Introduction to Stochastic Search and Optimization: Estimation,
Simulation, and Control. James C. Spall
Copyright © 2003 John Wiley & Sons, Inc.,
ISBN: 0-471-33052-3

CHAPTER 15

SIMULATION-BASED OPTIMIZATION II: STOCHASTIC GRADIENT AND SAMPLE PATH METHODS

The gradient-free methods that were emphasized in Chapter 14 essentially treat the simulation as a black box, assuming minimal knowledge of the inner workings of the simulation. In some cases, however, one *does* have knowledge of how the simulation is constructed. This chapter describes how that knowledge can be exploited to enhance simulation-based optimization. Calculation of the stochastic gradient (as in Chapter 5) provides the mechanism through which the knowledge is used. Over the past several decades, a large amount of effort has gone into developing stochastic gradient methods for simulation-based optimization. This chapter is a sampling of available methods.

Section 15.1 covers some general issues in gradient estimation, building on some of the results in the stochastic gradient analysis of Chapter 5. This section includes a discussion on the role of interchange of derivative and integral. Section 15.2 considers two important cases of gradient estimation methods in a simulation-based context—the likelihood ratio (score function) method and the infinitesimal perturbation method. The resulting search algorithms have an interpretation as a root-finding (Robbins–Monro) stochastic approximation algorithm (Chapter 4). Section 15.3 considers the implementation of the stochastic gradient-based optimization methods, including a numerical example. Section 15.4 presents the sample path method of simulation-based optimization. Here, the *stochastic* simulation-based problem is converted to a *deterministic* problem that provides a solution closely approximating the stochastic problem. Section 15.5 closes the chapter with some concluding remarks and a mention of some other methods for simulation-based gradient estimation.

15.1 FRAMEWORK FOR GRADIENT ESTIMATION

15.1.1 Some Issues in Gradient Estimation

Let us consider some of the main gradient-based methods for optimization in a simulation-based context. Such methods have been a major focus of research in simulation modeling for many years. The primary motivation for use of such methods is their potential for faster convergence (i.e., fewer simulation runs) than the nongradient methods discussed in Chapter 14. The focus continues to be on the optimization problem introduced in Section 14.1, namely, to minimize

$$L(\boldsymbol{\theta}) = E[Q(\boldsymbol{\theta}, V)],$$

where V represents the amalgamation of the (pseudo) random effects in the simulation (presumably manifesting themselves in a way indicative of the random effects in the actual system) and Q represents a *sample* realization of the loss function calculated from running the simulation.

As discussed in Chapter 1 and elsewhere, the gradient $\boldsymbol{g}(\boldsymbol{\theta}) = \partial L/\partial\boldsymbol{\theta}$—or a noisy estimate of the gradient—is an important tool in optimization of differentiable loss functions. A number of methods exist for estimating the gradient as it arises in the simulation-based context. We discuss methods here that use the structure of the root-finding stochastic approximation (SA) algorithm (Chapters 4 and 5) and the structure of deterministic nonlinear programming (briefly introduced in Section 1.4). While the emphasis in most stochastic or deterministic gradient-based methods is local optimization, recall from Section 8.4 that gradient-based methods can also be used in *global* optimization via the principle of injected noise with annealing.

Within the root-finding context, Section 5.3 was a brief introduction to one of the more popular approaches—the infinitesimal perturbation analysis (IPA) method. This chapter will broaden our horizons, considering a general setting that includes IPA as a special case. Another method that has received considerable attention is the *likelihood ratio*—sometimes called *score function*—approach (LR/SF). LR/SF for simulation-based optimization is discussed, for example, in Rubinstein (1986, Sect. 3.6), Glynn (1987), L'Ecuyer (1990a), Fu (1994), and Rubinstein and Melamed (1998, Part II). The aim of both IPA and LR/SF is to produce an estimate of $\boldsymbol{g}(\boldsymbol{\theta}) = \partial L/\partial\boldsymbol{\theta}$. While this estimate has several applications, one of the most important is in the context of a root-finding SA-based optimization method. Recall from Chapter 5 that root-finding SA with a noisy measurement of the gradient is called a *stochastic gradient method.* For purposes of efficiency, an aim in simulation-based optimization using the stochastic gradient formulation is to produce the gradient measurement with only one simulation run (versus, say, the $2p$ runs of FDSA or the two runs of SPSA, as discussed in Section 14.3[1]). Another aim in the gradient estimation is to produce *unbiased* estimates of $\boldsymbol{g}(\boldsymbol{\theta})$.

[1] Recall from Section 7.7 that there is also a one-measurement version of SPSA (versus the standard two-measurement form). This yields an SPSA gradient estimate from one simulation run without the need for the modeling information required in the gradient-based SA methods. However, as discussed in Section 7.7, this one-run SPSA is generally inferior to the standard two-run form. Further, it has a slower asymptotic rate of convergence than the gradient-based methods.

This chapter also considers a method that is based on converting the inherently stochastic simulation-based problem to a potentially easier-to-solve deterministic problem. The solution via this *sample path method* can be made arbitrarily close to the true optimal solution $\boldsymbol{\theta}^*$ by increasing the number of simulation runs in an averaging process. One can use deterministic nonlinear programming methods (e.g., Bazaraa et al., 1993) to find the solution in the sample path method. Because of the relatively well-developed methods for constrained optimization in nonlinear programming, this approach has particular appeal when there are difficult constraints to be satisfied.

In discussing the methods for gradient estimation, it is useful to distinguish between two types of parameters:

> ***Distributional parameters*** are those elements of $\boldsymbol{\theta}$ that enter via their effect on the probability distribution of the random terms V in the system. So, for example, if a process involves $N(\mu, \sigma^2)$-generated inputs, where μ and σ^2 are subject to design considerations (i.e., are a part of $\boldsymbol{\theta}$), then μ and σ^2 would be distributional parameters. These parameters enter the probability density or mass function for the random effects V, $p_V(\cdot\,|\boldsymbol{\theta})$, discussed below.
>
> ***Structural parameters*** in $\boldsymbol{\theta}$ have their effect directly on the output Q, typically involving an input–output relationship due, say, to the physics of a process. The plant layout problem in Chapter 14 (Examples 14.1 and 14.5) depicts a case with structural parameters when the probability distributions for the random effects in the process (e.g., the distribution of the parts arrivals in the plant layout example) do not depend on $\boldsymbol{\theta}$. The machine locations ($\boldsymbol{\theta}$) have a direct effect on the product output Q by governing the flow of material and people that have an impact on operations. For general problems, structural parameters are those entering the observed cost $Q(\boldsymbol{\theta}, V)$.

It is easy to envision cases where the division between distributional and structural parameters is not obvious. See, for example, L'Ecuyer (1990a), Fu and Hu (1997, pp. 3–4 and 11–12), Rubinstein and Melamed (1998, pp. 115–117), and Subsection 5.1.1 (Example 5.1), Subsection 15.1.2, and Section 15.2 here. In fact, the division is often somewhat arbitrary in the sense that V could represent the raw uniform random variables generated for the simulation. Then all the distributional transformation operations are embedded in Q, implying that all parameters $\boldsymbol{\theta}$ are structural parameters. In a mathematical sense, therefore, the distinction may not always be critical. But, for *applications*, the distinction may be important since the parameters may be allocated to the two categories in a physically meaningful way. As in Chapter 14, we have adopted the common approach here of having V be the "natural" random variables of the process (rather than the underlying $U(0, 1)$ random variables) because of the easier physical interpretability of the analysis. Sections 15.2 and 15.3 show that the long-standing division between two of the most popular gradient estimators

(LR/SF and IPA) rests critically on the distinction between whether $\boldsymbol{\theta}$ represents distributional or structural parameters.

Initially, the LR/SF and IPA approaches developed largely independent of each other. However, it has long been recognized that there is a common basis for the approaches (e.g., L'Ecuyer, 1990a). We adopt this more general framework here, but first let us make a few comments about the original versions of LR/SF and IPA. A critical assumption of the original LR/SF formulation (e.g., Aleksandrov et al., 1968; Glynn, 1987) is that the elements of $\boldsymbol{\theta}$ enter only as *distributional* parameters rather than through some explicit structure in the problem affecting Q. So, the original LR/SF would not apply in the plant layout problem of Chapter 14 (Examples 14.1 and 14.5) when $\boldsymbol{\theta}$ corresponds to the machine locations as they appear in the sample loss Q. LR/SF would apply in the case where $\boldsymbol{\theta}$ appears in the distribution function for V, representing the means of the service times for a set of machines as derived from prior knowledge of how location and service time are related (so that optimizing the service times is tantamount to optimizing the machine locations).

In a complementary manner, the original IPA approach (e.g., Ho, 1987) applies to problems where the $\boldsymbol{\theta}$ dependence appears only as structural parameters in the sample loss function Q, not in the distribution function generating the V. Of course, much research has occurred since the original work in extending the LR/SF and IPA approaches to broader settings; a summary is in Rubinstein and Melamed (1998, pp. 112–118). We will adopt the broader setting where $\boldsymbol{\theta}$ can enter both (or either) of Q or the distribution function.

15.1.2 Gradient Estimation and the Interchange of Derivative and Integral

Consistent with previous usage, suppose that there exists a probability density or mass function for the cumulative random effects V in the simulation, $p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta})$, where $\boldsymbol{\upsilon}$ represents the dummy variable for use in integrals associated with expected value calculations. In general, this density or mass function depends on $\boldsymbol{\theta}$, although there are special cases where it does not (i.e., $p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta}) = p_V(\boldsymbol{\upsilon})$). For the discussion below, let us assume that $p_V = p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta})$ is a *density* function. When p_V is a mass function, we simply replace the indicated integrals with sums. The "bottom line" result below, estimate (15.4), applies for either the density or mass case. Recall the familiar formulation

$$L(\boldsymbol{\theta}) = E[Q(\boldsymbol{\theta}, V)] = \int_{\Lambda} Q(\boldsymbol{\theta}, \boldsymbol{\upsilon}) p_V(\boldsymbol{\upsilon} | \boldsymbol{\theta})\, d\boldsymbol{\upsilon}, \qquad (15.1)$$

where Λ is the domain for V (i.e., $p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta}) = 0$ outside of Λ).

