Pricing American Options: A Duality Approach*

Martin B. Haugh† and Leonid Kogan‡

December 2001

Abstract

We develop a new method for pricing American options. The main practical contribution of this paper is a general algorithm for constructing upper and lower bounds on the true price of the option using any approximation to the option price. We show that our bounds are tight, so that if the initial approximation is close to the true price of the option, the bounds are also guaranteed to be close. We also explicitly characterize the worst-case performance of the pricing bounds. The computation of the lower bound is straightforward and relies on simulating the suboptimal exercise strategy implied by the approximate option price. The upper bound is also computed using Monte Carlo simulation. This is made feasible by the representation of the American option price as a solution of a properly defined dual minimization problem, which is the main theoretical result of this paper. Our algorithm proves to be accurate on a set of sample problems where we price call options on the maximum and the geometric mean of a collection of stocks. These numerical results suggest that our pricing method can be successfully applied to problems of practical interest.

*An earlier draft of this paper was titled *Pricing High-Dimensional American Options: A Duality Approach* . We thank Mark Broadie, Domenico Cuoco, Andrew Lo, Jacob Sagi, Jiang Wang and seminar participants at Columbia University, INSEAD, London Business School, MIT, Princeton, Stanford, and Wharton for valuable comments. We thank an anonymous referee for pointing out the alternative proof of Theorem 1 in Section 3.1. Financial support from the MIT Laboratory for Financial Engineering, the Rodney L. White Center for Financial Research at the Wharton School, and TIAA-CREF is gratefully acknowledged. We are grateful to Anargyros Papageorgiou for providing the *FINDER* software to generate the low discrepancy sequences used in this paper.

†Operations Research Center, MIT, Cambridge, MA 02139; email: mhaugh@alum.mit.edu.

‡Corresponding author: Sloan School of Management, MIT, 50 Memorial Drive, E52-455, Cambridge, MA 02142; email: lkogan@mit.edu.

1 Introduction

Valuation and optimal exercise of American options remains one of the most challenging practical problems in option pricing theory. The computational cost of traditional valuation methods, such as lattice and tree-based techniques, increases rapidly with the number of underlying securities and other payoff-related variables. Due to this well-known curse of dimensionality, practical applications of such methods are limited to problems of low dimension.

In recent years, several methods have been proposed to address this curse of dimensionality. Instead of using traditional deterministic approaches, these methods use Monte Carlo simulation to estimate option prices. Bossaerts (1989) and Tilley (1993) are among the first attempts to use simulation techniques for pricing American options. Other important work in this literature includes Barraquand and Martineau (1995), Carriere (1996), Raymar and Zwecher (1997), Ibanez and Zapatero (1999) and Garcia (1999). Longstaff and Schwartz (2001) and Tsitsiklis and Van Roy (1999, 2000) have proposed an approximate dynamic programming approach that can compute good price estimates and is very fast in practice. Tsitsiklis and Van Roy also provide theoretical results that help explain the success of approximate dynamic programming methods. The price estimates these techniques construct, however, typically only give rise to lower bounds on the true option price. As a result, there is usually no formal method for evaluating the accuracy of the price estimates.

In an important contribution to the literature, Broadie and Glasserman (1997a,b) develop two stochastic mesh methods for American option pricing. One of the advantages of their procedure over the previously proposed methods is that it allows them to generate both lower and upper bounds on the option price that converge asymptotically to the true option price. Their bounds are based on an application of Jensen's inequal-

ity and can be evaluated by Monte Carlo simulations. However, such bounds do not necessarily generalize to other pricing methods. The complexity of their first method is exponential in the number of exercise periods. The second approach does not suffer from this drawback but nonetheless appears to be computationally demanding. In an effort to address this drawback, Boyle, Kolkiewicz and Tan (2001) generalize the stochastic mesh method of Broadie and Glasserman (1997b) using low discrepancy sequences to improve the efficiency of the approach.

The main practical contribution of this paper is a general algorithm for constructing upper and lower bounds on the true price of the option using any approximation to the option price. We show that our bounds are tight, so that if the initial approximation is close to the true price of the option, the bounds are also guaranteed to be close. In addition, we explicitly characterize the worst-case performance of the pricing bounds. The computation of the lower bound is straightforward and relies on simulating the suboptimal exercise strategy implied by the approximate option price. The upper bound is obtained by simulating a different stochastic process that is determined by choosing an appropriate supermartingale. We justify this procedure by representing the American option price as a solution of a dual minimization problem, which is the main theoretical result of this paper.

In order to determine the option price approximation underlying the estimation of bounds, we also implement a fast and accurate valuation method based on approximate dynamic programming (see Bertsekas and Tsitsiklis 1996) where we use non-linear regression techniques to approximate the value function. Unlike most procedures that use Monte Carlo simulation to estimate the continuation value of the option, our method is deterministic and relies on low discrepancy sequences as an alternative to Monte Carlo simulation. For the examples considered in this paper, we find that low discrepancy

sequences provide a significant computational improvement over simulation.

While the duality-based approach to portfolio optimization problems has proved successful and is now widely used in finance, (see, for example, Karatzas and Shreve 1998), the dual approach to the American option pricing problem does not seem to have been previously developed other than in recent independent work by Rogers (2001). Rogers establishes a dual representation of American option prices similar to ours and applies the new representation to compute upper bounds on several types of American options using Monte Carlo simulation. However, he does not provide a formal systematic approach for generating tight upper bounds and his computational approach is problem specific.

Andersen and Broadie (2001) use the methodology developed in this paper to formulate another computational algorithm based on Monte Carlo simulation. A distinguishing feature of their approach is the use of an approximate *exercise policy*, as opposed to an approximate *option price*, to estimate the bounds on the true price of the option. They also observe that a straightforward modification (taking the martingale part of the supermartingale) of the stochastic process used to estimate the upper bound leads to more accurate estimates. This observation also applies to the algorithm we use in this paper where we begin with an initial approximation to the option price. This led to a significant improvement in the computational results that were presented in an earlier draft of this paper.

The algorithm of Andersen and Broadie is quadratic in the number of exercise periods, given knowledge of the approximate exercise policy, while our approach is linear but requires knowledge of the approximate option price. In general, constructing an accurate approximation to the option price is a more challenging task than approximating the optimal exercise policy. In the absence of formal complexity results, we cannot

conclude that one approach is generally more efficient than the other. Nor is it clear which algorithm should be preferred in a given application. That would depend on how difficult it is to construct an accurate approximation to the option price as opposed to the exercise policy, as well as on the number of exercise periods. Careful comparison of the two algorithms is beyond the scope of this paper and remains an open question for future research.

The rest of the paper is organized as follows. In Section 2 we formulate the problem. In Section 3, we derive the new duality result for American options and use it to derive an upper bound on the option price. In Section 4 we describe the implementation of the algorithm. We report numerical results in Section 5 and we conclude in Section 6.

2 Problem Formulation

In this section we formulate the American option pricing problem.

Information Set. We consider an economy with a set of dynamically complete financial markets, described by the underlying probability space, Ω, the sigma algebra $\mathcal{F}$, and the risk-neutral valuation measure $\mathcal{Q}$. It is well known (see Harrison and Kreps 1979) that if financial markets are dynamically complete, then under mild regularity assumptions there exists a unique risk-neutral measure, allowing one to compute prices of all state-contingent claims as the expected value of their discounted cash flows. The information structure in this economy is represented by the augmented filtration $\{\mathcal{F}_t : t \in [0,T]\}$. More specifically, we assume that $\mathcal{F}_t$ is generated by Z_t, a d-dimensional standard Brownian motion, and the state of the economy is represented by an $\mathcal{F}_t$-adapted Markovian process $\left\{X_t \in \Re^d : t \in [0,T]\right\}$.

Option Payoff. Let $h_t = h(X_t)$ be a nonnegative adapted process representing the payoff of the option, so that the holder of the option receives h_t if the option is exercised

at time t. We also define a riskless account process $B_t = \exp\left(\int_0^t r_s ds\right)$, where r_t denotes the instantaneously risk-free rate of return. We assume that the discounted payoff processes satisfies the following integrability condition

$$\mathrm{E}_0\left[\max_{t=0,1,\ldots,T}\left|\frac{h_t}{B_t}\right|\right] < \infty \tag{1}$$

where $\mathrm{E}_t[\cdot]$ denotes the expected value under the risk-neutral probability measure, conditional on the time t information, $\mathcal{F}_t$.

