ELSEVIER European Journal of Operational Research 88 (1996) 413-427

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

Invited Review

Optimization and sensitivity analysis of computer simulation models by the score function method

Jack P.C. Kleijnen [a,*], Reuven Y. Rubinstein [b]

[a] Department of Information Systems and Center for Economic Research (CentER), Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands

[b] Faculty of Industrial Engineering and Management, Technion—Israel Institute of Technology, Haifa 32000, Israel

Received March 1995

Abstract

This paper surveys some recent results on the score function (SF) method. This method is suitable for performance evaluation, sensitivity analysis, and optimization of complex discrete-event systems such as non-Markovian queueing systems.

Keywords: Experimental design; Score function; Sensitivity analysis; Optimization; Simulation

1. Introduction

Many complex real world systems can be modeled as discrete-event systems (DES). Examples are computer-communication networks, flexible manufacturing systems, PERT-project networks, and flow networks. These systems are typically driven by the occurrence of discrete events, so their states change with time. In view of the complex interactions among such discrete events, DES are typically studied via stochastic simulation.

In designing, analyzing, and operating such complex DES we are interested, not only in *performance evaluation* but also in *sensitivity analysis* and *optimization*. Consider the following examples.

(1) *Traffic light systems.* (i) The performance measure may be a vehicle's average delay as it proceeds from a given point of origin to a given destination, or the average number of vehicles waiting for a green light at a given intersection in the system. (ii) The sensitivity and decision parameters may be the average rate at which the vehicles arrive at the intersections in the system, and the rate at which the light changes from green to red.

(2) *Manufacturing systems.* (i) The performance measure may be the average waiting time of an item to be processed at several workstations (robots) according to a given schedule and route. (ii) The sensitivity and decision parameters may be the average rate at which the workstations (robots) process the item. In such systems we might be interested in minimizing the average make-span (consisting of the processing time and delay time), accounting for some constraints (for example, cost).

Until about a decade ago, sensitivity analysis and optimization of DES was associated with the classic statistical design of experiments. Compared with naive, common sense approaches, statistical designs require less computer time and give more general and accurate results (Kleijnen, 1987, 1994). However,

[*] Corresponding author.

SSDI 0377-2217(95)00107-7

these designs assume that the simulation model is run repeatedly, namely for different combinations of 'factor levels'; these levels correspond with the values of the (say) n parameters of the simulation model of the DES. In Section 4, we shall return to these experimental designs.

In the last decade, two new methods for sensitivity analysis and optimization of DES have been developed. They are called *infinitesimal perturbation analysis* (IPA) (e.g., Glasserman, 1991; Fu, 1994) and *score function* (SF) (also called *likelihood ratio*) (e.g., Glynn , 1990; L'Ecuyer, 1990; Reiman and Weiss, 1989).

This paper is about the SF method. We shall show that this method allows us to evaluate, *simultaneously from a single sample path* (simulation run) not only the performance and all its sensitivities (gradient, Hessian, etc.), but also to solve an entire optimization problem. Today, the SF method allows us to perform sensitivity and optimization of hundreds of decision parameters. The SF algorithms and procedures are implemented in a simulation package called QNSO (Queueing Network Stabilizer and Optimizer); they can be readily adapted to any existing discrete-event simulation language, such as SLAM , SIMAN, and GPSS. The extra computational time required by SF is about 10-50% of the time of the underlying simulation run.

To the best of our knowledge the SF method in a simulation context was introduced independently in the late 1960s by Aleksandrov, Sysoyev and Shemeneva (1968), Mikhailov (1967), Miller (1967), and Rubinstein (1969). Related references in the early 1980s are Ermakov and Mikhailov (1982), Kreimer (1984), and Rubinstein and Kreimer (1983). In 1986 Glynn and Reiman and Weiss independently rediscovered the score function method, and called it the *likelihood ratio* method, (Glynn, 1990; Reiman and Weiss, 1989; and references therein).

Sections 2 and 3 deal with sensitivity analysis of discrete-event static systems (DESS) and discrete-event dynamic systems (DEDS), respectively. The main difference between these two types is that DESS do not evolve with time, whereas DEDS do. Examples of DESS are stochastic PERT networks and $GI/G/\infty$ queues; an example of DEDS is a queueing network. Section 4 shows how to combine the SF method with classic experimental design. Section 5 discusses optimization of DEDS from a single simulation run. Finally, Section 6 gives conclusions.

2. Sensitivity analysis of discrete-event static systems

Assume that the expected performance $\ell(\boldsymbol{v})$ can be represented in the form

$$\ell(\boldsymbol{v}) = \mathbb{E}_{\boldsymbol{v}} L(\boldsymbol{Y}) = \int L(\boldsymbol{y})\, \mathrm{d}F(\boldsymbol{y},\boldsymbol{v}), \qquad (2.1)$$

where $L(\boldsymbol{Y})$ is the sample performance of the simulated DESS, driven by an m-dimensional input vector $\boldsymbol{Y}$ with a cumulative distribution function (cdf) $F(\boldsymbol{y},\boldsymbol{v})$, $\boldsymbol{v}$ is a vector of parameters lying in a parameter set $V \subset \mathbb{R}^n$, and the subscript $\boldsymbol{v}$ in $\mathbb{E}_{\boldsymbol{v}} L$ means that the expectation is taken with respect to $F(\boldsymbol{y},\boldsymbol{v})$. A technical assumption is that $F(\boldsymbol{y},\boldsymbol{v})$ belongs to a family of distributions that are absolutely continuous with respect to the Lebesgue measure. The treatment of the case where $F(\boldsymbol{y},\boldsymbol{v})$ belongs to a family of discrete or mixture distributions is similar.

Suppose first that the parameter $\boldsymbol{v}$ is a scalar v and the parameter set V is an open interval of the real line. Suppose also that for all $\boldsymbol{y}$ the pdf (probability distribution function) $f(\boldsymbol{y},v)$ is continuously differentiable in v and that there exists an integrable (with respect to the Lebesgue measure) function $h(\boldsymbol{y})$ such that

$$|L(\boldsymbol{y})\partial f(\boldsymbol{y},v)/\partial v| \leqslant h(\boldsymbol{y}) \qquad (2.2)$$

for all $v \in V$. Then by the Lebesgue dominated convergence theorem the operators of differentiation and expectation (integration) are interchangeable, so (2.1) yields

$$\begin{aligned}\frac{\mathrm{d}\ell(v)}{\mathrm{d}v} &= \frac{\mathrm{d}}{\mathrm{d}v}\int L(\boldsymbol{y})f(\boldsymbol{y},v)\,\mathrm{d}\boldsymbol{y} \\ &= \int L(\boldsymbol{y})\frac{\mathrm{d}f(\boldsymbol{y},v)}{\mathrm{d}v}\frac{f(\boldsymbol{y},v)}{f(\boldsymbol{y},v)}\,\mathrm{d}\boldsymbol{y} \\ &= \int L(\boldsymbol{y})\frac{\mathrm{d}\log f(\boldsymbol{y},v)}{\mathrm{d}v}f(\boldsymbol{y},v)\,\mathrm{d}\boldsymbol{y} \\ &= \mathbb{E}_v\left\{L(\boldsymbol{Y})\frac{\mathrm{d}\log f(\boldsymbol{Y},v)}{\mathrm{d}v}\right\}.\end{aligned}$$

The extension to the multidimensional case where $\boldsymbol{v} \in \mathbb{R}^n$ is straightforward. Indeed, by similar arguments we can write the gradient of $\ell(\boldsymbol{v})$ in the form

$$\nabla \ell(v) = \mathbb{E}_v\{L(Y)\nabla \log f(Y,v)\} = \mathbb{E}_v\{L(Y)S^{(1)}(Y,v)\}, \tag{2.3}$$

where

$$S^{(1)}(y,v) = \frac{\nabla f(y,v)}{f(y,v)} = \nabla \log f(y,v) \tag{2.4}$$

is called the *efficient score function*. Similarly, the higher order derivatives can be written as

$$\nabla^{(k)}\ell(v) = \mathbb{E}_v\{L(Y)S^{(k)}(Y,v)\}, \tag{2.5}$$

where

$$S^{(k)}(y,v) = \frac{\nabla^{(k)} f(y,v)}{f(y,v)}. \tag{2.6}$$

Let $Y_1,\dots,Y_N$ be a sample of size N from $f(y,v)$. Then $\nabla^k\ell(v)$ can be estimated *simultaneously* from a *single simulation run* by

$$\bar{\nabla}^{(k)}\ell_N(v) = N^{-1}\sum_{i=1}^{N} L(Y_i)S^{(k)}(Y_i,v). \tag{2.7}$$

Formula (2.7) is also valid for $k = 0$ if we define $\nabla^0\ell(v) \equiv \ell(v)$ and $S^{(0)}(y,v) \equiv 1$. Since the estimator, $\bar{\nabla}\ell_N(v)$ is based on the *efficient score* defined in (2.4), the proposed method is called the *score function* (SF) method.

