## APPENDIX

### 1. Notations

All the notations used in the paper are listed in Table A1.

**Table A1.** Notations

| Variables | Description | Mapping to WATuning |
|---|---|---|
| $w$ | Workload | Workload applied to systems |
| $k$ | Knob | Tunable knobs of systems |
| $s$ | State | Internal metrics of systems |
| $r$ | Reward | The performance of systems |
| $\omega$ | Weight matrix of the state | Initialized to Uniform [0,1] |
| $a$ | Action | Tunable knobs of systems |
| $\pi_A$ | Actor's policy (mapping state $s_t$ to action $a_t$) | deep neural network |
| $\pi'_A$ | Target actor's policy | ditto |
| $\pi_C$ | Critic's policy (mapping $s_t$ and $a_t$ to Q-value) | ditto |
| $\pi'_C$ | Target critic's policy | ditto |
| $\theta$ | Parameters | ditto |
| $\tau$ | Moderator (trade-off Q-value and reward value) | Set to 0.99 |
| $Y$ | Q-value label | - |
| $L$ | Loss function | - |
| $\rho$ | The amount of difference from the optimal performance | Set to 0.05 |
| $\mu$ | The tolerance factor | Set to 0.5 |
| $\delta$ | The lower bounds of the tolerance interval for throughput | Set to 30000 $kps$ |
| $\varepsilon$ | The upper bounds of the tolerance interval for latency | Set to 6000 $\mu s$ |

### 2. More Experiments

#### 2.1 varying $\rho$ and $\mu$.

**Experiment 8**: In Section 4.2.2, we improved the reward in CDBTune to make it more practical in the actual environment and explained the reasons for our improvement. Here we present the corresponding experimental results.

Table A2 represents throughput, and Table A3 represents latency. The experimental data show that the latency is not much different between the different parameter. In our environment, according to throughput, we will use the parameters $\rho$=0.05 and $\mu$=0.5 for the best results. Note that the test results show that ATT-Tune will become difficult to converge when $\rho$=0.5.

It should be noted that when both $\rho$ and $\mu$ are 0 in the table, it is the reward function proposed by CDBTune.

#### 2.2 Supplement to experiment 2 in Section 5.1

**Experiment 9**: In the experiment 2, we conducted experiments under three workloads (model$_0$, model$_5$, and model$_{11}$). based on Table 1, we will list the comparative experiments for all workloads, as shown in Fig.A1-A9.

**Table A2.** System throughput under different parameters

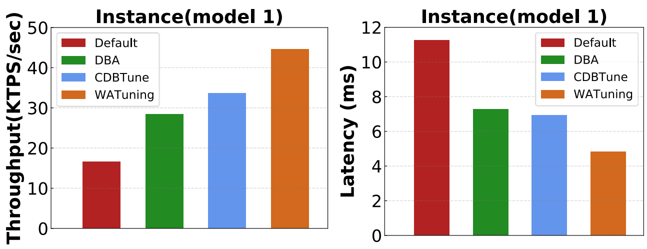
| $\mu$ \ $\rho$ | 0 | 0.01 | 0.05 | 0.1 | 0.5 |
|---|---|---|---|---|---|
| 0 | 46.93 | - | - | - | - |
| 0.2 | - | 45.47 | 47.31 | 44.68 | 36.57 |
| 0.5 | - | 43.61 | 49.85 | 42.01 | NC |
| 0.8 | - | 44.13 | 46.82 | 40.19 | NC |
| 1 | - | 44.29 | 46.34 | 41.22 | NC |

**NOTICE:** NC means not converging. The unit of data in the table is KTPS.

**Table A3.** System latency under different parameters

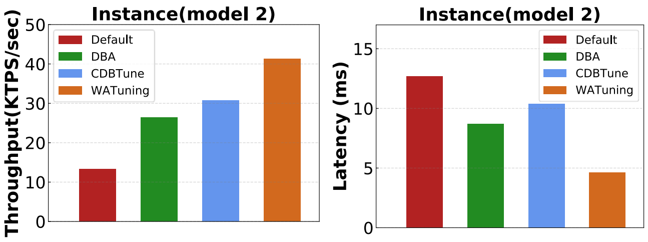
| $\mu$ \ $\rho$ | 0 | 0.01 | 0.05 | 0.1 | 0.5 |
|---|---|---|---|---|---|
| 0 | 5.88 | - | - | - | - |
| 0.2 | - | 4.79 | 3.98 | 4.47 | 6.32 |
| 0.5 | - | 3.14 | 4.05 | 4.98 | NC |
| 0.8 | - | 3.85 | 4.41 | 4.11 | NC |
| 1 | - | 4.23 | 4.38 | 5.12 | NC |

