4OR-Q J Oper Res (2014) 12:301–333
DOI 10.1007/s10288-014-0275-2

INVITED SURVEY

Simulation optimization: a review of algorithms and applications

Satyajith Amaran · Nikolaos V. Sahinidis · Bikram Sharda · Scott J. Bury

Received: 12 February 2014 / Revised: 13 October 2014 / Published online: 14 November 2014

Abstract Simulation optimization refers to the optimization of an objective function subject to constraints, both of which can be evaluated through a stochastic simulation. To address specific features of a particular simulation—discrete or continuous decisions, expensive or cheap simulations, single or multiple outputs, homogeneous or heterogeneous noise—various algorithms have been proposed in the literature. As one can imagine, there exist several competing algorithms for each of these classes of problems. This document emphasizes the difficulties in simulation optimization as compared to algebraic model-based mathematical programming makes reference to state-of-the-art algorithms in the field, examines and contrasts the different approaches used, reviews some of the diverse applications that have been tackled by these methods, and speculates on future directions in the field.

Keywords Simulation optimization · Optimization via simulation · Derivative-free optimization

S. Amaran · N. V. Sahinidis (✉)
Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA
e-mail: niksah@gmail.com; sahinidis@cmu.edu

S. Amaran
e-mail: samaran@andrew.cmu.edu

B. Sharda
Engineering and Process Sciences, Core R&D, The Dow Chemical Company, 2301 N. Brazosport Blvd., Freeport, TX 77541, USA
e-mail: BRSharda@dow.com

S. J. Bury
Engineering and Process Sciences, Core R&D, The Dow Chemical Company, Research Campus 1776 Building, Midland, MI 48667-1776, USA
e-mail: SJBury@dow.com

Mathematics Subject Classification 90-02 Operations research, mathematical programming: Research exposition (monographs, survey articles) · 65-02 Numerical analysis: Research exposition (monographs, survey articles) · 90C56 Derivative-free methods and methods using generalized derivatives

1 Introduction

Advances in modeling and availability of cheap computational power have enabled the science, engineering, and business research communities to make use of simulations to model phenomena and systems. It is only natural that there be a great interest in manipulating degrees of freedom in the simulations to optimize them.

The term Simulation Optimization (SO) is an umbrella term for techniques used to optimize stochastic simulations. Simulation optimization involves the search for those specific settings of the input parameters to a stochastic simulation such that a target objective, which is a function of the simulation output, is, without loss of generality, minimized.

As opposed to algebraic model-based mathematical programming SO does not assume that an algebraic description of the simulation is available—the simulation may be available as a black box that only allows the evaluation of the objective and constraints for a particular input. In fact, many SO algorithmic approaches solely depend on such input-output data from the simulation in their search for optimal input settings.

In addition, many large-scale and/or detailed simulations may be expensive to run, in terms of time, money, or resources. As a result, there is also a need to perform few simulations in this search for optimal parameters. Outputs from these stochastic simulations are not deterministic, and usually follow some output distribution, which may or may not vary across the parametric space. This uncertainty or variability in output also adds to the challenge of optimization, as it becomes harder to discern the quality of the parametric input in the presence of this output noise. In addition, when an algebraic description of the simulation is not accessible, derivative information is usually unavailable, and the estimation of derivatives from the use of finite differences may not be suitable due to noisy outputs and the expensive nature of simulations.

The nature of the stochastic simulations under study will determine the specific technique chosen to optimize them. The simulations, which are often discrete-event simulations, may be partially accessible to us in algebraic form, or may be purely available as an input-output model (as a black box); they may have single or multiple outputs; they may have deterministic or stochastic output(s); they may involve discrete or continuous parameters; and they may or may not involve explicit, or even implicit/hidden constraints.

A very general simulation optimization problem can be represented by P1.

$$\begin{aligned} \min\ & \mathbb{E}_{\omega}[f(x, y, \omega)] \\ \text{s.t.}\ & \mathbb{E}_{\omega}[g(x, y, \omega)] \leq 0 \\ & h(x, y) \leq 0 \\ & x_l \leq x \leq x_u \\ & x \in \mathbb{R}^n, y \in \mathbb{D}^m. \end{aligned} \tag{P1}$$

The function f can be evaluated through simulation for a particular instance of the continuous inputs x, discrete inputs y, and a realization of the random variables in the simulation, the vector ω (which may or may not be a function of the inputs, x and y). Similarly, the constraints defined by the vector-valued function g are also evaluated with each simulation run. In this formulation, expected values for these stochastic functions are used. There may be other constraints (represented by h) that do not involve random variables, as well as bound constraints on the decision variables.

The relaxation of any of these conditions would constitute a problem that would fall under the purview of SO. Most algorithms focus on problems that either have solely discrete choices, or solely continuous decisions to make. Each constraint may be thought of as representing additional outputs from the simulation that need to be taken into consideration. In addition, there may be bound constraints imposed on decision variables, that may either be available or obtained from domain-specific knowledge. Relatively few existing algorithms attempt to address both discrete and continuous choices simultaneously, although some broad classes of approaches naturally lend themselves to be applicable in either, and therefore both, settings. Further, the discrete variables may either be binary, integer-ordered, or categorical and lie in some discrete space $\mathbb{D}$.

As can be seen, the formulation P1 is extremely general, and therefore a wide variety of applications fall under the scope of simulation optimization. Various applications of simulation optimization in diverse research fields are tabulated in Sect. 2.

Another common assumption is that f is a real-valued function and g is a real vector-valued function, both of whose expected values may or may not be smooth or continuous functions. The most common objective in SO is to optimize the expected value of some performance metric, but other objective functions may be appropriate depending on the application. For instance, an objective that minimizes risk could be a possible alternative, in which case one would incorporate some sort of variance measure as well into the objective.

This paper is meant to be a survey of available techniques as well as recent advances in simulation optimization. The remainder of the introduction section provides a literature survey of prior reviews, and elaborates on the relationship of simulation optimization to algebraic model-based mathematical programming derivative-free optimization, and machine learning. Section 2 provides a glimpse into the wide variety of applications of simulation optimization that have appeared in the literature. Section 3 focuses on various algorithms for discrete and continuous simulation optimization, provides basic pseudocode for major categories of algorithms, and provides comprehensive references for each type of algorithm. Section 4 provides a listing of available software for simulation optimization and Sect. 5 discusses means to compare their performance. Section 6 summarizes the progress of the field, and outlines some current and future topics for research.

1.1 Prior reviews of simulation optimization

Several review papers (cf. Meketon 1987; Jacobson and Schruben 1989; Safizadeh 1990; Azadivar 1992; Fu 1994, 2002; Carson and Maria 1997; Andradóttir 1998, 2006b; Azadivar 1999; Swisher et al. 2000; Fu et al. 2000, 2005; Tekin and Sabun-

cuoglu 2004; Hong and Nelson 2009; Ammeri et al. 2011; Pasupathy and Ghosh 2013), books and research monographs (cf. Spall 2003; Rubinstein and Kroese 2004; Kleijnen 2008; Chen and Lee 2010), and theses (cf. Angün 2004; Driessen 2006; Deng 2007; Chang 2008; Frazier 2009; Kabirian 2009) have traced the development of simulation optimization.

Meketon (1987) provides a classification of algorithmic approaches for optimization over simulations based on how much information or structure about the underlying model is known. The paper surveys the progress of the field between 1975 and 1987, and focuses on continuous simulation optimization. Andradóttir (1998) provides a tutorial on gradient-based procedures for continuous problems. Carson and Maria (1997) and Azadivar (1999) also give brief outlines of and pointers to prevailing simulation optimization algorithms.

Fu et al. (2000) contains several position statements of eminent researchers and practitioners in the field of simulation, where the integration of simulation with optimization is discussed. The issues addressed include generality versus specificity of an algorithm, the wider scope of problems that simulation optimization methodologies have the potential to address, and the need for integrating provably convergent algorithms proposed by the research community with metaheuristics often used by commercial simulation software packages.

Of the more recent surveys, Fu (1994) provides an excellent tutorial on simulation optimization, and focuses on continuous optimization problems more than discrete optimization problems. The paper focuses specifically on discrete-event simulations. Fu (2002) provides a comprehensive survey of the field and its scope—the paper outlines the different ways in which optimization and simulation interact, gives examples of real-world applications, introduces simulation software and the optimization routines that each of them use, provides a very basic tutorial on simulation output analysis and convergence theory for simulation optimization, elaborates on algorithms for both continuous and discrete problems, and provides pointers to many useful sources. Fu et al. (2005) provide a concise, updated version of all of this, and also talk about estimation of distribution algorithms.

Tekin and Sabuncuoglu (2004) provide a table that analyzes past review papers and the techniques they focus on. Apart from providing detailed updates on advances in approaches and algorithms, the paper also lists references that attempt to compare different SO techniques. Hong and Nelson (2009) classify simulation optimization problems into those with (1) a finite number of solutions; (2) continuous decision variables; and (3) discrete variables that are integer-ordered. The paper describes procedures for each of these classes. The recent survey by Ammeri et al. (2011), classifies simulation optimization algorithms and provides a survey of methods as well as applications appearing in the literature between 1995 and 2010.

This work provides an overview of techniques, and briefly outlines well-established methods with pointers to more detailed surveys, while expounding on more recent methods in a concise manner. Though several reviews exist, we catalog the most recent developments—the emergence of derivative-free optimization and its relationship with simulation optimization, the appearance of simulation test-beds for comparing algorithms, the recent application of simulation optimization in diverse fields, the development of and interest in related techniques and theory by the machine learn-

ing community and the optimization community, as well as the sheer unprecedented nature of recent interest in optimizing over simulations. A reflection of a surge in recent interest is evidenced by the fact that more than half of the works we reference were published in the last decade. The intent is to not only trace the progress of the field, but to provide an update on state-of-the-art methods and implementations, point the familiar as well as the uninitiated reader to relevant sources in the literature, and to speculate on future directions in the field.

1.2 A note on terminology and scope

As simulation optimization involves the use of algorithms that arose from widely differing fields (Sect. 3), has relationships to many diverse disciplines (Sect. 1.3), and has been applied to many different practical applications from biology to engineering to logistics (Sect. 2), it is not surprising that it is known by various names in different fields. It has also been referred to as simulation-based optimization, stochastic optimization, parametric optimization, black-box optimization, and Optimization via Simulation (OvS), where the continuous and discrete versions are accordingly known as Continuous Optimization via Simulation (COvS) and Discrete Optimization via Simulation (DOvS). Each algorithmic technique may also go by different names, and we attempt to reconcile these in Sect. 3.

Inputs to the simulation may be variously referred to as parameter settings, input settings, variables, controls, solutions, designs, experiments (or experimental designs), factors, or configurations. Outputs from the simulation are called measurements, responses, performance metrics, objective values, simulation replications, realizations, or results. The performance of a simulation may also be referred to as an experiment, an objective function evaluation, or simply a function evaluation. We will use the term 'iteration' to refer to a fixed number of function evaluations (usually one) performed by a simulation optimization algorithm.

A note of caution while using SO methods is to incorporate as much domain specific knowledge as possible in the use of an SO algorithm. This may be in terms of (1) screening relevant input variables, (2) scaling and range reduction of decision variables, (3) providing good initial guesses for the algorithm; and (4) gleaning information from known problem structure, such as derivative estimates.

Table 1 classifies the techniques that are usually most suitable in practice for different scenarios in the universe of optimization problems. Certain broad classes of algorithms, such as random search methods, may be applicable to all of these types of

Table 1 Terminology of optimization problems

	Algebraic model available	Unknown/complex problem structure
Deterministic	Traditional math programming (linear, integer, and nonlinear programming)	Derivative-free optimization
Uncertainty present	Stochastic programming, robust optimization	Simulation optimization

problems, but they are often most suitable when dealing with pathological problems (e.g., problems with discontinuities, nonsmoothness) and are often used because they are relatively easy to implement.