Example 2.1 (Exponential family). Let Y be a random vector distributed according to an exponential family, i.e.,

$$f(y,v) = a(v)\exp\left\{\sum_{k=1}^{s} b_k(v)t_k(y)\right\}h(y), \tag{2.8}$$

where $a(v) > 0$ and $b_k(v)$ are real-valued functions of the parameter vector v, and $t_k(y)$ and $h(y)$ are real-valued functions of y. Then

$$S^{(1)}(y,v) = a(v)^{-1}\nabla a(v) + \sum_{k=1}^{s} t_k(y)\nabla b_k(v). \tag{2.9}$$

Notice that in Example 2.1 the function $\ell(v)$ is differentiable and its derivatives can be taken inside the expected value, so that the corresponding expectations do exist.

It is important to note that the estimator $\bar{\nabla}^k\ell_N(v)$ given in (2.7) allows us to evaluate the performance $\ell(v)$ and its sensitivity $\nabla^k\ell(v)$ only at a *fixed* point v. We now present an extended version of the above estimators that allows us to evaluate $\ell(v)$ and $\nabla^k\ell(v)$, essentially *everywhere* in v, provided some regularity conditions are met (Rubinstein and Shapiro, 1993).

Let G be a probability measure (distribution) on $\mathbb{R}^m$ having a density function $g(y)$, so that $dG(y) = g(y)\,dy$. Suppose that for every permissible value of the parameter vector, the support of $f(y,v)$ lies within the support of $g(y)$, that is

$$\text{supp}\{f(y,v)\} \subset \text{supp}\{g(y)\}, \quad v \in V \tag{2.10}$$

(recall that $\text{supp}\{g(y)\}$ is the set of those values of y for which $g(y)$ is strictly greater than zero). Let further $f(y,v)$ be differentiable in v. Define $\nabla^k W(y,v) = \nabla^k f(y,v)/g(y)$. Then we can write $\nabla^k\ell(v)$ in (2.5) as follows:

$$\nabla^k\ell(v) = \int L(y)\nabla^k f(y,v)\,dy = \int L(y)\nabla^k W(y,v)\,dG(y) = \mathbb{E}_g\{L(Z)\nabla^k W(Z,v)\}, \tag{2.11}$$

where $Z \sim g(z)$ and we define $\nabla^0\ell(v) \equiv \ell(v)$ and $\nabla^0 W(y,v) \equiv W(y,v)$ (by definition, zero divided by zero is zero). Notice that the function $W(y,v)$ is well defined for all $v \in V$ because of the assumption (2.10). In the statistical literature, $W(Z,v)$ is called the *likelihood ratio* or the Radon-Nikodym derivative; in simulation, $W(Z,v)$ is the basis of *importance sampling*.

It is important to note that the original expectation of $L(Y)$ in (2.1) is taken with respect to the underlying pdf $f(y,v)$, whereas that given in the last expression of (2.11) is taken with respect to the pdf $g(y)$. It follows that changing the probability density from $f(y,v)$ to $g(y)$, we can express the performance measure $\ell(v)$ for all $v \in V$ as an expectation with respect to $g(y)$ and then estimate it accordingly. We shall call the pdf $g(y)$, satisfying condition (2.10), the *dominating* pdf.

Note that the sensitivities $\nabla^k\ell(v) = \mathbb{E}_v\{LS^{(k)}\}$ in (2.5) represent a particular case of (2.11), namely with $g(y) = f(y,v)$ so $W(y,v) = 1$.

An unbiased estimator of $\nabla^k \ell(v)$ analogous to (2.7) is:

$$\bar{\nabla}^k \ell_N(v) = N^{-1} \sum_{i=1}^{N} L(Z_i) \nabla^k W(Z_i, v) , \qquad (2.12)$$

$k = 0, 1, \ldots,$ where $Z_1, \ldots, Z_N$ is a sample from $g(z)$.

We shall call $\nabla^k W(Z, v)$, $k = 1, 2, \ldots,$ the *generalized scores*; $W(Z, v) = \nabla^0 W(Z, v)$ we called the likelihood ratio.

For a given dominating pdf $g(z)$, we can write the following algorithm for estimating $\nabla^k \ell(v)$ from a *single simulation run* for (say) s *different* values of v, namely $v_1, \ldots, v_s$.

Algorithm 2.1.

```
Select the simulation runlength $N$.
For $i := 1$ to $N$ do
BEGIN
   generate $Z_i$ from the dominating pdf $g(z)$;
   calculate the performance $L(Z_i)$;
   For $j := 1$ to $s$ {$s$ denotes # values of $v$} do
   BEGIN
      calculate the generalized scores $\nabla^k W(Z_i, v_j)$;
      update $L(Z_i) * \nabla^k W(Z_i, v_j)$
   END { of $j$}
END { of $i$}
For $j := 1$ to $s$ do
   {compute final values: divide by $N$}
   compute $L(Z_i) * \nabla^k W(Z_i, v_j)/N$
```

The accuracy (variance) of the estimators $\bar{\nabla}^k \ell_N(v)$, $k = 0, 1, \ldots$, depends on the particular choice of the dominating density $g(z)$ (Rubinstein and Shapiro, 1993). Actually the optimal $g(z)$, say $g^*(z)$, is $g^*(z) = |Lf|/\mathbb{E}(L)$, but we do not know $\mathbb{E}(L)$. We restrict ourselves to $g(z) = f(z, v_0)$, which denotes the same family of distributions as the original one, but with a different parameter v_0, where v_0 is called the *reference parameter*.

3. Sensitivity analysis of discrete-event dynamic systems

Let $Y_1, Y_2, \ldots$ be an input sequence of independently and identically distributed (iid) random input vectors, generated from a pdf $f(y, v)$ with the n-dimensional parameter vector v. Consider an output process $\{L_t : t > 0\}$ driven by the input sequence $\{Y_t\}$, that is, $L_t = L_t(\underline{Y}_t)$, where the vector $\underline{Y}_t = (Y_1, Y_2, \ldots, Y_t)$ represents a history of the input process up to time t and $L_t(\cdot)$ is a sequence of real-valued functions. Assume that $\{L_t\}$ is a discrete-time *regenerative* process. It is well known in the theory of regenerative processes (e.g., Asmussen, 1987) that the expected steady-state performance $\ell(v)$ of a regenerative process can be written as

$$\ell(v) = \mathbb{E}_v X / \mathbb{E}_v \tau, \qquad (3.1)$$

where $X = \sum_1^\tau L_t$ and τ is the length of the regenerative cycle. Similar results hold when $\{L_t\}$ is a continuous-time process where the sum in the definition of X is replaced by the corresponding integral. If not stated otherwise, we assume that L_t is the steady-state waiting process in the $GI/G/1$ queue with FIFO discipline.

Before proceeding with the calculation of $\nabla^k \ell(v)$ we first consider $\ell_1(v) = \mathbb{E}_v X$. It is shown in Feuerverger, McLeish, and Rubinstein (1986) that the gradient of $\mathbb{E}_v X$ can be expressed as

$$\nabla^k \ell_1(v) = \mathbb{E}_v \{ \sum_{t=1}^{\tau} L_t \tilde{S}_t^{(k)} \}, \qquad (3.2)$$

where, analogous to (2.6), we have

$$\tilde{S}_t^{(k)} = \nabla^k f_t(\underline{Y}_t, v) / f_t(\underline{Y}_t, v).$$

In particular, because the Y_t are i.i.d., we have

$$\tilde{S}_t^{(1)} = \sum_{i=1}^{t} \nabla \log f(Y_j, v). \qquad (3.3)$$

Example 3.1 (Gamma distribution). Let Y be gamma distributed, that is,

$$f(y, \lambda, \beta) = \frac{\lambda^\beta y^{\beta - 1} e^{-\lambda y}}{\Gamma(\beta)}, \quad y > 0.$$

Assume that we are interested in the sensitivities with respect to λ only (not β). We then have that

$$\tilde{S}_t^{(1)}(\underline{Y}_t, \lambda) = \frac{\partial}{\partial \lambda} \log f_t(\underline{Y}_t, \lambda, \beta) = t\beta\lambda^{-1} - \sum_{i=1}^{t} Y_i.$$

Clearly, $\nabla^k \ell_1(v)$, $k = 1,2,\ldots$, can be estimated as

$$\bar{\nabla}^k \ell_{1N}(v) = N^{-1}\sum_{i=1}^{N}\sum_{t=1}^{\tau_i} L_{ti}\tilde{S}_{ti}^{(k)}, \qquad (3.4)$$

where τ_i and N are the length of the i-th cycle and the number of generated cycles, respectively (so the corresponding sample is $Y_{11},\ldots,Y_{\tau_1 1},\ldots,Y_{1N},\ldots,Y_{\tau_N N}$).

