Experiments with Job Scheduling in MetaCentrum

Dalibor Klusáček, Hana Rudová, and Miroslava Plachá

Faculty of Informatics, Masaryk University
Botanická 68a, 602 00 Brno
Czech Republic
{xklusac,hanka@fi.muni.cz

1 Introduction

Large computing clusters and Grids have become common and widely used platforms for the scientific and the commercial community. Efficient job scheduling in these large, dynamic and heterogeneous systems is often a very difficult task [14]. Therefore, a lot of testing and evaluation is needed before some scheduling algorithm is applied in the production system such as PBSpro [6] or LSF [15]. Due to several reasons, such as the cost of resources, reliability, varying background load or the dynamic behavior of the components, experimental evaluation is not usually performed in the real systems. Many simulations with various setups that simulate different real-life scenarios must be performed using the same and controllable conditions to obtain reliable results. This is hardly achievable in the production environment.

2 Available Data Sets

When performing simulations and testing, usually the workload traces from the Parallel Workloads Archive (PWA) [3] or Grid Workloads Archive (GWA) [1] are used to represent users' jobs. However, these data do not contain several parameters that are important for realistic simulations. Typically, very limited information is available about the Grid or cluster resources. The number of machines in particular clusters, their architecture, the CPU speed, the RAM size or the resource specific policies are not usually known. However, these parameters often significantly influence the decisions and performance of the scheduler [10]. Moreover, no information concerning background load, resource failures, or specific users' requirements are available. In heterogeneous environments, users often specify some subset of machines or clusters that can process their jobs. This subset is usually defined either by the resource owners' policy (user is allowed to use such cluster), or by the user who requests some properties (cluster location, library, software license, execution time limit, etc.) offered by some clusters or machines only. Also, the combination of both owners' and users' restrictions is possible. When one tries to create a new scheduling algorithm all such information and constraints are crucial, since they make the algorithm design much more complex. If omitted, resulting simulation may provide misleading or unrealistic results as we show in Section 4.

3 MetaCentrum Data Set

Since the current archives miss to provide truly complete data sets, we were very happy that we were kindly allowed by the MetaCentrum team to create the data set covering many previously mentioned issues. Namely, this data set contains trace of 103,620 jobs executed during the first five months of 2009 as well as detailed description of computational nodes. Job description is very complex, including e.g., the maximum runtime limits for jobs or their specific requirements concerning target platform (CPU architecture, location, network interface, etc.). Also the description of clusters contains detailed information involving CPU speed, RAM size, CPU architecture, operating system and the list of supported properties (allowed queue(s), cluster location, network interface, etc.). Together, these detailed information about jobs and machines allow to use so called *specific job requirements* representing the "job-to-machine" suitability. Moreover, information about machines that were not available has been collected, covering the time periods when machines were either in maintenance (failure/restart) or dedicated for specific purposes. Finally, the list of queues including their time limits and priorities is provided. The MetaCentrum data set is publicly available at `http://www.fi.muni.cz/~xklusac/workload`. Certainly all information in the data set containing private information such as user, machine, queue or job names or names of specific parameters were anonymized.

4 Evaluation

Once the complex data set from the MetaCentrum was available, we could answer the question whether the additional information and constraints such as machine failures or specific job requirements influence the quality of the solution generated by the scheduling algorithms. For this purpose, two different problems have been considered and then simulated using the Alea job scheduling simulator [8]. The first BASIC problem involved the use of MetaCentrum data set where both machine failures and specific job requirements were ignored. This setup is quite similar to the typical amount of information available in the GWA or PWA archives that do not provide information about machine failures or specific job requirements. The second EXTENDED problem used all information available in the MetaCentrum data set, therefore both machine failures and specific job requirements have been used during the simulation. Five different scheduling algorithms have been used in this evaluation. The algorithms FCFS, EASY backfilling (EASY) [12], and conservative backfilling (CONS) [4, 13] represent standard queue-based algorithms. The other two algorithms were developed as a part of our work. The CP algorithm is based on the ideas of constraint programming [11] and the local search (LS) [9] is an optimization procedure using conservative backfilling for construction of the initial solution. The average slowdown [5] and average wait time [2] have been used as the evaluation criteria here (additional results are available in [10]).

The Figure 1 shows the results for all algorithms applied in this study. Clearly, when the BASIC problem is applied, the differences between algorithms are not very large, while the differences start to grow as soon as the EXTENDED problem is used. The solution produced by a given algorithm for the EXTENDED problem, is always worse than for the BASIC problem. Moreover, the relative differences between algorithms are much higher which can be seen especially in the extreme case of FCFS, which totally failed to generate acceptable results. On the other hand, LS optimization of CONS is very successful, significantly decreasing both slowdown and response time. Clearly, additional features such as machine failures or specific job requirements add nontrivial constraints into the decision making process of selected algorithms. Experimental results showed that these constraints should not be ignored otherwise the simulation results are very unrealistic.

