Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

OPTIMIZING THE DESIGN OF A LATIN HYPERCUBE SAMPLING ESTIMATOR

Alexander J. Zolan
John J. Hasenbein

Department of Mechanical Engineering
University of Texas at Austin
204 E Dean Keeton St.
Austin, TX 78712, USA

David P. Morton

Department of Industrial Engineering and Management Sciences
Northwestern University
2145 Sheridan Road
Evanston, IL 60208, USA

ABSTRACT

Stratified sampling and Latin hypercube sampling (LHS) reduce variance, relative to naïve Monte Carlo sampling, by partitioning the support of a random vector into strata. When creating these estimators, we must determine: (i) the number of strata; and, (ii) the partition that defines the strata. In this paper, we address the second point by formulating a nonlinear optimization model that designs the strata to yield a minimum-variance stratified sampling estimator. Under a discrete set of candidate boundary points, the optimization model can be solved via dynamic programming. We extend this technique to LHS, using an approximation of estimator variance to obtain strata for the domain of a multivariate function. Empirical results show significant variance reduction compared to using equal-probability strata for LHS or naïve Monte Carlo sampling.

1 INTRODUCTION

Variance reduction techniques are commonly used in Monte Carlo simulation to reduce the number of samples required to achieve a confidence interval of desired width when estimating the expectation of a univariate or multivariate function in cases where analytical calculations are intractable. In the simpler univariate case, Neyman (1934) develops a stratified sampling estimator, wherein the support of the random variable is partitioned into strata. Straightforward extension of this idea to higher dimensions does not scale well because the number of strata grows exponentially in the dimension, but Latin hypercube sampling (LHS), introduced by McKay, Beckman, and Conover (1979), provides a type of multivariate stratification.

When the underlying d-dimensional random vector has independent components, an LHS estimator partitions the support of each component into equal-probability strata, and exploits independence by randomly ordering samples from each component to form d-tuples. Iman and Conover (1980) generalize LHS to allow for cells of unequal probability, and characterize the sampling variance of an LHS estimator under specific conditions. Stein (1987) characterizes the asymptotic variance of an LHS estimator, relative to a naïve Monte Carlo estimator, and Owen (1992) establishes a central limit theorem for LHS. Drew and

Homem-de-Mello (2012) show that the upper bound on the probability of a large deviation under LHS is no higher than that of naïve Monte Carlo sampling.

LHS and its extensions are seen in a variety of applications. Olsson, Sandberg, and Dahlblom (2003) implement LHS in the estimation of structural reliability. Freimer, Linderoth, and Thomas (2012) and Stockbridge and Bayraksan (2016) assess the impact of LHS on reducing the the variance and bias when estimating optimal values in stochastic optimization problems. Packham and Schmidt (2010) establish a central limit theorem for LHS with dependent variables, and apply this to the valuation of first-to-default credit baskets and Asian basket options. Helton and Davis (2003) use the sample reweighting technique for nonuniform LHS developed by Iman and Conover (1980) to conduct uncertainty and sensitivity analysis for a two-phase fluid flow model. Morton et al. (2014) implement nonuniform LHS to ensure the sampling of rare events to assess risk in nuclear power.

Mease and Bingham (2006) study how to optimize the strata of a nonuniform LHS estimator, and to our knowledge this is the closest work in the literature to what we propose. They derive first-order optimality conditions when the dimension and number of samples are small, but they say that this approach does not scale well. So, they employ a heuristic search involving coordinate descent on a grid for larger problems. We similarly begin by formulating a nonlinear optimization model, but then recast that model as a dynamic program.

In this paper, we develop a method of choosing nonuniform strata over the support of a random variable with the goal of minimizing sampling variance for stratified sampling or, in the case of multivariate sampling, LHS. Section 2 formulates a nonlinear program to construct a stratified sampling estimator with minimum variance, and reformulates that optimization model as a tractable dynamic program. The development of this stratified sampling estimator is not particularly useful in its own right. Rather, we view it as a subroutine for developing the LHS estimator that we propose in Section 3. Section 4 details empirical results of our stratified sampling and LHS schemes for a collection of example functions. Section 5 concludes.

2 NONUNIFORM STRATIFIED SAMPLING

We wish to estimate $\mathbb{E}[h(X)]$, in which X is a univariate random variable and $h : \mathbb{R} \to \mathbb{R}$. To do so, we use a stratified sampling routine, which partitions the support of X into contiguous strata, $\mathscr{S}_k$, $k = 1, 2, \ldots, K$, which we also call cells. Here, each cell has probability mass $p_k = \mathbb{P}(X \in \mathscr{S}_k)$, and the estimator allocates a sample size n_k to cell k. In this section, we formulate a nonlinear program to find the cell widths that yield a minimum-variance estimator, restricting attention to the univariate case. In Section 3, we address multivariate sampling under an LHS framework.

2.1 Assumptions

We assume X is a continuous random variable with known probability density function (pdf), $f(x)$, and cumulative distribution function (cdf), $F(x)$. We assume that we have in analytical form, $F^{-1}(u)$, for $u \in [0,1]$, or that we can numerically evaluate this inverse cdf. We assume that we can compute $p_k = \mathbb{P}(X \in \mathscr{S}_k)$, for each cell as well as relevant expectations. Further, we assume we know the desired number of strata, K, and our total computational budget, N. Our goal is to select the breakpoints and sample sizes to yield a stratified sampling estimator of minimum variance.

