A stochastic mesh method for pricing high-dimensional American options

Mark Broadie
Graduate School of Business, Columbia University, 3022 Broadway, New York, NY, 10027-6902, USA

Paul Glasserman
Graduate School of Business, Columbia University, 3022 Broadway, New York, NY, 10027-6902, USA

High-dimensional problems frequently arise in the pricing of derivative securities – for example, in pricing options on multiple underlying assets and in pricing term structure derivatives. American versions of these options, ie, where the owner has the right to exercise early, are particularly challenging to price. We introduce a stochastic mesh method for pricing high-dimensional American options when there is a finite, but possibly large, number of exercise dates. The algorithm provides point estimates and confidence intervals; we provide conditions under which these estimates converge to the correct values as the computational effort increases. Numerical results illustrate the performance of the method.

1 Introduction

Pricing a derivative security entails calculating the expected discounted value of its payoff. This reduces, in principle, to a problem of numerical integration; but in practice this calculation is often difficult for high-dimensional pricing problems. High-dimensionality arises in pricing options on multiple underlying assets and in pricing options in models that capture many sources of risk, such as stochastic volatility, interest rates and exchange rates.

Pricing high-dimensional options is further complicated for American versions of these securities, ie, where the owner has the right to exercise early. Although there are many techniques for pricing American options on a single underlying asset – including lattices, PDE methods, variational inequalities, and integral equation methods – when these techniques are generalized to handle multiple state variables, they require work that is exponential in the number of state variables. This work requirement renders these methods ineffective for more than about three or four state variables.

This is a revised version of an article widely circulated as a working paper starting in 1997 and presented at numerous seminars and conferences, including talks at IBM, the Board of Governors of the Federal Reserve, the World Bank, McGill University, University of Warwick, Delft University of Technology, Aarhus University, Hong Kong University of Science and Technology, ETH Zurich, CREST (Paris), MIT, University of Minnesota, Purdue University, Courant Institute, the Center for Applied Finance in Singapore, and several RISK courses and conferences. We thank the referee for useful comments.

A distinct advantage of Monte Carlo simulation is that its convergence rate is typically independent of the number of state variables. Another advantage is the ease with which it can handle a wide range of models and payoff structures. However, the traditionally prevailing view has been that simulation methods are not applicable to American-style pricing problems. The major obstacle is that simulation typically generates trajectories of state variables forward in time, while the determination of optimal exercise policies requires backward-style dynamic programming techniques. That view has changed as several hybrid simulation–dynamic programming methods for attacking these problems have been proposed.

Heuristic methods for applying simulation to American option pricing include Tilley (1993), Barraquand and Martineau (1995), Raymar and Zwecher (1997), and Andersen (2000), among many others. These are heuristic in the sense that little can be said about the relation between the values to which they converge and the desired option price, though they may provide good approximations in specific cases. Broadie and Glasserman (1997) develop a method with theoretical support based on simulated trees. Their method generates two estimators, a lower bound and an upper bound (ie, one biased low and one biased high[1]), with both estimators convergent and asymptotically unbiased as the computational effort increases. A valid confidence interval for the true American price is obtained by taking the upper confidence limit from the "high" estimator and the lower confidence limit from the "low" estimator. The main drawback of this method is that the work is exponential in the number of exercise opportunities. A further discussion of these and other approaches is given in Boyle, Broadie, and Glasserman (1997) and in Glasserman (2004).

In this paper we introduce a stochastic mesh method for pricing high-dimensional American options when there is a finite, but possibly large, number of exercise dates. The method provides lower and upper bounds and confidence intervals for the true price, and we give conditions under which it converges as the computational effort increases. The work of the algorithm is linear in the number of exercise opportunities and quadratic in the number of points in the mesh. It is also linear in the work required to simulate a single state transition.[2] The linear, rather than exponential, dependence on the number of exercise dates is in marked contrast to the random tree method. The work requirement of the stochastic mesh method makes it viable for pricing high-dimensional American options.

Any method for pricing American options by simulation can be viewed as generating random approximations to the dynamic programming operator

[1] Throughout this paper, a *lower bound* in the simulation context means that the simulation estimator is biased low. In other words, $E(X) \le Q$, where the random variable X represents the simulation estimator and Q is the true American option price. Likewise, the simulation estimator X is an *upper bound* if $E(X) \ge Q$, ie, if X is biased high.

[2] The work required to simulate a state transition is often linear in the number of state variables but is potentially quadratic in, eg, simulating a discrete-time approximation to a stochastic differential equation. The time required to generate the mesh paths is, in any case, a relatively small portion of the total time required by the method.

that recursively determines the option value. The method of Barraquand and Martineau (1995) can be viewed as generating an approximation based solely on the evolution of the option's intrinsic value. The approximating dynamic program implicit in Broadie and Glasserman (1997) assigns equal weight to each branch in a randomly sampled tree. Carrière (1996), Longstaff and Schwartz (2001), and Tsitsiklis and Van Roy (1999) combine simulation with regression on a set of basis functions to develop low-dimensional approximations to high-dimensional dynamic programs, in the same spirit as some deterministic numerical methods (see, eg, Judd, 1998). As explained in Section 8.6.2 of Glasserman (2004), those methods are related to the stochastic mesh introduced here and correspond to an implicit choice of *mesh weights*. The stochastic mesh method and a random successive approximation method proposed and analyzed by Rust (1997) both approximate the dynamic programming operator using values of the transition density of the underlying process, but the methods differ in the way they use these values and in the scope of problems to which they apply. Subsequent work on the mesh method introduced here includes Avramidis and Hyden (1999), Avramidis and Matzinger (2004), Avramidis *et al* (2000), Boyle, Kolkiewicz, and Tan (2000, 2002), Broadie, Glasserman, and Ha (2000), and Broadie, Glasserman, and Jain (1997).

An interesting new line of research on the pricing of American options by simulation is the development of dual formulations by Haugh and Kogan (2004), Jamshidian (2003), and Rogers (2002). These provide a framework for calculating tight upper bounds on American option prices. Andersen and Broadie (2004) present a practical way of computing these bounds. They also show how to combine a lower bound computed from a heuristic or other method with an upper bound extracted from the same method through the dual formulation. This combination provides an interval estimate for the true price, in the same spirit as the interval estimates in Broadie and Glasserman (1997) and in this paper. Glasserman (2004, pp. 477–8), notes a connection between the upper bounds computed through duality and those developed through approximate dynamic programming, as in this paper.

The next section gives a description and theoretical analysis of the basic stochastic mesh method. This, however, is just the starting point, as it leaves open several questions of implementation. Section 3 develops a specific method based on a particularly effective choice of *mesh density*. Section 4 develops several enhancements that are crucial in practice to obtaining accurate price estimates in reasonable computing time. Computational results are given in Section 5. Proofs are given in the Appendix.

2 The stochastic mesh method

The stochastic mesh method is designed to solve a general optimal stopping problem, of which the American option pricing problem with discrete exercise opportunities is a special case. Let $S_t = (S_t^1, \ldots, S_t^n)$ be a vector-valued Markov process on R^n with fixed initial state S_0 and discrete time parameter $t = 0, 1, \ldots, T$.

The problem is to compute

$$Q = \max_{\tau} E\Big[h(\tau, S_\tau)\Big] \tag{1}$$

where τ is a stopping time taking values in the finite set $\{0, 1, \ldots, T\}$, and $h(t, x) \geq 0$ is interpreted as a payoff from exercise at time t in state x.[3] More generally, the value starting at time t in state x is

$$Q(t,x) = \max\Big(h(t,x), E\Big[Q(t+1, S_{t+1}) \Big| S_t = x\Big]\Big) \tag{2}$$

for $t < T$ and $Q(T, x) = h(T, x)$. We are interested in computing $Q \equiv Q(0, S_0)$. In an important special case, the vector of state variables S_t is governed by risk-neutral probabilities and $h(t, x)$ gives the payoff in state x at time t, discounted to time 0, with the possibly stochastic discount factor recorded in S_t. More generally, h could give the payoff in units of an arbitrary numeraire asset contained in the vector of state variables with the law of the state variables adjusted accordingly.

Examples: For illustration, we give a few selected examples of payoff functions on multiple assets. For a basket call option, the payoff function is $h(t, S_t) = (a_1 S_t^1 + \cdots + a_n S_t^n - K)^+$ for given constants $a_1, \ldots, a_n$ and strike K.[4] For a quanto spread option, $h(t, S_t) = S_t^1(a_2 S_t^2 - a_3 S_t^3 - K)^+$, where S_t^1 represents an exchange rate or another random quantity adjustment. For a spread option on two baskets, $h(t, S_t) = (a_1 S_t^1 + a_2 S_t^2 - (a_3 S_t^3 + a_4 S_t^4) - K)^+$. As a final example, $h(t, S_t) = \big(\max(a_1 S_t^1, \ldots, a_n S_t^n) - K\big)^+$ for a max-option (also called an outperformance option). If the S_t^i are prices of discount bonds of various maturities (in, eg, a Gaussian model of interest rates), then the payoff given above for a basket option becomes the payoff of an option on a coupon-paying bond.

The stochastic mesh method begins by generating random vectors $X_t(i)$ for $i = 1, \ldots, b$ and $t = 1, \ldots, T$. Methods for generating the *stochastic mesh* $X_t(i)$ will be described shortly. Since S_0 is given, we set $X_0(1) = S_0$. The mesh estimator is defined inductively by setting

$$\hat{Q}(T, X_T(i)) = h(T, X_T(i)) \tag{3}$$

for $i = 1, \ldots, b$. For times $t = T-1, \ldots, 0$ and $i = 1, \ldots, b$, the mesh estimator is

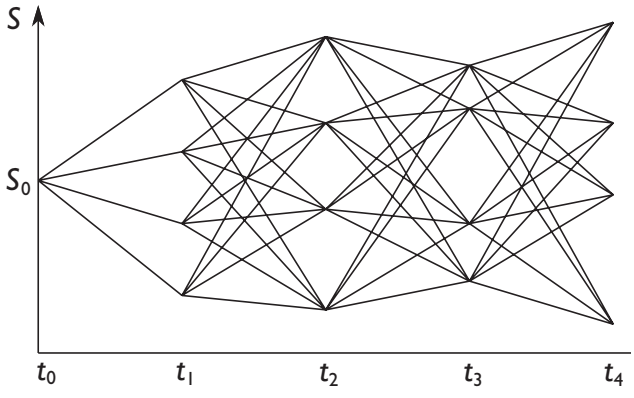
$$\hat{Q}(t, X_t(i)) = \max\left(h(t, X_t(i)), \frac{1}{b}\sum_{j=1}^{b} \hat{Q}\big(t+1, X_{t+1}(j)\big) w\big(t, X_t(i), X_{t+1}(j)\big)\right) \tag{4}$$

[3] See, eg, Karatzas (1988) for a justification of American option values as solutions to optimal stopping problems. Some authors restrict the term "American" to continuously exercisable securities and use the term "Bermudan" for securities that can be exercised on a finite number of dates. We consider only the latter, in some cases viewing it as an approximation to the former.

[4] The notation x^+ is short for $\max(x, 0)$.

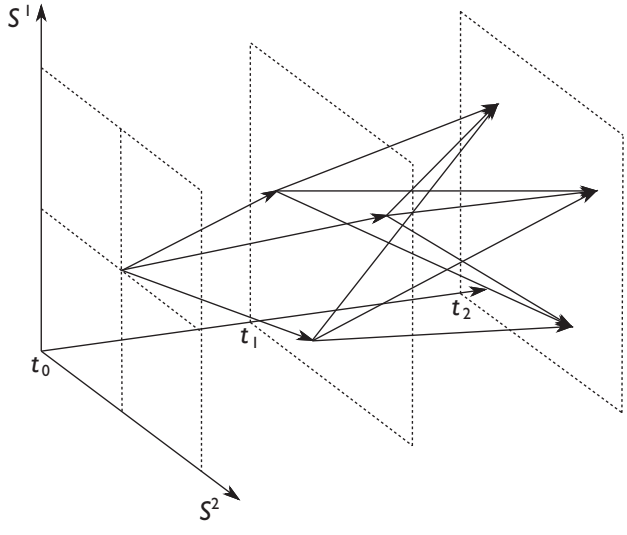
where $w(t, X_t(i), X_{t+1}(j))$ is a weight attached to the arc joining $X_t(i)$ to $X_{t+1}(j)$, which will be defined in a moment. We use the notation $\hat{Q}(t, X_t(i))$ to indicate the algorithm's estimate of the true American price $Q(t, X_t(i))$. At time $t = 0$ only $i = 1$ is applicable in equation (4) and $\hat{Q} \equiv \hat{Q}(0, S_0)$ is the final mesh estimator of the true price Q. Illustrations of the mesh are given in Figure 1 for $n = 1$, $T = 4$, and $b = 4$ and in Figure 2 for $n = 2$, $T = 2$, and $b = 3$.

FIGURE 1 Mesh illustrated for $n = 1$, $T = 4$, and $b = 4$.



A generic node in the mesh is denoted $X_t(i)$; a generic arc from one node to another has weight $w(t, X_t(i), X_{t+1}(k))$.

FIGURE 2 Mesh illustrated for $n = 2$, $T = 2$, and $b = 3$.



The arcs illustrate the calculation of the weighted average in (4).

In order to complete the description of the algorithm, we need to specify the details of how the random vectors are generated and how the weights on the arcs are determined. Suppose that conditional on $S_t = x$, S_{t+1} has density $f(t, x, \cdot)$ and let $f(t, \cdot)$ denote the marginal density of S_t (with S_0 fixed). In the simplest implementation, for $t = 1, \ldots, T$, the vectors $X_t(i)$, $i = 1, \ldots, b$, are generated as independent and identically distributed samples from some density function $g(t, \cdot)$. We require $g(t, u) > 0$ if $f(t-1, x, u) > 0$ for some x. The choices for the *mesh density functions* $g(t, \cdot)$ for $t = 1, \ldots, T$ are crucial to the practical success of the method. A seemingly natural choice is to set the mesh density functions to the *marginal density functions*, ie, to set $g(t, u) = f(t, u)$ for $t = 1, \ldots, T$. As shown in the next section, this choice can lead to estimators whose variance grows exponentially with the number of exercise opportunities. Another choice for the mesh density functions which avoids this problem is described in the next section.

