Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds

PROMOTING GREEN INTERNET COMPUTING THROUGHOUT SIMULATION-OPTIMIZATION SCHEDULING ALGORITHMS

Guillem Cabrera
Hebert Pérez-Roses

Internet Interdisciplinary Institute
Universitat Oberta de Catalunya
Barcelona, SPAIN

Angel A. Juan
Joan M. Marquès

Computer Science Department
IN3-Open University of Catalonia
Barcelona, SPAIN

Javier Faulin

Department of Statistics and Operations Research
Public University of Navarra
Pamplona, SPAIN

ABSTRACT

This work introduces an application of simulation-optimization techniques to the emerging field of green internet computing. The paper discusses the relevance of considering environmental factors in modern computing and then describes how simulation can be combined with scheduling metaheuristics in order to reduce the expected time needed to complete a set of tasks in a server under the realistic assumption of stochastic processing times. This, in turn, allows for a reduction in average energy consumption, which makes the computing facility more efficient from an environmental perspective. Some experiments have been carried out in order to illustrate these potential savings.

1 INTRODUCTION

Computational infrastructures supporting the Grid and the Cloud computing paradigms experienced a nearly-exponential growth in the last decade, and this trend is expected to continue with the increasing use of multiple Internet services –e.g., social networks, web mail, video-conferencing, shared repositories, etc. Maintaining these large infrastructures is expensive and far from being environmentally friendly (Garg et al. 2011, Glanz 2012). In effect, these infrastructures consume vast amounts of energy to run the servers, the network equipment, and the cooling systems. Some experts have estimated that only around 10% of the energy used in a datacenter goes to powering active servers. The remaining energy is used for keeping idle servers waiting for activities from user requests. In this context, governments and datacenter companies worldwide are starting to be concerned about the effects over the environment associated with the maintenance of these large computing infrastructures (European Commission 2009). Thus, there is a tendency towards "greener" computing practices. In the so called Green Computing (GC) paradigm, not only computing performance, quality of service (QoS), and traditional costs are considered, but also other factors such as: the carbon footprint of companies, the energy consumption, and the greenhouse gas emissions. The challenge is then to reduce the environmental impact of such infrastructures while maintaining, as much as possible, their performance and QoS levels (Zhang et al. 2010).

One of the critical issues in GC is the scheduling of tasks (jobs) to be processed by servers (machines) in large computing infrastructures. Optimal scheduling policies can represent noticeable reductions in the total time necessary to process a set of programmed tasks, also known as makespan (Figure 1). In turn, noticeable reductions in makespan can imply significant savings in energy consumption, since the computing infrastructures are utilized in a more efficient way and require less total power to complete the assigned tasks. Despite the existence of a large number of scheduling-optimization algorithms, most of them assume deterministic values for the time each task requires in order to be processed by each server. However, in our opinion this is a quite unrealistic assumption, since these task-server processing times are random variables –as opposed to deterministic values– in most realistic scenarios. In order to contribute to fill this gap, this paper discusses how Monte-Carlo simulation can be combined with existing metaheuristics in order to reduce the expected energy consumption in tasks scheduling problems with stochastic processing times. In particular, among the different scheduling problems existing in the literature, we will focus on the well-known Flow-Shop Problem (FSP) (Ruiz and Maroto 2005). Notice, however, that simulation-based approaches similar to the one considered in this work could be employed in solving other scheduling variants with stochastic components.

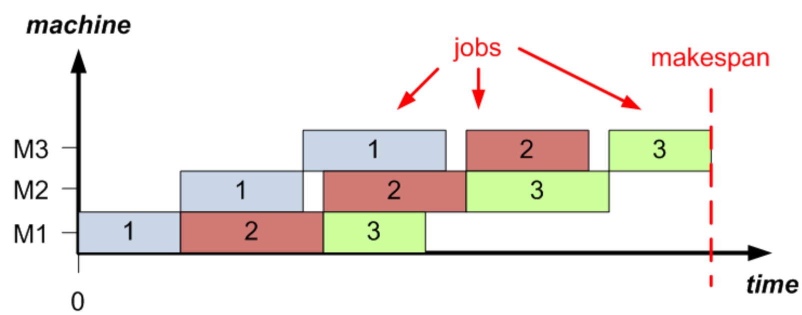

Figure 1: A simple flow-shop scheduling problem with 3 jobs and 3 machines

Section 2 offers a brief literature review on green computing as well as on stochastic scheduling. Section 3 provides an overview of how simulation can be combined with metaheuristics for solving stochastic scheduling problems. Computational experiments are then discussed in Section 4. These experiments illustrate how different scheduling policies compare in terms of their associated outcomes, i.e.: expected makespan and energy consumption. Finally, Section 5 highlights the main contributions of this work.