**NOTICE:** NC means not converging. The unit of data in the table is ms.



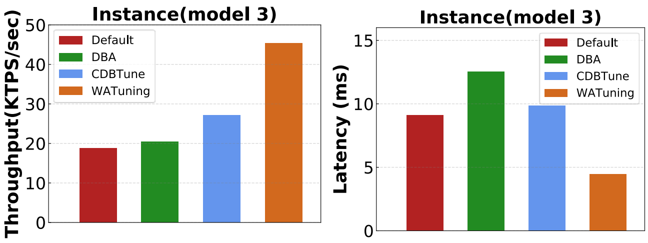

(a)Throughput (0-10% write) (b)Latency (0-10% write)

**Fig.A1.** Performance for 0-10% write operation.



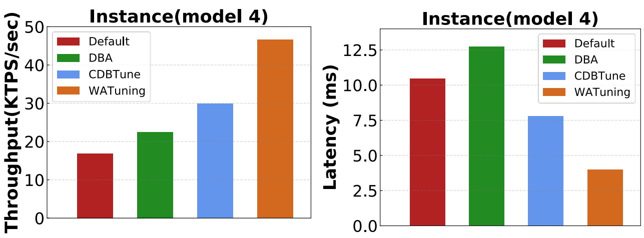

(a)Throughput (10-20% write) (b)Latency (10-20% write)

**Fig. A2.** Performance for 10-20% write operation.



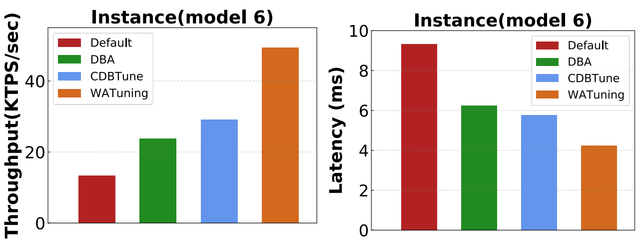

(a)Throughput (20-30% write) (b)Latency (20-30% write)

**Fig. A3.** Performance for 20-30% write operation.



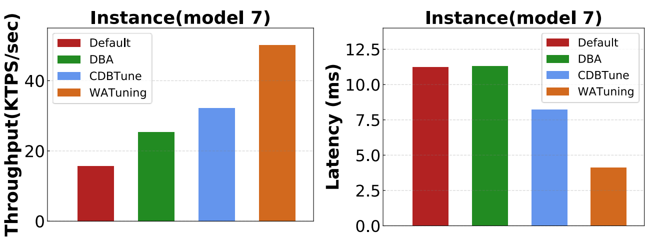

(a)Throughput (30-40% write) (b)Latency (30-40% write)

**Fig. A4.** Performance for 30-40% write operation.



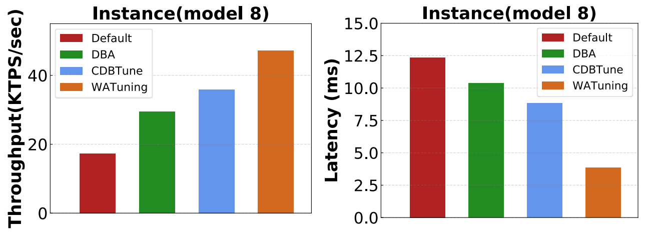

(a)Throughput (50-60% write) (b)Latency (50-60% write)

