Working Paper

Sample-Path Optimization in Simulation

Gül Gürkan
A. Yonca Özge
Stephen M. Robinson

WP-94-70
July 1994

IIASA International Institute for Applied Systems Analysis □ A-2361 Laxenburg □ Austria
Telephone: +43 2236 71521 □ Telex: 079 137 iiasa a □ Telefax: +43 2236 71313

Sample-Path Optimization in Simulation

Gül Gürkan
A. Yonca Özge
Stephen M. Robinson

WP-94-70
July 1994

International Institute for Applied Systems Analysis □ A-2361 Laxenburg □ Austria
Telephone: +43 2236 71521 □ Telex: 079 137 iiasa a □ Telefax: +43 2236 71313

Sample-Path Optimization in Simulation

Gül Gürkan
A. Yonca Özge
Stephen M. Robinson[1]

Abstract

This paper summarizes information about a method, called *sample-path optimization,* for optimizing performance functions in certain stochastic systems that can be modeled by simulation. We explain the method, give conditions under which it converges, and display some sample calculations that indicate how it performs. We also describe briefly some more extensive numerical experiments on large systems (PERT networks with up to 110 stochastic arcs, and tandem production lines with up to 50 machines). Details of these experiments are reported elsewhere; we give references to this and other related work. We conclude with some currently unanswered questions.

1 What is sample-path optimization?

In this section we explain briefly what sample-path optimization is, and give some reasons why it might be useful in practical situations. We also sketch the related work that has appeared in the literature, and give references where fuller discussions appear.

A general model for many problems in simulation optimization is an extended-real-valued stochastic process $\{X_n \mid n = 1, 2, \ldots\}$, where the X_n are random variables depending on parameters $\theta \in \mathbf{R}^k$. That is, for each n and each θ, $X_n(\theta)$ is a random variable defined on the probability space $(\Omega, \mathcal{F}, P)$, and it may take either real values or the value $+\infty$ (sometimes also $-\infty$). This is a convenient device for modeling constraints, because we can set $X_n(\theta) = +\infty$ for those θ that do not satisfy whatever constraints we have to work with. When a particular sample point ω is fixed, we write $X_n(\omega, \theta)$ to display the dependence on the sample point.

For simulation to be of interest we generally have some *a priori* knowledge that there exists a deterministic function $X_\infty(\theta)$ such that the functions X_n almost surely converge pointwise to X_∞ as $n \to \infty$. For example, in the case of static systems the strong law of large numbers provides this kind of information when X_n is the average of n independent replications. In the case of dynamic systems such as queues, if we think of X_n as the output of a simulation run of length n (*e.g.,* n service completions), we can often infer the existence of X_∞ from regeneration theorems. In this paper we *assume* that X_∞ exists for the systems we are considering. Our interest will be to find the infimum of X_∞ and, if possible, a value of θ at which that infimum is attained (a *minimizer* of X_∞).

Of course, in general we cannot observe X_∞, but only $X_n(\theta)$ for particular (finite) n and particular θ. Therefore, we have to use such observations to approximate the minimizer and minimum value that we are seeking. The method that we consider here, which we call sample-path optimization, is particularly simple: we fix n and the sample ω, and use deterministic optimization methods to find a minimizer θ_n^* of $X_n(\omega, \cdot)$ (assuming

[1]Authors' address: Department of Industrial Engineering, University of Wisconsin–Madison, 1513 University Avenue, Madison, WI 53706-1572, U.S.A. Email: smr@cs.wisc.edu

one exists; we shall give conditions in Section 2 under which it will). We simply take this θ_n^* as an estimate of a minimizer of X_∞.

This idea has several attractive features, as well as some that may be less attractive. First, in many cases we can apply very powerful methods to minimize $X_n(\omega, \cdot)$, even if the constraints on θ are numerous or complicated. The reason for this is that by using existing simulation methodology, such as the method of infinitesimal perturbation analysis (IPA), we can compute not only the value $X_n(\omega, \theta)$ for a given θ, but also its *exact* gradient (when the gradient exists). Therefore we can apply fast (superlinearly-convergent) methods, such as the BFGS algorithm and its relatives for constrained optimization, which may permit us to minimize $X_n(\omega, \cdot)$ to high accuracy in relatively few function and gradient evaluations. See, for example, Moré and Wright (1993) for a description of some of these methods and for further references.

