Proceedings of the 2018 Winter Simulation Conference
M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, eds.

AGENT-BASED SIMHEURISTICS: EXTENDING SIMULATION-OPTIMIZATION ALGORITHMS VIA DISTRIBUTED AND PARALLEL COMPUTING

Javier Panadero
Angel A. Juan
Jose M. Mozos

Universitat Oberta de Catalunya – IN3
Euncet Business School
08018 Barcelona, SPAIN

Canan Gunes Corlu

Metropolitan College
Boston University
Boston, MA USA

Bhakti Stephan Onggo

Trinity Business School
Trinity College Dublin
Dublin 2, IRELAND

ABSTRACT

This paper presents a novel agent-based simheuristic (ABSH) approach that combines simheuristic and multi-agent system to efficiently solve stochastic combinatorial optimization problems. In an ABSH approach, multiple agents cooperate in searching a near-optimal solution to a stochastic combinatorial optimization problem inside a vast space of feasible solutions. Each of these agents is a simheuristic algorithm integrating simulation within a metaheuristic optimization framework. Each agent follows a different pattern while exploring the solution space. However, all simheuristic agents cooperate in the search of a near-optimal solution by sharing critical information among them. The distributed nature of the multi-agent system makes it easy for ABSH to make use of parallel and distributed computing technology. This paper discusses the potential of this novel simulation-optimization approach and illustrates, with a computational experiment, the advantages that ABSH approaches offer over traditional simheuristic ones.

1 INTRODUCTION

Simulation with its capability of modeling complex systems involving uncertainty has become a popular tool for modeling real-life business processes. Today, we see simulations being used in several areas including but not limited to supply chain management, transportation and logistics, finance, telecommunication networks, and health care management. Typically, when an analytical model is not available to gain insights about how these systems and processes are operating, simulation is the tool of choice of practitioners in their decision-making process. Recent advances in computing hardware coupled with the offering of powerful software has made simulation even a more attractive method for analyzing complex systems under uncertainty (Lucas et al. 2015).

Despite the power of simulation in modeling complex systems, pure simulation should not be used alone to solve optimization problems. Instead, simulation-optimization methods are needed to obtain high-quality / near-optimal solutions to stochastic optimization problems. For instance, for a decision maker interested in developing an aggregate production plan that minimizes overload time, or a routing schedule that minimizes costs while adhering to some capacity constraints of the vehicles, optimization is the right

978-1-5386-6572-5/18/$31.00 ©2018 IEEE

tool of choice. It is important to note that most optimization models typically assume that the inputs and the constraints are deterministic and are known with certainty. This certainly simplifies the optimization problem to hand. However, this simplification comes at the expense of not being able to model the real-world process accurately. In the real-world, there are several uncertainties. For instance, for the decision maker developing an aggregate production plan that minimizes overload time, inputs such as customer demand and manufacturing times could be uncertain. Similarly, for the optimization problem that aims to find the routing schedule that minimizes costs, processing and traveling times could be uncertain. Failure to address the real-life uncertainty that characterizes these systems may lead to suboptimal decisions.

A close look at the literature reveals that it is actually possible to marry simulation and optimization tools (Coulouris et al. 2005). The so-called simulation-optimization (SO) methods are proposed to take the advantage of both methods simultaneously. In particular, SO methods can deal with optimization problems with stochastic components as well as simulation models with optimization requirements. One special class of SO methods is simheuristics (Juan et al. 2015), which is a promising approach for solving real-life stochastic combinatorial optimization problems (COPs) that are typically large-scale and NP-hard. As it is almost impossible to solve large-scale problems in reasonable computing times, traditionally, heuristics and metaheuristics have been employed to obtain near-optimal solutions in low computing times. The idea behind simheuristic algorithms is to integrate simulation methods into a metaheuristic optimization framework to deal with real life COPs. The use of simulation in a metaheuristic optimization framework allows one to address stochastic variables in the objective function and in the constraints of the optimization model. In other words, it is the simulation component that addresses the uncertainty in the model and that guides the metaheuristic component for a more efficient search. In addition, the simulation component allows one to perform risk analysis. Unlike deterministic COPs, where the focus is on finding the solution that minimizes costs or maximizes profits, in a stochastic COP, it is rarely enough to identify the solution that minimizes expected costs or maximizes expected profits. One would be interested in identifying some additional information such as the variance or the quantiles of each candidate solution to be able to make more informed decisions. As simulation component can provide these statistics, it also arises as a natural risk analysis tool.

We should also introduce distributed and parallel computing systems (DPCS), which have been utilized to solve complex COPs. In their most general form, DPCS aggregate multiple computing resources that work collaboratively to achieve a common objective (Coulouris et al. 2005). DPCS take different forms such as grid computing (Foster and Kesselman 2003), cloud computing (Armbrust et al. 2009), volunteer computing (Sarmenta 2001) and desktop grids (Cérin and Fedak 2012). Cloud computing allows users avoid large up-front investments on the resources and pay only for the resources they consume and thus is particulary attractive for small and medium enterprises (SMEs). Computing and desktop grids, on the other hand, work similarly by focusing on utilizing surplus or idle computing resources.

Multi-agent metaheuristic frameworks have been employed in the past to solve deterministic versions of COPs (Martin et al. 2016). Similarly, DPCS have been used in the past to solve stochastic COPs (Juan et al. 2013). Following these initiatives, the main contribution of this work is to discuss how simheuristic approaches can be extended into a more general approach that is called agent-based simheuristic (ABSH). In an ABSH approach, multiple agents cooperate in searching a near-optimal solution to a stochastic COP inside a vast space of feasible solutions. Each of these agents is a simheuristic algorithm integrating simulation within a metaheuristic optimization framework. Each agent follows a different pattern while exploring the solution space (e.g., by employing a different seed for the pseudo-random number generator, different parameter settings, or even different metaheuristic frameworks). However, all simheuristic agents cooperate in the search of a near-optimal solution by sharing critical information among them (e.g., by using a shared memory where recent 'discoveries' are registered).

