# Why Simheuristics? Benefits, Limitations, and Best Practices when Combining Metaheuristics with Simulation

Manuel Chica*[,a], Angel A. Juan[a], Óscar Cordón[b], W. David Kelton[c]

[a] IN3 - Department of Computer Science, Open University of Catalonia, 08018 Barcelona, Spain
[b] DECSAI and CITIC-UGR, University of Granada, 18071 Granada, Spain
[c] Department of Operations, Business Analytics, and Information Systems, University of Cincinnati, Cincinnati, Ohio 45221-0130, USA; and Department of Operations Research, Naval Postgraduate School, Monterey, California 93943-5219, USA

## Abstract

From smart cities to factories and business, many decision-making processes in our society involve NP-hard optimization problems. In a real environment, these problems are frequently large-scale, which limits the potential of exact optimization methods and justifies the use of metaheuristic algorithms in their resolution. Real-world problems are also distinguished by high levels of dynamism and uncertainty, which affect the formulation of the optimization model, its input data, and constraints. However, metaheuristic algorithms usually assume deterministic inputs and constraints, and thus end up solving oversimplified models of the real system being considered, casting doubt on validity and even meaning of the results and recommendations. Accordingly, this paper argues that approaches combining simulation with metaheuristics, i.e., *simheuristics*, not only constitute a natural extension of metaheuristics, but also should be considered as a "first-resort" method when dealing with large-scale stochastic optimization problems, which constitute most realistic problems in industry and business. To this end, this paper highlights the main benefits and limitations of these simheuristic algorithms, reviews some examples of applications to different fields, and analyzes the most suitable simulation paradigms to be used within a simheuristic. Finally, we outline a series of best practices to consider during the design and implementation stages of a simheuristic algorithm.

*Key words:* Optimization, Simheuristics, Metaheuristics, Simulation, Uncertainty

*Corresponding author
*Email addresses:* manuel.chica.serrano@gmail.com (Manuel Chica), ajuanp@uoc.edu (Angel A. Juan), ocordon@decsai.ugr.es (Óscar Cordón), david.kelton@uc.edu (W. David Kelton)

## 1. Introduction

Decision makers in areas such as transportation, logistics and supply-chain management, health care, production, telecommunications, and finance have to face complex challenges when tackling optimization problems in real-world systems. Most of these optimization problems are NP-hard, while others have a lack of complete information that makes their exact definition or formulation impossible (Talbi 2009). These facts limit the use of exact methods to small- and medium-sized instances, in which the optimal values can be obtained in reasonable computing times. Moreover, they require use of simplified models that sometimes do not properly reflect the actual system characteristics, casting doubt in terms of validity of the results. However, driven by economic and technological factors, real-world systems are becoming increasingly large and complex. Among these factors, we could include trends such as globalization, increased computing power, information technologies and the Internet, as well as the availability of vast amounts of data (Lucas et al. 2015).

Metaheuristic algorithms have gained popularity as the predominant approach for solving real-world optimization problems (Glover and Kochenberger 2006). These algorithms are able to deal with non-trivial objective functions (e.g., multi-objective, non-convex, non-smooth, noisy functions), complex constraints, and decision variables of different natures. Metaheuristics allow decision makers to obtain near-optimal solutions to large and complex problems in reasonably low computing times, sometimes even in “real time” (e.g., a few seconds). Therefore, they have become effective methodologies in application areas where optimization of system resources is needed. In addition, approaches hybridizing exact methods with metaheuristics are also widely used. For instance, matheuristics (Boschetti et al. 2009) combine both approaches to get the best from each of them. Typically, they employ the metaheuristic component to deal with the large global problem, while the exact component is used to cope with specific parts of it.

Nonetheless, both exact optimization methods and metaheuristics frequently assume that the problem inputs, the underlying objective functions, and the set of optimization constraints are deterministic and known. These are strong assumptions and, as a consequence, most deterministic or stochastic optimization models are oversimplified versions of real-world systems. In other words, we end up obtaining the optimal solution for the “wrong” model, i.e., a model that might not be a sufficiently accurate representation of the real system. Coping with the inherent uncertainty of the systems to optimize during problem solving has recently gained relevance and, for instance, robust approaches for metaheuristics have been proposed to handle such uncertainty (Beyer and Sendhoff 2007). Most of these approaches are extensions of extant optimization models, and they can be classified as deterministic (i.e., based on a set of plausible scenarios), probabilistic (i.e., assuming a given probabilistic function), or possibilistic (i.e., fuzzy-interval measures).

In general, simulation modeling is the best approach for dealing with complex systems under uncertainty (Borshchev and Filippov 2004). Simulation allows us to represent the real system in detail and can maintain better control over experimental conditions than we can by experimenting with the real system itself (Spall 2005). A valid simulation model can capture system variation in a realistic way while still producing results that can be made as precise as desired, while still faithfully representing the existing complexity. An approximate estimate from a valid (simulation) model is usually preferable to an exact result from an invalid (analytical) model. In the former case we can estimate its imprecision and, if needed, reduce it via more computing. But in the latter case we generally have no way of knowing "how wrong" the results from an overly simplified or perhaps even stylized analytical model might be. A well validated simulation should be the method of choice also for modeling uncertainty in real-world complex optimization problems. As Lucas et al. (2015) noted, "simulation is now an option that should be, in many ways, regarded as the method of choice for analyzing complex systems in the face of astounding advances in affordable processing power, modeling paradigms and tools, and supporting analysis capabilities."

Hybridization of simulation techniques with metaheuristics can provide practitioners with a flexible and rich tool when dealing with optimization problems in uncertain domains. We call this hybridization *simheuristics* (Juan et al. 2015). When properly designed, simheuristics are capable of solving real-world optimization problems where the simulation component deals with the uncertainty of the model and interacts with the metaheuristic component that, in turn, searches the solution space for a near-optimal solution. In the past, some optimization problems have been solved by using simulation to evaluate the quality of solutions in engineering (Talbi 2009). Notice, however, that simheuristic algorithms go one step beyond in the sense that: (*i*) the feedback from the simulation should also be used to guide the metaheuristic search process itself; and (*ii*) all the information about a stochastic solution provided by the simulation component allows introduction of risk or reliability analysis criteria during the assessment of alternative high-quality solutions to the optimization problem. All these characteristics, plus the fact that integration of simulation techniques with metaheuristic algorithms is relatively simple, make simheuristics a first-resort (Lucas et al. 2015) method when dealing with real-world optimization problems.

In this paper we analyze some of the advantages of using simheuristics over traditional methods, as well as some limitations. These advantages range from a better understanding of the system behavior, to the use of the generated information through the simheuristic. For example, visualization, machine learning, and sensitivity analysis can be easily used to obtain richer information automatically about the optimization process. We also describe how this combination of metaheuristics and simulation can be carried out to build a successful simheuristic. Several construction guidelines are given to help researchers and practitioners reach their goals. Thus, for instance, validation and stakeholders' discussion of the simulation model used within the simheuristic are encouraged. As simulation can tolerate far less restrictive modeling assumptions, even simple simulations must be correctly validated (Sargent 2005, Oliva 2003, Chica et al. 2017) and agreed to by as many decision makers as possible, in order to lead to better decisions (Voinov and Bousquet 2010). These guidelines promote use of different stages to avoid jeopardizing the optimization process itself, and obtain the best possible results with reduced computing time.

