LA-UR- 00-4528

Approved for public release; distribution is unlimited.

Title: ACCELERATED MONTE CARLO FOR PARTICLE DISPERSION

Author(s): Mark Fitzgerald
Richard Picard

Submitted to:

Los Alamos
NATIONAL LABORATORY

Form 836 (8/00)

DISCLAIMER

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

ACCELERATED MONTE CARLO FOR PARTICLE DISPERSION

Mark Fitzgerald and Rick Picard
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

Key Words: adaptive methods; exponential convergence; importance sampling; Markov chains; simulation.

ABSTRACT

We discuss adaptive importance sampling for individual steps in Markov chains. Such chains arise in many simulation problems, such as related to airborne particle dispersion. Through use of experimental design for simulation runs and regression methods for analysis of the output, importance sampling transition kernels are defined which substantially improve convergence times.

1. INTRODUCTION

Computer simulation is widely used to examine physical processes when controlled experiments are expensive or impractical to conduct. Indeed, recent advances (in computing hardware and in parallelization, for example) have made simulation increasingly valuable. In many cases, however, runtimes are still sufficiently lengthy so as to preclude obtaining estimates of good precision and to inhibit investigation of the full range of scenarios of interest.

In this paper, we describe how fitted regression models of simulation output can be embedded in Markovian transition kernels to greatly reduce computation time. Those transition kernels alter the trajectories of simulated chains and improve efficiency of estimation, in some cases by orders of magnitude.

The primary motivation for this work involves the simulation of turbulent

particle dispersion. One related computer code is used by the National Atmospheric Release Advisory Center, for example, to aid in real-time emergency response to natural and man-made disasters (Sullivan, Ellis, and Drumtra 1997). Similar codes have been used in a reconstructive vein to better understand past events – such as the Chernobyl nuclear accident (Lange, Dickerson, and Gudiksen 1988) – as well as in a futuristic vein to plan for hypothetical ones (Saltbones, Foss, and Bartnicki 2000). More generally, Markovian simulation models are used to address basic science problems in fluid mechanics.

2. PARTICLE DISPERSION

Consider the collection of particles involved in a release, such as from an industrial smokestack. Those particles are eventually dispersed over a wide area, their transport being governed by local wind patterns, terrain, ground surface deposition, and other factors.

Much effort has been invested in developing stochastic models to accurately represent observed behavior (Thomson 1987; Ermak and Nasstrom 2000). A review of related physics is beyond the scope of this paper, but the bottom line is that each particle path is a transient Markov chain. The underlying stochastic models are incorporated into computer codes that receive input on source conditions and on transport dynamics, and then generate a large number of i.i.d. trajectories, from which simple summary statistics are extracted.

The Lagrangian approach of generating 'natural' particle trajectories is time consuming, especially when rare events are involved. Consider, for example, estimating by simulation the portion of particles emitted from the Chernobyl accident that reached Paris, France. Generating each individual particle trajectory is nontrivial, and millions of such trajectories are required to yield a precise simulation estimate of the quantity of interest.

A solution, then, is to *not* generate natural particle trajectories but to

instead generate 'biased' ones that are important to the problem at hand. By reweighting the results, estimation is greatly improved.

3. METHODOLOGY

We now describe importance sampling for transient Markov chains, and show how experimental design and regression are useful in improving simulation efficiency. The chain is denoted $\{x_n\ ;\ n = 0, 1, 2, ...\}$ where, relative to the simulation, x_n is the state of the particle at the n-th time step. In general, the state space is multivariate, with components that summarize particle location, velocity, direction, and other aspects of interest such as 'real' time.

The chain's transition kernel is denoted $p(x_{n-1}, x_n)$; that is, simulating the next state x_n given the current state x_{n-1} is equivalent to sampling from $p(x_{n-1}, x_n)$. The step at which absorption occurs is denoted τ, and the probability density function for a trajectory emanating from x_0 is

$$\prod_{n=1}^{\tau} p(x_{n-1}, x_n)\,.$$

Simulation is used to estimate properties of the physical process, here expressed as 'scores' that accumulate with each step of the chain. Letting $s(x_{n-1}, x_n)$ denote the score accumulated on step n, the overall score for a chain starting in state x_0 is

$$S(x_0) \;=\; \sum_{n=1}^{\tau} s(x_{n-1}, x_n)\,.$$

One common score is the indicator function of an event of interest (i.e., if that event occurs on step n, then $s(x_{n-1}, x_n) = 1$ and the chain is terminated; all other scores are zero). For airborne particle dispersion, simple examples are whether the simulated particle enters a specified volume or is deposited within a specified surface area.