The availability of very effective methods for constrained deterministic optimization is one reason why sample-path optimization could be effective on problems with many variables and/or complicated constraints, with which simple gradient-step methods may have difficulty. In particular, superlinearly-convergent methods enable one to be fairly confident of the location and accuracy of the optimizer of $X_n(\omega, \cdot)$, and thereby to separate the error due to the approximation of X_∞ by X_n from the error due to inaccurate computation of a minimizer of X_n. With slower algorithms, such as stochastic approximation or stochastic quasigradient methods, such separation is likely to be difficult or impossible.

Second, because the method separates optimization from the computation of function and gradient values it is well suited to modular implementation. For example, codes already in existence for simulating particular systems could be used as part of an implementation (provided that they can accommodate some method of gradient evaluation, such as IPA), together with available optimization codes that call external subroutines for function and gradient evaluation. The simulation and the optimization codes need not interact with each other except through the function and gradient values that the first code supplies to the second. This modularity can be a substantial advantage if the system being simulated is large and complex, and/or if the optimization method being used is fairly sophisticated.

On the other hand, such modularity has some apparent disadvantages. For example, at this time we do not know how to apply sample-path optimization to multistage (dynamic) stochastic optimization problems in such a way as to take advantage of the time structure of such problems. This is an area of current research; several others are mentioned in Section 4. With numerous open questions such as these, the overall effectiveness of sample-path optimization is yet to be determined. Numerical experience to date is quite encouraging, but there is still much that we do not know about the method and its behavior.

In the rest of this section we give a brief survey of some existing literature related to this method. The method as described here has been analyzed and applied in three papers, two by Plambeck *et al.* (1993, 1994) and the other by Robinson (1994). The second of these gave extensive computational results, to which we return in Section 3 below; the third presented a convergence analysis. We shall extract parts of that analysis in Section 2 immediately following this one.

A method very similar in concept, but different in execution, is the technique of retrospective optimization proposed by Healy and Schruben (1991). Healy and Xu (1994) later analyzed this technique and gave some additional computational results. A key difference between the method we treat here and that studied by Healy and Xu (1994)

is that in their computational method they propose to store the information generated during a simulation run, then operate on that stored information to compute a minimizer. By contrast, we do not envision storing the information, but rather making repeated simulation runs using the method of common random numbers to ensure that we observe the same function $X_n(\omega, \cdot)$ on each run. Chen and Schmeiser (1994) applied retrospective optimization methodology to the problem of stochastic root finding.

King and Wets (1991) studied a method of this type applied to solution of stochastic programming problems, and in particular to linear recourse problems. They imposed additional technical requirements, suitable to the class of problems they considered, and obtained convergence results and related information. Their problem class was somewhat different from that considered here; on the other hand, Shapiro and Wardi (1994) and Rubinstein and Shapiro (1990, 1993) considered simulation optimization problems using likelihood ratio (LR) methods of gradient estimation. In these cases one of the main operational differences is the method of computing gradients: for many of the problems we envision solving by sample-path optimization, the easiest and cheapest way of computing gradients is by IPA.

This section has briefly described the sample-path optimization method and has outlined connections with existing literature. In the next section we look at conditions under which the method can be shown to be *a priori* convergent: that is, under which as we lengthen the sample path indexed by n, the computed optimizer θ_n^* could be expected to approach the set of optimizers of the limit function X_∞.

2 When does it work?

This section summarizes the conditions under which we can prove that the sample-path optimization method will converge. We do not give any proofs, but we indicate where these can be found.

It is clear that the problem of convergence in sample-path optimization is that of determining when, and how quickly, a sequence of optimizers of $X_n(\omega, \cdot)$ will converge to an optimizer of X_∞. The situation is complicated by the fact that ordinary pointwise convergence of a sequence of functions is well known to be insufficient for convergence of their optimizers to optimizers of the limit function. What is needed, generally speaking, is *epiconvergence,* which we now define.