The possibilities of combining simulation and optimization procedures are vast: simulation with optimization-based iterations; optimization with simulation-based iterations; sequential simulation and optimization; and alternate simulation and optimization are four such paradigms. A recent paper by Figueira and Almada-Lobo (2014) delves into the taxonomy of such problems, and provides a guide to choosing an appropriate approach for a given problem. As detailed by Meketon (1987), different techniques may be applicable or more suitable depending on how much is known about the underlying simulation, such as its structure or associated probability distributions. We focus on approaches that are applicable in situations where all the optimization scheme has to work with are evaluations of $f(x, y, \omega)$ and $g(x, y, \omega)$, or simply, observations with noise.

1.3 Relationship to other fields

Algebraic Model-based Mathematical Programming As mentioned earlier, most mathematical programming methods rely on the presence of an algebraic model. The availability of an algebraic model has many obvious implications to a mathematical programming expert, including the ability to evaluate a function quickly, the availability of derivative information, and the possibility of formulating a dual problem. None of these may be possible to do/obtain in an SO setting.

In the case with continuous decisions, derivative information is often hard to estimate accurately through finite differences, either due to the stochastic noise associated with objective function evaluations, or due to the large expense associated with obtaining function evaluations, or both. The inherent stochasticity in output also renders automatic differentiation (AD) (Rall 1981; Griewank and Walther 2008) tools not directly applicable. Moreover, automatic differentiation may not be used when one has no access to source code, does not possess an AD interface to proprietary simulation software, and, of course, when one is dealing with a physical experiment. The lack of availability of derivative information has further implications—it complicates the search for descent directions, proofs of convergence, and the characterization of optimal points.

Simulation optimization, like stochastic programming, also attempts to optimize under uncertainty. However, stochastic programming differs in that it makes heavy use of the model structure itself (Birge and Louveaux 2011). Optimization under uncertainty techniques that make heavy use of mathematical programming are reviewed in Sahinidis (2004).

Derivative-Free Optimization Both Simulation Optimization and Derivative-Free Optimization (DFO) are referred to in the literature as black-box optimization methods. Output variability is the key factor that distinguishes SO from DFO, where the output from the simulation is deterministic. However, there are many approaches to DFO that have analogs in SO as well (e.g., response surfaces, direct search methods, metaheuristics), cf. Sect. 3.

Another distinction is that most algorithms in DFO are specifically designed keeping in mind that function evaluations or simulations are expensive. This is not necessarily the case with SO algorithms.

With regard to rates of convergence, SO algorithms are generally inefficient and convergence rates are typically very slow. In general, one would expect SO to have a slower convergence rate than DFO algorithms simply because of the additional complication of uncertainty in function evaluations. As explained in Conn et al. (2009), some DFO algorithms, under certain assumptions, expect rates that are closer to linear than quadratic, and therefore early termination may be suitable. As described in some detail by Fu (1994), the best possible convergence rates for SO algorithms are generally $\mathcal{O}(1/\sqrt{k})$, where k is the number of samples. This is true from the central limit theorem that tells us the rate at which the best possible estimator converges to the true expected function value at a point. This implies that though one would ideally incorporate rigorous termination criteria in algorithm implementations, most practical applications have a fixed simulation or function evaluation budget that is reached first.

Machine Learning Several subcommunities in the machine learning community address problems closely related to simulation optimization. Traditional machine learning settings assume the availability of a fixed dataset. Active learning methods (Cohn et al. 1996; Settles 2010) extend machine learning algorithms to the case where the algorithms are allowed to query an oracle for additional data to infer better statistical models. Active learning is closely related in that this choice of sampling occurs at every iteration in a simulation optimization setting as well. The focus of active learning is usually to learn better predictive models rather than to perform optimization.

Reinforcement learning Stephens and Baritompa (1998) is broadly concerned with what set of actions to take in an environment to maximize some notion of cumulative reward. Reinforcement learning methods have strong connections to information theory, optimal control, and statistics. The similarity with simulation optimization is that the common problem of exploration of the search space versus exploitation of known structure of the cost function arises. However, in the reinforcement learning setting, each action usually also incurs a cost, and the task is to maximize the accumulated rewards from all actions—as opposed to finding a good point in the parameter space eventually.

Policy gradient methods (Peters et al. 2003) are a subfield of reinforcement learning, where the set of all possible sequences of actions form the policy space, and a gradient in this policy space is estimated and a gradient ascent-type method is then used to move to a local optimum. Bandit optimization (Gittins 1989) is another subfield of reinforcement learning that involves methods for the solution to the multi-armed bandit problem. The canonical example involves a certain number of slot machines, and a certain total budget to play them. Here, each choice of sample corresponds to which slot machine to play. Each play on a slot machine results in random winnings. This setting is analogous to discrete simulation optimization (DOvS) over finite sets, although with a different objective (Powell and Ryzhov 2012). Again, in DOvS over finite sets, we are only concerned with finding the best alternative eventually, whereas the cumulative winnings is the concern in the multi-armed bandit problem.

Relationship to other fields Most, if not all, simulation optimization procedures have elements that are derived from or highly related to several other fields. Direct search procedures and response surface methodologies (RSM) have strong relationships with the field of experimental design. RSM, sample path optimization procedures, and gradient-based methods heavily incorporate ideas from mathematical programming. RSM also involves the use of nonparametric and Bayesian regression techniques, whereas estimation of distribution algorithms involves probabilistic inference, and therefore these techniques are related to statistics and machine learning. Simulation optimization has been described as being part of a larger field called computational stochastic optimization. More information is available at Powell (2013).

2 Applications

SO techniques are most commonly applied to either (1) discrete-event simulations, or (2) systems of stochastic nonlinear and/or differential equations.

As mentioned in Fu (1994), discrete event simulations can be used to model many real-world systems such as queues, operations, and networks. Here, the simulation of a system usually involves switching or jumping from one state to another at discrete points in time as events occur. The occurrence of events is modeled using probability distributions to model the randomness involved.

Stochastic differential equations may be used to model phenomena ranging from financial risk (Merton 1974) to the control of nonlinear systems (Song and Grizzle 1995) to the electrophoretic separation of DNA molecules (Cho and Dorfman 2010).

With both discrete-event simulations and stochastic differential equation systems, there may be several parameters that one controls that affect some performance measure of the system under consideration, which are essentially degrees of freedom that may be optimized through SO techniques. Several applications of SO from diverse areas have been addressed in the literature and we list some of them in Table 2.

3 Algorithms

Algorithms for SO are diverse, and their applicability may be highly dependent on the particular application. For instance, algorithms may (1) attempt to find local or global solutions; (2) address discrete or continuous variables; (3) incorporate random elements or not; (4) be tailored for cases where function evaluations are expensive; (5) emphasize exploration or exploitation to different extents; (6) assume that the uncertainty in simulation output is homoscedastic or that it comes from a certain probability distribution; or (7) rely on underlying continuity or differentiability of the expectation (or some function of a chosen moment) of the simulation output. The sheer diversity of these algorithms also makes it somewhat difficult to assert which one is better than another in general, and also makes it hard to compare between algorithms or their implementations.

As mentioned in Sect. 1.3, many algorithms that are available for continuous simulation optimization have analogs in derivative-based optimization and in derivative-free optimization, where function evaluations are deterministic. In any case, the key lies

Table 2 Partial list of published works that apply simulation optimization

Domain of application	Application and citations
Operations	Buffer location (Lutz et al. 1998), nurse scheduling (Tein and Ramli 2010), inventory management (Köchel and Nieländer 2005; Schwartz et al. 2006), health care (Angelis et al. 2003), queuing networks (Fu and Hill 1997; Bhatnagar 2005; Mishra et al. 2007)
Manufacturing	PCB production (Dengiz and Akbay 2000), engine manufacturing (Syberfeldt and Lidberg 2012), production planning (Kenne and Gharbi 2001; Kleijnen 1993), manufacturing-cell design (Irizarry et al. 2001), kanban sizing (Hall et al. 1996)
Medicine and biology	Protein engineering (Romero et al. 2013), cardiovascular surgery (Xie et al. 2012), breast cancer epidemiology (Ferris et al. 2005), bioprocess control (Vande Wouwer et al. 2001; Renotte and Vande Wouwer 2003), ECG analysis (Gerencsér et al. 2002), medical image analysis (Merhof et al. 2007)
Engineering	Welded beam design (Yang and Deb 2010), solid waste management (Yeomans 2007), pollution source identification (Ayvaz 2010), chemical supply chains (Jung et al. 2004), antenna design (Prakash et al. 2008), aerodynamic design (Xing and Damodaran 2002, 2005a, b; Kothandaraman and Rotea 2005), distillation column optimization (Ramanathan et al. 2001), well placement (Bangerth et al. 2005), servo system control (Radac et al. 2011), power systems (Ernst et al. 2007), radar analysis (Khan et al. 2006)
Computer science, networks, electronics	Server assignment (Kulturel-Konak and Konak 2010), wireless sensor networks (Dhivya et al. 2011), circuit design (Li 2009), network reliability (Kroese et al. 2007)
Transportation and logistics	Traffic control and simulation (Yun and Park 2010; Balakrishna et al. 2007; Osorio and Bierlaire 2010), metro/transit travel times (Hill and Fu 1995; Yalçinkaya and Mirac Bayhan 2009), air traffic control (Kleinman et al. 1997; Hutchison and Hill 2001)

in the statistics of how noise is handled, and how it is integrated into the optimization scheme. We will provide pointers to references that are applicable to simulation optimization in particular. A comprehensive review of methods for derivative-free optimization is available in Rios and Sahinidis (2013).

Each major subsection below is accompanied by pseudocode to give researchers and practitioners unfamiliar with the field an idea of the general approach taken by each of these algorithms. Many of the sections include pointers to convergence proofs for individual algorithms. Optimality in simulation optimization is harder to establish than in algebraic model-based mathematical programming or derivative-free optimization due to the presence of output variability. Notions of optimality for simulation optimization

Table 3 Classification of simulation optimization algorithms

Algorithm class	Discrete	Continuous	Local	Global
Ranking and selection	×			×
Metaheuristics	×	×		×
Response surface methodology		×	×	×
Gradient-based methods		×	×	
Direct search	×	×	×	
Model-based methods	×	×	×	×
Lipschitzian optimization		×		×

are explored in Fu (1994); for the discrete case, Xu et al. (2010), for instance, establishes conditions for local convergence, where a point being 'better' than its $2m + 1$ neighboring solutions is said to be locally optimal. There has also been some work in establishing Karush-Kuhn-Tucker (KKT) optimality conditions for multiresponse simulation optimization (Bettonvil et al. 2009). Globally convergent algorithms will locate the global optimal solution eventually, but assuring this would require all feasible solutions to be evaluated through infinite observations; in practice, a convergence property that translates to a practical stopping criterion may make more sense (Hong and Nelson 2009).

Based on their scope, the broad classes of algorithms are classified in Table 3. Algorithms are classified based on whether they are applicable to problems with discrete/continuous variables, and whether they focus on global or local optimization. However, there may be specific algorithms that have been tweaked to make them applicable to a different class as well, which may not be captured by this table.

3.1 Discrete optimization via simulation

Discrete optimization via simulation is involved with finding optimal settings for input variables that can only take discrete values. This may be in the form of *integer-ordered* variables or *categorical* variables (Pasupathy and Henderson 2011). Integer-ordered variables are allowed to take on integer or discrete values within a finite interval, where the order of these values translates to some physical interpretation. For example, this could be the number of trucks available for vehicle routing, or the set of standard pipe diameters that are available for the construction of a manufacturing plant. Categorical variables refer to more general kinds of discrete decisions, ranging from conventional on-off (0–1 or binary) variables to more abstract decisions such as the sequence of actions to take given a finite set of actions. It should be noted that though integer-ordered variables, for instance, may be logically represented using binary variables, it may be beneficial to retain them as integer-ordered to exploit correlations in objective function values between adjacent integer values.

