ARTICLE

Nature Inspired Schedulers in Computational Grids

Fatos Xhafa* & Ajith Abraham**

* Technical University of Catalonia, Spain, fatos@lsi.upc.edu
** Machine Intelligence Research Labs (MIRLabs), USA, ajith.abraham@ieee.org

1. Introduction

Computational Grids (CGs) are a natural extension of classical distributed systems. CGs are expected to leverage unprecedented larger computing capacities by virtually joining together geographically distributed resources at large scale. To achieve this objective, the design of ecient Grid schedulers that map and allocate tasks and applications onto Grid resources is a key issue. In fact, CGs introduce new complexities to the scheduling problem making it even more challenging for the design of ecient schedulers. In this article, we illustrate how various nature inspired heuristic and meta-heuristic methods can be used to design ecient schedulers in CGs. Such heuristic and meta-heuristics include ad hoc methods, local search, population-based and hybrid approaches. In using these methods for designing ecient Grid schedulers we take into account the new characteristics of CGs and tackle the Grid scheduling as a family of problems.

2. Background

Scheduling in distributed systems is one of the most studied problems in the optimization theory. The intrinsics characteristics of CGs add new complexities to the problem as compared to its traditional version in conventional distributed systems.

Main Characteristics of Computational Grids. CGs distinguish for their *dynamic structure* (resources in a Grid system can join or leave the Grid in an unpredictable way), the *high heterogeneity of resources* (the computational resources could be very disparate in their computing capacities), the *high heterogeneity of tasks* (tasks arriving to any Grid system are diverse and heterogenous in terms of their computational needs), the high *heterogeneity of interconnection networks* (resources are connected through different interconnection networks). On the other hand, CGs are cross-administrative infrastructures and thus there are Grid schedulers which should co-exists with local schedulers and also local policies on access and usage of resources. *Scalability and efficiency* are as well essential features to Grid schedulers as CGs are expected to be large scale. Finally, beyond the efficiency, Grid schedulers should also take into account security and trust requirements.

An Example: Grid Batch Scheduling. Scheduling problem in Computational Grids, in its general form, can be stated as in the case of scheduling problem for distributed systems: compute a mapping of a given set of tasks onto a set of available machines. However, behind this general denition, there are many concrete versions of the problem. The Grid batch scheduling is an important version of the problem in which the processing of independent tasks in batches (see Fig. 1). The problem is formalized using the Expected Time To Compute matrix (ETC model) in which ETC[j][m] indicates an estimation of time to complete task j in resource m. Several objectives, including makespan, flowtime, resource utilization and matching proximity

