A Unified Approach for Modeling and Optimization of Energy, Makespan and Reliability for Scientific Workflows on Large-Scale Computing Infrastructures

Rafael Ferreira da Silva*, Thomas Fahringer‡, Juan J. Durillo‡, Ewa Deelman*
*University of Southern California, Information Sciences Institute, Marina Del Rey, CA, USA
‡University of Innsbruck, Institute of Computer Science, Technikerstrasse 21a, Innsbruck, Austria
{rafsilva,deelman}@isi.edu, {tf,juan}@dps.uibk.ac.at

I. Introduction

Green computing has received significant attention in the past few years. Although some research has addressed cooling and energy usage reduction in large data-centers [1], they do not control how resources are used by applications. Scientific workflows are a useful representation for managing the execution of large-scale computations on high performance computing (HPC) and high throughput computing (HTC) platforms [2]. In scientific workflow applications, resource provisioning and utilization optimizations have been investigated to reduce energy consumption on Cloud infrastructures [3], [4]. However, existing research is largely limited to the measurement of energy usage according to resource utilization when running a program on an execution node. Furthermore, most existing optimization techniques for workflows are limited to single objectives (e.g. makespan), and some can deal with only two objectives. There does not exist an approach that deals with an arbitrary number of objectives and no scheduling technique explored tradeoffs among makespan, energy consumption, and reliability.

We recently proposed [5] an energy consumption model for analyzing and profiling energy usage that addresses resource utilization, data movement, and I/O operations. Although our model assembles several models (computing, networking and storage systems) validated in a real execution environment, it still makes strong assumptions on the resource characteristics (e.g. single core homogeneous virtual machines), and ignores external loads.

In this work, we propose 1) an extension of our energy consumption model to address real large-scale infrastructure conditions (e.g. heterogeneity, resource unavailability, external loads); 2) the validation of the model in a fully instrumented platform able to measure the actual temperature and energy consumed by computing, networking, and storage systems; and 3) a multi-objective optimization approach to explore tradeoffs among makespan, energy consumption, and reliability for multi-objective workflow scheduling.

II. Modeling

Scientific workflows allow users to easily express multi-step computational tasks, for example retrieve data from an instrument or a database, reformat the data, and run an analysis. Scientific workflows are often modeled as a directed acyclic graph (DAG), where the nodes in the graph represent computational tasks and the edges represent data or control dependencies. In this model, tasks are typically command-line programs (a.k.a. transformations) that read one or more input files and produce one or more output files, and data dependencies are a result of output files from one program becoming input files for another program. Workflow interpretation and execution are handled by a workflow management system (WMS) that manages the execution of the application on the distributed computing infrastructure.

The execution system is modeled as an Infrastructure as a Service architecture where a submit host (client) interacts with a distributed system to store data and execute computations. Figure 1 illustrates this system model. A WMS running on submit host H sets up the application and manages workflow execution on the resources. Application setup includes providing a set of parameters for the execution, and uploading all input data to a storage server S (step 1). Workflow execution consists of provisioning virtual machines (VMs), scheduling workflow tasks according to data dependencies, and executing tasks on VMs (step 2). Task execution may transfer data using message passing or a shared file system. If the data cannot be transferred through the communication network, it is stored on the storage server (step 3). At the end of the workflow execution, any output data required by the user is downloaded from the storage server to the submit host (step 4).

Optimization parameters such as runtime and reliability are modeled from workflow execution traces. We have developed profiling tools [6], [7] to collect and summarize performance metrics for workflow applications. These tools capture profiling data such as process I/O, runtime, memory usage, and CPU utilization. This profile data is then used to build distributions

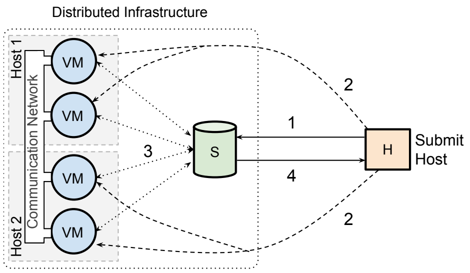

Figure 1. Overview of the system model for the execution of scientific workflows.

of workflow applications. Parameters such as temperature and energy consumed are modeled, however, using measurements acquired on a fully instrumented platform.