To estimate the expected score

$$\widetilde{S}(x_0) \;\equiv\; E_p\left[S(x_0)\right]$$

of the chain beginning in x_0 and evolving according to the natural transition kernel $p(\cdot,\cdot)$, numerous chains starting from x_0 are typically simulated, a score for each chain obtained, and those scores are then averaged.

Importance sampling in this context amounts to choosing a transition kernel $q(x_{n-1}, x_n) \neq p(x_{n-1}, x_n)$ and simulating the corresponding chains. For an importance-sampled chain, the counterpart to the standard score is

$$S(x_0) \;=\; \sum_{n=1}^{\tau} \left\{ s(x_{n-1}, x_n) \, \frac{\prod_{i=1}^{n} p(x_{i-1}, x_i)}{\prod_{i=1}^{n} q(x_{i-1}, x_i)} \right\} . \tag{3.1}$$

[Note: (3.1) is not the classical importance sampling estimator except when scoring occurs only on the final step, but we prefer this estimator for reasons discussed in Section 4.]

For transient Markov chains with nonnegative scoring, there is a zero-variance transition kernel (Baggerly, Cox, and Picard 2000). That kernel is

$$q_{\widetilde{S}}(x_{n-1}, x_n) \;=\; p(x_{n-1}, x_n) \, \frac{s(x_{n-1}, x_n) + \widetilde{S}(x_n)}{\widetilde{S}(x_{n-1})} . \tag{3.2}$$

In other words, simulating a single Markov chain from x_0 using the kernel (3.2) would yield the solution $\widetilde{S}(x_0)$ without error. Of course, (3.2) depends on knowledge of the function $\widetilde{S}(x)$ for all states x, so the zero-variance result may not appear to be of much value.

What is accomplished by (3.2) is to point the way towards good transition kernels. Consider approximating the expected score $\widetilde{S}(x)$ with the linear model

$$\widetilde{S}(x) \;\approx\; \sum \beta_i B_i(x) \, , \tag{3.3}$$

where the $\{\beta_i\}$ are unknown parameters and the $\{B_i(x)\}$ are known basis functions. By simulating natural chains from a number of different initial states, the resulting scores can be used to estimate model parameters. Upon substituting parameter estimates $\{\widehat{\beta}_i\}$ into (3.3) to obtain an estimate $\widehat{S}(x)$ of $\widetilde{S}(x)$ and normalizing, a transition kernel $q_{\widehat{S}}(x_{n-1}, x_n)$ is defined which can be used to produce biased trajectories.

Iteration of the above yields an adaptive procedure with good properties. That is, each iteration of the adaptive process involves

1) simulating importance-sampled chains using the kernel $q_{\widetilde{S}}(x_{n-1}, x_n)$,

2) using the simulation output in a regression to estimate the parameters $\{\beta_i\}$ in (3.3) and obtaining the fitted model $\widehat{S}(x) = \sum \widehat{\beta}_i B_i(x)$, and

3) substituting $\widehat{S}(x)$ for $\widetilde{S}(x)$ in (3.2) and normalizing to give an updated importance sampled transition kernel.

In the ideal case where the model (3.3) is formally correct, it has been shown (Baggerly, Cox, and Picard 2000) that under mild regularity conditions the adaptive process converges exponentially quickly to the zero-variance solution. As such, its simulation estimates converge much more rapidly than the $N^{-1/2}$ rate for the standard approach, where N denotes the number of simulated trajectories. An example of this remarkable phenomenon is given in Section 5.1. More generally, the model (3.3) is only approximate, and the adaptation is terminated upon reaching its limiting performance. At that point, replicated simulation runs starting from x_0 are used to estimate $\widetilde{S}(x)$.