2 LITERATURE REVIEW

2.1 Green Computing

Issues regarding the development of environmentally-sustainable computing have been identified and discussed by several authors since the emergence of the cloud and grid computing paradigms. For instance, Chen et al. (2008) analyze different algorithms for supporting connection-intensive Internet services taking into account several factors such as load distribution, user experience, and energy consumption. Likewise, Le et al. (2009) present a framework for optimization-based request distribution among datacenters by introducing the energy consumption when seeking for datacenters in under-peak demands, datacenters close to a renewable energy source, or datacenters with cheaper cost of the energy. Their heuristic redirects user requests to those datacenters that can offer the required Service Level Agreement while minimizing energy consumption and operation cost. In Garg et al. (2011), differences among the various data centers of the same company are considered in order to improve efficiency and reduce carbon emissions. They define an energy model for datacenters which includes metrics to measure energy cost, carbon emission, profits, etc. and propose heuristic scheduling strategies. Duy et al. (2010) design, implement, and evaluate a Green Scheduling Algorithm with a neural network predictor for optimizing power consumption in cloud computing datacenters. Their algorithm predicts future load demands from historical data and turns off or restarts servers according to the predicted demand, saving up to 46% of energy. Several innovative "green" task-scheduling algorithms are presented in Zhang et al. (2010). After a simulation experiment, the authors conclude that heuristically assigning tasks to computers with lower energy is significantly more energy-efficient than assigning tasks to random computers.

In Lee and Zomaya (2012) the authors discuss about energy wastage caused by under-utilized resources within a conventional cloud computing environment and argue that task consolidation is a good resource re-allocation procedure, thanks to the DVFS features included in modern processors. Two energy-conscious task consolidation heuristics are presented and analyzed, both of them reducing the energy consumption of clouds. In another recent paper, Beloglazov et al. (2011) address the problem of efficient power management in data centers or processor farms. They propose a technique for efficient power management based on virtualization that allocates virtual machines to actual servers dynamically, according to changes in the workload, and switch idle servers to sleep mode. Borgetto et al. (2012) study the problem of energy-aware resource allocation for long-term services or on-demand computing tasks hosted in clusters. These authors formalize the problem by three NP-hard constrained optimization problems: (a) maximize job performance under energy consumption constraints; (b) minimize power consumption under performance constraints; and (c) optimize a linear combination of power consumption and job performance. They propose several heuristics for the three problems and use simulation to validate their approaches by comparing their results with those in realistic scenarios.

All the works cited above are mainly concerned with the energy consumed by the servers themselves. However, the fraction of the energy consumed by other devices, such as network infrastructure, is by no means negligible. Interest in this topic is more recent, albeit growing rapidly. Berl et al. (2010) discuss such issues within the framework of the overall energy efficiency of different distributed information technologies, with emphasis in cloud computing. GreenCloud is a simulation environment for energy-aware cloud computing datacenters designed to forecast the energy consumption associated with each system component (Kliazovich et al. 2010).

2.2 The Flow-Shop Problem with Stochastic Times

The literature on the Flow Shop Problem with Stochastic Times (FSPST) is not as extensive as the deterministic case. In Pinedo (1982), models for the permutation flow-shop problem with and without intermediate storage are proposed, where the processing times of a given job on the various machines are independent realizations from the same probability distribution and the objective is to minimize the expected makespan. Dodin (1996) considered different probability distributions for processing times. He assumed these processing times were independent and not identically distributed random variables, and his goal was also to minimize the expected makespan. Gourgand et al. (2003, 2005) used recursive algorithms, based on Markov chains, to compute the expected makespan in conjunction with a simulation model as a backup to evaluate the expected makespan. Wang et al. (2005a, 2005b) proposed Genetic Algorithms (GA) approaches for minimizing the expected makespan when processing times follow a Uniform distribution. More recently, Baker and Trietsch (2011) developed heuristics for the two-machine flow-shop problem where the processing times are independent random variables following a known probability distribution –also with the objective of minimizing the expected makespan. Some authors also studied different variations of the flow-shop problem: Allaoui et al. (2006), Choi and Wang (2012), and Kianfar et al. (2012) considered the stochastic hybrid flow-shop scheduling problem. In Zhou and Cui (2008), an approach for the multi-objective stochastic flow shop problem is considered. Here, processing times are normally distributed with two objectives to minimize: flow time and delay time of the jobs.

