Proceedings of the 2006 Winter Simulation Conference
L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds.

APPLYING MODEL REFERENCE ADAPTIVE SEARCH TO AMERICAN-STYLE OPTION PRICING

Huiju Zhang

R. H. Smith School of Business
University of Maryland
College Park, MD 20742, U.S.A.

Michael C. Fu

R. H. Smith School of Business
University of Maryland
College Park, MD 20742, U.S.A.

ABSTRACT

This paper considers the application of stochastic optimization methods to American-style option pricing. We apply a randomized optimization algorithm called Model Reference Adaptive Search (MRAS) to pricing American-style options by parameterizing the early exercise boundary. Numerical results are provided for pricing American-style call and put options written on underlying assets following geometric Brownian motion and Merton jump-diffusion processes. The results from the MRAS algorithm are also compared with the Cross-Entropy (CE) method.

1 INTRODUCTION

Pricing American options is a challenging problem in financial engineering, due to the early exercise features. Because of the complexity of the underlying dynamics, analytical models for option pricing entail many restrictive assumptions. Indeed, there is no analytical solution for the valuation of an American option on a single dividend-paying asset in the standard Black-Scholes framework. A number of simulation-based approaches have been developed to price American options since the 1990s. In contrast to traditional finite difference and lattice methods such as binomial trees, which often only handle limited number of uncertainty sources and become impractical in situations where there are multiple factors, Monte Carlo simulation methods are more widely applicable, because they can manage complicated derivatives with more state variables.

We classify these algorithms into three main categories. The first class casts the problem in a stochastic dynamic programming framework and employs a backwards induction algorithm. At each early exercise date, the payoff from immediate exercise is compared to the holding value, i.e., the conditional expectation from keeping the derivative alive. However, computing this conditional expectation can become computationally prohibitive as the dimension of the problem increases, and the next-stage value function is calculated over its entire asset space domain. Tilley (1993) first applied a bundling technique to approximate the holding values at early exercise points in an arbitrage-free setting. Improvements on Tilley's methods include Carriere (1996), who used a spline local regression technique to approximate the conditional expectations and find the optimal stopping in finite discrete time; and Longstaff and Schwartz (2001), who used least-square regression to provide a direct estimate of the conditional expectation function in high-dimensional setting. Laprise et al. (2006) applied secant and tangent interpolations to construct a piecewise linear approximation of the value function, estimating the American-style derivative by pricing a portfolio of European options at varying strike prices.

The second class of algorithms characterizes the optimal early exercise policies directly rather than using dynamic programming. Grant et al. (1996, 1997) identified the optimal critical price, i.e., the price below (above) which it is optimal to exercise for American put (call), using the backward recursive technique of dynamic programming, and incorporated this early exercise feature into Monte Carlo simulation. Fu and Hu (1995) cast the American option pricing problem as an optimization problem of maximizing the expected payoff with respect to the early exercise thresholds. They incorporated the gradient estimates into an iterative stochastic approximation algorithm and obtained sensitivities of the option value with respect to various parameters of the pricing model (see also Fu et al. 2000). Fu et al. (2001) introduced another way to solve this optimization problem using the simultaneous perturbation stochastic approximation (SPSA) approach proposed by Spall (1992).

The third class of algorithms is based on obtaining upper and lower bounds from simulated paths and backwards recursion. Broadie and Glasserman (1997) proposed a method based on simulated nonrecombining trees, where both bounds converge to the true price as computational effort increases. Broadie and Glasserman (2004) presented a stochastic mesh method for pricing high-dimensional American options with a finite number of exercise dates. The computational effort of this mesh algorithm is linear in its dependence on the number of exercise dates, in contrast to the exponential dependence for the random tree method.

In this paper, we apply a randomized algorithm called Model Reference Adaptive Search (MRAS) for pricing American options by solving an optimization problem in the spirit to the second class of algorithms discussed above. We compare MRAS results with those computed from perturbation analysis stochastic approximation (PASA) and SPSA, as described in Fu et al. (2001).