Assume that Q and p_V are differentiable with respect to $\boldsymbol{\theta}$ (this assumption is also needed when p_V is a mass function). A central issue in the necessary gradient calculations of this section is the question of validity of the following interchange of the order of integration and differentiation:

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \frac{\partial}{\partial \boldsymbol{\theta}} \int_{\Lambda} Q(\boldsymbol{\theta}, \boldsymbol{\upsilon}) p_V(\boldsymbol{\upsilon} \mid \boldsymbol{\theta}) d\boldsymbol{\upsilon} \overset{?}{=} \int_{\Lambda} \frac{\partial [Q(\boldsymbol{\theta}, \boldsymbol{\upsilon}) p_V(\boldsymbol{\upsilon} \mid \boldsymbol{\theta})]}{\partial \boldsymbol{\theta}} d\boldsymbol{\upsilon}.$$

The following theorem provides sufficient conditions for this interchange to hold. Recall that $\Theta \times \Lambda = \{(\boldsymbol{\theta}, \boldsymbol{\upsilon}): \boldsymbol{\theta} \in \Theta, \boldsymbol{\upsilon} \in \Lambda\}$ (the Θ here is the constraint domain for $\boldsymbol{\theta}$ introduced in Chapter 1). The result is a direct application of Theorem A.3 in Appendix A. Similar theorems are given in L'Ecuyer (1990a), Glasserman (1991a, pp. 49–53, 102–104; 1991b), and Rubinstein and Melamed (1998, pp. 123–124).

Theorem 15.1. Suppose that $\Theta \subset \mathbb{R}^p$ is an open set. Let $Q \times p_V$ and $\partial(Q \times p_V)/\partial \boldsymbol{\theta}$ be continuous on $\Theta \times \Lambda$. Suppose that there exist nonnegative functions $q_0(\boldsymbol{\upsilon})$ and $q_1(\boldsymbol{\upsilon})$ such that

$$\left| Q(\boldsymbol{\theta}, \boldsymbol{\upsilon}) p_V(\boldsymbol{\upsilon} \mid \boldsymbol{\theta}) \right| \leq q_0(\boldsymbol{\upsilon}); \quad \left\| \frac{\partial [Q(\boldsymbol{\theta}, \boldsymbol{\upsilon}) p_V(\boldsymbol{\upsilon} \mid \boldsymbol{\theta})]}{\partial \boldsymbol{\theta}} \right\| \leq q_1(\boldsymbol{\upsilon}) \text{ for all } (\boldsymbol{\theta}, \boldsymbol{\upsilon}) \in \Theta \times \Lambda,$$

where $\int_{\Lambda} q_0(\boldsymbol{\upsilon}) d\boldsymbol{\upsilon} < \infty$ and $\int_{\Lambda} q_1(\boldsymbol{\upsilon}) d\boldsymbol{\upsilon} < \infty$. Then

$$\frac{\partial}{\partial \boldsymbol{\theta}} \int_{\Lambda} Q(\boldsymbol{\theta}, \boldsymbol{\upsilon}) p_V(\boldsymbol{\upsilon} \mid \boldsymbol{\theta}) d\boldsymbol{\upsilon} = \int_{\Lambda} \frac{\partial [Q(\boldsymbol{\theta}, \boldsymbol{\upsilon}) p_V(\boldsymbol{\upsilon} \mid \boldsymbol{\theta})]}{\partial \boldsymbol{\theta}} d\boldsymbol{\upsilon}.$$

Although the conditions of Theorem 15.1 may appear rather abstract, there is a strong association with practical concerns in simulations. In fact, it is not automatic that the conditions of Theorem 15.1 will hold in many practical simulations. For example, in the case where $\boldsymbol{\theta}$ represents only structural parameters (so $\boldsymbol{\theta}$ does not enter p_V), the continuity conditions on Q and $\partial Q/\partial \boldsymbol{\theta}$ require that small changes in $\boldsymbol{\theta}$ (the infinitesimal perturbations of IPA) correspond to small changes in the simulation output that is relevant to the computation of Q. In a discrete-event simulation, this is equivalent to assuming that small perturbations in $\boldsymbol{\theta}$ do not change the sequence of events but rather change in a small way only the occurrence times or possibly the magnitude of the events.

As discussed in several contexts in Chapter 14 (especially in the discussion of common random numbers—Section 14.4), small changes in $\boldsymbol{\theta}$ may sometimes lead to large changes in the output via a change in the way the random effects V are processed. If, for instance, the aim is to maximize the mean number of times that a queue is clear when a customer arrives (e.g., maximize the number of regeneration cycles in a setting such as Figure 14.1), then Theorem 15.1 will not apply when a small change in $\boldsymbol{\theta}$ causes one of the customers currently facing a zero wait to face a nonzero wait (e.g., a loss of one of the regeneration cycles in Figure 14.1). In general, because of the flexibility one may have in defining p_V vis-à-vis Q, there may be cases where one definition of p_V and Q may lead to a violation of the theorem conditions while another equivalent definition (in the sense that L is the same for any $\boldsymbol{\theta}$) will satisfy the conditions (recall that the conditions are *sufficient*, but not necessary). Example 5.1 (Subsection 5.1.1) illustrated the flexibility in a particular stochastic gradient problem; the same principles apply in the simulation-based context.

The nonuniqueness in defining distributional and structural parameters discussed above is one way in which equivalent definitions may lead to different conclusions regarding the applicability of Theorem 15.1. Example 15.1 and Exercise 15.1 illustrate this principle in a reliability problem. Similarly, sometimes a small modification in the goals of the optimization may alter the applicability of Theorem 15.1. For example, in the queuing problem of the preceding paragraph, if the aim is changed from maximizing the number of zero waits to minimizing mean wait time, then a small change in $\boldsymbol{\theta}$ may have only a small effect on Q, enhancing the applicability of Theorem 15.1.

Before presenting successful examples of the interchange of derivative and integral in Sections 15.2 and 15.3, let us present a conceptual example where the interchange is *not* allowed by Theorem 15.1. This example is a realistic problem in reliability with multiple components. Following some subsequent discussion on gradient estimates for implementation we will present a second example of the failure of interchange. Suri (1989), L'Ecuyer (1990a), and Fu and Hu (1997, pp. 146–147) present a number of other examples where the interchange of derivative and integral is not allowed (see also Example A.3 in Appendix A here). These include communications network simulations, repair/reliability models different from those below, and general queuing networks. The fundamental principle in these examples is that small changes in $\boldsymbol{\theta}$ can have a dramatic effect on the simulation output (e.g., number of customers processed in a queue).

Example 15.1—Reliability system (L'Ecuyer, 1990a). This example shows how a small change in $\boldsymbol{\theta}$ can sometimes lead to a large change in Q due to a different ordering of events. Such a discontinuous Q violates one of the conditions of Theorem 15.1. Consider a system with multiple identical components, each operating independently with a random failure time given by a density function that is strictly positive on the interval $(0, T)$, T being the maximum time of interest. The aim is to develop a component replacement strategy that minimizes the total costs over time $(0, T)$ due to component failures and replacements. The costs include a failure cost every time a component fails, a fixed cost for having to intervene in the system to replace one or more components, and a replacement cost that is directly proportional to the number of components that are to be replaced. Preventive replacements are used to reduce the number of component failures and to reduce the number of times that system intervention (with its associated fixed cost) is required.

Suppose that $\boldsymbol{\theta}$ contains two parameters: $t_1 > t_2 > 0$. Whenever a component fails or reaches age t_1 that component is replaced together with all other components having an age at least t_2. Suppose that V represents the collection of natural component failure times (i.e., the failure times that would occur if $t_1 = t_2 = \infty$). Note that $p_V(\upsilon|\theta) = p_V(\upsilon)$.

The aim is to optimize the mean cost of system operations, where the total cost reflects the sum of failure, fixed, and replacement cost; a single experiment produces an observed cost Q. Note, however, that $Q = Q(\theta, \upsilon)$ is discontinuous in θ whenever t_1 or t_2 is close enough to one of the failure times in the fixed value for υ (the dummy variable for V). So a small change in θ causes a change in the sequence of failures (i.e., a step change in the observed total cost). So, for example, if an element of V has a value 100, then $\lim_{\eta\to 0}\{Q([100+\eta, t_2]^T, V) - Q([100-\eta, t_2]^T, V)\} \neq 0$ due to the inclusion of the failure cost in $Q([100+\eta, t_2]^T, V)$ but not in $Q([100-\eta, t_2]^T, V)$ for any $\eta > 0$ (the indicated limit is as η *decreases* to 0). Hence, Theorem 15.1 does not apply here. As shown in Exercise 15.1, however, a redefinition of V can lead to the continuity condition being satisfied. ❑

We now show how the derivative–integral interchange is used to construct an estimate of $g(\theta) = \partial L/\partial\theta$. Under the conditions of Theorem 15.1 (or one of the equivalent results mentioned above),

$$\begin{aligned} g(\theta) &= \frac{\partial}{\partial\theta}\int_\Lambda Q(\theta,\upsilon)p_V(\upsilon\,|\,\theta)\,d\upsilon \\ &= \int_\Lambda\left[Q(\theta,\upsilon)\frac{\partial p_V(\upsilon\,|\,\theta)}{\partial\theta} + \frac{\partial Q(\theta,\upsilon)}{\partial\theta}p_V(\upsilon\,|\,\theta)\right]d\upsilon, \end{aligned} \tag{15.2}$$

where the last equality follows by the well-known product rule of calculus applied to $\partial[Q(\theta,\upsilon)p_V(\upsilon\,|\,\theta)]/\partial\theta$. The expression above does not lend itself to direct representation as an expectation since the first summand in the integral after the second equality does not represent an integral against a density function. Fortunately, it is easy to convert this part of the integral to an expected value. Assuming that $p_V(\upsilon|\theta) > 0$, simply multiply by $p_V(\upsilon\,|\,\theta)/p_V(\upsilon\,|\,\theta)$ to obtain

$$\begin{aligned} g(\theta) &= \int_\Lambda\left[Q(\theta,\upsilon)p_V(\upsilon\,|\,\theta)^{-1}\frac{\partial p_V(\upsilon\,|\,\theta)}{\partial\theta} + \frac{\partial Q(\theta,\upsilon)}{\partial\theta}\right]p_V(\upsilon\,|\,\theta)\,d\upsilon \\ &= E\left[Q(\theta,V)\frac{\partial\log p_V(V\,|\,\theta)}{\partial\theta} + \frac{\partial Q(\theta,V)}{\partial\theta}\right], \end{aligned} \tag{15.3}$$

where the second line follows by the fundamental formula $\partial\log f(x)/\partial x = f(x)^{-1}\partial f(x)/\partial x$ for a differentiable function $f(x) > 0$.