Exercise Dates. The American feature of the option allows its holder to exercise it at any of the pre-specified exercise dates in $\mathcal{T} = \{0, 1, \ldots, T\}$, equally spaced between 0 and T. Equal spacing of the exercise dates is assumed to simplify the notation and is not restrictive. Moreover, a unit time increment within the model can be mapped into any period of calendar time.

Option Price. The value process of the American option, V_t, is the price process of the option conditional on it not having been exercised before t. It satisfies

$$V_t = \sup_{\tau \geq t} \mathrm{E}_t\left[\frac{B_t h_\tau}{B_\tau}\right]. \tag{2}$$

where τ is any stopping time with values in the set $\mathcal{T} \cap [t, T]$. This is a well-known characterization of American options (see, for example, Bensoussan 1984, Karatzas 1988 and Pliska 1997).

3 Theory

Our approach to the American option pricing problem consists of the following steps.

Step 1. Compute an approximation to the market price of the option as a function of the time and state. Specifically, we use an approximate dynamic programming algorithm to determine the continuation value of the option, i.e., the value of the option conditional

on not exercising it at the current time period.

Step 2. Estimate the lower bound on the option price by simulating the approximate exercise strategy based on the option price approximation from Step 1.

Step 3. Based on the option price approximation, define a martingale process and use it to estimate the upper bound by Monte Carlo simulation. This last step is based on the new dual representation of the option price presented below.

We begin this section with our main theoretical result on the dual representation of the American option price. We then show how to use this price characterization to compute bounds on the option price, and study the properties of these bounds.

3.1 The Dual Problem

The problem of pricing an American option, the *primal* problem, is that of computing

$$V_0 = \sup_{\tau \in \mathcal{T}} \mathrm{E}_0 \left[\frac{h_\tau}{B_\tau} \right]. \tag{P}$$

For an arbitrary adapted supermartingale, π_t, we define the *dual* function $F(t, \pi)$ as

$$\frac{F(t,\pi)}{B_t} := \mathrm{E}_t \left[\max_{s \in \{[t,T] \cap \mathcal{T}\}} \left(\frac{h_s}{B_s} - \pi_s \right) \right] + \pi_t \tag{3}$$

Then the dual problem is to minimize the dual function at time 0 over all supermartingales, π_t. Let U_0 denote the optimal value of the dual problem, so that

$$U_0 = \inf_{\pi} \; F(0,\pi) = \inf_{\pi} \; \mathrm{E}_0 \left[\max_{t \in \mathcal{T}} \left(\frac{h_t}{B_t} - \pi_t \right) \right] + \pi_0 \tag{D}$$

The following theorem shows that the optimal values of the dual and primal problems coincide.

Theorem 1 ***(Duality Relation)*** *The optimal values of the primal problem (P) and the dual problem (D) are equal, i.e., $V_0 = U_0$. Moreover, an optimal solution of the dual*

problem is given by $\pi_t^* = V_t/B_t$*, where* V_t *is the value process for the American option,*

$$V_t = \sup_{\tau \in \{[t,T] \cap \mathcal{T}\}} \mathrm{E}_t \left[\frac{h_\tau B_t}{B_\tau} \right].$$

Proof For any supermartingale π_t,

$$\begin{aligned} V_0 &= \sup_{\tau \in \mathcal{T}} \mathrm{E}_0 \left[\frac{h_\tau}{B_\tau} \right] = \sup_{\tau \in \mathcal{T}} \mathrm{E}_0 \left[\frac{h_\tau}{B_\tau} - \pi_\tau + \pi_\tau \right] \leq \sup_{\tau \in \mathcal{T}} \mathrm{E}_0 \left[\frac{h_\tau}{B_\tau} - \pi_\tau \right] + \pi_0 \\ &\leq \mathrm{E}_0 \left[\max_{t \in \mathcal{T}} \left(\frac{h_t}{B_t} - \pi_t \right) \right] + \pi_0 \end{aligned} \tag{4}$$

where the first inequality follows from the optional sampling theorem for supermartingales (see Billingsley 1995) and condition (1). Taking the infimum over all supermartingales, π_t, on the right hand side of (4) implies $V_0 \leq U_0$. On the other hand, the process V_t/B_t is a supermartingale, which implies

$$U_0 \leq \mathrm{E}_0 \left[\max_{t \in \mathcal{T}} \left(h_t/B_t - V_t/B_t \right) \right] + V_0.$$

Since $V_t \geq h_t$ for all t, we conclude that $U_0 \leq V_0$. Therefore, $V_0 = U_0$, and equality is attained when $\pi_t^* = V_t/B_t$. ■

Theorem 1 shows that an upper bound on the price of the American option can be constructed simply by evaluating the dual function over an arbitrary supermartingale π_t. In particular, if such a supermartingale satisfies $\pi_t \geq h_t/B_t$, the option price V_0 is bounded above by π_0. Theorem 1 therefore implies the following well-known characterization of the American option price (see, for example, Pliska 1997).

Proposition 1 ***(Option Price Characterization)*** *The discounted option price process* V_t/B_t *is the smallest supermartingale that dominates the discounted payoff of the option at all exercise periods.*

The reverse is also true, i.e., one can use Proposition 1 to prove Theorem 1. Note that since both processes on the right-hand side of (3) are supermartingales, so is the discounted dual function $F(t,\pi)/B_t$. Clearly $F(t,\pi) \geq h_t/B_t$ for any t, implying that $F(0,\pi) \geq V_0$ by Proposition 1. When π_t is chosen to be the discounted option price, the reverse inequality holds, implying that the values of the primal and the dual problems coincide. Theorem 1 therefore expresses the well-known result of Proposition 1 in a new constructive form, which we use to evaluate the upper bound on the option price.

The pricing problem is closely related to the problem of dynamic replication of the American option, which is equally important in practice. While various methods for approximating American option prices have been suggested in the literature, computing reliable replication strategies has remained a challenging problem. The result of Theorem 1 suggests a method for super-replicating the American option.

Since the discounted dual function $F(t,\pi)/B_t$ is a supermartingale and financial markets in our model are dynamically complete, there exists a self-financing trading strategy with an initial cost $F(0,\pi)$ which almost surely dominates $F(t,\pi)$ (see Duffie 1996, Section 2.I). Such a trading strategy super-replicates the payoff of the American option, since $F(t,\pi)$ dominates the price of the option and hence its payoff at exercise. Using an approximation to the option price, we can define π_t and $F(t,\pi)$ explicitly, so that super-replicating the option can be a relatively straightforward task. In particular, Boyle et al. (1997) and Garcia et al. (2000) describe Monte Carlo methods for estimating the portfolio strategies replicating the present value process of a state contingent claim. This claim could correspond to a derivative security or some consumption process. Their results are directly applicable to (3).

3.2 The Upper Bound on the Option Price

When the supermartingale π_t in (3) coincides with the discounted option value process, V_t/B_t, the upper bound $F(0,\pi)$ equals the true price of the American option. This suggests that a tight upper bound can be obtained from an accurate approximation, $\widetilde{V}_t$, by defining π_t in such a way that when the approximate option price, $\widetilde{V}_t$, coincides with the exact price, V_t, π_t equals the discounted option price, V_t/B_t. It seems natural then to use either of the following two definitions of π_t:

$$\pi_0 = \widetilde{V}_0 \tag{5}$$

$$\pi_{t+1} = \pi_t + \frac{\widetilde{V}_{t+1}}{B_{t+1}} - \frac{\widetilde{V}_t}{B_t} - \mathrm{E}_t\left[\frac{\widetilde{V}_{t+1}}{B_{t+1}} - \frac{\widetilde{V}_t}{B_t}\right]^+ \tag{6}$$

or

$$\pi_{t+1} = \pi_t\left(\frac{B_t\widetilde{V}_{t+1}}{B_{t+1}\widetilde{V}_t} - \mathrm{E}_t\left[\frac{B_t\widetilde{V}_{t+1}}{B_{t+1}\widetilde{V}_t} - 1\right]^+\right) \tag{7}$$

where $(x)^+ := \max(x,0)$. By construction, $\mathrm{E}_t\left[\pi_{t+1} - \pi_t\right] \leq 0$ for either definition of π_t, implying π_t is an adapted supermartingale for any approximation, $\widetilde{V}_t$. Also, when $\widetilde{V}_t = V_t$, $\pi_t = V_t/B_t$, because the latter process is a supermartingale and the positive part of expectations in (5,6) and (5,7) equals zero. While we cannot claim a priori that the upper bound corresponding to (5,6) is superior to the bound determined by (5,7), the properties of the bound under the first definition are considerably easier to analyze.