2.2 Nonlinear Programming Formulation

Our stratified sampling estimator has the following form:

$$h_N = \sum_{k=1}^{K} p_k \bar{h}_{n_k}, \tag{1}$$

in which $\bar{h}_{n_k}$ is a sample mean of n_k independent and identically distributed (i.i.d.) observations of $[h(X)|X \in \mathscr{S}_k]$; that is, i.i.d. observations of $h(X)$, conditioned on X being in cell k. We therefore have that

$$\mathbb{V}\mathrm{ar}[h_N] = \sum_{k=1}^{K} p_k^2 \frac{\mathbb{V}\mathrm{ar}[h(X)|X \in \mathscr{S}_k]}{n_k}. \tag{2}$$

The optimization model that we formulate in this section assumes the cells $\mathscr{S}_k$ are of the form $\mathscr{S}_k = (F^{-1}(b_{k-1}), F^{-1}(b_k))$, in which $0 = b_0 < b_1 < b_2 < \cdots < b_K = 1$, and with this construct aims to minimize $\mathbb{V}\mathrm{ar}[h_N]$.

Sets and Indices
$k \in \mathscr{K} = \{1, 2, \ldots, K\}$: indices defining the strata
Data
h: a univariate function $h : \mathbb{R} \to \mathbb{R}$
f: pdf of X
F: cdf of X
N: total number of samples
Decision Variables
b_k: selection of breakpoint k in the interval $[0,1]$ used to create strata
μ_k: $\mathbb{E}[h(X)\,|\,X \in \mathscr{S}_k]$, where $\mathscr{S}_k = (F^{-1}(b_{k-1}), F^{-1}(b_k))$
σ_k^2: $\mathbb{E}[(h(X) - \mu_k)^2\,|\,X \in \mathscr{S}_k]$
p_k: $\mathbb{P}(X \in \mathscr{S}_k) = b_k - b_{k-1}$
n_k: number of samples allocated to cell k
Boundary Conditions
$b_0 = 0$
$b_K = 1$

Note:
We use b, μ, σ^2, p, and n to denote the vectors $(b_0, b_1, \ldots, b_K)$, $(\mu_1, \mu_2, \ldots, \mu_K)$, etc.
Formulation

$$\min_{b,\mu,\sigma^2,p,n} \quad \sum_{k \in \mathscr{K}} \frac{p_k^2 \sigma_k^2}{n_k} \tag{3a}$$

$$\text{s.t.} \quad \mu_k = \frac{\int_{F^{-1}(b_{k-1})}^{F^{-1}(b_k)} h(x) f(x) dx}{b_k - b_{k-1}}, \quad \forall k \in \mathscr{K}, \tag{3b}$$

$$\sigma_k^2 = \frac{\int_{F^{-1}(b_{k-1})}^{F^{-1}(b_k)} (h(x) - \mu_k)^2 f(x) dx}{b_k - b_{k-1}}, \quad \forall k \in \mathscr{K}, \tag{3c}$$

$$p_k = b_k - b_{k-1}, \quad \forall k \in \mathscr{K}, \tag{3d}$$

$$\sum_{k \in \mathscr{K}} n_k = N, \tag{3e}$$

$$p_k \geq 0, \; n_k \geq 0, \quad \forall k \in \mathscr{K}. \tag{3f}$$

Discussion
The objective in (3a) seeks a stratification of X's support to minimize the variance of the estimator, as indicated in equation (2). Constraints (3b) and (3c) define μ_k and σ_k^2, respectively; both constraints are nonlinear in the decision variables, b. Constraint (3d) relates cell k's width, p_k, to the location of its breakpoints, b_k and b_{k-1}. This, coupled with constraint (3f) and the boundary conditions, ensures that $0 = b_0 \leq b_1 \leq \cdots \leq b_{K-1} \leq b_K = 1$. Constraint (3e) restricts the sum of sample sizes to the computational

budget, N. We have relaxed an integer restriction on the sample sizes, n_k, in constraint (3f), which allows an optimal solution to allocate a fractional number of samples for each cell.

We view model (3) as notional in the sense that, if we could compute exactly terms like μ_k in (3b) then we could compute $\mathbb{E}[h(X)]$ exactly, and we would not employ Monte Carlo sampling. However, as indicated above, we extend this idea in the next section to an LHS estimator, in which we assume relevant one-dimensional integrals are tractable. In what follows, we reformulate model (3) in two ways. First, we remove n_k because we can analytically optimize with respect to n for fixed values of the breakpoints. Second, we create a discrete set of candidate breakpoints from which strata may be constructed. These two steps are discussed in Sections 2.3 and 2.4, respectively.

2.3 Objective Function Reformulation

Suppose the breakpoints $b_k, k \in \mathscr{K}$, are known. Then constraints (3b)-(3d) can be removed and the resulting optimization problem is:

$$\min_{n} \quad \sum_{k\in\mathscr{K}} \frac{p_k^2\sigma_k^2}{n_k} \tag{4a}$$

$$\text{s.t.} \quad \sum_{k\in\mathscr{K}} n_k = N \tag{4b}$$

$$n_k \geq 0, \quad \forall k \in \mathscr{K}. \tag{4c}$$

The optimal solution of model (4) is achieved when n_k is proportional to $p_k\sigma_k$, i.e.,

$$n_k = N\left(\frac{p_k\sigma_k}{\sum_{k\in\mathscr{K}} p_k\sigma_k}\right);$$

see, e.g., Neyman (1934). Substituting this value for n_k into the objective function in (4a) yields:

$$\frac{1}{N}\left(\sum_{k\in\mathscr{K}} p_k\sigma_k\right)^2. \tag{5}$$

The revised objective function in (5) allows model (3) to simplify to:

$$\min_{b,\mu,\sigma^2} \quad \sum_{k\in\mathscr{K}} (b_k - b_{k-1})\sigma_k \tag{6a}$$

$$\text{s.t.} \quad 0 = b_0 \leq b_1 \leq \cdots \leq b_{K-1} \leq b_K = 1 \tag{6b}$$

$$\text{(3b)-(3c).} \tag{6c}$$

The breakpoints b_k, $k \in \mathscr{K}$, are the primary decision variables in model (6), as variables μ_k and σ_k^2 are determined by the specification of these breakpoints. While model (6) is still nonconvex, the additive form of the objective function in (6a) allows for the development of a dynamic programming algorithm that we describe in Section 2.4, at least when we restrict the choices of b_k to a prespecified univariate grid.