Most of the aforementioned publications made restrictive assumptions on the probability distributions modeling the processing times of each job-machine pair and the independence and identical distribution of the tasks. Recently, Baker and Altheimer (2012) combined simulation and heuristics to solve the

FSPST with the objective of minimizing the expected makespan. They created three sets of stochastic flow shop problems where the processing times were distributed following different distributions. The authors proposed three heuristic methods, two of them based on the CDS heuristic by Campbell et al. (1970). The third one is based on the well known NEH heuristic by Nawaz et al. (1983).

3 METHODOLOGY OVERVIEW

In the stochastic scheduling scenario described before, our goal is to find a tasks schedule that allows us to reduce the expected makespan and, consequently, the associated energy consumption. Our approach is inspired in two facts: (a) the FSP is just a FSPST with constant demands –random demands with zero variance; and (b) while the FSPST is yet an emerging research area, extremely efficient metaheuristics do already exist for solving the FSP. Thus, given an initial FSPFST instance, our approach proposes to transform the stochastic instance into a deterministic instance by replacing each random variable time by its mean or expected time; and then to obtain a set of high-quality solutions for the deterministic problem by using any efficient FSP algorithm, e.g. the NEH heuristic from Nawaz et al. (1983) or the IG metaheuristic form Ruiz and Stützle (2007). A solution to the FSP instance is simply a given permutation of tasks, and therefore it is also a feasible solution of the original FSPST instance. Then, simulation is used to determine which solution, among the best-found deterministic ones, produces the lowest expected makespan when considering stochastic times. This simulation process works as follows. Given a solution (permutation of tasks) to the FSP, the probability distribution of each job-machine processing time is used to generate a random variate; and then, these random variates are used to compute the stochastic makespan associated with the given permutation of tasks. Iterating this process, it is possible to generate a random sample of makespan observations associated with the given solution. From this sample, different statistics for the stochastic makespan can be obtained, including point and interval estimates for the expected stochastic makespan. Our strategy assumes that there exists a strong correlation between solutions for the FSP and solutions for the FSPST. We expect this assumption to be particularly valid when stochastic times show a relatively low variability. Thus, Figure 2 shows a scatterplot of how this correlation behaves for one of the instances discussed in the experimental section and for different variability levels of the stochastic times.