MRAS was proposed by Hu et al. (2005, 2006). The main idea of this approach is similar to that of the Cross-Entropy (CE) method (Rubinstein and Kroese 2004), which has been successfully applied to a wide range of combinatorial optimization and rare-event estimation problems. In contrast to instance-based methods such as simulated annealing (Aarts and Korst 1989), threshold acceptance (Dueck and Scheur 1990), genetic algorithm (GA) (Srinivas and Patnaik 1994) and tabu search (Glover 1990), where the new candidate solutions generated in the next iteration depend directly on solution or the 'population' of solutions from previous step, both MRAS and CE fall in the category of model-based search algorithms, which construct a random sequence of solutions via an intermediate parameterized probabilistic model that is updated from the previous solutions in such a way that the search will concentrate in the regions containing high quality solutions, and usually involve the following two iterative phases:

1. Generate candidate solutions (random data samples, vectors, trajectories, etc.) according to a specified random mechanism, e.g., a parameterized probability distribution.
2. Update the parameters of the random mechanism, typically parameters of pdfs, on the basis of the data collected in the previous step, to produce a "better" sample of candidate solutions in the next iteration.

The obtained parameters tend to coincide with the parameters that minimize variance in most cases, such that the outcome converges probabilistically to the optimal or near-optimal solution (see Fu et al. 2006).

This paper is organized as follows. The problem setting is described in Section 2. The MRAS algorithm applied to American-style option pricing is described in Section 3, and in Section 4, it is implemented in pricing American-style call and put options written on underlying assets following geometric Brownian motion and Merton jump diffusion model. The results from the MRAS algorithm are compared with the CE method as well. Finally we offer some conclusions based on the numerical results in Section 5.

2 PROBLEM SETTING

We consider the American option pricing problem as a maximization problem and apply optimization techniques to parameterize the early exercise boundary. The value of an American call option written on a dividend-paying single stock with finite early exercise dates can be written as

$$\max_{\{S_i^*\}} E(L(S_1^*,\ldots,S_n^*)],$$

where

$$L(S_1^*,\ldots,S_n^*) = \sum_{i=0}^{n-1} 1\{\bigcap_{j=0}^{i-1} S_j < S_j^*, S_i > S_i^*\}(S_i - K)e^{-rt_i} + 1\{S_1 < S_1^*,\ldots,S_{n-1} < S_{n-1}^*\}(S_T - K)^+ e^{-rT}, \quad (1)$$

$1\{\bullet\}$: indicator function,
K: strike price,
r: risk free rate,
T: maturity,
n: number of exercise opportunities, including at maturity,
S_i^*: early exercise threshold at exercise date t_i.
S_i: stock price at exercise date t_i,
L: net present value of the option payoff.

The first term on the right side is the payoff of early exercise, and the second term is the payoff without early exercise, i.e., the payoff at the time of maturity. Throughout, we assume options are not exercisable at time 0. Similarly, the American put option can be written with payoff

$$L(S_1^*,\ldots,S_n^*) = \sum_{i=0}^{n-1} 1\{\bigcap_{j=0}^{i-1} S_j > S_j^*, S_i < S_i^*\}(K - S_i)e^{-rt_i} + 1\{S_1 > S_1^*,\ldots S_{n-1} > S_{n-1}^*\}(K - S_T)^+ e^{-rT}. \quad (2)$$

We use $S^* = (S_1^*,\ldots,S_n^*)$ to denote the set of critical prices. Once we find estimates for the thresholds at all exercise points through optimization, we obtain the value of the option through a forward simulation starting from time 0. The procedure simultaneously optimizes all parameters iteratively, and no dynamic programming is involved. In addition, this flexible value function can handle pure-jump and jump-diffusion processes, which can sometimes be problematic for the most popular pricing methods, such as partial differential equation methods, binomial trees, and other lattice methods. In the following numerical examples, we consider the underlying asset following two stochastic processes – geometric Brownian motion and the jump diffusion model from Merton (1976).

3 ALGORITHM DESCRIPTION

MRAS is an adaptive algorithm equipped with a random mechanism and a reference model, working with a family of parameterized distributions on the solution space. The basic idea is to assign more weight to the solutions that have better performance at each step. Kullback-Leibler (KL) divergence is a natural "distance" measure between two probability distributions. At each iteration, samples are generated according to the distribution that has the minimum KL-divergence with respect to the reference model from the previous iteration, and the parameters of the next distribution are updated based on those samples in a way so that the distribution possesses the minimum KL-divergence with respect to the current reference model.