From (15.3), there is a ready way to produce unbiased estimates of $g(\theta)$ at any θ. In particular, for a sample V, one unbiased gradient estimate is

$$Q(\theta,V)\frac{\partial\log p_V(V\,|\,\theta)}{\partial\theta} + \frac{\partial Q(\theta,V)}{\partial\theta}. \tag{15.4}$$

In practice, the V in (15.4) is associated with one simulation run. In contrast to the gradient-free methods of Sections 14.3 and 14.4 (FDSA and SPSA), detailed knowledge of the inner working of the simulation is required to form the estimate in (15.4). In particular, it must be possible to compute the two indicated gradients. To smooth out noise effects, an obvious variation on (15.4) is to average independent samples (independent V values), with each sample associated with one simulation run. Because this changes nothing fundamental in the discussion below, we focus on the one-simulation gradient estimate in (15.4).

In the spirit of Example 15.1, the example below is an invalid interchange of derivative and expectation, leading to a misapplication of the result (15.4). This is presented as a cautionary illustration on the importance of the regularity conditions for the interchange (such as in Theorem 15.1).

Example 15.2—Incorrect gradient estimate. This example demonstrates how a naïve approach to interchanging the derivative and integral can lead to an inappropriate gradient estimate. Suppose that θ and V are scalars with $Q(\theta, V) = (1 - V)^2$ and $\theta > 0$ a distributional parameter arising via the assumption that $V \sim U(0, \theta)$ (so $\Lambda = (0, \theta)$). Straightforward calculations show that $L(\theta) = 1 - \theta + \theta^2/3$, implying that $g(\theta) = -1 + 2\theta/3$ and $\theta^* = 3/2$.

Let us now show how the conditions of Theorem 15.1 are violated. The density function is $p_V(\upsilon|\theta) = \theta^{-1}I_{\{0<\upsilon<\theta\}}$, where $I_{\{\cdot\}}$ is the indicator function (defined in the section "Frequently Used Notation" near the back of the book). Then one needs to compute $\partial(Q \times p_V)/\partial\theta = Q\partial p_V/\partial\theta$. Following Fu and Hu (1997, pp. 15–16), the product rule for derivatives yields

$$\frac{\partial p_V(\upsilon\,|\,\theta)}{\partial\theta} = \theta^{-1}\delta(\theta-\upsilon) - \theta^{-2}I_{\{0<\upsilon<\theta\}}, \tag{15.5}$$

where the Dirac δ-function is defined in relation to the indicator (step) function by

$$I_{\{\upsilon\geq\theta\}} = \int_{-\infty}^{\upsilon} \delta(y-\theta)\,dy, \tag{15.6}$$

or, equivalently,

$$I_{\{\upsilon<\theta\}} = 1 - I_{\{\upsilon\geq\theta\}} = \int_{\upsilon}^{\infty} \delta(y-\theta)\,dy. \tag{15.7}$$

Note that the δ-function appearing in (15.5) can be thought of, informally, as the derivative resulting from the antiderivative property of calculus applied to the integral in (15.7) (also noting the equivalence of $I_{\{\upsilon<\theta\}}$ and $I_{\{0<\upsilon<\theta\}}$ given the definitions of Λ). In this sense, the δ-function can be thought of as a type of derivative of the indicator function. This is an intuitive—not mathematically rigorous—statement.

Observe that the δ-function in (15.5) implies that the gradient $\partial p_V/\partial\theta$ is unbounded, indicating that $\partial(Q\times p_V)/\partial\theta$ is unbounded. Hence, there exists no $q_1(\upsilon)$ with finite integral that can bound this gradient, a violation of a condition in Theorem 15.1.

Suppose that one ignores the violation of conditions for Theorem 15.1 and plows ahead with interchanging the derivative and integral via (15.2), leading to an estimator of the form (15.4). Because $\partial Q/\partial\theta = 0$, the invalid estimator of $g(\theta)$ at an observed V is then $Q\partial\log p_V/\partial\theta = (Q/p_V)\partial p_V/\partial\theta = (1-V)^2[\theta^{-1}\delta(\theta-V)-\theta^{-2}I_{\{0<V<\theta\}}]/\theta^{-1}$ on Λ. Given that $\delta(\theta - V) = 0$ on Λ, this estimator can be simplified to $-(1-V)^2/\theta = -Q(\theta,V)/\theta$. (This type of result also appears in Fu and Hu, 1997, p. 147, but in a different problem context.) The mean of this invalid gradient estimator is

$$E\left[-\frac{Q(\theta,V)}{\theta}\right] = -\int_{\Lambda}(1-\upsilon)^2\theta^{-2}d\upsilon = \frac{-3+3\theta-\theta^2}{3\theta}. \tag{15.8}$$

Because we are seeking an unbiased gradient estimator, the mean in (15.8) *should* equal $g(\theta) = -1 + 2\theta/3$. But it does not.

As an indication of the severity of the bias in the invalid estimator, let us compare values of $g(\theta)$ with the means from (15.8) at $\theta = 1$, 3/2 (the value θ^*), and 10. We find, respectively, that $g(\theta) = -0.33$, 0, and 5.67 while the corresponding means from (15.8) are −0.33, −0.17, and −2.43. These discrepancies illustrate the danger of a cavalier interchange of derivative and integral. □

The setting above is a broad basis for gradient estimation without regard to the specific search algorithm being used. The estimate in (15.4) can be used in general algorithms requiring an unbiased estimate of the gradient. The remainder of this chapter focuses on important search approaches that make use of the above gradient estimation framework.

15.2 PURE LIKELIHOOD RATIO/SCORE FUNCTION AND PURE INFINITESIMAL PERTURBATION ANALYSIS

As indicated in Section 15.1, there has been a traditional segregation of the LR/SF and IPA approaches. Although we have adopted a general framework that encompasses these two approaches—as manifested in the general form of gradient estimate in (15.4)—let us briefly comment on the relative properties of the traditional "pure" LR/SF and IPA approaches for simulation-based optimization. Recall from Section 15.1 that pure LR/SF corresponds to the setting where $Q(\boldsymbol{\theta}, V) = Q(V)$, and pure IPA is where $p_V(\upsilon|\boldsymbol{\theta}) = p_V(\upsilon)$. So, in the former, the parameter effects must all be captured in the distribution function for V, while in the latter they must all be captured in the observed loss (i.e., the distinction between distributional and structural parameters discussed in Section 15.1). This section discusses gradient estimation per se; the next section discusses the application to search and optimization.

For pure LR/SF, only a simulation output $Q(V)$ and density derivative $\partial \log p_V/\partial\boldsymbol{\theta}$ (derived directly from $\partial p_V/\partial\boldsymbol{\theta}$) are required, while with pure IPA, only the performance derivative $\partial Q/\partial\boldsymbol{\theta}$ is required. The labels LR and SF come from two related concepts. LR comes from the ratio $p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta})/p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta}')$ that appears in the sample path version of the method that is discussed in Section 15.5, with the ratio measuring the relative likelihood of the outcome $V = \boldsymbol{\upsilon}$ under the two values $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$. SF comes from standard statistics terminology, where $\partial \log p_V/\partial\boldsymbol{\theta}$ is the score function set to zero for finding a maximum likelihood estimate. In particular, from (15.4), the LR/SF and IPA gradient estimates at a given $\boldsymbol{\theta}$ are

$$\textbf{LR/SF:}\quad Q(V)\frac{\partial \log p_V(V\,|\,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}} = \frac{Q(V)}{p_V(V\,|\,\boldsymbol{\theta})}\frac{\partial p_V(V\,|\,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}},\quad p_V(V|\boldsymbol{\theta}) > 0, \tag{15.9}$$

$$\textbf{IPA:}\quad \frac{\partial Q(\boldsymbol{\theta}, V)}{\partial\boldsymbol{\theta}}. \tag{15.10}$$

In practice, neither LR/SF nor IPA may provide a framework for reasonable implementation. Pure LR/SF may impose excessive demands in deriving a distribution function that encompasses all of the $\boldsymbol{\theta}$ effects, beginning with the $U(0, 1)$ random variables. On the other hand, pure IPA requires that $Q(\boldsymbol{\theta}, V)$ absorb all of the $\boldsymbol{\theta}$ effects, possibly making the computation of $\partial Q/\partial\boldsymbol{\theta}$ intractable or invalidating the validity of interchange of derivative and integral due to nondifferentiability of Q. Between these extremes, the analyst may have many choices, all resulting in a gradient form of the type in (15.4), with *all* terms present. For example, if the underlying $U(0, 1)$ random variables go through many transformations in creating the ultimate random effects of interest in the simulation, then the analyst may define V as the random variables at any of the levels of transformation. So, given the choice in where the $\boldsymbol{\theta}$ effects are allocated for purposes of the optimization, it is worth considering an allocation that produces a lower variance in the gradient estimate.

Unfortunately, there are no easy general rules pointing toward the lowest variance gradient estimate that has reasonable implementation demands. At the extremes, with the pure LR/SF estimate it may be easier to satisfy the interchange conditions of Theorem 15.1 since the density (or mass) functions tend to be smoother (as a function of $\boldsymbol{\theta}$) than the measured performance Q when Q must capture all of the $\boldsymbol{\theta}$ effects (as we saw in Example 15.1 and Exercise 15.1; see, e.g., Fu and Hu, 1997, p. 146, and Rubinstein and Melamed, 1998, pp. 218–221). LR/SF, however, has the undesirable property that the variance of the gradient estimate generally grows with the number of elements in V (often corresponding to simulation length).