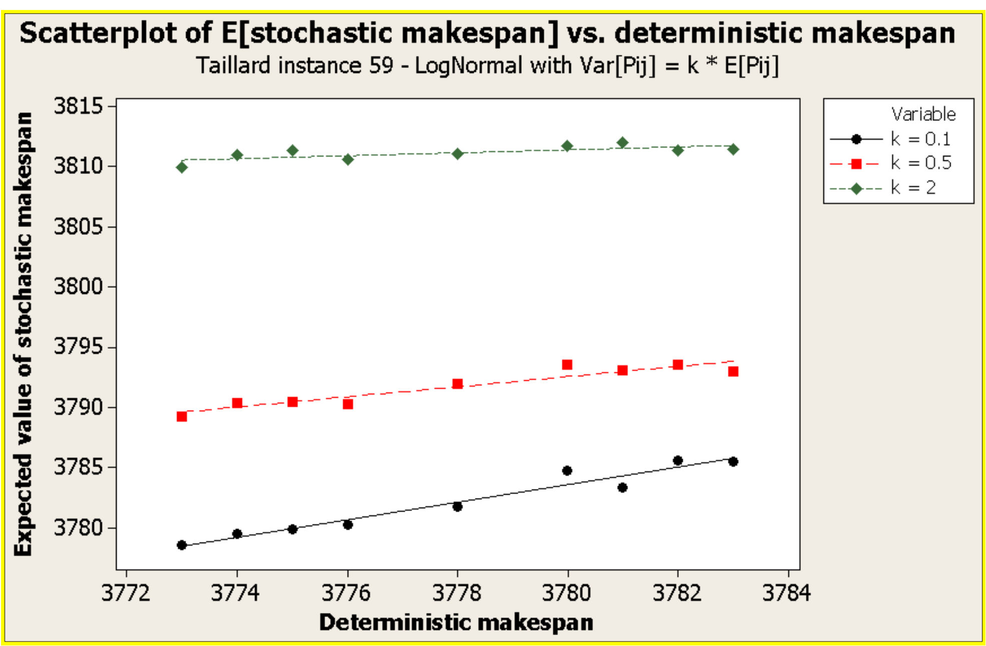

Figure 2: Testing the correlation assumption (axes in time units)

Notice that there is a clear correlation between expected makespans for the FSPST and deterministic makespans for the FSP. However, the best-found FSP solution –the one with the lowest 'deterministic' makespan– will not necessarily be the best-found FSPST solution –the one with the lowest expected stochastic makespan.

4 EXPERIMENTS

In order to test how our approach could contribute to reduce expected makespan and energy consumption in a computing facility, we developed a set of stochastic instances which can be seen as a generalization of the deterministic PFP instances proposed by Taillard (1993) (available at: `http://mistic.heig-vd.ch/taillard`). Taillard's instances are grouped in sets of 10, according to the number of jobs, *n*, and machines, *m*, considered. Usually, these numbers are represented at the end of the name of each instance in the form *n x m*. Thus, the following 10 Taillard's instances were randomly selected for our experiments: *tai007_20x5*, *tai011_20x10*, *tai036_50x5*, *tai045_50x10*, *tai067_100x5*, *tai087_100x20*, *tai097_200x10*, *tai104_200x20*, *tai112_500x20*, and *tai118_500_20*.

For each of these deterministic instances, we changed the constant processing times by random variables as follows: if p_{ij} represents the processing time (in hours) of the *i*-th task in the *j*-th server, then we assumed this processing time to be a random variable, P_{ij}, following a Log-Normal probability distribution with $E[P_{ij}] = p_{ij}$ and $Var[P_{ij}] = k\, p_{ij}$, where $k \geq 0$. In our experiments, we set $k = 5$ in order to consider a scenario with a relatively high variance. At this point, it is important to notice that, even when we used a particular probability distribution for generating our numerical experiments, the methodology explained here could be employed with any other probability distribution and parameters.

Then, we implemented our algorithm as a Java application. Even though Java may run slower than other compiled languages (such as C or C++), it offers some advantages like the fact that it is platform independent as well as that it facilitates duplicability of results (Luke 2009). An Intel Xeon at 2.0 GHz and 4 GB RAM was used to perform all tests, which were run directly on the Netbeans IDE platform for Java over Windows 7. Table 1 shows the results obtained by applying our simulation approach in combination with several scheduling algorithms: (a) Sim-Random refers to using a random order of tasks; (b) Sim-NEH refers to using the order provided by the NEH heuristic; and (c) Sim-IG refers to using the order provided by the IG metaheuristic.

Table 1: Comparison among different scheduling policies

		Expected makespan (in hours)			**Expected energy consumption* (in kWh)**			**Expected time and energy savings (in %)**	
Instance	**No. Servers**	**Sim-Random (1)**	**Sim-NEH (2)**	**Sim-IG (3)**	**Sim-Random (4)**	**Sim-NEH (5)**	**Sim-IG (6)**	**(2)-(1)**	**(3)-(1)**
tai007_20_5	5	1,535.8	1,287.6	1,252.7	6,927.8	6,437.9	6,263.5	-16.2%	-18.4%
tai011_20_10	10	2,012.8	1,696.7	1,613.3	18,156.4	16,966.8	16,133.2	-15.7%	-19.8%
tai036_50_5	5	3,2239	2,863.3	2,838.2	16,248.4	14,316.6	14,191.0	-11.2%	-12.0%
tai045_50_10	10	3,749.7	3,157.7	3,041.8	34,218.8	31,577.2	30,418.1	-15.8%	-18.9%
tai067_100_5	5	5,988.7	5,340.9	5,310.4	30,143.4	26,704.6	26,551.8	-10.8%	-11.3%
tai087_100_20	20	8,048.1	6,695.1	6,398.3	146,701.6	133,902.4	127,966.4	-16.8%	-20.5%
tai097_200_10	10	12,844.0	11,034.2	10,934.1	121,655.2	110,342.2	109,341.0	-14.1%	-14.9%
tai104_200_20	20	13,494.0	11,738.7	11,448.7	258,384.4	234,774.0	228,973.6	-13.0%	-15.2%
tai112_500_20	20	31,242.2	27,399.9	26,933.0	602,051.0	547,997.8	538,660.2	-12.3%	-13.8%
tai118_500_20	20	31,070.6	27,383.0	26,890.2	593,882.8	547,660.0	537,803.0	-11.9%	-13.5%
Averages								*-13.8%*	*-15.8%*