Differentiating $\ell(v)$ defined in (3.1), taking into account (3.2), and noting that $\mathbb{E}_v\,\tau$ and $\nabla\,\mathbb{E}_v\,\tau$ represent particular cases of $\mathbb{E}_v\,X$ and $\nabla\,\mathbb{E}_v\,X$, respectively, with $L_t = 1$, we obtain

$$\begin{aligned}\nabla\ell(v) &= \frac{\nabla\,\mathbb{E}_v X}{\mathbb{E}_v\,\tau} - \frac{\mathbb{E}_v X}{\mathbb{E}_v\,\tau}\cdot\frac{\nabla\,\mathbb{E}_v\,\tau}{\mathbb{E}_v\,\tau}\\ &= \frac{\mathbb{E}_v\{\sum_1^\tau L_t\tilde{S}_t^{(1)}\}}{\mathbb{E}_v\,\tau}\\ &\quad - \frac{\mathbb{E}_v\{\sum_1^\tau L_t\}}{\mathbb{E}_v\,\tau}\cdot\frac{\mathbb{E}_v\{\sum_1^\tau \tilde{S}_t^{(1)}\}}{\mathbb{E}_v\,\tau}. \qquad (3.5)\end{aligned}$$

Defining

$$\bar{s}^{(1)} = \frac{\mathbb{E}_v\{\sum_1^\tau \tilde{S}_t^{(1)}\}}{\mathbb{E}_v\,\tau},$$

$$Q_t^{(1)} = (L_t - \ell(v))\left(\tilde{S}_t^{(1)} - \bar{s}^{(1)}\right),$$

(3.5) can be rewritten (according to Rubinstein and Shapiro, 1993, p. 89) as

$$\begin{aligned}\nabla\ell(v) &= \mathbb{E}_v\{Q^{(1)}\} = \mathrm{Cov}_v\{L,\tilde{S}^{(1)}\}\\ &= \frac{\mathbb{E}_v\{\sum_1^\tau Q_t^{(1)}\}}{\mathbb{E}_v\,\tau}. \qquad (3.6)\end{aligned}$$

Similarly, we obtain

$$\begin{aligned}\nabla^k\ell(v) &= \mathbb{E}_v\{Q^{(k)}\} = \mathrm{Cov}_v\{L,\tilde{S}^{(k)}\}\\ &= \frac{\mathbb{E}_v\{\sum_1^\tau Q_t^{(k)}\}}{\mathbb{E}_v\,\tau}, \qquad (3.7)\end{aligned}$$

where

$$Q_t^{(k)} = (L_t - \ell(v))\left(\tilde{S}_t^{(k)} - \bar{s}^{(k)}\right),$$

$$\bar{s}^{(k)} = \frac{\mathbb{E}_v\{\sum_1^\tau \tilde{S}_t^{(k)}\}}{\mathbb{E}_v\,\tau}.$$

Thus, $\nabla^k\ell(v)$ can be expressed as the *covariance* between the steady-state process $\{L_t\}$ and $\{\tilde{S}_t^{(k)}\}$. The variable $\{\tilde{S}_t^{(k)}\}$ is based on the score function $\nabla\log f(Y,v)$; see (3.2).

Example 3.2. Let L_t be the steady-state sojourn time of a customer in the $GI/G/1$ system, where Y_{1j} is the service time of the j-th customer, $Y_{2j} = 0$ for $j = 1$, $Y_{2j} = A_j - A_{j-1}$ for $j \geqslant 2$, A_j is the arrival time of the j-th customer and $\tau = \min\{t : \sum_1^t (Y_{1j} - Y_{2j+1}) \leqslant 0\}$ is the number of customers served during the busy period. In this case we obtain

$$\begin{aligned}\ell(v) &= \frac{\mathbb{E}_v\{\sum_1^\tau L_t\}}{\mathbb{E}_v\,\tau}\\ &= \frac{\mathbb{E}_v\left\{\sum_{t=1}^{\tau}\sum_{j=1}^{t} Y_{1j} - \sum_{t=2}^{\tau}\sum_{j=2}^{t} Y_{2j}\right\}}{\mathbb{E}_v\,\tau}.\end{aligned}$$

Denoting $U_j = Y_{1j} - Y_{2j}$ we can rewrite $\ell(v)$ as

$$\ell(v) = \frac{\mathbb{E}_v\{\sum_{t=1}^{\tau}\sum_{j=1}^{t} U_j\}}{\mathbb{E}_v\,\tau}.$$

In this case

$$\nabla\ell(v) = \frac{\mathbb{E}_v\{\sum_1^\tau Q_t^{(1)}\}}{\mathbb{E}_v\,\tau},$$

where

$$Q_t^{(1)} = \left(\sum_{j=1}^{t} U_j - \ell\right)\left(\tilde{S}_t^{(1)} - \bar{s}^{(1)}\right)$$

and

$$\tilde{S}_t^{(1)} = \sum_{j=1}^{t}\nabla\log f(Y_j,v),$$

$$f(y,v) = f_1(y_1,v_1)f_2(y_2,v_2),$$

$$Y = (Y_1,Y_2), \quad Y_1 \sim f_1(y_1,v_1), \quad Y_2 \sim f_2(y_2,v_2).$$

Take a sample of N regenerative cycles from the pdf $f(y,v)$. Then, taking into account (3.7), we can estimate all the quantities $\nabla^k\ell(v)$, $k = 0,1,\ldots$, from a *single* simulation run by

$$\bar{\nabla}^k\ell_N(v) = \frac{\sum_{i=1}^{N}\sum_{t=1}^{\tau_i}\tilde{Q}_{ti}^{(k)}}{\sum_{i=1}^{N}\sum_{t=1}^{\tau_i} 1}, \quad k = 0,1,\ldots, \qquad (3.8)$$

where

$$\tilde{Q}_{ti}^{(0)} = L_{ti}, \qquad \tilde{Q}_{ti}^{(k)} = \left(L_{ti} - \bar{\ell}_N\right)\left(\tilde{S}_{ti}^{(k)} - \bar{s}_N^{(k)}\right), \tag{3.9}$$

and $\bar{\ell}_N$ and $\bar{s}_N^{(k)}$ are the sample estimators of $\ell = \mathbb{E}_{\boldsymbol{v}}\{\sum_k^\tau L_t\}/\mathbb{E}_{\boldsymbol{v}}\,\tau$ and $\bar{s}^{(k)} = \mathbb{E}_{\boldsymbol{v}}\{\sum_k^\tau \tilde{S}_t^{(k)}\}/\mathbb{E}_{\boldsymbol{v}}\,\tau$, respectively.

We now present the extended version of the above estimators which allows us to estimate $\nabla^k\ell(\boldsymbol{v})$, $k = 0, 1, \ldots$, at *any* point $\boldsymbol{v}$, provided some regularity conditions hold. Assume, as in Section 2, that $g(z)$ dominates the densities $f(z, \boldsymbol{v})$ in the sense of (2.10). It can then be shown (Rubinstein and Shapiro (1993)) that (analogous to (2.11))

$$\nabla^k\ell_1(\boldsymbol{v}) = \mathbb{E}_g\Big\{\sum_{t=1}^{\tau} L_t(\underline{Z}_t)\nabla^k\tilde{W}_t(\underline{Z}_t, \boldsymbol{v})\Big\}, \qquad k = 0, 1, \ldots, \tag{3.10}$$

where $\tilde{W}_t(\underline{Z}_t, \boldsymbol{v}) = \prod_{j=1}^t W_j(Z_j, \boldsymbol{v})$ with $W_j(Z_j, \boldsymbol{v}) = f(Z_j, \boldsymbol{v})/g(Z_j)$.

Taking into account (3.10) we estimate $\ell(\boldsymbol{v})$ *simultaneously for different values of* $\boldsymbol{v}$ by

$$\bar{\ell}_N((\upsilon) = \frac{\sum_1^N \sum_1^{\tau_i} L_{ti}\tilde{W}_{ti}}{\sum_1^N \sum_1^{\tau_i} \tilde{W}_{ti}}. \tag{3.11}$$

Note that (3.8) with $k = 0$ is a special case of (3.11), namely $f(z, \boldsymbol{v}) = g(z)$ so $\tilde{W}_{ti} = 1$. Similarly, we estimate$\bar{\nabla}\ell(\upsilon)$ by

$$\bar{\nabla}\ell_N(\boldsymbol{v}) = \frac{\sum_1^N \sum_1^{\tau_i} L_{ti}\nabla\tilde{W}_{ti}}{\sum_1^N \sum_1^{\tau_i} \tilde{W}_{ti}} - \frac{\sum_1^N \sum_1^{\tau_i} L_{ti}\tilde{W}_{ti}}{\sum_1^N \sum_1^{\tau_i} \tilde{W}_{ti}} \cdot \frac{\sum_1^N \sum_1^{\tau_i} \nabla\tilde{W}_{ti}}{\sum_1^N \sum_1^{\tau_i} \tilde{W}_{ti}}. \tag{3.12}$$

Note that $\nabla\tilde{W}_t = \tilde{W}_t\ \tilde{S}_t^{(1)}$. Similar estimators can be derived for $\nabla^k\ell(\boldsymbol{v})$, $k = 2, 3 \ldots$, by differentiating $\bar{\ell}_N(\boldsymbol{v})$ k times.