2.4 Dynamic Programming Algorithm

Let $\mathscr{B} = \{b^0, b^1, b^2, \ldots, b^L\}$ specify a partition of $[0,1]$, in which $0 \equiv b^0 < b^1 < b^2 < \cdots < b^L \equiv 1$. We consider the restriction of model (6) in which we add the constraint $b_k \in \mathscr{B}$, $k \in \mathscr{K}$. Each term in the objective function of (6a) then has the form $(b^{\ell'} - b^{\ell})\sigma(b^{\ell}, b^{\ell'})$, where

$$\sigma^2(b^{\ell}, b^{\ell'}) = \frac{\int_{F^{-1}(b^{\ell})}^{F^{-1}(b^{\ell'})} (h(x) - \mu(b^{\ell}, b^{\ell'}))^2 f(x)dx}{b^{\ell'} - b^{\ell}}, \tag{7}$$

and where

$$\mu(b^{\ell}, b^{\ell'}) = \frac{\int_{F^{-1}(b^{\ell})}^{F^{-1}(b^{\ell'})} h(x) f(x) dx}{b^{\ell'} - b^{\ell}}, \tag{8}$$

for $\ell = 1, 2, \ldots, L, \ell' = \ell + 1, \ell + 2, \ldots, L$.

We can solve this variant of model (6) via a dynamic programming algorithm, which can be visualized using the directed acyclic graph (DAG) shown in Figure 1. The DAG has nodes (k, ℓ) for $k = 0, 1, \ldots, K$,

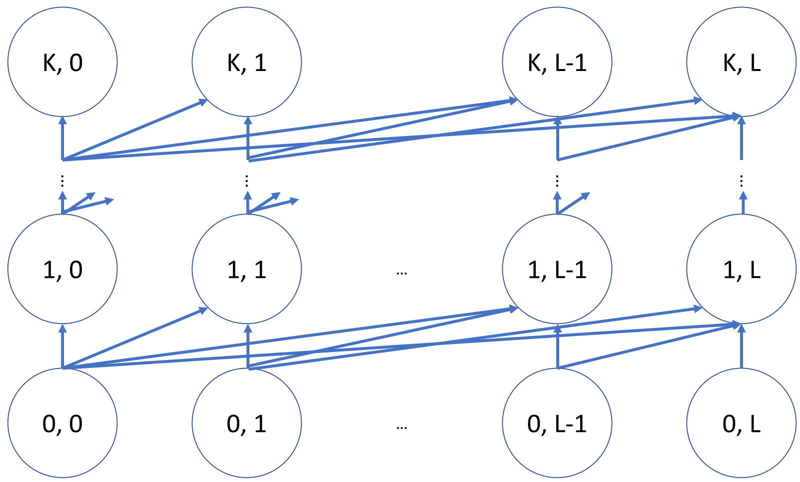

Figure 1: Shortest-path problem associated with the dynamic programming solution of model (6) under the restriction that each b_k comes from a set of finite, prespecified breakpoints. We create an edge from node (k, ℓ) to node $(k+1, \ell')$, for all $k = 0, \ldots, K-1$, $\ell = 0, \ldots, L$, $\ell' = \ell, \ldots, L$, with length $(b^{\ell'} - b^{\ell})\sigma(b^{\ell}, b^{\ell'})$, in which $\sigma^2(b^{\ell}, b^{\ell'})$ is defined in equation (7). If node (k, ℓ) is part of the shortest path from $(0,0)$ to (K, L), then breakpoint $b_k = b^{\ell}$ is in the obtained optimal solution.

and for $\ell = 0, 1, \ldots, L$. For all $k = 0, 1, \ldots, K-1$, we create an edge from node (k, ℓ) to node $(k+1, \ell')$, for all $\ell' = \ell, \ldots, L$, with length $(b^{\ell'} - b^{\ell})\sigma(b^{\ell}, b^{\ell'})$, in which $\sigma(b^{\ell}, b^{\ell'})$ is defined in equation (7). The shortest path from node $(0,0)$ to (K, L) then specifies an optimal solution to model (6), under the restriction $b_k \in \mathscr{B}, k \in \mathscr{K}$.

We note that computing the edge lengths in the DAG of Figure 1 requires more effort than computing $\mathbb{E}[h(X)]$, because $\mathbb{E}[h(X)]$ is given by the sum of $(b^{\ell'} - b^{\ell})\mu(b^{\ell}, b^{\ell'})$ along any path in the DAG from $(0,0)$ to (K, L). Therefore, we emphasize that we do not view this as useful for reducing the variance of stratified sampling estimators; rather, we view it as a subroutine for an optimized LHS estimator that we describe next.