Note also that the upper bound can be tightened further by omitting the positive part in the definition of π_t. The resulting process is a martingale (the martingale part of π_t) and therefore a supermartingale, so that it too can be used to construct an upper bound. It coincides with π_t at $t = 0$ and is always greater than or equal to π_t for $t > 0$. It therefore leads to a lower value of the upper bound defined by (3). (In an earlier draft of this paper, we used the formulation (5,6) to compute upper bounds on the option

price. Andersen and Broadie (2001) point out, however, that in general tighter upper bounds can be obtained by using the martingale component of the supermartingale, π_t. In our framework, the martingale component is obtained by simply omitting the positive in (5,6)).

For the remainder of the paper we will therefore take π_t to be defined by

$$\pi_0 = \widetilde{V}_0 \tag{8}$$

$$\pi_{t+1} = \pi_t + \frac{\widetilde{V}_{t+1}}{B_{t+1}} - \frac{\widetilde{V}_t}{B_t} - \mathrm{E}_t\left[\frac{\widetilde{V}_{t+1}}{B_{t+1}} - \frac{\widetilde{V}_t}{B_t}\right]. \tag{9}$$

Let $\overline{V}_0$ denote the upper bound corresponding to (8) and (9). Then it is easy to see that the upper bound is explicitly given by

$$\overline{V}_0 = \widetilde{V}_0 + \mathrm{E}_0\left[\max_{t\in\mathcal{T}}\left(\frac{h_t}{B_t} - \frac{\widetilde{V}_t}{B_t} + \sum_{j=1}^{t}\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]\right)\right]. \tag{10}$$

The following theorem relates the worst-case performance of the upper bound determined by (8) and (9) to the accuracy of the original approximation, $\widetilde{V}_t$.

Theorem 2 ***(Tightness of the Upper Bound)*** *Consider any approximation to the American option price, $\widetilde{V}_t$, satisfying $\widetilde{V}_t \geq h_t$. Then*

$$\overline{V}_0 \leq V_0 + 2\ \mathrm{E}_0\left[\sum_{t=0}^{T}\frac{\left|\widetilde{V}_t - V_t\right|}{B_t}\right]. \tag{11}$$

Proof See Appendix A.1. ■

Theorem 2 suggests two possible reasons for why the upper bound may be limited in practice. First, it suggests that the bound may deteriorate linearly in the number of exercise periods. However, this is a worst case bound. Indeed, the quantity of interest is

$$\mathrm{E}_0\left[\max_{t\in\mathcal{T}}\left(\frac{h_t}{B_t} - \frac{\widetilde{V}_t}{B_t} + \sum_{j=1}^{t}\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]\right)\right] \tag{12}$$

and while we would expect it to increase with the number of exercise periods, it is not clear that it should increase linearly. This is particularly true for pricing American options when we typically want to keep the horizon, T, fixed, while we let the number of exercise times in $[0, T]$ increase.

The second reason is due to the approximation error, $|\widetilde{V}_t - V_t|/B_t$. Tsitsiklis and Van Roy (2000) have shown that under certain conditions, and for certain approximate dynamic programming algorithms, this error can be bounded above by a constant, independent of the number of exercise periods. This result, however, is only applicable to perpetual options since it assumes that $T \to \infty$ while the interval between exercise times remains constant. As mentioned in the previous paragraph, however, we are typically interested in problems where T is fixed and the interval between exercise periods decreases. In this case, Tsitsiklis and Van Roy show that the approximation error is bounded above by a constant times $\sqrt{N}$, where N is the number of exercise periods.

These two observations suggest that the quality of the upper bound should deteriorate with N, but not in a linear fashion. In Section 5 we shall see evidence to support this when we successfully price options with as many as 100 exercise periods.

3.3 The Lower Bound on the Option Price

To construct a lower bound on the true option price, we define the Q-value function to be

$$Q_t(X_t) := \mathrm{E}_t \left[\frac{B_t}{B_{t+1}} V_{t+1}(X_{t+1}) \right] \tag{13}$$

The Q-value at time t is simply the expected value of the option, conditional on it not being exercised at time t. Suppose that an approximation to the Q-value, $\widetilde{Q}_t(X_t)$, is available, for $t = 1, \ldots, T-1$. Then, to compute the lower bound on the option price, we simulate a number of sample paths originating at X_0. For each sample path, we find

the first exercise period t, if it exists, in which $h(X_t) \geq \widetilde{Q}_t(X_t)$. The option is then exercised at this time and the discounted payoff of the path is given by $h(X_t)/B_t$. Since this is a feasible $\mathcal{F}_t$ - adapted exercise policy, it is clear that the expected discounted payoff from following this policy defines a lower bound, $\underline{V_0}$, on the true option price, V_0. Formally, $\widetilde{\tau} = \min\{t \in \mathcal{T} : \widetilde{Q}_t \leq h_t\}$ and

$$\underline{V_0} = \mathrm{E}_0 \left[\frac{h_{\widetilde{\tau}}}{B_{\widetilde{\tau}}} \right].$$

The following theorem characterizes the worst-case performance of the lower bound.

Theorem 3 ***(Tightness of the Lower Bound)*** *The lower bound on the option price satisfies*

$$\underline{V_0} \geq V_0 - \mathrm{E}_0 \left[\sum_{t=0}^{T} \frac{|\widetilde{Q}_t - Q_t|}{B_t} \right]. \tag{14}$$

Proof See Appendix A.2. ■

While this theorem suggests that the performance of the lower bound may deteriorate linearly in the number of exercise periods, numerical experiments indicate that this is not the case. Theorem 3 describes the worst case performance of the bound. However, in order for the exercise strategy that defines the lower bound to achieve the worst case performance, it is necessary that at each exercise period the option is mistakenly left unexercised, i.e., the condition $Q_t(X_t) < h(X_t) < \widetilde{Q}_t(X_t)$ is satisfied. For this to happen, it must be the case that at *each* exercise period, the underlying state variables are close to the optimal exercise boundary. In addition, $\widetilde{Q}_t$ must systematically overestimate the true value Q_t so that the option is not exercised while it is optimal to do so. In practice, the variability of the underlying state variables, X_t, might suggest that X_t spends little time near the optimal exercise boundary. This suggests that as long as $\widetilde{Q}_t$ is a good approximation to Q_t near the optimal exercise frontier, the lower bound should be a

good estimate of the true price, regardless of the number of exercise periods.

4 Implementation

In this section we describe in some detail approximate Q-value iteration, the algorithm that we use for obtaining the initial approximation to the value function, V_t. Algorithms of this kind are now standard in the approximate dynamic programming literature (see for example, Bertsekas and Tsitsiklis, 1996). An interesting feature of the particular algorithm we describe is that, in contrast to most approximate dynamic programming algorithms, it is deterministic. This deterministic property is achieved through the use of *low discrepancy sequences*. While such sequences are used in the same spirit as independent and identically distributed sequences of random numbers, we found that their application significantly improved the computational efficiency of the algorithm. They are described in further detail in Appendix B.

4.1 Q-Value Iteration

As before, the problem is to compute

$$V_0 = \sup_{\tau \in \mathcal{T}} \mathrm{E}_0 \left[\frac{h_\tau}{B_\tau} \right].$$

In theory this problem is easily solved using value iteration where we solve for the value functions, V_t, recursively so that

$$V_T = h(X_T) \tag{15}$$

$$V_t = \max \left(h(X_t), \mathrm{E}_t \left[\frac{B_t}{B_{t+1}} V_{t+1}(X_{t+1}) \right] \right). \tag{16}$$

The price of the option is then given by $V_0(X_0)$ where X_0 is the initial state of the economy. In practice, however, if d is large so that X_t is high dimensional, then the

'curse of dimensionality' implies that value iteration is not practical.

As an alternative to value iteration consider again the Q-value function, which was defined earlier to be the continuation value of the option

$$Q_t(X_t) = \mathrm{E}_t \left[\frac{B_t}{B_{t+1}} V_{t+1}(X_{t+1}) \right]. \tag{17}$$

The value of the option at time $t+1$ is then

$$V_{t+1}(X_{t+1}) = \max(h(X_{t+1}), Q_{t+1}(X_{t+1})) \tag{18}$$

so that we can also write

$$Q_t(X_t) = \mathrm{E}_t \left[\frac{B_t}{B_{t+1}} \max\left(h(X_{t+1}), Q_{t+1}(X_{t+1})\right) \right]. \tag{19}$$

Equation (19) clearly gives a natural extension of value iteration to Q-value iteration. The algorithm we use in this section consists of performing an approximate Q-value iteration.