III. Multi-Objective Optimization

Most existing workflow development environments focus on a single optimization goal when improving the execution of scientific workflows. For example, many workflows deal with complex, time and memory intensive scientific simulations. Thus, execution time is an important goal. Due to the increased costs of energy and cooling of compute infrastructures such as Clouds and high performance computers, energy efficiency [8], [9], [10] is becoming increasingly important as well. In addition, many researchers [11], [12] have emphasized the significance of fault tolerance that deals with software and hardware failures in particular for large scale compute infrastructures. Today, other objectives such as resource usage, economic costs, memory footprint may be equally important, and require to be optimized. In many cases, some of these criteria are in conflict, meaning that improving one metric implies deteriorating at least another one. Optimizing scientific workflow execution becomes then a multi-objective optimization problem. The main characteristic of this kind of problem is that there does not exist a single solution that is optimal with respect to all objectives. Instead, a set of tradeoff solutions known as Pareto front should be derived. Solutions within this set cannot be further improved in any of the objectives without causing the degradation of at least another problem's objective. Once the Pareto front is computed, an automatic or manual procedure selects the most viable/preferred solution out of this set.

IV. Research Dimensions

In this work, we propose a process to attain multi-objective optimization of energy consumption, makespan, and reliability for scientific workflows on large-scale computing infrastructures. Figure 2 describes the interaction between the components involved in the process. Workflow executions are constantly monitored to collect fine-grained information about task executions (e.g. CPU utilization, runtime, memory usage, I/O, and errors). This data has been collected as part of the DOE dV/dt project (ER26110) [13] using the profiling tools described in Section II, and is freely available online for the community. This monitoring acts at the application execution level, and is often performed by the workflow management system, which in our case is the Pegasus WMS [14]. Currently, our profiling tools work at mostly large-scale infrastructures, but for some HPC platforms such as the IBM Blue Gene, fine-grained monitoring still remains a challenge due to the system design (e.g. process forking is not allowed).

The monitoring of temperature and energy consumption, however, requires access to fully instrumented infrastructures, and involves monitoring at the infrastructure level. There are several studies that examined the energy consumption of applications on HPC systems [15], [16], but there is no study on the energy-aware profiling of scientific workflow executions on such platforms. Therefore, we plan to run scientific workflow experiments on these infrastructures to collect temperature and energy consumption data and build energy-aware profiles. We then plan to use these profiles to validate our energy consumption model [5], and at the same time extend it to accommodate real infrastructure conditions such as heterogeneity of resources and external loads. These models can then be used to drive multi-objective workflow scheduling optimizations in future workflow executions.

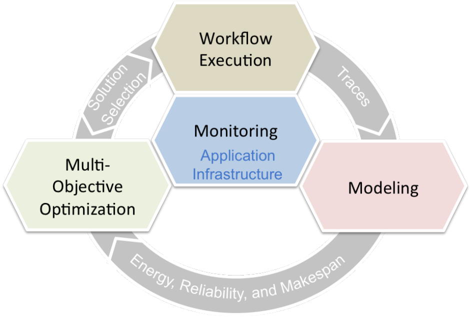

Figure 2. Multi-objective optimization process.

Finding the optimal mapping of tasks onto specific processors or compute nodes is known to be an NP-Hard problem in the case of optimizing for a single goal. Computing such a schedule to optimize several criteria is a problem of the same complexity class. We model the multi-objective optimization problem as a DAG representing the workflow. Tasks and edges of this DAG are annotated with models for the makespan, energy consumption, and reliability. In addition, prior to schedule a workflow DAG, it may be transformed by means of transformations and strategies that explore clustering, data chunk sizes, etc, that may highly impact one or more criteria. Multi-objective scientific workflow optimization is then confronted with a large search space of possible workflow executions. Every point of this search space consists in a transformed version of the workflow and a mapping of tasks onto processors or compute node. Finding these executions that are on the Pareto front with as little effort (reduced search space based on heuristics) as possible, is one important goal of this research. A solution out of the Pareto front is then selected to improve the workflow execution. Note that this optimization process may be dynamic, i.e. profiling data collected during the workflow execution are used to update the models, which may affect the Pareto front solutions if the execution behaves differently from the models.

As part of this research challenge, we plan to extend the MOHEFT (Multi-Objective HEFT) [17] method—a list based-heuristic for multi-objective optimization workflow scheduling based on the HEFT [18] method—with the energy consumption, makespan, and reliability models, and conduct experimental evaluation through simulations. The execution profiles will also be used to develop realistic simulation scenarios. This approach could be potentially extended to general parallel programs. The concept of parallel tasks occurs in numerous programming paradigms such as OpenMP, Intel Thread Building Blocks, OpenCL, etc.