The rest of the paper is organized as follows: Section 2 provides an overview of simheuristics algorithms; Section 3 proposes how these algorithms can be extended to an agent-based approach to increase their efficiency; Section 4 discusses the potential of DPCS-based approaches in solving real-life SME problems;

Section 5 introduces the stochastic COP that will be used in this paper to illustrate the use of our ABSH approach; Section 6 analyzes a numerical experiment that illustrates the benefits of using an ABSH approach over a classical simheuristic one. Finally, Section 7 highlights the main findings of our work and points out future research lines.

2 SIMHEURISTICS FUNDAMENTALS AND APPLICATIONS

As mentioned earlier, real-life COPs are usually large-scale problems involving some type of uncertainty. Therefore, simheuristics, which combine metaheuristics with simulation techniques, are particularly attractive to solve those problems. The simulation component addresses the stochastic nature of the model, which could be located in the objective function or in the set of constraints, while the metauheristic component deals with the optimization piece and focuses on identifying the "best" solution under the given set of constraints.

The underlying assumption behind a simheuristic approach is that promising solutions for a deterministic version of a COP are also likely to be good solutions for the stochastic version of the problem. It is important to note that this does not imply the deterministic version and the stochastic version share the same solution. The deterministic version of the COP only helps us to generate good solutions for the stochastic COP. Although this sounds like a simplifying assumption, our experience shows that indeed in most practical situations, this assumption is reasonable.

The typical approach in approximating a stochastic COP instance with its deterministic counterpart is to replace each random variable in the stochastic COP instance with its expected value. Notice that this approach is optimistic in the sense that it does not consider any variability around the random variables. Nevertheless, it is an easy way of obtaining the deterministic COP instance from a real-life stochastic COP instance. The next step is to solve the deterministic optimization problem. This is usually done by a metaheuristic-driven algorithm, which performs an (efficient) search in the solution space of the deterministic problem. This process continues iteratively until high-quality feasible solutions are identified for the deterministic optimization problem. This process is summarized in Figure 1.After some promising solutions are identified for the deterministic COP, these solutions are fed into the simulation. One important capability of the simulation model is its ability to model the stochastic variables with general theoretical probability distributions such as Normal distribution, Gamma distribution, and Beta distribution or even with an empirical distribution. The simulation runs are performed and the estimated values provided by the simulation are recorded. These solutions are considered as candidate solutions for the stochastic COP. Recall that the simulation component and the metaheuristic component work iteratively at this stage. Thus, the candidate solutions provided by the simulation serve as a feedback mechanism for the metaheuristic component. Usually, we set an upper bound for the computational time that can be consumed by the iterative process and the search stops when this upper bound is reached and a set of good solutions are provided.

As discussed earlier, the objective function of stochastic COPs usually includes randomness and the goal of the stochastic COP is to obtain a solution that maximizes (minimizes) the expected value of the objective function. However, this is usually not enough information for the decision maker, who might be also interested in some quartiles of the objective function. For instance, in addition to the expected profit, the decision maker may want to know about the probability of making a profit greater than a specific value. In order to compute this probability, the decision-maker needs to have access to the probability distribution of the values generated by several alternative solutions. An important aspect of the simheuristic method is its capability of providing these probability distributions (with the help of the simulation component) thus introducing risk analysis criteria in the decision-making process.

We conclude this section by providing a brief review of the literature (in the chronological order) on the use of simheuristics to solve stochastic COPs that arise in different fields. We note that our review is not exhaustive and we only discuss some notable applications here. We refer the reader to Table 1 for a summary of the related literature. The first application of simheuristics was in the area of vehicle routing

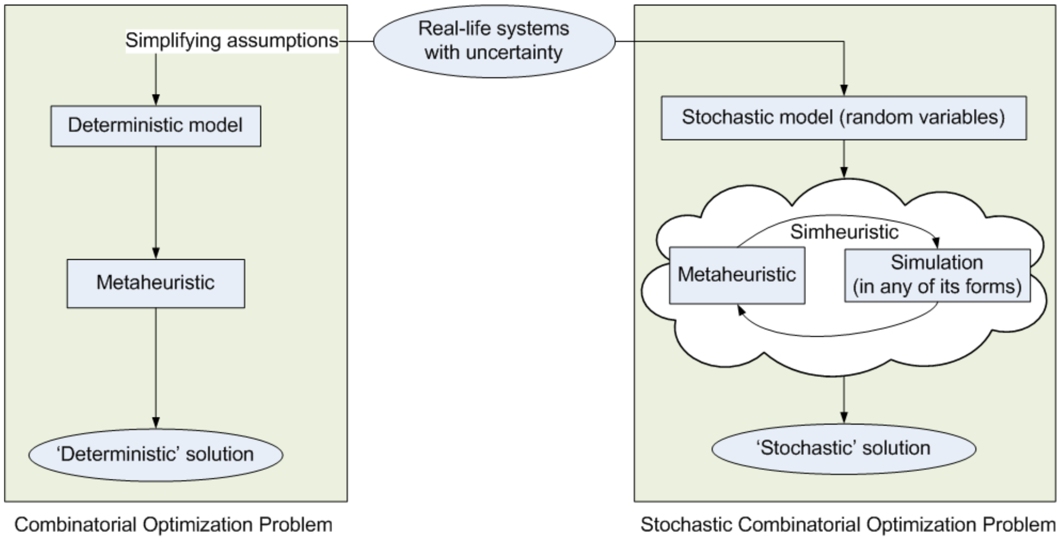

Figure 1: Solving deterministic vs. stochastic COPs.