Section 2 provides a short overview of metaheuristic algorithms and justifies their use in certain optimization problems. Section 3 reviews how uncertainty has been traditionally addressed in optimization problems. Section 4 reviews the basic concepts behind a simheuristic approach. Section 5 lists its most important advantages. Correspondingly, Section 6 analyzes the main limitations of this method and how they can be partially addressed. Section 7 reviews the main simulation paradigms that can be potentially used to extend a metaheuristic approach into a simheuristic one. Section 8 provides some guidelines that can be useful during the design and implementation stages of a simheuristic algorithm. Concluding remarks are in Section 9.

## 2. Basic Concepts on Metaheuristic Algorithms

According to Glover and Kochenberger (2006), metaheuristics can be defined as "an iterative process that guides the operation of one or more subordinate heuristics (which may be from a local search process, to a constructive process of random solutions) to efficiently produce quality solutions for a problem." Metaheuristics are a family of approximate non-linear optimization techniques that provide *acceptable* solutions in a reasonable amount of time for solving computationally hard and complex problems in science, engineering, and other fields. Unlike exact optimization algorithms, metaheuristics do not absolutely guarantee provably optimal solutions. However, for some complex real-world problems, metaheuristics are preferred over gradient-based methods or mathematical programming. Some reasons for this are use of minimal assumptions about a non-analytical model, the presence of a time-consuming objective function, the existence of an NP-hard problem with difficult input structures, and problems with large instances and hard constraints.

Metaheuristics can be classified according to various characteristics (Talbi 2009, Weise et al. 2012): nature-inspired vs. not nature-inspired; deterministic vs. stochastic; population-based vs. single-solution-based search; and iterative vs. greedy. Independently from the number of objectives to be optimized, one of the main criteria for grouping metaheuristics is the trade-off between the search space exploration and exploitation. This criterion is linked to two different sub-families with complementary characteristics. The first one includes single-solution-based algorithms, which manipulate and transform a single solution during the search (high intensification). The second one involves population-based algorithms, which evolve a whole population of solutions (high diversification).

Single-solution-based metaheuristics improve a single solution during their running. They could be viewed as "walks" through neighborhoods or search trajectories through the search space of the problem at hand. They are performed by iterative procedures that move from the current solution to another one based on local search methods. Among others, some of the most prominent metaheuristics of this sub-family are: tabu search (Glover and Laguna 1997), simulated annealing (Kirkpatrick et al. 1983), variable neighborhood search (Hansen et al. 2010), GRASP (Feo and Resende 1995), and iterated local search (Lourenço et al. 2003).

Within the set of population-based metaheuristics, evolutionary algorithms (Back et al. 1997) and concretely, genetic algorithms (Goldberg and Holland 1988), are frequently used in many engineering and production problems (Chica et al. 2012, Zhang and Chiong 2016). There are many other algorithms that are based on handling a set of solutions at every iteration. These are ant-colony optimization (Dorigo and Stützle 2004), particle-swarm optimization (Kennedy 2010), scatter search (Laguna and Marti 2012), and estimation of distribution algorithms (Larranaga and Lozano 2002), among others. Finally, memetic algorithms (Moscato 1989, Ong et al. 2006) can be seen as a marriage between a population-based metaheuristic (global search) and a single-solution metaheuristic that locally improves the globally obtained solutions.

## 3. Handling Uncertainty in Optimization Problems

Traditional formulation of optimization problems is inherently static and deterministic. However, reality is dynamic and uncertain: environmental parameters fluctuate, materials wear down, processing or transportation times vary, clients change their demands, etc. (Beyer and Sendhoff 2007). When uncertainty is absent from the optimization formulation, the optimized solutions for those systems may be unstable and sensitive to small changes in the input parameters. A traditional way to tackle this uncertainty in optimization is by providing a high degree of robustness of the solutions. Robustness measures how the optimization problem solutions remain relatively unchanged when exposed to uncertain conditions.

Taguchi (1989) envisioned a three-stage design methodology composed of the system, parameters, and tolerance designs. In Taguchi's method, there are two main kinds of optimization parameters: those controllable parameters $x$ that are to be tuned; and uncontrollable noise factors $\xi$ such as environmental conditions or production tolerances. In a real-world system, an optimal design has to face different kinds of robustness depending on the source of uncertainties on the latter parameters: changing environmental and operating conditions, production tolerances and actuator imprecision, uncertainties in the system output, and feasibility uncertainties. These kinds of uncertainties are usually handled by optimization methods in three different ways (Beyer and Sendhoff 2007): deterministic, probabilistic, and possibilistic. Independently from the latter approaches, a common philosophy followed in robust optimization is to consider the worst-case scenario, but this is a conservative approach because it can result in poor optimization performance, and sometimes a solution that is useless in reality.

For production and design problems, many authors consider a pre-defined set of deterministic scenarios where some of the parameters of the problem are uncertain or depend on future actions (Scheffermann et al. 2009, Chica et al. 2016a). As an extension of this approach, an associated probability distribution could be assigned to each of these potential scenarios. The probabilistic way of solving an optimization problem under uncertainty is to specify a probability distribution for those incomplete or uncertain parameters.

Another way of dealing with uncertainty is use of possibilistic methods such as fuzzy-set-based methodologies (Zadeh 1965). Rather than characterizing a possible solution to an optimization problem as feasible or infeasible, these solutions are mapped to a membership function describing the degree of membership to the feasible set. If this membership function is equal to 1 it means it is completely robust, while if the function value is 0 it is completely non-robust.

The search for optimal robust designs often appears as a multi-criteria decision problem optimizing a conditional expectation and large dispersion or variance. In all these cases there is a trade-off between maximal performance and variance. For example, one proposal along these lines is the multi-objective six sigma of Shimoyama et al. (2005), where robustness is defined as stability of the system against uncertainty.

## 4. The Simheuristic Approach

Simulation has traditionally been recommended only when all other (exact mathematical-analytical) methods have failed due to problem/model complexity; thus simulation was regarded as an inferior "method of last resort." However, modern computing hardware, modeling paradigms, and advanced simulation software have together made simulation the method of choice that can produce results that are just as insightful and practically as precise as analytical methods (Lucas et al. 2015). Indeed, simulation results from well-designed studies are in many cases more accurate (i.e., less biased) and thus more valid than are corresponding results from exact mathematical-analytical models if, as is often the case, unrealistic simplifying assumptions (e.g., exponential distributions, with their mode of zero, to represent service-time durations) are required to push through the exact mathematical analysis. Simulation has benefited from the development of both general computing and simulation-specific modeling paradigms. Thus, in general, simulation modeling is a better approach than are exact mathematical-analytical approaches for complex problems where time dynamics and uncertainty are important (Borshchev and Filippov 2004).