The main difference between MRAS and CE is that CE method uses a single optimal (importance sampling) distribution focused on the set of optimal solutions (i.e., zero variance) to guide the updating of parameters, while the MRAS uses a sequence of intermediate reference distributions to direct its parameter updating associated with the family of parameterized distributions during the search process. We will compare the results from MRAS with those from CE method in the following sections.

The MRAS method also resembles another model-based method: the estimation of distribution algorithms (EDAs). EDAs were introduced in the field of evolutionary computation by Mühlenbein and Paaβ (1996). Problem-specific interactions among the variables of individuals are taken into consideration, and the interrelations are expressed explicitly through the joint probability distribution associated with the individuals of variables selected at each generation. A new population is generated by sampling the probability distribution, which is estimated from a database containing selected individuals of the previous generation. Larranaga et al. (1999) and Paul and Iba (2002) give reviews of implementing EDA approaches using various underlying probabilistic models. However, the estimation of the joint probability distribution associated with the selected samples is a bottleneck of this method, because it is often computationally burdensome to calculate. In contrast, MRAS uses the sequence of reference models implicitly to guide the parameter updating procedure, and there is no need to calculate them explicitly; therefore MRAS overcomes the most difficult obstacle of EDAs.

Hu et al. (2006) demonstrate the global convergence of MRAS for a class of parameterized probability distributions called the Natural Exponential Family, which includes the multivariate normal distribution used in all of our numerical experiments, where we assume the estimated parameters are multivariate-normally distributed with p.d.f.

$$f(S^*,\mu_k,\Sigma_{k+1}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_k|}} \exp(-\frac{1}{2}(S^*-\mu_k)^T \Sigma_k^{-1}(S^*-\mu_k)), \quad (3)$$

where μ_k is the mean vector and $\sum_k$ the covariance matrix at iteration k, and the parameters are updated as

$$\mu_{k+1} = \frac{\sum\{[U(L(S^*))^k / f(S^*,\mu_{k+1},\Sigma_{k+1})] I_{\{L(S^*)\geq\bar{\gamma}_{k+1}\}} S^*\}}{\sum\{[U(L(S^*))^k / f(S^*,\mu_{k+1},\Sigma_{k+1})] I_{\{L(S^*)\geq\bar{\gamma}_{k+1}\}}\}}, \quad (4)$$

$$\Sigma_{k+1} = \frac{\sum\{[U(L(S^*))^k / f(S^*,\mu_{k+1},\Sigma_{k+1})] I_{\{L(S^*)\geq\bar{\gamma}_{k+1}\}} (S^*-\mu_{k+1})(S^*-\mu_{k+1})^T\}}{\sum\{[U(L(S^*))^k / f(S^*,\mu_{k+1},\Sigma_{k+1})] I_{\{L(S^*)\geq\bar{\gamma}_{k+1}\}}\}}, \quad (5)$$

where U(·) is a continuous and strictly positive increasing function used to ensure positive values.

For pricing American-style put options, the critical price increases as time approaches maturity, and the critical price at maturity is the strike price K. We generate the critical price increments at the exercise dates from a truncated multivariate normal distribution with given parameters using the acceptance-rejection method. For all increments except at the first exercisable date, we accept positive values and rule out negative ones. In addition, we only accept those samples in which the critical price at the last exercise date before maturity is less than the strike price K, the critical price at maturity. Similarly, for the call options, we accept samples that give negative increments at the exercisable dates except the first date, and satisfy the constraint that the threshold at the last exercise date before maturity is larger than the strike price K.