In order to motivate the weights on the arcs, recall that the American option value at time t in state $S_t = x$ is

$$Q(t,x) = \max\Big(h(t,x), E\Big[Q(t+1, S_{t+1}) \,\Big|\, S_t = x\Big]\Big)$$

We need to approximate $Q(t, x)$ at all points $x = X_t(1), \ldots, X_t(b)$ using the available information from the mesh, ie, using $\hat{Q}(t+1, X_{t+1}(j))$ for $j = 1, \ldots, b$. To do this, we need to estimate all of the quantities $E[Q(t+1, S_{t+1}) \mid S_t = X_t(i)]$, $i = 1, \ldots, b$, using the same information $\hat{Q}(t+1, X_{t+1}(j))$, $j = 1, \ldots, b$. The main difficulty is that the density of S_{t+1} given $S_t = x$ is $f(t, x, \cdot)$ while the mesh points $X_{t+1}(j)$, $j = 1, \ldots, b$, were generated from the density function $g(t+1, \cdot)$. However, observe that

$$\begin{aligned} E\Big[Q(t+1, S_{t+1}) \,\Big|\, S_t = x\Big] &\equiv \int Q(t+1,u) f(t,x,u) \mathrm{d}u \\ &= \int Q(t+1,u) \frac{f(t,x,u)}{g(t+1,u)} g(t+1,u) \mathrm{d}u \\ &\equiv E\left[Q(t+1, X_{t+1}(j)) \frac{f(t,x,X_{t+1}(j))}{g(t+1, X_{t+1}(j))} \right] \end{aligned} \tag{5}$$

The final expression allows us to approximate the expectations $E[Q(t+1, S_{t+1}) \mid S_t = X_t(i)]$ for $i = 1, \ldots, b$, even though the points $X_{t+1}(j)$ for $j = 1, \ldots, b$ were generated according to the density $g(t+1, \cdot)$ and not according to $f(t, X_t(i), \cdot)$. Define

$$\hat{Q}(t,x) = \max\left(h(t,x), \frac{1}{b} \sum_{j=1}^{b} \hat{Q}(t+1, X_{t+1}(j))\, w(t,x,X_{t+1}(j)) \right) \tag{6}$$

where $w(t, \mathrm{x}, X_{t+1}(j)) = f(t, x, X_{t+1}(j)) / g(t+1, X_{t+1}(j))$. The mesh estimator

approximates $Q(t, X_t(i))$ by $\hat{Q}(t, X_t(i))$.[5]

The computational effort in generating the mesh is proportional to $b \times T$. The effort in the recursive pricing of equation (6) is proportional to $b^2 \times T$. Hence the overall effort is quadratic in the mesh parameter (b) and linear in the number of exercise opportunities ($T+1$).

We make the dependence of $\hat{Q}(0, S_0)$ on b explicit by denoting the mesh estimator $\hat{Q}_b(0, S_0)$. For any $b \geq 1$, the mesh estimator is an upper bound on the true price, ie, the bias of the mesh estimator is always positive:

THEOREM 1 (Mesh estimator bias) *The mesh estimator $\hat{Q}_b(0, S_0)$ is biased high, ie,*

$$E\left[\hat{Q}_b(0,S_0)\right] \geq Q(0,S_0)$$

for all b.

Theorem 1 can be proved using Jensen's inequality (in particular, $E[\max(a, Y)] \geq \max(a, E[Y])$) and an induction argument. Details are given in the Appendix.

In order to state the convergence result for the mesh estimator, we give some additional notation and assumptions. For $t = 1,\ldots,T$ and $k = 0, 1,\ldots,T-t$ define

$$R(t,t+k) = \left(\prod_{i=0}^{k-1} \frac{f\left(t+i, X_{t+i}(1), X_{t+i+1}(1)\right)}{g\left(t+i+1, X_{t+i+1}(1)\right)}\right) h\left(t+k, X_{t+k}(1)\right) \tag{7}$$

(where $\prod_{i=0}^{-1} \equiv 1$). We require three moment assumptions, stated below for some constants $r > p > 1$.

ASSUMPTION 1

$$E\left[\left(\frac{g(t_1, S_{t_1})}{f(t_1, S_{t_1})}\right) h\left(t_2, S_{t_2}\right)^r\right] < \infty$$

for all $t_2 = t_1,\ldots,T$.

ASSUMPTION 2

$$E\left[R^r(t_1,t_2)\right] < \infty$$

for all $t_2 = t_1,\ldots,T$.

[5] This choice of weights assumes that the transition density f of the underlying state variables is known or can be evaluated numerically. In practice, complicated diffusions are usually simulated using an Euler discretization (as described in, eg, Kloeden and Platen, 1999) with simpler transition densities approximating the true transition densities, and these can be used in the mesh. An alternative strategy for selecting weights that avoids densities entirely is proposed in Broadie, Glasserman, and Ha (2000).

ASSUMPTION 3

$$E\left[\left(\frac{f(t,x,X_{t+1}(1))}{g(t+1,X_{t+1}(1))}\right)^q\right]<\infty \tag{8}$$

for all x and $t=0,1,\ldots,T-1$, for all $q\geq 1$.

Assumptions 1–3 are usually difficult to verify in specific cases. But as they impose conditions solely on moments of payoffs, weights, and likelihood ratios, they do not appear unreasonable from a practical perspective.

Write $\|\cdot\|_p$ for the p-norm $E[(\cdot)^p]^{1/p}$ of a random variable. Convergence of the mesh estimator is given by:

THEOREM 2 (Mesh estimator convergence) *Let $r>p>1$. Under assumptions 1–3,*

$$\left\|\hat{Q}_b(t,x)-Q(t,x)\right\|\to 0$$

as $b\to\infty$, for all x and t.

Convergence in p-norm implies $\hat{Q}_b(0,S_0)$ converges to $Q(0,S_0)$ in probability and thus $\hat{Q}_b(0,S_0)$ is a consistent estimator of the option value. A consequence of this result is that

$$E\left[\hat{Q}_b(0,S_0)\right]\to Q(0,S_0)$$

as $b\to\infty$, so the mesh estimator is asymptotically unbiased.

2.1 Path estimator

Next we develop an estimator based on simulated paths which is biased low. By combining the high-biased mesh estimator with a low-biased path estimator, we can generate a valid confidence interval for the American option price. The path estimator is defined by simulating a trajectory of the underlying process S_t until the exercise region determined by the mesh is reached. Denote the simulated path by $S=(S_0,S_1,\ldots,S_T)$. The path S is simulated (independent of the mesh points $X_t(i)$) according to the density function of the process S_t, ie, the density of the simulated point S_{t+1} given $S_t=x$ is $f(t,x,\cdot)$. Along this path, the optimal policy exercises at $\tau^*(S)=\min\{t: h(t,S_t)\geq Q(t,S_t)\}$ for a payoff of $h(\tau^*,S_{\tau^*})$. The approximate optimal policy determined by the mesh exercises at

$$\hat{\tau}(S)=\min\left\{t: h(t,S_t)\geq \hat{Q}(t,S_t)\right\} \tag{9}$$

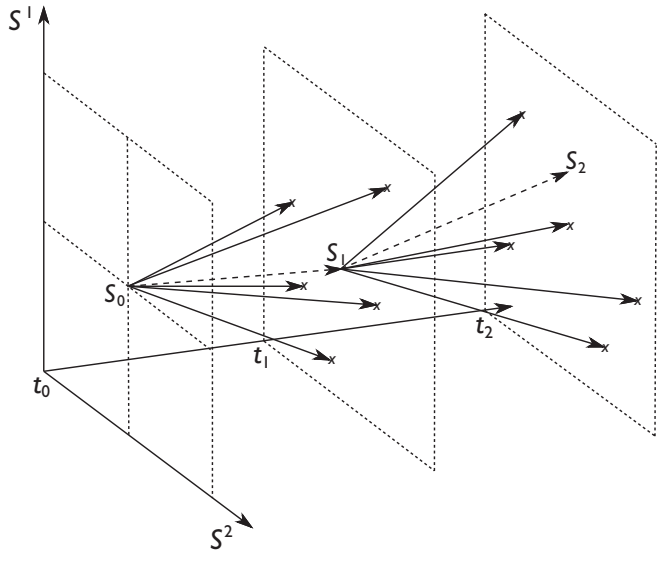
where $\hat{Q}(t,S_t)$ is given in equation (6). Define the *path estimator* by

$$\hat{q}=h\left(\hat{\tau},S_{\hat{\tau}}\right) \tag{10}$$

An illustration of the path estimator is given in Figure 3.

We make the dependence of $\hat{q}$ on b explicit by denoting the mesh policy $\hat{\tau}_b$

FIGURE 3 Path estimator illustrated for $n = 2$, $T = 2$, and $b = 5$.



Each mesh point is labeled with an 'x.' The simulated path $S = (S_0, S_1, S_2)$ is shown with dashed arrows. The solid arrows illustrate the points used in the computation of $\hat{Q}(t, S_t)$.

and the path estimator $\hat{q}_b = \hat{q}_b(\hat{\tau}_b)$. Since the stopping time $\hat{\tau}_b$ defined in (9) is not necessarily an optimal stopping time, an immediate consequence is that the path estimator is a lower bound on the true price:

THEOREM 3 (Path estimator bias) *The path estimator $\hat{q}_b$ is biased low, ie,*

$$E\left[\hat{q}_b\right] \le Q\left(0, S_0\right)$$

for all b.

Convergence of the path estimator is given by:

THEOREM 4 (Path estimator convergence) *Suppose the conditions in Theorem 2 are in effect and that $E[h(t, S_t)^{1+\varepsilon}] < \infty$ for all $t = 1, \ldots, T$, for some $\varepsilon > 0$. Suppose also that $P\left(h(t, S_t) = Q(t, S_t)\right) = 0$ for all $t = 0, 1, \ldots, T-1$. Then*

$$E\left[\hat{q}_b\right] \to Q\left(0, S_0\right)$$

as $b \to \infty$, ie, $\hat{q}_b$ is asymptotically unbiased.

Equation (9) shows that the mesh estimator must be computed before the path estimator. Once the mesh estimator has been computed, the additional effort to generate the path estimator is proportional to $n \times b \times T$. In our numerical implementation, we average the results from n_p independent paths to give the final path

estimator for each mesh. For the path and mesh estimators to have comparable variances, we take n_p proportional to b. Hence, the overall work associated with the path estimator is proportional to $n \times b^2 \times T$, the same as the mesh estimator.

2.2 Interval estimation

In order to give a confidence interval for the option price Q, generate N independent meshes with corresponding mesh estimates $\hat{Q}^{(i)} = \hat{Q}_b^{(i)}(0, S_0)$, $i = 1, \ldots, N$, and then combine them to give

$$\bar{Q}(N) = \frac{1}{N} \sum_{i=1}^{N} \hat{Q}^{(i)}$$

For each mesh i, $i = 1, \ldots, N$, generate n_p independent paths and corresponding path estimates. Average these individual estimates to give the path estimates $\hat{q}^{(i)} = \hat{q}_b^{(i)}(0, S_0)$, $i = 1, \ldots, N$.[6] These N path estimates, each based on n_p paths, are combined to give

$$\bar{q}(N) = \frac{1}{N} \sum_{i=1}^{N} \hat{q}^{(i)}$$

With $\bar{Q}(N)$ and $\bar{q}(N)$ replacing $\hat{Q}_b$ and $\hat{q}_b$, respectively, Theorems 1–4 hold for any $N \geq 1$. Finally, form the confidence interval

$$\left[\bar{q}(N) - z_{\alpha/2} \frac{s(\hat{q})}{\sqrt{N}},\ \bar{Q}(N) + z_{\alpha/2} \frac{s(\hat{Q})}{\sqrt{N}} \right] \tag{11}$$

where $z_{\alpha/2}$ is the $1 - \alpha/2$ quantile of the standard normal distribution, and $s(\hat{q})$ and $s(\hat{Q})$ are the sample standard deviations of $\hat{q}$ and $\hat{Q}$, respectively.[7] Theorems 1 and 3 show that taking the lower confidence limit from the path estimator together with the upper confidence limit from the mesh estimator as indicated in (11) yields a valid $100(1 - \alpha)\%$ confidence interval for Q. In fact, the expected coverage of the interval will exceed the nominal coverage of $(1 - \alpha)$, depending on the extent of the bias in the estimators, ie, the interval in (11) is *conservative*.

3 Selection of the mesh density

As described in the previous section, the stochastic mesh method leaves a lot of latitude in implementation. For the method to be practically viable, it is essential to exploit efficiencies in the computation of the estimators wherever possible. This

[6] It is convenient, though not necessary, for n_p to be a constant independent of the mesh. Likewise, it is convenient to have the same number of mesh and path estimates.

[7] This implicitly assumes that the estimators have finite second moments. Increasing the exponents in Assumptions 1–3 by one more than suffices to ensure this for $\hat{Q}$; requiring $E[h^2(t, S_t)] < \infty$ for all t ensures it for $\hat{q}$.

requires, in particular, careful choice of the density used to generate the mesh. It also motivates the use of control variates, a topic discussed in the next section.