4. CONSIDERATIONS FOR IMPLEMENTATION

Several implementation details must be resolved in order for the approach to succeed. They involve

- experimental design (i.e., choosing a set of initial states for obtaining simulation data and determining the number of replications needed for each initial state),

- importance sampling estimation (i.e. incorporation of variance reduction techniques such as splitting and survival biasing, as well as the definition of the score such as (3.1) for each simulated trajectory),

- regression (i.e., choosing the basis functions in the model (3.3) and estimating parameters from the heteroscedastic regression), and
- identifying a stopping rule for the adaptation to assess when the limiting ability of the fitted model to describe the data is reached.

These details are discussed in turn.

In each iteration of the adaptation, an experimental design is used to produce data $\{x^{(i)}, \bar{s}(x^{(i)})\}$, where $\bar{s}(x^{(i)})$ denotes the average of the replicated scores for simulated chains whose initial state is the i-th design point $x^{(i)}$. From the data $\{x^{(i)}, \bar{s}(x^{(i)})\}$, a fitted model $\widehat{S}(x)$ of the true mean response function $\widetilde{S}(x)$ obtained by regression. In determining the design points $\{x^{(i)}\}$, the set of trajectories produced by the design should encompass the chains important to the calculation at hand. Examples are given in the next section.

The number of replications per design point is a related design issue. In what follows, we use the same number of replications for each design point, though this is not necessary. If the number of replications is much too small, the algorithm may not converge; otherwise, use of a nonoptimal number of replications has a mild effect on computational efficiency.

When simulating importance-sampled chains, actions such as particle splitting and survival biasing should be taken to reduce variance. These techniques are not new (see, e.g., Lux and Kolbinger 1991), but are reviewed briefly for a self-contained exposition.

As expected from (3.1), if a simulated trajectory has a large likelihood ratio $R_n = \prod_{i=1}^{n} p(x_{i-1}, x_i) \;/\; \prod_{i=1}^{n} q(x_{i-1}, x_i)$, the trajectory's score could destabilize the estimate. This phenomenon is well known for ordinary importance sampling (Owen and Zhou 2000) and the same is true in a random walk context. The average likelihood ratio, with respect to the kernel $q(\cdot, \cdot)$, is 1, but large ratios are possible. To reduce the effects of large ratios, R_n is monitored during each simulated trajectory and, if it becomes too large, the particle is 'split.'

Splitting amounts to replacing a particle with several subparticles. That is, if at step n a trajectory in state x_n has a likelihood ratio R_n that is too large, the particle is replaced by k subparticles at x_n with likelihood ratios R_n/k; subparticle paths are then simulated and their scores added to replace the original particle score that would have been obtained. In what follows, we split particles whenever $R_n > 2$, and the number of subparticles being the smallest k such that $R_n/k < 2$. Note that there exists the possibility that subparticles may themselves be split at later time steps.

Survival biasing is useful in conjunction with partially reflecting barriers, as often occur in airborne dispersion. Similar to particle splitting, it reduces the likelihood ratio to improve estimation. If a particle reaches a partially reflecting barrier and is to be absorbed with probability π or be reflected with probability $1 - \pi$, the simulation *always* reflects the particle and updates the likelihood ratio to $(1 - \pi)R_N$. In this way, the particle continues to score in a way that gives unbiased estimation with reduced variance.

In terms of scoring, various estimators can be used, as noted by Hesterberg (1995) for ordinary importance sampling. We choose (3.1) in place of the classical estimator because it is frequently superior to the standard estimate (Glasserman 1993) and is also amenable to Rao-Blackwellization. As implemented here, the Rao-Blackwellized score generalizing (3.1) is

$$S^*(x_0) = \sum_{n=1}^{\tau} \left\{ E[s(x_{n-1}, x_n) \,|\, x_{n-1}] \, \frac{\prod_{i=1}^{n} p(x_{i-1}, x_i)}{\prod_{i=1}^{n} q(x_{i-1}, x_i)} \right\} , \qquad (4.1)$$

The one-step conditional expectations in (4.1) are often simple to compute and aid in reducing variability.