A rich literature in DOvS has developed over the last 50 years, and the specific methods developed are tailored to the specific problem setting. Broadly, methods are tailored for finite or for very large/potentially infinite parameter spaces.

3.1.1 Finite parameter spaces

In the finite case, where the number of alternatives is small and fixed, the primary goal is to decide how to allocate the simulation runs among the alternatives. In this setting, there is no emphasis on 'search', as the candidate solution pool is small and known; each iteration is used to infer the best, in some statistical sense, simulation run(s) to be performed subsequently.

The optimization that is desired may differ depending on the situation, and could involve:

1. The selection of the best candidate solution from a finite set of alternatives;
2. The comparison of simulation performance measures of each alternative to a known standard or control; or
3. The pairwise comparison between all solution candidates.

Item (1) is referred to as the *ranking and selection* problem. Items (2) and (3) are addressed under literature on *multiple comparison procedures*, with the former referred to as *multiple comparisons with a control*.

Ranking and Selection In traditional ranking and selection, the task is to minimize the number of simulation replications while ensuring a certain probability of correct selection of alternatives. Most procedures try to guarantee that the design ultimately selected is better than all competing alternatives by δ with a probability at least $1-\alpha$. δ is called the indifference zone, and is the value deemed to be sufficient to distinguish between expected performance among solution candidates.

Conventional procedures make use of the Bonferroni inequality which relates probabilities of the occurrence of multiple events with probabilities of each event. Other approaches involve the incorporation of covariance induced by, for example, the use of common random numbers to expedite the algorithmic performance over the more conservative Bonferroni approach. Kim and Nelson (2006, 2007) and Chick (2006) provide a detailed review and provide algorithms and procedures for this setting. Extensions of fully sequential ranking and selection procedures to the constrained case have been explored as well, e.g., Andradóttir and Kim (2010).

An alternative formulation of the ranking and selection of the problem would be to try to do the best within a specified computational budget, called the *optimal computing budget allocation* formulation (Chen 1995). Chen et al. (2009) present more recent work, while the stochastically constrained case is considered in Lee et al. (2012).

Recent work (Hunter and Pasupathy 2013) in the area of DOvS over finite sets provides a quick overview of the field of ranking and selection, and considers general probability distributions and the presence of stochastic constraints simultaneously.

A basic ranking and selection procedure (Kim and Nelson 2007) is outlined in Algorithm 1, where it is assumed that independent data comes from normal distributions with unknown, different variances.

Multiple comparison procedures Here, a number of simulation replications are performed on all the potential designs, and conclusions are made by constructing confidence intervals on the performance metric. The main ideas and techniques for multiple

Algorithm 1 Basic ranking and selection procedure for SO

Require: Confidence level $1-\alpha$, indifference zone parameter δ
1: Take n_0 samples from each of the $1, \ldots, K$ potential designs
2: Compute sample means, $\bar{t}_{k,n_0}$ and sample variances, S_k, for each of the designs
3: Determine how many new samples, $N_k := \max\left\{n_0, \left\lceil \frac{\psi^2 S_k^2}{\delta^2} \right\rceil\right\}$, to take from each system, where the Rinott constant ψ is obtained from Bechhofer et al. (1995)
4: Select the system with the best new sample mean, $\bar{t}_{k,N_k+n_0}$.

comparisons in the context of pairwise comparisons, or against a known standard are presented in Hochberg and Tamhane (1987), Fu (1994) and Hsu (1996). Recent work in multiple comparisons with a control include Kim (2005) and Nelson and Goldsman (2001), which provide fully sequential and two-stage frequentist procedures respectively; and Xie and Frazier (2013), which addresses the problem using a Bayesian approach.

Comprehensive treatment of ranking and selection and multiple comparison procedures may be found in Goldsman and Nelson (1998) and Bechhofer et al. (1995). A detailed survey that traces the development of techniques in simulation optimization over finite sets is available in Tekin and Sabuncuoglu (2004).

3.1.2 Large/infinite parameter spaces

To address DOvS problems with a large number of potential alternatives, algorithms that have a search component are required. Many of the algorithms that are applicable to the continuous optimization via simulation case are, with suitable modifications, applicable to the case with large/infinite parameter spaces. These include (1) ordinal optimization (2) random search methods and (3) direct search methods.

Ordinal optimization methods (Ho 1999) are suitable when the number of alternatives is too large to find the globally optimal design in the discrete-event simulation context. Instead, the task is to find a satisfactory solution with some guarantees on quality (called alignment probability) (Lau and Ho 1997). Here, the focus is on sampling a chosen subset of the solutions and evaluating them to determine the best among them. The key lies in choosing this subset such that it contains a subset of satisfactory solutions. The quality or satisfaction level of this selected subset can be quantified (Chen 1996). A comparison of subset selection rules is presented in Jia et al. (2006) and the multi-objective case is treated in Teng et al. (2007).

Random search methods, include techniques such as simulated annealing (e.g., Alrefaei and Andradóttir 1999), genetic algorithms, stochastic ruler methods (e.g., Yan and Mukai 1992), stochastic comparison (e.g., Gong et al. 2000), nested partitions (e.g., Shi and Ólafsson 2000), ant colony optimization (e.g., Dorigo and Stützle 2004; Dorigo and Blum 2005), and tabu search (e.g., Glover and Hanafi 2002). Some of these—simulated annealing, genetic algorithms, and tabu search—are described in Sect. 3.6). Ant colony optimization is described under model-based methods (cf. Sect. 3.7.2). Proofs of global convergence, i.e., convergence to a global solution, or local convergence are available for most of these algorithms (Hong and Nelson 2009) (note that these definitions differ from mathematical programming where *global con-*

vergence properties ensure convergence to a *local optimum* regardless of the starting point).

Nested partition methods (Shi and Ólafsson 2007) attempt to adaptively sample from the feasible region. The feasible region is then partitioned, and sampling is concentrated in regions adjudged to be the most promising by the algorithm from a predetermined collection of nested sets. Hong and Nelson propose the COMPASS algorithm (Hong and Nelson 2006) which uses a unique neighborhood structure, defined as the most promising region that is fully adaptive rather than pre-determined; a most promising 'index' is defined that classifies each candidate solution based on a nearest neighbor metric. More recently, the Adaptive Hyberbox Algorithm (Xu et al. 2013) claims to have superior performance on high-dimensional problems (problems with more than ten or fifteen variables); and the R-SPLINE algorithm (Wang et al. 2012), which alternates between a continuous search on a continuous piecewise-linear interpolation and a discrete neighborhood search, compares favorably as well.

A review of random search methods is presented in Andradóttir (2006a), Ólafsson (2006). Recent progress, outlines of basic algorithms, and pointers to specific references for some of these methods are presented in Bianchi et al. (2009), Hong and Nelson (2009) and Nelson (2010).

Direct search methods such as pattern search and Nelder–Mead simplex methods are elaborated on in Sect. 3.5.

3.2 Response surface methodology

Response surface methodology (RSM) is typically useful in the context of continuous optimization problems and focuses on learning input-output relationships to approximate the underlying simulation by a surface (also known as a metamodel or surrogate model) for which we define a functional form. This functional form can then be made use of by leveraging powerful derivative-based optimization techniques. The literature in RSM is vast and equivalent approaches have variously been referred to as multi-disciplinary design optimization, metamodel-based optimization, and sequential parameter optimization. RSM was originally developed in the context of experimental design for physical processes (Box and Wilson 1951), but has since been applied to computer experiments. Metamodel-based optimization is a currently popular technique for addressing simulation optimization problems (Barton and Meckesheimer 2006; Kleijnen 2008).

Algorithm 2 Basic RSM procedure

Require: Initial region of approximation $\mathcal{X}$, choice of regression surface r
1: **while** not converged or under simulation budget **do**
2: Perform a design of experiments in relevant region, using k data points
3: $t_i \leftarrow \text{simulate}(x_i), \quad i = \{1, \ldots, k\}$ {Evaluate noisy function $f(x_i, \omega)$}
4: $\lambda^* \leftarrow \arg\min_\lambda \sum (t_i - r(x_i, \lambda))^2$ {Fit regression surface r through points using squared loss function}
5: $x^* \leftarrow \{\arg\min_x r(x, \lambda^*) : x \in \mathcal{X}\}$ {Optimize surface}
6: Update set of available data points and region of approximation
7: **end while**

Different response surface algorithms differ in the choice between regression and interpolation; the nature of the functional form used for approximation (polynomials, splines, Kriging, radial basis functions, neural networks); the choice of how many and where new samples must be taken; and how they update the response surface.

RSM approaches can either (1) build surrogate models that are effective in local regions, and sequentially use these models to guide the search, or; (2) build surrogate models for the entire parameter space from space-filling designs, and then use them to choose samples in areas of interest, i.e., where the likelihood of finding better solutions is good according to a specified metric. A generic framework for RSM is presented in Algorithm 2.

Classical sequential RSM Originally, RSM consisted of a Phase I, where first order models were built using samples from a design of experiments. A steepest descent rule was used to move in a certain direction, and this would continue iteratively until the estimated gradient would be close to zero. Then, a Phase II procedure that built a more detailed quadratic model would be used for verifying the optimality of the experimental design. A thorough introduction to response surface methodology is available in Myers et al. (2009). Recent work in the field includes automating RSM (Neddermeijer et al. 2000; Nicolai and Dekker 2009) and the capability to handle stochastic constraints (Angün et al. 2009).

Bayesian global optimization These methods seek to build a global response surface, commonly using techniques such as Kriging/Gaussian process regression (Sacks et al. 1989; Rasmussen and Williams 2006). Subsequent samples chosen based on some sort of improvement metric may balance exploitation and exploration. The seminal paper by Jones et al. (1998) which introduced the EGO algorithm for simulations with deterministic output, uses Kriging to interpolate between function values, and chooses future samples based on an expected improvement metric (Mockus et al. 1978). Examples of analogs to this for simulation optimization are provided in Huang et al. (2006), Kleijnen et al. (2012). The use of Kriging for simulation metamodeling is explored in van Beers and Kleijnen (2004), Kleijnen and Beers (2005), Kleijnen (2009). Other criteria that have been used to choose samples are most probable improvement (Mockus 1989), knowledge gradient for continuous parameters (Scott et al. 2011), and maximum information gain (Srinivas et al. 2012).

Trust region methods Trust region methods (Conn et al. 2000) can be used to implement sequential RSM. Trust regions provide a means of controlling the region of approximation, providing update criteria for surrogate models, and are useful in analyzing convergence properties. Once a metamodel or response surface, g, is built around a trust region center x_i, trust region algorithms involve the solution of the trust-region subproblem ($\min_s g(x_i + s) : s \in \mathcal{B}(x_i, \Delta)$), where $\mathcal{B}$ is a ball defined by the center-radius pair (x_i, Δ). There are well-defined criteria to update the trust region center and radius (Conn et al. 2000) that will define the subsequent region of approximation.

The use of trust regions in simulation optimization is relatively recent, and has been investigated to some extent (Deng and Ferris 2006; Chang et al. 2013). Trust-

region algorithms have been used, for example, to optimize simulations of urban traffic networks (Osorio and Bierlaire 2010).

3.3 Gradient-based methods

Stochastic approximation methods or gradient-based approaches are those that attempt to descend using estimated gradient information. Stochastic approximation techniques are one of the oldest methods for simulation optimization. Robbins and Monro (1951) and Kiefer and Wolfowitz (1952) were the first to develop stochastic approximation schemes in the early 1950s. These procedures initially were meant to be used under very restrictive conditions, but much progress has been made since then.