(*) Assuming each server consumes 1kW/h including cooling activities.

Figure 3 shows a comparison among the different scheduling strategies and the savings each of them can provide. The following conclusions can be extracted for the sample set of instances considered:

- By using an extremely fast heuristic like the NEH, it is possible to obtain in 'real time' –less than a second for the size of the instances considered– a tasks schedule which is able to generate noticeable savings (about 13.8% on the average) in both expected makespan and expected energy consumption.
- These savings can be increased even further (up to a 15.8%) by employing a state-of-the-art metaheuristic, which employs just a few seconds in generating a near-optimal tasks schedule.

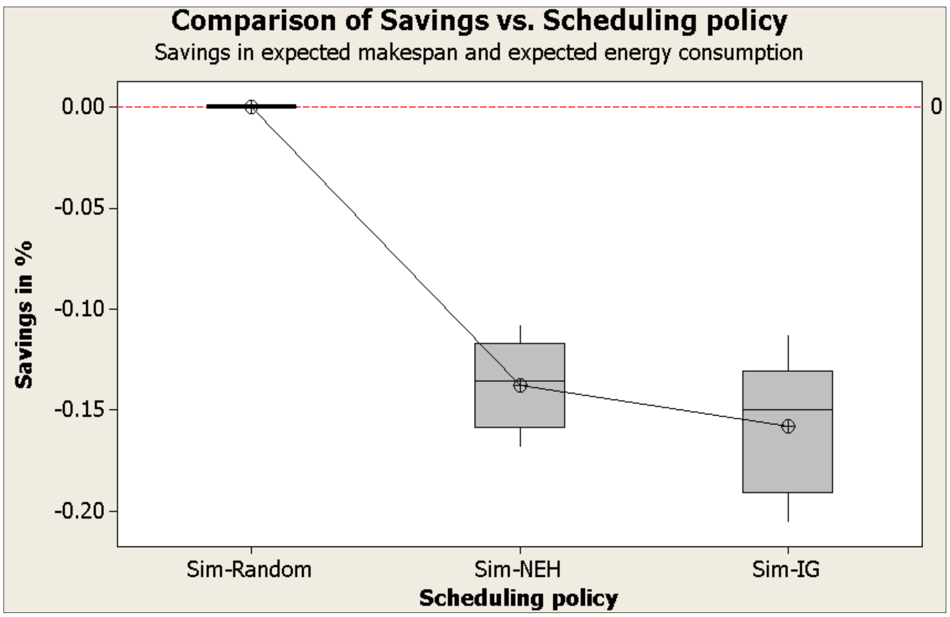

Figure 3: Boxplot comparison of savings vs. scheduling policy

5 CONCLUSIONS

In this article, the relevance of considering energy consumption in growing Internet computing infrastructures as the demand for new services grows at an outstanding rate. We have argued that by using optimization algorithms to schedule tasks in servers, energy consumption in these facilities could be reduced in a noticeable way. In our approach, we have considered stochastic processing times as a more realistic assumption than just considering constant ones. Thus, simulation has been combined with existing metaheuristic algorithms in order to obtain numerical values in a set of benchmarks. The results of these initial experiments show remarkable reductions, both in expected makespan as well as in energy consumption, with regards to using a random scheduling strategy.

ACKNOWLEDGMENTS

This work has been partially supported by the Spanish Ministry of Science and Innovation (TRA2010-21644-C03), the ICSO-HAROSA Network (CYTED2010-511RT0419, `http://dpcs.uoc.edu`), and the IN3-UOC Doctoral Program.