In order to illustrate the impact that the mesh density function can have on the mesh estimator variance, consider pricing a European option on the stochastic mesh. Since early exercise is not allowed, the mesh estimator of the European option price from equation (6) simplifies to

$$\hat{Q}\big(t, X_t(i)\big) = \frac{1}{b}\sum_{j=1}^{b} \hat{Q}\big(t+1, X_{t+1}(j)\big)\frac{f\big(t, X_t(i), X_{t+1}(j)\big)}{g\big(t+1, X_{t+1}(j)\big)}$$

with $\hat{Q}\big(T, X_T(i)\big) = h\big(T, X_T(i)\big)$ as before. For ease of notation, we consider the case $T = 3$ and see that $\hat{Q}(0, S_0)$ can be written as

$$\begin{aligned}
&\frac{1}{b}\sum_{j_1=1}^{b} \frac{f\big(0, S_0, X_1(j_1)\big)}{g\big(1, X_1(j_1)\big)} \hat{Q}\big(1, X_1(j_1)\big) \\
&= \frac{1}{b}\sum_{j_1=1}^{b} \frac{f\big(0, S_0, X_1(j_1)\big)}{g\big(1, X_1(j_1)\big)} \left[\frac{1}{b}\sum_{j_2=1}^{b} \frac{f\big(1, X_1(j_1), X_2(j_2)\big)}{g\big(2, X_2(j_2)\big)} \hat{Q}\big(2, X_2(j_2)\big) \right] \\
&= \frac{1}{b}\sum_{j_1=1}^{b} \frac{f\big(0, S_0, X_1(j_1)\big)}{g\big(1, X_1(j_1)\big)} \left[\frac{1}{b}\sum_{j_2=1}^{b} \frac{f\big(1, X_1(j_1), X_2(j_2)\big)}{g\big(2, X_2(j_2)\big)} \right. \\
&\qquad \left. \times \left[\frac{1}{b}\sum_{j_3=1}^{b} \frac{f\big(2, X_2(j_2), X_T(j_3)\big)}{g\big(T, X_T(j_3)\big)} h\big(T, X_T(j_3)\big) \right] \right] \\
&= \frac{1}{b}\sum_{j_3=1}^{b} h\big(T, X_T(j_3)\big) \left[\frac{1}{b}\sum_{j_2=1}^{b} \frac{f\big(2, X_2(j_2), X_T(j_3)\big)}{g\big(T, X_T(j_3)\big)} \right. \\
&\qquad \left. \times \left[\frac{1}{b}\sum_{j_1=1}^{b} \frac{f\big(1, X_1(j_1), X_2(j_2)\big)}{g\big(2, X_2(j_2)\big)} \frac{f\big(0, S_0, X_1(j_1)\big)}{g\big(1, X_1(j_1)\big)} \right] \right] \qquad (12)
\end{aligned}$$

The last equality shows that the mesh estimator is simply a linear combination of the terminal payoffs. Generalizing the previous expression for arbitrary T and simplifying, the mesh estimator of the European value can be written as $\hat{Q}(0, S_0) = (1/b)\sum_{j_T=1}^{b} h\big(T, X_T(j_T)\big) L(T, j_T)$, where the coefficients $L(T, j_T)$ are given by

$$L(T, j_T) = \frac{1}{b^{T-1}} \sum_{j_1, \ldots, j_{T-1}}^{b} \left(\prod_{i=1}^{T} \frac{f\big(i-1, X_{i-1}(j_{i-1}), X_i(j_i)\big)}{g\big(i, X_i(j_i)\big)} \right) \qquad (13)$$

with the convention $X_0(j_0) \equiv S_0$. Thus the likelihood ratio $L(T, j_T)$ can be interpreted as the weight associated with the j_Tth terminal point in the mesh.

It is natural to expect that the main contribution to the variance of the estimator

$\hat{Q}(0, S_0)$ comes from the likelihood ratio multiplying the payoff function, rather than the payoff function itself. We therefore analyze the variance of $L(T, j)$ (for fixed $b > 1$). Because the points in the mesh at each time slice are identically distributed, $L(T, j), j = 1, \ldots, b$, are identically distributed, though not independent. To simplify notation, we write $L(T)$ for $L(T, 1)$ (or any other $L(T, j)$ with fixed j). For all T, $E[L(T)] = 1$. However, we will now argue that the variance of each $L(T, j)$ often grows exponentially in T.
Observe that

$$E\left[\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\right] = \int \frac{f(t,x,y)}{g(t+1,y)} g(t+1,y)\,\mathrm{d}y$$
$$= \int f(t,x,y)\,\mathrm{d}y = 1$$

for all x. Unless $f(t, x, X_{t+1}(j)) = g(t+1, X_{t+1}(j))$ with probability 1, the strict form of Jensen's inequality gives

$$E\left[\left(\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\right)^2\right] > \left(E\left[\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\right]\right)^2 = 1$$

An additional condition that we now impose is that this strict inequality hold uniformly in x and t. We also require that likelihood ratios involving the same mesh point $X_{t+1}(1)$ at time $t+1$ but different mesh points at time t be positively correlated.

PROPOSITION 1 (Variance build-up) *Suppose that $b > 1$, that*

$$\inf_{t=0,1,\ldots} \inf_{x} E\left[\left(\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\right)^2\right] > 1 \tag{14}$$

and that

$$E\left[\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\frac{f\left(t,y,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\right] \geq 1 \tag{15}$$

for all t, x, and y. Then there is an $a > 0$ and $\lambda > 1$ (both possibly depending on b) for which

$$\mathrm{var}\left[L(t+1)\right] \geq a\lambda^t \tag{16}$$

for all sufficiently large t.

Remark: Replacing the lower bound in the proposition with $a\lambda^t - 1$ makes the inequality valid for all t.

Whether or not the conditions of this proposition hold may be difficult to determine for specific choices of g. However, the importance of the result lies in showing that if the mesh density is not chosen carefully there is a risk of an exponential growth in variance. The *average density method* defined below is significant because it eliminates this risk. Indeed, it reduces the potentially exponential variance of the $L(T, j)$ to zero!

As noted above, Proposition 1 suggests that for the stochastic mesh to be practically viable, the distributions used to sample the mesh points must be chosen carefully to avoid exponential growth in variance. Fortunately, by inspecting equation (12) or (13), we see that the coefficients $L(T, j)$, $j = 1,\ldots,b$, will be constant (and equal to one) if we choose

$$g(t,u) = f(0, S_0, u) \quad \text{for} \quad t = 1 \tag{17}$$

and

$$g(t,u) = \frac{1}{b}\sum_{j=1}^{b} f\left(t-1, X_{t-1}(j), u\right) \quad \text{for} \quad t = 2,\ldots,T \tag{18}$$

We refer to the mesh density functions in equations (17) and (18) as the *average density functions*. A mesh generated with the average density function has the attractive feature that the estimate it provides of the European value of an option is simply the average of the terminal payoffs:

PROPOSITION 2 *Using the average density function, each* $L(T, j)$ *is identically equal to* 1. *Consequently, the mesh estimate of a European option price is*

$$\frac{1}{b}\sum_{i=1}^{b} h\left(T, X_T(j)\right)$$

and each $X_T(j)$ *has the distribution of* S_T.

Taken together, Propositions 1 and 2 show that judicious choice of mesh density can have an enormous impact on the performance of the method.

Using the average density method to generate the mesh can be interpreted in the following way. Suppose that from each of the mesh nodes $X_{t-1}(j)$, $j = 1,\ldots,b$, we generate exactly one successor $X_t(j)$ from the underlying transition density $f(t-1, X_{t-1}(j), \cdot)$. If we then draw a value randomly and uniformly from $\{X_t(1),\ldots,X_t(b)\}$, the value drawn is distributed according to the average density $g(t, \cdot)$ in (18), conditional on $\{X_{t-1}(1),\ldots,X_{t-1}(b)\}$. Using the average density is thus equivalent to generating b independent paths of the underlying and then "forgetting" which nodes were on which paths.

Taking this observation one step further leads to the following implementation: simulate b independent paths $\left(X_0(i),\ldots,X_T(i)\right)$, $i = 1,\ldots,b$, as in an ordinary simulation and then apply the weight

$$\frac{f\left(t-1, X_{t-1}(i), X_t(j)\right)}{b^{-1}\sum_{k=1}^{b} f\left(t-1, X_{t-1}(k), X_t(j)\right)}$$

to the transition from $X_{t-1}(i)$ on the ith path to $X_t(j)$ on the jth path. These weights define the mesh; recall equation (6). Since this construction generates exactly one successor from each of the b transition densities $f(t-1, X_{t-1}(i), \cdot)$, $i = 1, \ldots, b$, it may be viewed as a *stratified* implementation of the average mesh density. This is the construction we use in our numerical experiments.

The idea of simulating independent paths and then interconnecting them with weights in order to apply dynamic programming is also implicit in the methods of Longstaff and Schwartz (2001) and Tsitsiklis and Van Roy (1999); their weights are produced implicitly by a least-squares procedure. Thus, although arrived at by a different argument, those methods may be viewed as stochastic mesh methods with different choices for weights.

4 Algorithm enhancements

This section describes enhancements to the basic mesh algorithm that can substantially improve its efficiency. We first explain the use of control variates with the mesh estimate and then enhancements to the path estimator.

4.1 Control variates with the mesh estimator

We detail two applications of control variates for improving the mesh estimator. In the first application, control variates are used to improve the estimates $\hat{Q}(t, X_t(i))$ of the option value at each mesh point. These are called the *inner* controls, because they are applied within each mesh. We also use control variates to improve the mesh estimates $\bar{Q}(N)$. These are called the *outer* controls because they are applied after the N individual mesh estimates at time $t = 0$ in state S_0 are computed.

We begin by describing the inner controls. From equation (6), the mesh estimate $\hat{Q}(t, X_t(i))$ depends on the continuation value

$$C(t, i) \equiv E\left[Q(t+1, S_{t+1}) \mid S_t = X_t(i)\right]$$

which is estimated by

$$\frac{1}{b}\sum_{j=1}^{b} \hat{Q}\left(t+1, X_{t+1}(j)\right) w\left(t, X_t(i), X_{t+1}(j)\right)$$

Suppose that there is a known formula for $v = E\left[v(t+1, S_{t+1}) \mid S_t = X_t(i)\right]$ or that an accurate numerical estimate of v can be obtained very quickly. For example, v could represent the expected future value of the first underlying asset, $E\left[S^1_{t+1} \mid S_t = X_t(i)\right]$, or it could represent the value of the related European option, $E\left[h(t+1, S_{t+1}) \mid S_t = X_t(i)\right]$. We can also construct the mesh estimate of v:

$$\hat{v} = \frac{1}{b}\sum_{j=1}^{b} v\big(t+1, X_{t+1}(j)\big) w\big(t, X_t(i), X_{t+1}(j)\big)$$

By the argument leading to equation (5), it follows that $E[\hat{v}] = v$. Information about the known error, $\hat{v} - v$, can be used to reduce the unknown error in the estimate of the continuation value. However, the presence of the weights complicates the procedure. We use the controlled estimator of the continuation value $C(t, i)$ defined by

$$\left[\frac{1}{b}\sum_{j=1}^{b} \hat{Q}\big(t+1, X_{t+1}(j)\big) w(t,i,j) - \beta\left(\frac{1}{b}\sum_{j=1}^{b} v\big(t+1, X_{t+1}(j)\big) w(t,i,j) - v\frac{1}{b}\sum_{j=1}^{b} w(t,i,j)\right)\right] \Bigg/ \frac{1}{b}\sum_{j=1}^{b} w(t,i,j) \tag{19}$$

where the notation $w(t, i, j)$ is short for $w\big(t, X_t(i), X_{t+1}(j)\big)$. This expression can be explained in several ways. First, note that the term in the numerator, $\sum_{j=1}^{b} v\big(t+1, X_{t+1}(j)\big) w(t, i, j)/b - v\sum_{j=1}^{b} w(t, i, j)/b$ has expectation zero. If β is positive, then the estimate of the continuation value will be decreased if $\sum_{j=1}^{b} v\big(t+1, X_{t+1}(j)\big) w(t, i, j)/b > v\sum_{j=1}^{b} w(t, i, j)/b$, and will be increased otherwise. Second, the denominator has expectation one, and if the average of the weights, $\sum_{j=1}^{b} w(t, i, j)/b$, is greater than one, the estimate of the continuation value will be deflated by this amount (or inflated by the corresponding amount if the average is less than one). Thus, the denominator in (19) also acts like a control variable. We choose β to solve the weighted least-squares problem:

$$\min_{\alpha,\beta} \frac{1}{b}\sum_{j=1}^{b} w(t,i,j)\Big[\hat{Q}\big(t+1, X_{t+1}(j)\big) - \big(\alpha + \beta v\big(t+1, X_{t+1}(j)\big)\big)\Big]^2$$

With this choice for β, it can be shown that the controlled estimator in (19) simplifies to $\alpha + \beta v$.[8]

[8] In contrast, consider the usual (unweighted) control variate procedure. Suppose we want to estimate $E(Y)$ and we know that the random variable X has expectation x. Given a sample (X_j, Y_j), $j = 1, \ldots, b$, the usual procedure is to form the controlled estimator

$$\frac{1}{b}\sum_{j=1}^{b} Y_j - \beta\left(\frac{1}{b}\sum_{j=1}^{b} X_j - x\right), \quad \text{where } \beta \text{ is chosen to solve } \min_{\alpha,\beta} \frac{1}{b}\sum_{j=1}^{b}\big[Y_j - (\alpha + \beta X_j)\big]^2.$$

In this case, the controlled estimator simplifies to $\alpha + \beta x$. The effectiveness of this procedure depends on the correlation of X and Y. In the case with weights above, we could follow the usual procedure with the identification $X_j = \hat{Q}\big(t+1, X_{t+1}(j)\big) w(t, i, j)$ and $Y_j = v\big(t+1, X_{t+1}(j)\big) w\big(t, X_t(i), X_{t+1}(j)\big)$. However, the effectiveness of the procedure depends on the correlation of the weighted products $\hat{Q}\big(t+1, X_{t+1}(j)\big) w(t, i, j)$ and $v\big(t+1, X_{t+1}(j)\big) \times w\big(t, X_t(i), X_{t+1}(j)\big)$. It is usually easier to find a control $v\big(t+1, X_{t+1}(j)\big)$ that is correlated with $\hat{Q}\big(t+1, X_{t+1}(j)\big)$, and that is the reason for the procedure described in the text.

The outer controls are fairly standard. We use N independent meshes to generate estimates $\hat{Q}^{(i)}$, $i = 1,\ldots,N$, of the option price, $Q = Q(0, S_0)$. Suppose that quantity $u = u(0, S_0)$ is known in closed form or can be quickly computed. For example, u might represent the European option value $E[h(T, S_T)]$. We then use each mesh to generate unbiased estimates $\hat{u}^{(i)}$, $i = 1,\ldots,N$, of u, using, for example, equation (12) or (13). Then we form the controlled estimator of Q:

$$\frac{1}{N}\sum_{i=1}^{N}\hat{Q}^{(i)} - \beta\left(\frac{1}{N}\sum_{i=1}^{N}\hat{u}^{(i)} - u\right) \tag{20}$$

Sometimes it will be useful to use multiple controls, $u_1,\ldots,u_K$, giving the analogous controlled estimator:

$$\frac{1}{N}\sum_{i=1}^{N}\hat{Q}^{(i)} - \sum_{k=1}^{K}\beta_k\left(\frac{1}{N}\sum_{i=1}^{N}\hat{u}_k^{(i)} - u_k\right) \tag{21}$$

The coefficients β_k can be estimated by solving a least-squares problem or the equivalent multiple linear regression problem.