These methods can be thought of being analogous to steepest descent methods in derivative-based optimization. One may obtain direct gradients or may estimate gradients using some finite difference scheme. Direct gradients may be calculated by a number of methods: (1) Perturbation Analysis (specifically, Infinitesimal Perturbation Analysis) (PA or IPA), (2) Likelihood Ratio/Score Function (LR/SF), and (3) Frequency Domain Analysis (FDA). Detailed books on these methods are available in the literature (Ho and Cao 1991; Glasserman 1991; Rubinstein and Shapiro 1993; Pflug 1996; Fu and Hu 1997) and more high-level descriptions are available in papers (Tekin and Sabuncuoglu 2004; Fu 2002). Most of these direct methods, however, are either applicable to specific kinds of problems, need some information about underlying distributions, or are difficult to apply. Fu (2002) outlines which methods are applicable in which situations, and Tekin and Sabuncuoglu (2004) discuss a number of applications that have used these methods.

Stochastic approximation schemes attempt to estimate a gradient by means of finite differences. Typically, a forward difference estimate would involve sampling at least $n + 1$ distinct points, but superior performance has been observed by simultaneous perturbation estimates that require samples at just two points (Spall 2003), a method referred to as Simultaneous Perturbation Stochastic Approximation (SPSA). The advantage gained in SPSA is that the samples required are now independent of the problem size, and, interestingly, this has been shown to have the same asymptotic convergence rate as the naive method that requires $n+1$ points (Spall 1992). A typical gradient-based scheme is outlined in Algorithm 3.

Algorithm 3 Basic gradient-based procedure

Require: Specify initial point, x_0. Define initial parameters such as step size (α), distances between points for performing finite difference, etc.

1: $i \leftarrow 0$
2: **while** not converged or under simulation budget **do**
3: Perform required simulations, $t_i^{j_i} \leftarrow \text{simulate}(x_i)$, with j_i replications to estimate gradient, $\widehat{J}$, using either IPA, LR/SF, FDA or finite differences
4: $x_{i+1} \leftarrow x_i - \alpha \widehat{J}$
5: $i \leftarrow i + 1$
6: **end while**

Recent extensions of the SPSA method include introducing a global search component to the algorithm by injecting Monte Carlo noise during the update step (Maryak and Chin 2008), and using it to solve combined discrete/continuous optimization prob-

lems (Wang and Spall 2011). Recent work also addresses improving Jacobian as well as Hessian estimates in the context of the SPSA algorithm (Spall 2009). Much of the progress in stochastic approximation has been cataloged in the proceedings of the Winter Simulation Conference over the years (http://informs-sim.org/). A recent review of stochastic approximation methods is available in Spall (2012), and an excellent tutorial and review of results in stochastic approximation is presented in Pasupathy and Kim (2011).

3.4 Sample path optimization

Sample path optimization involves working with an estimate of the underlying unknown function, as opposed to the function itself. The estimate is usually a consistent estimator such as the sample mean of independent function evaluations at a point, or replications. For instance, one may work with $F_n = \frac{1}{n}\sum_{i=1}^{n} f(x, y, \omega_i)$, instead of the underlying function $E[f(x, y, \omega)]$ itself. It should be noted that the functional form of F_n is still unknown, it is just that F_n can be observed or evaluated at a point in the search space visited by an algorithm iteration. The alternative name of sample average approximation reflects this use of an estimator.

As the algorithm now has to work with an estimator, a deterministic realization of the underlying stochastic function, sophisticated techniques from traditional mathematical programming can now be leveraged. Sample path methods can be viewed as the use of deterministic optimization techniques within a well-defined stochastic setting. Yet another name for them is stochastic counterpart. Some of the first papers using sample path methods are Healy and Schruben (1991) and Shapiro (1991). Several papers Rubinstein and Shapiro (1993), Chen and Schmeiser (1994), Gürkan et al. (1994), Plambeck et al. (1996), Robinson (1996), Shapiro (1996), Deng and Ferris (2006) discuss convergence results and algorithms in this context.

3.5 Direct search methods

Direct search can be defined as the sequential examination of trial solutions generated by a certain strategy (Hooke and Jeeves 1961). As opposed to stochastic approximation, direct search methods rely on direct comparison of function values without attempting to approximate derivatives. Direct search methods typically rely on some sort of ranking of quality of points, rather than on function values.

Most direct search algorithms developed for simulation optimization are extensions of ideas for derivative-free optimization. A comprehensive review of classical and modern methods is provided in Kolda et al. (2003). A formal theory of direct search methods for stochastic optimization is developed in Trosset (2000). Direct search methods can be tailored for both discrete and continuous optimization settings. Pattern search and Nelder–Mead simplex procedures are the most popular direct search methods. There is some classical as well as relatively recent work done on investigating both pattern search methods (Trosset 2000; Anderson and Ferris 2001; Lucidi and Sciandrone 2002) and Nelder–Mead simplex algorithms (Nelder and Mead 1965; Barton and Ivey 1996; Humphrey and Wilson 2000; Chang 2012) and their convergence in the context of simulation optimization.

These methods remain attractive as they are relatively easy to describe and implement, and are not affected if a gradient does not exist everywhere, as they do not rely on gradient information. Since conventional procedures can be affected by noise, effective sampling schemes to control the noise are required. A basic Nelder–Mead procedure is outlined in Algorithm 4.

Algorithm 4 Basic Nelder–Mead simplex procedure for SO

Require: A set of $n-1$ points in the parameter space to form the initial simplex
1: **while** not satisfied prespecified convergence criterion or under simulation budget **do**
2: Generate a new candidate solution, x_i, through simplex centroid reflections, contractions or other means
3: $t_i^{j_i} \leftarrow \text{simulate}(x_i), \quad i = \{i-n+1, \ldots, i\}, j_i = \{1, \ldots, N_i\}$ {Evaluate noisy function $f(x, \omega)$ N_i times, where N_i is determined by some sampling scheme}
4: Calculate $\frac{\sum_{j_i} t_i^{j_i}}{N_i}$, or some similar metric to determine which point (i.e., with the highest metric value) should be eliminated
5: **end while**

3.6 Random search methods

3.6.1 Genetic algorithms

Genetic algorithms use concepts of mutation and selection from theory of evolution (Reeves 1997; Whitley 1994). In general, The genetic algorithm works by creating a population of strings and each of these strings are called chromosomes. Each of these chromosome strings is basically a vector of point in the search space. New chromosomes are created by using selection, mutation and crossover functions. The selection process is guided by evaluating the fitness (or objective function) of each chromosome and selecting the chromosomes according to their fitness values (using methods such as mapping onto Roulette Wheel). Additional chromosomes are then generated using crossover and mutation functions. The cross over and mutation functions ensures that a diversity of solutions is maintained. Genetic algorithms are popular as they are easy to implement and are used in several commercial simulation optimization software packages (Table 4). The GECCO (Genetic and Evolutionary Computation Conference) catalogs progress in genetic algorithms and implementations.

3.6.2 Simulated annealing

Simulated Annealing uses a probabilistic method that is derived from the annealing process in which the material is slowly cooled so that its structure is frozen and it reaches a minimum energy state (Kirkpatrick et al. 1983; Bertsimas and Tsitsiklis 1993). Starting with a current point i in a state j, a neighborhood point i' of the point i is generated. The algorithm moves from point i to i' using a probabilistic criteria that is dependent on the 'temperature' in state j. This temperature is analogous to that in physical annealing, and serves here as a control parameter. If the solution at i' is better than the existing solution, then this new point is accepted. If the new solution is

Table 4 Simulation optimization packages in commercial simulation software

Optimization package	Vendor	Simulation software supported	Optimization methodology
AutoStat	Applied Materials, Inc.	AutoMod	Evolutionary strategy
Evolutionary Optimizer	Imagine That, Inc.	ExtendSim	Evolutionary strategy
OptQuest	OptTek Systems, Inc.	FlexSim, @RISK, Simul8, Simio, SIMPROCESS, AnyLogic, Arena, Crystal Ball, Enterprise Dynamics, ModelRisk	Scatter search, tabu search, neural networks, integer programming
SimRunner	ProModel Corp.	ProModel, MedModel, ServiceModel	Genetic algorithms and evolutionary strategies
RISKOptimizer	Palisade Corp.	@RISK	Genetic algorithm
WITNESS Optimizer	Lanner Group, Inc.	WITNESS	Simulated annealing, tabu search, hill climbing
GoldSim Optimizer	GoldSim Technology Group	GoldSim	Box's complex method
Plant Simulation Optimizer	Siemens AG	Siemens PLM software	Genetic algorithm
ChaStrobeGA	N/A	Stroboscope	Genetic algorithm
Global Optimization toolbox	The MathWorks	SimEvents (Matlab)	Genetic algorithms, simulated annealing, pattern search

worse than existing solution, then the probability of accepting the point is defined as $\exp(-(f(i') - f(i))/T(j))$, where $f(.)$ is the value of objective function at a given point, and $T(j)$ is temperature at the state j. After a certain number of neighborhood points are evaluated, the temperature is decreased and new state is $j + 1$ is created. Due to the exponential form, the probability of acceptance of a neighborhood point is higher at high temperature, and is lower as temperature is reduced. In this way, the algorithm searches for a large number of neighborhood points in the beginning, but a lower number of points as temperature is reduced.

Implementation of simulated annealing procedures require choosing parameters such as the initial and final temperatures, the rate of cooling, and number of function evaluations at each temperature. A variety of cooling 'schedules' have been suggested in Collins et al. (1988) and Hajek (1988). Though simulated annealing was originally meant for optimizing deterministic functions, the framework has been extended to the case of stochastic simulations (Alkhamis et al. 1999). The ease of implementing a simulated annealing procedure is high and it remains a popular technique used by several commercial simulation optimization packages.

3.6.3 Tabu search

Tabu search (Glover 1990) uses special memory structures (short-term and long-term) during the search process that allow the method to go beyond local optimality to

explore promising regions of the search space. The basic form of tabu search consists of a modified neighborhood search procedure that employs adaptive memory to keep track of relevant solution history, together with strategies for exploiting this memory (Gendreau and Potvin 2010). More advanced forms of tabu search and its applications are described in Glover and Laguna (1997).

3.6.4 Scatter search

Scatter search and its generalized form, path relinking, were originally introduced by Glover and Laguna (2000). Scatter search differs from other evolutionary approaches (such as Genetic Algorithms (GA)) by using strategic designs and search path construction from a population of solutions as compared to randomization (by crossover and mutation in GA). Similar to Tabu search, Scatter Search also utilize adaptive memory in storing best solutions (Glover and Laguna 2000; Martí et al. 2006). Algorithm 5 provides the scatter search algorithm.

Algorithm 5 Basic scatter search procedure for SO

Require: An initial set of trial points $x \in P$, chosen to be diversified according to a pre-specified metric
1: $t_j \leftarrow$ simulate(x_j), where $j = 1, \ldots, |P|$
2: $k \leftarrow 0$
3: Use a comparison procedure (such as ranking and selection) to gather the best b solutions (based on objective value or diversity) from the current set of solutions P, called the reference set, R_k
4: $R_{-1} = \varnothing$
5: **while** under simulation budget and $R_k \neq R_{k-1}$ **do**
6: $k \leftarrow k + 1$
7: Choose $S_i \subset R$, where $i = 1, \ldots, r$ {Use a subset generation procedure to select r subsets of set R, to be used as a basis for generating new solution points}
8: **for** $i = 1$ to r **do**
9: Combine the points in S_i, to form new solution points, x_j, where $j \in \mathcal{J} = |P|+1, \ldots, |P|+J$, using weighted linear combinations, for example
10: $t_j \leftarrow$ simulate(x_j), $j \in \mathcal{J}$ {sample the objective function at new trial solutions}
11: Update sets R_k, P
12: **end for**
13: **end while**

3.7 Model-based methods

Model-based simulation optimization methods attempt to build a probability distribution over the space of solutions and use it to guide the search process.

