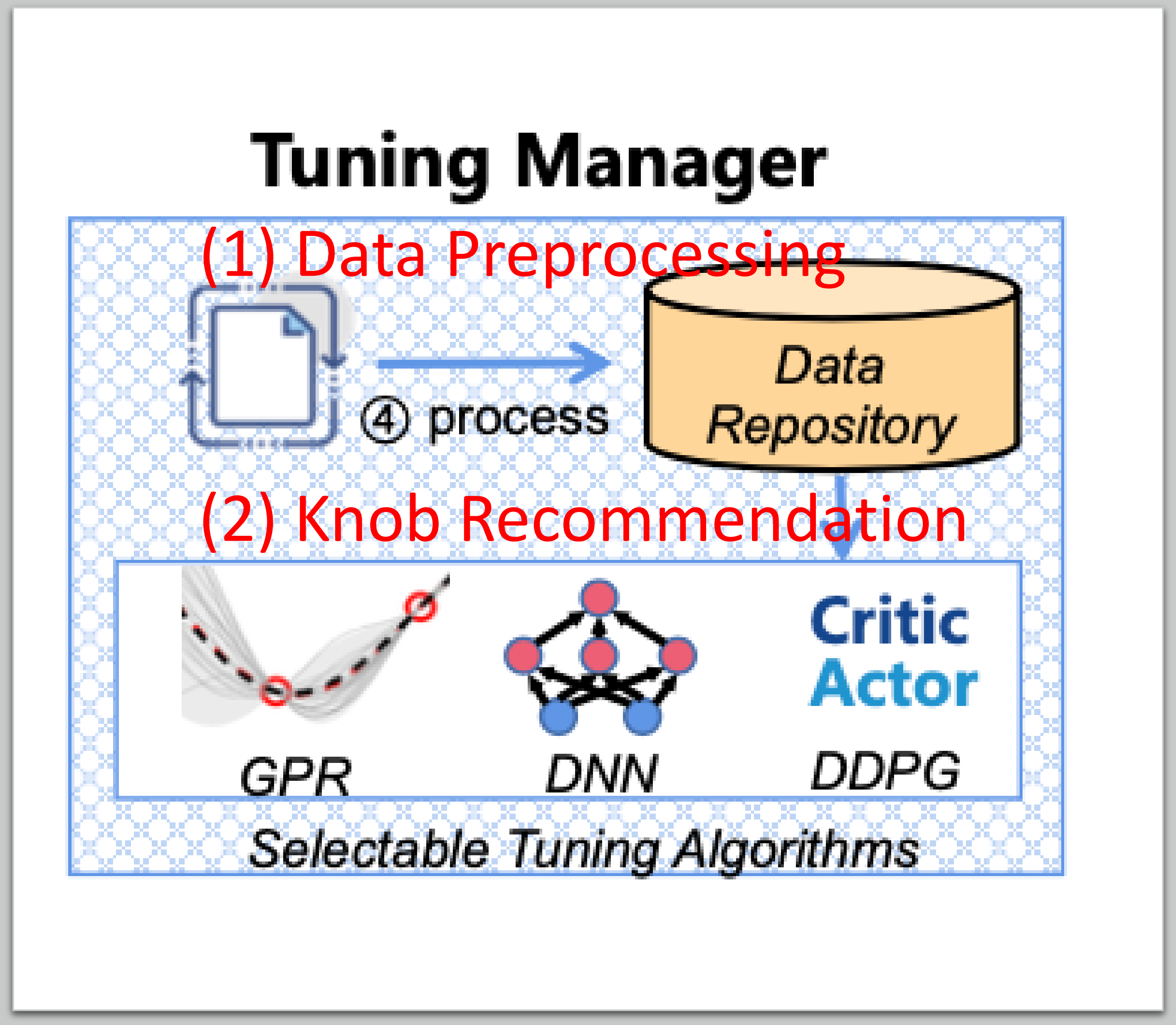
- Previous ML-based tuning studies did not consider Real-world
  - Workload Complexity
  - System Complexity
  - Operating Environment
- This paper
  - Tries to model real-world complexity
  - Focus on enterprise Oracle DBMS (v12) instance
  - Use a real-world workload in a production environment
  - Use virtualised computing infrastructure with non-local storage