There are a number of reasons for why it is preferable to concentrate on approximating Q_t rather than approximating V_t directly. Letting $\widetilde{Q}_t$ and $\widetilde{V}_t$ denote our estimates of Q_t and V_t respectively, we can write the defining equations for approximate Q-value and value iteration as follows:

$$\widetilde{Q}_t(X_t) = \mathrm{E}_t \left[\frac{B_t}{B_{t+1}} \max\left(h(X_{t+1}), \widetilde{Q}_{t+1}(X_{t+1})\right) \right] \tag{20}$$

$$\widetilde{V}_t(X_t) = \max\left(h(X_t), \mathrm{E}_t \left[\frac{B_t}{B_{t+1}} \widetilde{V}_{t+1}(X_{t+1}) \right] \right). \tag{21}$$

The functional forms of (20) and (21) suggest that Q_t is smoother than V_t, and therefore more easily approximated.

More importantly, however, Q_t is the unknown quantity of interest, and the decision to exercise or continue at a particular point will require a comparison of $\widetilde{Q}_t(X_t)$ and

$h(X_t)$. If we only have $\widetilde{V}_t$ available to us then such a comparison will often be very difficult to make. For example, if $\widetilde{V}_t(X_t) > h(X_t)$ then we do not exercise the option. However, if $\widetilde{V}_t(X_t)$ is only marginally greater than $h(X_t)$, then it may be the case that $h(X_t) > Q_t(X_t)$ and $\widetilde{V}_t(X_t)$ is actually attempting to approximate $h(X_t)$. In this situation, we misinterpret $\widetilde{V}_t(X_t)$ and assume that it is optimal to continue when in fact it is optimal to exercise. This problem can be quite severe when there are relatively few exercise periods because in this case, there is often a significant difference between the value of exercising now and the continuation value. When we have a direct estimate of $Q_t(X_t)$ available, this problem does not arise.

4.2 Approximate Q-Value Iteration

The first step in approximate Q-value iteration is to select an *approximation architecture*, $\left\{\widetilde{Q}_t(.;\beta_t) : \beta_t \in \Re^N\right\}$, which is a class of functions from which we select $\widetilde{Q}_t$. This class is parametrized by the vector $\beta_t \in \Re^N$ so that the problem of determining $\widetilde{Q}_t$ is reduced to the problem of selecting β_t where β_t is chosen to minimize some approximation error. The choice of architecture does not seem to be particularly important as long as it is sufficiently 'rich' to accurately approximate the true value value.

Possible architectures are the linearly parametrized architectures of Longstaff and Schwartz (2000) and Tsitsiklis and Van Roy (2000), or non-linearly parametrized architectures such as neural networks or support vector machines (see Vapnik 1999). In this paper we use a multi-layered perceptron with a single hidden layer, a particular class of neural networks. Multi-layered perceptrons with a single hidden layer are known to possess the universal approximation property so that they are able to approximate any continuous function over a compact set arbitrarily well, provided that a sufficient number of neurons are used (see Hornick, Stinchcombe and White 1989).

The second step in the procedure is to select for each $t = 1, \ldots, K-1$, a set

$$S_t := \left\{ P_1^t, \ldots, P_{N_t}^t \right\}$$

of *training points* where $P_n^t \in \Re^d$ for $n = 1, \ldots, N_t$. It makes sense to choose the sets S_t in such a way that they are representative of the probability distribution of X_t. We do this using low discrepancy sequences so that if N_t is the desired number of training points, then we simply take N_t points from a particular low discrepancy sequence. Using the technique described in Appendix B, it is then straightforward to convert these points into training points that represent the distribution of X_t. Of course, it is also possible to select the training points by simply simulating from the distribution of the state vector, X_t. Our limited experience shows that both simulation and low discrepancy sequences work very well in practice. The performace of the low discrepancy sequences, however, appeared to be marginally superior when applied to the problems we consider in this paper.

The third step is to perform a *training point evaluation*. Defining $\widetilde{Q}_T \equiv 0$, we begin with $t = K-1$ and for $n = 1, \ldots, N_t$, we use $\hat{Q}_t(P_n^t)$ to estimate $Q_t(P_n^t)$ where

$$\hat{Q}_t(P_n^t) := \widehat{\mathrm{E}}_t \left[\frac{B_t}{B_{t+1}} \max \left(h(X_{t+1}), \widetilde{Q}_{t+1}(X_{t+1}) \right) \right]. \tag{22}$$

The operator $\widehat{\mathrm{E}}[\ .\ ]$ in (22) is intended to approximate the expectation operator, $\mathrm{E}[\ .\ ]$. This is necessary as it is usually not possible to compute the expectation exactly on account of the high dimensionality of the state space. For example, $\hat{\mathrm{E}}[\ .\ ]$, could correspond to Monte Carlo simulation where we simulate from the distribution of X_{t+1} given that $X_t = P_n^t$. Then $\hat{\mathrm{E}}[\ .\ ]$ is defined by

$$\hat{\mathrm{E}}_t \left[\frac{B_t}{B_{t+1}} \max \left(h(X_{t+1}), \widetilde{Q}_{t+1}(X_{t+1}) \right) \right] := \frac{B_t}{M B_{t+1}} \sum_{l=1}^{M} \max \left(h(x_l), \widetilde{Q}_{t+1}(x_l) \right) \tag{23}$$

where M x_l's are drawn randomly from the conditional distribution of X_{t+1}. The problem with this method is that the rate of convergence to the true expectation is $O(\frac{1}{\sqrt{M}})$ which can be too slow for our purposes. Instead, we use a low discrepancy sequence to generate the x_l's, with M chosen in advance. For the 5-dimensional problems of Section 5, M was set equal to 1000. For the 10-dimensional problems, we set M equal to 2000. Had we used Monte Carlo simulation, then in order to achieve a comparable level of accuracy, a substantially larger value of M would have been required.

Finally, we estimate Q_t with $\widetilde{Q}_t(,;\hat{\beta}_t)$ where

$$\widetilde{Q}_t(x;\beta_t) = \sum_{j=1}^{K} r_t(j)\theta\left(b_t(j) + \sum_{l=1}^{d}(r_t(j,l)x(l))\right) \tag{24}$$

and

$$\hat{\beta}_t = \arg\min_{\beta_t} \sum_{n=1}^{K}\left(\hat{Q}_t(P_n^t) - \widetilde{Q}_t(P_n^t;\beta_t)\right)^2. \tag{25}$$

The parameters $b_t(j)$, $r_t(j)$ and $r_t(j,l)$ in (24) constitute the parameter vector β_t, while $\theta(.)$ denotes the logistic function so that

$$\theta(x) = \frac{1}{1+e^{-x}}.$$

K refers to the number of 'neurons', and d is the dimensionality of the input vector, x. While the input vector x may simply correspond to a sample state vector, it is common to augment x with certain *features*, that is, easily computed functions of the current state vector. From an informational point of view, features add no new information to the neural network. However, they often make it easier for the neural network (or other approximation architecture) to approximate the true value function.

In practice, we usually minimize a variant of the quantity in (25) in order to avoid the difficulties associated with overfitting. This is done using cross validation, an approach

that requires the training points to be divided into three sets, namely training, validation and test sets. Initially, only the training and validation sets are used in the minimization so that the quantity

$$\sum \left(\hat{Q}_t(P_i^t) - \widetilde{Q}_t(P_i^t; \beta_t) \right)^2 \tag{26}$$

is minimized where the sum in (26) is taken over points in the training set. The minimization is performed using the *Levenberg Marquardt* method for least squares optimization (see Bertsekas and Tsitsiklis 1996). At each iteration of the minimization, the error in the validation set is also computed and as long as overfitting is not taking place, then the validation error should decrease along with the training set error. However, if the validation error starts increasing at any point then it is likely that overfitting is taking place. The algorithm then terminates if the validation error increases for a prespecified number of iterations, and $\hat{\beta}_t$ is then set equal to the value of β_t in the last iteration of the minimization before the validation error began to increase.

There is one further difficulty with the neural network architecture that needs to be addressed. The neural network will typically have many local minima and it is often the case that the algorithm will terminate at a local minimum that is far from the global minimum. In this case, it is necessary to repeat the minimization again, this time using a different starting value for β_t. This may be repeated until a satisfactory local minimum has been found, where 'satisfactory' refers to performance on the test set.

We use this training algorithm for finding $\widetilde{Q}_{T-1}$. For the remaining Q-value functions, however, the problem is now somewhat simplified since it is usually the case that $\widetilde{Q}_t \approx \widetilde{Q}_{t-1}$. We can therefore use $\hat{\beta}_t$ as the initial solution for training the time $t-1$ neural network. In practice, this means that the other neural networks can be trained very quickly and that we only need to perform the minimization once. It also means that we

can dispense with the need for having a test set for all but the terminal neural network.