Once scores are obtained, regression is used to produce the fitted model $\widehat{S}(x)$. Not surprisingly, choice of basis functions is problem-dependent. There is a tradeoff of statistical efficiency versus computational efficiency: statistical efficiency involves minimum variance per simulated trajectory, motivating a fitted model resembling the actual mean response as closely as possible, while

computational efficiency involves simulation from the transition kernel $q_{\widehat{S}}(\cdot,\cdot)$, motivating a form of $\widehat{S}(x)$ to allow efficient random number generation. Moreover it is necessary that $\widehat{S}(x) > 0$, so that substitution into (3.2) gives a well defined kernel. Fortunately, good results can often be achieved with simple models, especially when fit piecewise to localized regions of the space – a truncated second-order response surface performed well in the examples here.

Use of fitted models requires a stopping rule in adaptation. The goal is to reach the limiting ability of the model to explain the data, at which point importance-sampled chains from x_0 are used to estimate the quantity of interest. In examples to follow, adaptation ceases when the moving average over three iterations of adaptation of the residual sum of squares for the regression increases. Other convergence diagnostics could be considered, of course.

5. EXAMPLES

5.1 Plume Spread for Dispersion in Homogeneous Turbulence

To begin, we take a 'toy' problem, namely dispersion in homogeneous turbulence. The simulated process is as follows. At time $t_0 = 0$, a particle is released in the atmosphere at height z_0 having vertical velocity $w_0 = 0$. Indexing time in units of the simulated time step Δt, the states of the Markov chain are denoted $x_t = (t, z_t, w_t)$, and the natural process evolves as

a) update the time to $t + \Delta t$,

b) update the vertical velocity to $w_t = \phi w_{t-1} + \eta_t$ as per a first-order autoregressive model, where $\phi = 1 - \Delta t/T_L > 0$ with T_L the Lagrangian time scale, and $\eta_t \sim N(0, \sigma_w^2)$,

c) update the particle height to $z_t = z_{t-1} + \Delta t\,[w_t + w_{t-1}]/2$, and

d) terminate the particle trajectory if the event(s) of interest have occurred; otherwise, return to (a) and continue.

We are interested in estimating the plume spread at time $\tau = 1000$,

$$\widetilde{S}(x_0) \;=\; E_p[(z_\tau - z_0)^2 \,|\, x_0] \;.$$

Owing to the simple recursion involved, it is not hard to show that the actual regression model has the form

$$\begin{aligned}\widetilde{S}(x_t) \;=\; & \beta_0 + \beta_1 t + \beta_2(1/\phi)^{\tau-t} + \beta_3(1/\phi^2)^{\tau-t} + \beta_4 z_t^2 + \beta_5 w_t^2 + \beta_6 z_t w_t \\ & + \beta_7 z_t w_t(1/\phi)^{\tau-t} + \beta_8 w_t^2(1/\phi)^{\tau-t} + \beta_9 w_t^2(1/\phi^2)^{\tau-t}\,, \qquad (5.1)\end{aligned}$$

where the parameters $\{\beta_j\}$ depend on specifics such as $\Delta t, \sigma_w$, and so on. For the results to follow, we use values $T_L = 10$ and $\sigma_w = 0.1$.

The experimental design used is a factorial with 750 design points, using all possible (t, z, w) combinations from the sets

$$\begin{aligned} t \;&\in\; \{0, 50, 100, 150, 200, 250, 300, 350, 400, 450, \\ &\qquad 500, 550, 600, 650, 700, 750, 800, 850, 900, 910, \\ &\qquad 920, 930, 940, 950, 960, 970, 980, 990, 995, 998\}\;, \\ z \;&\in\; \{z_0, z_0 \pm 200, z_0 \pm 400\}\;, \text{ and} \\ w \;&\in\; \{0, \pm 1, \pm 2\}\;. \end{aligned}$$

These design points encompass the set of trajectories likely to be encountered, and devote greater emphasis to the (more important) later times.

In the first iteration of the adaptive algorithm, 25 natural trajectories are simulated from each design point, and the resulting scores used to estimate parameters in (5.1). In subsequent iterations, 25 biased trajectories are simulated using importance sampled kernels $q_{\widetilde{S}}(\cdot,\cdot)$. Simulated trajectories incorporated particle splitting, scores were obtained using (4.1), and regression parameters are estimated by ordinary least squares.