4.2 Path estimator enhancements

We briefly describe three techniques that can be used to improve the path estimator: (i) control variates, (ii) antithetics, and (iii) policy fixing. In determining whether to stop or continue, the path estimator compares the exercise value $h(t, S_t)$ with the estimated continuation value $\hat{Q}(t, S_t)$. The latter estimate can be improved using inner controls exactly as described for the mesh estimator. Similarly, outer controls can be used to improve the n_p independent path estimates in each mesh. However, since the path estimator stops at a random time, we use controls that stop at the same random time. The controlled path estimator is given by equation (20) or (21), with $\hat{Q}^{(i)}$ replaced by $\hat{q}^{(i)}$.

The use of antithetic variates with the path estimator is fairly standard. For each simulated path $S = (S_0, S_1,\ldots,S_T)$ we also generate an antithetic path $S' = (S_0', S_1',\ldots,S_T')$. For example, if the original path is driven by standard normal increments, then the antithetic path is driven by the negative of the normal increments. The two option estimates, which in general involve different stopping times, are then averaged to give the path estimate. When controls are used, they are computed in the same way for the antithetic paths. More detailed discussion of the antithetic technique is given in Boyle, Broadie, and Glasserman (1997).

The path estimator stops at the first time at which the exercise value equals or exceeds the estimated continuation value, ie, when $h(t, S_t) \geq \hat{Q}(t, S_t)$. Bias is introduced whenever the estimator stops earlier or later than is optimal. Suppose that we have an easily computed lower bound $\underline{P}(t, S_t)$ on the option price $Q(t, S_t)$, ie, $Q(t, S_t) \geq \underline{P}(t, S_t)$. Then, if $\underline{P}(t, S_t) > h(t, S_t)$ it must be that the optimal decision is to continue. In this case there is no need to even compute $\hat{Q}(t, S_t)$. This saves computation time and reduces bias, since there is some possibility that

$h(t, S_t) \geq \hat{Q}(t, S_t)$ and the original path estimator would stop when it is not optimal to do so. We call this enhancement *policy fixing*, since it uses the lower bound $\underline{P}(t, S_t)$ to set the exercise policy where possible.[9]

5 Computational results

In this section we first give numerical examples to illustrate the degree of variance reduction possible with the estimator enhancements described in the previous section. Then we test the stochastic mesh method on two types of high-dimensional options. These numerical results illustrate the bias and convergence results of Theorems 1–4, illustrate the convergence rate of the method, and also demonstrate the practical viability of the method.

5.1 Comparison of mesh estimator variance with two mesh density functions

We illustrate that the theoretical variance build-up described in Proposition 1 has severe practical implications. We examine the impact of the two different choices for the mesh density functions in a particular example. We compare the marginal density functions (ie, $g(t, u) = f(t, u)$, for $t = 1, \ldots, T$) with the average density functions (given in equations (17) and (18)). For simplicity, consider pricing a European call option on a single asset under the usual Black–Scholes assumptions. That is, the risk-neutral process for the underlying asset S_t satisfies

$$\mathrm{d}S_t = S_t\left[(r-\delta)\mathrm{d}t + \sigma \mathrm{d}z_t\right] \tag{22}$$

where z_t is a standard Brownian motion process, r is the riskless interest rate, δ is the dividend rate, and $\sigma > 0$ is the volatility parameter. Under the risk-neutral measure, $\ln(S_{t_i}/S_{t_{i-1}})$ is normally distributed with mean $(r - \delta - \sigma^2/2)(t_i - t_{i-1})$ and variance $\sigma^2(t_i - t_{i-1})$. In the example, we set $r = 3\%$, $\delta = 10\%$, $\sigma = 30\%$, and $S_0 = 100$.[10] The call option payoff is $h(T, S_T) = (S_T - K)^+$. With $K = 100$ and an expiration of three years, the European option value is 0.777.

In order to keep the European option value constant, in Table 1 we fix the maturity of the option at three years while increasing the number of exercise opportunities, denoted by d in the table.[11] Consistent with the insights of Propositions 1 and 2, Table 1 shows that the difference between the two choices

[9] We could also use policy fixing to determine when to stop. For example, suppose that we have an easily computed upper bound $\underline{P}(t, S_t)$ on the option price $Q(t, S_t)$, ie, $Q(t, S_t) \leq \underline{P}(t, S_t)$. Then if $\underline{P}(t, S_t) < h(t, S_t)$ it must be that the optimal decision is to stop. Again, this eliminates the need to compute $\hat{Q}(t, S_t)$ and it reduces bias as well. However, in most of our applications, it seems to be difficult to determine easily computed and relatively tight upper bounds on the option price. A similar policy fixing idea could be applied to the mesh estimator as well.

[10] A large dividend rate could arise with foreign currency options, where r represents the domestic interest rate and δ the foreign interest rate.

[11] Similar results are obtained if we let both the number of time steps and the maturity increase, as in Proposition 1.

TABLE 1 Comparison of mesh estimator variance with two mesh density functions.

	European estimator variance		American estimator variance	
d	Marginal density function	Average density function	Marginal density function	Average density function
2	(1.1, 1.5)	(0.54, 0.55)	(1.1, 2.0)	(0.7, 0.7)
4	(3.2, 115.0)	(0.54, 0.55)	(6.9, 305.4)	(0.7, 0.7)
8	(13.2, 540.1)	(0.55, 0.55)	(93.3, 6366.9)	(0.7, 0.7)
16	N/A	(0.55, 0.55)	N/A	(0.8, 0.8)
32	N/A	(0.55, 0.55)	N/A	(1.1, 1.1)
64	N/A	(0.55, 0.55)	N/A	(1.8, 1.8)
128	N/A	(0.55, 0.55)	N/A	(3.0, 3.0)

The call option parameters are $r = 3\%$, $\delta = 10\%$, $\sigma = 30\%$, $S_0 = 100$, $K = 100$ with an expiration of $T_{\text{mat}} = 3$ years. All results are based on a mesh parameter of $b = 20$. Equal time steps are used, with exercise opportunities at iT_{mat}/d, i = 0, 1, … , d. The variance is estimated by taking the sample variance of 100,000 independent replications of the mesh estimators. Because the error in the variance estimates is so large in some cases, the process was repeated seven times. In the notation (x, y) used in the table, x represents the minimum and y the maximum of the seven variance estimates.

of mesh density functions can be enormous. For the European case, the variance of the mesh estimator with the marginal density function is too large for practical computations with d as small as four. The variance with the average density function is independent of d (the only contribution is from the variance of the payoff function). In the American case, we allow exercise at each of the time steps iT_{mat}/d, $i = 0, \ldots, d$, with $T_{\text{mat}} = 3$ years. The variance in the American case is greater for both mesh density functions. However, the growth in variance with the average density function is slow enough to be practical for large values of d. In all of the numerical results that follow, we use the average density function as the mesh density function.

5.2 Comparison of mesh estimator variance with various inner and outer controls

Control variates can be a powerful tool for reducing estimator variance, but the choice of good control variates is an art – the best choices are problem specific. In order to illustrate the type of process one might follow, we pick a particular example and examine several choices for inner and outer controls. We consider pricing an American call option on the maximum of five assets under the usual Black–Scholes assumptions. The payoff upon exercise of this max-option is $h(t, S_t) = \left(\max(S_t^1, \ldots, S_t^5) - K\right)^+$. Under the risk-neutral measure asset prices are assumed to follow correlated geometric Brownian motion processes, ie,

$$\mathrm{d}S_t^i = S_t^i\left[(r - \delta_i)\mathrm{d}t + \sigma_i \mathrm{d}z_t^i\right] \tag{23}$$

where z_t^i is a standard Brownian motion process and the instantaneous correlation of z^i and z^j is ρ_{ij}. For simplicity, in our numerical results we take $\delta_i = \delta$ and $\rho_{ij} = \rho$

for all $i, j = 1,\ldots,k$ and $i \neq j$. We allow exercise at equally spaced dates.

We test three inner controls and two outer controls. The first inner control we test is

$$v^{(1)} = E\left[\mathrm{e}^{-r\Delta t} \left(S_{t+1}^{i^*} - K \right)^+ \,\middle|\, S_t = X_t(i) \right] \tag{24}$$

where $i^* = \text{argmax}\{S_t^i,\ i = 1,\ldots,5\}$. That is, the first control is a European option on a single asset with a time to maturity of Δt, and $v^{(1)}$ is easily evaluated using the Black–Scholes formula. The second inner control we test is

$$v^{(2)} = E\left[S_{t+1}^{i^*} \,\middle|\, S_t = X_t(i) \right] = S_t^{i^*} \mathrm{e}^{(r-\delta)\Delta t} \tag{25}$$

where $i^* = \text{argmax}\{S_t^i,\ i = 1,\ldots,5\}$ as before. The largest underlying asset at the mesh point $X_t(i)$ is used as the second control. The third inner control is

$$v^{(3)} = E\left[\mathrm{e}^{-r\Delta t} \left(\max\left(S_{t+1}^{i^*}, S_{t+1}^{j^*} \right) - K \right)^+ \,\middle|\, S_t = X_t(i) \right] \tag{26}$$

where $i^* = \text{argmax}\{S_t^i,\ i = 1,\ldots,5\}$ and $j^* = \text{argmax}\{S_t^i,\ i = 1,\ldots,5,\ i \neq i^*\}$. Thus, the third control is a European max-option on two assets with a time to maturity of Δt. Note that $v^{(3)}$ is easily evaluated using the formula in Stulz (1982). The first outer control we test is the European max-option

$$u^{(1)} = E\left[\mathrm{e}^{-rt} \left(\max\left(S_T^1, \ldots, S_T^5 \right) - K \right)^+ \,\middle|\, S_0 \right] \tag{27}$$

A formula for this value is given in Johnson (1987). Quasi Monte Carlo methods can be used to evaluate $u^{(1)}$ quickly and accurately. In particular, we use the low discrepancy Sobol' sequence for this purpose. See Boyle, Broadie, and Glasserman (1997) for an overview and Bratley and Fox (1998) or Press *et al* (1992) for implementation details. For the second outer control, $u^{(2)}$, we replace T by $2T/3$ in equation (27). When working backwards through the mesh, we found that better estimates are obtained by using the inner control as indicated in equation (19) to compute both the American price estimate $\hat{Q}$ and the mesh estimates of the outer controls $\hat{u}^{(1)}$ and $\hat{u}^{(2)}$.

The results in Table 2 show that considerable reductions in variance are possible using control variates. The relative magnitudes of the variances are important for comparing various controls; the absolute levels are often difficult to interpret. Inner control 3, ie, $v^{(3)}$, consistently outperforms inner controls 1 and 2. The best combination tested is inner control 3 together with the two outer controls. This combination reduces estimator variance by about a factor of 100. Even this impressive figure understates the true gains in performance, because the inner controls also reduce estimator bias. Including controls increases computation time, typically by a factor of two to three, but the estimator improvement far outweighs this increased computational effort.

TABLE 2 Comparison of mesh estimator variance with various inner and outer controls.

	No	Inner controls			Inner + 1 Outer			Inner + 2 Outer		
S	control	1	2	3	1	2	3	1	2	3
90	3.55	1.22	1.31	0.91	0.17	0.21	0.06	0.08	0.09	0.03
100	5.06	1.85	1.94	1.47	0.24	0.28	0.10	0.10	0.11	0.05
110	6.93	2.53	2.62	2.08	0.35	0.37	0.16	0.14	0.14	0.07

Max-option example with $n = 5$ assets. The parameters are $r = 5\%$, $\delta = 10\%$, $\sigma = 20\%$, $\rho = 0$, $K = 100$, and three-year maturity. The initial vector is $S_0 = (S, \ldots, S)$, with $S = 90$, 100, or 110 as indicated in the table. All results are based on a mesh parameter of $b = 100$. Equal time steps are used, with exercise opportunities at $t = 0$, 1, 2, and 3 years. The variance is estimated by taking the sample variance of 10,000 independent replications of the mesh estimators. Inner controls $v^{(1)}$, $v^{(2)}$, $v^{(3)}$ and outer controls $u^{(1)}$ and $u^{(2)}$ are defined in the text. Column 2 under the heading "Inner + 1 Outer" refers to using inner control $v^{(2)}$ together with outer control $u^{(1)}$, column 3 under the heading "Inner + 2 Outer" refers to using inner control $v^{(3)}$ together with outer controls $u^{(1)}$ and $u^{(2)}$, etc.

5.3 Comparison of path estimator enhancements

We continue with the previous max-option example on five assets to illustrate the process of evaluating path estimator enhancements. Based on the previous experiment, we use the inner control $v^{(3)}$ defined in equation (26) for all path estimator tests. For outer controls, we test the geometric average control[12]

$$w^{(1)} = E\left[e^{-c\tau} \left(S_\tau^1 S_\tau^2 \cdots S_\tau^n \right)^{(1/n)} \middle| S_0 \right] = \left(S_0^1 S_0^2 \cdot S_0^n \right)^{(1/n)} \tag{28}$$

We also test the underlying asset controls

$$w^{(2)}(i) = E\left[e^{(-r+\delta)\tau} S_\tau^i \middle| S_0 \right] = S_0^i \tag{29}$$

for $i = 1, \ldots, n$.

Table 3 shows the results using various combinations of outer controls and antithetics. As before, while it is difficult to interpret the absolute estimator variance levels, the relative differences in the table show that the path estimator variance can be reduced by a factor of 10 to 20. The largest gains are achieved by using both types of outer controls in combination with antithetics. The improvements in variance are easily worth the additional computational effort associated with the controls and antithetics.