Once $\widetilde{Q}_t$ has been found, we then iterate in the manner of value iteration until we have found $\widetilde{Q}_0$. We could then use

$$\widetilde{V}_0(X_0) = \max\left(h(X_0), \widetilde{Q}_0(X_0)\right) \tag{27}$$

to estimate the value of the option. While such an estimate may be quite accurate, it is of limited value since we can say very little about the estimation error. In addition, we do not have at hand an exercise strategy that has expected value equal to $\widetilde{V}_0(X_0)$ nor do we know if such a strategy even exists. Finally, it provides little information with regards to hedging the option.

4.3 Computing the Upper and Lower Bounds

In Section 3.1 we showed that an upper bound for the price of the American option is given by (10). However, in (10) we can alternatively set $\widetilde{V}_0 = \underline{V_0}$, where $\underline{V_0}$ is the estimated lower bound. (Since $\underline{V_0} \geq h_0$, this new definition satisfies $\widetilde{V}_t \geq h_t$ for all t, and so Theorem 2 continues to hold.) In this case, the upper bound is given by

$$\overline{V}_0 = \underline{V_0} + \mathrm{E}_0\left[\max_{t\in\mathcal{T}}\left(\frac{h_t}{B_t} - \frac{\widetilde{V}_t}{B_t} + \sum_{j=1}^{t}\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]\right)\right]. \tag{28}$$

This then gives a natural decomposition of the upper bound into a sum of two components. The first component is the estimated lower bound, while the second component in some sense measures the extent to which the discounted approximate value function fails to be a supermartingale.

We estimate $\overline{V}_0$ by simulating sample paths of the state variables, evaluating

$$\max_{t\in\mathcal{T}}\left(\frac{h_t}{B_t} - \frac{\widetilde{V}_t}{B_t} + \sum_{j=1}^{t}\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]\right) \tag{29}$$

along each path and taking the average over all paths. Evaluating (29) numerically is time consuming since at each point, (t, X_t), on the path, we need to compute

$$\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]. \tag{30}$$

Any unbiased noisy estimate of the expectation in (30), however, will result in an upwards biased estimate of the upper bound, due to the application of the maximum operator. It is therefore important that accurate estimates of the expectation in (30) can be computed. As before, we again use low discrepancy sequences to estimate these high-dimensional integrals. Since we wish to compute an accurate estimate of the upper bound, it is important to use a good stopping criterion to determine the number of points that will be used to estimate the expectation in (30).

Let $\mathrm{E}(N)$ denote the estimate of (30) when N low discrepancy points are used. For some fixed value of M, we then examine $\mathrm{E}(Mi)$ for $i = 1, \ldots, L$ and terminate either when

$$|\mathrm{E}(M(i+1)) - \mathrm{E}(Mi)| < \epsilon \tag{31}$$

or when $i = L$, if the condition in (31) is not satsified for any $i < L$.

In the numerical results of Section 5, we set M equal to 2000 and 4000 for the 5 and 10-dimensional problems, respectively. For all problems, we set $\epsilon = .2$ cents and $L = 48000$. These values typically result in estimates of the upper bound that appear to be very accurate for practical purposes. This observation is based upon further numerical experiments where M and ϵ were increased and decreased respectively. The results of these experiments invariably resulted in estimates of the upper bound that were within 1 or 2 cents of, and often considerably closer to, the original estimate.

One possible concern about the use of low discrepancy sequences for estimating

the upper bound is that even a small bias in the estimate of the expectation in (30) might result in a considerable bias in the estimate of the upper bound when there are many exercise periods. This is not a problem, however, for reasons mentioned before. In particular, as the number of exercise periods increases, the time interval between exercise periods decreases which means that the variability of the value function in the next period also decreases. As a result, the conditional expectation in (30) can be estimated more accurately for fixed values of M and ϵ.

4.4 An Automated Pricing Algorithm

Perhaps the obvious way to compute the lower and upper bounds is in a sequential fashion so that after estimating the Q-value functions, we simulate a number of sample paths to compute the lower bound, and simulate another set to compute the upper bound. One difficulty with this strategy, however, is that the difference between $\underline{V_0}$ and $\overline{V}_0$ might be significant so that there is a large duality gap.

When this occurs, we are forced to re-estimate the Q_t's, possibly using more training points or a more flexible approximation architecture. This process may need to be repeated a number of times before we obtain a sufficiently small duality gap. For American options that need to be priced regularly, this is unlikely to be a problem since experience should allow us to determine in advance the appropriate parameter settings.

However, for pricing more exotic American options, we might need to use such an ad hoc approach. This might be very inefficient in practice, so we now briefly outline a method to address this problem.

We have already mentioned that while the upper bound should work very well in practice, the lower bound should still be superior. This observation is borne out in Section 5 when we price call options on the geometric mean of a number of stocks. For these problems, where it is actually possible to compute option prices exactly, we see

that the lower bound is closer than the upper bound to the true price.

This then suggests that when there is a large duality gap it is usually because the upper bound is not sufficiently close to V_0. The expression for the upper bound, as given in (28), suggests that if $\mathrm{E}_{t-1}\left[\widetilde{V}_t/B_t - \widetilde{V}_{t-1}/B_{t-1}\right]$ tends to be large, then the upper bound will not be very tight. Based on this observation, we propose the following modification to the approximate Q-value iteration algorithm.

After $\widetilde{Q}_t$ has been computed, we do not proceed directly to computing $\widetilde{Q}_{t-1}$. Instead, we simulate a number of points from the distribution of X_t and for each point, we compute $\mathrm{E}_t\left[\widetilde{V}_{t+1}/B_{t+1} - \widetilde{V}_t/B_t\right]$. If the average value of these quantities is below some threshold, ϵ_t, then believing that we have a good estimate of $\widetilde{V}_t$, we proceed to estimate $\widetilde{Q}_{t-1}$. Otherwise we re-estimate $\widetilde{Q}_t$, either by increasing the number of time t training points or by refining the approximation architecture, depending on the remedy that seems more appropriate. We then repeat the process until the average is less than ϵ_t. The resulting estimates of the Q-value functions should lead to tight lower and upper bounds.

A further advantage of this proposal is that it allows us to determine how much computational effort is required to obtain a good solution. In particular, we can now determine online how many training points are needed or how complex the approximation architecture needs to be in order to obtain good estimates of the option price. We do not implement this algorithm in Section 5 but merely point out that it may prove useful for certain classes of problems. This is a subject for future research.

5 Numerical Results

In this section we illustrate our methodology by pricing call options on the maximum of 5 and 10 stocks respectively, and the geometric mean of 5 stocks. We do not present

results for call options on the geometric mean of 10 stocks since this problem is in fact easier to solve than the 5 stock case. While somewhat counterintuitive, this is explained by noting that the volatility of the geometric mean *decreases* as the number of stocks increases.

We also mention at this point that problem specific information, which we describe below, was used to generate the results of this section. This is consistent with the approximate dynamic programming literature where it is well known that using such information can improve algorithmic performance. Of course, such improvement comes at the cost of having to specify more parameters and having to tailor the algorithm to each specific class of problems. We could avoid this by instead using the same generic algorithm for each problem class. Generally this results in only a slight, though sometimes possibly significant, loss in accuracy. In this paper, our motivation for using problem specific information is to highlight what can be achieved with the methodology.

In the problems that follow, we assume that the market has N traded securities with price processes given by

$$dS_t^i = S_i^t[(r - \delta_i)dt + \sigma_i dZ_t^i] \tag{32}$$

where Z_t^i is a standard Brownian motion and the instantaneous correlation of Z_t^i and Z_t^j is ρ_{ij}. Each security pays dividends at a continuous rate of δ_i. We assume that the option expires at time T and that there are n equally spaced exercise dates in the interval $[0, T]$. The first date occurs at $t = 0$ which we call the 1^{st} exercise period and the n^{th} exercise date occurs at $t = T$. We use k to denote the strike price of the option and let r be the annual continuously compounded interest rate. It is assumed that r is constant though this assumption is easily relaxed.