Combining metaheuristics with other solution approaches such as simulation is becoming popular as an effective procedure to deal with complex combinatorial optimization problems (Ferreira 2013). Bianchi et al. (2009) review seminal works on the combination of simulation with metaheuristics but, to the best of our knowledge, it was with the work of Glover et al. (1996; 1999) and April et al. (2003) where this combination was popularized. These authors were the promoters of OptQuest, a "black-box" optimum-seeking software product that is currently integrated into several commercial simulation-modeling packages (Eskandari et al. 2011) including Arena (`www.arenasimulation.com`), Simio (`www.simio.com`), and SIMUL8 (`www.simul8.com`). In the use of this commercial software in concert with simulation-modeling packages, a stochastic simulation model is developed for a given system and then the input parameters of interest are changed in an attempt to optimize a designated output performance metric (Kleijnen and Wan 2007). However, as discussed in Hubscher-Younger et al. (2012), it is not always possible to apply this software directly out of the box. Instead, it needs to be adapted to the specific characteristics of the problem. Thus, researchers in the optimization community proposed more flexible and "white-box" approaches. In particular, Juan et al. (2015) introduced the term *simheuristics*, described the fundamental concepts behind this hybrid methodology, and reviewed some practical applications in the fields of transportation, logistics, health-care, production, and telecommunication systems.

Basically, simheuristics make use of a simulation paradigm to extend existing and efficient metaheuristics. As metaheuristics are designed to cope with deterministic problems, simheuristics can be seen as a metaheuristic extension to be employed when solving optimization problems under uncertainty (Juan et al. 2015). This simheuristics approach is similar to the simulation-for-optimization paradigm, which was previously defined by Fu (2002) and others. For example, Andradóttir (2006) elaborates on the subject of simulation-based optimization methods, providing a survey on optimization add-ons for discrete-event simulation software. These add-ons undergo fast development in commercial products such as OptQuest (`www.optquest.com`), Arena, Simio, and SIMUL8.

A simheuristic algorithm contains a particular simulation for an optimization approach and it is oriented efficiently to tackle an optimization problem involving stochastic components. These stochastic components can be either located in the objective function (e.g., random customers' demands, random processing times, etc.) or in the set of constraints (e.g., customers' demands that must be satisfied with a given probability, deadlines that must be met with a given probability, etc.). Therefore, most of the metaheuristic frameworks can be easily extended to simheuristics by replacing the analytic evaluation of the objective function or the constraints by a probabilistic assessment based on a simulation model. For this reason, when dealing with large-scale NP-hard optimization problems where uncertainty is present, researchers should consider simheuristics as a first-resort (Lucas et al. 2015) method since they empower metaheuristic approaches to cope with more realistic stochastic models.

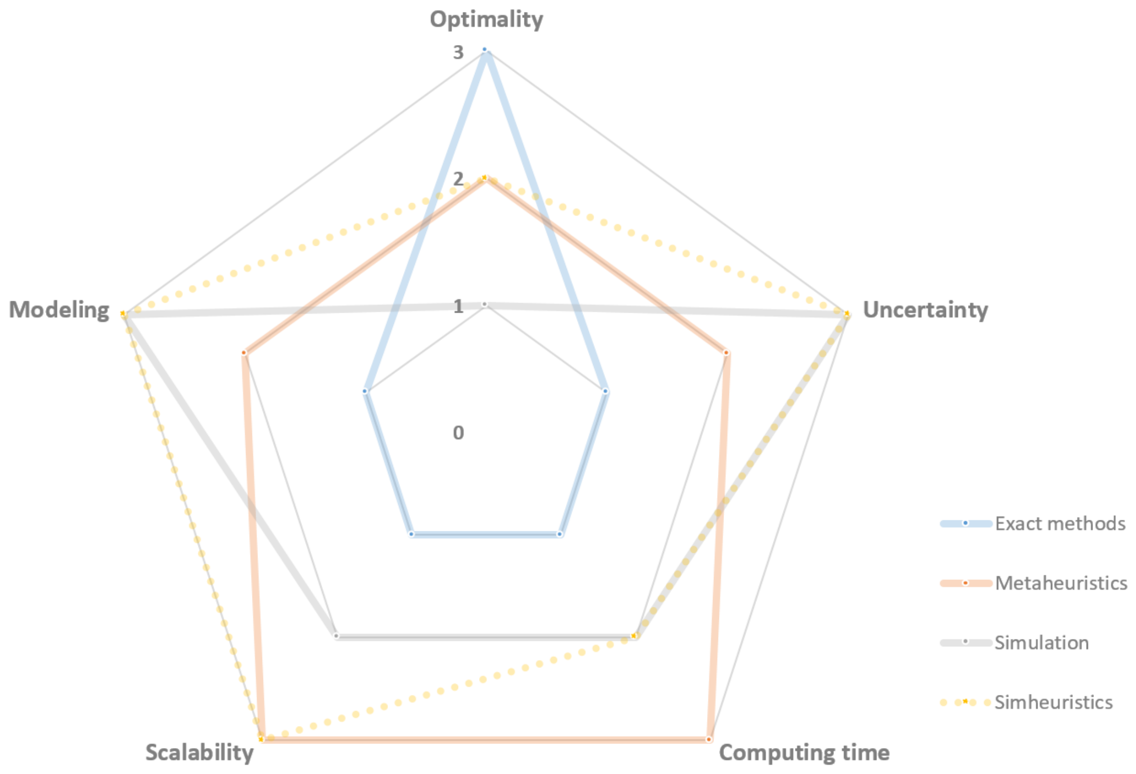

Figure 1: Comparison of optimization approaches based on their performance with respect to different dimensions.

Figure 1 shows a visual comparison, for different dimensions, of the main approaches discussed. While exact/analytical methods offer superior performance in the *optimality* dimension (i.e., the capacity to reach optimal values), they have severe limitations in other relevant dimensions such as *scalability* (i.e., ability to deal with large-scale problems), *modeling* (i.e., capacity to develop models that accurately represent the real-life system), *uncertainty* (i.e., ability to cope with non-deterministic scenarios), or *computing times* (especially for large-scale problems). Being an “offspring” of metaheuristics and simulation, simheuristics “inherits” the best properties of both methodologies, thus extending metaheuristics so they can deal with uncertainty and providing simulation techniques with the capability of coping with large-scale optimization problems successfully.

Recent research on simheuristics has applied it to different fields. Juan et al. (2011) propose a basic simheuristic approach to solve the vehicle-routing problem with stochastic demands. An improved algorithm for the same problem is introduced in Juan et al. (2013), where use of parallel and distributed computing techniques is also analyzed as a way to reduce computing times. Juan et al. (2014b) propose a simheuristic algorithm for solving the inventory-routing problem with stock-outs, while Juan et al. (2014a) present a simheuristic approach for solving the permutation flow-shop problem with stochastic processing times. An example of simheuristic application to distributed computer networks can be found in Cabrera et al. (2014), discrete-event simulation is combined with a simple metaheuristic framework to optimize a very large, dynamic network of non-dedicated computers offering online services over the Internet. Grasas et al. (2016) discuss how an iterated local-search framework can be extended to deal with stochastic combinatorial optimization problems. Finally, Gruler et al. (2016) develop a simheuristic approach for supporting stochastic waste-collection management in urban areas. April et al. (2006) constructed a discrete-event simulation model of a hospital emergency room to determine the optimal configuration of resources that results in the shortest average length of stay for patients, and also developed a simulation model to minimize staffing levels for personal claims processing in an insurance company.

