

Poster: Job Simulation for large-scale PBS based Clusters with the Maui Scheduler

Georg Zitzlsberger
IT4Innovations
VSB – Technical University of
Ostrava
Czech Republic
georg.zitzlsberger@vsb.cz

Branislav Janský
IT4Innovations
VSB – Technical University of
Ostrava
Czech Republic
branislav.janský@vsb.cz

Jan Martinovič
IT4Innovations
VSB – Technical University of
Ostrava
Czech Republic
jan.martinovic@vsb.cz

ABSTRACT

For large-scale High Performance Computing centers with a wide range of different projects and heterogeneous infrastructures, efficiency is an important consideration. Understanding how compute jobs are scheduled is necessary for improving the job scheduling strategies in order to optimize cluster utilization and job wait times. This increases the importance of a reliable simulation capability, which in turn requires accuracy and comparability with historic workloads from the cluster. Not all job schedulers have a simulation capability, including the Portable Batch System (PBS) resource manager. Hence, PBS based centers have no direct way to simulate changes and optimizations before they are applied to the production system. We propose and discuss how to run job simulations for large-scale PBS based clusters with the Maui Scheduler. For validation purposes, we use historic workloads collected at the IT4Innovations supercomputing center, and demonstrate the viability of our approach.

CCS CONCEPTS

• Computing methodologies • Simulation evaluation

KEYWORDS

High Performance Computing (HPC), Simulation, PBS, Maui Scheduler, Job Scheduling

1 Introduction

For large-scale High Performance Computing (HPC) centers with a wide range of different projects and heterogeneous infrastructure, efficiency is an important consideration. Depending on the computing center and its objectives, efficiency comprises the optimization of the utilization of the available systems, or minimization of the wait times for users' jobs.

HPC centers continually strive to improve efficiency. Unconfirmed changes and experiments involving live production systems are not justifiable due to an increased risk of undesirable performance effects, such as job congestion or starvation of jobs, unexpected job scheduling behavior for users, etc., or simply because of a lack of comparability as usage patterns of users

change with projects. This problem has already been addressed by HPC job schedulers and resource managers by adding simulation capabilities (e.g. Slurm Simulator [1], Maui Scheduler [2], LSF [3]). However, not all solutions used today offer a simulator, which is also the case for PBS Professional [4]. It hence does not allow the analysis of changes to the job scheduler configuration or effects of node reservations before deployment to the live system.

There exists a large range of simulation frameworks for HPC and grid systems. For research purposes, simulators like SimMatrix [5], GridSim [6], SimGrid [7], and CQSim [8] are widely used to study the effects of large-scale systems. Whilst SimMatrix demonstrated better resource usage than GridSim and SimGrid, and is capable of simulating Exascale level systems [9], it was only validated against real systems of up to 4096 cores with artificial tasks. Another simulator is Alea [10], which is based on the GridSim toolkit. It is validated with real workloads [11] for approximately 5,200 CPU cores, containing 102,657 jobs over a span of 4 months. CQSim enables plan based and energy aware simulations, and is validated with systems up to 164k CPU cores and 69k jobs. However, all of the above are only toolboxes, and require a manual implementation of the job scheduling algorithms to approximate an existing system. They are also not a suitable in situ replacement for the scheduling algorithms of HPC job management systems.

The Slurm Simulator is an extension of the HPC scheduler Slurm, originally developed by A. Lucero [12] and improved further by S. Trofinoff and M. Benini [1]. As such, Slurm based systems have the advantage of making use of the simulation capabilities directly. The Slurm Simulator has been validated for large-scale simulations of real workloads from TACC [13] with 6,400 nodes over 12 weeks. For LSF based systems, simulations are so far only capable of predicting the start times of jobs. However, further plans to develop simulation capabilities have been announced [14]. The Cobalt Scheduler, developed by Argonne National Laboratory, comes with a simulator called Qsim [15]. It only is applicable within an existing Cobalt environment.

The Maui Scheduler (MS in the following) has existed since the 1990s, and even though there is no active development anymore it is open source, mature and highly configurable. Its simulation

capability has already been used to analyze HPC workloads [16]. Optionally, it can also be applied as a meta-scheduler to replace the job scheduling components of HPC resource managers like PBS [17] and TORQUE [18]. This allows the simulation and the job management system to use the same scheduling algorithms and configurations.