3 NONUNIFORM LHS

This section extends the method described in Section 2 to higher dimensions to optimize an LHS routine. Let $X = (X(1), X(2), \ldots, X(d))$ be a vector of independent random variables, and let $h : \mathbb{R}^d \to \mathbb{R}$; further, suppose we plan to use LHS to estimate $\mathbb{E}[h(X)]$. Similar to the stratified sampling procedure in Section 2, we partition the support of each random variable $X(i)$ into K strata, $\mathscr{S}_k(i), k = 1, 2, \ldots, K$. However, for each random variable $X(i)$, exactly one value $X^k(i)$ is sampled from each cell, $k = 1, \ldots, K$. Next, the K realizations from $X(1)$ are randomly paired, without replacement, with the realizations from $X(2)$. These are, in turn, paired at random with the other components of X, until there are K generated d-tuples:

$$X^k = (X^k(1), X^k(2), \ldots, X^k(d)), k = 1, 2, \ldots, K.$$

We obtain these d-tuples in the following way. For $i = 1,2,\dots,d$, let $(\pi(i,1),\pi(i,2),\dots,\pi(i,K))$ denote a random permutation of $\{1,2,\dots,K\}$. Then, for $i = 1,2,\dots,d$, and $k \in \mathscr{K}$, let

$$X^k(i) \sim [X(i)\,|\,X(i) \in \mathscr{S}_{\pi(i,k)}(i)].$$

The set of possible combinations for a K-tuple, generated by $\pi(\cdot)$, represents a partition of the support of X into K^d cells.

Let $\Gamma^k = \mathscr{S}_{\pi(1,k)}(1) \times \mathscr{S}_{\pi(2,k)}(2) \times \cdots \times \mathscr{S}_{\pi(d,k)}(d)$ be the Cartesian product of the chosen strata for each random variable $X^k(i),\ i = 1,\dots,d$, and let

$$\mathbb{P}(\Gamma^k) = \prod_{i=1}^{d} \mathbb{P}[X(i) \in \mathscr{S}_{\pi(i,k)}(i)],\ k = 1,2,\dots,K.$$

In this setting the LHS estimator given by

$$h_K^{LHS} = \sum_{k=1}^{K} K^{d-1}\mathbb{P}(\Gamma^k)h(X^k) \tag{9}$$

was proposed by Iman and Conover (1980) to extend the work of McKay, Beckman, and Conover (1979). In McKay et al., the support of each random variable has equal-probability strata, meaning

$$\mathbb{P}[X(i) \in \mathscr{S}_{\pi(i,k)}(i)] = \frac{1}{K},\ i = 1,2,\dots,d,\ k = 1,2,\dots,K.$$

In this case, the weights on $h(X^k)$ in equation (9) are simply $1/K$. Under nonuniform LHS, the weights are instead random due to $\mathbb{P}(\Gamma^k)$, because the cells Γ^k are determined by a random permutation and can have unequal probability.

3.1 Assumptions

Similar to the assumptions of Section 2.1, we wish to estimate $\mathbb{E}[h(X)]$ via an estimator with minimum variance, except we assume that X is multivariate, with independent components. For each random variable $X(i),\ i = 1,\dots,d$, we assume we have, or can numerically evaluate, the inverse cdf, $F_i^{-1}(u)$, for $u \in [0,1]$. Our goal is to select a set of breakpoints that define the strata of each random variable $X(i),\ i = 1,\dots,d$, to minimize total sampling error under an LHS routine.

3.2 LHS Optimization Model

While the variance of LHS estimators has been characterized in various ways in the literature (see, e.g., Iman and Conover 1980, Stein 1987, Homem-de-Mello 2008, Drew and Homem-de-Mello 2012), these characterizations provide insight as opposed to lending themselves to estimation. In order to guide design of the cells we use in our LHS estimator, we make the following approximation:

$$h(X) \approx \sum_{i=1}^{d} h_i(X), \tag{10}$$

in which

$$h_i(X) = h(a(1),a(2),\dots,a(i-1),X(i),a(i+1),\dots,a(d-1),a(d)),$$

where in the numerical experiments we describe in Section 4 we use $a(i) = \mathbb{E}[X(i)]$, for $i = 1,2,\dots,d$. Applying the LHS estimator to the right-hand side of the approximation (10) amounts to performing stratified sampling on each term $h_i(X)$, where the stratification is only on component $X(i)$, albeit with one sample per cell. Thus, following equation (2) with $n_k = 1,\ k \in \mathscr{K}$,

$$\mathbb{V}\mathrm{ar}[h_K^{LHS}] \approx \sum_{i=1}^{d} \sum_{k\in\mathscr{K}} p_k^2(i)\sigma_k^2(i), \tag{11}$$

in which $\sigma_k^2(i)$ is given by

$$\sigma_k^2(i) = \mathbb{V}\mathrm{ar}\left[h_i(X) | X(i) \in \mathscr{S}_k(i)\right], \tag{12}$$

for $k = 1,2,\ldots,K$. Minimizing the LHS variance of the right-hand side of approximation (11) leads to d separate optimization problems of the form:

$$\min_{b(i),\mu(i),\sigma^2(i)} \quad \sum_{k\in\mathscr{K}} (b_k(i) - b_{k-1}(i))^2 \sigma_k^2(i) \tag{13a}$$

$$\text{s.t.} \quad \mu_k(i) = \frac{\int_{F_i^{-1}(b_{k-1}(i))}^{F_i^{-1}(b_k(i))} h_i(x) f_i(x)dx}{b_k(i) - b_{k-1}(i)}, \quad \forall k \in \mathscr{K} \tag{13b}$$

$$\sigma_k^2(i) = \frac{\int_{F_i^{-1}(b_{k-1}(i))}^{F_i^{-1}(b_k(i))} (h_i(x) - \mu_k(i))^2 f_i(x)dx}{b_k(i) - b_{k-1}(i)}, \quad \forall k \in \mathscr{K} \tag{13c}$$

$$0 = b_0(i) \le b_1(i) \le \cdots \le b_{K-1}(i) \le b_K(i) = 1, \tag{13d}$$

for $i = 1,2,\ldots,d$. Here, f_i denotes the marginal pdf of $X(i)$, and the vectors $b(i)$, $\mu(i)$, and $\sigma^2(i)$ are as defined in Section 2.2, except that $\mu(i)$, and $\sigma^2(i)$ are now defined with respect to the univariate $h_i(X)$.