The algorithm for estimating the gradient $\nabla\ell(\boldsymbol{v})$, based on the sensitivity estimator (3.12), can be written as follows.

Algorithm 3.1.

(1) Generate a random sample $Z_1, \ldots, Z_T$, $T = \sum_1^N \tau_i$, from $g(z)$.

(2) Generate the output processes L_t, $\tilde{S}_t^{(1)}$, $\tilde{W}_t$, and $\nabla\tilde{W}_t = \tilde{W}_t\tilde{S}_t^{(1)}$.

(3) Calculate $\bar{\ell}_N((\upsilon)$ and $\nabla\bar{\ell}_N((\upsilon)$ according to (3.11) and (3.12), respectively.

Assume further that we restrict $g(\boldsymbol{y})$ to the same parametric family that $f(\boldsymbol{y}, \boldsymbol{v})$ belongs to; that is, $g(\boldsymbol{y}) = f(\boldsymbol{y}, \boldsymbol{v}_0)$.

Remark 3.1. Rubinstein and Shapiro (1993) show how to obtain reasonably "good" estimators of $\nabla^k\ell(\boldsymbol{v})$, $k = 0, 1$ simultaneously for different values of $\boldsymbol{v}$, say $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_s$. Let ρ be the traffic intensity in the $GI/G/1$ queue, or in more complex queueing models. Then one has to choose the reference parameter $\boldsymbol{v}_0$ such that the traffic intensity $\rho(\boldsymbol{v}_0)$ is either

$$\rho(\boldsymbol{v}_0) = \max_{j=1,\ldots,s} \rho(\boldsymbol{v}_j) \tag{3.13}$$

or moderately larger than $\rho(\boldsymbol{v}_0)$; that is, the reference parameter $\boldsymbol{v}_0$ must correspond with the *highest traffic intensity* among all traffic intensities associated with the selected values $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_s$.

Fig. 3.1 depicts the estimator of the response curve $\ell(\boldsymbol{v})$, namely the curve $\bar{\ell}_N(\rho) = \bar{\ell}_N(\rho \mid \rho_0)$ (denoted by ℓ_N in that figure) along with the two curves

$$J_1 = \{\bar{\ell}_N(\rho \mid \rho_0) - w_r\},$$

and

$$J_2 = \{\bar{\ell}_N(\rho \mid \rho_0) + w_r\},$$

where

$$w_r = \frac{1.96\hat{\sigma}(\rho \mid \rho_0)}{\bar{\ell}_N(\rho \mid \rho_0)}$$

(denoted by 95%CI) as functions of ρ for the $M/G/1$ queue with $\rho_0 = 0.8$; here $\hat{\sigma}(\rho \mid \rho_0)$ is the estimate of the standard deviation of $\bar{\ell}_N((\upsilon)$ in (3.11), so ω_r represents the half width of the 95% relative confidence interval. Note that $\bar{\ell}_N(\rho \mid \rho_0)$ and w_r in J_1, J_2 are given in different scales.

Fig. 3.2 depicts similar data for the derivative of the expected waiting time in the $M/G/1$ queue with respect to the service rate λ.

In those two figures we assumed that $\ell(\boldsymbol{v})$ is the steady-state expected waiting time of a customer in

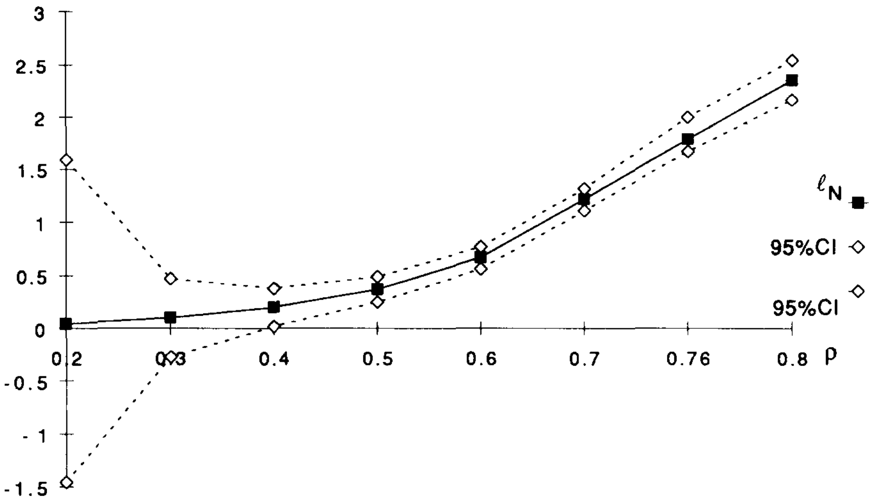

Fig. 3.1. Performance of the "what if" estimator $\bar{\ell}_N(\rho \mid \rho_0)$ as function of ρ for the $M/G/1$ queue with $\rho_0 = 0.8$.

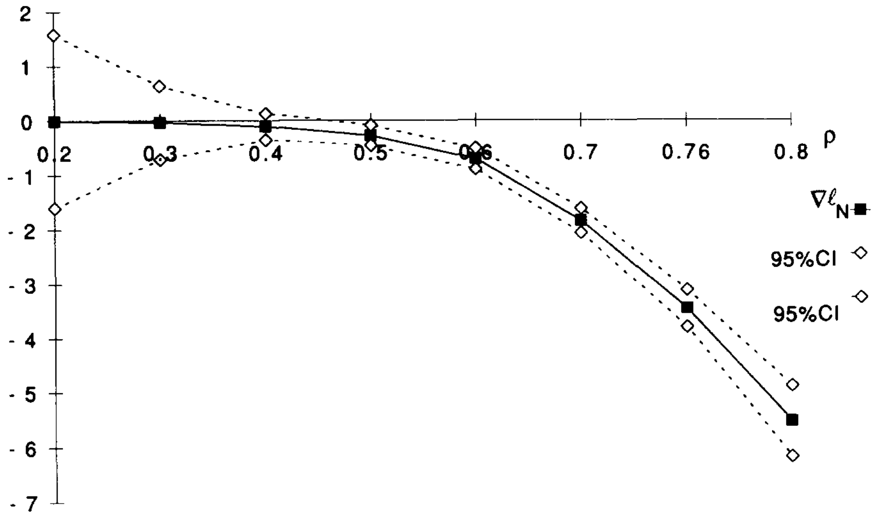

Fig. 3.2. Performance of the "what if" estimator $\nabla\bar{\ell}_N(\rho \mid \rho_0)$ as function of ρ for the $M/G/1$ queue with $\rho_0 = 0.8$.

the $M/G/1$ queue. We took $\lambda_0 = v_0$ as the reference parameter; that is, we assumed $\rho_0 = \lambda_0\beta$, chose the arrival rate equal to 1, the scale $\lambda_0 = v_0 = 0.4$, the shape $\beta = 2$, so $\rho_0 = E(Y) = \lambda_0\beta = 0.8$; see Example 3.1. From a *single simulation run* we estimated the performance $\ell(v)$ and the derivative *simultaneously* for $\rho = 0.2, 0.3, \ldots, 0.8$, while simulating 10,000 customers.

It is readily seen that the "what if" estimators $\bar{\ell}_N(\rho \mid \rho_0)$ and $\nabla\bar{\ell}_N(\rho \mid \rho_0)$ perform reasonably well in the range $\rho \in (0.4, 0.8)$. For larger perturbations the SF process $\nabla^k\tilde{W}$ $(k = 0, 1)$ blows up the variance of the estimators $\bar{\ell}_N(\rho \mid \rho_0)$ and $\nabla\bar{\ell}_N(\rho \mid \rho_0)$. Note that the *true* $\ell(v)$ and $d\,\ell(v)$, which are known for the $M/G/1$ case, lie within the confidence bands, but they are not shown in the figure.

The basic formulas (3.11)–(3.12) developed for the $GI/G/1$ queue with the FIFO discipline can be extended to more general open and closed queueing systems in the sense that (3.11)–(3.12) still hold, provided the *indexing* in the likelihood ratio process $\tilde{W}_t$ and the associated quantities, such as $\sum_1^\tau L_t\tilde{W}_t$ and $\sum_1^\tau L_t\nabla\tilde{W}_t$, are defined in a more sophisticated way. When the process L_t is nonregenerative, but stationary and ergodic, we can use the so-called *decomposable* and *truncated* estimators. For details we refer to Rubinstein and Shapiro (1993).

4. Combining the score function with classic experimental design

This section deals with extensions of the SF method for the following model:

$$\ell(v) = \mathbb{E}_{v_1}\{L(\underline{Y}_t, v_2)\}, \tag{4.1}$$

where $L(\underline{Y}_t)$ is the sample performance, $Y_1, \ldots, Y_t$ are iid random vectors with common pdf $f(y, v_1)$, the combined vector of parameters is given here by $v = (v_1, v_2)$, and the subscript v_1 in $\mathbb{E}_{v_1}[L]$ indicates that the expectation is taken with respect to the pdf $f(y, v_1)$. So we assume that the pdf f depends on the parameter vector v_1 but not on v_2, and that the sample performance L depends on v_2, but not on v_1. We shall call v_1 and v_2 the *distributional* and the *structural* parameter vector, respectively. Note also that the model in Section 3, $\ell(v) = \mathbb{E}_v[L(\underline{Y}_t)]$, can be considered as a particular case of the model (4.1) with L not dependent on v_2 and with $v = v_1$.