In order to generate the initial approximation to the option price, certain problem

specific information was also used, as discussed earlier. For example, we have already mentioned *feature extraction*, where functions of the current state vector are used as inputs for the neural network. In all the problems of this section, we found it advantageous to order the stock prices before using them as inputs to the neural network. In addition, we used the current intrinsic value of the option as a feature for options on the maximum, while for options on the geometric mean, the corresponding European option value was used. Though it is true in general that the exact European value of a high-dimensional option will not be available, this is not important since it is known (see Hutchinson et al (1994) that European options prices can be quickly and accurately approximated using learning networks . (This observation also suggests another way of implementing the approximate dynamic programming algorithm. Instead of using the approximation architecture to estimate the Q-value, we could use it to estimate the *early exercise premium*, conditional on not exercising the option at the current exercise period. This conditional early exercise premium plus the estimated European option price might give a more accurate estimate of the Q-value function.)

Another method by which problem specific information was used is *policy fixing* (see Broadie and Glasserman 1997). We know, for example, that the American option price is greater than or equal to the price of the corresponding European option. Such information can easily be incorporated to the approximate dynamic programming algorithm. We do this by simply redefining the estimated Q-value to be the maximum of the estimated value function one time period ahead, and a European option that is a lower bound on the Q-value. For options on the geometric mean, we again use the corresponding European option value to bound the Q-value from below. For options on the maximum, we use the European option on the stock that is most 'in the money'.

The initial approximation to the option price was obtained using 2000 and 2500

training points per period for the 5 and 10-dimensional problems, respectively. We also assigned 65%, 25% and 10% of the training points for the time $T-1$ neural network to the training, validation and test sets in turn. This network was trained a maximum of 5 times and stopped as soon as the mean-squared error on the test set was less than 1 cent. The remainder of the neural networks were trained only once, using the solution of the time $t+1$ network as the starting point for training the time t network. For these networks, 70% of the training points were assigned to the training set with the remainder assigned to the validation set.

The lower bound was obtained by simulating the approximate exercise strategy along eight million sample paths. The upper bound was computed using one thousand sample paths to estimate the expectation in (28), based on the estimated value of the lower bound $\underline{V}_0$. When estimating (28), we are computing the difference between the upper bound and the lower bound, $\overline{V}_0 - \underline{V}_0$, which is usually orders of magnitude smaller than the lower bound itself. As a result, starting from a close approximation to the option price, the upper bound may often be estimated very accurately using a relatively small number of sample paths. This can be seen in Tables 1–3. The upper bound depends on the constant $\underline{V}_0$ and the expression in (28) is valid for any choice of $\underline{V}_0$. The estimates of the upper bound reported in Tables 1–3 are based on the corresponding values of $\underline{V}_0$, reported in the same tables. For a fixed value of $\underline{V}_0$, the sample moments of the upper bound are defined in a standard manner.

The numerical results were generated using *Matlab* on a Dell workstation with a 2 GHz Pentium 4 processor. The running times for a typical 50 period, 5 asset problem were approximately: (i) 40 minutes to generate the initial approximation to the value function (ii) 100 minutes to generate the lower bound and (iii) 140 minutes to generate the upper bound. Since the focus of this paper is methodological, we did not concentrate

on computational efficiency. Significant improvements could clearly be achieved by using a faster programming language. Further gains in efficiency could also be obtained by using variance reduction techniques to generate the lower and upper bounds. Finally, we note that training and implementing neural networks is quite slow relative to linear architectures.

5.1 Call on the Maximum of 5 Assets

We assume that there are 5 assets, $r = 0.05$, $T = 3$ years and $k = 100$. We let $\delta_i = 0.1$, $\sigma_i = 0.2$ and $\rho_{ij} = 0$ for $i \neq j$. All stocks are assumed to have the same initial price S_0. The results are given in Table 1 for problems where there are 10, 25, 50 and 100 time periods.

It can be seen that the estimated lower and upper bounds are typically very close, thereby providing very accurate estimates of the true price. While it is true that the duality gap widens with the number of exercise periods it does so quite gradually so that even for the problems with 100 exercise periods, we can still obtain very good estimates. As we argued before, this widening of the duality gap is typically due to the gradual deterioration of the upper bound as the number of exercise periods increases. We will see further evidence to support this when we examine options on the geometric mean.

In Table 1 we also report the prices of the corresponding European options which allow us to compute the early exercise premia of the American options. We see that the duality gap is approximately 1% of the early exercise premium for options with 25 exercise periods or less. For problems with as many as 50 or 100 exercise periods, the duality gap is between 1% and 3%. Using the midpoint of the lower and upper bounds should therefore enable us to price the early exercise premium of these options to within 1% or 2%. Even more accurate price estimates could be obtained by noting that the lower bound is usually closer to the true price than the upper bound.

5.2 Call on the Geometric Mean of 5 Assets

We assume that there are 5 assets, $r = 0.03$, $T = 1$ years and $k = 100$. We let $\delta_i = 0.05$, $\sigma_i = 0.4$ and $\rho_{ij} = 0$ for $i \neq j$. All stocks are again assumed to have the same initial price S_0. The results are given in Table 1 for problems where there are 10, 25, 50 and 100 time periods.

For the American call option on the geometric mean of a collection of stocks the true price of the option can be computed using a standard binomial tree, since the stochastic process that describes the evolution of the geometric mean is itself a geometric Brownian motion. We therefore report the true price of the American options on the geometric mean together with our numerical results in Table 2.

The results are similar to those in Table 1, though the duality gap, measured as a percentage of the early exercise premium, now tends to be somewhat wider than before for the options that start out of the money. Again, this duality gap increases with the number of exercise periods though at a very gradual pace.

5.3 Call on the Maximum of 10 Assets

We make the same assumptions for the call option on the maximum of 10 assets as we did for the 5 asset case except now $T = 1$ year. The results are displayed in Table 3. Measured in absolute terms, the duality gap is again very small for these problems. However, measured as a percentage of the early exercise premium, the duality gap, though still quite small, can be as large as 20%. This could be due to a number of reasons.

First, the early exercise premium now represents a much smaller component of the overall option value than before, so that using the duality gap (measured as a percentage of the early exercise premium) to measure performance is likely to accentuate pricing

errors. This problem could possibly be overcome by approximating the early exercise premium rather than the continuation value of the option, as mentioned earlier.

The more likely reason for the wider duality gap is that an insufficient number of training points or neurons were used. Indeed, while we doubled the dimensionality of the state space, we made only a moderate increase in the number of training points and did not increase at all the number of neurons. As a result, we expect performance to suffer. Consider, for example, the 50 exercise period problem with $S_0 = \$90$. The estimated lower bound for this problem, \$15.181, is *greater* than the estimated lower bound, \$15.178, for the corresponding 100 period problem. Since we know that the true option price for the 100 period problem is greater than the true option price for the corresponding 50 period problem, it is clear that the initial option price approximation for the 100 period problem is inferior. Since the upper bound is quite sensitive to the initial approximation, we are not surprised to see that the duality gap for the 100 period problem is considerably wider than the duality gap for the 50 period problem.

When we re-solved the 100 period problem using $4,000$ training points per period and 25 neurons, the estimated lower bound increased to \$15.196, while the upper bound decreased to \$15.228. The resulting duality gap is only 3 cents which, for all practical purposes, is very tight.

6 Conclusions

In this paper we have developed a new method for pricing and exercising American options. Our main theoretical result is a representation of the American option price as a solution of a dual minimization problem. Based on this dual characterization of the price function, we use Monte Carlo simulation and an initial approximation to the value function to construct tight upper and lower bounds on the option price. These bounds

do not rely on a specific approximation algorithm and can be used in conjunction with other methods for pricing American options. We characterize the theoretical worst-case performance of the pricing bounds and show that they are very accurate on a set of sample problems where we price call options on the maximum and the geometric mean of a collection of stocks. These numerical results suggest that our pricing method can be successfully applied to problems of practical interest.

A Proofs

A.1 Proof of Theorem 2

Simplifying (8) and (9), and using Theorem 1, we obtain

$$\overline{V}_0 = \widetilde{V}_0 + \mathrm{E}_0\left[\max_{t\in\mathcal{T}}\left(\frac{h_t}{B_t} - \frac{\widetilde{V}_t}{B_t} + \sum_{j=1}^{t}\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]\right)\right] \tag{A1}$$

as an upper bound on the price of the American option. We then have

$$\begin{aligned}
\overline{V}_0 &= \widetilde{V}_0 + \mathrm{E}_0\left[\max_{t\in\mathcal{T}}\sum_{j=1}^{t}\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{V_j}{B_j} + \frac{V_j}{B_j} - \frac{V_{j-1}}{B_{j-1}} + \frac{V_{j-1}}{B_{j-1}} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]\right] \\
&\leq \widetilde{V}_0 + \mathrm{E}_0\left[\sum_{j=1}^{T}\left|\mathrm{E}_{j-1}\left[\frac{\widetilde{V}_j}{B_j} - \frac{V_j}{B_j} + \frac{V_{j-1}}{B_{j-1}} - \frac{\widetilde{V}_{j-1}}{B_{j-1}}\right]\right|\right] \\
&\leq V_0 + \left|\widetilde{V}_0 - V_0\right| + \mathrm{E}_0\sum_{j=1}^{T}\left(\mathrm{E}_{j-1}\left[\left|\frac{\widetilde{V}_j}{B_j} - \frac{V_j}{B_j}\right|\right] + \left|\frac{\widetilde{V}_{j-1}}{B_{j-1}} - \frac{V_{j-1}}{B_{t-1}}\right|\right),
\end{aligned}$$

where the second inequality is due to the supermartingale property of the discounted option price process, V_t, and the last step follows from the triangle inequality. The result of Theorem 2 then follows.

