Clustering Based Job Runtime Prediction for Backfilling Using Classification

Hang Cui
Graduate School of Information Sciences, Tohoku University
Sendai, Miyagi, Japan

Hiroyuki Takizawa
Keichi Takahashi
Yoichi Shimomura
Cyberscience Center, Tohoku University
Sendai, Miyagi, Japan

1 INTRODUCTION

How to effectively schedule massive number of jobs submitted to a High Performance System (HPC) system is essential. A good job scheduler can provide both high system resource utilization and user satisfaction.

An underestimation of job runtime causes the job scheduler to terminate the job before it completes. In order to avoid this circumstance, users of HPC systems are likely to overestimate their job runtimes compared to the real runtime[1]. This causes some jobs not being able to be backfilled, even if there are actually enough amounts of system resources available for those jobs.

2 METHODOLOGY

As Figure 1 shows, the previous approaches did not differentiate underestimation and overestimation. However, runtime prediction that can avoid underestimation is important for increasing the system utilization of HPC systems [2].

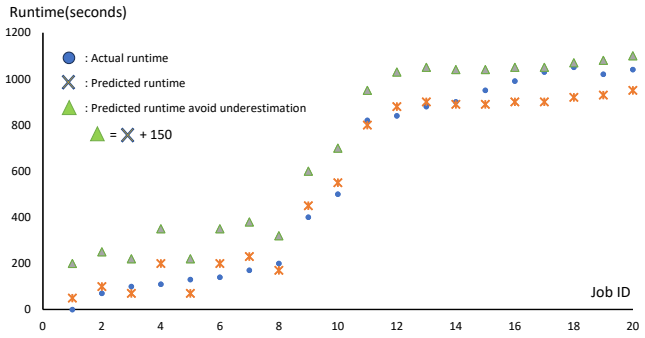

Figure 1: Actual and predicted runtime in previous approaches

As Figure 2 shows, in the proposed approach, the runtime prediction is translated into classification of jobs into categories. The mean and standard deviation of runtimes within each category are calculated in advance, and the predicted runtime of a job is the mean runtime plus two-sigma, means two standard deviations.

3 EVALUATION

After performing the prediction of the runtime, it is passed to the CQSim[1] simulator. The CQSim simulator is modified to use the predicted runtime for backfilling to generate the scheduling result.

[1]https://github.com/SPEAR-UIC/CQSim

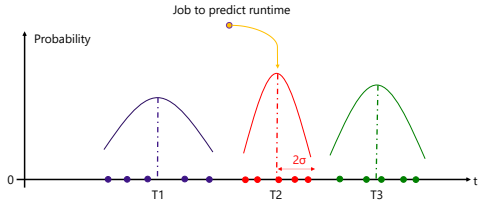

Figure 2: Runtime prediction of the proposed approach

The datasets used in the ecaluation are ANL and SDSC job trace from Parallel Workloads Archive (PWA)[2].

The number of the backfilled jobs when scheduling the jobs in the HPC system are counted using different approaches to predict the runtime of these jobs. The comparison of these numbers are shown in Figure 3. As the Figure shows, more jobs are backfilled when the proposed approach is used to predict the runtime.

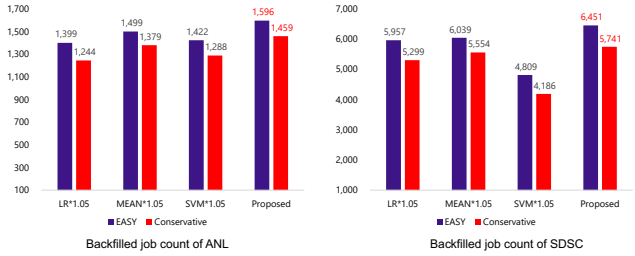

Figure 3: Number of successfully backfilled jobs of ANL and SDSC

4 FUTURE WORK

More evaluation can be carried out using some different job traces and runtime predicting approaches. Also, some hyperparameters are not deeply discussed by far, such as the number of categories and so on.

REFERENCES

[1] Zhengxiong Hou et al., Optimizing job scheduling by using borad learning to predict execution time on HPC clusters, CCF Transactions on High Performance Computing, 23, February, 2023
[2] Mohammed Tanash et al., "Ensemble Prediction of Job Resources to Improve System Performance for Slurm-Based HPC Systems", PEARC '21: Practice and Experience in Advanced Research Computing, July 2021, Article No.: 21, Pages 1-8

[2]https://www.cs.huji.ac.il/labs/parallel/workload/index.html