Results are averaged over 50 runs of more than 2000 CPU seconds each and displayed in Figure 1 for the adaptive approach and for simulation of the

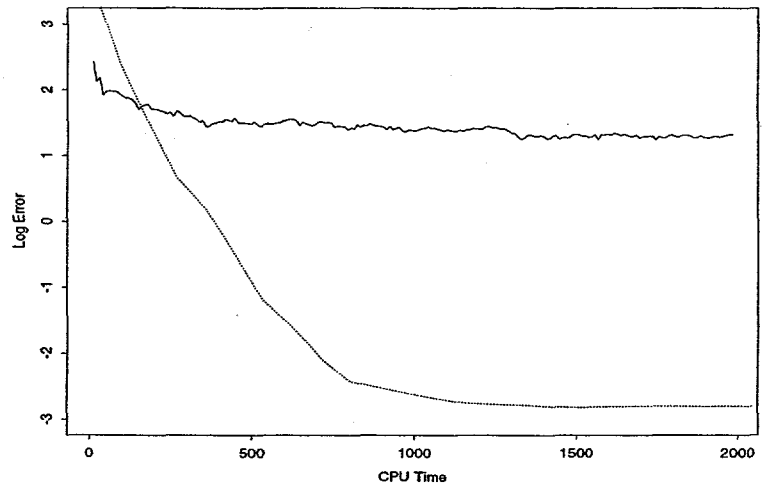

Figure 1: Log_{10} simulation error as a function of CPU time for the plume spread example, for simulation of the natural process (solid curve) and for simulation of the adaptive process (dashed curve).

natural process. Log error for the adaptive procedure decreases linearly as a function of CPU time until machine precision for the calculation is reached, a phenomenon known as exponential convergence (e.g., Booth 1997). By comparison, log error for simulation of the natural process decreases as log $N^{-1/2}$.

5.2 Approximate Zero-Variance Models

Comparisons such as Figure 1 are unfair, in that adaptation to the zero-variance kernel requires far more knowledge than does simulation of the natural process. Consider extending the previous problem by using a quadratic model

$$\widetilde{S}(x_t) \approx \beta_0^* + \beta_1^* t + \beta_2^* z_t + \beta_3^* w_t + \beta_4^* t^2 + \beta_5^* z_t^2 + \beta_6^* w_t^2 + \beta_7^* t z_t + \beta_8^* t w_t + \beta_9^* z_t w_t \,, \tag{5.2}$$

The second-order response surface (5.2) provides a smooth approximation to the actual model (5.1) over the range of interest.

Using the above experimental design, the adaptive algorithm is implemented with the approximate model. Adaptation terminates when the moving

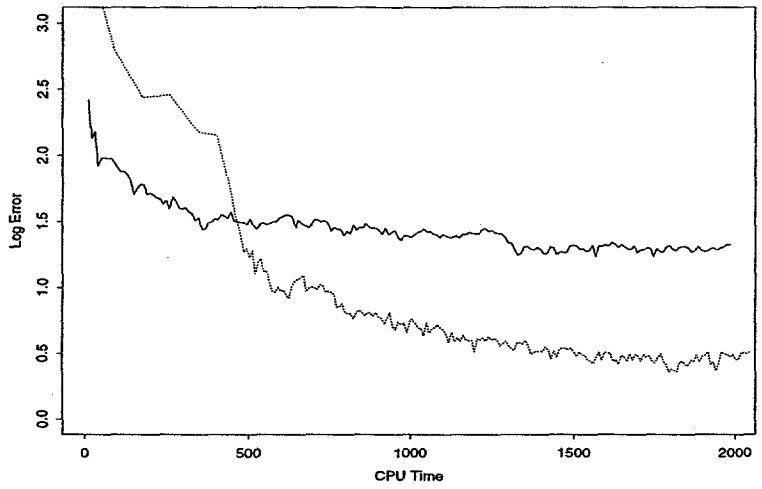

Figure 2: Log_{10} simulation error as a function of CPU time for the plume spread example, for simulation of the natural process (solid curve) and for simulation of the adaptive approximate zero-variance process (dashed curve).

average over three iterations of the residual sum of squares increases, at which point the resulting kernel $q_{\widehat{S}}(\cdot,\cdot)$ is used to simulated biased trajectories from x_0. Results, again averaged over 50 runs for simulation of the natural process and of the adaptive approach, are displayed in Figure 2.