Definition 1 *A sequence f_n of extended-real-valued functions defined on $\mathbf{R}^k$* epiconverges *to an extended-real-valued function f_∞ defined on $\mathbf{R}^k$ if for each $\theta \in \mathbf{R}^k$ the following hold:*

a. For each sequence $\{\theta_n\}$ converging to θ, $f_\infty(\theta) \leq \liminf_{n\to\infty} f_n(\theta_n)$.

b. For some sequence $\{\theta_n\}$ converging to θ, $f_\infty(\theta) \geq \limsup_{n\to\infty} f_n(\theta_n)$.

In (b) we actually have $f_\infty(\theta) = \lim_{n\to\infty} f_n(\theta_n)$, because of (a).

It is known that epiconvergence is independent of pointwise convergence. For a very readable elementary treatment of relationships among various notions of convergence, see Kall (1986); a comprehensive treatment of epiconvergence and related issues is in Attouch (1984). The forthcoming book of Rockafellar and Wets (1994) will treat this area from the perspective of optimization.

A rough (and incorrect) summary of the convergence properties of this method would be the statement that if the functions X_n almost surely epiconverge to the function X_∞, then local minimizers of the X_n almost surely approach local minimizers of X_∞ as $n \to \infty$. Unfortunately, things are not quite so simple. To state the conditions precisely we need

more terminology, in particular the concept of *complete local minimizing set* (Robinson 1987), which extends the idea of isolated local minimizer to cases in which the set of minimizers might not be a singleton. The symbol "cl" denotes the closure of a set.

Definition 2 *Let Z be a topological space and let f be an extended-real-valued function on Z. A nonempty subset M of Z is a* complete local minimizing (CLM) set *for f with respect to an open set $G \supset M$, if the set of minimizers of f on* cl G *is M.*

We say a function is *proper* if it never takes $-\infty$ and does not always take $+\infty$. We also use the notation $e(S,T)$ for the *excess* of a set $S \subset \mathbf{R}^k$ over a set $T \subset \mathbf{R}^k$: that is,

$$e(S,T) = \sup_{s \in S} d(s,T); \qquad d(s,T) = \inf_{t \in T} \|s - t\|.$$

If $e(S,T)$ is small, then *each* point of S is close to *some* point of T, even though T may be much larger than S.

The following theorem summarizes the convergence properties of this method. It does not give the most general statement possible, but rather tries to strike a balance between generality on the one hand and simplicity on the other. For more general results see Robinson (1994); the theorem here is a combination of Theorem 3.7 and Propositions 3.8 and 3.9 of that paper.

Theorem 1 *Suppose that the following assumptions hold:*

a. X_∞ is a proper deterministic function whose infimum is μ_∞ and whose set of minimizers M_∞ is nonempty and compact.

b. With probability one, each X_n $(1 \leq n < \infty)$ is lower semicontinuous and proper.

c. With probability one, the X_n epiconverge to X_∞ as $n \to \infty$.

Let G be any open bounded subset of $\mathbf{R}^k$ containing M_∞. Then there is a subset Γ of Ω having measure zero, with the following properties: suppose that $\omega \notin \Gamma$, and define for $1 \leq n < \infty$

$$\hat{\mu}_n(\omega) = \inf_{\theta \in \text{cl } G} X_n(\omega, \cdot),$$

and

$$\hat{M}_n(\omega) = \{\theta \in \text{cl } G \mid X_n(\omega,\theta) = \hat{\mu}_n(\omega)\}.$$

One then has

1. $\lim_{n\to\infty} \hat{\mu}_n(\omega) = \mu_\infty$, and μ_∞ is finite.

2. There is a finite positive integer N_ω such that for each $n \geq N_\omega$, $\hat{M}_n(\omega)$ is a nonempty, compact CLM set for $X_n(\omega, \cdot)$ with respect to G.

3. $\lim_{n\to\infty} e(\hat{M}_n(\omega), M_\infty) = 0$.

If in addition X_∞ is convex and for each n X_n is almost surely convex, then the above conclusions hold with $G = \mathbf{R}^k$.

Theorem 1 describes exact minimization, but it is also possible to obtain results for approximate minimization; these can be useful in connection with numerical methods that compute points near, but not equal to, the actual minimizer. For these results we refer to Section 4 of Robinson (1994).