REFERENCES

Allaoui, H., S. Lamouri, and M. Lebbar. 2006. "A Robustness Framework for a Stochastic Hybrid Flow Shop to Minimize the Makespan." In *Proceedings of the International Conference on Service Systems and Service Management*, 1097–1102. Troyes, France.

Baker, K. R., and D. Altheimer. 2012. "Heuristic Solution Methods for the Stochastic Flow Shop Problem." *European Journal of Operational Research* 216:172–177.

Baker, K. R., and D. Trietsch. 2011. "Three Heuristic Procedures for the Stochastic, Two-machine Flow Shop Problem." *Journal of Scheduling* 14:445–454.

Beloglazov, A., J. Abawajy, and B. Rajkumar. 2011. "Energy-aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing." *Future Generation Computer Systems* 28:755–768.

Berl A., E. Genlebe, M. Di Girolamo, G. Giuliani, H. De Meer, M. Quan Dang, and K. Pentikousis. 2010. "Energy-efficient Cloud Computing." *The Computer Journal* 53:1045–1051.

Borgetto D., H. Casanova, G. Da Costa, and J. M. Pierson. 2012. "Energy-aware Service Allocation". *Future Generation Computer Systems* 28(5):769–779.

Campbell, H. G., R. A. Dudek, and M. L. Smith. 1970. "A Heuristic Algorithm for the n Job, m Machine Sequencing Problem." *Management Science* 16:630–637.

Chen, G., W. He, J. Liu, S. Nath, L. Rigas, L. Xiao, and F. Zhao. 2008. "Energy-aware Server Provisioning and Load Dispatching for Connection-intensive Internet Services." In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, 337–350. San Francisco, CA, USA.

Choi, S. H., and K. Wang. 2012. "Flexible Flow Shop Scheduling with Stochastic Processing Times: a Decomposition-based Approach." *Computers and Industrial Engineering* 63:362–373.

Dodin, B. 1996. "Determining the Optimal Sequences and the Distributional Properties of their Completion Times in Stochastic Flow Shops." *Computers and Operations Research* 23:829–843.

Duy, T., Y. Sato, and Y. Inoguchi. 2010. "Performance Evaluation of a Green Scheduling Algorithm for Energy Savings in Cloud Computing." In *International Symposium on Parallel & Distributed Processing*, 1–8. Atlanta, GA, USA.

Gourgand, M., N. Grangeon, and S. Norre. 2003. "A Contribution to the Stochastic Flow Shop Scheduling Problem." *European Journal of Operational Research* 151:415–433.

Gourgand, M., N. Grangeon, and S. Norre. 2005. "Markovian Analysis for Performance Evaluation and Scheduling in m Machine Stochastic Flow-shop with Buffers of any Capacity." *European Journal of Operational Research* 161:126–147.

European Commission. 2009. "Carbon Footprint: What it is and how to measure it." Accessed August 17, 2013. http://www.envirocentre.ie/includes/documents/Carbon_Footprint-what_it_is_and_how_to_measure_it-JRC_IES-Feb09-b[1].pdf.

Garg, S. K., C. S. Yeo, A. Anandasivam, and R. Buyya. 2011. "Environment-conscious Scheduling of HPC Applications on Distributed Cloud-oriented Data Centers." *Journal of Parallel and Distributed Computing* 71:732–749.

Glanz, J. 2012. "Power, Pollution and the Internet – the Cloud Factories." *The New York Times*, September 2012.

Kliazovich, D., P. Bouvry, Y. Audzevich, and S. U. Khan. 2010. "Greencloud: a Packet-level Simulator of Energy-aware Cloud Computing Data Centers." In *Global Telecommunications Conference*, 1–5. Miami, FL, USA.

Le, K., R. Bianchini, M. Martonosi, and T. Nguyen. 2009. "Cost- and Energy-aware Load Distribution across Data Centers." In *Proceedings of HotPower*, 1–5.

Lee, Y. C., and A. Y. Zomaya. 2012. "Energy Efficient Utilization of Resources in Cloud Computing Systems." *The Journal of Supercomputing* 60(2):268–280.

Luke, S. 2009. *Essentials of Metaheuristics*. Lecture notes, George Mason University. Accessed August 17, 2013. http://cs.gmu.edu/~sean/book/metaheuristics.

Nawaz, M., J. E. Enscore, and I. Ham. 1983. "A Heuristic Algorithm for the m-Machine, n-Job Flowshop Sequencing Problem." *Omega* 11:91–95.