To see that the variance grows in the LR/SF estimate, suppose that V is composed of M independent random effects $\{\mathcal{V}_j\}$ with associated density/mass functions $p_{\mathcal{V}_j}(\upsilon\,|\,\boldsymbol{\theta})$, $j = 1, 2,\ldots, M$ (υ the dummy variable for $\mathcal{V}_j$). Because of the independence among the elements of V, the number of summands in $\log p_V(V|\boldsymbol{\theta})$ is the same as the number of elements (i.e., M). From the fundamental LR/SF assumption that $\partial Q/\partial\boldsymbol{\theta} = \mathbf{0}$ (i.e., Q does not depend on $\boldsymbol{\theta}$), the gradient estimate (15.9) is

$$Q(V)\frac{\partial \log p_V(V\,|\,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}} = Q(V)\sum_{j=1}^{M}\frac{\partial \log p_{\mathcal{V}_j}(\mathcal{V}_j\,|\,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}. \qquad (15.11)$$

Hence, the gradient estimate tends to have a variance of magnitude $O(M)$ (the "tends to" qualifier is due to the dependence introduced in the M-term sum through the $Q(V)$ multiplier).

A number of techniques have been introduced to reduce the variance of pure LR/SF. One such technique exploits regenerative structure (Section 14.2). With regeneration, one can average over the regenerative cycles as a means of variance reduction (e.g., Glynn, 1987, 1990; L'Ecuyer et al., 1994). In particular, the above sum containing M components can be reduced to a sum of a smaller number of components, each representing the result of one regenerative cycle. So, the sum in (15.11) is replaced by an analogous averaging over the number of regenerative cycles.

At the other extreme, pure IPA does not generally suffer from the increasing variance problem mentioned above for pure LR/SF because $\partial Q/\partial\boldsymbol{\theta}$ does not generally grow with the number of components in V and because $\partial \log p_V/\partial\boldsymbol{\theta} = \mathbf{0}$. However, the complexity of Q as a function of $\boldsymbol{\theta}$ is a frequent barrier to implementation. Sometimes, a straightforward definition of a gradient estimate may yield a biased estimate, as we saw in Example 15.2 (other examples are in Cao, 1985; L'Ecuyer and Glynn, 1994; and Fu and Hu, 1997, Chap. 4). However, special structure can sometimes be exploited to create an unbiased estimate.

A large literature has evolved for the development of techniques for computing $\partial Q/\partial\boldsymbol{\theta}$ in a simulation-based optimization context. The books by Glasserman (1991a), Fu and Hu (1997), and Cassandras and Lafortune (1999, especially Chap. 11) and survey papers by Suri (1989) and Ho and Cassandras (1997) are some of the references that summarize many of the available techniques. These techniques are designed to exploit special structure in the simulation to allow for the practical calculation of the gradient $\partial Q/\partial\boldsymbol{\theta}$ and to provide the necessary smoothness in Q to allow for the required interchange of derivative and integral (i.e., to yield an unbiased estimator). In cases where such techniques are feasible, IPA—or close cousins termed *smoothed PA*, *rare PA*, and so on—frequently provide the lowest variance gradient estimate (e.g., L'Ecuyer et al., 1994; Rubinstein and Melamed, 1998, pp. 221–223).

The example below contrasts pure LR/SF and pure IPA on a simple problem of gradient estimation (not necessarily in an optimization context). This example shows how different gradient estimators—with different properties—can result from the seemingly arbitrary decision to allocate $\boldsymbol{\theta}$ totally to the distribution function or totally to the performance measure.

Example 15.3—IPA and LR/SF estimates for exponentially distributed random variable. Let Z be an exponential random variable with mean θ. That is, $p_Z(z\,|\,\theta) = e^{-z/\theta}/\theta$ for $z \geq 0$. Define $L = E(Z) = \theta$. We can derive both LR/SF and IPA estimates of the gradient of L for illustration. (Note that in this trivial case, the gradient of L can be computed as $\partial L/\partial\theta = 1$.)

To compute the LR/SF gradient estimate, we define $V = Z$ and Q as the identity function, that is, $Q(\theta, V) = V$. Because θ does not enter Q, θ is only a distributional parameter. Therefore, only the first term in equation (15.4) contributes to the derivative estimate (as in (15.9)). Using $p_Z(Z|\theta) = p_V(V|\theta)$, the LR/SF gradient estimate can be derived as

$$V\frac{\partial \log p_V(V\,|\,\theta)}{\partial\theta} = \frac{V}{\theta}\left(\frac{V}{\theta}-1\right).$$

To apply the IPA technique, we change the definition of V. Let $V \sim U(0, 1)$ and $Z = -\theta \log V$. As above, Z is exponentially distributed with mean θ. Then, $L(\theta) = E(Z) = E[Q(\theta, V)] = -E[\theta \log V]$. With these new definitions, θ only appears as a structural parameter (i.e., the distribution of V is independent of θ). Therefore, the IPA gradient estimate can be derived from the second term in (15.4) (as in (15.10)):

$$\frac{\partial Q(\theta, V)}{\partial\theta} = -\log V .$$

We can easily check that both LR/SF and IPA gradient estimates are unbiased. It can also be shown that the IPA estimate has smaller variance than the LR/SF estimate (Exercise 15.2). ❑

15.3 GRADIENT ESTIMATION METHODS IN ROOT-FINDING STOCHASTIC APPROXIMATION: THE HYBRID LR/SF AND IPA SETTING

Let us return to the general setting of Section 15.1, where the gradient estimation form is generally a hybrid LR/SF and IPA form. Recall the basic root-finding SA recursions in Section 4.1:

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k Y_k(\hat{\theta}_k) \quad \text{(unconstrained)},$$

$$\hat{\theta}_{k+1} = \Psi_{\Theta}[\hat{\theta}_k - a_k Y_k(\hat{\theta}_k)] \quad \text{(constrained)},$$

where $Y_k(\theta)$ represents a (noisy) measurement of the function for which a zero is to be found. Under appropriate conditions (e.g., Section 4.3), it is known that $\hat{\theta}_k$ converges to an optimum $\theta^* = \arg\min_{\theta \in \Theta} L(\theta)$ as $k \to \infty$. The point θ^* may be unique or may be one of multiple points in a *set* of equivalent minima Θ^*, although, as we have done previously for ease of discussion, we usually consider θ^* as a unique minimum. For simulation-based optimization, the stochastic gradient setting of Chapter 5 applies. Hence, $Y_k(\theta)$ represents a measurement of $g(\theta) = \partial L/\partial\theta$. (As discussed in Section 8.4, if there are multiple local minima to $L(\theta)$, the SA recursions above produce a *global* optimum in the limit if a Monte Carlo noise term is injected on the right-hand side.)

Recall that a basic philosophy in SA is to perform a type of *across-iteration* averaging to smooth out noise effects versus expending a large amount of resources in getting accurate estimates for $g(\theta)$ at each iteration. The implication relative to (15.4) is that the amount of *per-iteration* averaging of $Y_k(\theta)$ values (corresponding to multiple samples of V) should be small. Clearly, in a dynamic system where the simulation environment is changing rapidly (possibly affecting the optimal value θ^*), there are advantages to avoiding a large amount of averaging at a given θ since the simulations being averaged may not represent identically distributed samples.

The bottom line relative to a root-finding SA implementation is that (15.4) is usually implemented directly as it is presented, that is, without averaging from multiple values of V at each value of θ. So the unbiased estimate of $g(\theta)$ for use in the SA recursions above is

$$Y_k(\hat{\theta}_k) = Q(\hat{\theta}_k, V_k)\frac{\partial \log p_V(V_k \mid \theta)}{\partial\theta}\bigg|_{\theta=\hat{\theta}_k} + \frac{\partial Q(\theta, V_k)}{\partial\theta}\bigg|_{\theta=\hat{\theta}_k}, \tag{15.12}$$

where V_k represents the realization of V for use in the kth iteration, generated from $p_V(\upsilon \mid \theta = \hat{\theta}_k)$. Note that *one* simulation run—corresponding to one realization of V—can be used to produce this estimate at each value of θ. (A closely related approach is summarized in Section 15.4, where one realization of V can be repeatedly used to produce unbiased gradient estimates for *multiple* values of θ.)

Unlike the FDSA and SPSA methods, one must be able to compute the gradients $\partial \log p_V(V_k \mid \theta)/\partial\theta$ and $\partial Q(\theta, V_k)/\partial\theta$ to obtain this estimate (with FDSA and SPSA, only values of $Q(\theta, V)$ are needed). As discussed in Sections 6.5 and 7.4, however, the stochastic gradient algorithm buys an increased rate of convergence relative to FDSA or SPSA, with the optimal stochastic rate increasing from $O(1/k^{1/3})$ to $O(1/\sqrt{k})$. We also saw in Section 14.4 that common random numbers in FDSA and SPSA can increase the rate from

$O(1/k^{1/3})$ to $O(1/\sqrt{k})$, although the relative mean-squared errors of root-finding SA and CRN-based FDSA or SPSA at the common rate of convergence do not appear to have been analyzed.

The two-part example below illustrates the method above for computing the gradient estimate together with the application in an SA-based optimization process.

Example 15.4—Laboratory experimental response (part 1—general setting). Consider a laboratory setting where a specimen is exposed to a sequence $\{V_k\}$ of randomly generated independent, identically distributed (i.i.d.) "on–off" stimuli with "on" probability λ (and, of course, "off" probability of $1 - \lambda$). After each exposure, the specimen responds in some way to reveal important information to an experimenter. The aim is to design an experiment such that the specimen response is maximized. Suppose that the function measuring the negative of this response is $Q(\boldsymbol{\theta}, V_k) = Q(\lambda, \beta, V_k)$, where β is a design parameter (negative response is used here so that we have a minimization problem). Further, suppose that $Q(\lambda, \beta, V_k)$ is differentiable in $\boldsymbol{\theta} = [\lambda, \beta]^T$ and that the derivative can be computed. The aim is to pick $\boldsymbol{\theta}$ in a way to maximize the expected response. Note that this problem does not have a clear demarcation between distributional parameters (λ) and structural parameters (λ, β) since Q depends on both λ and β.