In order to test the policy fixing technique, we use three easily computed lower

[12] The constant c which makes equation (28) hold is

$$c = -r + \delta + \sum_{i=1}^{n} \sigma_i^2 / (2n) + \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_i \sigma_j \rho_{ij} / (2n^2)$$

TABLE 3 Comparison of path estimator variance with several variance reduction techniques

S	No Inner control	Outer control			Antithetic + Outer control		
		1	2	3	1	2	3
90	295	265	149	64	118	61	23
100	375	335	171	67	173	91	25
110	530	469	223	79	190	111	24

Max-option with $n = 5$ assets. The parameters are $r = 5\%$, $\delta = 10\%$, $\sigma = 20\%$, $\rho = 0$, $K = 100$, and three-year maturity. The initial vector is $S_0 = (S, \ldots, S)$, with $S = 90$, 100, or 110 as indicated in the table. All results are based on a mesh parameter of $b = 20$. Equal time steps are used, with exercise opportunities at $t = 0$, 1, 2, and 3 years. The variance is estimated by taking the sample variance of 100,000 independent replications of the path estimators. All results use the inner control $v^{(3)}$. For the outer controls, column 1 refers to the geometric control $w^{(1)}$, column 2 refers to using the five underlying assets as multiple controls, and column 3 refers to both types of controls.

bounds on the continuation value. The first is the trivial nonnegativity bound, ie, $\underline{P}(t, S_t)^{(1)} = 0$. The second is the European option value on a single asset with a time to maturity $T - t$. Thus,

$$\underline{P}^{(2)}\left(t, S_t\right) = E\left[\mathrm{e}^{-r(T-t)}\left(S_T^{i^*} - K\right)^+ \,\middle|\, S_t\right] \tag{30}$$

where $i^* = \operatorname{argmax}\{S_t^i, i = 1, \ldots, 5\}$. The third is a European max-option on two assets with a time to maturity of $T - t$:

$$\underline{P}^{(3)}\left(t, S_t\right) = E\left[\mathrm{e}^{-r(T-t)}\left(\max\left(S_T^{i^*}, S_T^{j^*}\right) - K\right)^+ \,\middle|\, S_t\right] \tag{31}$$

where i^* is as before and $j^* = \operatorname{argmax}\{S_t^i, i = 1, \ldots, 5, i \neq i^*\}$. The three bounds satisfy $\underline{P}^{(1)}(t, S_t) \leq \underline{P}^{(2)}(t, S_t) \leq \underline{P}^{(3)}(t, S_t) \leq Q(t, S_t)$. We first check if $\underline{P}(t, S_t)^{(1)} \geq h(t, S_t)$. If so, we know it is optimal to continue. Otherwise we check if $\underline{P}(t, S_t)^{(2)} \geq h(t, S_t)$. If so, we know it is optimal to continue, and otherwise we check if $\underline{P}(t, S_t)^{(3)} \geq h(t, S_t)$. The same numerical results would be obtained if we simply used to the tightest bound $\underline{P}(t, S_t)^{(3)}$. However, the three bounds are progressively more difficult to compute, so checking the bounds in order typically saves computation time. We measured the computation time corresponding to the last column in Table 3 with and without policy fixing. The computation time ratios were 39%, 58%, and 86%, corresponding to the rows $S = 90$, 100, and 110, respectively. In addition to the computation time savings, policy fixing also reduces the path estimator bias.

5.4 Option pricing results

Next we give numerical results with the stochastic mesh method based on two types of options. The first type is the max-option on five assets described earlier. The second type is the geometric average option on five assets. The payoff upon

exercise of this option is $h(t, S_t) = \left((S_t^1 \cdots S_t^5)^{(1/5)} - K\right)^+$. Since this option payoff is different, we use a slightly different set of inner and outer controls.[13]

Tables 4–6 show five asset max-option results with, respectively, $T = 3$, 6, and 9 (and thus 4, 7, and 10 exercise dates including time zero). Since the true values are not known, the pricing errors must be estimated. The columns labeled "Estim error" are based on the confidence intervals defined in (11). The error estimates in the columns labeled "'Actual' error" are based on the most accurate answers, which are obtained with the greatest computational effort. Tables 7 and 8 show five and seven asset geometric average option results with $T = 10$ (ie, 11 exercise dates). We use this option because the pricing problem can be reduced to a single-asset American option, which can be priced accurately using a one-dimensional binomial tree.

The initial parameters b, n_p, and N were chosen so that the bias of the mesh and path estimators, and the standard errors of the mesh and path estimators were all the same order of magnitude. In all of the tables, the mesh parameter b and path parameter n_p doubles from one row to the next within each panel. Hence the computational effort increases by roughly a factor of four from one row to the next. The CPU time for the first row (in each panel) of Table 4 is about 25 seconds (on a 266 MHz Pentium II processor). Computation times for each successive row are 1.5, 5.3, 20, 76, 307, and 1,217 minutes. Roughly, the first rows can be computed in seconds, the middle rows in minutes, and the last rows in hours.

Several features are notable in the tables. Most importantly, the method generally gives good results for a modest amount of computation time and the convergence of the method is apparent as the effort increases. For example, in the top panel of Table 4 corresponding to $S = 90$, the estimated error decreases from 2.50% in the first row to 0.20% in the seventh row. Throughout the seven rows, the point estimates vary from 16.438 to 16.481, a difference of only 4.3 cents. So even though the half-width of the first confidence interval is over 40 cents, the true error of the point estimate appears to be less than two cents. Throughout the tables, the 'actual' or true error is typically much smaller than the estimated error. In the top rows within each panel, the ratio of estimated to true error is often a factor of 10 or more. This is consistent with the observation that the intervals are conservative due to estimator bias. The average of the mesh and path estimators significantly reduces this bias, leading to smaller errors in the point estimates than are suggested by the confidence intervals.

[13] For the inner control we use the European geometric average option with a maturity of Δt. For the mesh estimator outer controls we also use European geometric average options, one with a maturity of T and one with a maturity of $3T/5$. For the path estimator outer controls we use the same controls $w^{(1)}$ and $w^{(2)}$ as for the max-option path estimator. For policy fixing with the path estimator, we use $\underline{P}(t, S_t)^{(1)} = 0$ and $\underline{P}(t, S_t)^{(2)}$ equal to the European option of the geometric average with a time to maturity $T - t$. Easily computed formulas are available for these European option controls. However, even if they were not available, they can be computed reasonably quickly using the Sobol' sequence or another low-discrepancy sequence. The numerical results for this option could be improved with a better choice of controls.

TABLE 4 American max-option on five assets, $T = 3$.

S_0	Path est $\bar{q}$	Std err of $\bar{q}$	Mesh est $\bar{Q}$	Std err of $\bar{Q}$	90% confidence bounds	Point est	Estim error	"Actual" error
90	15.867	0.038	16.115	0.038	[15.804, 16.177]	15.991	1.17%	(−0.16%, −0.03%)
90	15.929	0.036	16.042	0.022	[15.870, 16.078]	15.985	0.65%	(−0.19%, −0.06%)
90	15.979	0.022	16.060	0.017	[15.942, 16.089]	16.020	0.46%	(0.02%, 0.16%)
90	15.986	0.014	16.042	0.010	[15.963, 16.058]	16.014	0.30%	(−0.01%, 0.12%)
90	15.997	0.013	16.029	0.007	[15.976, 16.040]	16.013	0.20%	(−0.02%, 0.11%)
90	16.012	0.009	16.014	0.005	[15.997, 16.022]	16.013	0.08%	(−0.02%, 0.11%)
90	16.003	0.005	16.010	0.003	[15.995, 16.016]	16.006	0.07%	(−0.06%, 0.07%)
100	25.092	0.043	25.378	0.049	[25.022, 25.460]	25.235	0.87%	(−0.26%, −0.13%)
100	25.208	0.031	25.379	0.030	[25.157, 25.428]	25.294	0.54%	(−0.03%, 0.11%)
100	25.216	0.019	25.342	0.020	[25.184, 25.375]	25.279	0.38%	(−0.09%, 0.05%)
100	25.256	0.018	25.312	0.012	[25.226, 25.332]	25.284	0.21%	(−0.07%, 0.07%)
100	25.248	0.012	25.305	0.010	[25.228, 25.321]	25.277	0.18%	(−0.10%, 0.04%)
100	25.275	0.007	25.275	0.007	[25.265, 25.286]	25.275	0.04%	(−0.11%, 0.03%)
100	25.274	0.005	25.294	0.005	[25.267, 25.302]	25.284	0.07%	(−0.07%, 0.07%)
110	35.449	0.041	35.943	0.056	[35.382, 36.036]	35.696	0.92%	(−0.04%, 0.05%)
110	35.618	0.035	35.811	0.040	[35.561, 35.877]	35.715	0.44%	(0.01%, 0.10%)
110	35.626	0.023	35.757	0.024	[35.588, 35.796]	35.691	0.29%	(−0.05%, 0.03%)
110	35.670	0.015	35.743	0.018	[35.645, 35.772]	35.706	0.18%	(−0.01%, 0.08%)
110	35.691	0.011	35.711	0.011	[35.673, 35.730]	35.701	0.08%	(−0.03%, 0.06%)
110	35.685	0.007	35.696	0.007	[35.673, 35.708]	35.691	0.05%	(−0.05%, 0.03%)
110	35.688	0.006	35.701	0.005	[35.679, 35.710]	35.695	0.04%	(−0.04%, 0.04%)

Max-option with $n = 5$ assets. The parameters are $r = 5\%$, $\delta = 10\%$, $\sigma = 20\%$, $\rho = 0$, $K = 100$, and three-year maturity. The initial vector is $S_0 = (S, \ldots, S)$, with $S = 90$, 100, or 110 as indicated in the table. Equal time steps are used, with exercise opportunities at $t = 0$, 1, 2, and 3 years. The number of replications is $N = 50$ for each row. For each panel, the parameters (b, n_b) are (50, 500), (100, 1000), (200, 2000), (400, 4000), (800, 8000), (1600, 16000), (3200, 32000) for each of the seven rows, respectively. The point estimate is $(\bar{q} + \bar{Q})/2$. The estimated error is $(y - x)/2z$, where the 90% confidence interval is represented as $[x, y]$ and the point estimate is z. The "actual" error is $((z - y_7)/y_7, (z - x_7)/x_7)$, where $[x_7, y_7]$ represents the best 90% confidence interval from the seventh row of each panel. The European values are 14.586, 23.052, and 32.685 for $S = 90$, 100, and 110, respectively.

Regarding the convergence rate, note that the estimated error decreases by about a factor of two when comparing every other row of the tables. Since the work increases by a factor of about four from one row to the next, the results are consistent with fourth-root convergence. That is, the convergence appears to be $O(\text{work}^{-1/4})$. In fact, this convergence result is immediate when the stochastic mesh method is used to price European options. In this case, the decrease in error is order $b^{-1/2}$, the usual simulation result. However, the work is quadratic in b, so the $O(\text{work}^{-1/4})$ convergence result follows.

Comparing the results in Tables 4–6 shows that increasing the number of exercise opportunities increases estimator error. This is consistent with Table 1.

TABLE 5 American max-option on five assets, $T = 6$.

S_0	Path est $\bar{q}$	Std err of $\bar{q}$	Mesh est $\bar{Q}$	Std err of $\bar{Q}$	90% confidence bounds	Point est	Estim error	"Actual" error
90	16.159	0.049	16.768	0.082	[16.079, 16.903]	16.464	2.50%	(–0.25%, 0.16%)
90	16.257	0.029	16.675	0.042	[16.209, 16.744]	16.466	1.62%	(–0.24%, 0.17%)
90	16.294	0.022	16.581	0.019	[16.258, 16.612]	16.438	1.08%	(–0.41%, 0.00%)
90	16.351	0.016	16.570	0.015	[16.324, 16.595]	16.460	0.82%	(–0.27%, 0.13%)
90	16.439	0.012	16.522	0.009	[16.419, 16.536]	16.481	0.36%	(–0.15%, 0.26%)
90	16.441	0.010	16.488	0.005	[16.425, 16.496]	16.465	0.22%	(–0.24%, 0.16%)
90	16.448	0.006	16.500	0.003	[16.438, 16.505]	16.474	0.20%	(–0.19%, 0.22%)
100	25.469	0.046	26.432	0.072	[25.393, 26.550]	25.951	2.23%	(0.01%, 0.24%)
100	25.686	0.041	26.203	0.042	[25.619, 26.272]	25.945	1.26%	(–0.01%, 0.22%)
100	25.761	0.018	26.059	0.022	[25.730, 26.094]	25.910	0.70%	(–0.15%, 0.08%)
100	25.807	0.019	25.987	0.016	[25.776, 26.014]	25.897	0.46%	(–0.20%, 0.03%)
100	25.873	0.010	25.942	0.012	[25.857, 25.963]	25.908	0.20%	(–0.15%, 0.08%)
100	25.894	0.009	25.965	0.007	[25.880, 25.976]	25.930	0.18%	(–0.07%, 0.16%)
100	25.900	0.007	25.940	0.005	[25.889, 25.948]	25.920	0.12%	(–0.11%, 0.12%)
110	35.927	0.055	37.070	0.106	[35.836, 37.245]	36.499	1.93%	(–0.08%, 0.09%)
110	36.190	0.036	36.882	0.062	[36.131, 36.985]	36.536	1.17%	(0.02%, 0.19%)
110	36.308	0.027	36.726	0.035	[36.263, 36.783]	36.517	0.71%	(–0.03%, 0.14%)
110	36.378	0.018	36.574	0.020	[36.349, 36.607]	36.476	0.35%	(–0.14%, 0.03%)
110	36.443	0.012	36.566	0.013	[36.423, 36.588]	36.505	0.23%	(–0.06%, 0.11%)
110	36.460	0.008	36.532	0.008	[36.446, 36.546]	36.496	0.14%	(–0.08%, 0.08%)
110	36.477	0.007	36.517	0.006	[36.466, 36.527]	36.497	0.08%	(–0.08%, 0.09%)

Max-option with $n = 5$ assets. The parameters are $r = 5\%$, $\delta = 10\%$, $\sigma = 20\%$, $\rho = 0$, $K = 100$, and six-year maturity. The initial vector is $S_0 = (S, \ldots, S)$, with $S = 90$, 100, or 110 as indicated in the table. Equal time steps are used, with exercise opportunities at $t = 0, 1, \ldots, 6$ years. The number of replications is $N = 35$ for each row. For each panel, the parameters (b, n_p) are (50, 500), (100, 1000), (200, 2000), (400, 4000), (800, 8000), (1600, 16000), and (3200, 32000) for each of the seven rows, respectively. The point estimate is $(\bar{q} + \bar{Q})/2$. The estimated error is $(y - x)/2z$, where the 90% confidence interval is represented as $[x, y]$ and the point estimate is z. The "actual" error is $((z - y_7)/y_7, (z - x_7)/x_7)$, where $[x_7, y_7]$ represents the best 90% confidence interval from the seventh row of each panel. The European values are 14.586, 23.052, and 32.685 for $S = 90$, 100, and 110, respectively.

As the problem dimension increases, Tables 7 and 8 show that the estimator error also increases.

The enormous computational effort required for the last rows of each panel shows that this method is not generally useful for generating extremely accurate pricing results. However, the stochastic mesh method can easily be parallelized for implementation on multi-processor computers. Since the estimates are based on N independent meshes, it is straightforward to parallelize these computations. This can reduce the work by about a factor of N if there are N or more processors available. Further speedups are possible if the computations within each mesh are parallelized.[14] The results from the bottom rows of the tables could then be

[14] Avramidis *et al* (2000) report nearly perfect speed-up in parallelizing this method.