Pinedo, M. 1982. "Minimizing the Expected Makespan in Stochastic Flow Shops." *Operations Research* 30:148–162.

Ruiz, R., and C. Maroto. 2005. "A Comprehensive Review and Evaluation of Permutation Flow Shop Heuristics." *European Journal of Operational Research* 165:479–494.

Ruiz, R., and T. Stützle. 2007. "A Simple and Effective Iterated Greedy Algorithm for the Permutation Flow-shop Scheduling Problem." *European Journal of Operational Research* 177:2033–2049.

Taillard, E. 1993. "Benchmarks for Basic Scheduling Problems." *European Journal of Operational Research* 64:278–285.

Wang, L., L. Zhang, and D. Z. Zheng. 2005a. "A Class of Hypothesis-test-based Genetic Algorithms for Flow Shop Scheduling with Stochastic Processing Times." *International Journal of Advanced Manufacturing Technology* 25:1157–1163.

Wang, L., L. Zhang, and D. Z. Zheng. 2005b. "Genetic Ordinal Optimization for Stochastic Flow Shop Scheduling." *International Journal of Advanced Manufacturing Technology* 27:166–173.

Zhang, L. M., K. Li, and Y. Q. Zhang. 2010. "Green Task Scheduling Algorithms with Speeds Optimization on Heterogeneous Cloud Servers." In *Proceedings of the International Conference on Green Computing and Communications*, 76–80. Hangzhou, China.

Zhou, Q., and X. Cui. 2008. "Research on Multi-objective Flow Shop Scheduling with Stochastic Processing Times and Machine Breakdowns." In *Proceedings of International Conference on Service Operations and Logistics and Informatics*, 1718–1724. Beijing, China.

AUTHOR BIOGRAPHIES

GUILLEM CABRERA is a PhD candidate in the Distributed, Parallel and Collaborative Systems (DPCS) research group of the Internet Interdisciplinary Institute (IN3) at the Universitat Oberta de Catalunya (UOC). He holds an MSc in telecommunications from the Universitat Politècnica de Catalunya (UPC). His research interests include scalable and availability-aware distributed mechanisms for non-dedicated resources, peer-to-peer systems and optimization of large scale computing infrastructures. His e-mail is: gcabreraa@uoc.edu.

ANGEL A. JUAN is an Associate Professor of Applied Optimization, Simulation, and Data Analysis in the Computer Science Department at the Open University of Catalonia (UOC). He is also a Researcher at the Internet Interdisciplinary Institute. He holds a Ph.D. in Applied Computational Mathematics, a MSc in Information Systems & Technology, and a MSc in Applied Mathematics. His research interests include Applied Optimization and Simulation. Currently, he is the coordinator of the CYTED-IN3-HAROSA@IB Network and editorial board member of both the Int. J. of Data Analysis Techniques and Strategies and the Int. J. of Information Systems & Social Change. He is also a member of the INFORMS society. His website is: http://ajuanp.wordpress.com and his e-mail is ajuanp@uoc.edu.

HEBERT PÉREZ-ROSÉS obtained a PhD in Mathematics from the Universidad de Oriente, Cuba, in 2003, and a PhD in Computer Science from the University of Newcastle, Australia, in 2012. His research interests include graph theory and combinatorial algorithms, in particular algorithms on graphs. He is a member of MAA and ACM. His e-mail is: hebert.perez@gmail.com.

JOAN M. MARQUÈS is an Associate Professor at Computer Sciences, Multimedia & Telecommunication Studies at Universitat Oberta de Catalunya (UOC) since 1997 and currently is the co-

Director of the Distributed, Parallel and Collaborative Systems research group (DPSC). He graduated as a Computer Science Engineer from the Facultat d'Informàtica de Barcelona (UPC) in 1991 and received his PhD from UPC in 2003. His research interests include the design of scalable and cooperative Internet services and applications. He is member of the Association for Computing Machinery. His e-mail is: jmarquesp@uoc.edu.

JAVIER FAULIN is a Professor of Statistics and Operations Research at the Public University of Navarre (Pamplona, Spain). He received a PhD in Economics and a Master's in Applied Mathematics. His research interests include logistics, vehicle routing problems and simulation modeling and analysis. His e-mail is: javier.faulin@unavarra.es.