**Fig. A5.** Performance for 50-60% write operation.



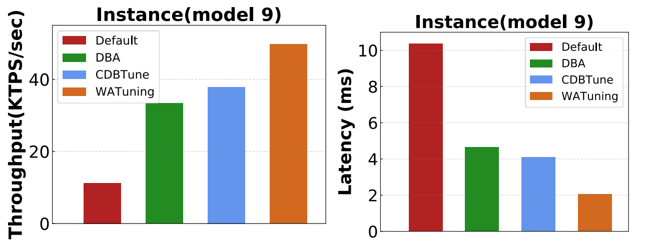

(a)Throughput (60-70% write) (b)Latency (60-70% write)

**Fig. A6.** Performance for 60-70% write operation.



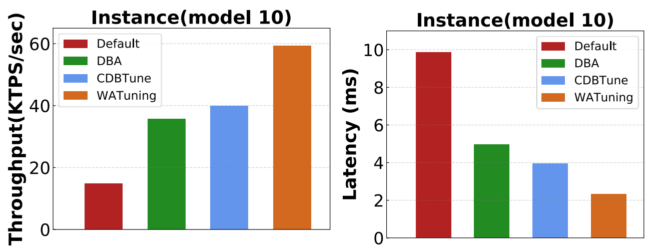

(a)Throughput (70-80% write) (b)Latency (70-80% write)

**Fig. A7.** Performance for 70-80% write operation.




(a)Throughput (80-90% write) (b)Latency (80-90% write)

**Fig. A8.** Performance for 80-90% write operation.



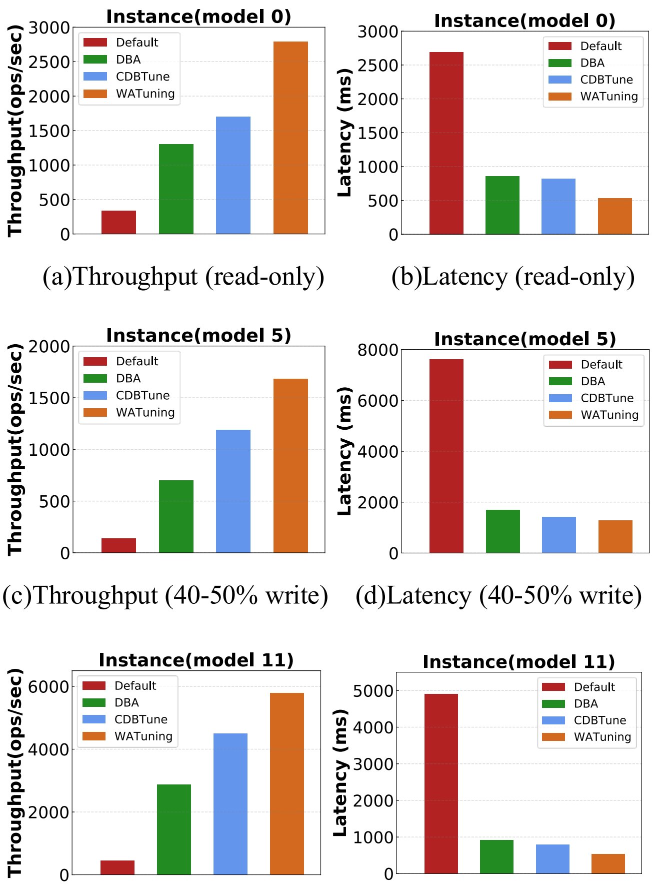

(a)Throughput (90-100%write) (b)Latency(90-100% write)

**Fig. A9.** Performance for 90-100% write operation.

## 2.3 Experiment based on MySQL

**Experiment 10**: In Section 5, target system of WATuning is LevelDB. In this case, we are using MySQL as a service for WATuning, as a complementary experiment. Here, we experimented with MySQL using YCSB as a test tool. For YCSB, we generate 10GB data and operations of 20M. The experimental environment is instance A in Table 2. The experimental results are shown in Fig.A10.

The results shown that WATuning still achieved the best results when MySQL was used as a server and significantly outperformed the most advanced tuning tools and DBA experts. This shows that our model is very extensible in system tuning.




(a)Throughput (read-only) (b)Latency (read-only)

(c)Throughput (40-50% write) (d)Latency (40-50% write)

(e)Throughput (write-only) (f)Latency (write-only)

**Fig. A10.** Comparison of performance for DBA, Default settings and CDBTune based on the workload with different write operation ratio (on MySQL)