Ottertune – ML-based DB tuner



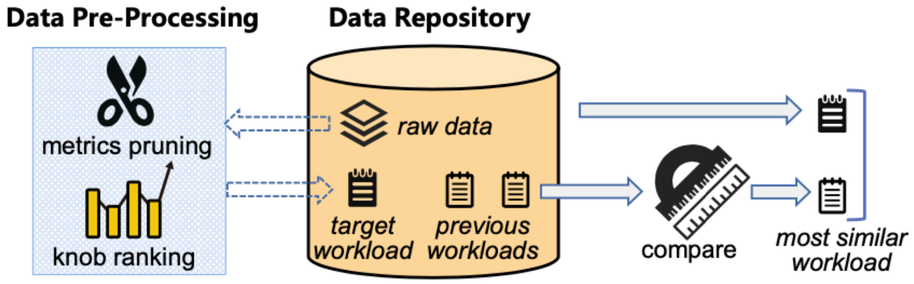
Ottertune ML Algorithms

- Gaussian Process Regression (GPR)
- Deep Neural Networks (DNN)
- Deep Deterministic Policy Gradient (DDPG)



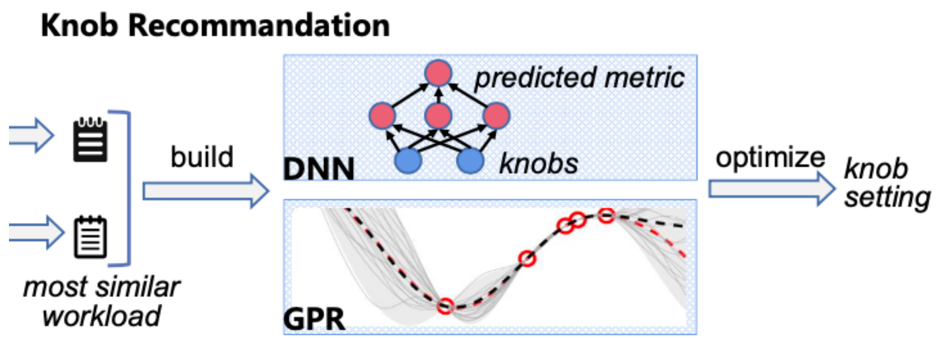
GPR and DNN Data Preprocessing

- Metric Pruning
  - Factor Analysis
  - K-means Clustering
- Knob Ranking
  - Lasso Regression
    - Y = w1x1 + w2x2 + ….
- Workload Mapping
  - Workload Characterisation (Metrics)
  - Euclidean Distance

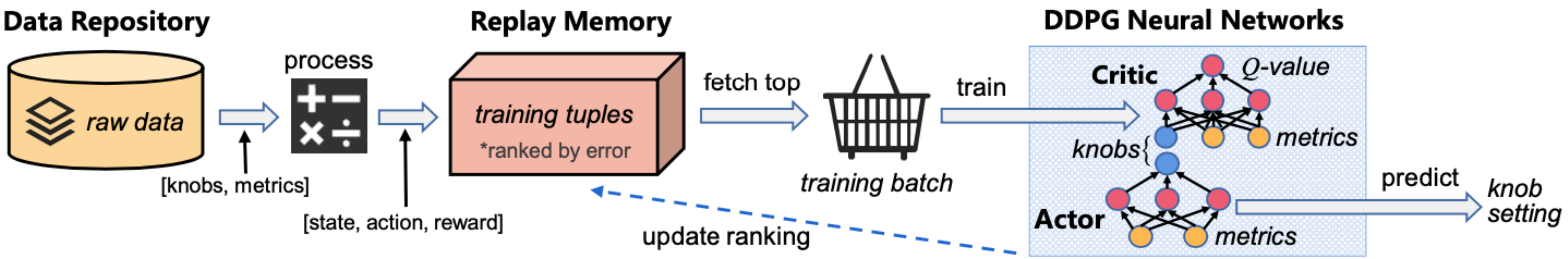


GPR and DNN Knob Recommendation

- GPR
  - Input: Array of knobs
  - Output: Target Metrics and Uncertainty Value
  - Acquisition Func: Upper Confidence Bound
  - Cons: Do not perform well on high dimension
- DNN
  - Input: Knobs
  - Output: Predicted Metrics
  - Structure: Two hidden layers with 64 neurons each + Dropout Regularisation