Each stimulus will be associated with one iteration of the stochastic gradient algorithm. The Bernoulli probability mass function can be expressed as $p_V(\upsilon|\boldsymbol{\theta}) = \lambda^{\upsilon}(1 - \lambda)^{1-\upsilon}$, where $\upsilon = 0$ or 1. This function is clearly differentiable in λ as required in Theorem 15.1, yielding the following contribution to the unbiased gradient estimate in (15.12):

$$\frac{\partial \log p_V(\upsilon|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \begin{bmatrix} \dfrac{\lambda - \upsilon}{(\lambda - 1)\lambda} \\ 0 \end{bmatrix}.$$

Assume that Q is such that the conditions of Theorem 15.1 are satisfied (see Example 15.5 below). Then, from (15.12), a valid input to the root-finding algorithm at the kth iteration is

$$\boldsymbol{Y}_k(\hat{\boldsymbol{\theta}}_k) = \begin{bmatrix} Q(\hat{\boldsymbol{\theta}}_k, V_k)\dfrac{\hat{\lambda}_k - V_k}{(\hat{\lambda}_k - 1)\hat{\lambda}_k} + Q'_\lambda(\hat{\boldsymbol{\theta}}_k, V_k) \\ Q'_\beta(\hat{\boldsymbol{\theta}}_k, V_k) \end{bmatrix}, \tag{15.13}$$

where $\hat{\boldsymbol{\theta}}_k = [\hat{\lambda}_k, \hat{\beta}_k]^T$ and $Q'_x(\boldsymbol{\theta}, V) = \partial Q(\boldsymbol{\theta}, V)/\partial x$, $x = \lambda$ or β. (See also Exercise 15.3.) □

Example 15.5—Laboratory experimental response (part 2—numerical results). We continue the experimental response example with a simple case where an analytical solution is available. This example is intended to convey concepts and provide a known solution to compare to the stochastic gradient (hybrid LR/SF and IPA) solution. In practice, of course, simulation-based optimization is oriented to cases where such simple solutions do not exist. Suppose that the simulation output produces a (negative) response function as follows:

$$Q(\boldsymbol{\theta}, V) = \beta^2 + (1-\lambda)(\beta - V),$$

where $V = 1$ represents the "on" event (and, of course, $V = 0$ represents the "off" event) and $-1 < \beta < 1$.

Given that this response function depends on both λ and β (i.e., there is no strict segregation of the distributional and structural parameters), this is a setting where the response of the system to one stimulus is dependent on the likelihood of other stimuli (i.e., dependent on λ). So there will be some cases where the system tends to respond strongly to a stimulus $V = 1$ when the probability (λ) of other stimuli is high (say, a drug response increases with the likelihood of greater past or future dosages, perhaps through a placebo effect) and other cases where the system responds weakly in the presence of a high probability of other stimuli (say, as when a drug loses effectiveness with repeated usage).

With the above response function, we obtain the following closed-form expression for L:

$$L(\boldsymbol{\theta}) = \sum_{\upsilon=0}^{1} Q(\boldsymbol{\theta}, \upsilon)\lambda^{\upsilon}(1-\lambda)^{1-\upsilon} = -\lambda + \lambda^2 + \beta - \lambda\beta + \beta^2.$$

Setting the gradient $\boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{0}$ and examining the Hessian matrix yields a unique minimum at $\boldsymbol{\theta}^* = [1/3, -1/3]^T$, having $L(\boldsymbol{\theta}^*) = -1/3$.

Before proceeding with the implementation for optimization, let us note that the full power of Theorem 15.1 is unnecessary in this problem. The expectation "integral" in this case is simply a summation over the two possible outcomes for V. Hence, the interchange of derivative and "integral" (i.e., summation) is trivially true because the required derivatives exist.

Figure 15.1 shows the result of one typical realization of 1000 iterations of the stochastic gradient algorithm as applied with the Q above. Each iteration of the algorithm depends on one simulation producing an output Q. Based on the principles of Section 4.4, the gain sequence $a_k = 0.1/(100+k)^{0.501}$ is used. From an initial condition of $\hat{\boldsymbol{\theta}}_0 = [1/2, 1/2]^T$, the algorithm moves to $\hat{\boldsymbol{\theta}}_{1000} = [0.33028, -0.32361]^T$. This yielded a final loss value of -0.33320 and a final $\boldsymbol{\theta}$ estimate having Euclidean distance of 0.0102 to the solution. As an indication of the "typical" nature of this run, a separate Monte Carlo study based on 30

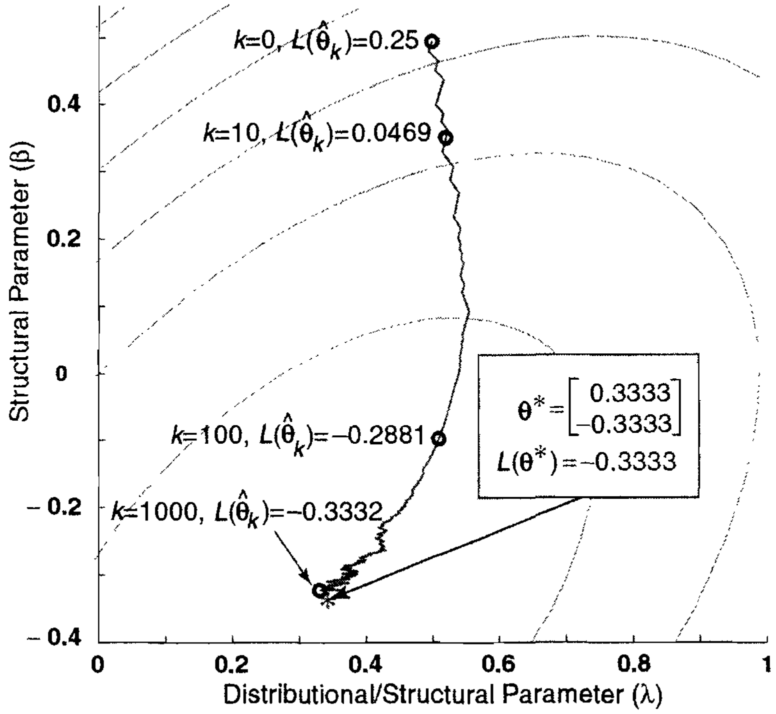

Figure 15.1. Search path for typical realization of the stochastic gradient algorithm in experimental response model.

independent replications of the root-finding algorithm was conducted, yielding final loss values ranging from −0.33069 to −0.33333 and final distances to the solution of 0.0612 to 0.0058.

Figure 15.1 shows that the distance from the final estimate to the optimal point θ^* is visually small and that the loss value has effectively converged (note that the noise e in the gradient measurements is relatively small in this problem). The θ estimate being relatively farther from θ^* than the corresponding $L(\theta)$ is from $L(\theta^*)$ is a typical pattern in simulation-based (and other) optimization given the flatness of loss functions near θ^*.

As an illustration of the degrading effects of the noise in the measurements of $g(\theta)$, let us consider a noise-free steepest descent algorithm based on direct values of $g(\theta)$ (as in Section 1.4). Using the same a_k as above, the steepest descent algorithm was able to get within a distance of 0.0069 of the solution, which is significantly better than the 0.0102 distance in the typical run of Figure 15.1, but not as good as the minimum distance of 0.0058 in the 30 noisy test cases mentioned above. (However, an a_k tuned to the noise-free case leads to better results for steepest descent, significantly better than the best noisy run.) Of course, this noise-free solution is not available in practical simulation-based optimization unless one does a very large (and costly!) amount of averaging of the noisy measurements at each iteration. (A different form of noise-free optimization is considered in the next section.) □

15.4 SAMPLE PATH OPTIMIZATION

In general, the main cost of each iteration of the stochastic gradient algorithm is the simulation needed to produce the gradient estimate. Hence, n iterations require n simulations. The *sample path* (sometimes called *stochastic counterpart*) method of optimization is based on a different principle. Here, structure in the problem is exploited to allow for the *multiple* gradient estimates (say, n) needed in the search process to be computed from a relatively small *single* set of simulation runs. That is, a surrogate loss function is created from a set of V values, say V_0, $V_1,\ldots, V_{N-1}$, representing N simulation runs. (We are using indices $0, 1,\ldots, N-1$, rather than $1, 2,\ldots, N$, to be consistent with the indexing in the stochastic gradient form of Section 15.3, where the first V value is V_0.) These N runs are then averaged in an appropriate way, leading to an approximation to the true loss function. The n gradients needed in the search process can be computed based on this approximate loss function serving as a proxy for the unavailable true loss $L(\boldsymbol{\theta})$. This is tantamount to reusing the same set of N realizations of V (the sample path) for multiple gradient estimates. Note that, in general, there is no relationship between n and N.

Because the sample path method is based on reusing the fixed set of simulation runs, one may effectively treat the resulting optimization problem as a *deterministic* optimization problem, using standard deterministic nonlinear programming techniques to solve for the optimal $\boldsymbol{\theta}$. This reuse of the same sequence of V values for multiple gradient calculations is a means for squeezing more information from the limited number of simulation runs. The nonlinear programming aspect is especially useful in problems with complicated constraints because there is much machinery in deterministic search and optimization for handling constraints that is not readily available in stochastic optimization (e.g., Bazaraa et al., 1993; Polak, 1997).

An additional significant benefit of the sample path method follows if the simulation is very costly to run. Because the number of simulations is fixed (N), a large number of iterations of the deterministic optimization method can be conducted at nominal cost beyond the cost of the N simulations. The nominal cost here reflects the cost of the gradient calculation based on the simulation output and possible other costs of the algorithm if an approach more complex than steepest descent (Section 1.4) is being used.

In the sample path method, rather than optimize $L(\boldsymbol{\theta})$ in (15.1), one optimizes a function $\overline{L}_N(\boldsymbol{\theta})$, representing an average of N independent simulation runs. (In particular, it is shown in (15.15) and (15.16) below how knowledge of the functional relationships, $Q(\boldsymbol{\theta}, V)$ and $p_V(\upsilon|\boldsymbol{\theta})$, is used to form $\overline{L}_N(\boldsymbol{\theta})$.) So, although there are the potential implementation and computational benefits mentioned above, a price to be paid in the sample path method is that the limiting point for the $\boldsymbol{\theta}$ estimate depends on the specific sequence of N random processes, V_0, $V_1, \ldots, V_{N-1}$. That is, under appropriate conditions, $\hat{\boldsymbol{\theta}}_n$ computed in a sample path implementation of a search algorithm converges as $n \to \infty$ to

$$\boldsymbol{\theta}_N^* \equiv \arg\min_{\boldsymbol{\theta}\in\Theta} \bar{L}_N(\boldsymbol{\theta})$$

where $\boldsymbol{\theta}_N^* = \boldsymbol{\theta}_N^*(V_0, V_1, \ldots, V_{N-1})$ and $\boldsymbol{\theta}_N^* \neq \boldsymbol{\theta}^*$ in general. (For the discussion here, $\boldsymbol{\theta}_N^*$ is assumed to be unique.) So, a central point is to establish conditions such that $\boldsymbol{\theta}_N^*$ is very close to $\boldsymbol{\theta}^*$ $(= \arg\min_{\boldsymbol{\theta}\in\Theta} L(\boldsymbol{\theta})$, of course). This implies that for sufficiently large N, the optimization of $\bar{L}_N(\boldsymbol{\theta})$ yields a solution that is close to the optimization of the true loss function $L(\boldsymbol{\theta})$.