3.3 LHS Strata Selection

With an objective function that separates by each component of X, we can apply the dynamic programming approach of Section 2.4, d times in solving model (13). We assume that we can numerically compute the univariate integrals that define the $O(dKL^2)$ edges that compose the d DAGs; i.e., we can numerically compute $\sigma^2(i)(b^\ell, b^{\ell'})$ and $\mu(i)(b^\ell, b^{\ell'})$ as defined by equations (7) and (8), respectively, for each component i. The shortest paths from the d DAGs define the K^d cells from which we obtain LHS samples.

4 RESULTS

In this section, we solve model (3) for a collection of univariate functions for stratified sampling, and we solve model (13) d times for a collection of multivariate functions for LHS. We then compare the sampling error under the strata we obtain to that of equal-probability strata and, in the case of LHS, to that of naïve Monte Carlo sampling. The results demonstrate that sampling with strata that are optimized to approximately minimize the variance of the corresponding estimators via the techniques of Sections 2 and 3 can yield significant variance reduction compared to using equal-probability strata.

4.1 Stratified Sampling

We illustrate the potential of our procedure by first applying the dynamic programming procedure of Section 2.4 to a collection of univariate functions. We do not perform any Monte Carlo simulation for the results in this section. Rather, we numerically compute the optimal value of model (3), which we denote z^*. Then, we compute the same objective function, i.e., the variance of a stratified estimator, using equal-probability strata; we denote that value z^u, and we form the variance reduction factor z^u/z^*. All experiments use $K = 10$ cells and $L = 100$ candidate breakpoints, and we note that the ratio we report is independent of the total sample size, N.

Table 1: Variance reduction factors and optimized strata boundary points for a collection of univariate functions of random variables. Notation z^* and z^u denote the stratified sampling estimator variance under optimized and equal-probability strata, respectively.

$h(x)$	Distribution	z^u/z^*	$b=(b_0,b_1,\ldots,b_K)$
x	χ_1^2	3.32	[0.00,0.29,0.47,0.61,0.72,0.81,0.88,0.93,0.97,0.99,1.00]
$\log(x)$	χ_1^2	1.69	[0.00,0.02,0.06,0.13,0.22,0.33,0.46,0.61,0.76,0.90,1.00]
x^2	χ_1^2	10.48	[0.00,0.51,0.69,0.79,0.86,0.91,0.94,0.96,0.98,0.99,1.00]
e^x	$N(0,1)$	4.02	[0.00,0.22,0.42,0.59,0.72,0.82,0.89,0.94,0.97,0.99,1.00]
x	$N(0,1)$	1.25	[0.00,0.04,0.12,0.23,0.36,0.50,0.64,0.77,0.88,0.96,1.00]
e^x	Beta(1,5)	1.86	[0.00,0.18,0.34,0.48,0.61,0.72,0.81,0.88,0.94,0.98,1.00]
x	Beta(1,5)	1.52	[0.00,0.16,0.31,0.44,0.56,0.67,0.77,0.85,0.92,0.97,1.00]
$\log(x)$	Beta(1,5)	1.59	[0.00,0.02,0.07,0.14,0.23,0.34,0.47,0.61,0.76,0.90,1.00]
x^2	Beta(1,5)	3.91	[0.00,0.33,0.52,0.65,0.75,0.83,0.89,0.94,0.97,0.99,1.00]
e^x	Beta(5,1)	1.30	[0.00,0.04,0.10,0.18,0.27,0.37,0.48,0.60,0.73,0.86,1.00]
x	Beta(5,1)	1.52	[0.00,0.03,0.08,0.15,0.23,0.33,0.44,0.56,0.69,0.84,1.00]
$\log(x)$	Beta(5,1)	2.13	[0.00,0.01,0.04,0.09,0.16,0.25,0.36,0.49,0.64,0.81,1.00]
x^2	Beta(5,1)	1.23	[0.00,0.04,0.10,0.18,0.27,0.37,0.48,0.60,0.73,0.86,1.00]
x	Exp(1)	2.13	[0.00,0.19,0.36,0.51,0.64,0.75,0.84,0.91,0.96,0.99,1.00]
$\log(x)$	Exp(1)	1.52	[0.00,0.02,0.07,0.15,0.25,0.37,0.51,0.65,0.79,0.92,1.00]
x^2	Exp(1)	6.77	[0.00,0.39,0.59,0.72,0.81,0.88,0.93,0.96,0.98,0.99,1.00]
x	Gamma(2,1)	1.64	[0.00,0.12,0.26,0.41,0.55,0.67,0.78,0.87,0.94,0.98,1.00]
$\log(x)$	Gamma(2,1)	1.38	[0.00,0.02,0.07,0.15,0.26,0.39,0.53,0.67,0.81,0.93,1.00]
x^2	Gamma(2,1)	4.16	[0.00,0.27,0.46,0.61,0.73,0.82,0.89,0.94,0.97,0.99,1.00]
x	Gamma(5,1)	1.39	[0.00,0.09,0.22,0.36,0.50,0.63,0.75,0.85,0.93,0.98,1.00]
$\log(x)$	Gamma(5,1)	1.30	[0.00,0.03,0.10,0.20,0.32,0.45,0.59,0.72,0.84,0.94,1.00]
x^2	Gamma(5,1)	2.36	[0.00,0.17,0.34,0.50,0.64,0.75,0.84,0.91,0.96,0.99,1.00]
e^x	Weibull(2,1)	2.43	[0.00,0.15,0.32,0.48,0.62,0.74,0.84,0.91,0.96,0.99,1.00]
x	Weibull(2,1)	1.26	[0.00,0.09,0.21,0.34,0.47,0.60,0.72,0.82,0.91,0.97,1.00]
$\log(x)$	Weibull(2,1)	1.52	[0.00,0.02,0.07,0.15,0.25,0.37,0.51,0.65,0.79,0.92,1.00]
x^2	Weibull(2,1)	2.13	[0.00,0.19,0.36,0.51,0.64,0.75,0.84,0.91,0.96,0.99,1.00]