3.7.1 Estimation of distribution algorithms

Estimation of distribution algorithms (EDAs) (Larrañaga and Lozano 2002) are model-based methods that belong to the evolutionary computation field. However, generation of new candidate solutions is done by sampling from the inferred probability distribution over the space of solutions, rather than, say, a genetic operator such as crossover or mutation. A comprehensive review of estimation of distribution algorithms is pre-

sented in Fu et al. (1996). EDAs usually consider interactions between the problem variables and exploit them through different probability models.

Cross-entropy methods and Model Reference Adaptive Search (MRAS) are discussed next and can be seen as specific instances of EDAs.

Cross-Entropy Methods Cross-entropy methods first sample randomly from a chosen probability distribution over the space of decision variables. For each sample, which is a vector defining a point in decision space, a corresponding function evaluation is obtained. Based on the function values observed, a pre-defined percentile of the best samples are picked. A new distribution is built around this 'elite set' of points via maximum likelihood estimation or some other fitting method, and the process is repeated. One possible method that implements cross-entropy is formally described in Algorithm 6.

Algorithm 6 Pseudocode for a simple cross-entropy implementation

Require: θ, an initial set of parameters for a pre-chosen distribution $p(x;\theta)$ over the set of decision variables; k, a number of simulations to be performed; e, the number of elite samples representing the top δ percentile of the k samples

while not converged or under simulation budget **do**
  for $i = 1 \rightarrow k$ **do**
    sample x_i from $p(x;\theta)$
    $t_i \leftarrow \text{simulate}(x_i)$
  end for
  $E \leftarrow \emptyset$
  for $i = 1 \rightarrow e$ **do**
    $E_j \leftarrow \arg\max_{i \notin E} t_i$
  end for
  $p(x;\theta) \leftarrow \text{fit}(x_E)$
end while

The method is guaranteed (probabilistically) to converge to a local optimum, but it also incorporates an exploration step as random samples are obtained at each step. However, the intuition behind the selection of subsequent samples can be shown to be analogous to minimizing the Kullback–Leibler divergence (KL-divergence) between the optimal importance sampling distribution and the distribution used in the current iterate (Rubinstein and Kroese 2004).

There exist variants of the cross-entropy method to address both continuous (Kroese et al. 2006) and discrete optimization (Rubinstein 1999) problems. A possible modification is to use mixtures of distributions from current and previous iterations, with the current distribution weighted higher. This can be done by linearly interpolating the mean covariance in the case of Gaussian distributions. This also helps in avoiding singular covariance matrices. Cross-entropy can also deal with noisy function evaluations, with irrelevant decision variables, and constraints (Kroese et al. 2006). If decision variables are correlated, the covariance of the distribution will reflect this.

The immediately apparent merits of cross-entropy methods are that they are easy to implement, require few algorithmic parameters, are based on fundamental principles such as KL-divergence and maximum likelihood, and give consistently accurate

results (Kroese et al. 2006). A potential drawback is that cross-entropy may require a significant number of new samples at every iteration. It is not clear as to how this would affect performance if samples were expensive to obtain. The cross-entropy method has analogs in simulated annealing, genetic algorithms, and ant colony optimization, but differs from each of these in important ways (de Boer et al. 2005).

More detailed information on the use of cross-entropy methods for optimization can be found in de Boer et al. (2005), a tutorial on cross-entropy and in Rubinstein and Kroese (2004), a monograph. The cross-entropy webpage, http://iew3.technion.ac.il/CE/ provides up-to-date information on progress in the field.

Model reference adaptive search (MRAS) The MRAS method (Hu et al. 2005, 2007) is closely related to the cross-entropy method. It also works by minimizing the Kullback–Leibler divergence to update the parameters of the inferred probability distribution. However, the parameter update step involves the use of a sequence of implicit probability distributions. In other words, while the cross-entropy method uses the optimal importance sampling distribution for parameter updates, MRAS minimizes the KL-divergence with respect to the distribution in the current iteration, called the reference model.

Covariance Matrix Adaptation–Evolution Strategy (CMA-ES) In the CMA-ES algorithm (Hansen 2006), new samples are sampled from a multivariate normal distribution, and inter-variable dependencies are encoded in the covariance matrix. The CMA-ES method provides a way to update the covariance matrix. Updating the covariance matrix is analogous to learning an approximate inverse Hessian, as is used in Quasi-Newton methods in mathematical programming. The update of the mean and covariance is done by maximizing the likelihood of previously successful candidate solutions and search steps, respectively. This is in contrast to other EDAs and the cross-entropy method, where the covariance is updated by maximizing the likelihood of the successful points. Other sophistications such as step-size control, and weighting of candidate solutions are part of modern implementations (Hansen 2011).

3.7.2 Ant colony optimization

Ant colony optimization methods (Dorigo and Stützle 2004; Dorigo and Blum 2005) are heuristic methods that have been used for combinatorial optimization problems. Conceptually, they mimic the behavior of ants to find shortest paths between their colony and food sources. Ants deposit pheromones as they walk; and are more likely to choose paths with higher concentration of pheromones. This phenomenon is incorporated in a pheromone update rule, which increases the pheromone content in components of high-quality solutions, and causes evaporation of pheromones in less favorable regions. Probability distributions are used to make the transition between each iteration. These methods differ from EDAs in that they use an iterative construction of solutions.

This and other algorithms that incorporate self-organization in biological systems are said to use the concept of ‘swarm intelligence’.

3.8 Lipschitzian optimization

Lipschitzian optimization is a class of space-partitioning algorithms for performing global optimization, where the Lipschitz constant is pre-specified. This enables the construction of global search algorithms with convergence guarantees. The caveat of having prior knowledge of the Lipschitz constant is overcome by the DIRECT (DIviding RECTangles) algorithm (Jones et al. 1993) for deterministic continuous optimization problems. An adaptation of this for noisy problems is provided in Deng and Ferris (2007).

4 Software

4.1 Simulation optimization in commercial simulation software

Many discrete-event simulation packages incorporate some methodology for performing optimization. A comprehensive listing of simulation software, the corresponding vendors, and the optimization packages and techniques they use can be found in Table 4. More details on the specific optimization routines can be found in Law and Kelton (2000).OR/MS-Today, the online magazine of INFORMS, conducts a biennial survey of simulation software packages, the latest of which is available at (http://www.orms-today.org/surveys/Simulation/Simulation.html). The survey lists 43 simulation software packages, and 31 of these have some sort of optimization routine; fewer still have black-box optimizers that interact with the simulation.

4.2 Academic implementations of simulation optimization

Table 5 contains a small subset of academic implementations of SO algorithms, and classifies them by type. Some of these are available for download from the web, some have code with suggested parameters in corresponding papers themselves, and others are available upon request from the authors.

5 Comparison of algorithms

As far as comparisons between algorithms are concerned, the literature does not yet provide a comprehensive survey of the performance of different implementations and approaches on large test beds. In this regard, simulation optimization lags behind other optimization fields such as linear, integer, and nonlinear programming, global optimization and even derivative-free optimization, where the first comprehensive comparison appeared in Rios and Sahinidis (2013). A study of prior comparisons in simulation optimization is provided by Tekin and Sabuncuoglu (2004), but these comparisons are fairly dated, are inconclusive about which algorithms perform better in different situations, and compare only a small subset of available algorithms. One difficulty lies in the inherent difficulty of comparing solutions between algorithms over true black-box simulations, as one does not usually know the true optimal point and

Table 5 Academic simulation optimization implementations

Algorithm	Type	Citation
Continuous		
SPSA	Stochastic approximation	Spall (2003)
SPSA 2nd Order	Stochastic approximation	Spall (2003)
SKO	Global response surface	Huang et al. (2006)
CE method	Cross-entropy	Kroese et al. (2006)
APS	Nested partitioning	Kabirian and Ólafsson (2007)
SNOBFIT	Multi-start local response surface	Huyer and Neumaier (2008)
CMA-ES	Evolutionary strategy	Hansen (2011)
KGCP	Global response surface	Scott et al. (2011)
STRONG	Local response surface, trust region	Chang et al. (2013)
GR	Golden region search	Kabirian and Ólafsson (2011)
SNM	Direct search (Nelder–Mead)	Chang (2012)
DiceOptim	Global response surface	Roustant et al. (2012)
Discrete		
KG	Global response surface	Frazier et al. (2009)
COMPASS	Neighborhood search (integer-ordered problems)	Xu et al. (2010)
R-SPLINE	Neighborhood search (integer-ordered problems)	Wang et al. (2012)
Discrete and continuous		
MRAS	Estimation of distribution	Hu et al. (2005, 2007)
NOMADm	Mesh adaptive direct search	Abramson (2007)

can only compare between noisy estimates observed by the solvers. Less impeding difficulties, but difficulties nonetheless, include the need to interface algorithms to a common wrapper, the objective comparison with solvers that incorporate random elements as their results may not be reproducible, and lack of standard test simulations for purposes of benchmarking.

The benchmarking of algorithms in mathematical programming is usually done by performance profiles (Dolan and Moré 2002), where the graphs show the fraction of problems solved after a certain time. For derivative-free algorithms, data profiles are commonly used (Moré and Wild 2009), where the fraction of problems solved after a certain number of iterations (function evaluations) or 'simplex gradients' is shown. The definition of when a problem is 'solved' may vary—when the true global optimum is known, the solutions found within a certain tolerance of this optimal value may be called solutions, but when this optimum is not known, the solvers that find the best solution (within a tolerance) for a problem, with respect to the other solvers being compared, may be said to have solved the problem. The latter metric may also be used when function evaluations are expensive, and no solver is able to reach within this tolerance given the limited simulation budget.

In both of these cases, the output of the simulations are deterministic, and so it is clear as to which algorithms have performed better than others on a particular problem. In simulation optimization, however, usually one does not know the true solution for the black box system, nor does one see deterministic output. All that one possesses are mean values and sample variances obtained from sample paths at different points. There does not exist a standard method to compare simulation optimization algorithms on large test beds. Many papers perform several macroreplications and report the macroreplicate average of the best sample means (along with the associated sample variance) at the end of the simulation budget. The issue with this is that the performance of the algorithms with different simulation budgets is not seen, as in the case of performance or data profiles. Other papers report the average number of evaluations taken to find a sample mean that is within the global tolerance for each problem. Here, results are listed for each problem and one does not get an idea of overall performance. In addition, the difference in sample variance estimates is not highlighted. As simulation optimization develops, there is also a need for methods of comparison of algorithms on test beds with statistically significant number of problems.

With regard to standardized simulation testbeds, to our knowledge, the only testbed that provides practical simulations for testing simulation optimization algorithms is available at http://www.simopt.org (Pasupathy and Henderson 2011). At the point of writing this paper, just 20 continuous optimization problems were available from this repository. Most testing and comparisons happen with classical test problems in nonlinear optimization (many of which have been compiled in Rios and Sahinidis (2013) and available at http://archimedes.cheme.cmu.edu/?q=dfocomp), to which stochastic noise has been added. There is a need for more such repositories, not only for testing of algorithms over statistically significant sizes of problem sets, but for comparison between different classes of algorithms. The need for comparison is evident, given the sheer number of available approaches to solving simulation optimization problems, and the lack of clarity and lack of consensus on which types of algorithms are suitable in which contexts.