problems (VRP). Juan et al. (2011) consider a vehicle-routing problem with stochastic demands and design a basic simheuristic approach to solve this problem. The authors continue the study of the stochastic VRP in Juan et al. (2013) with the analysis of the parallel and distributed computing techniques to efficiently solve this problem. Juan et al. (2014b) consider the stochastic version of an inventory routing problem with stock-outs and design a simheuristic to solve this problem. Juan et al. (2014a), on the other hand, propose a simheuristic algorithm for solving the permutation flow-shop problem with stochastic processing times. An example of simheuristic application for solving the arc-routing problem with stochastic demands is discussed in Gonzalez-Martin et al. (2018). Another interesting application of the simheuristic approach is in the area of waste collection management. Gruler et al. (2017b) consider the stochastic waste-collection problem with a single-depot, while Gruler et al. (2017a) extend the analysis to the multiple-depot case. De Armas et al. (2017) analyze the stochastic uncapacitated facility location problem with a simheuristic approach. Pagès-Bernaus et al. (2017) consider the problem of designing e-commerce supply chains and propose a simheuristic approach for the stochastic capacitated facility location problem that arises in this context. Very recently, Gruler et al. (2018) tackle a stochastic inventory routing problem with a variable neighborhood search simheuristic and Panadero et al. (2018) use the same approach in the context of selecting a portfolio of projects under uncertainty. We contribute to this literature by offering an agent-based simheuristic that employs distributed and parallel computing techniques to efficiently solve stochastic COP instances. We illustrate our approach in the context of a stochatic team orienteering problem discussed in Section 5.

3 AGENT-BASED SIMHEURISTICS

We propose agent-based simheuristics as a multi-agent extension of the simheuristic concept reviewed in the previous section. In agent-based simulation, systems are modeled as a set of autonomous agents that interact among them in order to achieve a common goal. Similarly, in ABSH, each agent is an autonomous and differentiated simheuristic algorithm that interacts with the rest of the agents while searching for a near-optimal solution to a complex and stochastic COP (Figure 2). Similar to the way multi-agent system benefits from distributed and parallel computing systems (Macal and North 2010), the distributed and autonomous nature of agents in ABSH makes it a good candidate for executing it on a distributed and parallel computing platform.

Table 1: Application of simheuristic approaches in different areas.

Application/Problem	Paper
Stochastic vehicle-routing problem	Juan et al. (2011); Juan et al. (2013)
Stochastic inventory-routing with stock-outs	Juan et al. (2014b)
Stochastic multi-period inventory routing	Gruler et al. (2018)
Stochastic permutation flow-shop problem	Juan et al. (2014a)
Distributed computer networks	Cabrera et al. (2014)
Stochastic arc-routing problem	Gonzalez-Martin et al. (2018)
Stochastic waste collection	Gruler et al. (2017a), Gruler et al. (2017b)
Stochastic uncapacitated facility location	De Armas et al. (2017)
Stochastic capacitated facility location	Pagès-Bernaus et al. (2017)
Stochastic project portfolio selection	Panadero et al. (2018)

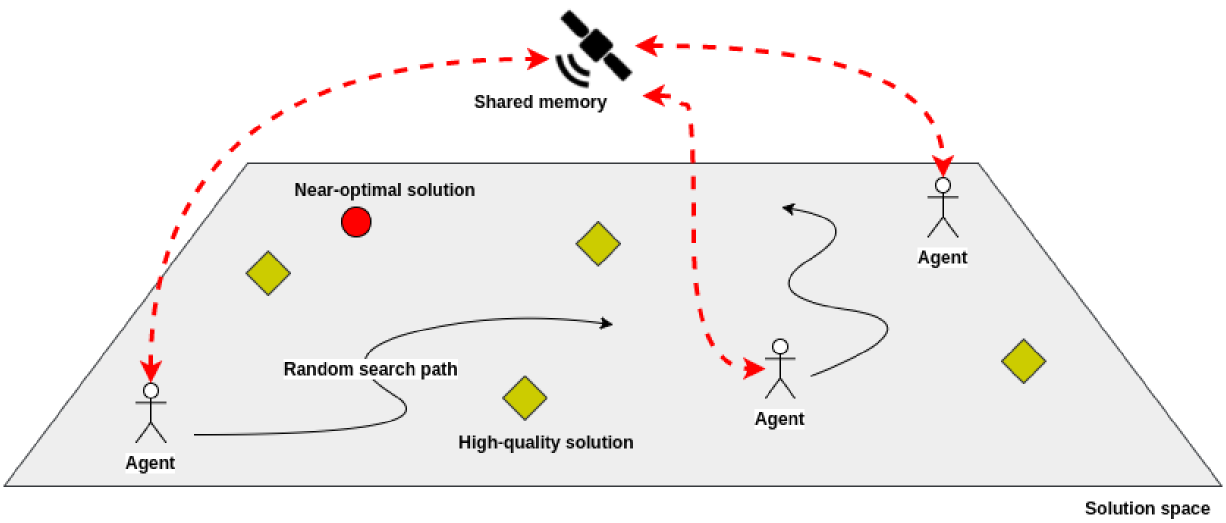

Figure 2: Scheme of an agent-based simheuristic approach.