One might suspect that $\boldsymbol{\theta}_N^*$ and $\boldsymbol{\theta}^*$ will be close for large N since, by the laws of large numbers, $\bar{L}_N(\boldsymbol{\theta}) \to L(\boldsymbol{\theta})$ for each $\boldsymbol{\theta}$ in some stochastic sense when $N \to \infty$. But this is not automatic. For example, well-known counterexamples exist where having a function converge does not imply that the derivative of that function will converge (see, e.g., Rudin, 1976, pp. 152–154). Specifically, $\bar{L}_N(\boldsymbol{\theta}) \to L(\boldsymbol{\theta})$ does not imply $\partial\bar{L}_N(\boldsymbol{\theta})/\partial\boldsymbol{\theta} \to \partial L(\boldsymbol{\theta})/\partial\boldsymbol{\theta}$; so, a zero of $\partial\bar{L}_N(\boldsymbol{\theta})/\partial\boldsymbol{\theta}$ may not be close to a zero of $\partial L(\boldsymbol{\theta})/\partial\boldsymbol{\theta}$. Hence, in general, one cannot assume that $\boldsymbol{\theta}_N^*$ and $\boldsymbol{\theta}^*$ are approximately equal, even for large N. Shapiro (1991, 1996), Robinson (1996), and Rubinstein and Melamed (1998, p. 152), among others, discuss conditions under which $\boldsymbol{\theta}_N^*$ *does* converge to $\boldsymbol{\theta}^*$. Let us present one of those results here. As in Theorem 15.1, $p_V(\upsilon\,|\,\boldsymbol{\theta})$ may represent either a probability density or a mass function; integrals are to be interpreted as sums in the discrete (mass function) case. The conditions here bear some similarity to the conditions of Theorem 15.1, but there are some nontrivial differences as well.

Theorem 15.2 (Rubinstein and Melamed, 1998, p. 152). Suppose that the search space $\Theta \subset \mathbb{R}^p$ is a compact (i.e., closed and bounded) set known to contain $\boldsymbol{\theta}^*$. Assume that for almost all V (recall definition of "almost all" in Subsection C.2.1, Appendix C), $p_V(V|\boldsymbol{\theta})$ is continuous on Θ and that there exists a nonnegative function $q_0(\upsilon)$ such that

$$|Q(\boldsymbol{\theta},\upsilon)p_V(\upsilon\,|\,\boldsymbol{\theta})| \leq q_0(\upsilon) \text{ for all } (\boldsymbol{\theta},\upsilon) \in \Theta\times\Lambda$$

where $\int_\Lambda q_0(\upsilon)\,d\upsilon < \infty$. Then $\boldsymbol{\theta}_N^* \to \boldsymbol{\theta}^*$ a.s. as $N \to \infty$.

Shapiro (1991) and Rubinstein and Melamed (1998, p. 152) also present asymptotic distribution results for $\boldsymbol{\theta}_N^*$, showing that $\sqrt{N}(\boldsymbol{\theta}_N^* - \boldsymbol{\theta}^*)$ is asymptotically normally distributed under conditions stronger than those in

Theorem 15.2. These results can be used, as in other applications of asymptotic distributions in this book, to assess the approximate error of an estimator ($\boldsymbol{\theta}_N^*$ in this case). The error is proportional in a stochastic sense to $1/\sqrt{N}$. Based on a large number of studies with typical discrete-event simulations, Rubinstein and Melamed (1998, pp. 151–152) state that when $\overline{L}_N(\boldsymbol{\theta})$ is a convex function (Appendix A), values of N on the order of 1000 or more are typically needed to ensure that $\boldsymbol{\theta}_N^*$ is an acceptable approximation of $\boldsymbol{\theta}^*$.

More discussion on Theorem 15.2 and the associated asymptotic normality, including more detail on the implications of the constraints embodied in Θ, are given in Shapiro (1991, 1996). Shapiro (1996) includes the study of regenerative processes (Section 14.2) as a special case. He also shows how asymptotic normality results pertaining to the error in using $\overline{L}_N(\boldsymbol{\theta})$ as an approximation to $L(\boldsymbol{\theta})$ can be applied in making statistical inference about the solution $\boldsymbol{\theta}_N^*$. Analogous to what has been done in other contexts in the preceding chapters, Shapiro (1991, 1996) uses the asymptotic normality to assess the efficiency of the sample path method relative to SA-based stochastic gradient method.

Although the sample path method converts the problem to a deterministic process, the LR/SF and IPA forms in Sections 15.2 and 15.3 are still relevant for gradient calculation. In particular, one must still make a choice about the allocation of the elements in $\boldsymbol{\theta}$ to being distributional parameters, structural parameters, or a hybrid form. That choice affects whether (15.4) or one of the "pure" forms discussed in Section 15.2 may be used to construct the gradient estimate.

In some sense, however, Theorem 15.2 begs the fundamental question: How close will the estimate $\hat{\boldsymbol{\theta}}_n$ be to $\boldsymbol{\theta}^*$ when a sample path method is used? This question cannot be answered without knowledge of the convergence properties of the specific nonlinear programming algorithm being used to minimize $\overline{L}_N(\boldsymbol{\theta})$. Section 1.4 discussed the convergence properties of the steepest descent and Newton–Raphson algorithms. Bazaraa et al. (1993), Polak (1997), and many other texts discuss these properties in more detail, including the convergence properties of other popular deterministic search algorithms not discussed in Section 1.4. To assess the closeness of $\hat{\boldsymbol{\theta}}_n$ to $\boldsymbol{\theta}^*$, one also needs insight into the error in using $\boldsymbol{\theta}_N^*$ to approximate $\boldsymbol{\theta}^*$. As mentioned above, an asymptotic normality result exists that may provide an approximate error bound. Combining the two sources of error, the total error can then be bounded by the triangle inequality,

$$\left\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*\right\| \leq \left\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_N^*\right\| + \left\|\boldsymbol{\theta}_N^* - \boldsymbol{\theta}^*\right\|, \tag{15.14}$$

where the first error on the right-hand side reflects the inaccuracy in the nonlinear program and the second error is due to substitution of the sample path minimum for the true minimum. Let us present an application of this result.

Example 15.6—Error analysis for sample path optimization. Suppose that $p = 5$. From the properties of the deterministic search algorithm, suppose that each of the elements of $\hat{\theta}_n$ is known to be within 0.2 unit of the corresponding elements in the optimum θ_N^*. Further, assume that from asymptotic theory, the approximate distribution of θ_N^* is $N(\theta^*, 0.05I_5)$ (see the discussion above following Theorem 15.2). Suppose that we wish to identify a distance that it is known with 90 percent certainty to be at least as large as the true distance between $\hat{\theta}_n$ and θ^*.

From (15.14), one knows that the true distance can be bounded by the sum of the distance between $\hat{\theta}_n$ and θ_N^* and the distance between θ_N^* and θ^*. By the above-mentioned assumption on the deterministic search algorithm, the distance between $\hat{\theta}_n$ and θ_N^* is known to be no greater than $\sqrt{5\times 0.2^2} = 0.45$. From the asymptotic normality for θ_N^*, it is known that $(\theta_N^* - \theta^*)^T(\theta_N^* - \theta^*)/0.05$ is approximately chi-squared distributed with five degrees of freedom (the 0.05 term is a *variance*, not a standard deviation). The 90 percent critical value from a chi-squared distribution table is 9.25, indicating that the distance between θ_N^* and θ^* is known with 90 percent certainty to be no greater than $\sqrt{0.05\times 9.25} = 0.68$. Hence, from (15.14), and subject to the asymptotic distribution of θ_N^* being a credible representation of the actual finite sample distribution, one can be 90 percent certain that the distance between $\hat{\theta}_n$ and θ^* is no more than $0.45 + 0.68 = 1.13$. □

A fundamental issue in sample path optimization is the determination of the distribution for generating the fixed sample $V_0, V_1, \ldots, V_{N-1}$ that is used throughout the search process. Clearly, the user must be concrete and specific in this choice since Monte Carlo random variables cannot be generated without an exact distribution, including full specification of the distribution's parameters. In a simulation setting, this is equivalent to choosing the distributional form and associated parameter values for all of the random effects being generated inside the simulation. It seems, however, that a potential problem arises because the distribution for the V_i typically depends on the very parameters θ being estimated during the search process. That is, it seems that a new set of V_i must be generated for each new value of θ during the search process. This would defeat the very principle of sample path optimization! Fortunately, there is a ready way around this problem, as we now describe.

Let us now show how knowledge of the functional relationships, $Q(\theta, V)$ and $p_V(\upsilon|\theta)$, is used to form $\bar{L}_N(\theta)$ via an averaging scheme. Suppose that there exists a density function $q_V(\upsilon)$ such that for every υ where $q_V(\upsilon) = 0$, then $p_V(\upsilon|\theta) = 0$ for all possible values of $\theta \in \Theta$. It is sometimes said that $q_V(\upsilon)$ *dominates* $p_V(\upsilon|\theta)$ (more formally, that the associated *distribution*—rather than density/mass—function dominates the distribution function associated with $p_V(\upsilon|\theta)$). Then, the fundamental definition for the loss function, (15.1), can be expressed as

$$\begin{aligned}
L(\theta) &= \int_{\Lambda} Q(\theta,\upsilon)p_V(\upsilon\,|\,\theta)\,d\upsilon \\
&= \int_{\Lambda} Q(\theta,\upsilon)\frac{p_V(\upsilon\,|\,\theta)}{q_V(\upsilon)}q_V(\upsilon)\,d\upsilon \\
&= E_{q_V}\left[Q(\theta,V)\frac{p_V(V\,|\,\theta)}{q_V(V)}\right],
\end{aligned} \tag{15.15}$$

where $E_{q_V}[\cdot]$ denotes the expectation based on $q_V(\upsilon)$.[2] An important special case is where $q_V(\upsilon) = p_V(\upsilon|\theta')$ for some fixed value θ'. The significance of (15.15) is quite profound. Namely, when $q_V(\upsilon) = p_V(\upsilon|\theta')$, one can generate sample values of V from the distribution based on the *fixed* θ' to obtain an unbiased estimate of the loss function at *any* value of θ. We refer to θ' as the *reference value* of θ.