We use the MS to complement PBS with the analysis of historic workloads of large-scale clusters. To show the viability of the MS for larger systems, we describe the configuration and changes to approximate the behavior of the PBS job scheduler for a non-trivial 2 PetaFLOP/s system. We furthermore discuss the results and problems we faced whilst using real workload data of that system.

2 Poster Structure

The poster summarizes our work on approximating an existing, large-scale, non-trivial PBS based job-scheduling configuration for simulation with the MS. We investigated whether simulations with the MS are possible for large-scale (>1k nodes) and PBS based systems. The investigation includes the analysis of accuracies of the simulations compared to real workloads.

For the poster, we highlight our simulation setup and results¹. We describe the large-scale PBS environment (i.e. IT4Innovations' Salomon cluster) for which we simulated the job scheduling. Its configuration is non-trivial due to supporting different queues with individual limitations, a job priority formula that considers different queue priorities, project based fairshare with conservative backfilling, and job wait times to avoid job starvation.

Furthermore, we describe the changes applied to the MS in order to support clusters of the size of Salomon with over 24k cores, 1008 nodes and over 60k jobs in the queues at peak times. Our simulation also considers node downtimes, both unexpected (i.e. node failures) and system wide (i.e. planned maintenance). Whilst the former can be directly expressed with MS capabilities, the latter required an extension of the simulation control flow, which we describe as well.

Results show two simulated weeks, week 6 and 7, taken from 50 simulation windows in 2017. Simulations are carried out for every week independently, which requires its preceding week for warm-up of the simulator. The warm-up weeks are shown for completeness but are not the subject of our measurements. Weeks 6 and 7 have been selected due to prominent properties. Week 6 has one of the highest amounts of job submissions of the 50 weeks we analyzed with a high utilization of the job backlog (queues). It also contains larger unexpected node downtimes. On the other hand, during week 7, the cluster underwent system wide maintenance, which required the cluster to shut down. The change

of the cluster utilization before the maintenance is clearly visible, as well as its behavior when going back online. Because of the interruption, the amount of jobs submitted during that week are one of the lowest of the 50 week window.

Both weeks are characterized by real and simulated cluster utilization in terms of the number of active nodes and queued jobs waiting for execution. We also show the qualitative differences of wait times of the jobs for the simulation against the real system. For quantitative analysis, we use Cumulative Density Functions (CDFs) to characterize the simulated job wait times back to back with the real system for different simulation time steps. More details on deviation (i.e. errors) of the wait times are shown with standard deviation, mean, and median for time step resolutions of 60, 30, 10, 5 and 1 s against the real wait times.

The results also show the limitations of the simulation due to missing job dependency information including the lack of a corresponding simulation method, and the effects of terminating entire jobs with failed nodes. The latter is caused by the assumption of the simulation that jobs are terminated if any of the nodes fail, which is not always the case in the real workloads. The former is caused by both missing historic data (i.e. PBS does not store job dependency information in the SQL database) and the lack of MS to simulate those. However, both limitations only have a minor impact on the overall error.

3 Collateral Material

With the poster, we make available collateral on our GitHub repository [19], such as the *Dockerfile* and environment of our setup, configuration files for the simulations, interactive plots of the diagrams from the poster, and a link to the original paper.

The *Dockerfile* is used to build a Docker image that contains the entire MS simulation environment with the changes we applied as a patch file. The configuration files for simulating the Salmon cluster's PBS scheduler with our approximated setup are also available. For privacy reasons, we only provide an example workload and control script to emulate the node downtimes during the example workload period. However, our full results as shown in the paper are available as interactive plots via a Jupyter notebook. They contain tens of thousands of data points each, which allows for a more detailed insight into the results. Our GitHub repository describes how to reproduce our setup in detail.

ACKNOWLEDGEMENT

This work was supported by The Ministry of Education, Youth and Sports from the Large Infrastructures for Research, Experimental Development and Innovations project "IT4Innovations National Supercomputing Center – LM2015070", by the ERDF in the IT4Innovations national supercomputing center – path to exascale project (CZ.02.1.01/0.0/0.0/16_013/0001791) within the OPRDE, and ANTAREX, a project supported by the EU H2020 FET-HPC program under grant agreement No. 671623.