Most of these applications refer to the integration of Monte-Carlo (MC) simulation with a metaheuristic framework, although some of them also discuss use of discrete-event simulation. However, other simulation paradigms are also possible. Overall, we distinguish four main simulation paradigms to be used within a simheuristic. Apart from MC simulation, discrete event simulation (DES) (Heath et al. 2011), system dynamics (SD) (Forrester 1961, Sterman 2001), and agent-based modeling (ABM) (Macal and North 2005, Epstein 2006, Kasaie and Kelton 2015) are specially suitable depending on the optimization-problem characteristics and available resources. Figure 2 shows these paradigms and how the components of a simheuristic are related. Additionally, these simulation paradigms will be discussed in Section 7.

## 5. Advantages of Using Simheuristics in Optimization

This section highlights the main advantages of employing simheuristics, which justify why we propose this methodology as a first-resort method for dealing with optimization problems under uncertainty:

- *Embracing reality by a validated simheuristic:* As opposed to use of analytical models, integrating simulation within a simheuristic approach allows researchers and decision makers to construct and study valid models of complex systems. Most recent simulation paradigms also allow for analysis of optimization problems under uncertainty without assumptions made

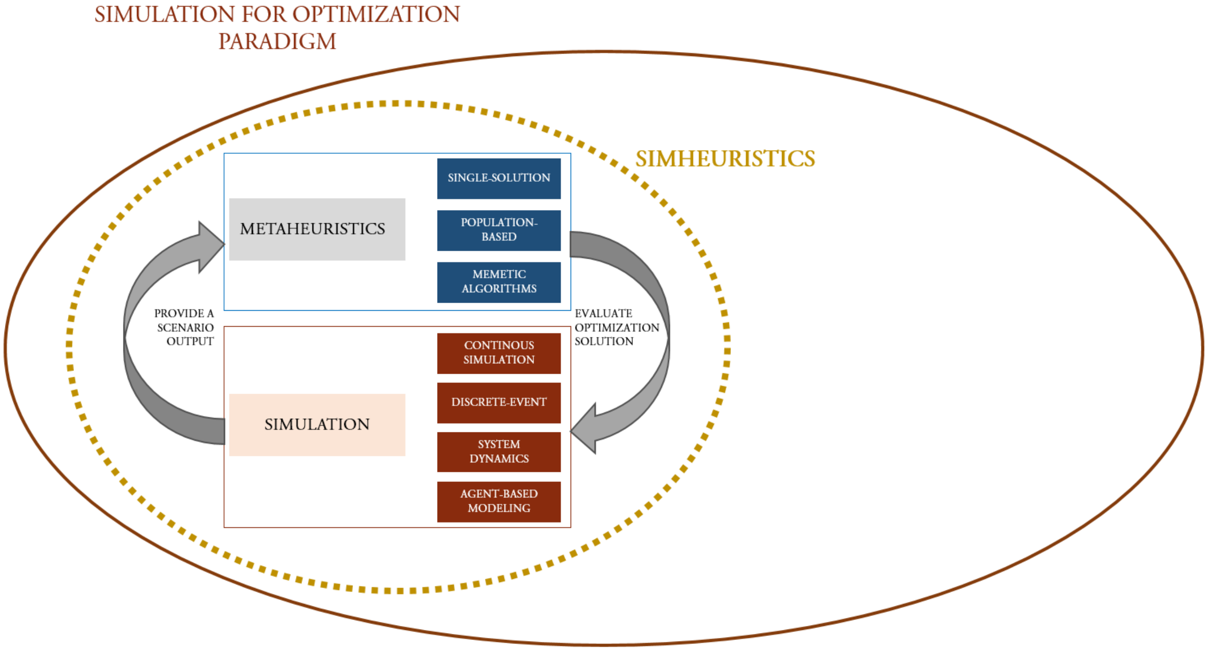

Figure 2: Main components of a simheuristics as a simulation-for-optimization paradigm.

solely for analytical tractability. These paradigms also facilitate involvement of stakeholders who are not directly the modelers of the simheuristic, i.e., participatory modeling (Voinov and Bousquet 2010). There are now new simulation paradigms that can better represent complex reality, and powerful computational resources to run demanding simulations for addressing models that are too complicated for over-simplified stylized analytical models. Model validation is a central pillar within the simulation community, as evidenced by its ubiquity in the leading texts over the years (Kelton et al. 2015). But validation should be applied to all modeling, including analytical, so this is not a disadvantage when using simulation but a requirement. As Lucas et al. (2015) stated, a model is not intrinsically valid just because it is analytical.

- *Risk assessment of alternative solutions and sensitivity analysis:* Once a simulation is built and validated, finding robust policies and comparing the merits of various policies are two of the main goals (Kleijnen et al. 2005). Joint use of simulation and metaheuristics within a simheuristic framework can help attain these two goals and has advantages compared to traditional analytical and exact optimization methodologies. The results of the simulations can be used to obtain additional information about the probability distribution of the quality of each simheuristic solution. This information is then used to introduce a risk/reliability analysis within the decision-making process. The risk-analysis capability of simheuristics is one of its major advantages due to the ability of metaheuritics to generate a set of different solutions and the ability of the simulation model to provide an observational sampling of the system. Running a sensitivity analysis (Saltelli et al. 2008) is another advantage of using a simulation together with a metaheuristic method. Sensitivity analysis reveals those input parameters that are most critical in determining the value of key output performance metrics. Generally, this is achieved by exploring the model's sensitivity to a particular parameter's configuration and input-value options. Sensitivity analysis is usually carried out to gain insights into existing or prospective systems, and this should lead to better decisions and to improved managerial outcomes.

This sensitivity analysis can be directly run by studying the output of the different simulation runs. Although a complete sensitivity analysis requires more advanced methods and specific tools to this end (Saltelli et al. 2008, Chica et al. 2017), the simheuristic learning process can give the modeler a first approach to a deeper sensitivity analysis of the system whose optimization is sought.

- *System understanding and output analysis:* When the simheuristic finishes, we can collect the output-data results and analyze them through machine-learning algorithms to discover hidden properties or relationships. The goal is to enable researchers to identify system patterns interactively, run high-dimensional explorations, or even check the veracity of the approximately-optimized simulation system (Lucas et al. 2015). This is also called the *innovization process* in evolutionary-computation research (Deb and Srinivasan 2006, Deb et al. 2014). It means that a set of trade-off optimal or near-optimal solutions, found using metaheuristics, are analyzed to decipher useful relationships among problem entities. It provides a better understanding of the problem to a designer or a practitioner. We extend here this concept by adding the simulation face of the simheuristic to enrich the *innovization process.*

  Additionally, visualization methods (e.g., histograms, box plots, or scatter plots) can be directly used to visualize post-run simulation outputs that go beyond the traditional analysis of the results. There is an increasing number of studies demonstrating that visualization combined with optimization can promote design innovations and provide decision makers with an improved understanding of the problem (Fleming et al. 2005, Stump et al. 2009, Bonissone et al. 2009). A good visualization enables decision makers to enhance insight into the problem and the different solutions to identify differences and similarities before coming to the final decision (Miettinen 2012).

Exploratory analysis of the input/output variables space of a model is also employed to strengthen trust in the model realism and to improve understanding of the behavior of the optimization and simulation models. By analyzing the distribution of the model variables and parameters, the modeler can move forward to a simpler and easier-to-understand setting. Use of this exploration, together with sensitivity analysis, provides information on influential factors that significantly affect the variability of model's results, and allow modelers to reach a deeper understanding of the complexity of the model, its uncertainties, interrelationships, and its potential future scenarios (Ligmann-Zielinska et al. 2014).