Thus, in an ABSH environment, a number of agents cooperate among them to complete a more effective search of the vast solution space. Each of these agents is an autonomous simheuristic algorithm integrating simulation within a differentiated metaheuristic optimization framework. Accordingly, each agent follows a distinct pattern while exploring the solution space. This diversified behavior could be achieved by employing different strategies, e.g.: *(i)* assigning a different seed to the pseudo-random number generator that the metaheuristic component utilizes to introduce a random behavior into its searching process; *(ii)* in the case of metaheuritics employing biased-randomized techniques (Grasas et al. 2017), using different skewed probability distributions (e.g., geometric, decreasing triangular, etc.) to introduce some bias in the searching process; *(iii)* assigning different configurations to the parameters that characterize the metaheuristic component in each agent, including those affecting the local search, perturbation, and acceptance criterion stages; *(iv)* using different metaheuristic frameworks for each agent, e.g., while some agents can use an iterated local search framework (Lourenço et al. 2010), others can use a variable neighborhood search (Hansen and Mladenović 2014), a greedy randomized adaptive search procedure (Festa and Resende 2002), a tabu search (Glover and Laguna 2013), or any other similar metaheuristic framework; and *(v)* any combination of the above. Despite each agent being autonomous, all agents cooperate by sharing relevant information among them during the searching process. In particular, agents can: *(i)* make use of a quick-access shared memory (e.g., a hash map) where they register high-quality components for a potential solution, such as the best way found so far to connect a given set of customers in the case of a routing problem; or *(ii)* share a new base solution that improves the existing ones and that can be used by a user-defined percentage of the agents to update their base searching position.

Notice that ABSH algorithms can be used in solving all the stochastic COPs where single-agent simheuristics have been used already. The main difference, however, is that ABSH algorithms benefit from distributed and parallel computing systems as well as from the cooperation among agents, which might reduce the computation times requested to reach near-optimal solutions in the case of complex, large-scale, and stochastic COPs. This might be an important contribution specially in sectors such as logistics and transport systems in smart cities or telecommunication systems, where decisions need to be made in very short time periods.

4 POTENTIAL FOR DISTRIBUTED AND PARALLEL COMPUTING SYSTEM

The distributed and autonomous nature of agents in ABSH means that agents can work independently but with regular communications between them to share information as explained in the previous section. Hence, ABSH is a good candidate for parallelization. We can use suitable distributed and parallel computing system such as cloud computing, grid computing, computer cluster, or distributed computer. In our research, we are interested in helping small and medium enterprises because they generate a critical piece of riches in all developed economies. However, these enterprises usually lack computational resources despite the fact that a significant number of them does need these resources to be able to solve their business problems that involve simulation-optimization.

DPCS offer two main alternatives to the computing needs of the SMEs. The first alternative for the SME is to use the resources provided by an external source. These resources could be in the form of virtual machines that provide a cloud platform for the company. The second and perhaps more trivial alternative for the SME is to use its underutilized computing resources (i.e. computer or desktop grid). This approach is particularly attractive to the SME due to the issues related to costs, information security, and energy consumption. It is cost-effective as the company is using its own resources and does not need to pay to a third party for resources. It is more secure because the company does not have to share any information with an external source. Finally, it is environmentally friendly as it consumes less energy (Cabrera et al. 2014).

A natural question to ask is then how the company can actually form these desktop grid systems and use its underutilized resources. The first task would be to identify computers with more computing power or computers that are idle during some days or parts of the day. The idea is to aggregate computational resources from these different computers that form a network and run simultaneously thousands of instances of a simulation-optimization algorithm. With the availability of more resources, more instances will be executed simultaneously and this will reduce the computational time needed to identify near-optimal solutions. It is important to note that this concept has been used successfully in several realistic applications. We refer the reader to (Lázaro et al. 2012) for an example application.

5 THE STOCHASTIC TEAM ORIENTEERING PROBLEM

The team orienteering problem (TOP) is a variant of the well-known vehicle routing problem in which a set of vehicle tours are constructed in such in a way that: *(i)* the total collected reward received from visiting a subset of customers is maximized; and *(ii)* the length of each vehicle tour is restricted by a pre-specified limit (Figure 3).

While most existing works refer to the deterministic version of the problem and focus on maximizing total reward, some degree of uncertainty (e.g., in customers' service times or in travel times) should be expected in real-life applications. Accordingly, some authors have proposed a simheuristic algorithm for solving the TOP with stochastic travel times (Panadero et al. 2017). These authors consider the problem of a company that carries out repairs. The company employs *m* repair people who are each paid to work

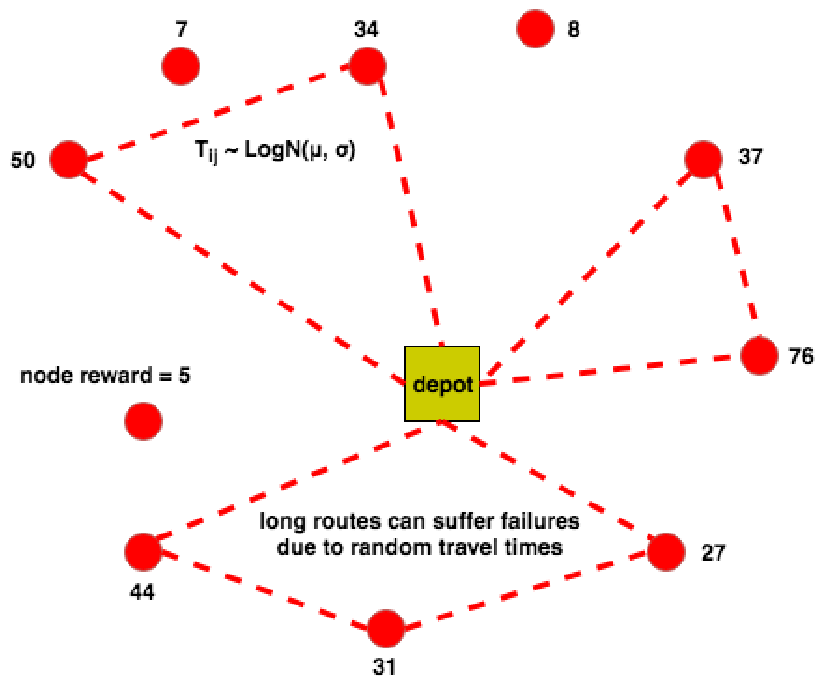

Figure 3: The team orienteering problem with stochastic travel times.