As observed by several papers (Fu et al. 2000; Tekin and Sabuncuoglu 2004; Hong and Nelson 2009), there continues to exist a significant gap between research and practice in terms of algorithmic approaches. Optimizers bundled with simulation software, as observed in Sect. 4, tend to make use of algorithms which seem to work well but do not come with provable statistical properties or guarantees of local or global convergence. Academic papers, on the other hand, emphasize methods that are more sophisticated and prove convergence properties. One reason that may contribute to this is that very few simulation optimization algorithms arising from the research community are easily accessible. We wholeheartedly encourage researchers to post their executable files, if not their source code. This could not only encourage practitioners to use these techniques in practice, but allow for comparisons between methods and the development of standardized interfaces between simulations and simulation optimization software.

6 Conclusions

The field of simulation optimization has progressed significantly in the last decade, with several new algorithms, implementations, and applications. Contributions to the field arise from researchers and practitioners in the industrial engineering/operations research, mathematical programming, statistics and machine learning, as well as the computer science communities. The use of simulation to model complex, dynamic, and stochastic systems has only increased with computing power and availability of a wide variety of simulation languages. This increased use is reflected in the identification and application of simulation and simulation optimization methods to diverse fields in science, engineering, and business. There also exist strong analogies between, and ideas that may be borrowed from recent progress in related fields. All of these factors, along with the ever increasing number of publications and rich literature in this area, clearly indicate the interest in the field of simulation optimization, and we have tried to capture this in this paper.

With increased growth and interest in the field, there are also arise opportunities. Potential directions for the field of simulation optimization are almost immediately apparent. Apart from the ability to handle simulation outputs from any well-defined probability distribution, the effective use of variance reduction techniques when possible, and the improvement in theory and algorithms, there is a requirement to address (1) large-scale problems with combined discrete/continuous variables; (2) the ability to effectively handle stochastic and deterministic constraints of various kinds; (2) the effective utilization of parallel computing at the linear algebra level, sample replication level, iteration level, as well as at the algorithmic level; (3) the effective handling of multiple simulation outputs; (4) the incorporation of performance measures other than expected values, such as risk; (5) the continued consolidation of various techniques and their potential synergy in hybrid algorithms; (6) the use of automatic differentiation techniques in the estimation of simulation derivatives when possible; (7) the continued emphasis on providing guarantees of convergence to optima for local and global optimization routines in general settings; (8) the availability and ease of comparison of the performance of available approaches on different applications; and (9) the continued reflection of sophisticated methodology arising from the literature in commercial simulation packages.

References

Abramson MA (2007) NOMADm version 4.5 user's guide. Air Force Institute of Technology, Wright-Patterson AFB, OH

Alkhamis TM, Ahmed MA, Tuan VK (1999) Simulated annealing for discrete optimization with estimation. Eur J Oper Res 116:530–544

Alrefaei MH, Andradóttir S (1999) A simulated annealing algorithm with constant temperature for discrete stochastic optimization. Manag Sci 45:748–764

Ammeri A, Hachicha W, Chabchoub H, Masmoudi F (2011) A comprehensive literature review of mono-objective simulation optimization methods. Adv Prod Eng Manag 6(4):291–302

Anderson EJ, Ferris MC (2001) A direct search algorithm for optimization with noisy function evaluations. SIAM J Optim 11:837–857

Andradóttir S (1998) Chapter 9: Simulation optimization. In: Banks J (ed) Handbook of simulation: principles, methodology, advances, applications, and practice. Wiley, New York

Andradóttir S (2006a) An overview of simulation optimization via random search. In: Henderson SG, Nelson BL (eds) Handbooks in operations research and management science: simulation, vol 13, chap 20. Elsevier, Amsterdam, pp 617–631
Andradóttir S (2006b) Simulation optimization. In: Handbook of simulation: principles, methodology, advances, applications and practice. Wiley, New York, pp 307–333
Andradóttir S, Kim SH (2010) Fully sequential procedures for comparing constrained systems via simulation. Naval Res Logist 57(5):403–421
Angün E (2004) Black box simulation optimization: generalized response surface methodology. Ph.D. thesis, Tilburg University
Angün E, Kleijnen JPC, Hertog DD, Gurkan G (2009) Response surface methodology with stochastic constraints for expensive simulation. J Oper Res Soc 60(6):735–746
Ayvaz MT (2010) A linked simulation-optimization model for solving the unknown groundwater pollution source identification problems. J Contam Hydrol 117(1–4):46–59
Azadivar F (1992) A tutorial on simulation optimization. In: Swain JJ, Goldsman D, Crain RC, Wilson JR (eds) Proceedings of the 1992 winter simulation conference, pp 198–204
Azadivar J (1999) Simulation optimization methodologies. In: Farrington PA, Nembhard HB, Sturrock DT, Evans GW (eds) Proceedings of the 1999 winter simulation conference, pp 93–100
Balakrishna R, Antoniou C, Ben-Akiva M, Koutsopoulos HN, Wen Y (2007) Calibration of microscopic traffic simulation models: methods and application. Transp Res Rec J Transp Res Board 1999(1):198–207
Bangerth W, Klie H, Matossian V, Parashar M, Wheeler MF (2005) An autonomic reservoir framework for the stochastic optimization of well placement. Clust Comput 8(4):255–269
Barton RR, Ivey JS Jr (1996) Nelder–Mead simplex modifications for simulation optimization. Manag Sci 42:954–973
Barton RR, Meckesheimer M (2006) Metamodel-based simulation optimization. In: Henderson S, Nelson B (eds) Handbook in operations research and management science: simulation 13. Elsevier, Amsterdam, pp 535–574
Bechhofer RE, Santner TJ, Goldsman DM (1995) Design and analysis of experiments for statistical selection, screening, and multiple comparisons. Wiley, New York
Bertsimas D, Tsitsiklis J (1993) Simulated annealing. Stat Sci 8(1):10–15
Bettonvil B, del Castillo E, Kleijnen JPC (2009) Statistical testing of optimality conditions in multiresponse simulation-based optimization. Eur J Oper Res 199:448–458
Bhatnagar S (2005) Adaptive multivariate three-timescale stochastic approximation algorithms for simulation based optimization. ACM Trans Model Comput Simul (TOMACS) 15(1):74–107
Bianchi L, Dorigo M, Gambardella LM, Gutjahr WJ (2009) A survey on metaheuristics for stochastic combinatorial optimization. Nat Comput 8(2):239–287
Birge JR, Louveaux F (2011) Introduction to stochastic programming, 2nd edn. Springer, Berlin
Box GEP, Wilson KB (1951) On the experimental attainment of optimum conditions. J R Stat Soc XII XIII(1):1–35
Carson Y, Maria A (1997) Simulation optimization: Methods and applications. In: Andradóttir S, Healy KJ, Winters DH, Nelson BL (eds) Proceedings of the 1997 winter simulation conference, pp 118–126
Chang KH (2008) Stochastic trust region response surface convergent method for continuous simulation optimization. Ph.D. thesis, Purdue University
Chang KH (2012) Stochastic Nelder–Mead simplex method-A new globally convergent direct search method for simulation optimization. Eur J Oper Res 220:684–694
Chang KH, Hong LJ, Wan H (2013) Stochastic trust-region response-surface method (STRONG): a new response-surface framework for simulation optimization, vol 25(2), pp 230–243
Chen CH (1995) An effective approach to smartly allocate computing budget for discrete event simulation. In: Proceedings of the 34th IEEE conference on decision and control, pp 2598–2605
Chen CH (1996) A lower bound for the correct subset selection probability and its application to discrete event system simulations. IEEE Trans Autom Control 41:1227–1231
Chen CH, Lee LH (2010) Stochastic simulation optimization: an optimal computing budget allocation. System engineering and operations research. World Scientific, Singapore
Chen H, Schmeiser BW (1994) Retrospective optimization algorithms for stochastic root finding. In: Tew J, Manivannan S, Sadowski D, Seila A (eds) Proceedings of 1994 winter simulation conference, pp 255–261

Chen CH, Yücesan E, Dai L, Chen HC (2009) Optimal budget allocation for discrete-event simulation experiments. IIE Trans 42(1):60–70
Chick SE (2006) Subjective probability and bayesian methodology. In: Henderson SG, Nelson BL (eds) Simulation, handbooks in operations research and management science, vol 13. Elsevier, Amsterdam, pp 225–257
Cho J, Dorfman KD (2010) Brownian dynamics simulations of electrophoretic DNA separations in a sparse ordered post array. J Chromatog A 1217:5522–5528
Cohn DA, Ghahramani Z, Jordan MI (1996) Active learning with statistical models. J Artif Intell Res 4:129–145
Collins NE, Eglese RW, Golden BL (1988) Simulated annealing—an annotated bibliography. Am J Math Manag Sci 8:209–308
Conn AR, Gould NIM, Toint PL (2000) Trust-region methods. MOS-SIAM series on optimization
Conn AR, Scheinberg K, Vicente LN (2009) Introduction to derivative-free optimization. SIAM, Philadelphia
de Angelis V, Felici G, Impelluso P (2003) Integrating simulation and optimisation in health care centre management. Eur J Oper Res 150:101–114
de Boer PT, Kroese DP, Mannor S, Rubinstein RY (2005) A tutorial on the cross-entropy method. Ann Oper Res 134:19–67
Deng G (2007) Simulation-based optimization. Ph.D. thesis, University of Wisconsin-Madison
Deng G, Ferris MC (2006) Adaptation of the UOBYQA algorithm for noisy functions. In: Perrone LF, Wieland FP, Liu J, Lawson BG, Nicol DM, Fujimoto RM (eds) Proceedings of the 2006 winter simulation conference, pp 312–319
Deng G, Ferris MC (2007) Extension of the DIRECT optimization algorithm for noisy functions. In: Henderson SG, Biller B, Hsieh MH, Shortle J, Tew JD, Barton RR (eds) Proceedings of the 2007 winter simulation conference, pp 497–504
Dengiz B, Akbay KS (2000) Computer simulation of a PCB production line: metamodeling approach. Int J Prod Econ 63(2):195–205
Dhivya M, Sundarambal M, Anand LN (2011) Energy efficient computation of data fusion in wireless sensor networks using cuckoo-based particle approach (cbpa). Int J Commun Netw Syst Sci 4(4):249–255
Dolan ED, Moré JJ (2002) Benchmarking optimization software with performance profiles. Math Program 91:201–213
Dorigo M, Blum C (2005) Ant colony optimization theory: a survey. Theor Comput Sci 344(2–3):243–278
Dorigo M, Stützle T (2004) Ant colony optimization. MIT Press, Cambridge
Driessen LT (2006) Simulation-based optimization for product and process design. Ph.D. thesis, Tilburg University
Ernst D, Glavic M, Stan GB, Mannor S, Wehenkel L (2007) The cross-entropy method for power system combinatorial optimization problems. In: Power tech, pp 1290–1295. IEEE
Ferris MC, Deng G, Fryback DG, Kuruchittham V (2005) Breast cancer epidemiology: calibrating simulations via optimization. Oberwolfach Rep 2:9023–9027
Figueira G, Almada-Lobo B (2014) Hybrid simulation-optimization methods: a taxonomy. Simul Model Pract Theory 46:118–134
Frazier PI (2009) Knowledge-gradient methods for statistical learning. Ph.D. thesis, Princeton University
Frazier P, Powell W, Dayanik S (2009) The knowledge-gradient policy for correlated normal beliefs. INFORMS J Comput 21(4):599–613
Fu MC (1994) Optimization via simulation: a review. Ann Oper Res 53:199–247
Fu MC (2002) Optimization for simulation: theory vs practice. INFORMS J Comput 14(3):192–215
Fu MC, Hill SD (1997) Optimization of discrete event systems via simulataneous perturbation stochastic approximation. IIE Trans 29(233-243)
Fu MC, Hu JQ (1997) Conditional Monte Carlo: gradient estimation and optimization applications. Kluwer, Dordrecht
Fu MC, Hu J, Marcus SI (1996) Model-based randomized methods for global optimization. In: Proceedings of the 17th international symposium on mathematical theory of networks and systems, Kyoto, Japan, pp 355–363
Fu MC, Andradóttir S, Carson JS, Glover FW, Harrell CR, Ho YC, Kelly JP, Robinson SM (2000) Integrating optimization and simulation: research and practice. In: Joines JA, Barton RR, Kang K, Fishwick PA (eds) Proceedings of the 2000 winter simulation conference

