Contribution ID: **351**   Type: **Poster**

Parameter tuning of distributed storage system based on reinforcement learning

HEP computing is a typical data intensive computing. Performance of distributed storage system, can largely defines the efficiency of HEP data processing and analysis. There is a large number of parameters that can be adjusted in a distributed storage system. The setting of these parameters has a great influence on the performance. At present, these parameters are either set with static values or automatically tuned by some heuristic rules defined by experienced administrators. Diversity of data access patterns and hardware capabilities, interactions between tuning actions, latency between parameter tuning and performance rewarding, and the large parameter search space determines these methods are incapable to cater the performance tuning requirements of a modern data center.

Reinforcement Learning (RL) is a branch of machine learning concerned with how an agent ought to take actions within an environment in order to maximize a certain reward. In recent years, it has many successful applications in areas of robotics and game play. Similarities between parameter tuning and these tasks encourages our idea of implementing a RL based automatic performance tuning method. Therefore, we evaluated this idea by the case of performance tuning of Lustre file system client. We used 17 different performance metrics as "state"and 8 different parameters as "tuning target", IO throughput increase as "reward". In each tuning period, a tuning agent will input the "state"to a deep neural network, and use its inference result as instructions for performance tuning, make the tuning actions on "turning target", and then read the throughput metric as "reward". After that, the "State->Tuning Action->Reward"sequence will be stored in a training database as training sample for online training of the neural network. By repeating these steps, the neural network is gradually empowered with the intelligence to make right tuning decision given an arbitrary new state input. The whole process is unsupervised, while the model can learn to adapt to new working load from its unsuccessful tuning actions.

We implemented three reinforcement learning algorithms: DQN, A2C and PPO with PyTorch (version 0.4) in this case. Experiments show that, in a small test bed with Iozone (version 3.479) workload, this method can increase the throughput by about 30% comparing to static default settings of Lustre (version 2.5.3) as the baseline. In the future, it possible to apply this method to other parameter tuning use cases in the operations of data centers.

Authors: WANG, Lu (Computing Center,Institute of High Energy Physics, CAS); Mr ZHANG, Wentao (Institute of High Energy Physics, Chinese Academy of Science); CHENG, Yaodong (IHEP)

Presenter: WANG, Lu (Computing Center,Institute of High Energy Physics, CAS)

Session Classification: Poster Session

Track Classification: Track 2: Data Analysis - Algorithms and Tools