for T hours per day. The repairs are spread out over a geographical area and the repair people must travel to each site to work on a job. At the start of each day, the company receives a list of repair jobs that it has been asked to carry out, and it needs to make decisions over which of these jobs are accepted and how to assign them to its staff. Each job has an associated reward value, which is known in advance. It is assumed that the travel time between each pair of jobs is a random variable following a known distribution. This means that standard methods for deterministic problems are not suitable, and a simulation-optimization approach might be necessary to account for the system randomness.

In a stochastic TOP, several statistical properties of the generated solution should be considered apart from its associated expected reward. In effect, in a stochastic environment one could be interested in solutions that offer high reliability or robustness in terms of the number of times that the threshold is violated or the number of jobs served out of time, i.e.: one solution (distribution plan) A would be more robust than other B if, and only if, A shows a better behavior than B when both are considered under a stochastic scenario. In other words, apart from measuring the cost of each solution, it is necessary to analyze how well each solution can support uncertainty conditions without degenerating in other properties such as reliability or robustness.

6 COMPUTATIONAL EXPERIMENTS

With the objective of analyzing how an agent-based approach affects both the quality of the solutions provided by our algorithm and the associated computational time, we performed an experiment using a subset of benchmark instances proposed in Chao et al. (1996), which are available in the repository (https://www.mech.kuleuven.be/en/cib/op/instances). We have selected this benchmark for our experimentation, because is widely used in the literature. The instances that compose the benchmark are divided into seven different sets as a function of the number of customers. Each instance in the benchmark is identified following the nomenclature *pX.Y.Z*, where X is the number of sets; Y is the total number of vehicles; and Z identifies the maximum allowed length for the vehicle tours. These instances are deterministic, so we extended them to stochastic ones by employing Log-Normal distributions to model travel times, as described in Panadero et al. (2017). The experiment was aimed at answering the following question: "for each instance, which is the best solution that our algorithm can provide in a reasonable time (less than 1 minute) using different scales of agents?".

To answer the previous question, we designed a simple multi-agent model in which each agent was an instance of our algorithm. Each agent used a different seed for the pseudo-random number generator. This made each agent explore different locations in the solution space. At this early stage, we have not addressed issues related to the synchronization and communications between agents. The solutions depend on the distributed platform used and the characteristics of the stochastic COP. In our experimental setting, all agents share a cache memory mechanism (Juan et al. 2011), which records the best-found-so-far routes. This mechanism is implemented using a hash-map data structure, which is constantly updated by all the agents whenever a better route is found. We also had not applied any fine-tuning to optimize the parallel execution of the agents, since our goal was to prove that our algorithm is robust and can provide efficient solutions to any STOP problem without any initial adjustments. We ran our experiment on a multi-core processor Intel Xeon E5-2650 v4 with 32GB RAM. This platform represents a simple setting in a small enterprise.

Table 2 shows the computational results obtained in the previously described experiment. The column 1 shows the obtained reward. The remaining columns show the time required to reach this reward using different number of agents. The results show that the time required to obtain the reward is reduced, for all the considered instances, as the number of parallel agents is increased. Figure 4 depicts a visual representation of the speedup as the number of parallel agents is increased. As is shown in this figure, as the number of parallel agents increases, the speedup also increases. Focusing on the maximum number of parallel agents used in this experiment (8), we obtain a minimum speedup of 3.76 for the instance p6.2.n, and a maximum speedup of 8.24 for the instance p3.3.o, making clear the benefits of using paralell agents.

Table 2: Reward obtained for a subset of TOP instances, and the computational time needed to reach the reward using 1,2,4 and 8 agents.

Instance	Reward [1]	1 Agent (Serial exec.) (sec.) [2]	2 Parallel Agents (sec.) [3]	4 Paralell Agents (sec.) [4]	8 Parallel Agents (sec.) [5]
p3.2.s	800	2.61	2.13	1.46	0.35
p3.3.o	580	47.92	31.44	26.86	5.81
p5.2.p	1095	59.07	41.11	25.90	11.86
p5.2.z	1545	40.97	32.95	26.35	7.42
p6.2.n	1218	100.30	74.25	43.78	26.62
p6.3.k	828	48.16	32.75	28.66	11.74
p6.4.n	954	120.00	66.70	50.98	20.51

Finally, Figure 5 shows a 3D representation for one instance (i.e. p6.4.n). This figure shows the evolution of the solution quality as we vary both the computing time and the number of agents. As expected, an increase in the number of agents increases the quality of the solution. The increase in the computation time increases the solution quality.