Fu MC, Glover FW, April J (2005) Simulation Optimization: a review, new developments, and applications. In: Kuhl ME, Steiger NM, Armstrong FB, Joines JA (eds) Proceedings of the 2005 winter simulation conference, pp 83–95
Gendreau M, Potvin JY (2010) Tabu search. In: Handbook of metaheuristics, international series in operations research & management science, vol 146, 2nd ed. Springer, Berlin, pp 41–60
Gerencsér L, Kozmann G, Vágó Z, Haraszti K (2002) The use of the SPSA method in ECG analysis. IEEE Trans Biomed Eng 49(10):1094–1101
Gittins JC (1989) Multi-armed bandit allocation indices. Wiley-interscience series in systems and optimization. Wiley, New York
Glasserman P (1991) Gradient estimation via perturbation analysis. Kluwer, Dordrecht
Glover F (1990) Tabu search: a tutorial. Interfaces 20(4):77–94
Glover F, Hanafi S (2002) Tabu search and finite convergence. Discret Appl Math 119(1–2):3–36
Glover F, Laguna M (1997) Tabu search. Kluwer, Boston
Glover F, Laguna M (2000) Fundamentals of scatter search and path relinking. Control Cybern 29(3):653–684
Goldsman D, Nelson BL (1998) Comparing systems via simulation. In: Banks J (ed) Handbook of simulation: principles, methodology, advances, applications, and practice, chap. 8. Wiley, New York
Gong WB, Ho YC, Zhai W (2000) Stochastic comparison algorithm for discrete optimization with estimation. SIAM J Optim 10:384–404 (49)
Griewank A, Walther A (2008) Evaluating derivatives: principles and techniques of algorithmic differentiation, 2nd ed. No. 105 in other titles in applied mathematics. SIAM, Philadelphia, PA. http://www.ec-securehost.com/SIAM/OT105.html
Gürkan G, Ozge AY, Robinson SM (1994) Sample path optimization in simulation. In: Tew J, Manivannan S, Sadowski D, Seila A (eds) Proceedings of 1994 winter simulation conference, pp 247–254
Hajek B (1988) Cooling schedules for optimal annealing. Math Oper Res 13:311–329
Hall JD, Bowden RO, Usher JM (1996) Using evolution strategies and simulation to optimize a pull production system. J Mater Process Technol 61(1–2):47–52
Hansen N (2006) The CMA evolution strategy: a comparing review. In: Lozano JA, Larrañaga P, Inza I, Bengoetxea E (eds) Towards a new evolutionary computation. Advances on estimation of distribution algorithms. Springer, Berlin, pp 75–102
Hansen N (2011) The CMA Evolution strategy: a tutorial. http://www.lri.fr/hansen/cmaesintro.html
Healy K, Schruben LW (1991) Retrospective simulation response optimization. In: Nelson BL, Kelton DW, Clark GM (eds) Proceedings of the 1991 winter simulation conference, pp 954–957
Hill SD, Fu MC (1995) Transfer optimization via simultaneous perturbation stochastic approximation. In: Alexopoulos C, Kang K, Lilegdon WR, Goldsman D (eds) Proceedings of the 1995 winter simulation conference, pp 242–249
Ho YC (1999) An explanation of ordinal optimization: soft computing for hard problems. Inf Sci 113:169–192
Ho YC, Cao XR (1991) Discrete event dynamic systems and perturbation analysis. Kluwer, Dordrecht
Hochberg Y, Tamhane AC (1987) Multiple comparison procedures. Wiley, New York
Hong LJ, Nelson BL (2006) Discrete optimization via simulation using COMPASS. Oper Res 54(1):115–129
Hong LJ, Nelson BL (2009) A brief introduction to optimization via simulation. In: Rossetti MD, Hill RR, Johansson B, Dunkin A, Ingalls RG (eds) Proceedings of the 2009 winter simulation conference
Hooke R, Jeeves TA (1961) Direct search solution of numerical and statistical problems. J Assoc Comput Mach 8:212–219
Hsu JC (1996) Multiple comparisons: theory and methods. CRC Press, Boca Raton
Hu J, Fu MC, Marcus SI (2005) Stochastic optimization using model reference adaptive search. In: Kuhl ME, Steiger NM, Armstrong FB, Joines JA (eds) Proceedings of the 2005 winter simulation conference, pp 811–818
Hu J, Fu MC, Marcus SI (2007) A model reference adaptive search method for global optimization. Oper Res 55(3):549–568
Huang D, Allen TT, Notz WI, Zeng N (2006) Global optimization of stochastic black-box systems via sequential kriging meta-models. J Glob Optim 34:441–466
Humphrey DG, Wilson JR (2000) A revised simplex search procedure for stochastic simulation response-surface optimization. INFORMS J Comput 12(4):272–283

Hunter SR, Pasupathy R (2013) Optimal sampling laws for stochastically constrained simulation optimization on finite sets. INFORMS J Comput 25(3):527–542
Hutchison DW, Hill SD (2001) Simulation optimization of airline delay with constraints. In: Peters BA, Smith JS, Medeiros DJ, Rohrer MW (eds) Proceedings of the 2001 winter simulation conference, pp 1017–1022
Huyer W, Neumaier A (2008) SNOBFIT—stable noisy optimization by branch and fit. ACM Trans Math Softw 35:1–25
Irizarry MDLA, Wilson JR, Trevino J (2001) A flexible simulation tool for manufacturing-cell design, II: response surface analysis and case study. IIE Trans 33(10):837–846
Jacobson SH, Schruben LW (1989) Techniques for simulation response optimization. Oper Res Lett 8:1–9
Jia QS, Ho YC, Zhao QC (2006) Comparison of selection rules for ordinal optimization. Math Comput Model 43(9–10):1150–1171
Jones DR, Perttunen CD, Stuckman BE (1993) Lipschitzian optimization without the Lipschitz constant. J Optim Theory Appl 79:157–181
Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. J Glob Optim 13:455–492
Jung JY, Blau G, Pekny JF, Reklaitis GV, Eversdyk D (2004) A simulation based optimization approach to supply chain management under demand uncertainty. Comput Chem Eng 28:2087–2106
Kabirian A (2009) Continuous optimization via simulation using golden region search. Ph.D. thesis, Iowa State University
Kabirian A, Ólafsson S (2007) Allocation of simulation runs for simulation optimization. In: Henderson SG, Biller B, Hsieh MH, Shortle J, Tew JD, Barton RR (eds) Proceedings of the 2007 winter simulation conference, pp 363–371
Kabirian A, Ólafsson S (2011) Continuous optimization via simulation using golden region search
Kenne JP, Gharbi A (2001) A simulation optimization approach in production planning of failure prone manufacturing systems. J Intell Manuf 12:421–431
Khan HA, Zhang Y, Ji C, Stevens CJ, Edwards DJ, O'Brien D (2006) Optimizing polyphase sequences for orthogonal netted radar. IEEE Signal Process Lett 13(10):589–592
Kiefer J, Wolfowitz J (1952) Stochastic estimation of the maximum of a regression function. Ann Math Stat 23(3):462–466
Kim SH (2005) Comparison with a standard via fully sequential procedures. ACM Trans Model Comput Simul (TOMACS) 15(2):155–174
Kim SH, Nelson BL (2006) Selecting the best system. In: Henderson SG, Nelson BL (eds) Handbooks in operations research and management science: simulation, chap 17. Elsevier, Amsterdam, pp 501–534
Kim SH, Nelson BL (2007) Recent advances in ranking and simulation. In: Henderson SG, Biller B, Hsieh MH, Shortle J, Tew JD, Barton RR (eds) Proceedings of the 2007 winter simulation conference, pp 162–172
Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220:671–680
Kleijnen JPC (1993) Simulation and optimization in production planning: a case study. Decis Support Syst 9:269–280
Kleijnen JPC (2008) Design and analysis of simulation experiments. Springer, New York
Kleijnen JPC (2009) Kriging metamodeling in simulation: a review. Eur J Oper Res 192(3):707–716
Kleijnen JPC, van Beers WCM (2005) Robustness of kriging when interpolating in random simulation with heterogeneous variances: some experiments. Euro J Oper Res 165:826–834
Kleijnen JPC, Beers WCM, van Nieuwenhuyse I (2012) Expected improvement in efficient global optimization through bootstrapped kriging. J Glob Optim 54(1):59–73
Kleinman NL, Hill SD, Ilenda VA (1997) SPSA/SIMMOND optimization of air traffic delay cost. In: Proceedings of the 1997 American control conference, vol 2, pp 1121–1125
Köchel P, Nieländer U (2005) Simulation-based optimisation of multi-echelon inventory systems. Int J Prod Econ 93–94:505–513
Kolda TG, Lewis RM, Torczon VJ (2003) Optimization by direct search: new perspectives on some classical and modern methods. SIAM Rev 45:385–482
Kothandaraman G, Rotea MA (2005) Simultaneous-perturbation-stochastic-approximation algorithm for parachute parameter estimation. J Aircr 42(5):1229–1235
Kroese DP, Porotsky S, Rubinstein RY (2006) The cross-entropy method for continuous multi-extremal optimization. Methodol Comput Appl Probab 8(3):383–407

Kroese DP, Hui KP, Nariai S (2007) Network reliability optimization via the cross-entropy method. IEEE Trans Reliab 56(2):275–287
Kulturel-Konak S, Konak A (2010) Simulation optimization embedded particle swarm optimization for reliable server assignment. In: Johansson B, Jain S, Montoya-Torres J, Hugan J, Yücesan E (eds) Proceedings of the 2010 winter simulation conference, pp 2897–2906
Larrañaga P, Lozano JA (2002) Estimation of distribution algorithms: a new tool for evolutionary computation. Kluwer, Dordrecht
Lau TWE, Ho YC (1997) Universal alignment probabilities and subset selection for ordinal optimization. J Optim Theory Appl 93(3):455–489
Law AM, Kelton WD (2000) Simulation modeling and analysis, 3rd edn. McGraw-Hill, Singapore
Lee LH, Pujowidianto NA, Li LW, Chen CH, Yap CM (2012) Approximate simulation budget allocation for selecting the best design in the presence of stochastic constraints. IEEE Trans Autom Control 57(11):2940–2945
Li Y (2009) A simulation-based evolutionary approach to LNA circuit design optimization. Appl Math Comput 209(1):57–67
Lucidi S, Sciandrone M (2002) On the global convergence of derivative-free methods for unconstrained minimization. SIAM J Optim 13:97–116
Lutz CM, Davis KR, Sun M (1998) Determining buffer location and size in production lines using tabu search. Eur J Oper Res 106:301–316
Martí R, Laguna M, Glover F (2006) Principles of scatter search. Eur J Oper Res 169(2):359–372
Maryak JL, Chin DC (2008) Global random optimization by simulataneous perturbation stochastic approximation. IEEE Trans Autom Control 53:780–783
Meketon MS (1987) Optimization in simulation: a survey of recent results. In: Thesen A, Grant H, Kelton WD (eds) Proceedings of the 1987 winter simulation conference, pp 58–67
Merhof D, Soza G, Stadlbauer A, Greiner G, Nimsky C (2007) Correction of susceptibility artifacts in diffusion tensor data using non-linear registration. Med Image Anal 11(6):588–603
Merton RC (1974) On the pricing of corporate debt: the risk structure of interest rates. J Financ 29(2):449–470
Mishra V, Bhatnagar S, Hemachandra N (2007) Discrete parameter simulation optimization algorithms with applications to admission control with dependent service times. In: Proceedings of the 46th IEEE conference on decision and control, New Orleans, LA, pp 2986–2991
Mockus J (1989) Bayesian approach to global optimization. Kluwer, Dordrecht
Mockus J, Tiesis V, Zilinskas A (1978) Towards global optimisation, vol. 2, chap. The application of Bayesian methods for seeking the extremum. North-Holland, Amsterdam
Moré J, Wild S (2009) Benchmarking derivative-free optimization algorithms. SIAM J Optim 20:172–191
Myers RH, Montgomery DC, Anderson-Cook CM (2009) Response surface methodology: process and product optimization using designed experiments. wiley series in probability and statistics. Wiley, New York
Neddermeijer HG, Oortmarssen GJV, Piersma N, Dekker R (2000) A framework for response surface methodology for simulation optimization. In: Joines JA, Barton RR, Kang K, Fishwick PA (eds) Proceedings of the 2000 winter simulation conference, pp 129–136
Nelder JA, Mead R (1965) A simplex method for function minimization. Comput J 7:308–313
Nelson BL (2010) Optimization via simulation over discrete decision variables. Tutor Oper Res 7:193–207
Nelson BL, Goldsman D (2001) Comparisons with a standard in simulation experiments. Manag Sci 47(3):449–463
Nicolai R, Dekker R (2009) Automated response surface methodology for simulation optimization models with unknown variance. Qual Technol Qual Manag 6(3):325–352
Ólafsson S (2006) Metaheuristics. In: Henderson S, Nelson B (eds) Handbook in operations research and management science: simulation, vol 13. Elsevier, Amsterdam, pp 633–654
Osorio C, Bierlaire M (2010) A simulation-based optimization approach to perform urban traffic control. In: Proceedings of the triennial symposium on transportation analysis
Pasupathy R, Ghosh S (2013) Simulation optimization: a concise overview and implementation guide. Tutor Oper Res 10:122–150
Pasupathy R, Henderson SG (2011) SIMOPT: a library of simulation-optimization problems. In: Jain S, Creasey RR, Himmelspach J, White KP, Fu M (eds) Proceedings of the 2011 winter simulation conference