A key hypothesis of Theorem 1 is that the X_n almost surely epiconverge to X_∞. It would be of interest to have general results, for classes of stochastic optimization problems arising in applications, to guarantee such epiconvergence. However, we give here a particular set of assumptions that, while rather special, will cover the examples of Section 3. This result is a specialization of Proposition 2.5 of Robinson (1994). The *effective domain* of an extended-real-valued function f, written dom f, is the set of points $\theta \in \mathbf{R}^k$ at which $f(\theta) < +\infty$.

Proposition 1 *Suppose that the following hypotheses hold:*

a. X_∞ *is a deterministic closed proper convex function whose effective domain has a nonempty interior.*

b. For each n, X_n *is with probability one a closed proper convex function.*

c. There is a countable dense subset Θ *of* $\mathbf{R}^k$ *such that whenever* $\theta \in \Theta$, *then with probability one* $X_n(\theta)$ *converges to* $X_\infty(\theta)$ *as* $n \to \infty$.

Then with probability one the X_n *epiconverge to* X_∞ *as* $n \to \infty$.

The requirement that dom X_∞ have a nonempty interior does not prevent us from dealing with problems in which the presence of linear constraints may cause the effective domain of X_∞ to have an affine hull A of dimension less than k, provided that the effective domains of the X_n also lie in A. We then simply apply Proposition 1 to the restrictions of the functions X_n and X_∞ to the set A; the technique is described in more detail in Section 2 of Robinson (1994).

We have summarized briefly the known results about when the method of sample-path optimization converges, in the sense that minimizers of the approximating functions X_n almost surely approach the set of minimizers of the limit function X_∞ as $n \to \infty$. In the process, we saw that a key requirement was that the X_n epiconverge to X_∞, and we showed that for the particular case of convex functions, this epiconvergence hypothesis held under mild assumptions. We now turn from the question of theoretical convergence to that of actual numerical convergence: that is, does the method work? The following section presents a few numerical results to indicate what the method can do, and refers to much more comprehensive numerical experiments reported in Plambeck *et al.* (1994).

3 How well does it work?

In this section we summarize the available computational evidence for the effectiveness of sample-path optimization. We illustrate some interesting aspects of the method's performance by giving numerical results for small problems, and we describe the results of applying the method to much larger problems of different types.

The example problem we chose for the numerical illustrations was a closed queueing network with four servers. Customers move through Server 1, then go either to Server 2 (with probability .3) or to Server 3 (with probability .7). After this they go through Server 4, and then return to Server 1. Service time at Server i is exponentially distributed with parameter θ_i (a decision variable). Calculations presented here are for this system with 20 customers, and we assumed that the buffers at each server could accommodate all customers, so that blocking did not occur. The quantity of interest for optimization is the steady-state throughput $T(\theta)$ consisting of service completions at Server 4 per unit time. The decision variable is the vector θ consisting of the four service rates.

The numerical results reported in this section were obtained using the deterministic nonlinear optimization code NLPQL (Schittkowski 1985/86). We thank Prof. Dr. K. Schittkowski, Universität Bayreuth, Germany, for making this code available to us. The code determined the total number of simulation runs to make, one run being required each time either one or both elements of a function/gradient pair were to be computed. This decision was controlled by the accuracy parameter ACC in the code, which we set to 10^{-4}.

In order to use NLPQL one must be able to compute both the objective function value and its gradient. To compute the gradients we used the IPA method for closed queueing networks described in Suri (1989) and Leung (1990). IPA enabled us to obtain

the gradients in a single simulation run. The effort expended in the IPA calculations was negligible compared to the effort required for the simulation, and the only storage requirement was for a 4×4 matrix.

Table 1 shows the results from unconstrained minimization of the function

$$X_n(\theta) = 400T(\theta)^{-1} + \sum_{i=1}^{4} \theta_i.$$

We chose this functional form to model a problem in which one wants to maximize throughput, but in which there is some cost (in this case, 1) for increasing the service rate of a server. The optimization problem then is to find the best tradeoff of cost against throughput.