## 6. Limitations of Simheuristics

As with any methodology, there are also limitations when using simheuristics. In this section, we highlight some of these limitations as well as some positive aspects that ameliorate their negative impact on the optimum-seeking process.

- *Results are not expected to be truly provably optimal:* Metaheuristics do not ensure an optimal solution to an optimization problem, but rather an acceptable solution in a reasonable amount of time. This fact is amplified when using a simulation to be optimized. Even more, this simulation is a non-linear complex stochastic system that cannot be analytically treated. Therefore, simheuristics are an interesting alternative for practical cases requiring simple and flexible methods rather than obtaining an optimal solution to a "simplified" model with an excessive number of assumptions and without considering all the existing uncertainties.

- *Additional stakeholders' effort is demanded to define the system*: The set of advantages and "white-box" paradigms used in a simheuristic also requires additional effort when defining the simulation system and analyzing the results provided by the simheuristic. However, we think this design and validation effort is justified as modelers and decision makers can better understand their system from the results of the simheuristics and can adopt the final optimum-seeking results with higher confidence.

- *More computational resources are required compared to traditional methods*: The integration of a simulation engine within a metaheuristic requires high computational effort and also depends on the selected type of simulation paradigm. As will be discussed in Section 8, different strategies can be applied in order to alleviate this effort, such as: *(i)* 'filtering' the solutions generated by the metaheuristic engine, so that only the 'promising' ones are actually sent to the simulation engine; and *(ii)* using a small number of simulation runs in a first stage, and then analyzing in more detail only those that can be classified as 'very promising' solutions.

## 7. Simulation Paradigms

A simulation model can be defined as a set of rules (e.g. equations, flowcharts, or state machines) that define how the system being modeled will change in the future, given its present state (Borshchev and Filippov 2004). Simulation enables researchers to construct and study valid models of complex systems in relatively simple and straightforward ways by studying problems without assumptions made principally to obtain an analytically tractable model, which may jeopardize the veracity of the model. Simulation is the process of model "execution" that takes the model through state changes over time; these changes can be discretely or continuously occurring through time. In contrast with continuous simulations, a discrete simulation relies upon countable phenomena and produces a system that changes its behavior only in response to specific events, and typically model changes to a system resulting from a finite number of events distributed over time.

Different simulation paradigms can be integrated within a simheuristic. Here are four of the most significant ones (Borshchev and Filippov 2004), from faster (less computationally intensive) to richer approaches (more computationally intensive):

- *Monte Carlo (MC):* These techniques refer to probabilistic random sampling, and they are typically used to model non-dynamic stochastic systems. Most basic forms of MC simulations are quite straightforward and can be easily integrated inside a simheuristic framework, but more advanced versions might require use of variance-reduction techniques (e.g., common random numbers) or other statistical methods for input modeling, design of experiments, or output analysis.

- *Discrete event (DES):* This is a modeling method for stochastic, dynamic models where simulation state variables may change only at instantaneous, separated points in time, called *discrete events*. The simulation-clock time *of* occurrence of events can be continuous (i.e. a real number), as can be the time between successive events, but the occurrence of the events is instantaneous. Strictly speaking, this excludes state variables that change continuously with respect to time (Heath et al. 2011). Because of its stochasticity, multiple replications of simulation runs are generally required to obtain statistically precise results. The *process-oriented* world-view for DES generally consists of describing how entities move through various *processes*, where each process may require one or more *resources* and takes a certain (usually stochastic) amount of time. There is another approach, the *event-oriented* view, that works at a more basic and fundamental level, i.e., with the logic occurring at the instantaneous discrete events themselves rather than with entities and resources.

- *System dynamics (SD):* The main assumption here is that systems can be represented as a collection of *stocks* connected by *flows*, so material or energy accumulates in stocks and moves between them through flows (Forrester 1961, Sterman 2001). SD has been successfully applied to a diverse set of modeling applications (Winz et al. 2009, Wei et al. 2012, Chica et al. 2016b). To approach the problem in an SD way of modeling, one has to describe the system behavior as a number of interacting feedback loops, balancing or reinforcing, and delay structures. A whole set of sub-models or solvers for mathematical equations, such as partial differential equations, may be embedded within an SD model. A clear major advantage of this approach to modeling is that it can be extended to include almost any processes. However, a major disadvantage is that SD tools in most cases are not well suited for spatially explicit formalization.

- *Agent-based modeling (ABM):* This methodology relies on a population of autonomous entities called *agents* that behave according to rules and by interacting with other agents. The aggregation of these rules and interactions allows representation of complex and emerging dynamics as well as defining what-if scenarios and forecasting hypothetical scenarios (Macal and North 2005, Epstein 2006, Kasaie and Kelton 2015). This methodology is more appropriate than other quantitative tools when research questions emphasize groups of heterogeneous entities that operate in a complex environment with a measure of interest that comes from an emergent result. Also, researchers can perform modeling at the individual level, which does not require knowledge of higher-level assumptions (Rand and Rust 2011, Chiong and Kirley 2012, Chica and Rand 2016). In general, the computational requirements for these kind of models are high as many different operations take place at each simulation step and, as said, an MC simulation needs to be carried out because of the stochastic nature of ABM.

## 8. Best Design and Implementation Practices

In this section we outline a set of guidelines or best practices to build a simheuristic algorithm appropriately.

- *Do not overload simheuristics with long simulations:* In general, the modeler has to be careful not to let the simulation jeopardize the computing time given to the entire simulation-metaheuristic process. Otherwise, the metaheuristic would not have time to converge to a good solution if the dimension of the search space is high. Therefore, we recommend decomposing the simheuristic into various stages. For instance, some authors follow a three-stage approach (Figure 3). During the first stage, only fast simulations are included in the simheuristic framework. This is done by, for instance, running the simulation only a limited number of times to obtain rough estimates, or by running the simulation for only those new solutions of the metaheuristic that can be considered "promising" (e.g., solutions with good deterministic performance). During this stage, the simulation component of the simheuristic is used not only as a natural way to model the real system, but it also can provide valuable information to the metaheuristic component (i.e., the search process is simulation-driven). For example, it can be used to filter low quality solutions quickly. In a second stage, the best solutions identified in the previous stage are sent throughout a new simulation process with a larger number of iterations to obtain more precise estimates of the uncertain values of the model. The specific number of iterations might be given by error measures such as confidence intervals of the parameters with high uncertainty. Finally, a third and final stage can be used