Pasupathy R, Kim S (2011) The stochastic root finding problem: overview, solutions, and open questions. ACM Trans Model Comput Simul (TOMACS) 21(3):19:1–19:23
Peters J, Vijayakumar S, Schaal S (2003) Reinforcement learning for humanoid robotics. In: Third IEEE-RAS international conference on humanoid robots, Karlsruhe, Germany, pp 1–20
Pflug GC (1996) Optimization of stochastic models: the interface between simulation and optimization. Kluwer, Dordrecht
Plambeck EL, Fu BR, Robinson SM, Suri R (1996) Sample-path optimization of convex stochastic performance functions. Math Program 75(2):137–176
Powell WB (2013) http://www.castlelab.princeton.edu/cso.htm. Accessed 23 Oct 2013
Powell WB, Ryzhov IO (2012) Optimal learning. Wiley, New York
Prakash P, Deng G, Converse MC, Webster JG, Mahvi DM, Ferris MC (2008) Design optimization of a robust sleeve antenna for hepatic microwave ablation. Phys Med Biol 53:1057–1069
Radac MB, Precup RE, Petriu EM, Preitl S (2011) Application of ift and SPSA to servo system control. IEEE Trans Neural Netw 22(12):2363–2375
Rall, LB (1981) Automatic differentiation: techniques and applications, lecture notes in computer science, vol 120. Springer, Berlin. doi:10.1007/3-540-10861-0
Ramanathan SP, Mukherjee S, Dahule RK, Ghosh S, Rahman I, Tambe SS, Ravetkar DD, Kulkarni BD (2001) Optimization of continuous distillation columns using stochastic optimization approaches. Trans Inst Chem Eng 79:310–322
Rasmussen CE, Williams CKI (2006) Gaussian processes for machine learning. MIT Press, Cambridge
Reeves CR (1997) Genetic algorithms for the operations researcher. INFORMS J Comput 9(3):231–250
Renotte C, Vande Wouwer A (2003) Stochastic approximation techniques applied to parameter estimation in a biological model. In: Proceedings of the second IEEE international workshop on Intelligent data acquisition and advanced computing systems: technology and applications, 2003, IEEE, pp 261–265
Rios LM, Sahinidis NV (2013) Derivative-free optimization: a review of algorithms and comparison of software implementations. J Glob Optim 56:1247–1293
Robbins H, Monro S (1951) A stochastic approximation method. Ann Math Stat 22(3):400–407
Robinson SM (1996) Analysis of sample-path optimization. Math Oper Res 21(3):513–528
Romero PA, Krause A, Arnold FH (2013) Navigating the protein fitness landscape with gaussian processes. Proc Natl Acad Sci (PNAS) 110(3). doi:10.1073/pnas.1215251110
Roustant O, Ginsbourger D, Deville Y (2012) Dicekriging, diceoptim: two r packages for the analysis of computer experiments by kriging-based metamodeling and optimization. J Stat Softw 51(1):1–55
Rubinstein R (1999) The cross-entropy method for combinatorial and continuous optimization. Methodol Comput Appl Probab 1:127–190
Rubinstein RY, Kroese DP (2004) The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning. Springer, New York
Rubinstein RY, Shapiro A (1993) Discrete event systems: sensitivity analysis and stochastic optimization by the score function method. Wiley, New York
Sacks J, Schiller SB, Welch WJ (1989) Designs for computer experiments. Technometrics 31:41–47
Safizadeh MH (1990) Optimization in simulation: current issues and the future outlook. Naval Res Logist 37:807–825
Sahinidis NV (2004) Optimization under uncertainty: State-of-the-art and opportunities. Comput Chem Eng 28(6–7):971–983
Schwartz JD, Wang W, Rivera DE (2006) Simulation-based optimization of process control policies for inventory management in supply chains. Automatica 42:1311–1320
Scott W, Frazier PI, Powell W (2011) The correlated knowledge gradient for simulation optimization of continuous parameters using gaussian process regression. SIAM J Optim 21(3):996–1026
Settles B (2010) Active learning literature survey. Tech. rep., University of Wisconsin-Madison
Shapiro A (1991) Asymptotic analysis of stochastic programs. Ann Oper Res 30:169–186
Shapiro A (1996) Simulation based optimization. In: Charnes JM, Morrice DJ, Brunner DT, Swain JJ (eds) Proceedings of the 1996 winter simulation conference, pp 332–336
Shi L, Ólafsson S (2000) Nested partitions method for stochastic optimization. Methodol Comput Appl Probab 2:271–291
Shi L, Ólafsson (2007) Nested partitions optimization: methodology and applications, international series in operations research & management science, vol 109. Springer, Berlin
Song Y, Grizzle JW (1995) The extended kalman filter as a local asymptotic observer for discrete-time nonlinear systems. J Math Syst Estim Control 5(1):59–78

Spall JC (1992) Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Trans Autom Control 37:332–341
Spall JC (2003) Introduction to stochastic search and optimization: Estimation, simulation, and control. Wiley-Interscience
Spall JC (2009) Feedback and weighting mechanisms for improving Jacobian estimates in the adaptive simultaneous perturbation algorithm. IEEE Trans Autom Control 54(6):1216–1229
Spall JC (2012) Stochastic optimization. In: Gentle JE, Härdle WK, Mori Y (eds) Handbook of computational statistics: concepts and methods, 2nd ed, chap 7. Springer, Berlin, pp 173–201
Srinivas N, Krause A, Kakade SM, Seeger M (2012) Information-theoretic regret bounds for gaussian process optimization in the bandit setting. IEEE Trans Inf Theory 58(5):3250–3265
Stephens CP, Baritompa W (1998) Global optimization requires global information. J Optim Theory Appl 96:575–588
Swisher JR, Hyden PD, Jacobson SH, Schruben LW (2000) A survey of simulation optimization techniques and procedures. In: Joines JA, Barton RR, Kang K, Fishwick PA (eds) Proceedings of the 2000 winter simulation conference
Syberfeldt A, Lidberg S (2012) Real-world simulation-based manufacturing optimization using cuckoo search. In: Laroque C, Himmelspach J, Pasupathy R, Rose O, Uhrmacher A (eds) Proceedings of the 2012 winter simulation conference
Tein LH, Ramli R (2010) Recent advancements of nurse scheduling models and a potential path. In: Proceedings of the 6th IMT-GT conference on mathematics, statistics and its applications, pp 395–409
Tekin E, Sabuncuoglu I (2004) Simulation optimization: a comprehensive review on theory and applications. IIE Trans 36:1067–1081
Teng S, Lee LH, Chew EP (2007) Multi-objective ordinal optimization for simulation optimization problems. Automatica 43(11):1884–1895
Trosset MW (2000) On the use of direct search methods for stochastic optimization. Tech. rep., Rice University, Houston, TX
van Beers AC, Kleijnen JPC (2004) Kriging interpolation in simulation: a survey. In: Proceedings of the 2004 winter simulation conference, vol 1, pp 121–129
Vande Wouwer A, Renotte, Bogaerts P, Remy M (2001) Application of SPSA techniques in nonlinear system identification. In: Proceedings of the European control conference, p 2835
Wang Q, Spall JC (2011) Discrete simultaneous perturbation stochastic approximation on loss functions with noisy measurements. In: Proceedings of the American control conference. IEEE, San Francisco, pp 4520–4525
Wang H, Pasupathy R, Schmeiser BW (2012) Integer-ordered simulation optimization using R-SPLINE: retrospective search with piecewise-linear interpolation and neighborhood enumeration. ACM Trans Model Comput Simul (TOMACS) 23:17:1–17:24
Whitley D (1994) A genetic algorithm tutorial. Stat Comput 4:65–85
Xie J, Frazier PI (2013) Sequential bayes-optimal policies for multiple comparisons with a known standard. Oper Res 61(5):1174–1189
Xie J, Frazier PI, Sankaran S, Marsden A, Elmohamed S (2012) Optimization of computationally expensive simulations with gaussian processes and parameter uncertainty: application to cardiovascular surgery. In: 50th Annual allerton conference on communication, control, and computing
Xing XQ, Damodaran M (2002) Assessment of simultaneous perturbation stochastic approximation method for wing design optimization. J Aircr 39:379–381
Xing XQ, Damodaran M (2005a) Application of simultaneous perturbation stochastic approximation method for aerodynamic shape design optimization. AIAA J 43(2):284–294
Xing XQ, Damodaran M (2005b) Inverse design of transonic airfoils using parallel simultaneous perturbation stochastic approximation. J Aircr 42(2):568–570
Xu J, Nelson BL, Hong LJ (2010) Industrial strength COMPASS: a comprehensive algorithm and software for optimization via simulation. ACM Trans Model Comput Simul (TOMACS) 20(1):1–29
Xu J, Nelson BL, Hong LJ (2013) An adaptive hypberbox algorithm for high-dimensional discrete optimization via simulation problems. INFORMS J Comput 25(1):133–146
Yalçinkaya Ö, Mirac Bayhan G (2009) Modelling and optimization of average travel time for a metro line by simulation and response surface methodology. Eur J Oper Res 196:225–233
Yan D, Mukai H (1992) Stochastic discrete optimization. SIAM J Control Optim 30:594–612

Yang XS, Deb S (2010) Engineering optimisation by cuckoo search. Int J Math Model Numer Optim 1(4):330–343

Yeomans JS (2007) Solid waste planning under uncertainty using evolutionary simulation–optimization. Socio-Econ Plan Sci 41:38–60

Yun I, Park B (2010) Application of stochastic optimization method for an urban corridor. In: Perrone LF, Wieland FP, Liu J, Lawson BG, Nicol DM, Fujimoto RM (eds) Proceedings of the 2010 winter simulation conference, pp 1493–1499