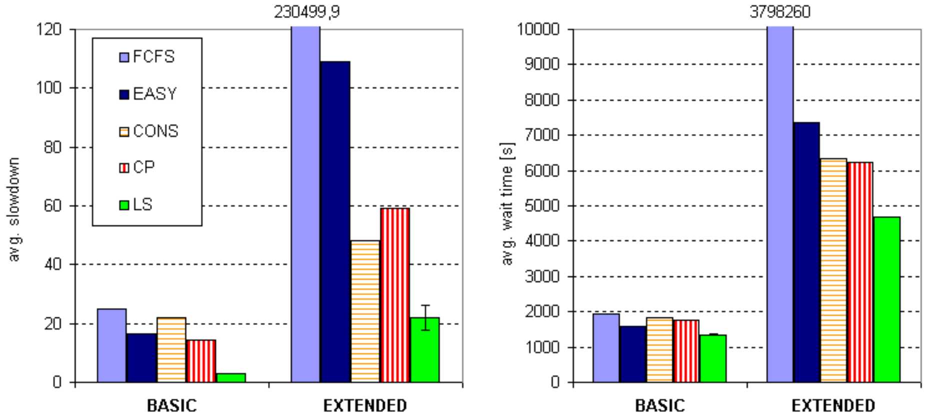

Fig. 1. The average slowdown (left) and average wait time (right) for BASIC and EXTENDED problem.

5 Conclusion

The use of complete and “rich” data set may significantly influence the quality of generated solution as we have published in [10, 7]. In addition, we have shown that similar observations can be made also for other publicly available data sets [10]. If possible, complete data sets should be collected and used to evaluate scheduling algorithms under harder conditions. Their use may narrow the gap between the “ideal world” of simulations and the real-life experience, producing more reliable and realistic experiments. Realistic simulations help to quickly identify possible weaknesses in the algorithm design, allowing to make them more robust and scalable. From this point of view, complex data set from MetaCentrum represents an important source of valuable information for the scientific community.

Acknowledgment

We appreciate the gracious support of the Ministry of Education, Youth and Sports of the Czech Republic under the research intent No. 0021622419. We also highly appreciate the help of MetaCentrum team who provided us the workload data and help us to interpret them correctly, as well as the access to the MetaCentrum computing facilities provided under the research intent MSM6383917201.

References

1. Dick Epema, Shanny Anoep, Catalin Dumitrescu, Alexandru Iosup, Mathieu Jan, Hui Li, and Lex Wolters. Grid workloads archive (GWA). Available at: `http://gwa.ewi.tudelft.nl/pmwiki/`.
2. Carsten Ernemann, Volker Hamscher, and Ramin Yahyapour. Benefits of global Grid computing for job scheduling. In *GRID '04: Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing*, pages 374–379. IEEE, 2004.
3. Dror G. Feitelson. Parallel workloads archive (PWA). Available at: `http://www.cs.huji.ac.il/labs/parallel/workload/`.
4. Dror G. Feitelson. Experimental analysis of the root causes of performance evaluation results: A backfilling case study. *IEEE Transactions on Parallel and Distributed Systems*, 16(2):175–182, 2005.
5. Dror G. Feitelson, Larry Rudolph, Uwe Schwiegelshohn, Kenneth C. Sevcik, and Parkson Wong. Theory and practice in parallel job scheduling. In Dror G. Feitelson and Larry Rudolph, editors, *Job Scheduling Strategies for Parallel Processing*, volume 1291 of *LNCS*, pages 1–34. Springer Verlag, 1997.
6. James Patton Jones. *PBS Professional 7, administrator guide.* Altair, April 2005.
7. Dalibor Klusáček and Hana Rudová. Complex real-life data sets in Grid simulations. In *Cracow Grid Workshop 2009 Abstracts (CGW'09)*, 2009.
8. Dalibor Klusáček and Hana Rudová. Alea 2 – job scheduling simulator. In *Proceedings of the 3rd International Conference on Simulation Tools and Techniques (SIMUTools 2010)*, Torremolinos, Malaga, Spain, 2010.
9. Dalibor Klusáček and Hana Rudová. Efficient grid scheduling through the incremental schedule-based approach. *Computational Intelligence: An International Journal*, 2010. To appear.
10. Dalibor Klusáček and Hana Rudová. The importance of complete data sets for job scheduling simulations. In *Proceedings of the 15th Workshop on Job Scheduling Strategies for Parallel Processing*, Atlanta, USA, 2010.
11. Miroslava Plachá. Dynamické rozvrhování úloh na výpočetní zdroje, 2010. Submitted as a Master Thesis at Faculty of Informatics, Masaryk University, Brno, Czech Republic.

12. Joseph Skovira, Waiman Chan, Honbo Zhou, and David Lifka. The EASY - LoadLeveler API project. In Dror G. Feitelson and Larry Rudolph, editors, *Job Scheduling Strategies for Parallel Processing*, volume 1162 of *LNCS*, pages 41–47. Springer, 1996.
13. Srividya Srinivasan, Rajkumar Kettimuthu, Vijay Subramani, and P. Sadayappan. Selective reservation strategies for backfill job scheduling. In Dror G. Feitelson, Larry Rudolph, and Uwe Schwiegelshohn, editors, *Job Scheduling Strategies for Parallel Processing*, volume 2537 of *LNCS*, pages 55–71. Springer Verlag, 2002.
14. Fatos Xhafa and Ajith Abraham. Computational models and heuristic methods for grid scheduling problems. *Future Generation Computer Systems*, 26(4):608–621, 2010.
15. Ming Q. Xu. Effective metacomputing using LSF multicluster. In *CCGRID '01: Proceedings of the 1st International Symposium on Cluster Computing and the Grid*, pages 100–105. IEEE, 2001.