7 CONCLUSIONS

Simulation with its capability of mimicking complex systems has been a very popular approach in recent years to solve real-life problems involving uncertainty. However, simulation itself is not a technique to be used for optimization purposes. For example, simulation alone cannot be used to solve combinatorial optimization problems. Optimization models, which are designed to find the "best" solution to a problem, on the other hand, usually work under the assumption that all inputs are deterministic and thus lack the ability to incorporate random inputs into the solution process. For instance, a combinatorial optimization

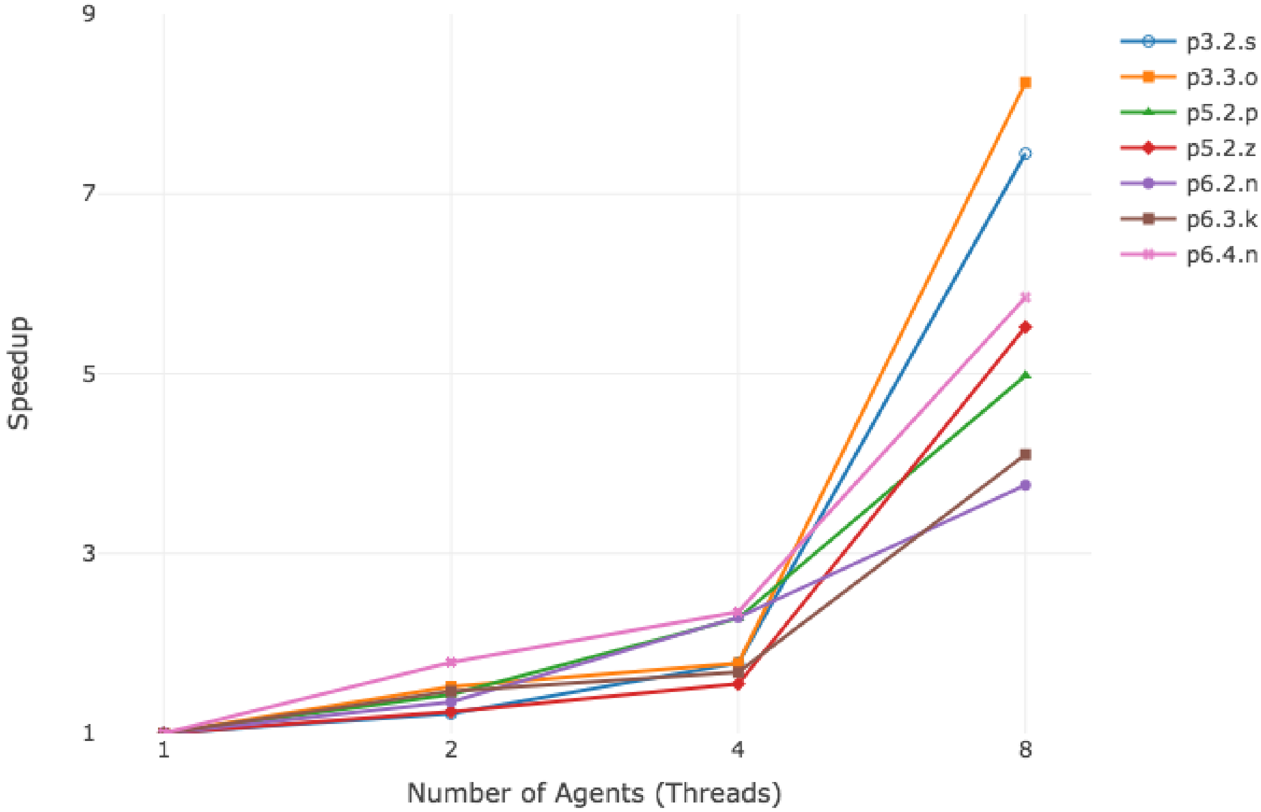

Figure 4: Speedup of each instance as the number of parallel agents increases.

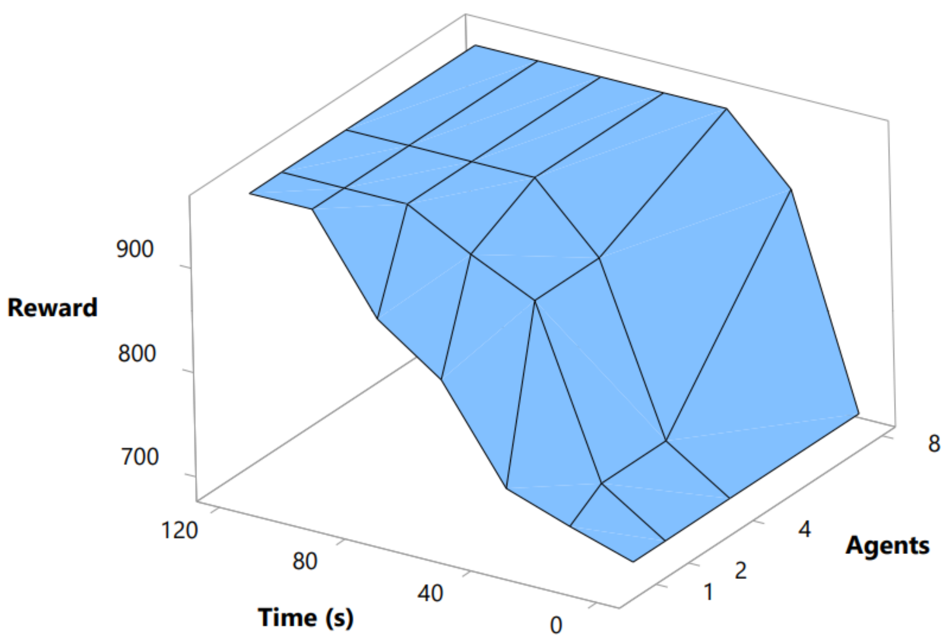

Figure 5: Surface plot for expected reward vs. time and number of agents for the instance p6.4.n.

problem involving uncertainty cannot be solved with a pure deterministic optimization algorithm. In other words, neither simulation nor optimization alone is enough to address stochastic combinatorial optimization problems. Hybrid approaches such as simheuristics have been proposed to combine the power of simulation and optimization to solve real-life optimization problems under uncertainty. It is no surprise that these approaches have already been used in several areas such as transportation, logistics, supply chain management, and telecommunication networks, and are gaining more importance as the real-life systems are becoming more complex.