Table 1: Unconstrained, Closed Queueing Network

n	i	θ_1	θ_2	θ_3	θ_4	z
1+4	20	12.080	4.078	8.617	12.069	.715
1+5	20	12.010	4.153	8.717	12.084	.942
1+6	16	12.057	4.111	8.758	12.066	.980
2+6	16	12.048	4.115	8.752	12.058	.954
3+6	15	12.048	4.106	8.757	12.043	.938
4+6	15	12.052	4.106	8.754	12.054	.937
5+6	16	12.047	4.119	8.754	12.050	.951

Each row of this table shows the final result of optimizing X_n for a distinct n, including the simulation run length n (consisting of the total number of service completions at Server 4, with the notation $a + b$ for $a \times 10^b$), the total number i of simulation runs of length n required, the final values of the θ_i, and the quantity z consisting of the objective function value less its integer part (which was 73). In each case the starting point was $\theta_i = 10$ for all i. We also ran the optimization code starting from $\theta_i = 1$; the number of iterations was larger, as would be expected.

We also experimented with a constrained version of the same problem, in which we minimized $400T(\theta)^{-1}$ subject to the three inequality constraints

$$\theta_1 \leq 15, \qquad \theta_4 \leq 14, \qquad \sum_{i=1}^{4} \theta_i \leq 50.$$

The results are shown in Table 2, in which the quantities shown are the same as those shown in Table 1 except that z is the objective function value reduced by a different integer part (29 instead of 73). The starting point and the tolerance ACC for these computations were the same as for those of Table 1.

We suspected that the number of iterations required for this constrained optimization would be less than that required for the unconstrained problem, because of the well known dimensionality reduction produced by active constraints (with four variables and three active constraints one is effectively minimizing in a subspace of dimension 1). This was in fact the case, as one can see by comparing the values of i in the two tables. This phenomenon, incidentally, provides another argument for handling constraints directly, rather than by devices such as unconstrained minimization combined with projection.

Table 2: Constrained, Closed Queueing Network

n	i	θ_1	θ_2	θ_3	θ_4	z
1+4	10	15.000	6.898	14.102	14.000	.40
1+5	8	15.000	6.955	14.045	14.000	.51
1+6	9	15.000	7.013	13.987	14.000	.55

One of the particularly interesting aspects of these two tables is the rather small change in the solution produced by a large increase in computational effort. For example, in Table 1 the solutions produced using a total of 2 million and 80 million service completions are not very different (2nd and 7th lines of table). Similarly, in the first and third lines of Table 2 one sees that an increase from 100,000 to 9 million service completions produced relatively little change. This suggests that even a fairly small computational effort may produce a solution accurate enough for practical purposes. As we note below, a similar phenomenon appeared in the numerical experiments carried out, on much larger systems, by Plambeck *et al.* (1994).

These computational examples may help to give some insight into how the method behaves in particular cases; however, they do not demonstrate that it will be useful in solving larger or more complex problems. Solution of such problems was the object of the work reported in Plambeck *et al.* (1993, 1994), which we now briefly describe.

This work dealt with two different types of problems: tandem queues, modeling manufacturing lines, and stochastic PERT (Program Evaluation and Review Technique) networks, modeling complex projects in which some activities must be completed before others can begin. The tandem queues consisted of servers running at deterministic speeds but subject to breakdowns. The volume of product processed until breakdown and the time to repair after a breakdown were random. The servers were separated by buffers of fixed sizes, and the presence of these buffers subjected the servers to blockage and/or starvation resulting from the behavior of the servers downstream or upstream, respectively. The authors of Plambeck *et al.* (1993, 1994) optimized lines with up to 50 servers, so that simulation time was an important issue. To increase the speed of the simulations, they employed continuous flow models instead of discrete models, using methods described by Suri and Fu (1991, 1994). The objective function to be minimized was the reciprocal of throughput, the decision variables were machine cycle times (reciprocals of speeds) and these were subject to linear equation and/or inequality constraints.