A.2 Proof of Theorem 3

At time t, the following six mutually exclusive events are possible: **(i)** $\widetilde{Q}_t \leq Q_t \leq h_t$, **(ii)** $Q_t \leq \widetilde{Q}_t \leq h_t$, **(iii)** $\widetilde{Q}_t \leq h_t \leq Q_t$, **(iv)** $Q_t \leq h_t \leq \widetilde{Q}_t$, **(v)** $h_t \leq Q_t \leq \widetilde{Q}_t$, **(vi)** $h_t \leq \widetilde{Q}_t \leq Q_t$. We define $\widetilde{\tau}_t = \min\{s \in [t,T] \cap \mathcal{T} : \widetilde{Q}_s \leq h_s\}$ and

$$\underline{V_t} = B_t\mathrm{E}_t\left[\frac{h_{\widetilde{\tau}_t}}{B_{\widetilde{\tau}_t}}\right].$$

For each of the six scenarios above, we establish a relation between the lower bound and the true option price.

(i),(ii) The algorithm for estimating the lower bound correctly prescribes immediate

exercise of the option so that $V_t - \underline{V_t} = 0$.

(iii) In this case the option is exercised incorrectly. $\underline{V_t} = h_t$ and $V_t = Q_t$ implying $V_t - \underline{V_t} \leq \left|\widetilde{Q}_t - Q_t\right|$.

(iv) In this case the option is not exercised though it is optimal to do so. Therefore

$$\underline{V_t} = \frac{B_t}{B_{t+1}} \mathrm{E}_t \left[\underline{V_{t+1}}\right]$$

while

$$V_t = h_t \leq Q_t + \left(\widetilde{Q}_t - Q_t\right) = \frac{B_t}{B_{t+1}} \mathrm{E}_t \left[V_{t+1}\right] + \left(\widetilde{Q}_t - Q_t\right).$$

This implies

$$V_t - \underline{V_t} \leq \left|\widetilde{Q}_t - Q_t\right| + \frac{B_t}{B_{t+1}} \mathrm{E}_t \left[V_{t+1} - \underline{V_{t+1}}\right].$$

(v),(vi) In this case the option is correctly left unexercised so that

$$V_t - \underline{V_t} = \frac{B_t}{B_{t+1}} \mathrm{E}_t \left[V_{t+1} - \underline{V_{t+1}}\right].$$

Therefore by considering the four possible scenarios, we find that

$$V_t - \underline{V_t} \leq \left|\widetilde{Q}_t - Q_t\right| + \frac{B_t}{B_{t+1}} \mathrm{E}_t \left[V_{t+1} - \underline{V_{t+1}}\right].$$

Iterating and using the fact that $\underline{V_T} = V_T$ implies the result.

B Low Discrepancy Sequences

A low discrepancy sequence is a deterministic sequence of points that is evenly dispersed in some fixed domain. Often, and without loss of generality, we take this domain to be the unit cube $[0,1]^d$. Because the points in a low discrepancy sequence are evenly dispersed, they are often used to numerically integrate some function $f(.)$ over $[0,1]^d$ so that

$$\int_{[0,1]^d} f(x)dx \approx \frac{\sum_{i=1}^{N} f(y_i)}{N} \tag{B1}$$

where $\{y_i : i = 1, \ldots, N\}$, is a set of N consecutive terms from the low discrepancy sequence. An important property that low discrepancy sequences possess is that as new terms are added, the sequence remains evenly dispersed. This property implies that, in contrast to other numerical integration schemes, the term N in (B1) need not be determined in advance and can therefore be chosen according to some termination criterion. Because these sequences are evenly dispersed, their use in numerical integration often results in a rate of convergence that is much faster than Monte Carlo simulation where the convergence rate is $O(\frac{1}{\sqrt{N}})$. For the technical definition of a low discrepancy sequence and a more detailed introduction to their properties and financial applications, see Boyle, Broadie and Glasserman (1997). See Birge (1994), Joy, Boyle and Tan (1996) and Paskov and Traub (1995) for some of these applications.

In this paper we use low discrepancy sequences for training point selection and training point evaluation. The low discrepancy sequences are of particular value for training point evaluation since a good estimate of

$$\mathrm{E}_t\left[\frac{B_t}{B_{t+1}} \max\left(h(X_{t+1}), \widetilde{Q}_{t+1}(X_{t+1})\right)\right]$$

can usually be computed much faster by using a low discrepancy sequence in place of Monte Carlo simulation.

Even though a low discrepancy point $y \in [0,1]^d$ is deterministic it can be useful to interpret it as being sampled from a uniform distribution in $[0,1]^d$. With this in mind, it is then straightforward to convert y into a point, x, that is representative of a random variable Z with cumulative distribution function $F(.)$. For example, suppose Z is a d-dimensional standard normal random variable with correlation matrix equal to the identity. We can then construct a point $z \in \Re^d$ that is representative of Z by setting

$$z = F^{-1}(y) \tag{B2}$$

where the operation F^{-1} is taken componentwise in (B2). If instead we wish to 'simulate' a multivariate normal random variable, X with covariance matrix Σ, then we simply premultiply z by the Cholesky decomposition of Σ.

In financial applications random variables are often assumed to be lognormally distributed. Since transforming a normal random variable into a lognormal random variable is easy, however, we can do likewise with z. We can therefore easily convert a d-dimensional low discrepancy sequence into a sequence of points that is representative of a d-dimensional multivariate log-normal distribution. The low discrepancy sequences we use in this paper are *Sobol* sequences.

References

Andersen, L. and M. Broadie, 2001, A Primal-Dial Simulation Algorithm for Pricing Multi-Dimensional American Options, Working Paper, Columbia University.

Barraquand, J. and D. Martineau, 1995, Numerical Valuation of High-Dimensional Multivariate American Securities, *Journal of Financial and Quantitative Analysis* 30, 383–405.

Bensoussan, A., 1984, On the Theory of Option Pricing, *Acta Applicandae Mathematicae* 2, 139–158.

Bertsekas, D. and J. Tsitsiklis, 1996, *Neuro-Dynamic Programming*, Athena Scientific, Belmont, MA.

Billingsley, P., 1995, *Probability and Measure*, Third Edition, John Wiley & Sons.

Birge, J. R., 1994, Quasi-Monte Carlo Approaches to Option Pricing, Technical Report 94–19. Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor.

Bossaerts, P., 1988, Simulation Estimators of Optimal Early Exercise, Working Paper, Carnegie Mellon University.

Boyle, P., M. Broadie, and P. Glasserman, 1997, Monte Carlo Methods for Security Pricing. *Journal of Economic Dynamics and Control* 21, 1267–1321.

Boyle, P., A.W. Kolkiewicz and K.S. Tan, 2001, Pricing American Style Options Using Low Discrepancy Mesh Methods, Working Paper, University of Waterloo, Waterloo.

Broadie, M., P. Glasserman, 1997a, Pricing American-style Securities Using Simulation. *Journal of Economic Dynamics and Control* 21, 1323–1352.

Broadie, M., P. Glasserman, 1997b, A Stochastic Mesh Method for Pricing High-Dimensional American Options, Working Paper, Columbia Business School, Columbia University, New York.

Broadie, M., P. Glasserman, and G. Jain, 1997c, Enhanced Monte Carlo Estimates for American Option Prices, *Journal of Derivatives*, 5, 25–44.

Carriere, J., 1996, Valuation of Early-Exercise Price of Options using Simulations and Nonparametric Regression, *Insurance: Mathematics and Economics* 19, 19–30.

Duffie, Darrell, 1996. *Dynamic Asset Pricing Theory*, Princeton University Press, Princeton, New Jersey.

Garcia, D., 1999, A Monte Carlo Method for Pricing American Options, Working Paper, University of California at Berkeley.