¹ For generality, our original paper mainly describes the mapping of a PBS configuration with a job formula as an additive term to the MS; we find the simulation setup and results more appropriate for the poster, and refer the reader to our paper for more information.

REFERENCES

- [1] Trofinoff, S., and Benini, M., 2015, "Using and Modifying the BSC Slurm Workload Simulator, Slurm User Group Meeting, September 2015," https://github.com/benini/slurm_simulator/. [accessed 31-07-2018]
- [2] Adaptive Computing, 2014, "Maui Scheduler, Release 3.3.1," <https://github.com/LabAdvComp/maui/>. [accessed 31-07-2018]
- [3] IBM, 2016, "IBM Spectrum LSF, Version 10 Release 1.0, Configuration Reference."
- [4] Altair, 2018, "PBS Works Suite, PBS Professional," <https://pbsworks.com/>. [accessed 31-07-2018]
- [5] Wang, K., 2014, "SimMatrix," <https://github.com/kwangit/SimMatrix/>. [accessed 31-07-2018]
- [6] Buyya, R., and Murshed M., 2002, "Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing," *Concurrency and Computation: Practice and Experience (CCPE)*, vol. 14, no. 13, pp. 1175–1220.
- [7] Casanova, H., Giersch, A., Legrand, A., Quinson, M., and Suter, F., June 2014, "Versatile, scalable, and accurate simulation of distributed applications and platforms," *Journal of Parallel and Distributed Computing*, vol. 74, pp. 2899–2917.
- [8] X. Yang, Z. Zhou, S. Wallace, Z. Lan, W. Tang, S. Coghlan, and M. Papka, "Integrating Dynamic Pricing of Electricity into Energy Aware Scheduling for HPC Systems", *Proc. of SC'13*, 2013.
- [9] Wang, K., Brandstatter, K., and Raicu, I., 2013, "Simmatrix: Simulator for many-task computing execution fabric at exascale," in *Proceedings of the High Performance Computing Symposium, HPC '13*, (San Diego, CA, USA), pp. 9:1–9:9, Society for Computer Simulation International.
- [10] Klusáček, D., and Rudová, H., 2010, "Alea 2 – Job Scheduling Simulator," in *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques (SIMUTools 2010)*, ICST.
- [11] Klusáček, D., and Chlumský, V., 2016, "Planning and metaheuristic optimization in production job scheduler," in *Job Scheduling Strategies for Parallel Processing* (N. D. Walfredo Cirne, ed.), pp. 1–19, Neuveden.
- [12] Lucero, A., 2011, "Slurm Simulator, Slurm User Group Meeting, September 2011."
- [13] Simakov, N. A., Innus, M. D., Jones, M. D., DeLeon, R. L., White, J. P., Gallo, S. M., Patra, A. K., and Furlani, T. R., 2017, "A Slurm simulator: Implementation and parametric analysis," in *PMBS@SC*, vol. 10724 of *Lecture Notes in Computer Science*, pp. 197–217, Springer.
- [14] Ding, Z. H., and Li, X. Q., 2017, "IBM Spectrum LSF 10 — Taking the guesswork out." <https://developer.ibm.com/storage/2016/10/19/ibm-spectrum-lsf-10-taking-the-guesswork-out/>. [accessed 31-07-2018]
- [15] Argonne National Laboratory, 2018, "Qsim," <https://trac.mcs.anl.gov/projects/cobalt/wiki/qsim>
- [16] Jackson, D. B., Jackson, H. L., and Snell, Q. O., 2000, "Simulation based hpc workload analysis," in *Parallel and Distributed Processing Symposium, Proceedings 15th International*, IEEE.
- [17] Bode, B. Halstead, D. M., Kendall, R., Lei, Z., and Jackson, D., 2000, "The portable batch scheduler and the maui scheduler on linux clusters," in *Proceedings of the 4th Annual Linux Showcase & Conference - Volume 4, ALS'00*, (Berkeley, CA, USA), pp. 27–27, USENIX Association.
- [18] Advanced Computing, 2018, "Torque Resource Manager," <http://www.adaptivecomputing.com/products/open-source/torque/>. [accessed 31-07-2018]
- [19] Zitzlsberger, G., 2018, "Maui simulator configuration and environment for IT4Innovations' Salomon cluster," <https://github.com/It4innovations/Maui-Simulation>. [accessed 31-07-2018]