The PERT networks studied in Plambeck *et al.* (1994) contained up to 70 nodes and 110 stochastic arcs. Each stochastic arc had a duration specified by a probability distribution that was either uniform (with variable mean) or triangular (with a variable scale factor in the maximum, minimum, and mode). The problem was to minimize the sum of (1) expected project duration and (2) a cost term of the form $\sum_i k_i z_i^{-1}$, where the k_i were specified numbers and the z_i were the variable quantities in the probability distributions just described. The z_i were also subject to linear inequality constraints.

As both the limit function X_∞ and the approximating functions X_n in problems of these types are convex but may be nonsmooth, it was necessary to use a method designed for nondifferentiable convex optimization; in this case, the bundle/trust region method of Schramm and Zowe (Zowe 1989, Schramm and Zowe 1990). Some heuristic modifications were also made to the basic sample-path optimization method in order to re-use information from past simulations.

Complete numerical results for these systems appear in Plambeck *et al.* (1994); however, we note that in these computations, as in the small examples illustrated above, substantial increases in computation time resulted in fairly small relative changes in the decision variables. In most of these problems no analytic formulas for the optimal solutions are known, so one cannot tell for sure whether this means that the earlier solutions (those with smaller computation times) were fairly accurate or that none of the computed solutions was accurate. However, in the one case (2-server tandem queues) for which analytic formulas were available, the solutions computed using sample-path optimization were very accurate (errors ranging from 1.8×10^{-4} to 1.4×10^{-15}), and this is an encouraging sign.

This section has presented computational results using sample-path optimization for some small queueing networks, and has briefly summarized much more extensive results available elsewhere in the literature. In the concluding section we discuss some of the areas where we think more information about this method would be very helpful.

4 Some unanswered questions

Here we list a number of questions about the sample-path optimization method to which we currently lack answers. Research is currently underway which we hope will answer some of these questions.

First, consider the question of whether the method works at all: that is, whether optimizers of X_n converge to optimizers of X_∞ as n becomes large. We dealt with this question in Section 2; recall that a key hypothesis of Theorem 1 in that section was that the X_n almost surely epiconverge to X_∞. As we indicated above, it would be of great interest to have general results, for classes of stochastic optimization problems arising in applications, to guarantee such epiconvergence.

Second, if the method in fact converges then one can reasonably ask how fast the convergence is likely to be. This means that we would like to have some indication of the *rate* of convergence of an optimizer θ_n^* of X_n to the set of optimizers of X_∞ as n increases. There is a considerable amount of recent work on asymptotic analysis applicable to stochastic programming, some of which may help to answer this question; for example, King (1989), King and Rockafellar (1993), Shapiro (1993). However, since this work focuses on the convergence in distribution of solutions to stochastic optimization problems, and since we are interested in the behavior of the solutions along a single sample path, we do not yet know whether these results can yield rate information of the kind we need.

Finally, quite apart from the theoretical questions of convergence and rate of convergence, there is the practical question of whether this method can contribute, over the long run, to the solution of significant problems that otherwise could not be solved at all, or could not be solved so well. Clearly, we do not know the answer to this question now, and no mathematical investigation will provide it. It will be necessary to have more computational experience, on a wide selection of problems from applications, before we have an indication of the method's overall usefulness.

Nevertheless, as we argued in Section 1 above, we think this technique has very attractive aspects for – in particular – the solution of constrained problems. It makes available for stochastic optimization problems the immense amount of work invested over the past 50 years in learning how to deal with constraints in numerical optimization. We hope that further work, both numerical and theoretical, will clarify the question of whether that availability can be translated into a broadly effective family of solution methods for

important stochastic optimization problems.

Acknowledgments

The research reported here was sponsored by the National Science Foundation under Grant CCR-9109345, and by the Air Force Systems Command, USAF, under Grant F49620-93-1-0068. The U. S. Government has certain rights in this material, and is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon.

This paper was written at the International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria, during the third author's visit there in July 1994. The support of IIASA and of its Project on Optimization Under Uncertainty, led by Prof. Dr. Andrzej Ruszczyński, are gratefully acknowledged.

References

Attouch, H. 1984. *Variational Convergence for Functions and Operators.* Boston: Pitman.

Chen, H. and B. W. Schmeiser. 1994. Stochastic root finding: Problem definition, examples, and algorithms. In *Proceedings of the 3rd Industrial Engineering Research Conference,* ed. L. Burke and J. Jackman, 605–610. Institute of Industrial Engineers, Atlanta, Georgia.