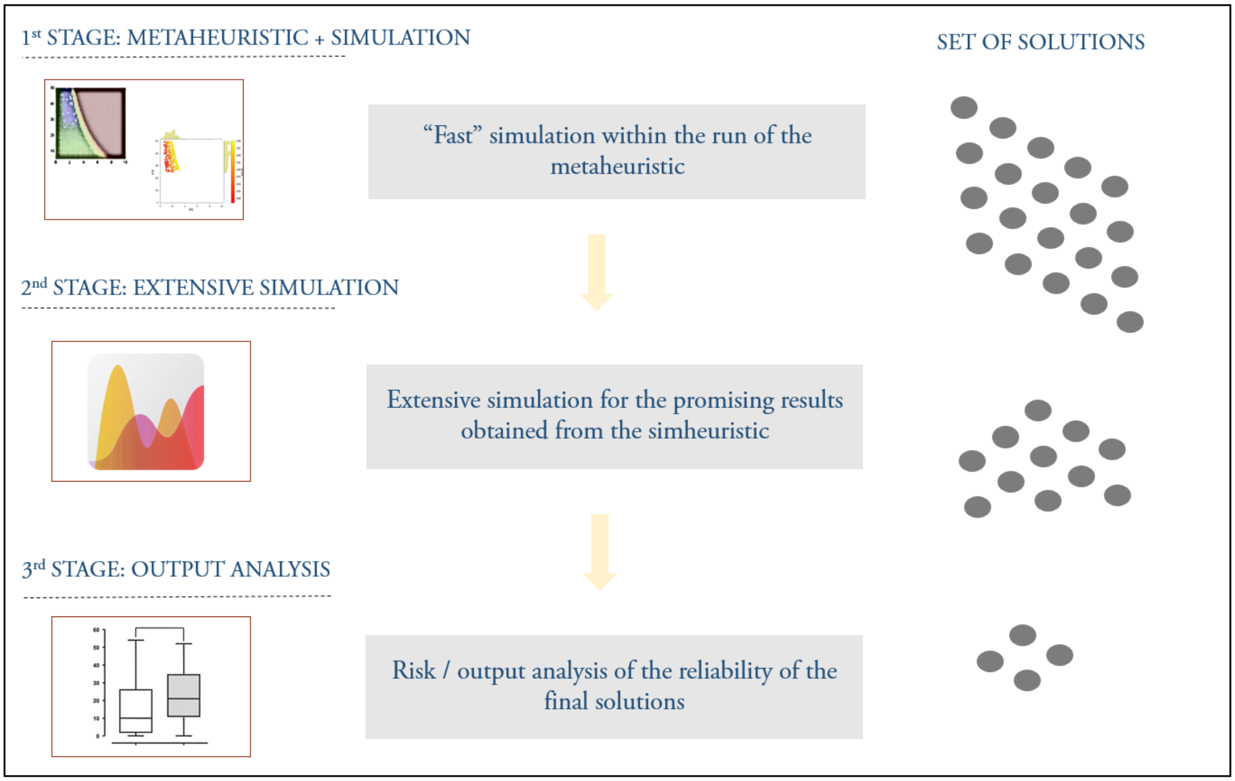
1st STAGE: METAHEURISTIC + SIMULATION
SET OF SOLUTIONS
"Fast" simulation within the run of the metaheuristic
2nd STAGE: EXTENSIVE SIMULATION
Extensive simulation for the promising results obtained from the simheuristic
3rd STAGE: OUTPUT ANALYSIS
Risk / output analysis of the reliability of the final solutions

Figure 3: A three-stage design for a simheuristic algorithm.

to complete a risk/robustness analysis on the best solution(s) selected by the decision maker. Dimensions other than the expected value of the solution need to be considered in a high-uncertainty environment because a solution with a low expected value could also show more variability than other alternative solutions. Consequently, the decision maker would need more information to decide which solution to choose based on her/his utility function and aversion to risk, or would even need more advanced optimization methods such as multi-objective optimization to have a set of solutions with different trade-offs between expected cost value and robust behavior in the environment (Chica et al. 2010; 2016a).

- *Choose a simulation paradigm that is understandable to decision makers:* Three main goals must be accomplished when developing and selecting the simulation model (Kleijnen et al. 2005): (*i*) develop a basic understanding of the simulation model and the system it emulates; (*ii*) find robust policies and decisions; and (*iii*) compare the merits of various policies or decisions. As mentioned, there is a wide set of available simulation paradigms and within each variant, many variations and possible designs arise. Our guideline here is to use, as much as possible, a participatory simulation-modeling process (Voinov and Bousquet 2010) to increase and share the knowledge and understanding of the system between all the actors involved in the optimization process. This involvement would also clarify and identify the impacts of solutions to a given problem, usually related to the final decision-making support. Additionally, different simulation paradigms can be used for each of the stages of the simheuristic. Then, a more enriched and computationally-intensive simulation model (e.g., ABM) can be used for the last stages of the simheuristic and applied only to a reduced set of the solutions provided by the metaheuristic. In contrast, lighter computational simulation models (e.g., a simple Monte-Carlo simulation over the stochastic simulation model) might be required in the first stage of the simheuristic. Each individual modeling paradigm has a rich history and exemplar cases in which the strengths of the respective methodology make it a good choice for a particular modeling situation. There also possibilities for combining each pair of approaches to develop hybrid models where each paradigm exploits its strengths (Heath et al. 2011). For instance, Djanatliev and German (2013) present different multi-paradigm simulation methods such as using SD for simulations at a high abstraction level and ABM/DES at an individual level. To sum up, the selection of the most appropriate simulation method depends on the questions to be answered.

- *Validate the simulation model before running the simheuristic:* A decisive phase when modeling a real-world system is model validation (Oliva 2003). We think that this is also a main guideline when designing the simheuristic as it applies to the simheuristic itself and specifically to its simulation model. The validation requires testing a set of hypotheses, the significance of their behavioral components (by assuming that the behavior is a consequence of the system structure), and the historical model fitting (Oliva 2003). Validation is also measured in terms of degrees of confidence or quality, which is usually difficult to obtain for most non-linear simulation models in use (Forrester 2007). The validation and testing of any model or decision-support system is a decisive step for ensuring its managerial adoption. Decision makers are all rightly concerned about whether results of each model are *correct* (Sargent 2005, Oliva 2003). However, the validation of non-linear models such as SD and ABM and their effectiveness for real-world problems is not straightforward. The validation process can be seen as a learning process where the modeler's understanding is enhanced through her/his interaction with the formal and mental model (Morecroft 2007). As this process evolves, both the formal and mental perceptions of the modelers change, leading to a successive approximation of the formal model to reality. Additionally, the utility and effectiveness of many non-linear models and their outputs are often judged by stakeholders and decision makers (Voinov and Bousquet 2010). Therefore, it is highly recommended to perform the validation of the models correctly. A set of validation techniques such as calibration (Sargent 2005), sensitivity analysis (Saltelli et al. 2008), boundary adequacy, and extreme cases tests (Qudrat-Ullah and Seong 2010) should be carried out for the corresponding simheuristic component in order to guarantee that the simulation model is a valid representation of the underlying system.

## 9. Concluding Remarks

The motivation of our work is to advocate that a combination of simulation models and metaheuristics should be considered as a first-resort method when dealing with real-world optimization problems. Apart from the fact that most of these problems are large-scale, they frequently involve many uncertainties regarding their constraints, input values, and objective functions. Although there are metaheuristic applications that add probabilistic and robustness capabilities to analytical models, they are extensions or patches to the original deterministic mathematical-model formulation. As we have discussed, integration of simulation methods with metaheuristics is a natural way to cope with these problems and it is currently the most convenient tool for practitioners. Although prohibitive and unaffordable in the past, advanced simulation methods are now commonly used in research and practice due to widespread and affordable availability of high-performance computing resources and much-improved software for simulation modeling and analysis. The simheuristics methodology can better face complex reality when seeking optima in uncertain environments like in many (if not most) production, logistics, and management problems, while presenting additional advantages.