Table 1 displays the results for a collection of univariate functions that use the notation from Section 2.2. Figure 2 shows the variance reduction factor achieved as a function of the skewness of each univariate function, and suggests that the value of optimized nonuniform strata increases with skewness. Figure 3 plots the optimal breakpoints for a subset of the univariate cases, and illustrates the variety of solutions we obtain from different functions of the same random variable. The maximum time to create and solve the dynamic program for a test case was less than one minute, using a Cray XC40 compute node with two Intel E5-2690 v3 12-core (Haswell) processors and 64 GB of DDR4 memory.

4.2 Nonuniform LHS

To illustrate the impact of our method on reducing the variance of a multivariate LHS estimator, we use repeated experiments, implemented in Python (Rossum 1995) using uniform random variates generated via the WELL512 implementation developed by Panneton, L'Ecuyer, and Matsumoto (2006) and available in the Stochastic Simulation in Java library created by L'Ecuyer, Meliani, and Vaucher (2002). We use a nonlinear function $h(x_1,x_2,\ldots,x_d)=\prod_{i=1}^{d} x_i$, and estimate $\mathbb{E}[h(X)]$ for different values of d across a collection of probability distributions.

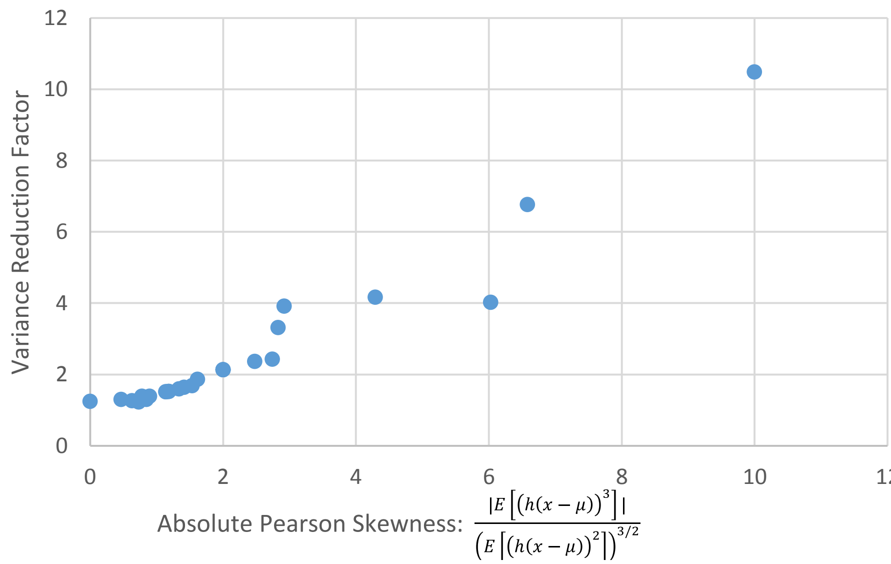

Figure 2: Variance reduction factors of optimized vs. equal-probability strata, plotted as a function of skewness for the collection of univariate functions given in Table 1.

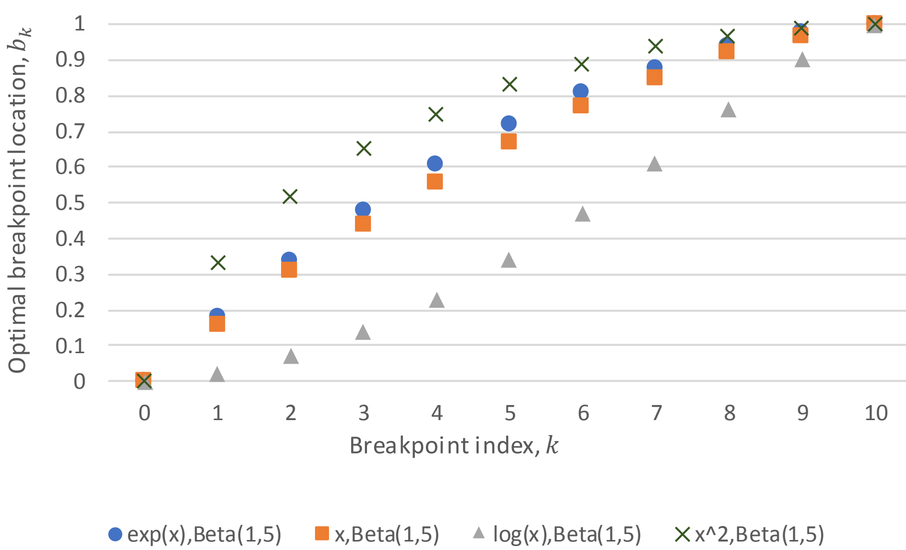

Figure 3: Plot of optimal breakpoints for a collection of functions of a Beta(1,5) random variable.