Healy, K. and L. W. Schruben. 1991. Retrospective simulation response optimization. In *Proceedings of the 1991 Winter Simulation Conference,* ed. B.L. Nelson, W.D. Kelton, and G.M. Clark, 901–906. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Healy, K. J. and Y. Xu. 1994. Simulation based retrospective approaches to stochastic system optimization. Preprint, Purdue University, West Lafayette, Indiana.

Kall, P. 1986. Approximation to optimization problems: An elementary review. *Mathematics of Operations Research* 11:9–18.

King, A. J. 1989. Generalized delta theorems for multivalued mappings and measurable selections. *Mathematics of Operations Research* 14:720–736.

King, A. J. and R. T. Rockafellar. 1993. Asymptotic theory for solutions in statistical estimation and stochastic programming. *Mathematics of Operations Research* 18:148–162.

King, A. J. and R. J-B Wets. 1991. Epi-consistency of convex stochastic programs. *Stochastics and Stochastics Reports* 34:83–92.

Leung, Y.–T. 1990. Single-run optimization of discrete-event simulations. Ph.D. Dissertation, Department of Industrial Engineering, University of Wisconsin–Madison, Madison, Wisconsin.

Moré, J., and S. J. Wright. 1993. *Optimization Software Guide.* Frontiers in Applied Mathematics Vol. 14, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania.

Plambeck, E. L., B.–R. Fu, S. M. Robinson, and R. Suri. 1993. Throughput optimization in tandem production lines via nonsmooth programming. In *Proceedings of 1993 Summer Computer Simulation Conference,* ed. J. Schoen, 70–75. Society for Computer Simulation, San Diego, California.

Plambeck, E. L., B.–R. Fu, S. M. Robinson, and R. Suri. 1994. Sample-path optimization of convex stochastic performance functions. Preprint, Department of Industrial Engineering, University of Wisconsin–Madison, Madison, Wisconsin.

Robinson, S. M. 1987. Local epi-continuity and local optimization. *Mathematical Programming* 37:208–222.

Robinson, S. M. 1994. Analysis of sample-path optimization. Preprint, Department of Industrial Engineering, University of Wisconsin–Madison, Madison, Wisconsin.

Rockafellar, R. T. and R. J–B Wets. 1994. *Variational Analysis.* Berlin: Springer-Verlag (forthcoming).

Rubinstein, R. Y. and A. Shapiro. 1990. Optimization of static simulation models by the score function method. *Mathematics and Computers in Simulation* 32:373–392.

Rubinstein, R. Y. and A. Shapiro. 1993. *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method.* Chichester and New York: John Wiley & Sons.

Schramm, H. and J. Zowe. 1990. A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results. Technical Report 209, Mathematisches Institut, Universität Bayreuth, Bayreuth, Germany.

Shapiro, A. 1993. Asymptotic behavior of optimal solutions in stochastic programming. *Mathematics of Operations Research* 18:829–845.

Shapiro, A. and Y. Wardi. 1994. Convergence analysis of stochastic algorithms. Preprint, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia.

Schittkowski, K. 1985/86. NLPQL: A FORTRAN subroutine solving constrained nonlinear programming problems. *Annals of Operations Research* 5:485–500.

Suri, R. 1989. Perturbation analysis: The state of the art and research issues explained via the GI/G/1 queue. *Proceedings of the IEEE* 77:114–137.

Suri, R. and B.–R. Fu. 1991. On using continuous flow lines for performance estimation of discrete production lines. In *Proceedings of the 1991 Winter Simulation Conference,* ed. B.L. Nelson, W.D. Kelton, and G.M. Clark, 968–977. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Suri, R. and B.–R. Fu. 1994. On using continuous flow lines to model discrete production lines. *Journal of Discrete Event Dynamic Systems* (forthcoming).

Zowe, J. 1989. The BT–algorithm for minimizing a nonsmooth functional subject to linear constraints. In *Nonsmooth Optimization and Related Topics,* ed. F. H. Clarke *et al.*. New York: Plenum Press.