Although simulating a biased trajectory is roughly 6 times more computational work than simulating a natural one, overall gains are impressive. At 2000 CPU seconds, the difference in error is roughly $10^{0.83} \approx 6.7$; owing to the $N^{-1/2}$ behavior of the simulation error, this means that the natural process would have to be run a factor of $6.7^2 \approx 45$ times longer in order to achieve the same accuracy as the adaptive approach.

5.3 Particle Deposition

Consider another example, intended to illustrate other problems encountered in particle dispersion. A particle is released at a low height ($z_0 = 5\,m$) and has a downward drift of $\delta = 0.5\ m/s$. The ground (the plane $z = 0\ m$) acts as a partially reflecting barrier: when the particle reaches the ground there

is a probability π that the particle is deposited there; otherwise, the particle is reflected upward and continues on. The goal is to estimate the probability that the particle is airborne after $t = 1000\ s$.

The corresponding natural process is

a) update the time to $t + \Delta t$,

b) update the vertical velocity to $w_t = \phi w_{t-1} + \eta_t$, where $\phi = 1 - \Delta t/T_L > 0$ with T_L the Lagrangian time scale, and $\eta_t \sim N(0, \sigma_w^2)$,

c) update the particle height to $z_t = z_{t-1} + \Delta t\,[w_t + w_{t-1}]/2 - \Delta t\,[\delta]$,

d) if $z_t < 0$, the particle is deposited on the ground with probability π; if it is not deposited, then set $z_t = |z_t|$ and set $w_t = |w_t + \delta|$, thereby ensuring that $z_t \geq 0$ and $w_t \geq 0$ after reflection, and

e) terminate the process if the particle has been deposited on the ground or if the time step $t = 1000$; otherwise, return to (a) and continue.

We use deposition probability $\pi = 0.75$ for each reflection in (d).

Here, most simulated particles are deposited on the ground well in advance of the $1000th$ simulated time step. Indeed, the probability of a particle being airborne after such a long time is roughly 5×10^{-5}. Importance sampling offers potentially great benefits in the estimation of rare event probabilities.

We use a simple factorial design with structure

$$\begin{aligned} t &\in \{0\,s,\ 250\,s, 500\,s, 750\,s, 995\,s\}\ , \\ z &\in \{1.33\,m, 5\,m, 15\,m, 45\,m, 135\,m\}\ , \text{ and} \\ w &\in \{-1\,m/s, 0\,m/s, 1\,m/s, 2\,m/s, 3\,m/s\}\ , \end{aligned}$$

and 50 reps per design point per run. Log deposition probabilities are modeled with a second-order response surface as before (which is not a particularly good fit, but illustrates that convergence is greatly accelerated even when coarse

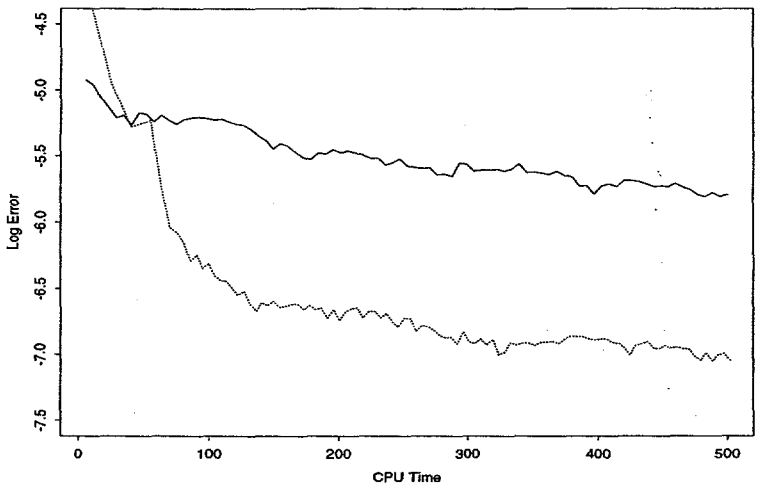

Figure 3: Log_{10} simulation error as a function of CPU time for the particle deposition example, for simulation of the natural process (solid curve) and for simulation of the adaptive approximate zero-variance process (dashed curve).