In particular, letting $V_0, V_1, \ldots, V_{N-1}$ be a sequence of N values of V generated according to $q_V(\upsilon) = p_V(\upsilon|\theta')$, then

$$\bar{L}_N(\theta) \equiv \frac{1}{N}\sum_{i=0}^{N-1} Q(\theta,V_i)\frac{p_V(V_i|\theta)}{p_V(V_i|\theta')} \tag{15.16}$$

is an unbiased estimate of $L(\theta)$. For a fixed reference value θ', (15.16) is viewed solely as a function of θ, as is the true loss function $L(\theta)$. Further, from (15.15), the famous weak or strong laws of large numbers (Section C.2 in Appendix C here, or Laha and Rohatgi, 1979, Chap. 2) state that $\bar{L}_N(\theta)$ converges in probability (pr.) or almost surely (a.s.) to $L(\theta)$ for all θ. Let us illustrate the sample path method in a simple problem involving scalar Bernoulli-distributed V. Although much simpler than most realistic simulation-based problems, this example depicts the essential properties of the sample path method.

Example 15.7—Sample path gradient estimate for binary-outcome problem. Suppose V is a binary random variable such that $V = 1$ with probability $0 < \theta < 1$ and $V = 0$ with probability $1 - \theta$. Let $Q(\theta, V) = V + b/\theta$ where $0 < b < 1$. This leads to $L(\theta) = \theta + b/\theta$, from which it is easily seen that the unique minimizing

[2] The discussion here represents a special case of a broader measure-theoretic framework of probability, where instead of a density or mass function for V, a general probability measure is used (L'Ecuyer, 1990a). (These results require a background beyond the prerequisites for this book.) Here the primary measure of interest for V (based on θ) is said to be *absolutely continuous* with respect to a dominating measure based on $q_V(\upsilon)$. Further, the ratio $p_V(\upsilon\,|\,\theta)/q_V(\upsilon)$ appearing in (15.15) represents the *Radon–Nikodym derivative* of the primary measure with respect to the dominating measure. This more general framework allows for some elegant convergence results and is potentially useful if $p_V(\upsilon|\theta)$ and $q_V(\upsilon)$ do not represent "nice" density or mass functions.

point is $\theta^* = \sqrt{b}$. Let us now construct a sample path gradient estimate and compare the zero of that function with the known value of θ^*. The mass function is $p_V(\upsilon|\theta) = \theta^{\upsilon}(1-\theta)^{1-\upsilon}$ where $\upsilon = 0$ or 1. From (15.16),

$$\frac{\partial \bar{L}_N(\theta)}{\partial\theta} = \frac{1}{N}\sum_{i=0}^{N-1}\left[V_i \frac{p_V(V_i\,|\,\theta)}{p_V(V_i\,|\,\theta')}\times\frac{V_i-\theta}{\theta(1-\theta)}\right] - \frac{b}{\theta^2}. \tag{15.17}$$

When $V_i = 0$, then $[\cdot] = 0$ in (15.17); when $V_i = 1$, then $[\cdot] = 1/\theta'$. Let us take $\theta' = 1/2$ and $b = 0.01$.

For $N = 5$ and 25, we generate the following random sequences for $\{V_i\}$ from the same seed:

$$N = 5:\ \{0, 0, 0, 1, 0\},$$

$$N = 25:\ \{0, 0, 0, 1, 0;\ 1, 1, 0, 1, 0;\ 1, 0, 1, 0, 0;\ 0, 1, 0, 1, 1;\ 0, 1, 0, 0, 1\}.$$

From (15.17), the calculations for $N = 5$ and 25, respectively, are

$$\frac{\partial \bar{L}_N(\theta)}{\partial\theta} = \frac{1}{5}\left[4\times 0 + 1\times\left(\tfrac{1}{2}\right)^{-1}\right] - \frac{0.01}{\theta^2} = 0 \Rightarrow \theta_N^* = \sqrt{\frac{0.01}{0.400}} = 0.158,$$

$$\frac{\partial \bar{L}_N(\theta)}{\partial\theta} = \frac{1}{25}\left[14\times 0 + 11\times\left(\tfrac{1}{2}\right)^{-1}\right] - \frac{0.01}{\theta^2} = 0 \Rightarrow \theta_N^* = \sqrt{\frac{0.01}{0.880}} = 0.107.$$

(These solutions are global minimums of $\bar{L}_N(\theta)$ since $\partial^2\bar{L}_N(\theta)/\partial\theta^2 > 0$ for all θ.) The above solutions compare with $\theta^* = \sqrt{0.01} = 0.10$. As expected, the difference between θ_N^* and θ^* decreases when N is increased from 5 to 25. ❑

Let us return to an issue mentioned at the beginning of this section: How does the efficiency of the sample path method compare with the classical stochastic gradient (i.e., root-finding SA) approach? In seeking a *general* answer to this question, we are, of course, forced into a theoretical (versus numerical) framework. But as we have seen repeatedly in this book in similar contexts, this question is unanswerable in finite samples due to a lack of finite-sample theory. However, asymptotically it is possible to partially answer this question. Suppose that the true sample path minimum $\boldsymbol{\theta}_N^*$ is available (i.e., we are assuming that the deterministic search algorithm has converged perfectly to the minimum). How does the accuracy of $\boldsymbol{\theta}_N^*$ compare to the solution resulting from the N simulations runs used in N iterations of the stochastic gradient algorithm?

Shapiro (1996) shows that, asymptotically, both the sample path and stochastic gradient (root-finding SA) with optimal gains yield statistically equivalent solutions for the same amount of input information in the case of *unconstrained* optimization ($\Theta = \mathbb{R}^p$). As discussed in Subsection 4.5.2, the optimal gain for root-finding SA is the matrix gain $\boldsymbol{a}_k = \boldsymbol{H}(\boldsymbol{\theta}^*)^{-1}/(k+1)$, $k \geq 0$, which, of course, is unavailable in practice ($\boldsymbol{H}$ is the Hessian matrix of the loss function). Practical methods of achieving the asymptotic rate of convergence associated with the optimal gain are the iterate averaging method (Subsection 4.5.3) and the second-order stochastic gradient (2SG) implementation of the adaptive SPSA approach (Section 7.8). (Recall, however, that the iterate averaging method may not yield improved performance in finite samples and that the 2SG implementation requires *three* simulations per iteration as long as the Hessian matrix is being estimated.) Using a simple quadratic loss function, the example below shows that sometimes the sample path and stochastic gradient methods are equivalent in finite-sample problems.

Example 15.8—Equivalence between sample path and stochastic gradient methods. Consider $Q(\boldsymbol{\theta}, \boldsymbol{V}) = \boldsymbol{\theta}^T\boldsymbol{B}\boldsymbol{\theta}/2 + \boldsymbol{\theta}^T\boldsymbol{V}$, where $\boldsymbol{B}$ is positive definite and $\boldsymbol{V}$ is p-dimensional. Then the sample path loss is $\bar{L}_N(\boldsymbol{\theta}) = \boldsymbol{\theta}^T\boldsymbol{B}\boldsymbol{\theta}/2 + \boldsymbol{\theta}^T\bar{\boldsymbol{V}}_N$, where $\bar{\boldsymbol{V}}_N = N^{-1}\sum_{i=0}^{N-1}\boldsymbol{V}_i$. It follows easily that $\boldsymbol{\theta}_N^* = -\boldsymbol{B}^{-1}\bar{\boldsymbol{V}}_N$. Relative to the root-finding SA approach, $\boldsymbol{H}(\boldsymbol{\theta}^*) = \boldsymbol{B}$, indicating that $\boldsymbol{a}_k = \boldsymbol{B}^{-1}/(k+1)$. The root-finding algorithm then has the form

$$\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k - \frac{\boldsymbol{B}^{-1}}{k+1}(\boldsymbol{B}\hat{\boldsymbol{\theta}}_k + \boldsymbol{V}_k),\ k \geq 0,$$

which can be solved in closed form as $\hat{\boldsymbol{\theta}}_N = -\boldsymbol{B}^{-1}\bar{\boldsymbol{V}}_N$ (Exercise 15.11). Hence $\hat{\boldsymbol{\theta}}_N = \boldsymbol{\theta}_N^*$ (both using N simulations). □

In constrained cases ($\Theta \neq \mathbb{R}^p$), it is no longer true that the stochastic gradient and sample path methods are asymptotically equivalent. Shapiro (1996) presents examples showing that the sample path method has a faster rate of convergence in at least some constrained problems, but it appears to be unknown if this is a general result for constrained problems (it likely depends on the specific form of the constraints). The examples presented in Shapiro (1996) pertain to constrained problems where the solution lies on the boundary of the constraint set.

One obvious question for the sample path method is: What is a good choice for the reference value $\boldsymbol{\theta}'$ in the usual case where $q_V(\boldsymbol{\upsilon}) = p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta}')$? Choosing $\boldsymbol{\theta}'$ wisely can have a large impact on the performance of the sample path method. A good choice of $\boldsymbol{\theta}'$ can reduce the variance of the loss and gradient estimators, $\bar{L}_N(\boldsymbol{\theta})$ and $\partial\bar{L}_N(\boldsymbol{\theta})/\partial\boldsymbol{\theta}$, and contribute to a better estimate for $\boldsymbol{\theta}$. One formulation for picking $\boldsymbol{\theta}'$ is to solve the auxiliary optimization problem, $\min_{\boldsymbol{\theta}'} \mathrm{var}[\overline{L}_N(\boldsymbol{\theta})]$, prior to solving the main problem of interest (recall that $\overline{L}_N(\boldsymbol{\theta})$ depends on $\boldsymbol{\theta}'$ as shown in (15.16)). Unfortunately, this auxiliary problem is unsolvable in most practical applications. Special cases of this auxiliary problem have been considered in a number of references, including Dussault et al. (1997). For instance, in the context of queuing networks, it is beneficial to choose $\boldsymbol{\theta}'$ such that the amount of traffic in the network *exceeds* that found under more typical values of $\boldsymbol{\theta}$. Nevertheless, despite the lack of an easy general result, we saw in Example 15.7 above that even if $\boldsymbol{\theta}'$ is not chosen optimally, the sample path method may still yield reasonable results.