As before, we suppose that $\ell(v)$ is not available analytically, so we use simulation to estimate $\ell(v)$ as well as the associated sensitivities $\nabla^k \ell(v)$, $k = 1, 2, \ldots$, for multiple values of $v = (v_1, v_2)$. Consider the following examples.

(1) *GI/G/1 queue.* Suppose that it is desired to estimate the cdf

$$P_v\{L \leqslant x\} \tag{4.2}$$

with L the sample performance in the steady state, and the associated derivative, $\partial P_v(L \leqslant x)/\partial x$, for multiple values of $v_1 = v$ and $v_2 = x$. In this case, we can represent $P_v(L \leqslant x)$ as

$$P_v(L \leqslant x) = \mathbb{E}_{v_1}\{I_{(-\infty,0]}(L - x)\}, \tag{4.3}$$

where $I_{(-\infty,0]}(\cdot)$ is the indicator function of the interval $(-\infty, 0]$.

(2) *GI/D/1 and D/G/1 queues*, where D stands for deterministic. For the $GI/D/1$ queue, $Y \sim f_1(y_1, v_1)$ represents the random interarrival time with interarrival rate v_1, and v_2 is the length of the constant service time. Similar definitions hold for the $D/G/1$ queue.

(3) *GI/G/1/m queue*, where m denotes the buffer size. Suppose it is desired to estimate the steady-state expected waiting time $\ell(v) = \mathbb{E}_{v_1}[L(Y, v_2)]$ of a stable $GI/G/1/m$ queue for multiple values of $v_2 = m$. (This problem is treated rigorously in Kriman, 1994.)

One approach (not the focus of this paper) uses "push-out" and "push-in" , respectively (Rubinstein, 1992). These terms derive from the fact that in the first case we "push out" the parameter vector, v_2, from the original sample performance $L(Y, v_2)$ into an auxiliary pdf via a suitable transformation, and then apply the standard SF method; in the second case, we operate the other way around, namely, we first "push in" (via a suitable transformation) the parameter vector v_1 into the sample performance $L(Y, v_2)$ and then differentiate the resulting (auxiliary) sample performance with respect to $v = (v_1, v_2)$. Conditions under which such transformations are useful, in the sense that they either generate *smooth sample performances* or lead to *variance reduction*, are discussed in Marti (1990) and Uryas'ev (1994). It is also shown that the *infinitesimal perturbation analysis* (IPA) method introduced by Ho and his co-workers (Ho and Cao, 1991), corresponds with the "push in" technique; the latter can be viewed as a dual of the "push out" technique.

A second approach, discussed in the remainder of this section, is based on the idea that the effects of changing one or more components of the distributional vector v_1 can be estimated with relatively little effort (in the way we discussed in the preceding sections), whereas the effects of changing one or more components of the structural vector v_2 are estimated with more effort, using classic *experimental design* (ED). We shall also show that the ideas of ED might be utilized in SF, in order to further reduce computer time. Details on the application of ED in simulation can be found in Kleijnen (1987, 1994).

Suppose v_2 has k_2 components; that is, the vector has dimensionality k_2. In ED terminology we say that there are k_2 *factors*. The number of *levels* or 'values' per factor, denoted by s_k, is usually limited to a small number, say $2 \leqslant s_k \leqslant 5$ $(k = 1, \ldots, k_2)$. The values for s_k are selected as follows.

If we assume that the effects of the k_2 factors are additive, then we can estimate the k_2 main effects from n simulation runs, where n is the smallest multiple of four that is larger than k_2 (for example, if $8 \leqslant k_2 \leqslant 11$, then $n = 12$). So, only a fraction of all possible 2^{k_2}

combinations or *scenarios* is simulated. Each factor is simulated for only two different values ($s_k = 2$).

If, in addition, we assume that besides main effects, there may also be interactions between factors, then a larger fraction is simulated (still simulating only two values per factor). If morever we assume quadratic effects, then more than two values per factor must be simulated. So-called central composite designs require five values per factor; they do not simulate all 5^{k_2} combinations, but only a fraction (combining the designs for main effects and interactions with designs that change only one factor at a time).

Let us now turn to the distributional parameter vector $\boldsymbol{v}_1$. Suppose $\boldsymbol{v}_1$ has k_1 components. SF gives $\bar{\ell}_N(\boldsymbol{v})$ as an explicit function of $\boldsymbol{v}_1$. In order to get a better understanding of this function, we evaluate this function for a set of values of $\boldsymbol{v}_1$. So far we supposed that component k of $\boldsymbol{v}_1$ is studied for s_k ($k = 1, \ldots, k_1$) values; see Algorithm 2.1 and Fig. 3.1. Now, however, we point out that if k_1 is high, then we may restrict the computer time required and the amount of output data; that is, we drastically restrict the number of values per component, say, $2 \leqslant s_k \leqslant 5$. To the s_k estimates of $E(L)$ we can then fit a curve, such as a polynomial in $\boldsymbol{v}_1$ of degree 1 or 2. We emphasize that these $\prod_k s_k$ responses are positively correlated, since they are based on the same random number stream (namely the one used for the reference parameter $\boldsymbol{v}_{01}$; also see (4.5) below). Note that a 'distributional scenario' is a combination of values for the k_1 components of $\boldsymbol{v}_1$.

We emphasize that ED without SF would experiment with $k_1 + k_2$ factors, whereas ED with SF considers only k_2 factors. In SF the estimation of the gradient $\nabla\ell(\boldsymbol{v}_1)$ is analogous to the estimation of $\ell(\boldsymbol{v}_1)$, as we saw in the preceding sections. In ED the estimation of the gradient $\nabla\ell(\boldsymbol{v}_2)$ follows from differentiating the estimated response curve or metamodel; for example, in a first order polynomial the marginal effects equal the main effects, whereas in a regression metamodel with interactions, by definition, the marginal effects also depend on these interactions (Kleijnen, 1987, 1994).

In summary, *Score Function-Experimental Design* (SFED) considers the set of factors

$$\{(\boldsymbol{v}_1, v_2) = \{v_{11}, \ldots, v_{1k_1},\ v_{21}, \ldots, v_{2k_2}\}. \tag{4.4}$$

SFED selects an ED for the k_2 factors in $\boldsymbol{v}_2$ and an ED for the k_1 factors in $\boldsymbol{v}_1$. Unless otherwise stated, we assume further that $g(\boldsymbol{y}) = f(\boldsymbol{y}, \boldsymbol{v}_{01})$, where $\boldsymbol{v}_{01}$ is the *reference parameter*, and that $\ell(\boldsymbol{v})$ is the mean sojourn time in a $GI/G/c/m$ queue. Let $\boldsymbol{v}_2$ be fixed, while $\boldsymbol{v}_1$ takes values in the set $\{\boldsymbol{v}_{11}, \ldots, \boldsymbol{v}_{1r_1}\}$ where $r_1 = \prod_{k=1}^{k_1} s_k$. Consider the following mathematical program problem:

$$\min_{\boldsymbol{v}_{01}} \max_{\boldsymbol{v}_1 = \{\boldsymbol{v}_{11}, \ldots, \boldsymbol{v}_{1r_1}\}} \mathrm{Var}_{\boldsymbol{v}_{01}}\{\bar{\ell}_N((v)\}. \tag{4.5}$$

Arguing as in Rubinstein and Shapiro (1993) (see also Section 5 below), it seems *natural* to choose the reference parameter, $\boldsymbol{v}_{01}$, in such a way that

$$\rho(\boldsymbol{v}_{01}) = \max_{j=1,\ldots,r_1} \rho(\boldsymbol{v}_{1j}). \tag{4.6}$$

Eq. (4.6) means that $\boldsymbol{v}_{01}$ should correspond to the *highest traffic intensity* among all traffic intensities associated with the permissible values $\boldsymbol{v}_{11}, \ldots, \boldsymbol{v}_{1r_1}$.

Example 4.1. Consider the $GI/G/1/m$ queue. Assume that the buffer size is *fixed* at m and suppose we wish to estimate the expected sojourn time, $\ell(\boldsymbol{v}) = \ell(\boldsymbol{v}_1, \boldsymbol{v}_2)$, simultaneously for all $\boldsymbol{v}_1 = \boldsymbol{v}_{11}, \ldots, \boldsymbol{v}_{1r_1}$ by using the "what-if" estimator $\bar{\ell}_N((v)$. Let v_1 be the service rate. It readily follows from (4.6) that in this case, $\boldsymbol{v}_{01}$ must satisfy $v_{01} = \min(v_{11}, \ldots, v_{1r_1})$, which is the same as

$$\rho_0 = \max(\rho_1, \ldots, \rho_{r_1}), \tag{4.7}$$

where ρ_j corresponds to the service rate v_{ij}, $1 \leqslant j \leqslant r_1$.