models are used), the stopping rule for the adaptation is as before, and survival biasing is used in addition to particle splitting and Rao-Blackwellization. Figure 3 displays results averaged over 50 runs.

Although simulation of a natural trajectory is 35 times faster, on average, than simulation of a biased one (recall that most natural trajectories involve early deposition on the ground), the adaptive approach is vastly superior. The difference in error at 500 CPU seconds is $10^{1.23} \approx 17$. In other words, the natural process would have to be simulated a factor of terms of $17^2 \approx 289$ times longer to achieve the same precision.

6. CONCLUDING REMARKS

Regression modeling of simulation output, which is also helpful in understanding the simulated process, is useful in defining importance sampling kernels as per approximations to (3.2). As illustrated, the approximate zero-variance transition kernels offer the potential to accelerate convergence relative to standard simulation by orders of magnitude.

To date, the adaptive approach described herein has not been used on large-scale problems. Theoretical underpinnings of the adaptive zero-variance approach are recent. Moreover, owners of simulation codes, who typically have nonstatistical backgrounds, tend to think solely in terms of simulating the natural physical process and the notion of generating 'unnatural' trajectories may not only be counter-culture but also disconcerting. And, no doubt, the necessary regression modeling and multivariate simulation from kernels $q_{\widehat{S}}(\cdot,\cdot)$ are more difficult in higher dimensions than in the examples here.

The potential of adaptive importance sampling to greatly accelerate the convergence of particle dispersion codes is considerable. For codes that are run frequently (e.g., with different input parameters), an effort to augment them with importance sampling techniques can have significant payoffs.

ACKNOWLEDGMENT

This work was supported by the Mathematical, Information, and Computational Sciences (MICS) program of the U.S. Department of Energy.

BIBLIOGRAPHY

Baggerly, K., Cox, D., and Picard R. (2000). "Exponential Convergence of Adaptive Importance Sampling for Markov Chains." *Journal of Applied Probability*, 37, 342-358.

Booth, T. E. (1997). "Exponential Convergence on a Continuous Monte Carlo Transport Problem." *Nuclear Science and Engineering*, 127, 338-345.

Ermak, D. L. and Nasstrom, J. S. (2000). "A Lagrangian Stochastic Diffusion Method for Inhomogeneous Turbulence." *Atmospheric Environment*, 34, 1059-1068.

Glasserman, P. (1993). "Stochastic Monotonicity and Conditional Monte

Carlo for Likelihood Ratios." *Advances in Applied Probability* 25, 103-115.

Hesterberg, T. (1995). "Weighted Average Importance Sampling and Defensive Mixture Distributions." *Technometrics* 37, 185-194.

Lange, R., Dickerson, M. H., and Gudiksen, P.H. (1988). "Dose Estimates from the Chernobyl Accident." *Nuclear Technology*, 82, 311-323.

Lux, I. and Koblinger, L. (1991). *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations.* Boca Raton, CRC Press.

Owen, A. and Zhou, Y. (2000). "Safe and Effective Importance Sampling." *Journal of the American Statistical Association*, 95, 135-143.

Saltbones, J., Foss, A., and Bartnicki, J. (2000). "Threat to Norway from Potential Accidents at the Kola Nuclear Power Plant." *Atmospheric Environment*, 34, 407-418.

Sullivan, T. J., Ellis, J. S., and Drumtra, P. (1997). "The Atmospheric Release Advisory Capability." Lawrence Livermore National Laboratory Technical Report UCRL-TB-125345.

Thomson, D. J. (1987). "Criteria for the Selection of Stochastic Models of Particle Trajectories in Turbulent Flows." *Journal of Fluid Mechanics*, 180, 529-556.