This paper has also demonstrated the benefit of distributed and parallel computing techniques to agent-based simheuristics, in which each agent implements a different simheuristic setting, and all agents cooperate among them by sharing critical information during the exploration of the solution space. These concepts have been tested in solving a popular stochastic combinatorial optimization problem, and the benefits of using an agent-based simheuristic approach over using a traditional one have been analyzed. In addition, we have discussed the potential benefits of distributed and parallel computing systems for small and medium enterprises, which often lack advanced technical skills and modern equipment.

Being an early work, many research lines remain open to be explored. Among them: *(i)* to test agent-based simheuristic approaches in solving stochastic combinatorial optimization problems in the fields of logistics, transportation, telecommunication networks, production, or finance; *(ii)* to develop cooperation protocols among the multiple agents that increase the effectiveness of the proposed agent-based simheuristic approach under various distributed and parallel computing platforms.

ACKNOWLEDGEMENTS

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness and FEDER (TRA2015-71883-REDT), and the Erasmus+ programme (2016-1-ES01-KA108-023465).

REFERENCES

Armbrust, M., A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica et al. 2009. “Above the Clouds: a Berkeley view of Cloud Computing”. Technical report, Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley.

Cabrera, G., A. A. Juan, D. Lázaro, J. M. Marquès, and I. Proskurnia. 2014. “A Simulation-Optimization Approach to deploy Internet Services in Large-scale Systems with User-provided Resources”. *Simulation* 90 (6): 644–659.

Cérin, C., and G. Fedak. 2012. *Desktop Grid Computing*. Boca Raton, FL, USA: CRC Press.

Chao, I.-M., B. Golden, and E. Wasil. 1996. “The Team Orienteering Problem”. *European Journal of Operational Research* 88:464–474.

Coulouris, G. F., J. Dollimore, and T. Kindberg. 2005. *Distributed Systems: Concepts and Design*. Harlow, UK: Pearson Education.

De Armas, J., A. A. Juan, J. M. Marquès, and J. P. Pedroso. 2017. “Solving the Deterministic and Stochastic Uncapacitated Facility Location Problem: from a Heuristic to a Simheuristic”. *Journal of the Operational Research Society* 68 (10): 1161–1176.

Festa, P., and M. G. Resende. 2002. “GRASP: an Annotated Bibliography”. In *Essays and Surveys in Metaheuristics*, edited by C. C. Ribeiro and P. Hansen, 325–367. New York, NY, USA: Springer.

Foster, I., and C. Kesselman. 2003. *The Grid 2: Blueprint for a new Computing Infrastructure*. San Francisco, CA, USA: Elsevier.

Glover, F., and M. Laguna. 2013. “Tabu Search”. In *Handbook of Combinatorial Optimization*, edited by P. M. Pardalos et al., 3261–3362. New York, NY, USA: Springer.

Gonzalez-Martin, S., A. A. Juan, D. Riera, M. Elizondo, and J. Ramos. 2018. “A Simheuristic Algorithm for solving the Arc-Routing Problem with Stochastic Demands”. *Journal of Simulation* 12 (1): 53–66.

Grasas, A., A. A. Juan, J. Faulin, J. de Armas, and H. Ramalhinho. 2017. “Biased Randomization of Heuristics using Skewed Probability Distributions: a Survey and some Applications”. *Computers & Industrial Engineering* 110:216–228.

Gruler, A., C. Fikar, A. A. Juan, P. Hirsch, and C. Contreras-Bolton. 2017a. “Supporting Multi-Depot and Stochastic Waste Collection Management in Clustered Urban Areas via Simulation-Optimization”. *Journal of Simulation* 11 (1): 11–19.

Gruler, A., J. Panadero, J. DeArmas, J. A. Moreno-Perez, and A. A. Juan. 2018. “A Variable Neighborhood Search based Simheuristic for the Inventory Routing Problem with Stochastic Demands”. *Computers & Industrial Engineering* 123:278–288.

Gruler, A., C. L. Quintero-Araújo, L. Calvet, and A. A. Juan. 2017b. “Waste Collection under Uncertainty: a Simheuristic based on Variable Neighbourhood Search”. *European Journal of Industrial Engineering* 11 (2): 228–255.

Hansen, P., and N. Mladenović. 2014. “Variable Neighborhood Search”. In *Search Methodologies*, edited by E. K. Burke and G. Kendall, 313–337. New York, NY, USA: Springer.

Juan, A. A., B. B. Barrios, E. Vallada, D. Riera, and J. Jorba. 2014a. “A Simheuristic Algorithm for solving the Permutation Flow Shop Problem with Stochastic Processing Times”. *Simulation Modelling Practice and Theory* 46:101–117.

Juan, A. A., J. Faulin, S. Grasman, D. Riera, J. Marull, and C. Mendez. 2011. “Using Safety Stocks and Simulation to solve the Vehicle Routing Problem with Stochastic Demands”. *Transportation Research Part C: Emerging Technologies* 19 (5): 751–765.

Juan, A. A., J. Faulin, S. E. Grasman, M. Rabe, and G. Figueira. 2015. “A Review of Simheuristics: extending Metaheuristics to deal with Stochastic Combinatorial Optimization Problems”. *Operations Research Perspectives* 2:62–72.