Consider now the general case (4.4). In typical applications, the traffic intensity is *monotonic* in each component of $\boldsymbol{v}_2$, in which case, formula (4.6) is applicable again in the sense that once a "good" reference parameter $\rho_0(\boldsymbol{v}_2)$ is chosen, it remains a "good" one for *all* $\boldsymbol{v}_2$ in (4.4). In other words, in order to find a "good" reference parameter $\boldsymbol{v}_{01}(\boldsymbol{v}_2)$ (and the corresponding $\rho_0(\boldsymbol{v}_2)$) suitable for *all* combinations of $\{\boldsymbol{v}_1, \boldsymbol{v}_2\}$, we have to first fix an arbitrary value $\boldsymbol{v}_2$ from the set $\{\boldsymbol{v}_{2j},\ j = 1, \ldots, r_2\}$ and then apply formula (4.6).

Example 4.2. Suppose that we need to estimate the expected waiting time in the $GI/G/1/m$ queue for different combinations of the service rate v_1 and the buffer size $v_2 = m$. We may then choose any buffer size m from the set $\{m_1, \dots, m_{s_2}\}$, find v_{01} according to (4.6), and finally run s_2 simulations corresponding to the chosen values $m_1, \dots, m_{s_2}$, respectively.

The "what-if" estimator of $\ell(\boldsymbol{v})$ given the j-th structural scenario ($j = 1, \dots, r_2$, where $r_2 = \prod_{k=1}^{k_2} s_k$) can be written (analogous to (3.11)) as

$$\bar{\ell}_N(\boldsymbol{v}_1, \boldsymbol{v}_{2j}) = \frac{\sum_1^N \sum_1^{\tau_i(\boldsymbol{v}_{2j})} L_{ti}(\boldsymbol{v}_{2j}) \tilde{W}_{ti}(\boldsymbol{v}_1)}{\sum_1^N \sum_1^{\tau_i(\boldsymbol{v}_{2j})} \tilde{W}_{ti}(\boldsymbol{v}_1)}, \tag{4.8}$$

where we write $\tau_i(\boldsymbol{v}_{2j})$ rather than τ_i, to indicate that its distribution depends on $\boldsymbol{v}_{2j}$.

The SFED algorithm for estimating the response surface, $\ell(\boldsymbol{v})$, can be written as follows:

Algorithm 4.1. The SFED Algorithm

(1) Specify the experimental design for $\boldsymbol{v}_1$ and for $\boldsymbol{v}_2$ respectively; see (4.4). Let r_1 (r_2) denote the number of combinations of values in the ED for v_1 (v_2).

(2) Find the reference parameter, $\boldsymbol{v}_{01}$, via (4.6).

(3) Select the number of renewal cycles N.

```
For $i$ := 1 to $N$ do
BEGIN
  while cycle not ended do
  BEGIN
    generate $\boldsymbol{Z}_i$ from the dominating pdf $f(z, \boldsymbol{v}_{01})$;
    for $j_2$ := 1 to $r_2$ do {perform ED for $\boldsymbol{v}_2$}
    BEGIN
      calculate the performance $L(\boldsymbol{Z}_i, \boldsymbol{v}_{2,j_2})$;
      for $j_1$ := 1 to $r_1$ do {perform SF for $\boldsymbol{v}_1$}
      BEGIN
        calculate the likelihood ratios
          $\tilde{W}(\boldsymbol{Z}_i, \boldsymbol{v}_{1,j_1})$;
        update $L(\boldsymbol{Z}_i, \boldsymbol{v}_{2,j_2}) * \tilde{W}(\boldsymbol{Z}_i, \boldsymbol{v}_{1,j_1})$ and
          $\tilde{W}(\boldsymbol{Z}_i, \boldsymbol{v}_{1,j_1})$;
      END {of $j_1$}
    END {of $j_2$}
  END {of while }
END {of $i$}
for $j_2$ := 1 to $r_2$ do {see (4.8) }
  compute $L(\boldsymbol{Z}_i, \boldsymbol{v}_{2,j_2}) * \tilde{W}(\boldsymbol{Z}_i, \boldsymbol{v}_{1,j_1})$ and
    $\tilde{W}(\boldsymbol{Z}_i, \boldsymbol{v}_{1,j_1})$;
```

Example 4.3.

Consider the estimation of the steady-state mean waiting time, $\ell(\boldsymbol{v})$, in the $M/M/c/m$ queue with $\boldsymbol{v}_1 = (v_{11}, v_{12})$ denoting the vector of the interarrival and service rates and $\boldsymbol{v}_2 = (m)$ denoting the buffer size. Assume that v_{11} is fixed, while v_{12} and m may vary. In particular, set $v_{11} = 1$, while $v_{12} = 2, 1.5, 1.4, 1.25$, and $m = 5, 10, 15$. According to (4.6), we first select some buffer size from the set $\{5, 10, 15\}$, say $m = 5$. Next we choose the reference parameter value for v_{12} as the one that corresponds with the highest traffic intensity, ρ_0, among the values $v_{12} = 2, 1.5, 1.4, 1.25$; in our case $v_{12} = 1.25$ (slowest service rate). Finally, we make three separate runs, with $m = 5, 10, 15$, to estimate $\ell(\boldsymbol{v})$ for the above $r_1 \times r_2 = 12$ scenarios. Here, the SFED estimator is more efficient (only 3 runs instead of 12). Numerical experiments also indicate that the 3 SF runs are more accurate than its crude Monte Carlo (CMC) counterpart.

Example 4.4. Consider the estimation of the steady-state expected waiting time, $\ell(\boldsymbol{v})$ for r $M/M/1/m$ queues in tandem, where $\boldsymbol{v}_1 = (v_{11}, \dots, v_{1r})$ and $\boldsymbol{v}_2 = (v_{21}, \dots, v_{2r}) = (m_1, \dots, m_r)$ are the vectors of service rates and buffer sizes, respectively. In this case, the full factorial ED method applied to $k_1 + k_2 = 2r$ factors, would require a total of 2^{2r} Monte Carlo experiments, whereas the full factorial applied to $k_2 = r$ factors requires only 2^r such experiments. Thus, the latter is approximately 2^r times faster than the former, since the overhead of computing $\tilde{W}_t(\boldsymbol{v}_1)$, in the corresponding likelihood ratio estimators, is relatively small. This speed up has been confirmed by various simulation studies. Further reduction of computer time can be realized by assuming a first order polynomial response curve in $\boldsymbol{v}_2$, and executing not 2^{k_2} runs but only $k_2 + 1$ runs.

5. Optimization

Consider the following mathematical programming problem:

$$
\begin{aligned}
&(\mathrm{P}_0) \\
&\text{minimize} && \ell_0(\boldsymbol{v}), && \boldsymbol{v} \in V, \\
&\text{subject to} && \ell_j(\boldsymbol{v}) \leqslant 0, && j = 1, \dots, k, \\
& && \ell_j(\boldsymbol{v}) = 0, && j = k+1, \dots, M
\end{aligned}
\tag{5.1}
$$

where

$$\ell_j(v) = \mathbb{E}_v(L_j) = \frac{\mathbb{E}_v(\sum_{t=1}^{\tau} L_{jt})}{\mathbb{E}_v \tau}, \quad j = 0, 1, \ldots, M, \tag{5.2}$$

are the steady-state expected performances corresponding to the output processes $\{L_{jt}\}$.

To estimate the optimal solution of this problem (P_0) from simulation, we first approximate it by its stochastic counterpart (see (5.4) below), and then solve this counterpart problem by standard techniques of mathematical programming (see also a recent survey on optimization in simulation by Fu, 1994).

In order to construct such a stochastic counterpart, we argue as follows. Consider first the estimators of $\ell_j(v)$, defined in (3.11):

$$\bar{\ell}_{jN}(v) = \frac{\sum_{i=1}^{N}\sum_{t=1}^{\tau_i} L_{jti}\tilde{W}_{ti}}{\sum_{i=1}^{N}\sum_{t=1}^{\tau_i} \tilde{W}_{ti}}, \quad j = 0, 1, \ldots, M. \tag{5.3}$$

Second, viewing $\bar{\ell}_{jN}(v)$ as functions of v rather than as estimators for fixed v, we define the *stochastic counterpart* of (P_0) as follows:

$$\begin{array}{lll} (\bar{\mathrm{P}}_N) & & \\ \text{minimize} & \bar{\ell}_{0N}(v), & v \in V, \\ \text{subject to} & \bar{\ell}_{jN}(v) \leqslant 0, & j = 1, \ldots, k, \\ & \bar{\ell}_{jN}(v) = 0, & j = k+1, \ldots, M. \end{array} \tag{5.4}$$

Notice that as soon as the input sample $Z_1, \ldots, Z_N$, is generated, the functions $\tilde{W}_{ti}$ and hence $\bar{\ell}_{jN}(v)$, $j = 0, \ldots, M$, are given *explicitly* through the known density functions $f(Z_i, v)$: substitute $z_1, \ldots, z_N$ into $\tilde{W}_{ti} = \Pi_{j=1}^{t} W_j$ with $W_j = f(z_j)/g(z_j)$. The corresponding gradients $\nabla\bar{\ell}_{jN}(v)$ can be calculated from a single simulation by the SF method according to (2.12). Consequently, in principle the optimization problem $(\bar{\mathrm{P}}_N)$ can be solved by standard methods of mathematical programming (e.g., Rubinstein, 1986). The resulting optimal value $\ell_N(\bar{v})$ and the optimal solution $\bar{v}_N$ of the program $(\bar{\mathrm{P}}_N)$ provide estimators of the optimal value $\ell(v^*)$ and the optimal solution v^* of the program (P_0), respectively. Note that this solution is feasible, since we assumed that the sample functions $L_j(y)$ *do not depend* on v.

