High Performance Computing Scheduler Parameter Optimization using Simulation and Regression

Scott Hutchison (scotthutch@ksu.edu), Daniel Andresen (advisor) (dan@ksu.edu), William Hsu (advisor) (bhsu@ksu.edu)
Department of Computer Science, Kansas State University

Research Question

- Is our HPC scheduler optimally configured?
- Are the default SLURM scheduler settings good/optimal for our HPC hardware with the jobs our users typically run?

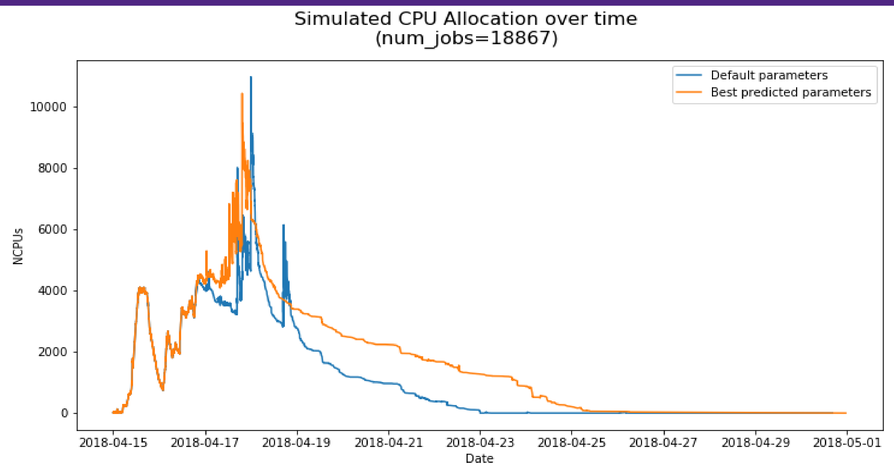
Background

- Test data: ~18,500 jobs from HPC log data from Sept. 2018
- Validation data: ~18,800 jobs from HPC log data from Apr ’18
- SLURM simulator developed by SUNY University at Buffalo [2]
- Simulations were run in parallel using KSU HPC resources
- PySpark [3] was used for data wrangling, statistical analysis, and regression tasks

Methodology

- Identify 11 scheduler parameters of interest
- Generate scheduler parameter values to test
- Conduct 80,000 scheduler simulations
- Perform Gradient Boosted Tree Regression [1] to predict the average job queue time for parameter settings
- Generate ~12.4 billion combinations of parameter values
- Predict avg. queue time for each using GBTR model
- Simulate 10,000 parameters with best predicted avg. queue time
- Compare to default scheduler parameters

Simulated CPU Allocation over Time



Best recommended scheduler parameters better utilize HPC resources than the default parameters and finish scheduling all jobs earlier

SLURM Parameters Investigated

Scheduler Parameter	Test Values	Prediction Ranges	Recommendation
bf_continue	True, **False**	True, False	True. **Opposite of the default**
bf_interval	10, **30**, 3000	1-10800 by 1000’s	Lower is better. **Default too low**
bf_max_time	10, **30**, 300	1-3600 by 500’s	Lower is better. Default ok.
bf_resolution	20, **60**, 360	1-3600 by 500’s	Lower is better. Default ok.
bf_running_job_reserve	True, **False**	True, False	False. Default ok.
bf_yield_interval	0.5, **2**, 9	1-10 by 3’s	Lower is better. Default is good
default_queue_depth	30, **100**, 3000	1-17000 by 3000’s	Lower is better. Default is good
sched_interval	20, **60**, 180	20-2000 by 400’s	Inconclusive
sched_min_interval	0, **2**, 6	2-200 by 20’s	Inconclusive
bf_max_job_test	100, **500**, 5000	1-1000000 by 100000’s	Inconclusive
bf_yield_sleep	2, **5**, 45	1-10 by 3’s	Inconclusive

See https://slurm.schedmd.com/slurm.conf.html for detailed parameter description

Results and Discussion

- **Avg job queue time**: **Best predicted settings: 2,719s Default: 12,970s**
- **Best predicted parameter settings decreased job avg. queue time by *79%***
- **Predicted parameter settings decreased avg. job queue time 51.8%** on average
- In all cases, predicted best parameter settings decreased avg. queue time.
- Required over 840,000 HPC core hours for simulation
- RMSE on training data was ~250 sec.

Conclusion

- Optimizing scheduler parameters for a particular configuration can significantly improve system performance.
- Technique shows promise and can be used by HPC administrators to optimize scheduler settings for a particular HPC system for jobs its users frequently run

References

[1] J. H. Friedman, “Greedy function approximation: a gradient boosting machine,” Annals of statistics, pp. 1189–1232, 2001.
[2] N. A. Simakov, M. D. Innus, M. D. Jones, R. L. DeLeon, J. P. White, S. M. Gallo, A. K. Patra, and T. R. Furlani, “A slurm simulator: Implementation and parametric analysis,” in International Workshop on Performance Modeling, Benchmarking and Simulation of High-Performance Computer Systems, pp. 197–217, Springer, 2017.
[3] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen, et al., “Mllib: Machine learning in apache spark,” The Journal of Machine Learning Research, vol. 17, no. 1, pp. 1235–1241, 2016.