Garcia, R., J. Detemple and M. Rindisbacher, 2000, A Monte-Carlo Method for Optimal Portfolios, Working Paper.

Geske, R. and H.E. Johnson, 1984, The American Put Options Valued Analytically, *Journal of Finance* 39, 1511–1524.

Goonatilake, S. and P. Treleaven, Eds., 1995, *Intelligent Systems for Finance and Business*, John Wiley & Sons.

Harrison, J. M. and D. Kreps, 1979, Martingales and Arbitrage in Multiperiod Securities Markets, *Journal of Economic Theory* 20, 381–408.

Harrison, J. M. and S. R. Pliska 1981, Martingales and Stochastic Integrals in the Theory of Continuous Trading, *Stochastic Processes and their Applications* 11, 215–260.

Hornick, K., M. Stinchcombe and H. White, 1989, Multilayer feedforward Networks are Universal Approximators, *Neural Networks* 2, 359–366.

Hutchinson, A., A. Lo and T. Poggio, 1994, A Nonparametric Approach to the Pricing and Hedging of Derivative Securities Via Learning Networks, *Journal of Finance* 49, 851–889.

Ibanez, A. and F. Zapatero, 1998, Monte Carlo Valuation of American Options through Computation of the Optimal Exercise Frontier, Instituto Tecnologico Autonomo de Mexico .

Joy, C., P.P. Boyle, and K.S. Tan, 1996, Quasi-Monte Carlo Methods in Numerical Finance, *Management Science* 42, 926–938.

Karatzas, I, 1998, On the Pricing of American Options, *Applied Mathematics and Optimization* 17, 37–60.

Karatzas, I. and S. Shreve, 1998, *Methods of Mathematical Finance*, Springer-Verlag New York.

Longstaff, F. and E. Schwartz, 2001, Valuing American Options by Simulation: A Simple Least-Squares Approach, *Review of Financial Studies* 14, 113–148.

Niederreiter, H., 1992, Random Number Generation and Quasi-Monte Carlo Methods. CBMS-NSF 63, SIAM, Philadelphia, PA.

Paskov, S. and J. Traub, 1995, Faster Valuation of Financial Derivatives, *Journal of Portfolio Management* 22, Fall, 113–120.

Pliska, S., 1997, *Introduction to Mathematical Finance*, Blackwell Publishers Inc., Malden, MA.

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

Raymar, S. and M. Zwecher, 1997, A Monte Carlo Valuation of American Call Options on the Maximum of Several Stocks, *Journal of Derivatives* 5, 7–23.

Rogers, L.C.G., 2001, Monte Carlo Valuation of American Options, Working Paper, University of Bath.

Tezuka, S., 1995, *Uniform Random Numbers: Theory and Practice*, Kluwer Academic Publishers, Boston.

Tilley, J.A., 1993, Valuing American Options in a Path Simulation Model, *Transactions of the Society of Actuaries* 45, 83–104.

Tsitsiklis, J., and B. Van Roy, 1999, Optimal Stopping of Markov Processes: Hilbert Space Theory, Approximation Algorithms, and an Application to Pricing Financial Derivatives, *IEEE Transactions on Automatic Control* 44, 1840–1851.

Tsitsiklis, J., and B. Van Roy, 2000, Regression Methods for Pricing Complex American–Style Options, Working Paper, Laboratory for Information and Decision Systems, MIT, Cambridge.

Vapnik, V.N., 1999, *The Nature of Statistical Learning Theory (Statistics for Engineering and Information Science)*, 2nd Ed, Springer Verlag.

Table 1: **Call on the maximum of** 5 **assets**

Table 1 contains estimates of the price of an American call option on the maximum of 5 assets. We use the following set of parameter values: $r = 0.05$, $T = 3$, $k = 100$, $\delta_i = 0.1$, $\sigma_i = 0.2$ and $\rho_{ij} = 0$ for $i, j = 1, ..., 5$. All stocks are assumed to have the same initial price S_0. The columns "Lower Bound" and "Upper Bound" contain estimates of the lower and upper bounds, respectively. The standard errors of these estimates are given in brackets. Results are displayed for problems with 10, 25, 50 and 100 time periods. We report estimated options prices in $'s and their standard errors in cents.

S_0	Lower Bound	Upper Bound	European Price
	10 exercise periods		
90	16.640	16.658	14.586
	(0.57)	(0.49)	
100	26.151	26.177	23.052
	(0.68)	(0.46)	
110	36.758	36.826	32.685
	(0.77)	(1.48)	
	25 exercise periods		
90	16.866	16.889	14.586
	(0.56)	(0.67)	
100	26.465	26.511	23.052
	(0.67)	(1.31)	
110	37.172	37.207	32.685
	(0.75)	(0.99)	
	50 exercise periods		
90	16.949	16.989	14.586
	(0.55)	(1.57)	
100	26.555	26.656	23.052
	(0.67)	(2.85)	
110	37.280	37.350	32.685
	(0.75)	(1.59)	
	100 exercise periods		
90	16.962	17.030	14.586
	(0.56)	(2.18)	
100	26.611	26.666	23.052
	(0.66)	(1.84)	
110	37.332	37.442	32.685
	(0.75)	(2.47)	

Table 2: **Call on the geometric mean of** 5 **assets**

Table 2 contains estimates of the price of an American call option on the geometric mean of 5 assets. We use the following set of parameter values: $r = 0.03$, $T = 1$, $k = 100$, $\delta_i = 0.05$, $\sigma_i = 0.4$ and $\rho_{ij} = 0$ for $i, j = 1, ..., 5$. All stocks are assumed to have the same initial price S_0. The columns "Lower Bound" and "Upper Bound" contain estimates of the lower and upper bounds, respectively. The standard errors of these estimates are given in brackets. Results are displayed for problems with 10, 25, 50 and 100 time periods. We report estimated options prices in \$'s and their standard errors in cents.

S_0	Lower Bound	Upper Bound	True Price	European Price
		10 exercise periods		
90	1.359	1.366	1.359	1.172
	(.13)	(.08)		
100	4.282	4.292	4.282	3.445
	(.21)	(.08)		
110	10.177	10.188	10.179	7.521
	(.25)	(.12)		
		25 exercise periods		
90	1.380	1.390	1.381	1.172
	(0.13)	(0.14)		
100	4.340	4.353	4.342	3.445
	(0.21)	(0.14)		
110	10.363	10.374	10.365	7.521
	(0.21)	(0.19)		
		50 exercise periods		
90	1.387	1.396	1.388	1.172
	(0.13)	(0.19)		
100	4.360	4.376	4.361	3.445
	(0.20)	(0.21)		
110	10.409	10.427	10.411	7.521
	(0.20)	(0.27)		
		100 exercise periods		
90	1.389	1.400	1.391	1.172
	(0.12)	(0.52)		
100	4.368	4.388	4.371	3.445
	(0.20)	(0.30)		
110	10.422	10.464	10.431	7.521
	(0.19)	(0.38)		

Table 3: **Call on the maximum of** 10 **assets**

Table 3 contains estimates of the price of an American call option on the maximum of 10 assets. We use the following set of parameter values: $r = 0.05$, $T = 1$, $k = 100$, $\delta_i = 0.1$, $\sigma_i = 0.2$ and $\rho_{ij} = 0$ for $i, j = 1, ..., 10$. All stocks are assumed to have the same initial price S_0. The columns "Lower Bound" and "Upper Bound" contain estimates of the lower and upper bounds, respectively. The standard errors of these estimates are given in brackets. Results are displayed for problems with 10, 25, 50 and 100 time periods. We report estimated options prices in $'s and their standard errors in cents.

S_0	Lower Bound	Upper Bound	European Price
	10 exercise periods		
90	15.082	15.092	14.747
	(0.40)	(0.37)	
100	26.860	26.863	26.403
	(0.46)	0.60)	
110	39.076	39.076	38.522
	(0.51)	(0.48)	
	25 exercise periods		
90	15.158	15.172	14.747
	(0.39)	(0.51)	
100	26.960	26.970	26.403
	(0.45)	(0.74)	
110	39.182	39.193	38.522
	(0.50)	(0.64)	
	50 exercise periods		
90	15.181	15.216	14.747
	(0.38)	(0.79)	
100	26.978	27.092	26.403
	(0.45)	(3.46)	
110	39.225	39.261	38.522
	(0.49)	(0.96)	
	100 exercise periods		
90	15.178	15.283	14.747
	(0.39)	(1.36)	
100	26.996	27.070	26.403
	(0.45)	(1.17)	
110	39.223	39.306	38.522
	(0.50)	(2.01)	