The algorithm for estimating the optimal solution v^* of the program (P_0) while using the stochastic counterpart $(\bar{\mathrm{P}}_N)$ can be written as follows.

Algorithm 5.1.

(1) Generate a random sample $Z_{11}, \ldots, Z_{1\tau_1}, \ldots, Z_{N1}, \ldots, Z_{N\tau_N}$ from $g(z)$.

(2) Generate the output (sample performance) processes L_{jti} and the likelihood ratio process $\tilde{W}_{ti}(v), j = 0, \ldots, M, t = 1, \ldots, \tau_i; i = 1, \ldots, N$.

(3) Solve the program $(\bar{\mathrm{P}}_N)$ by the techniques of mathematical programming.

(4) Deliver the solution $\bar{v}_N$ of $(\bar{\mathrm{P}}_N)$ as an estimator of v^*.

Consider the following *unconstrained* program:

$$\begin{array}{l} (\mathrm{P}_0) \\ \text{minimize } \ell(v), \quad v \in V. \end{array} \tag{5.5}$$

Its stochastic counterparts can be written as

$$\begin{array}{l} (\bar{\mathrm{P}}_N) \\ \text{minimize } \ell_N(v), \quad v \in V. \end{array} \tag{5.6}$$

Before turning to numerical results with the stochastic counterpart (5.6), we assume the following: (1) The parameter set V is given by

$$V = \{v : \mathbf{0} \leqslant \boldsymbol{\rho}(v) \leqslant \boldsymbol{\rho}^0 < \mathbf{1}, \quad \boldsymbol{\rho} = (\rho_1, \ldots, \rho_r)\}, \tag{5.7}$$

where $\rho_k = \rho_k(v)$, $k = 1, \ldots, r$, is the traffic intensity at the k-th queue, r is the number of nodes in the network, and the inequalities between the vectors must be taken componentwise. (2) The expected performance $\ell(v)$ is given as

$$\ell(v) = c\,\mathbb{E}_v L_t + \sum_{k=1}^{r} b_k v_k, \tag{5.8}$$

where L_t is the steady-state sojourn time process, c is the cost of a waiting customer, $v = (v_1, \ldots, v_r)$ is the service rate vector, and b_k is the cost per unit increase (decrease) of v_k. Note that under some mild regularity conditions, $\mathbb{E}_v L_t$ is a strictly convex differentiable function with respect to v. Thus, v^* is a unique minimizer of (P_0) over the convex region V.

We can then solve the stochastic counterpart ($\bar{P}_N$) by using the following nonlinear system of equations (first order conditions for extreme values):

$$\nabla \bar{\ell}_N(\boldsymbol{v}) = \mathbf{0}, \quad \boldsymbol{v} \in V.$$

In particular, for a queueing model with a single node ($r = 1$) and FIFO discipline $\bar{P}_N$ reduces to

$$\nabla \bar{\ell}_N((v) = c \left\{ \frac{\sum_{i=1}^{N} \sum_{t=1}^{\tau_i} L_{ti} \nabla \tilde{W}_{ti}}{\sum_{i=1}^{N} \sum_{t=1}^{\tau_i} \tilde{W}_{ti}} - \frac{\sum_{i=1}^{N} \sum_{t=1}^{\tau_i} L_{ti} \tilde{W}_{ti}}{\sum_{i=1}^{N} \sum_{t=1}^{\tau_i} \tilde{W}_{ti}} \cdot \frac{\sum_{i=1}^{N} \sum_{t=1}^{\tau_i} \nabla \tilde{W}_{ti}}{\sum_{i=1}^{N} \sum_{t=1}^{\tau_i} \tilde{W}_{ti}} \right\} + \boldsymbol{b} = \mathbf{0}, \quad \boldsymbol{v} \in V, \tag{5.9}$$

where

$$\tilde{W}_{ti} = \prod_{k=1}^{t} W(\boldsymbol{Z}_{ki}, \boldsymbol{v})$$

and

$$W(\boldsymbol{Z}, \boldsymbol{v}) = \frac{f(\boldsymbol{Z}, \boldsymbol{v})}{g(\boldsymbol{Z})}.$$

Remark 5.1. It is shown in Rubinstein and Shapiro (1993) that while solving the stochastic counterpart

$$\nabla \bar{\ell}_N(\boldsymbol{v}) = \mathbf{0}, \quad \boldsymbol{v} \in V,$$

where

$$V = \{\boldsymbol{v} : 0 < \rho(\boldsymbol{v}) \leqslant \rho^0(\boldsymbol{v}) < 1\},$$

a "good" reference parameter ρ_0 must be chosen, *either equal to ρ^0 or moderately larger than ρ^0.*

We now present *numerical* results for the stochastic counterpart ($\bar{P}_N$) for several queueing decision models, assuming that $g(\boldsymbol{y}) = f(\boldsymbol{y}, \boldsymbol{v}_0)$ where $\boldsymbol{v}_0$ is the reference parameter.

Example 5.1 ($M/M/1$ queue). Let λ and v be the arrival and service rates, and let v be the decision parameter. Taking into account the analytical result $\mathbb{E}_v L = 1/(v - \lambda)$, it is readily seen that the true optimal v^* that minimizes the performance measure given in (5.8) is $v^* = \lambda + (c/b)^{1/2}$.

Table 5.1 represents theoretical values of v^*, point estimators $\bar{v}_N$, and 95% confidence intervals for v^* (denoted 95%CI), as functions of b ($\rho^* = \lambda/v^*$ and $\alpha = [\rho_0 - \rho^*]/\rho^*$). We chose $\lambda = 1$, $c = 1$, the reference traffic intensity $\rho_0 = 0.8$, and ran $N = 10{,}000$ cycles (approximately 50,000 customers). Note that *all* estimators $\bar{v}_N(b)$ were obtained *simultaneously* from a single simulation run (with $\rho_0 = 0.8$) by solving the system of equations (5.9) for different values of b.

It is readily seen that the estimator $\bar{v}_N$ performs reasonably well for $\rho \in (0.3, 0.8)$. The poor performance of the estimator $\bar{v}_N$ for $\rho > 0.8$ is caused by the violation of the requirement of Remark 5.1 (according to that remark we must have $\rho \leqslant \rho_0 = 0.8$, whereas in fact we have $0.88 \geqslant \rho \geqslant \rho_0 = 0.8$.). The poor performance for $\rho < 0.2$ is the result of very large relative perturbations ($\alpha > 3$) in the likelihood ratio process $\tilde{W}_t$.

Example 5.2 ($M/G/1$ queue). Assume that the pdf of service time y is given by

$$f(y, p) = \begin{cases} p, & \text{if } y = a_1, \\ 1 - p, & \text{if } y = a_2, \end{cases}$$

where $0 < a_1 < a_2$ and $0 < p < 1$. By the Pollaczek-Khinchin formula (e.g., Gross and Harris, 1985) the expected sojourn time can then be written as

$$\ell(p) = \mathbb{E}_p L = \beta_1 + \frac{\lambda \beta_2}{2(1 - \lambda \beta_1)},$$

where λ still denotes the arrival rate,

$$\beta_1 = \mathbb{E} Y = p a_1 + (1 - p) a_2,$$

$$\beta_2 = \mathbb{E} Y^2 = p a_1^2 + (1 - p) a_2^2.$$

In this case the likelihood ratio W in (5.9) reduces to

$$W(Z, p) = \left(\frac{p}{p_0}\right)^{\frac{a_2 - Z}{a_1 - a_2}} \left(\frac{1 - p}{1 - p_0}\right)^{\frac{Z - a_1}{a_2 - a_1}}.$$

Table 5.1
Performance of the stochastic counterpart $(\bar{\mathrm{P}}_N)$ for the $M/M/1$ queue with reference traffic intensity $\rho_0 = 0.8$

ρ^*	b	α	$\bar{v}_N(b)$	v^*	95%CI
0.88	53.77	−0.091	1.278	1.136	0.65, 1.81
0.85	32.11	−0.058	1.208	1.176	0.91, 1.51
0.8	16.00	0.000	1.261	1.255	1.15, 1.37
0.7	5.444	0.143	1.433	1.429	1.33, 1.54
0.6	2.225	0.333	1.654	1.667	1.58, 1.73
0.5	1.000	0.600	1.971	2.000	1.91, 2.03
0.4	0.444	1.000	2.467	2.500	2.42, 2.51
0.3	0.184	1.667	3.324	3.333	3.25, 3.39
0.2	0.063	3.000	4.947	5.000	4.74, 5.15
0.1	0.012	7.000	9.741	10.00	9.42, 10.06