In this paper we highlighted three main advantages of using simheuristics. First, it is a better way to embrace the reality of the systems we are seeking to optimize. There is no need to include many strong and over-simplifying assumptions to render an analytical model tractable, and in contrast, a valid simulation model, as rich and even as messy as needed to represent the actual system validly, can be easily included within the simheuristic. Second, a simheuristic can easily provide a risk assessment of the optimization-problem solutions. Third, simheuristics facilitate the understanding of the system's behavior by providing direct ways to perform sensitivity analyses, statistical analysis, and visualization based on the run's output data.

Additionally, we have presented the main simulation paradigms to be used within a simheuristic and a list of guidelines to take into account when designing a simheuristic. We suggested use of a multi-stage approach to alleviate the required computation effort of the simulation, and use of different simulation paradigms within the simheuristic. Likewise, the need for using a validated simulation model was affirmed. Finally, we encourage the use of a simulation paradigm that can be aligned with the "white-box" paradigm of simheuristics: being understandable and enhancing the decision makers' participation.

## Acknowledgments

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness and FEDER (TRA2013-48180-C3-P, TRA2015-71883-REDT, and NEWSOCO TIN2015-67661-P). Likewise, we want to acknowledge the support received by the Department of Universities, Research & Information Society of the Catalan Government (2014-CTP-00001).

## References

Andradóttir, S., 2006. An overview of simulation optimization via random search. Handbooks in operations research and management science 13, 617–631.

April, J., Better, M., Glover, F., Kelly, J., Laguna, M., 2006. Enhancing business process management with simulation optimization, in: Proceedings of the 2006 Winter Simulation Conference, pp. 642–649.

April, J., Glover, F., Kelly, J.P., Laguna, M., 2003. Practical introduction to simulation optimization, in: Proceedings of the 2003 Winter Simulation Conference, pp. 71–78.

Back, T., Fogel, D.B., Michalewicz, Z., 1997. Handbook of evolutionary computation. IOP Publishing Ltd., Bristol (UK).

Beyer, H., Sendhoff, B., 2007. Robust optimization - a comprehensive survey. Computer Methods in Applied Mechanics and Engineering 196, 3190–3218.

Bianchi, L., Dorigo, M., Gambardella, L.M., Gutjahr, W.J., 2009. A survey on metaheuristics for stochastic combinatorial optimization. Natural Computing: an international journal 8, 239–287.

Bonissone, P.P., Subbu, R., Lizzi, J., 2009. Multicriteria decision making (MCDM): a framework for research and applications. IEEE Computational Intelligence Magazine 4, 48–61.

Borshchev, A., Filippov, A., 2004. From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools, in: Proceedings of the 22nd international conference of the system dynamics society, Citeseer.

Boschetti, M.A., Maniezzo, V., Roffilli, M., Röhler, A.B., 2009. Matheuristics: optimization, simulation and control, in: Hybrid metaheuristics. Springer, pp. 171–177.

Cabrera, G., Juan, Á., Lázaro, D., Marquès, J.M., Proskurnia, I., 2014. A simulation-optimization approach to deploy internet services in large-scale systems with user-provided resources. Simulation 90, 644–659.

Chica, M., Barranquero, J., Kajdanowicz, T., Cordón, O., Damas, S., 2017. Multimodal optimization: an effective framework for model calibration. Information Sciences 375, 79–97.

Chica, M., Bautista, J., Cordón, Ó., Damas, S., 2016a. A multiobjective model and evolutionary algorithms for robust time and space assembly line balancing under uncertain demand. Omega 58, 55–68.

Chica, M., Cordón, O., Damas, S., Bautista, J., 2010. Multiobjective, constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search. Information Sciences 180, 3465–3487.

Chica, M., Cordón, O., Damas, S., Bautista, J., 2012. Multiobjective memetic algorithms for time and space assembly line balancing. Engineering Applications of Artificial Intelligence 25, 254–273.

Chica, M., Cordón, O., Damas, S., Iglesias, V., Mingot, J., 2016b. Identimod: modeling and managing brand value using soft computing. Decision Support Systems 89, 41–55.

Chica, M., Rand, W., 2016. Building agent-based decision support systems for word-of-mouth programs. A freemium application. Journal of Marketing Research (in press) .

Chiong, R., Kirley, M., 2012. Effects of iterated interactions in multiplayer spatial evolutionary games. IEEE Transactions on evolutionary computation 16, 537–555.

Deb, K., Bandaru, S., Greiner, D., Gaspar-Cunha, A., Tutum, C.C., 2014. An integrated approach to automated innovization for discovering useful design principles: Case studies from engineering. Applied Soft Computing 15, 42–56.

Deb, K., Srinivasan, A., 2006. Innovization: Innovating design principles through optimization, in: Proceedings of the 8th annual conference on Genetic and evolutionary computation, ACM. pp. 1629–1636.

Djanatliev, A., German, R., 2013. Prospective healthcare decision-making by combined system dynamics, discrete-event and agent-based simulation, in: Proceedings of the 2013 Winter Simulation Conference, IEEE. pp. 270–281.

Dorigo, M., Stützle, T., 2004. Ant Colony Optimization. MIT Press, Cambridge.

Epstein, J.M., 2006. Generative social science: Studies in agent-based computational modeling. Princeton University Press.

Eskandari, H., Mahmoodi, E., Fallah, H., Geiger, C.D., 2011. Performance analysis of comercial simulation-based optimization packages: Optquest and witness optimizer, in: Proceedings of the 2011 Winter Simulation Conference, Winter Simulation Conference. pp. 2363–2373.

Feo, T.A., Resende, M.G.C., 1995. Greedy randomized adaptive search procedures. Journal of Global Optimization 6, 109–133.

Ferreira, J.S., 2013. Multimethodology in metaheuristics. Journal of the Operational Research Society 64, 873–883.

Fleming, P.J., Purshouse, R.C., Lygoe, R.J., 2005. Many-objective optimization: An engineering design perspective, in: Evolutionary Multi-Criterion Optimization, Springer. pp. 14–32.

Forrester, J.W., 1961. Industrial dynamics. MIT Press.

Forrester, J.W., 2007. System dynamics: The next fifty years. System Dynamics Review 23, 359–370.

Fu, M.C., 2002. Optimization for simulation: Theory vs. practice. INFORMS Journal on Computing 14, 192–215.

Glover, F., Kelly, J.P., Laguna, M., 1996. New advances and applications of combining simulation and optimization, in: Proceedings of the 1996 Winter Simulation Conference, IEEE Computer Society. pp. 144–152.

Glover, F., Kelly, J.P., Laguna, M., 1999. New advances for wedding optimization and simulation, in: Simulation Conference Proceedings, 1999 Winter, IEEE. pp. 255–260.

Glover, F., Laguna, M., 1997. Tabu search. Kluwer Academic.

Glover, F.W., Kochenberger, G.A., 2006. Handbook of metaheuristics. volume 57. Springer Science & Business Media.

Goldberg, D.E., Holland, J.H., 1988. Genetic algorithms and machine learning. Machine Learning 3, 95–99.

Grasas, A., Juan, A.A., Lourenço, H.R., 2016. Simils: a simulation-based extension of the iterated local search metaheuristic for stochastic combinatorial optimization. Journal of Simulation 10, 69–77.

Gruler, A., Fikar, C., Juan, A., Hirsch, P., Contreras-Bolton, C., 2016. Supporting multi-depot and stochastic waste collection management in clustered urban areas via simulation–optimization. Journal of Simulation , 1–9.