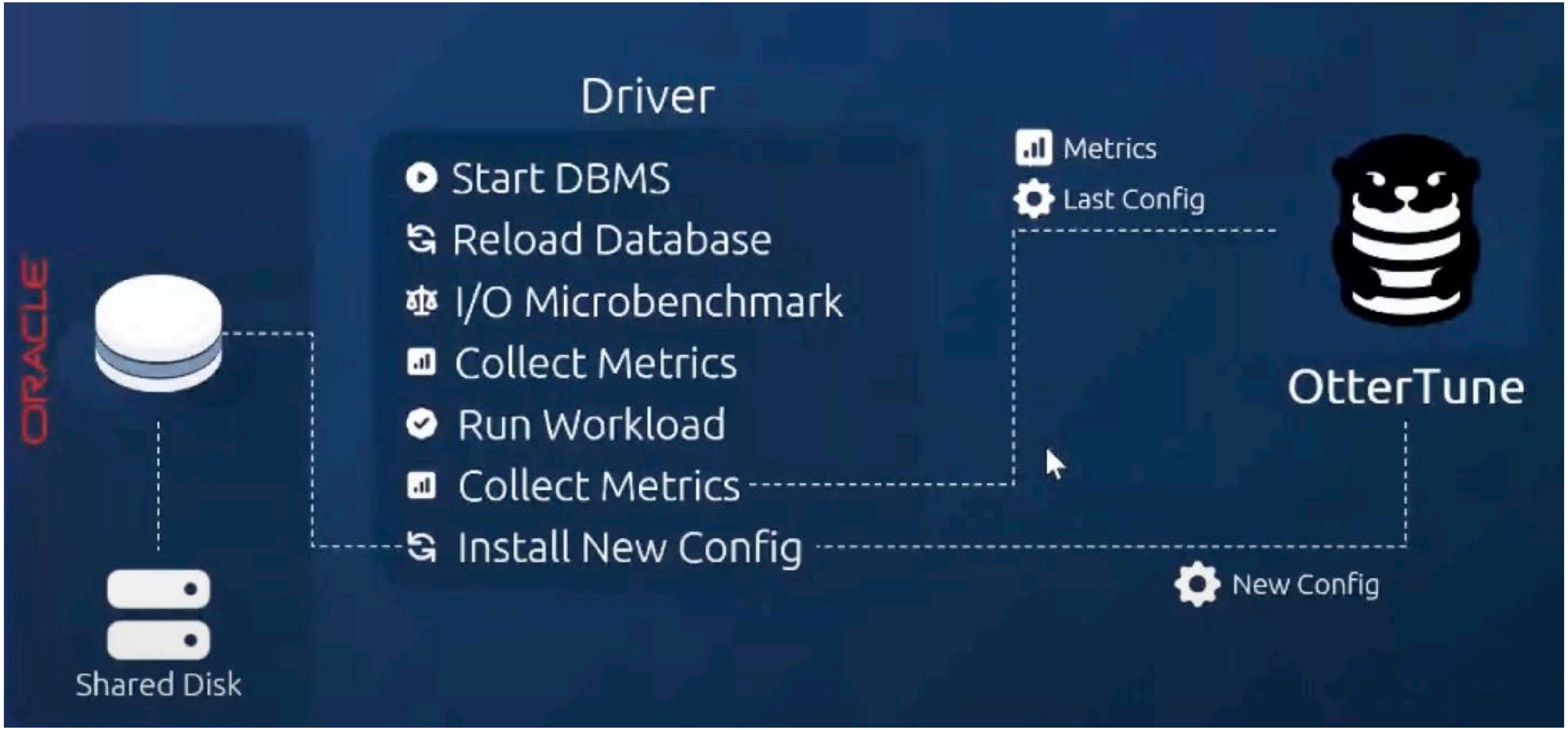


DDPG



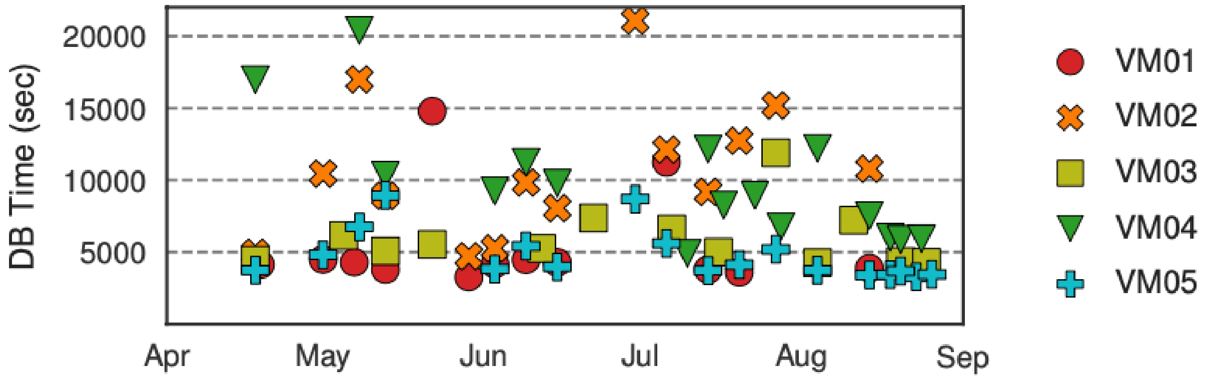
- Actor
  - Input: State (Metrics)
  - Output: Action (Which value to use for a knob)
  - Decide how to set a knob
- Critic
  - Input: Action, State
  - Output: Q-value
  - Provide feedback on the choice of knob
- Replay Memory
  - Store training tuples in ranked order
  - Ranked by the error of predicted Q-value