TABLE 6 American max-option on five assets, $T = 9$.

S_0	Path est $\bar{q}$	Std err of $\bar{q}$	Mesh est $\bar{Q}$	Std err of $\bar{Q}$	90% confidence bounds	Point est	Estim error	"Actual" error
90	16.094	0.057	18.252	0.290	[16.001, 18.729]	17.173	7.94%	(2.77%, 3.44%)
90	16.317	0.037	17.220	0.064	[16.256, 17.325]	16.768	3.19%	(0.35%, 1.00%)
90	16.412	0.020	16.912	0.033	[16.379, 16.966]	16.662	1.76%	(–0.29%, 0.36%)
90	16.471	0.019	16.838	0.014	[16.440, 16.861]	16.655	1.27%	(–0.33%, 0.32%)
90	16.546	0.014	16.789	0.010	[16.522, 16.806]	16.667	0.85%	(–0.26%, 0.39%)
90	16.573	0.010	16.738	0.007	[16.557, 16.748]	16.656	0.58%	(–0.32%, 0.33%)
90	16.613	0.007	16.704	0.003	[16.602, 16.710]	16.659	0.32%	(–0.31%, 0.34%)
100	25.362	0.050	28.165	0.455	[25.280, 28.913]	26.764	6.79%	(2.11%, 2.54%)
100	25.675	0.038	26.618	0.062	[25.612, 26.720]	26.146	2.12%	(–0.25%, 0.17%)
100	25.887	0.029	26.660	0.062	[25.840, 26.761]	26.274	1.75%	(0.24%, 0.66%)
100	25.969	0.017	26.333	0.023	[25.941, 26.370]	26.151	0.82%	(–0.23%, 0.19%)
100	26.045	0.010	26.266	0.011	[26.029, 26.283]	26.155	0.49%	(–0.21%, 0.21%)
100	26.081	0.011	26.195	0.006	[26.063, 26.205]	26.138	0.27%	(–0.28%, 0.14%)
100	26.113	0.007	26.204	0.004	[26.101, 26.211]	26.158	0.21%	(–0.20%, 0.22%)
110	35.815	0.062	38.040	0.196	[35.713, 38.362]	36.928	3.59%	(0.23%, 0.57%)
110	36.293	0.042	37.457	0.162	[36.224, 37.723]	36.875	2.03%	(0.09%, 0.42%)
110	36.370	0.025	37.083	0.033	[36.329, 37.137]	36.727	1.10%	(–0.31%, 0.02%)
110	36.575	0.018	36.958	0.023	[36.546, 36.996]	36.767	0.61%	(–0.20%, 0.13%)
110	36.654	0.015	36.944	0.011	[36.629, 36.962]	36.799	0.45%	(–0.12%, 0.22%)
110	36.694	0.011	36.880	0.008	[36.676, 36.893]	36.787	0.29%	(–0.15%, 0.19%)
110	36.731	0.008	36.832	0.006	[36.719, 36.842]	36.782	0.17%	(–0.16%, 0.17%)

Max-option with n = 5 assets. The parameters are $r = 5\%$, $\delta = 10\%$, $\sigma = 20\%$, $\rho = 0$, $K = 100$, and nine-year maturity. The initial vector is $S_0 = (S, \ldots, S)$, with $S = 90$, 100, or 110 as indicated in the table. Equal time steps are used, with exercise opportunities at $t = 0, 1, \ldots, 9$ years. The number of replications is $N = 25$ for each row. For each panel, the parameters (b, n_b) are (50, 500), (100, 1000), (200, 2000), (400, 4000), (800, 8000), (1600, 16000), and (3200, 32000) for each of the seven rows, respectively. The point estimate is $(\bar{q} + \bar{Q})/2$. The estimated error is $(y - x)/2z$, where the 90% confidence interval is represented as $[x, y]$ and the point estimate is z. The "actual" error is $((z - y_7)/y_7, (z - x_7)/x_7)$, where $[x_7, y_7]$ represents the best 90% confidence interval from the seventh row of each panel. The European values are 14.586, 23.052, and 32.685 for $S = 90$, 100, and 110, respectively.

computed in seconds or minutes instead of hours.

In order to place these results in some perspective, consider the convergence of the binomial method. For single asset pricing problems, Leisen and Reimer (1996) show that the binomial method converges linearly with the number of time steps when applied to European options. Broadie and Detemple (1996) offer compelling empirical evidence that linear convergence also holds for the binomial method applied to American options. Since the computational work is quadratic in the number of time steps, the convergence rate for the binomial method is $O(\text{work}^{-1/2})$. All of the multi-dimensional generalizations of the binomial method (eg, Boyle, Evnine, and Gibbs, 1989; He, 1990; and Kamrad and Ritchken, 1991) have work which increases as m^{n+1}, where m is the number of time steps and n is the number

TABLE 7 American max-option on five assets, $T = 10$.

S_0	Path est $\bar{q}$	Std err of $\bar{q}$	Mesh est $\bar{Q}$	Std err of $\bar{Q}$	90% confidence bounds	Point est	Estim error	"Actual" error
90	1.308	0.025	1.582	0.030	[1.266, 1.631]	1.445	12.63%	6.03%
90	1.361	0.021	1.497	0.032	[1.327, 1.550]	1.429	7.79%	4.88%
90	1.353	0.012	1.388	0.005	[1.333, 1.396]	1.370	2.29%	0.57%
90	1.355	0.008	1.392	0.006	[1.342, 1.403]	1.373	2.23%	0.81%
90	1.348	0.007	1.386	0.002	[1.337, 1.389]	1.367	1.93%	0.33%
90	1.362	0.004	1.380	0.001	[1.355, 1.382]	1.371	0.98%	0.66%
90	1.356	0.003	1.375	0.001	[1.351, 1.376]	1.365	0.90%	0.22%
100	4.166	0.035	4.522	0.034	[4.108, 4.577]	4.344	5.40%	1.23%
100	4.258	0.019	4.439	0.017	[4.226, 4.467]	4.348	2.77%	1.34%
100	4.272	0.017	4.392	0.009	[4.244, 4.407]	4.332	1.87%	0.96%
100	4.282	0.015	4.368	0.005	[4.258, 4.376]	4.325	1.37%	0.79%
100	4.267	0.008	4.348	0.003	[4.253, 4.352]	4.308	1.15%	0.39%
100	4.290	0.007	4.320	0.002	[4.279, 4.323]	4.305	0.52%	0.33%
100	4.283	0.004	4.309	0.001	[4.276, 4.311]	4.296	0.40%	0.12%
110	10.156	0.037	10.527	0.036	[10.096, 10.587]	10.341	2.37%	1.28%
110	10.170	0.018	10.401	0.022	[10.140, 10.436]	10.285	1.44%	0.73%
110	10.192	0.013	10.369	0.017	[10.171, 10.396]	10.280	1.10%	0.68%
110	10.193	0.009	10.240	0.013	[10.178, 10.262]	10.216	0.41%	0.05%
110	10.203	0.007	10.252	0.004	[10.191, 10.258]	10.228	0.33%	0.16%
110	10.199	0.004	10.238	0.002	[10.193, 10.242]	10.218	0.24%	0.07%
110	10.208	0.002	10.230	0.002	[10.205, 10.233]	10.219	0.14%	0.08%

Geometric average option with $n = 5$ assets. The parameters are $r = 3\%$, $\delta = 5\%$, $\sigma = 40\%$, $\rho = 0$, $K = 100$, and one year maturity. The initial vector is $S_0 = (S, \ldots, S)$, with $S = 90$, 100, or 110 as indicated in the table. Equal time steps are used, with exercise opportunities at $t = 0, 1, 2, \ldots, 1$ years. The number of replications is $N = 25$ for each row. For each panel, the parameters (b, n_p) are (50, 500), (100, 1000), (200, 2000), (400, 4000), (800, 8000), (1600, 16000), and (3200, 32000) for each of the seven rows in order. The point estimate is $(\bar{q} + \bar{Q})/2$ The estimated error is $(y - x)/2z$, where the 90% confidence interval is represented as $[x, y]$ and the point estimate is z. The actual error is $(z - Q)/Q$, where Q is the true value determined from a single asset binomial tree. The European and American values are (1.172, 1.362), (3.445, 4.291), and (7.521, 10.211) corresponding to $S = 90$, 100, and 110, respectively.

of underlying assets.[15] Hence, the convergence rate of the multi-dimensional binomial method appears to be $O(\text{work}^{-1/(n+1)})$. A second-order finite-difference method may converge an order of magnitude faster than a binomial approximation, but its computational requirements still grow exponentially in the dimension of the problem. These considerations suggest that the stochastic mesh dominates binomial and finite difference methods in sufficiently high dimensions. Numerical results in this paper indicate that the crossover may occur at four or five dimensions.

[15] Storage is another problem when applying the binomial method to high-dimensional problems. Storing all of the terminal option values requires order m^n storage.

TABLE 8 American max-option on seven assets, $T = 10$.

S_0	Path est $\bar{q}$	Std err of $\bar{q}$	Mesh est $\bar{Q}$	Std err of $\bar{Q}$	90% confidence bounds	Point est	Estim error	"Actual" error
90	0.728	0.019	0.763	0.044	[0.697, 0.835]	0.745	9.21%	–1.99%
90	0.744	0.008	0.762	0.041	[0.731, 0.831]	0.753	6.60%	–0.93%
90	0.741	0.009	0.876	0.028	[0.727, 0.922]	0.809	12.01%	6.37%
90	0.756	0.006	0.772	0.017	[0.747, 0.800]	0.764	3.47%	0.46%
90	0.753	0.004	0.789	0.004	[0.747, 0.796]	0.771	3.17%	1.42%
90	0.758	0.002	0.777	0.001	[0.754, 0.779]	0.767	1.60%	0.91%
100	3.159	0.034	3.929	0.140	[3.103, 4.160]	3.544	14.92%	8.38%
100	3.232	0.026	3.447	0.036	[3.190, 3.507]	3.340	4.75%	2.12%
100	3.220	0.010	3.426	0.011	[3.204, 3.444]	3.323	3.62%	1.62%
100	3.250	0.012	3.408	0.016	[3.230, 3.434]	3.329	3.05%	1.80%
100	3.256	0.010	3.361	0.003	[3.239, 3.367]	3.308	1.93%	1.17%
100	3.260	0.006	3.347	0.002	[3.251, 3.350]	3.304	1.50%	1.03%
100	3.264	0.004	3.314	0.001	[3.258, 3.316]	3.289	0.88%	0.58%
110	9.812	0.072	10.324	0.068	[9.693, 10.436]	10.068	3.69%	0.68%
110	9.954	0.046	10.093	0.053	[9.878, 10.180]	10.023	1.51%	0.23%
110	10.000	0.000	10.065	0.017	[10.000, 10.092]	10.033	0.46%	0.33%
110	10.000	0.000	10.000	0.000	[10.000, 10.000]	10.000	0.00%	0.00%
110	10.000	0.000	10.000	0.000	[10.000, 10.000]	10.000	0.00%	0.00%

Geometric average option with n = 7 assets. The parameters are $r = 3\%$, $\delta = 5\%$, $\sigma = 40\%$, $\rho = 0$, $K = 100$, and one-year maturity. The initial vector is $S_0 = (S, \ldots, S)$, with $S = 90$, 100, or 110 as indicated in the table. Equal time steps are used, with exercise opportunities at $t = 0, 1, 2, \ldots, 1$ years. The number of replications is $N = 25$ for each row. For each panel, the parameters (b, n_p) are (50, 500), (100, 1000), (200, 2000), (400, 4000), (800, 8000), etc., for each of the rows in order. The point estimate is $(\bar{q} + \bar{Q})/2$. The estimated error is $(y - x)/2z$ where the 90% confidence interval is represented as $[x, y]$ and the point estimate is z. The actual error is $(z - Q)/Q$, where Q is the true value determined from a single asset binomial tree. The European and American values are (0.628, 0.761), (2.419, 3.270), and (6.201, 10.000) corresponding to $S = 90$, 100, and 110, respectively.

6 Conclusions

American-style securities whose value depends on multiple assets or on multiple state variables are increasingly common. With this comes a growing need for methods to price and hedge these securities. Approximation methods have been proposed for some types of high-dimensional securities. However, no convergent algorithm has been proposed and tested for any general class of such securities.

In this paper, we propose, analyze, and test the stochastic mesh method for pricing a general class of high-dimensional pricing problems with a finite number of exercise dates. The computational effort increases quadratically with the number of mesh points and linearly with the number of exercise opportunities. We show that the method converges as the computational effort increases. Numerical results illustrate this convergence and demonstrate the viability of the method. Practical success of the method depends critically on the use of effective variance

reduction techniques. In particular, our results indicate the necessity of using control variates well-suited to the specific pricing problem.

An evident limitation of the method is its reliance on explicit knowledge of the transition density of the underlying state variables. In many cases, the transition density is unknown or may even fail to exist. In such settings one may consider using a normal or lognormal density as an approximation. An alternative strategy for selecting mesh weights is proposed and tested in Broadie, Glasserman, and Ha (2000). That method does not use a transition density but instead uses information about moments of the underlying state variables or the prices of easily computed European options. Glasserman and Yu (2003) show that this method is equivalent to a regression-based estimator.

Another important topic not investigated here is the calculation of price *sensitivities* for hedging purposes. The problem of estimating sensitivities in pricing European options by simulation has been considered in Broadie and Glasserman (1996), and those methods are potentially applicable in the stochastic mesh as well. There are at least two natural strategies to consider – estimating sensitivities of the mesh estimator and estimating sensitivities of the path estimator. The second of these is the more straightforward and would likely give better results. See Piterbarg (2003) and Kaniel, Tompaidis, and Zemlianov (2003) for recent work on this problem.