Juan, A. A., J. Faulin, J. Jorba, J. Caceres, and J. M. Marquès. 2013. “Using Parallel & Distributed Computing for Real-Time solving of Vehicle Routing Problems with Stochastic Demands”. *Annals of Operations Research* 207 (1): 43–65.

Juan, A. A., J. Faulin, J. Jorba, D. Riera, D. Masip, and B. Barrios. 2011. “On the use of Monte Carlo simulation, cache and splitting techniques to improve the Clarke and Wright savings heuristics”. *Journal of the Operational Research Society* 62 (6): 1085–1097.

Juan, A. A., S. E. Grasman, J. Caceres, and T. Bektaş. 2014b. “A Simheuristic Algorithm for the Single-Period Stochastic Inventory-Routing Problem with Stock-Outs”. *Simulation Modelling Practice and Theory* 46:40–52.

Lázaro, D., D. Kondo, and J. M. Marquès. 2012. “Long-Term Availability Prediction for Groups of Volunteer Resources”. *Journal of Parallel and Distributed Computing* 72 (2): 281–296.

Lourenço, H. R., O. C. Martin, and T. Stützle. 2010. “Iterated Local Search: Framework and Applications”. In *Handbook of Metaheuristics*, edited by M. Gendreau and J.-Y. Potvin, 363–397. New York, NY, USA: Springer.

Lucas, T. W., W. D. Kelton, P. J. Sanchez, S. M. Sanchez, and B. L. Anderson. 2015. “Changing the Paradigm: Simulation, now a Method of First Resort”. *Naval Research Logistics* 62 (4): 293–303.

Macal, C. M., and M. J. North. 2010. “Tutorial on Agent-Based Modelling and Simulation”. *Journal of Simulation* 4 (3): 151–162.

Martin, S., D. Ouelhadj, P. Beullens, E. Ozcan, A. A. Juan, and E. K. Burke. 2016. “A Multi-Agent Based Cooperative Approach to Scheduling and Routing”. *European Journal of Operational Research* 254 (1): 169–178.

Pagès-Bernaus, A., H. Ramalhinho, A. A. Juan, and L. Calvet. 2017. “Designing E-Commerce Supply Chains: a Stochastic FacilityLocation Approach”. *International Transactions in Operational Research* doi:10.1111/itor.12433.

Panadero, J., J. DeArmas, C. Currie, and A. A. Juan. 2017. "A Simheuristic to solve the Stochastic Team Orienteering Problem". In *Proceedings of the 2017 Winter Simulation Conference*, edited by W. K. V. Chan et al., 3208–3217. Piscataway, New Jersey: IEEE.

Panadero, J., J. Doering, R. Kizys, A. A. Juan, and A. Fito. 2018. "A Variable Neighborhood Search Simheuristic for Project Portfolio Selection under Uncertainty". *Journal of Heuristics* doi:10.1007/s10732-018-9367-z.

Sarmenta, L. F. G. 2001. *Volunteer Computing*. Ph. D. thesis, Massachusetts Institute of Technology.

AUTHOR BIOGRAPHIES

JAVIER PANADERO is an Assistant Professor of Simulation and High Performance Computing in the Computer Science, Multimedia and Telecommunication Department at the Universitat Oberta de Catalunya (Barcelona, Spain). He is also a post-doctoral researcher at the ICSO@IN3 group. He holds a Ph.D. and a M.S. in Computer Science. His major research areas are: high performance computing, modeling and analysis of parallel applications, and simheuristics. He has co-authored more than 18 articles published in journals and conference proceedings. His website address is http://www.javierpanadero.com and his email address is jpanaderom@uoc.edu.

ANGEL A. JUAN is a Full Professor of Operations Research & Industrial Engineering in the Computer Science Dept. at the Universitat Oberta de Catalunya. He is also the coordinator of the ICSO research group at the IN3. Dr. Juan holds a Ph.D. in Industrial Engineering and an M.S. in Mathematics. He completed a predoctoral internship at Harvard University and postdoctoral internships at Massachusetts Institute of Technology and Georgia Institute of Technology. His main research interests include applied optimization and simulation (metaheuristics and simheuristics) in computational transportation & logistics, Internet computing, and computational finance. He has published 65+ articles in JCR-indexed journals and 170+ documents indexed in Scopus. His website address is http://ajuanp.wordpress.com and his email address is ajuanp@uoc.edu.

CANAN GUNES CORLU is an Assistant Professor of Administrative Sciences Department of Metropolitan College at Boston University. She holds a Ph.D. in Operations Management from the Tepper School of Business at Carnegie Mellon University. Her research interest is in the area of the design and analysis of stochastic simulations with applications in operations management. She has published more than 20 articles in journals and conference proceedings. Her recent work on the simulation of inventory systems with unknown input models has been selected as a finalist for INFORMS Minority Issues Forum Best Paper Competition. Her email address is canan@bu.edu.

JOSE M. MOZOS is a PhD student at IN3 – Universitat Oberta de Catalunya (Barcelona, Spain). He holds a BSc in Telecommunication Engineering and a MS in Computational Engineering and Mathematics. His work is related to metaheuristic and simheuristic algorithms for solving combinatorial optimization problems. His email address is josemiguelmozosruiz@gmail.com.

BHAKTI STEPHAN ONGGO is an Associate Professor of Data Analytics at Trinity Business School, Trinity College Dublin, Ireland. His research interests lie in the areas of predictive analytics using simulation (symbiotic simulation, hybrid modelling, agent-based simulation, discrete-event simulation) with applications in operations and supply chain management (e.g. hospital, manufacturing, transportation) and social dynamics (e.g. social risk). He is the associate editor for the Journal of Simulation and the secretary of The OR Societys Simulation SIG. His email address isstephan.onggo@tcd.ie.