15.5 CONCLUDING REMARKS

This chapter continued in the spirit of Chapter 14 in showing how simulations can be used for optimizing physical processes. Of course, as in Chapter 14, the results here are critically dependent on the simulation being a valid representation of the actual process. In contrast to Chapter 14, however, the focus here has been on methods involving the direct measurement of the gradient of the loss function.

The gradient-based methods require much more information than the nongradient methods of Chapter 14, which use only simulation inputs and outputs. In particular, the analyst must know the inner workings of the simulation and/or the complete probability distribution(s) of the random variables being generated as part of the Monte Carlo aspect of the simulation. If such information is available, the gradient-based methods of this chapter usually result in improved performance (e.g., increased rate of convergence) relative to the methods of Chapter 14. Nevertheless, while the bulk of published research in simulation-based optimization for discrete-event systems has focused on gradient-based methods, most *applications* appear to rely on the simpler finite-difference and related search methods (Fu and Hu, 1997, p. 5).

Section 15.1 discussed conditions under which a derivative and integral can be interchanged (see also Appendix A). Such an interchange is critical to the validity of the gradient-based optimization methods. Sections 15.2 and 15.3 focused on the LR/SF and IPA forms of gradient estimators, and, through (15.4) in Section 15.1, on a general form including LR/SF and IPA as limiting special cases. The LR/SF and IPA algorithms are special cases of the stochastic gradient algorithm of Chapter 5, which itself is a special case of root-finding stochastic approximation (Chapter 4). Section 15.4 considered a sample path version of the gradient-based algorithms. Unlike the SA-based methods of Sections 15.2 and 15.3, the sample path method is based on a gradient from a fixed set of data resulting from a fixed set of simulation runs (not directly connected to the number of iterations of the algorithm). In this way, the *stochastic* optimization problem associated with Monte Carlo simulations is converted to a *deterministic* problem for which classical deterministic search methods can be used.

Arguably, among many possible unbiased direct gradient measurements, the basic gradient estimators of Sections 15.2–15.4 have attracted the most attention in the simulation-based optimization community. For the sake of thoroughness, let us briefly discuss some other approaches.

There is a whole family of *perturbation analysis methods* besides the IPA approach. One such relative of IPA is the smoothed perturbation analysis (SPA) method. Here, the observed loss $Q(\boldsymbol{\theta}, V)$ is replaced by a conditional expectation of Q, with the conditioning on a part of V. The book by Fu and Hu (1997) is devoted largely to this SPA approach. Such conditioning can often cope with cases where discontinuities cause a violation of the conditions for the interchange of derivative and integral. Another gradient estimation method is rare perturbation analysis (see, e.g., Vázquez-Abad, 1999, and references therein), which is based on deleting or adding events with very small probabilities of occurrence. Like SPA, this has the effect of smoothing out the measured loss function Q.

In a different spirit from the PA family of estimation methods is the frequency domain (harmonic analysis) method. This method is based on building an approximation to the simulation (the approximation is sometimes called a metamodel) and then estimating the gradient directly from the approximation (Mitra and Park, 1991; Jacobson, 1994). The frequency domain approach is based on oscillating the components of $\boldsymbol{\theta}$ in a sinusoidal fashion during a single simulation run. This provides sensitivity information that in turn leads to a gradient estimate through the metamodel. A key aspect of this approach is the determination of the time index, the frequency, and the amplitude of the sinusoidal input. However, there appear to be no recent results that alter the conclusion of L'Ecuyer (1991): "Whether or not (and in what situations) frequency domain estimation would be competitive in practice for derivative estimation is not clear at this point."

Other gradient estimation methods are discussed in L'Ecuyer (1991), Pflug (1996), and Arsham (1998), among other references. These include the weak derivative method, the gradient surface method, and higher-order (Newton–Raphson-like) methods. All of these methods are based on exploiting structure internal to the simulation to improve the performance of the search and optimization process. A good recent discussion of other issues associated with simulation-based optimization, including a list of some commercial packages, is Fu (2002).

EXERCISES

15.1 Consider the reliability problem in Example 15.1. Assume that instead of defining V as the "natural" component lifetimes, it is defined as having three elements, the elements being the number of times that the system incurs a fixed, failure, and replacement cost over the interval $[0, T]$. Derive the expression for $Q(\boldsymbol{\theta}, \boldsymbol{\upsilon})$ with the new definition of V. In contrast to Example 15.1, sketch an argument to show that $Q(\boldsymbol{\theta}, \boldsymbol{\upsilon})p_V(\boldsymbol{\upsilon}|\boldsymbol{\theta})$ is now a continuous function in $\boldsymbol{\theta}$. This may be done by considering the special case of two components in the system and deriving the probability of observing one failure and two replacements over $[0, T]$. (Note that the full power of Theorem 15.1 is unnecessary because the expectation integral is replaced by a triple summation.)

15.2 Verify that both the LR/SF and IPA gradient estimates in Example 15.3 are unbiased and show that the IPA estimate has smaller variance than the LR/SF estimate. (Hint: for the exponential random variable Z of Example 15.3, $E(Z^3) = 6\theta^3$ and $E(Z^4) = 24\theta^4$.)

15.3 Fill in the missing details in going from the definition of $p_V(\upsilon|\boldsymbol{\theta})$ to the stochastic gradient input $Y_k(\hat{\boldsymbol{\theta}}_k)$ in Example 15.4.

15.4 Consider the setting of Example 15.4. With the same probability mass function, but with the negative response function Q changed to $Q(\boldsymbol{\theta}, V) = (\beta - 10)^2 + 2V\lambda - V$, do the following:

(a) Determine $L(\boldsymbol{\theta})$.

(b) Analytically derive the value for $\boldsymbol{\theta}^*$ and $L(\boldsymbol{\theta}^*)$.

(c) Derive the stochastic gradient input $Y_k(\hat{\boldsymbol{\theta}}_k)$.

15.5 For the problem of Exercise 15.4, do the following:

(a) Using an initial $\boldsymbol{\theta}$ of $[0.5, 10.5]^T$ and $a_k = 0.1/(50+k)^{0.501}$, run five independent replications of the stochastic gradient algorithm, each for 1000 iterations. Show that the distance from $\boldsymbol{\theta}^*$ to the final $\boldsymbol{\theta}$ estimate (i.e., $\hat{\boldsymbol{\theta}}_{1000}$) is negligibly different for each of the five runs. For one of the runs, produce a table showing the $\boldsymbol{\theta}$ estimates and loss values at 0, 10, 100, and 1000 iterations (analogous to the numbers shown in Figure 15.1)

(b) Interpret the results here vis-à-vis the results of Example 15.5. Provide some conceptual discussion on the reasons for the difference in performance.

15.6 Consider the setting of Example 15.4. With the same probability mass function, but with the negative response function Q changed to $Q(\boldsymbol{\theta}, V) = (\beta - 1)^2 + 2V(\lambda - 1) + \beta(V + 1)$, do the following three tasks:

(a) Determine $L(\boldsymbol{\theta})$.

(b) Analytically derive the value for $\boldsymbol{\theta}^*$ and $L(\boldsymbol{\theta}^*)$.

(c) Derive the stochastic gradient input $Y_k(\hat{\boldsymbol{\theta}}_k)$.

15.7 For the problem of Exercise 15.6 with an initial $\boldsymbol{\theta}$ of $[0.75, -0.75]^T$ and $a_k = 0.025/(100+k)^{0.501}$, run five independent replications of the stochastic gradient algorithm, each for 10,000 iterations (not 1000 as in Example 15.4). Report the distance from $\boldsymbol{\theta}^*$ to the final $\boldsymbol{\theta}$ estimate (i.e., $\hat{\boldsymbol{\theta}}_{10,000}$) for each of the five runs. For one of the runs, produce a table showing the $\boldsymbol{\theta}$ estimate and loss values at 0, 10, 100, 1000, and 10,000 iterations (analogous to the numbers shown in Figure 15.1).

15.8 Consider the sample path method. Suppose that $p = 4$ and that each of the elements of $\hat{\boldsymbol{\theta}}_n$ are known to be within 0.1 unit of the corresponding elements in the optimum $\boldsymbol{\theta}_N^*$. Let the approximate distribution of $\boldsymbol{\theta}_N^*$ be $N(\boldsymbol{\theta}^*, \boldsymbol{C})$, $\boldsymbol{C} = [c_{ij}]$, $c_{ii} = 0.02$ for all i; $c_{ij} = 0.015$ for all $i \neq j$.

(a) Show that $\boldsymbol{C}$ is a valid covariance matrix.

(b) Using simulation or analytical analysis, provide a distance that is known with 99 percent certainty to be at least as large as the true distance between $\hat{\boldsymbol{\theta}}_n$ and $\boldsymbol{\theta}^*$.

(c) Contrast the solution in part (b) with an erroneous solution resulting from ignoring the correlations in $\boldsymbol{C}$ (i.e., a calculation based on incorrectly assuming that $\boldsymbol{C} = 0.02\boldsymbol{I}_4$).

15.9 Suppose that a scalar V has an exponential distribution with mean θ. Let $L(\theta) = \theta$. Derive the sample path gradient estimate using the generic LR/SF form and a reference parameter value θ'.

15.10 Consider the binary-outcome setting of Example 15.7. Generate one realization of $N = 20$ measurements based on taking $\theta' = 1/2$. Letting $b = 1/16$, contrast θ_N^* and θ^*.

15.11 Letting $\hat{\boldsymbol{\theta}}_0 = \boldsymbol{0}$, show by the method of induction that $\hat{\boldsymbol{\theta}}_N = -\boldsymbol{B}^{-1}\overline{\boldsymbol{V}}_N$, as stated in Example 15.8.