We consider three estimators in total: (i) naïve Monte Carlo; (ii) LHS with equal-probability cells; and, (iii) LHS as defined in equation (9). For estimator (iii), we optimize the strata for each component of X by the dynamic programming scheme of Sections 2.4 and 3.3.

For each of the three estimators, we form M i.i.d. replicates of the estimator which, in turn, use K samples. For example, under the LHS estimator of equation (9), we form $\bar{h}_K^m$, $m = 1,2,\ldots,M$, i.i.d. replicates and form the sample mean estimator

$$\bar{h}_{K,M} = \frac{1}{M}\sum_{m=1}^{M}\bar{h}_K^m. \tag{14}$$

Analogous estimators are formed for the uniform LHS estimator and for naïve Monte Carlo, and for all three estimators we use common random numbers, i.e., identical streams of uniform random variates.

We then compare the sample variance of $\bar{h}_{K,M}$ obtained under each method to obtain empirical variance reduction factors. In the results we report, we use $M = 10,000$ replicates, $K = 100$ strata, and $L = 1,000$ candidate breakpoints, and we report 95% confidence intervals (CIs) for the variance reduction factors in Table 2 by replicating this experiment 100 times, comparing estimator (iii) with both (i) and (ii). More specifically, our variance reduction factors correspond to the ratio of sample variances of $1,000,000 = 100 \cdot 10,000$ terms, $\bar{h}^m_K$, from the right-hand side of equation (14) for estimators (i), (ii), and (iii). We note that while an LHS application would likely use a much lower value for M, and instead increase K according to the computational budget, our sample sizes are inflated to illustrate the value of optimizing the LHS cells in higher dimensions, for which the sampling error can be volatile.

Table 2 demonstrates the significant variance reduction offered by the LHS procedure in Section 3.3, when compared to LHS with equal-probability strata. The lower bound of the 95% confidence interval estimate for the variance reduction factor versus uniform LHS is greater than one for all cases, and it exceeds an order of magnitude for more than half the test cases. Further, the factor tends to grow as the dimension d grows larger. The maximum time to create and solve the dynamic programs for a test case was 15 minutes, using the same computational resources as in Section 4.1.

5 CONCLUSION

This paper presents a method of minimizing the variance of a stratified sampling estimator by formulating a nonlinear program, and solving this problem exactly via a dynamic program using a discrete set of candidate stratum boundary points. We extend this technique to the multivariate setting and reduce the variance of an LHS estimator as compared to equal-probability strata, using an approximation of the estimator's variance and solving a dynamic program for each random component. Finally, this paper details empirical results that exhibit significant LHS variance reduction under this technique, compared to that of equal-probability strata and naïve Monte Carlo sampling.

We note that the collection of examples we use are functions for which $\mathbb{E}[h(X)]$ can be obtained analytically, and are not practical in their own right. However, we believe they illustrate the potential of our proposal for improving the efficiency of LHS estimators by optimizing nonuniform strata. For applications in which sampling is computationally expensive, we suspect that the improved efficiency will make it worthwhile to incur the cost of solving the dynamic programs.

Topics of future research include the exploration of alternatives to taking $a(i) = \mathbb{E}[X(i)]$ in the approximation of equation (10), and applying the procedure from Section 3 to LHS applications in the literature. In particular, we are interested in investigating the impact of optimizing nonuniform LHS on reducing the bias and variance associated with the estimator for the cost of an optimal solution to stochastic programming problems in the literature, such as those in Freimer, Linderoth, and Thomas (2012) and Stockbridge and Bayraksan (2016), and comparing the performance of our method to that of the variance reduction techniques the authors use. The approach we propose could further be compared with that of Mease and Bingham (2006). It would be interesting to explore conditions for which the procedure in Section 3 guarantees variance reduction compared to LHS using equal-probability cells, for both equation (10) and other approximations. Finally, extension of the method in Section 3 to design optimal LHS strata with the independence assumption relaxed is an opportunity for further research.

ACKNOWLEDGMENTS

This work was supported by the South Texas Project Nuclear Operating Company, via grant number BO4425. The authors express sincere gratitude for the support. We thank two anonymous referees who made suggestions that improved the paper.

Table 2: Empirically obtained point estimates and 95% CI half-widths of variance reduction factors associated with estimating $\mathbb{E}[h(X)]$, in which $h(x) = \prod_{i=1}^{d} x_i$, using optimized LHS strata obtained by the procedure in Section 3.3, as compared to LHS with equal-probability strata and naïve Monte Carlo sampling. Confidence intervals were obtained via 100 repeated experiments, each of which use $M = 10,000$ replicates, $K = 100$ strata, and $L = 1,000$ candidate breakpoints for each dynamic programming routine. CI half-width values are reported as a percentage of the corresponding point estimate. The experiments were implemented in Python 2.7.9 using the WELL512 generator developed by Panneton, L'Ecuyer, and Matsumoto (2006), via the Stochastic Simulation in Java library created by L'Ecuyer, Meliani, and Vaucher (2002). Common random numbers are used in each experiment; separate substreams were used for permutations and for the uniform random variates used to generate realizations of X.