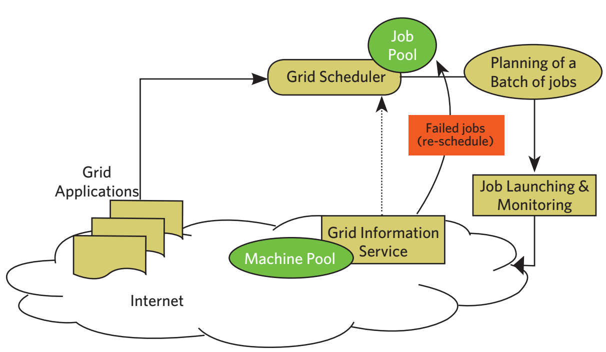

Fig. 1 : Batch Scheduler View

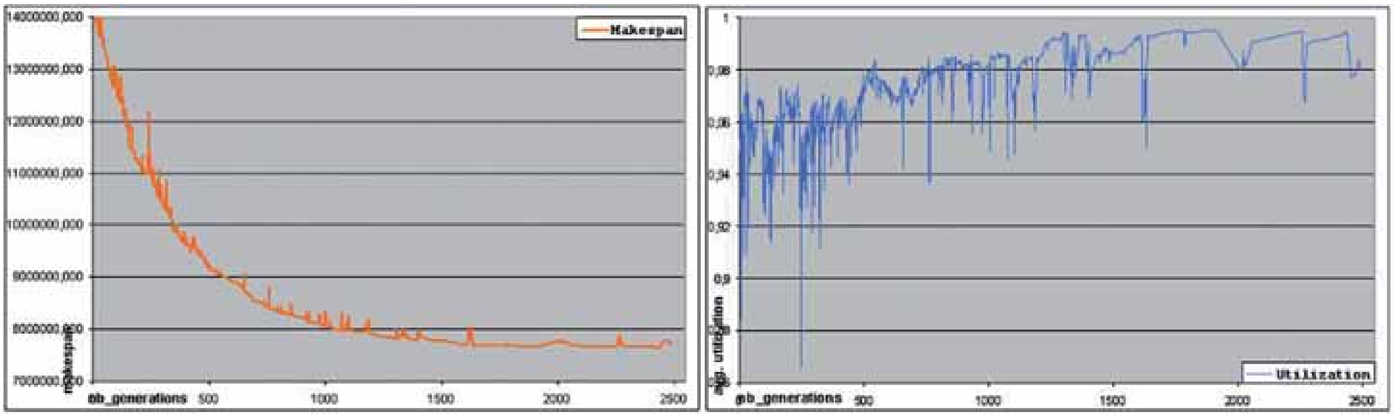

Figure 2: The fluctuations of makespan and flowtime – simultaneous optimization.

can be formulated. In fact, these objectives could contradict each other especially close to optimality (see fluctuations in Fig. 2).

3. Heuristic Methods for Scheduling in Grid Systems

At a higher level, the heuristic methods are usually classified as calculus based, random or enumerative. In the first group we can find greedy algorithms and ad hoc methods, which implement a direct strategy for computing the solution. In the second group, which is the largest one, we have guided and non-guided methods. The former includes Simulated Annealing, Tabu Search, and the large family of evolutionary algorithms. The final group comprises of dynamic programming and branch-and-bound algorithms.

In the case of Grid scheduling problem, many methods from the aforementioned groups, must be applied to the problem in order to design the most efficient scheduler depending on the Grid scenario. We list below the most used methods for the problem for the two groups, namely, ad hoc, local search-based and population-based methods.

Ad hoc methods Ad hoc methods comprise both immediate and batch scheduling and are usually used for single-objective optimization case. The group of immediate mode methods includes Opportunistic Load Balancing, Minimum Completion Time, Minimum Execution Time, Switching Algorithm and k-Percent Best. An example of their efficacy for a set of three tasks and three machines is given in Figure 3. The group of batch mode methods comprises Min-Min, Max-Min, Suerage, Relative Cost and Longest Job to Fastest Resource - Shortest Job to Fastest Resource.

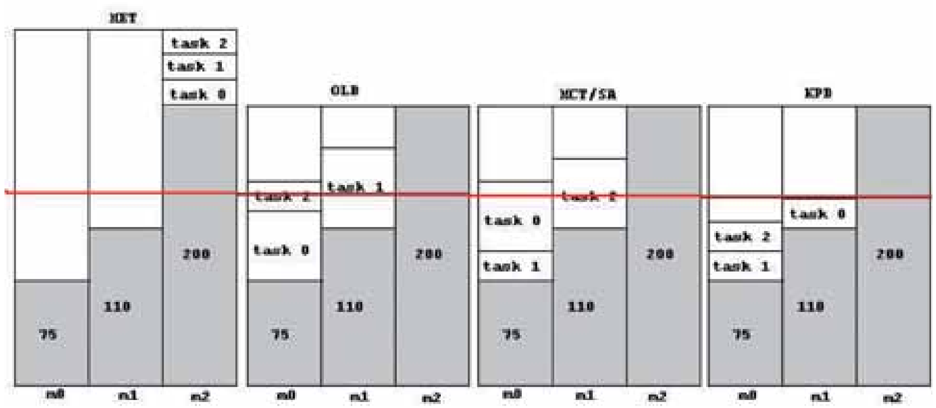

Fig. 3 : An example of scheduling computed by ad hoc methods.