References

[1] A. Beloglazov, R. Buyya, Y. C. Lee, and A. Y. Zomaya, "A taxonomy and survey of energy-efficient data centers and cloud computing systems," *Advances in Computers*, vol. 82, pp. 47–111, 2011.

[2] I. Taylor, E. Deelman, D. Gannon, and M. Shields, *Workflows for e-Science*. Springer, 2007.

[3] I. Pietri, M. Malawski, G. Juve, E. Deelman, J. Nabrzyski, and R. Sakellariou, "Energy-constrained provisioning for scientific workflow ensembles," in *Third International Conference on Cloud and Green Computing (CGC'13)*, 2013, pp. 34–41.

[4] T. Guérout, T. Monteil, G. D. Costa, R. N. Calheiros, R. Buyya, and M. Alexandru, "Energy-aware simulation with DVFS," *Simulation Modelling Practice and Theory*, vol. 39, pp. 76–91, 2013.

[5] R. Ferreira da Silva, G. Juve, T. Fahringer, and E. Deelman, "Analyzing energy-efficiency in data-intensive scientific workflows," in *10th IEEE International Conference on e-Science*, 2014, p. submitted.

[6] J. S. Vockler, G. Mehta, Y. Zhao, E. Deelman, and M. Wilde, "Kick-starting remote applications," in *2nd International Workshop on Grid Computing Environments*, 2006.

[7] G. Juve, B. Tovar, R. Ferreira da Silva, C. Robinson, D. Thain, E. Deelman, W. Allcock, and M. Livny, "Practical resource monitoring for robust high throughput computing," in *Middleware 2014*, 2014, p. submitted.

[8] H. M. Fard, R. Prodan, J. J. D. Barrionuevo, and T. Fahringer, "A multi-objective approach for workflow scheduling in heterogeneous environments," in *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (Ccgrid 2012)*, ser. CCGRID '12, 2012, pp. 300–309.

[9] B. Gao, L. He, L. Liu, K. Li, and S. A. Jarvis, "From mobiles to clouds: Developing energy-aware offloading strategies for workflows," in *Proceedings of the 2012 ACM/IEEE 13th International Conference on Grid Computing*, ser. GRID'12, 2012, pp. 139–146.

[10] J. Durillo, V. Nae, and R. Prodan, "Multi-objective workflow scheduling: An analysis of the energy efficiency and makespan tradeoff," in *13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, 2013, pp. 203–210.

[11] Y. Zhang, M. Squillante, A. Sivasubramaniam, and R. Sahoo, "Performance implications of failures in large-scale cluster scheduling," in *Job Scheduling Strategies for Parallel Processing*, ser. Lecture Notes in Computer Science, D. Feitelson, L. Rudolph, and U. Schwiegelshohn, Eds., 2005, vol. 3277, pp. 233–252.

[12] B. Schroeder and G. A. Gibson, "A large-scale study of failures in high-performance computing systems," in *Proceedings of the International Conference on Dependable Systems and Networks*, ser. DSN'06, 2006, pp. 249–258.

[13] dv/dt: Accelerating the rate of progress towards extreme scale collaborative science. [Online]. Available: http://sites.google.com/site/acceleratingexascale

[14] E. Deelman, G. Singh, M.-H. Su, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, K. Vahi, G. B. Berriman, J. Good, A. Laity, J. C. Jacob, and D. S. Katz, "Pegasus: A framework for mapping complex scientific workflows onto distributed systems," *Sci. Program.*, vol. 13, no. 3, pp. 219–237, 2005.

[15] L. Carrington, M. Laurenzano, and A. Tiwari, "Characterizing large-scale HPC applications through trace extrapolation," *Parallel Processing Letters*, vol. 23, no. 04, p. 1340008, 2013.

[16] H. Shoukourian, T. Wilde, A. Auweter, and A. Bode, "Monitoring power data: A first step towards a unified energy efficiency evaluation toolset for HPC data centers," *Environmental Modelling & Software*, vol. 56, pp. 13–26, 2014.

[17] J. Durillo, H. Fard, and R. Prodan, "Moheft: A multi-objective list-based method for workflow scheduling," in *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*, 2012, pp. 185–192.

[18] H. Topcuoglu, S. Hariri, and M.-Y. Wu, "Performance-effective and low-complexity task scheduling for heterogeneous computing," *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, vol. PDS-13, no. 3, pp. 260–274, Mar. 2002.