Ottertune – Field Study

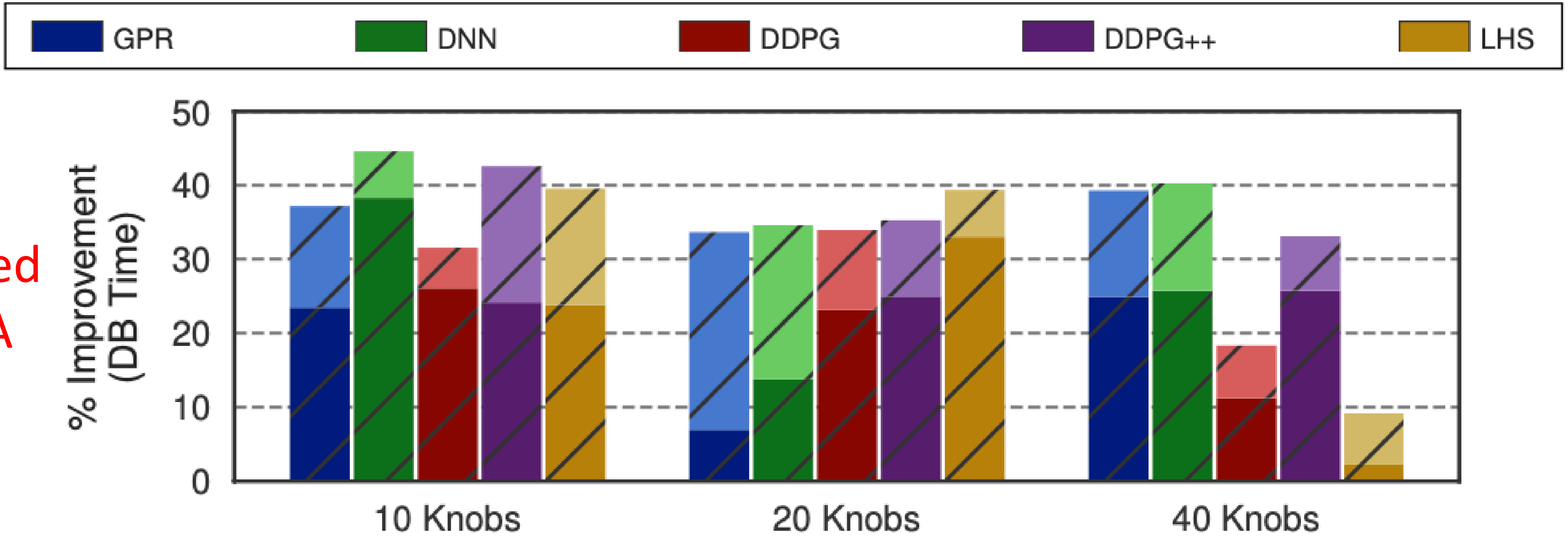


Evaluation – Performance Variability

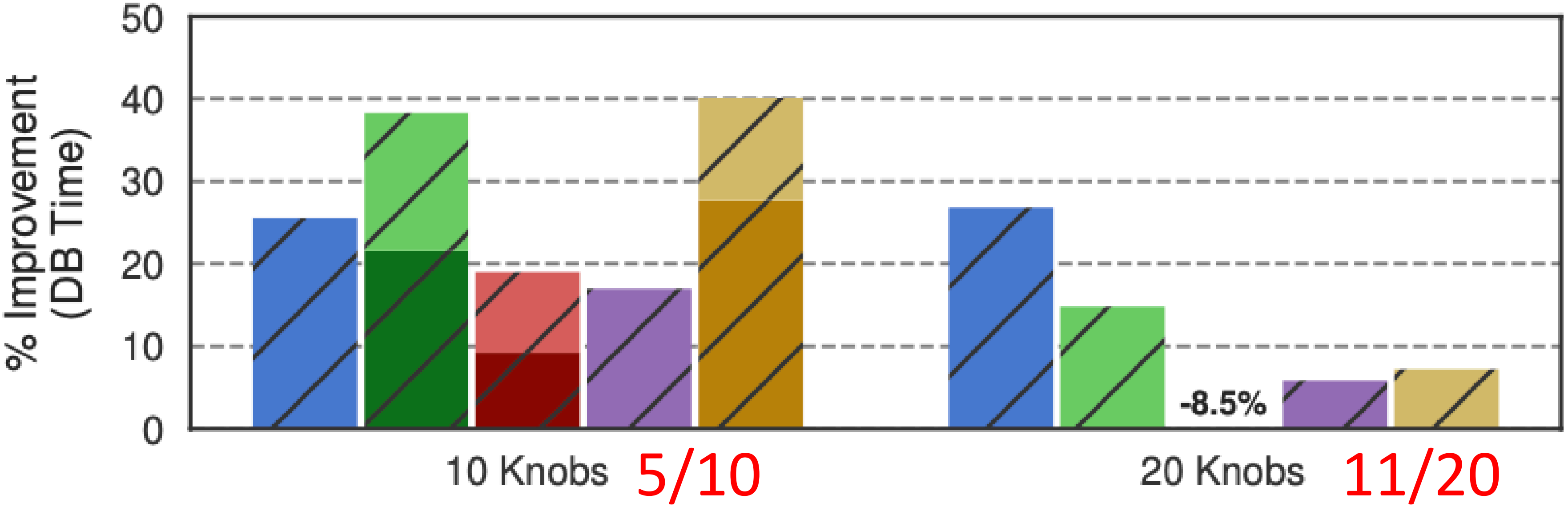
- Problem
  - Latency in shared-disk -> Inconsistent results
  - Performance on same VM can fluctuate
  - Cannot reliability compare tuning sessions
- Solution
  - Three tuning sessions per algorithm
  - Run optimal configurations consecutively, 3 times, on 3 different VMs



Knobs Selected by DBA



Knobs Selected by Ottertune



Minor Criticism

- No Comparison to other ML-based tuner
- Each tuning session is extremely time consuming
  - 3 to 5 days to complete
- Missing some minor details on
  - No explanation on how reward is calculated in DDPG
  - How measurement of workload similarity is conducted in GPR and DNN
- Evaluation is heavily affected by latency of non-local storage