Appendix

PROOF OF THEOREM 1 The proof is by induction. At the terminal time we have $\hat{Q}_b(T,x) = h(T,x) = Q(T,x)$ for all x. Take as induction hypothesis that $E[\hat{Q}_b(t+1,x)] \geq Q(t+1,x)$ for all x. Now we have

$$
\begin{aligned}
&E\left[\hat{Q}_b(t,x)\right] \\
&= E\left[\max\left(h(t,x), \frac{1}{b}\sum_{j=1}^{b}\frac{f\left(t,x,X_{t+1}(j)\right)}{g\left(t+1,X_{t+1}(j)\right)}\hat{Q}_b\left(t+1,X_{t+1}(j)\right)\right)\right] \\
&\geq \max\left(h(t,x), E\left[\frac{1}{b}\sum_{j=1}^{b}\frac{f\left(t,x,X_{t+1}(j)\right)}{g\left(t+1,X_{t+1}(j)\right)}\hat{Q}_b\left(t+1,X_{t+1}(j)\right)\right]\right) \\
&= \max\left(h(t,x), E\left[\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\hat{Q}_b\left(t+1,X_{t+1}(1)\right)\right]\right) \\
&= \max\left(h(t,x), E\left[\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}E\left[\hat{Q}_b\left(t+1,X_{t+1}(1)\right)\middle|X_{t+1}(1)\right]\right]\right) \\
&\geq \max\left(h(t,x), E\left[\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}Q\left(t+1,X_{t+1}(1)\right)\right]\right) \\
&= \max\left(h(t,x), E\left[Q\left(t+1,S_{t+1}\right)\middle|S_t = x\right]\right) \\
&= Q(t,x)
\end{aligned}
$$

The first three steps use the definition of $\hat{Q}_b$, Jensen's inequality, and the fact that the mesh points at each time slice are identically distributed. The fourth uses a basic property of conditional expectations and the fifth uses the induction hypothesis. The sixth step follows from the identity

$$E\left[\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}Q\left(t+1,X_{t+1}(1)\right)\right]=\int f(t,x,y)Q(t+1,y)\mathrm{d}y$$

and the last step follows from the optimality equation (2). □

In order to prove Theorem 2, we prove a preliminary lemma (see Assumptions 1 and 2).

LEMMA 1 *For any $r \geq 1$, (i) if Assumption (1) holds then $E[Q(t, X_t(1))^r] < \infty$. (ii) If Assumption (2) holds then* $\sup_{b \geq 1} E[\hat{Q}_b(t_1, X_{t1}(1))^r] < \infty$.

Proof of Lemma 1: (i) Repeatedly applying the simple bound

$$\begin{aligned}Q(t,x)&=\max\left(h(t,x),E\left[Q\left(t+1,S_{t+1}\right)\middle|S_t=x\right]\right)\\&\leq h(t,x)+E\left[Q\left(t+1,S_{t+1}\right)\middle|S_t=x\right]\end{aligned}$$

we find that

$$Q(t,x)\leq h(t,x)+E\left[h\left(t+1,S_{t+1}\right)\middle|S_t=x\right]+\cdots+E\left[h\left(T,S_T\right)\middle|S_t=x\right] \quad (32)$$

Thus, $Q(t, X_t(1))$ has finite rth moment if each of the terms on the right does; ie, if

$$\int E\left[h\left(\tau,S_\tau\right)\middle|S_t=x\right]^r g(t,x)\mathrm{d}x<\infty$$

Applying Jensen's inequality and then the definition of $f(t, \cdot)$, we get

$$\begin{aligned}\int E\left[h\left(\tau,S_\tau\right)\middle|S_t=x\right]^r g(t,x)\mathrm{d}x&\leq\int E\left[h\left(\tau,S_\tau\right)^r\middle|S_t=x\right]g(t,x)\mathrm{d}x\\&=\int E\left[h\left(\tau,S_\tau\right)^r\middle|S_t=x\right]\frac{g(t,x)}{f(t,x)}f(t,x)\mathrm{d}x\\&=E\left[h\left(\tau,S_\tau\right)^r\left(\frac{g(t,S_t)}{f(t,S_t)}\right)\right]\end{aligned}$$

which is finite by hypothesis.

(ii) Paralleling (32), we have

$$\hat{Q}_b\left(T-k, X_{T-k}(1)\right) \le h\left(T-k, X_{T-k}(1)\right) +$$
$$\frac{1}{b}\sum_{j_1=1}^{b} \frac{f\left(T-k, X_{T-k}(1), X_{T-k+1}(j_1)\right)}{g\left(T-k+1, X_{T-k+1}(j_1)\right)} h\left(T-k+1, S_{T-k+1}(j_1)\right)$$
$$+\cdots+\frac{1}{b^k}\sum_{j_1=1}^{b}\cdots\sum_{j_k=1}^{b} \frac{f\left(T-k, X_{T-k}(1), X_{T-k+1}(j_1)\right)}{g\left(T-k+1, X_{T-k+1}(j_1)\right)}\times\cdots$$
$$\times\frac{f\left(T-1, X_{T-1}(j_{k-1}), X_{T-k}(j_k)\right)}{g\left(T, X_T(j_k)\right)} h\left(T, X_T(j_k)\right)$$

The r-norm $E[\hat{Q}_b(T-k, X_{T-k}(1))^r]^{1/r}$ is bounded by the sum of the r-norms of the terms on the right. The mth term on the right is the average of b^{m-1} terms, each having the same distribution as $R(T-k, T-k+m-1)$. The r-norm of each such average is bounded above by the r-norm of any one of the terms in the average. Thus,

$$E\left[\hat{Q}_b\left(T-k, X_{T-k}(1)\right)^r\right]^{1/r} \le E\left[R\left(T-k, T-k\right)^r\right]^{1/r} +\cdots+ E\left[R\left(T-k, T\right)^r\right]^{1/r}$$

The right side is independent of b and finite by hypothesis; we conclude that $\sup_{b\ge 1} E[\hat{Q}_b(T-k, X_{T-k}(1))^r] < \infty$. □

Proof of Theorem 2: We prove the result by induction, proceeding backwards from the terminal time. At T there is nothing to prove because $\hat{Q}_b(T, \cdot) \equiv h(T, \cdot) \equiv Q(T, \cdot)$ for all b. Take as induction hypothesis that $\|\hat{Q}_b(t+1, x) - Q(t+1, x)\|_{p''} \to 0$ for all x, for some $p'' > p$. We will show that this implies $\|\hat{Q}_b(t, x) - Q(t, x)\|_{p'} \to 0$ for all x, for some $p' > p$. Using the fact that $|\max(a, b_1) - \max(a, b_2)| \le |b_1 - b_2|$ for any real numbers a, b_1, b_2, we get, for any $p' \in (p, p'')$,

$$\left\|\hat{Q}_b(t,x) - Q(t,x)\right\|_{p'}$$
$$\le \left\| \frac{1}{b}\sum_{j=1}^{b} \frac{f\left(t, x, X_{t+1}(j)\right)}{g\left(t+1, X_{t+1}(j)\right)} \hat{Q}_b\left(t+1, X_{t+1}(j)\right) - E\left[Q(t+1, S_{t+1}) \,\middle|\, S_t = x\right] \right\|_{p'}$$

And now by the triangle inequality,

$$\left\|\hat{Q}_b(t,x) - Q(t,x)\right\|_{p'} \le \Delta_b + \Delta$$

where

$$\Delta_b = \left\| \frac{1}{b}\sum_{j=1}^{b} \frac{f\left(t, x, X_{t+1}(j)\right)}{g\left(t+1, X_{t+1}(j)\right)} \left[\hat{Q}_b\left(t+1, X_{t+1}(j)\right) - Q\left(t+1, X_{t+1}(j)\right)\right] \right\|_{p'}$$

and

$$\Delta = \left\| \frac{1}{b}\sum_{j=1}^{b} \frac{f\left(t,x,X_{t+1}(j)\right)}{g\left(t+1,X_{t+1}(j)\right)} Q\left(t+1,X_{t+1}(j)\right) - E\left[Q(t+1,S_{t+1}) \mid S_t = x\right] \right\|_{p'}$$

We analyze these terms in turn. Because the summands appearing in Δ_b are identically distributed,

$$\Delta_b \le \left\| \frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)} \left[\hat{Q}_b\left(t+1,X_{t+1}(1)\right) - Q\left(t+1,X_{t+1}(1)\right)\right] \right\|_{p'}$$

Applying Hölder's inequality to the expression on the right we get, for any $q > 1$,

$$\Delta_b \le \left\| \frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)} \right\|_{\frac{qp'}{q-1}} \left\| \hat{Q}_b\left(t+1,X_{t+1}(1)\right) - Q\left(t+1,X_{t+1}(1)\right) \right\|_{qp'}$$

In particular, we can choose $q > 1$ to satisfy $qp' < p''$ because $p' < p''$. The first factor on the right is finite in light of (8) (Assumption 3). To show that $\Delta_b \to 0$ we need to show that the second factor converges to zero. By the induction hypothesis

$$E\left[\left|\hat{Q}_b\left(t+1,X_{t+1}(1)\right) - Q\left(t+1,X_{t+1}(1)\right)\right|^{qp'} \middle| X_{t+1}(1)\right] \to 0 \quad \text{a.s.} \tag{33}$$

In addition, for $\varepsilon > 0$ small enough that $p''(1+\varepsilon) \le r$, with r as in the statement of the theorem,

$$\begin{aligned}
&\sup_{b\ge 1} E\left[\left(E\left[\left|\hat{Q}_b\left(t+1,X_{t+1}(1)\right) - Q\left(t+1,X_{t+1}(1)\right)\right|^{qp'} \middle| X_{t+1}(1)\right]\right)^{1+\varepsilon}\right] \\
&\le \sup_{b\ge 1} E\left[\left|\hat{Q}_b\left(t+1,X_{t+1}(1)\right) - Q\left(t+1,X_{t+1}(1)\right)\right|^{qp'(1+\varepsilon)}\right] \\
&\le \sup_{b\ge 1} E\left[\hat{Q}_b\left(t+1,X_{t+1}(1)\right)^r\right] + E\left[Q\left(t+1,X_{t+1}(1)\right)^r\right] < \infty
\end{aligned}$$

with the last inequality ensured by Lemma 1. It follows that the sequence in (33) in uniformly integrable, and therefore that the expectation of the left side of (33) converges to 0:

$$E\left[\left|\hat{Q}_b\left(t+1,X_{t+1}(1)\right) - Q\left(t+1,X_{t+1}(1)\right)\right|^{qp'}\right] \to 0$$

from which $\Delta_b \to 0$ follows.

For the analysis of Δ, first observe that the b summands appearing in the summation in Δ are independent and identically distributed with mean

$$E\left[\frac{f\left(t,x,X_{t+1}(j)\right)}{g\left(t+1,X_{t+1}(j)\right)}Q\left(t+1,X_{t+1}(j)\right)\right]=\int\frac{f\left(t,x,y\right)}{g\left(t+1,y\right)}Q\left(t+1,y\right)g\left(t+1,y\right)\mathrm{d}y$$
$$=\int f\left(t,x,y\right)Q\left(t+1,y\right)\mathrm{d}y$$
$$=E\left[Q(t+1,S_{t+1})\,\middle|\,S_t=x\right]$$

Hence, $\Delta\to 0$ provided that

$$\frac{1}{b}\sum_{j=1}^{b}\frac{f\left(t,x,X_{t+1}(j)\right)}{g\left(t+1,X_{t+1}(j)\right)}Q\left(t+1,X_{t+1}(j)\right)$$

converges to its expectation in p'-norm as $b\to\infty$. This holds (eg, Theorem I.4.1 of Gut, 1988) provided the summands have finite p'-norm. By Hölder's inequality,

$$E\left[\left(\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}Q\left(t+1,X_{t+1}(1)\right)\right)^{p'}\right]$$
$$\le E\left[\left(\frac{f\left(t,x,X_{t+1}(1)\right)}{g\left(t+1,X_{t+1}(1)\right)}\right)^{\frac{rp'}{r-p'}}\right]^{\frac{r-p'}{r}}E\left[Q\left(t+1,X_{t+1}(1)\right)\right]^{\frac{p'}{r}}$$

The first factor on the right is finite by (8) (Assumption 3) and the finiteness of second factor follows from Lemma 1 (i). □

Proof of Theorem 4: We first show that

$$P(\hat{\tau}_b\ne\tau)\to 0 \tag{34}$$

establishing that the mesh exercise time converges in probability to the optimal exercise time. The two differ only if the mesh policy either exercises prematurely or fails to exercise at τ. Thus,

$$P(\hat{\tau}_b\ne\tau)$$
$$\le P\left(\hat{Q}_b(t,S_t)\le h(t,S_t)<Q(t,S_t),\ \text{ for some } t=1,\ldots,T-1\right)$$
$$+P\left(h(\tau,S_\tau)<\hat{Q}_b(\tau,S_\tau)\right)$$
$$\le\sum_{t=0}^{T-1}P\left(\hat{Q}_b(t,S_t)\le h(t,S_t)<Q(t,S_t)\ \text{ or }\ Q(t,S_t)\le h(t,S_t)<\hat{Q}_b(t,S_t)\right)$$

Under the hypothesis in the theorem that the "boundary" $\{x\colon h(t,x)=Q(t,x)\}$, $t<T$, is hit with probability zero, there exists an $\varepsilon>0$, almost surely, for which

$$\left|h(t,S_t)-Q(t,S_t)\right|>\varepsilon,\ \ t=0,1,\ldots,T-1$$

Using this together with the previous inequality we get

$$P\left(\hat{\tau}_b \neq \tau\right) \leq \sum_{t=0}^{T-1} P\left(\left|\hat{Q}_b(t,S_t) - Q(t,S_t)\right| > \varepsilon\right) \tag{35}$$

Notice that ε depends on the path $S = (S_0,\ldots,S_T)$ but is independent of the mesh. Because convergence in p-norm implies convergence in probability, we know from Theorem 2 that

$$P\left(\left|\hat{Q}_b(t,S_t) - Q(t,S_t)\right| > \varepsilon \,\middle|\, S\right) \to 0, \text{ a.s., } t = 0,1,\ldots,T \tag{36}$$

The dominated convergence theorem further implies that

$$P\left(\left|\hat{Q}_b(t,S_t) - Q(t,S_t)\right| > \varepsilon\right) \to 0, \quad t = 0,1,\ldots,T$$

which, together with (35), establishes (34).