		Variance reduction factor vs. equal-probability strata		Variance reduction factor vs. naïve Monte Carlo	
Distribution	d	Point estimate	95% CI half-width (%)	Point estimate	95% CI half-width (%)
χ_1^2	2	129.1	0.9	251.1	1.0
χ_1^2	4	341.4	5.6	372.5	7.3
χ_1^2	6	836.5	12.0	1,477.2	64.6
χ_1^2	8	2,510.1	48.1	3,478.3	43.3
Beta(1,2)	2	5.1	0.7	25.3	0.6
Beta(1,2)	4	6.1	1.2	11.9	1.1
Beta(1,2)	6	7.5	1.8	10.6	1.7
Beta(1,2)	8	9.6	2.9	11.3	3.4
Exponential	2	35.8	0.8	104.3	0.7
Exponential	4	57.8	2.7	77.9	2.8
Exponential	6	97.5	5.7	122.6	16.3
Exponential	8	180.1	14.5	219.0	19.8
Gamma(2,1)	2	9.7	0.7	47.0	0.6
Gamma(2,1)	4	11.7	1.7	22.8	1.5
Gamma(2,1)	6	14.7	2.9	21.1	4.0
Gamma(2,1)	8	19.2	5.0	23.8	7.2
Weibull(2,1)	2	2.2	0.8	17.7	0.6
Weibull(2,1)	4	2.3	1.2	6.8	1.1
Weibull(2,1)	6	2.4	1.7	4.8	1.5
Weibull(2,1)	8	2.6	2.2	4.1	2.4

REFERENCES

Drew, S., and T. Homem-de-Mello. 2012. "Some Large Deviations Results for Latin Hypercube Sampling". *Methodology and Computing in Applied Probability* 14 (2): 203–232.

Freimer, M. B., J. Linderoth, and D. J. Thomas. 2012. "The Impact of Sampling Methods on Bias and Variance in Stochastic Linear Programs". *Computational Optimization and Applications* 51 (1): 51–75.

Helton, J. C., and F. J. Davis. 2003. "Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems". *Reliability Engineering and System Safety* 81:23–69.

Homem-de-Mello, T. 2008. "On Rates of Convergence for Stochastic Optimization Problems Under Non-independent and Identically Distributed Sampling". *SIAM Journal on Optimization* 19 (2): 524–551.

Iman, R. L., and W. J. Conover. 1980. "Small Sample Sensitivity Analysis Techniques for Computer Models, with an Application to Risk Assessment". *Communications in Statistics - Theory and Methods* 9 (17): 1749–1842.

L'Ecuyer, P., L. Meliani, and J. Vaucher. 2002. "SSJ: A Framework for Stochastic Simulation in Java". In *Proceedings of the 2002 Winter Simulation Conference*, edited by E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 234–242. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

McKay, M. D., R. J. Beckman, and W. J. Conover. 1979. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code". *Technometrics* 21 (2): 239–245.

Mease, D., and D. Bingham. 2006. "Latin Hyperrectangle Sampling for Computer Experiments". *Technometrics* 48 (4): 467–477.

Morton, D., B. Letellier, J. Tejada, D. Johnson, Z. Mohaghegh, E. Kee, V. Moiseytseva, S. Reihani, and A. Zolan. 2014. "Sensitivity Analyses of a Simulation Model for Estimating Fiber-induced Sump Screen and Core Failure Rates". In *International Conference on Nuclear Engineering*, Volume 6: American Society of Mechanical Engineers.

Neyman, J. 1934. "On the Two Different Aspects of the Representative Method: the Method of Stratified Sampling and the Method of Purposive Selection". *Journal of the Royal Statistical Society* 97 (4): 558–625.

Olsson, A., G. Sandberg, and O. Dahlblom. 2003. "On Latin Hypercube Sampling for Structural Reliability Analysis". *Structural Safety* 25:47–68.

Owen, A. 1992. "A Central Limit Theorem for Latin Hypercube Sampling". *Journal of the Royal Statistical Society. Series B (Methodological)* 54 (2): 541–551.

Packham, N., and W. M. Schmidt. 2010, Spring. "Latin Hypercube Sampling with Dependence and Applications in Finance". *The Journal of Computational Finance* 13 (3): 81–111.

Panneton, F., P. L'Ecuyer, and M. Matsumoto. 2006. "Improved Long-period Generators Based on Linear Recurrences Modulo 2". *ACM Transactions on Mathematical Software* 32 (1): 1–16.

Rossum, G. 1995. "Python Reference Manual". Technical Report CS-R9525, Amsterdam, The Netherlands. https://ir.cwi.nl/pub/5008/05008D.pdf", accessed 04/01/2017.

Stein, M. 1987. "Large Sample Properties of Simulations Using Latin Hypercube Sampling". *Technometrics* 29 (2): 143–151.

Stockbridge, R., and G. Bayraksan. 2016. "Variance Reduction in Monte Carlo Sampling-based Optimality Gap Estimators for Two-stage Stochastic Linear Programming". *Computational Optimization and Applications* 64 (2): 407–431.

AUTHOR BIOGRAPHIES

ALEXANDER ZOLAN is a PhD candidate in the Graduate Program of Operations Research and Industrial Engineering at the University of Texas at Austin. His research interests include stochastic optimization and its interface with simulation, as well as simulation of energy systems. His email is alex.zolan@utexas.edu.

JOHN HASENBEIN is a professor in the Graduate Program of Operations Research and Industrial Engineering at the University of Texas at Austin. He has interests in the areas of queueing theory, Markov decision processes, and simulation of wafer fabrication systems. His email address is jhas@mail.utexas.edu and his webpage is http://www.me.utexas.edu/∼has/.

DAVID MORTON is a professor in the Department of Industrial Engineering and Management Sciences and the director of the Center for Optimization and Statistical Learning at Northwestern University. His research interests include stochastic optimization and its interface with simulation. His email address is david.morton@northwestern.edu and his webpage is http://www.mccormick.northwestern.edu/research-faculty/directory/profiles/morton-david.html.