Local search methods Local search is a family of methods that explore the solution space starting from an initial solution and constructing a path in solution space. Methods in this family comprise of Hill Climbing, Simulated Annealing (SA) and Tabu Search (TS), among others. SA is more powerful than simple local search by accepting also worse solutions with certain probability [1]. TS is a more sophisticated but also more computationally expensive due to its mechanisms of tabu lists, aspiration criteria, intensification and diversification. Abraham et. al. [1] considered TS as candidate solution method for the problem. Xhafa et. al. [5] presented a full featured TS for the scheduling problem under ETC model.

Population-based methods Population-based heuristics use populations of individuals to explore the solution space. This family comprises Genetic Algorithms (GAs), Memetic Algorithms (MAs), Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO). GAs for Grid scheduling have been addressed by Abraham et al. [1] and Xhafa et al. [6, 7]. MAs is class of population-based methods, which combine the concepts of evolutionary search and local search. Xhafa [3] applied unstructured MAs and Xhafa et al. [4] proposed Cellular MAs (structured MAs) for the independent scheduling problem under ETC model. ACO and PSO have also been considered for the scheduling problem. Abraham et al. [2] proposed an approach using fuzzy PSO algorithm.

Hybrid approaches Although, meta-heuristics are in nature hybrid, higher level approaches combining stand alone heuristics are also being reported in the literature for the problem. Recently, Xhafa et al. [8] proposed a GA(TS) algorithm for the scheduling of independent tasks under ETC model.

4. Conclusions

This article addressed the use of heuristic and meta-heuristic methods for designing efficient Grid schedulers. We have brie y reviewed different families of heuristic methods used for the resolution of the problem, including ad hoc methods, local search methods and population-based methods. The use of different heuristic methods is very useful to design efficient schedulers according to different Grid scenarios and Grid-enabled application types.

Future trends in this research will be considering the integration of security and trust requirements into Grid schedulers.

References

1. A. Abraham, R. Buyya, and B. Nath.

`Nature's heuristics for scheduling jobs on computational grids`, In *The 8th IEEE Int. Conf. on Advanced Computing and Communications (ADCOM 2000) India, 2000.*

2. A. Abraham, H. Liu, W. Zhang, and T. Chang. Scheduling jobs on computational grids using fuzzy particle swarm algorithm. In *10th Int. Conf. on Knowledge-Based & Intelligent Information & Engineering Systems*, LNCS, Springer, 2006.
3. F. Xhafa, A Hybrid Evolutionary Heuristic for Job Scheduling in Computational Grids. Springer Verlag Series: Studies in Computational Intelligence , Vol. 75 2007, Chapter 10, ISBN: 978-3-540-73296-9. September 2007.
4. F. Xhafa, E. Alba, B. Dorronsoro, and B. Duran. Efficient Batch Job Scheduling in Grids using Cellular Memetic Algorithms, Journal of Mathematical Modelling and Algorithms, 7(2):217-236, 2008.
5. F. Xhafa, J. Carretero, E. Alba, B. Dorronsoro. Tabu Search Algorithm for Scheduling Independent Jobs in Computational Grids, Computer And Informatics Journal, Vol. 28, No 2, 2009, pp 237-249.
6. F. Xhafa, J. Carretero, A. Abraham. Genetic Algorithm Based Schedulers for Grid Computing Systems. International Journal of Innovative Computing, Information and Control, Vol. 3, No.5, pp. 1-19, 2007.
7. F. Xhafa, B. Duran, A. Abraham and K. P. Dahal. Tuning Struggle Strategy in Genetic Algorithms for Scheduling in Computational Grids. Neural Network World. Vol. 18, No. 3, 209-225, Jul, 2008.
8. F. Xhafa, J.A. Gonzalez, K. P. Dahal, A. Abraham: A GA(TS) Hybrid Algorithm for Scheduling in Computational Grids. HAIS 2009: 285-292, LNCS, Springer