Asymptotic unbiasedness now follows:

$$\begin{aligned} 0 \leq Q(0,S_0) - E\left[h\left(\hat{\tau}_b, S_{\hat{\tau}_b}\right)\right] &= E\left[h(\tau,S_\tau) - h\left(\hat{\tau}_b, S_{\hat{\tau}_b}\right)\right] \\ &= E\left[h(\tau,S_\tau) - h\left(\hat{\tau}_b, S_{\hat{\tau}_b}\right); \hat{\tau}_b \neq \tau\right] \\ &\leq E\left[h(\tau,S_\tau); \hat{\tau}_b \neq \tau\right] \\ &\leq E\left[h(\tau,S_\tau)^{1+\varepsilon}\right]^{1/(1+\varepsilon)} P(\hat{\tau}_b \neq \tau)^{\varepsilon/(1+\varepsilon)} \to 0 \end{aligned}$$

the last inequality is Hölder's. □

Proof of Proposition 1: To lighten notation, we write $L_b(T)$ and $L_b(T, i)$ as $L(T)$ and $L(T, i)$ respectively. Let $c > 1$ be the infimum in (14). We will show that

$$E\left[L(t+1)^2\right] \geq \frac{1}{b}\left(cE\left[L(t)^2\right] + (b-1)E\left[L(t,1)L(t,2)\right]\right) \tag{37}$$

$$E\left[L(t+1,1)L(t+1,2)\right] \geq \frac{1}{b}\left(E\left[L(t)^2\right] + (b-1)E\left[L(t,1)L(t,2)\right]\right) \tag{38}$$

Here and throughout the proof we assume the expectations are finite; if any of them is infinite, (16) holds trivially. Set

$$w(t,i,j) = \frac{f\left(t, X_t(i), X_{t+1}(j)\right)}{g\left(t+1, X_{t+1}(j)\right)} \quad t = 0,1,\ldots,i, \; j = 1,\ldots,b$$

Then

$$L(t+1,j) = \frac{1}{b}\sum_{i=1}^{b} L(t,i)w(t,i,j)$$

Consequently,

$$
\begin{aligned}
&E\left[L(t+1)^2\right] \\
&\quad = \frac{1}{b^2} E\left[\left(\sum_{i=1}^{b} L(t,i)w(t,i,1)\right)^2\right] \\
&\quad = \frac{1}{b}\left\{E\left[L(t)w(t,1,1)\left(\sum_{i=1}^{b} L(t,i)w(t,i,1)\right)\right]\right\} \\
&\quad = \frac{1}{b}\left\{E\left[\left(L(t)w(t,1,1)\right)^2\right] + E\left[L(t,1)w(t,1,1)\left(\sum_{i=2}^{b} L(t,i)w(t,i,1)\right)\right]\right\} \\
&\quad = \frac{1}{b}\left\{E\left[\left(L(t)w(t,1,1)\right)^2\right] + (b-1)E\left[L(t,1)w(t,1,1)L(t,2)w(t,2,1)\right]\right\} \qquad (39)
\end{aligned}
$$

But

$$
\begin{aligned}
&E\left[\left(L(t)w(t,1,1)\right)^2\right] \\
&\quad = E\left[E\left[\left(L(t)w(t,1,1)\right)^2 \,\middle|\, \left\{X_u(i), u=0,\ldots,t,\ i=1,\ldots,b\right\}\right]\right] \\
&\quad = E\left[L(t)^2\, E\left[w(t,1,1)^2 \,\middle|\, X_t(1)\right]\right] \\
&\quad \geq cE\left[L(t)^2\right] \qquad (40)
\end{aligned}
$$

by (14), and

$$
\begin{aligned}
&E[L(t,1)w(t,1,1)L(t,2)w(t,2,1)] \\
&\quad = E\left[E\left[L(t,1)w(t,1,1)L(t,2)w(t,2,1)\,\middle|\,\left\{X_u(i), u=0,\ldots,t,\ i=1,\ldots,b\right\}\right]\right] \\
&\quad = E\left[L(t,1)L(t,2)E\left[w(t,1,1)w(t,2,1)\,\middle|\, X_t(1), X_t(2)\right]\right] \\
&\quad \geq L(t,1)L(t,2) \qquad (41)
\end{aligned}
$$

by (14). Combining (39), (40), and (41) gives (37). Similarly,

$$
\begin{aligned}
&E[L(t+1,1)L(t+1,2)] \\
&\quad = \frac{1}{b^2} E\left[\left(\sum_{i=1}^{b} L(t,i)w(t,i,1)\right)\left(\sum_{j=1}^{b} L(t,j)w(t,j,2)\right)\right] \\
&\quad = \frac{1}{b} E\left[L(t,1)w(t,1,1)\left(\sum_{j=1}^{b} L(t,j)w(t,j,2)\right)\right] \\
&\quad = \frac{1}{b}\left\{E\left[L(t,1)^2 w(t,1,1)w(t,1,2)\right] +\right.
\end{aligned}
$$

$$(b-1)E\Big[L(t,1)L(t,2)w(t,1,1)w(t,2,2)\Big]\Big\}$$

$$=\frac{1}{b}\Big\{E\Big[L(t,1)^2 w(t,1,1)w(t,1,2)\Big]+(b-1)E\Big[L(t,1)L(t,2)\Big]\Big\} \qquad (42)$$

where the last step following from the conditional independence of $w(t, 1, 1)$ and $w(t, 2, 2)$ given $\{X_u(i), u = 0,\ldots,t, i = 1,\ldots,b\}$. Arguing as in (41), we get

$$E\Big[L(t,1)^2 w(t,1,1)w(t,1,2)\Big]\geq E\Big[L(t,1)^2\Big] \qquad (43)$$

Combining (42) and (43) yields (38).

In vector-matrix notation, (37) and (38) become

$$\begin{pmatrix} E\Big[L(t+1)^2\Big] \\ E\Big[L(t+1,1)L(t+1,2)\Big]\end{pmatrix}\geq\begin{pmatrix}\frac{c}{b} & \frac{b-1}{b} \\ \frac{1}{b} & \frac{b-1}{b}\end{pmatrix}\begin{pmatrix} E\Big[L(t)^2\Big] \\ E\Big[L(t,1)L(t,2)\Big]\end{pmatrix}$$

with the inequality holding componentwise. Combining this with the bounds

$$E[L(1)^2]\geq c, \qquad E[L(1,1)L(1,2)]\geq 1$$

and the non-negativity of the matrix entries, we get

$$\begin{pmatrix} E\Big[L(t+1)^2\Big] \\ E\Big[L(t+1,1)L(t+1,2)\Big]\end{pmatrix}\geq\begin{pmatrix} u_{t+1} \\ v_{t+1}\end{pmatrix}\equiv\begin{pmatrix}\frac{c}{b} & \frac{b-1}{b} \\ \frac{1}{b} & \frac{b-1}{b}\end{pmatrix}^t\begin{pmatrix} c \\ 1\end{pmatrix} \qquad (44)$$

The eigenvalues of the matrix in this inequality are

$$\lambda_{\pm}=\frac{(b+c-1)\pm\sqrt{(b+c-1)^2+4(c-1)}}{2b}$$

so both are positive and λ_+ is strictly greater than 1. Write $(c, 1)' = a_+w_+ + a_-w_-$ where $w_\pm$ are eigenvectors with eigenvalues $\lambda_\pm$. It is easy to see that $(c,1)'$ itself is not an eigenvector and therefore that $a_+ \neq 0$. Using the fact that $E[L(t+1, 1)^2] \geq E[L(t+1, 1)L(t+1, 2)]$ (since $L(t+1, 1)$ and $L(t+1, 2)$ have the same distribution) and taking norms in (44) we get

$$E\Big[L(t+1)^2\Big]\geq\frac{1}{2}\sqrt{u_{t+1}^2+v_{t+1}^2}\geq\frac{|a_+|}{2}\lambda_+^t$$

Since $E[L(t+1)] = 1$,

$$\text{var}\Big[L(t+1)\Big]\geq\frac{|a_+|}{2}\lambda_+^t-1$$

By choosing $1 < \lambda < \lambda_+$ and t_0 large enough that

$$a \equiv \frac{|a_+|}{2}\left(\frac{\lambda_+}{\lambda}\right)^{t_0} - 1 > 0$$

we get (16) for all $t \geq t_0$. □

Proof of Proposition 2: A simple induction argument applied to (13) verifies that $L(T, j) \equiv 1$ using the average density function. That each $X_T(j)$ has the distribution of S_T is also proved by induction: it clearly holds for $T = 1$; and if the unconditional distribution of each $X_{T-1}(j)$ is that of S_{T-1}, then the unconditional distribution of each draw from

$$\frac{1}{b}\sum_{j=1}^{b} f\left(T-1, X_{T-1}(j), \cdot\right)$$

is that of S_T. □

REFERENCES

Andersen, L. (2000). A simple approach to the pricing of Bermudan swaptions in the multi-factor LIBOR market model. *Journal of Computational Finance* **3**(2), 5–32.

Andersen, L., and Broadie, M. (2004). A primal–dual simulation algorithm for pricing multi-dimensional American options. *Management Science,* to appear.

Avramidis, A., and Hyden, P. (1999). Efficiency improvements for pricing American options with a stochastic mesh. In *Proceedings of the Winter Simulation Conference*. IEEE Press, New York, pp. 344–50.

Avramidis, A. N., and Matzinger, H. (2004). Convergence of the stochastic mesh estimator for pricing Bermudan options. *Journal of Computational Finance* **7**(4), 73–91.

Avramidis, A. N., Zinchenko, Y., Coleman, T., and Verma, A. (2000). Efficiency improvements for pricing American options with a stochastic mesh: Parallel implementation. Cornell Theory Center Research Report, Cornell University, Ithaca, New York.

Barraquand, J., and Martineau, D. (1995). Numerical valuation of high dimensional multivariate American securities. *Journal of Financial and Quantitative Analysis* **30**(3), 383–405.

Boyle, P., Broadie, M., and Glasserman, P. (1997). Monte Carlo methods for security pricing. *Journal of Economic Dynamics and Control* **21**(8–9), 1267–322.

Boyle, P., Evnine, J., and Gibbs, S. (1989). Numerical evaluation of multivariate contingent claims. *Review of Financial Studies* **2**, 241–50.

Boyle, P., Kolkiewicz, A., and Tan, K. S. (2000). Pricing American style options using low discrepancy mesh methods. Technical Report IIPR 00-07, University of Waterloo.

Boyle, P., Kolkiewicz, A., and Tan, K. S. (2002). Pricing American derivatives using simulation: A biased low approach. In *Monte Carlo and Quasi-Monte Carlo Methods 2000.* K.-T. Fang, F. J. Hickernell, and H. Niederreiter, eds, Springer-Verlag, Berlin, pp. 181–200.

Bratley, P., and Fox B. (1988). ALGORITHM 659: Implementing Sobol's quasirandom sequence generator. *ACM Transactions on Mathematical Software* 14, 88–100.

Broadie, M., and Detemple, J. (1996). American option valuation: New bounds, approximations, and a comparison of existing methods. *Review of Financial Studies* **9**(4), 1211–50.

Broadie, M., and Glasserman, P. (1996). Estimating security price derivatives using simulation, *Management Science* **42**, 269–85.

Broadie, M., and Glasserman, P. (1997). Pricing American-style securities using simulation. *Journal of Economic Dynamics and Control* **21**(8–9), 1323–52.

Broadie, M., Glasserman, P., and Ha, Z. (2000). Pricing American options by simulation using a stochastic mesh with optimized weights. In *Probabilistic Constrained Optimization: Methodology and Applications*. S. Uryasev, ed., Kluwer Academic Publishers, Norwell, Mass., pp. 32–50.

Broadie, M., Glasserman, P., and Jain, G. (1997). Enhanced Monte Carlo estimates for American option prices. *Journal of Derivatives* **5**(1), 25–44.

Carrière, J. (1996). Valuation of early-exercise price of options using simulations and nonparametric regression. *Insurance: Mathematics and Economics* **19**, 19–30.

Glasserman, P. (2004). *Monte Carlo Methods in Financial Engineering*. Springer-Verlag, New York.

Glasserman, P., and Yu, B. (2003). Simulation for pricing American options: Regression now or regression later? In *Monte Carlo and Quasi-Monte Carlo Methods 2002*. H. Niederreiter, ed., Springer-Verlag, Berlin, pp. 213–26.

Gut, A. (1988). *Stopped Random Walks*. Springer-Verlag, New York.

Haugh, M., and Kogan, L. (2004). Pricing American options: A duality approach. *Operations Research* **52**(2) 258–70.

He, H. (1990). Convergence from discrete- to continuous-time contingent claim prices. *Review of Financial Studies* **3**, 523–46.

Jamshidian, F. (2003). Minimax optimality of Bermudan and American claims and their Monte Carlo upper bound approximation. Working paper, NIB Capital, The Hague, Netherlands.

Johnson, H. (1987). Options on the maximum or the minimum of several assets. *Journal of Financial and Quantitative Analysis* **22**, 227–83.

Judd, K. (1998). *Numerical Methods in Economics*. MIT Press, Cambridge, Mass.

Kamrad, B., and Ritchken, P. (1991). Multinomial approximating models for options with k state variables. *Management Science* **37**, 1640–52.

Kaniel, R., Tompaidis, S., and Zemlianov, A. (2003). Efficient computation of hedging parameters for discretely exercisable options. Working paper, University of Texas, Austin.

Karatzas, I. (1988). On the pricing of American options. *Applied Mathematics and Optimization* **17**, 37–60.

Kloeden, P., and Platen, E. (1999). *Numerical Solution of Stochastic Differential Equations*, 3rd edn. Springer-Verlag, New York.

Leisen, D., and Reimer, M. (1996). Binomial models for option valuation – examining and improving convergence. *Applied Mathematical Finance* **3**(4), 319–46.

Longstaff, F.A., and Schwartz, E.S. (2001). Valuing American options by simulation: A simple

least-squares approach. *Review of Financial Studies* **14**, 113–47.

Piterbarg, V. (2003). Computing deltas of callable Libor exotics in forward Libor models. *Journal of Computational Finance*, to appear.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*, 2nd edn. Cambridge University Press.

Raymar, S., and Zwecher, M. (1997). A Monte Carlo valuation of American call options on the maximum of several stocks. *Journal of Derivatives* **5**(1), 7–23.

Rogers, L. C. G. (2002). Monte Carlo valuation of American options. *Mathematical Finance* **12**, 271–86.

Rust, J. (1997). Using randomization to break the curse of dimensionality. *Econometrica* **65**(3), 487–516.

Stulz, R. (1982). Options on the minimum or the maximum of two risky assets. *Journal of Financial Economics* **10**, 161–85.

Tilley, J. A. (1993). Valuing American options in a path simulation model. *Transactions of the Society of Actuaries* **45**, 83–104.

Tsitsiklis, J., and Van Roy, B. (1999). Optimal stopping of Markov processes: Hilbert space theory, approximation algorithms, and an application to pricing high-dimensional financial derivatives. *IEEE Transactions on Automatic Control* **44**, 1840–51.