Hansen, P., Mladenović, N., Pérez, J.A.M., 2010. Variable neighbourhood search: methods and applications. Annals of Operations Research 175, 367–407.

Heath, S.K., Buss, A., Brailsford, S.C., Macal, C.M., 2011. Cross-paradigm simulation modeling: challenges and successes, in: Proceedings of the 2011 Winter Simulation Conference, Winter Simulation Conference. pp. 2788–2802.

Hubscher-Younger, T., Mosterman, P.J., DeLand, S., Orqueda, O., Eastman, D., 2012. Integrating discrete-event and time-based models with optimization for resource allocation, in: Proceedings of the 2012 Winter Simulation Conference, IEEE. pp. 1–15.

Juan, A., Faulin, J., Grasman, S., Riera, D., Marull, J., Mendez, C., 2011. Using safety stocks and simulation to solve the vehicle routing problem with stochastic demands. Transportation Research Part C: Emerging Technologies 19, 751–765.

Juan, A.A., Barrios, B.B., Vallada, E., Riera, D., Jorba, J., 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., Faulin, J., Grasman, S.E., Rabe, M., Figueira, G., 2015. A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems. Operations Research Perspectives 2, 62–72.

Juan, A.A., Faulin, J., Jorba, J., Caceres, J., Marquès, J.M., 2013. Using parallel & distributed computing for real-time solving of vehicle routing problems with stochastic demands. Annals of Operations Research 207, 43–65.

Juan, A.A., Grasman, S.E., Caceres-Cruz, J., Bektaş, T., 2014b. A simheuristic algorithm for the single-period stochastic inventory-routing problem with stock-outs. Simulation Modelling Practice and Theory 46, 40–52.

Kasaie, P., Kelton, W.D., 2015. Guidelines for design and analysis in agent-based simulation studies, in: 2015 Winter Simulation Conference (WSC), pp. 183–193.

Kelton, W.D., Sadowski, R., Zupick, N.B., 2015. Simulation with Arena (6th Edition). McGraw-Hill Education.

Kennedy, J., 2010. Particle swarm optimization, in: Encyclopedia of Machine Learning. Springer, pp. 760–766.

Kirkpatrick, S., C. D. Gelatt, J., Vecchi, M.P., 1983. Optimization by simulated annealing. Science 220, 671–680.

Kleijnen, J.P., Sanchez, S.M., Lucas, T.W., Cioppa, T.M., 2005. State-of-the-art review: a user's guide to the brave new world of designing simulation experiments. INFORMS Journal on Computing 17, 263–289.

Kleijnen, J.P., Wan, J., 2007. Optimization of simulated systems: Optquest and alternatives. Simulation Modelling Practice and Theory 15, 354–362.

Laguna, M., Marti, R., 2012. Scatter search: methodology and implementations in C. volume 24. Springer Science & Business Media.

Larranaga, P., Lozano, J.A., 2002. Estimation of distribution algorithms: A new tool for evolutionary computation. volume 2. Springer Science & Business Media.

Ligmann-Zielinska, A., Kramer, D.B., Cheruvelil, K.S., Soranno, P.A., 2014. Using uncertainty and sensitivity analyses in socioecological agent-based models to improve their analytical performance and policy relevance. PloS one 9, e109779.

Lourenço, H.R., Martin, O.C., Stützle, T., 2003. Iterated local search, in: Handbook of metaheuristics. Springer, pp. 320–353.

Lucas, T.W., Kelton, W.D., Sánchez, P.J., Sanchez, S.M., Anderson, B.L., 2015. Changing the paradigm: Simulation, now a method of first resort. Naval Research Logistics (NRL) 62, 293–303.

Macal, C.M., North, M.J., 2005. Tutorial on agent-based modeling and simulation, in: Proceedings of the 37th conference on Winter simulation, ACM. pp. 2–15.

Miettinen, K., 2012. Survey of methods to visualize alternatives in multiple criteria decision making problems. OR Spectrum , 1–35.

Morecroft, J., 2007. Strategic modelling and business dynamics: A feedback systems approach. John Wiley & Sons.

Moscato, P., 1989. On evolution, search, optimization, genetic algorithms and martial arts: towards memetic algorithms. Technical Report 826. Caltech Concurrent Computation Program. Pasadena, USA.

Oliva, R., 2003. Model calibration as a testing strategy for system dynamics models. European Journal of Operational Research 151, 552–568.

Ong, Y.S., Nair, P.B., Lum, K.Y., 2006. Max-min surrogate-assisted evolutionary algorithm for robust design. IEEE Transactions on Evolutionary Computation 10, 392–404.

Qudrat-Ullah, H., Seong, B.S., 2010. How to do structural validity of a system dynamics type simulation model: the case of an energy policy model. Energy Policy 38, 2216–2224.

Rand, W., Rust, R.T., 2011. Agent-based modeling in marketing: Guidelines for rigor. International Journal of Research in Marketing 28, 181–193.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S., 2008. Global sensitivity analysis: the primer. John Wiley & Sons.

Sargent, R.G., 2005. Verification and validation of simulation models, in: Proceedings of the 37th conference on Winter simulation, pp. 130–143.

Scheffermann, R., Bender, M., Cardeneo, A., 2009. Robust solutions for vehicle routing problems via evolutionary multiobjective optimization, in: IEEE Congress on Evolutionary Computation (CEC), Trondheim, Norway. pp. 1605–1612.

Shimoyama, K., Oyama, A., Fujii, K., 2005. A new efficient and useful robust optimization approach - design for multi-objective six sigma, in: IEEE Congress on Evolutionary Computation (CEC), Piscataway. pp. 950–957.

Spall, J.C., 2005. Introduction to stochastic search and optimization: estimation, simulation, and control. volume 65. John Wiley & Sons.

Sterman, J.D., 2001. System dynamics modeling: Tools for learning in a complex world. California Management Review 43, 8–25.

Stump, G., Lego, S., Yukish, M., Simpson, T.W., Donndelinger, J.A., 2009. Visual steering commands for trade space exploration: user-guided sampling with example. Journal of Computing and Information Science in Engineering 9, 044501.

Taguchi, G., 1989. Introduction to Quality Engineering. American Supplier Institute.

Talbi, E.G., 2009. Metaheuristics: from design to implementation. John Wiley & Sons.

Voinov, A., Bousquet, F., 2010. Modelling with stakeholders. Environmental Modelling & Software 25, 1268–1281.

Wei, S., Yang, H., Song, J., Abbaspour, K.C., Xu, Z., 2012. System dynamics simulation model for assessing socio-economic impacts of different levels of environmental flow allocation in the Weihe River Basin, China. European Journal of Operational Research 221, 248–262.

Weise, T., Chiong, R., Tang, K., 2012. Evolutionary optimization: Pitfalls and booby traps. Journal of Computer Science and Technology 27, 907–936.

Winz, I., Brierley, G., Trowsdale, S., 2009. The use of system dynamics simulation in water resources management. Water Resources Management 23, 1301–1323.

Zadeh, L.A., 1965. Fuzzy sets. Information and control 8, 338–353.

Zhang, R., Chiong, R., 2016. Solving the energy-efficient job shop scheduling problem: a multi-objective genetic algorithm with enhanced local search for minimizing the total weighted tardiness and total energy consumption. Journal of Cleaner Production 112, 3361–3375.