Table 5.2
Performance of the stochastic counterpart $(\bar{\mathrm{P}}_N)$for the $M/G/1$ queue with reference traffic intensity $\rho_0 = 0.85$

ρ^*	b	α	p^*	$\bar{p}_N$	95%CI
0.80	10.32	0.062	0.571	0.559	0.34, 0.78
0.70	4.978	0.214	0.714	0.716	0.65, 0.73
0.65	3.843	0.307	0.786	0.804	0.77, 0.83
0.60	3.106	0.417	0.857	0.868	0.82, 0.91
0.55	2.601	0.545	0.928	0.951	0.91, 0.99
0.50	2.240	0.700	1.000	1.000	0.95, 1.05

Table 5.3
Performance of the stochastic counterpart $(\bar{\mathrm{P}}_N)$ for two $M/M/1$ queues in tandem

ρ_1^*	ρ_2^*	b_1	b_2	α_1	α_2	$\ell(v^*)$	$\ell(\bar{v}_N)$	95%CI	γ
0.60	0.88	2.25	53.77	0.33	−0.09	73.69	76.01	72.3, 79.7	0.200
0.60	0.80	2.25	16.00	0.33	0.00	29.25	29.28	28.0, 30.6	0.010
0.60	0.60	2.25	2.25	0.33	0.143	10.50	10.48	10.3, 10.6	0.004
0.60	0.40	2.25	0.44	0.33	1.00	7.03	7.02	6.92, 7.12	0.004
0.60	0.20	2.25	0.06	0.33	3.00	5.81	5.81	5.72, 5.90	0.002
0.60	0.10	2.25	0.01	0.33	7.00	5.49	5.48	5.38, 5.58	0.003
0.88	0.60	53.77	2.25	−0.09	0.33	73.69	75.65	69.3, 82.0	0.154
0.80	0.60	16.00	2.25	0.00	0.33	29.25	29.30	28.1, 30.5	0.008
0.40	0.60	0.44	2.25	1.00	0.33	7.03	7.02	6.93, 7.12	0.002
0.20	0.60	0.06	2.25	3.00	0.33	5.81	5.80	5.71, 5.90	0.013
0.10	0.60	0.01	2.25	7.00	0.33	5.48	5.47	5.38, 5.57	0.007

Table 5.2 presents data similar to those of Table 5.1. We chose $\lambda = 0.6$, $p = 0.5$, $a_1 = 0.5$, $a_2 = 1.2$ ($\rho_0 = 0.85$), $c = 1$, $\rho^0 = 0.7$, and ran the $M/G/1$ queue for $N = 15{,}000$ cycles (approximately 100,000 customers). Following Remark 5.1 and taking into account that $\rho^0 = 0.7$, we chose the reference traffic intensity moderately larger than ρ^0, namely $\rho_0 = 0.85$.

Example 5.3 (Tandem queue). Table 5.3 represents data similar to that of table 5.1 for two $M/M/1$ queues in tandem, while using the stochastic counterpart ($\bar{\text{P}}_N$). It includes $\gamma = (\|v^* - \bar{v}_N\|)/\|v^*\|$, $v = (v_1, v_2)$, as functions of (b_1, b_2), $(\rho_1^* = \lambda/v_1^*, \rho_2^* = \lambda/v_2^*)$, and $(\alpha_1, \alpha_2) = ([\rho_{01} - \rho_1^*]/\rho_1^*, [\rho_{02} - \rho_2^*]/\rho_2^*)$. Again we choose $\lambda = 1$, $c = 1$, now with reference traffic intensities $\rho_{01} = 0.8$, $\rho_{02} = 0.6$, and run $N = 10{,}000$ cycles (approximately 50,000 customers).

It is readily seen from Tables 5.2 and 5.3 that the SF method performs well, provided $\rho^* \leqslant \rho_0$.

Itzhaki (1994) gives extensive supporting numerical results with both unconstrained and constrained mathematical programming methods (P_0) for different network topologies, different dimensionalities n ($1 \leqslant n \leqslant 100$) of the decision vector v (being the vector with the parameters of the interarrival and service time distributions and the routing probabilities), while using the research package QNSO.

6. Conclusion

The Score Function (SF) method uses a single simulation run to *simultaneously* estimate the simulation response and its derivatives, for different values of the parameters of the distribution function of the simulation inputs. SF applies to both discrete-event static systems (DESS) and discrete-event dynamic systems (DEDS). Parameters that do not occur in the input distribution, but that are structural parameters, can be examined through classic experimental designs (ED). SF and ED can be combined to obtain further efficiency gains. The optimal values of the distributional parameters can be obtained by solving the stochastic counterpart of the original mathematical programming problem.

Acknowledgments

This work was supported by the Technion V.P.R. Fund - B.R.L. Bloomfield Industrial Management R.F., and Center for Economic Research, Tilburg University. A previous version was reviewed by Onno Boxma (Center for Mathematics and Computer Science/Center for Economic Research), who pointed out several typos.

References

Aleksandrov, V.M., Sysoyev, V.I., and Shemeneva, V.V. (1968), "Stochastic Optimization", *Engineering Cybernetics* 5, 11-16.

Asmussen, S. (1987), *Applied Probability and Queues*, Wiley, New York.

Ermakov, C.M., and Mikhailov, G.A. (1982), *Statistical Modeling*, Nauka, Moscow (in Russian).

Feuerverger, A., McLeish, D.L., and Rubinstein, R.Y. (1986)., "A cross-spectral method for sensitivity analysis of computer simulation models", *Comptes Rendus: Mathematical Reports of the Academy of Sciences*, Royal Society of Canada, VIII/5, 335-339.

Fu, C.M. (1994), " Optimization in simulation: A review", *Annals of Operations Research* 53, 199-247.

Glasserman, P. (1991), *Gradient Estimation via Perturbation Analysis*, Kluwer, Norwell, MA.

Glynn, P.W. (1990), "Likelihood ratio gradient estimation for stochastic systems", *Communications of the ACM* 33/10, 75-84.

Gross, D., and Harris, C. (1985), *Fundamentals of Queueing Theory*, Wiley , New York.

Ho, Y.C., and Cao, X.R. (1991), *Perturbation Analysis of Discrete Event Systems*, Kluwer, Boston, MA.

Itzhaki, Ya. (1994), "Stochastic optimization of open queueing networks by the score function method", Master's Thesis, Technion, Haifa, Israel.

Kleijnen, J.P.C. (1987), *Statistical Tools for Simulation Practitioners*, Marcel Dekker, New York.

Kleijnen, J.P.C. (1994), "Sensitivity analysis and optimization of simulation models", *Proceedings of the 1994 European Simulation Symposium*, The Society for Computer Simulation, California.

Kreimer, J. (1984), "Stochastic optimization - An adaptive approach", D.Sc. Thesis, Technion, Haifa, Israel.

Kriman, V. (1994), "Sensitivity analysis of *GI/GI/m/B* queues with respect to buffer size by the score function method", *Stochastic Models*, to appear.

L'Ecuyer, P.L (1990), "A unified version of the IPA, SF, and LR gradient estimation techniques", *Management Science* 36/11, 1364-1383.

Marti, K. (1990). "Stochastic optimization methods of structural design", *Zeitschrift für Angewandte Mathematik und Mechanik* 4, T742-T745.

Mikhailov, G.A. (1967). "Calculation of system parameter derivatives of functionals of the solutions to the transport equations", *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki* 7, 915.

Miller, L.B. (1967), "Monte Carlo analysis of reactivity coefficients in fast reactors: general theory and applications", ANL-7307 (TID-4500), Argonne National Laboratory, IL.

Reiman, M.I., and Weiss, A. (1989), "Sensitivity analysis for simulations via likelihood ratios", *Operations Research* 37/5, 830–844.

Rubinstein, R.Y. (1969). "Some problems in Monte Carlo optimization", PhD Thesis, Riga, Latvia.

Rubinstein, R.Y. (1986), *Monte Carlo Optimization Simulation and Sensitivity of Queueing Network*, Wiley, New York.

Rubinstein, R.Y. (1992), " Sensitivity analysis of discrete event systems by the "Push out" method", *Annals of Operations Research* 39, 229-251.

Rubinstein, R.Y., and Kreimer, J., (1983),. "About one Monte Carlo method for solving linear equations", *Mathematics and Computers in Simulation* XXV, 321–334.

Rubinstein, R.Y., and Shapiro, A. (1993), *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization via the Score Function Method*, Wiley, New York.

Uryas'ev, S. (1994), "Analytic perturbation analysis for DEDS with discontinuous sample-path functions", Manuscript, Brookhaven National Laboratory, Upton, NY.