To avoid the (possible) premature convergence to a degenerate distribution and result in a sub-optimal solution, we applied a dynamic smoothing scheme as described in Kroese et al. (2004) instead of a fixed scheme. Define

$$\beta_k = \beta - \beta(1-\frac{1}{k})^q, \quad (6)$$

and the smoothed parameter updating procedure is

$$\hat{\mu}_k = \beta_k \mu_k + (1-\beta_k)\hat{\mu}_{k-1}, \quad (7)$$

$$\hat{\Sigma}_k = \beta_k \Sigma_k + (1-\beta_k)\hat{\Sigma}_{k-1}, \quad (8)$$

where k is the iteration number, β is a smoothing constant, and q is an integer. The implemented MRAS algorithm is as follows:

Algorithm MRAS

1. Initialize: quantile parameter ρ_0, initial sample size N_0, the multivariate normal distribution parameters μ_0 and Σ_0. Specify smoothing parameter β and q, sample size control parameter α, threshold increase parameter ε, and a continuous and strictly increasing positive function U (·). Set k=0.
2. Repeat until a specified stopping rule is satisfied:
   (a) Generate N_k i.i.d. samples $(S_1^*)^k,\ldots,(S_{N_k}^*)^k$ from the $N(\hat{\mu}_k,\hat{\Sigma}_k)$ distribution.
   (b) Find the sample (1 - ρ_k)-quantile $\gamma_{k+1}(\rho_k, N_k)$ of the samples $\{L(S_i^*)\}^k$, i = 1,…,N_k.
   (c) If k = 0 or $\gamma_{k+1}(\rho_k, N_k) \geq \bar{\gamma}_k + \varepsilon$, then

   Set $\bar{\gamma}_{k+1} \leftarrow \gamma_{k+1}(\rho_k, N_k), \rho_{k+1} \leftarrow \rho_k, N_{k+1} \leftarrow N_k$.

   Else, find the largest $\bar{\rho} \in (0, \rho_k)$ such that

   $\gamma_{k+1}(\bar{\rho}, N_k) \geq \bar{\gamma}_k + \varepsilon$.

   If such a $\bar{\rho}$ exists, then set

$\bar{\gamma}_{k+1} \leftarrow \gamma_{k+1}(\bar{\rho}, N_k), \rho_{k+1} \leftarrow \bar{\rho}, N_{k+1} \leftarrow N_k$.
Else set $\bar{\gamma}_{k+1} \leftarrow \bar{\gamma}_k, \rho_{k+1} \leftarrow \rho_k, N_{k+1} \leftarrow \alpha N_k$

(d) Update the distribution parameters μ_{k+1} and Σ_{k+1} according to Equations (4) and (5).
(e) Smooth the parameters via Equations (6), (7), and (8).
(f) Set k ← k+1.

$\hat{\mu}_k$ and $\hat{\Sigma}_k$ are the parameters after smoothing the μ_{k+1} and Σ_{k+1} originally computed from the samples. Step (c) calculate the non-decreasing threshold $\bar{\gamma}_k$, the sample size N_k, and threshold sample selection parameter ρ_k. If N_k is too small, the algorithm may fail to converge and result in poor quality solutions. Similarly, too large a value of ρ_k tends to use both the "good" and "bad" samples to update the probabilistic model, which slows down the convergence process. Therefore, N_k and ρ_k are dynamically adjusted, adaptively increasing and decreasing, respectively, where α is the rate of sample size increase. A small positive number ε is selected to ensure that $\{\bar{\gamma}_k\}$ is non-decreasing in the update procedure. At each iteration k, if the new quantile γ_{k+1} is large enough ($\gamma_{k+1}(\rho_k, N_k) \geq \bar{\gamma}_k + \varepsilon$), then we use this quantile as the new threshold and use the current sample size and ρ_k in the next iteration. Otherwise, it indicates that either ρ_k is too large or N_k is too small. First we try to find a smaller $\bar{\rho} < \rho_k$ such that the new sample $(1-\bar{\rho})$ quantile satisfies the above inequality. If such a $\bar{\rho}$ exists, then we decrease the ρ_k value and keep N_k unchanged in the next iteration. If no such $\bar{\rho}$ exists, then we increase the sample size by rate α, while ρ_k and $\bar{\gamma}_k$ remain unchanged. After we find ρ_{k+1}, N_{k+1}, and $\bar{\gamma}_{k+1}$, only those candidate solutions that have better performances than the new threshold will be used in the next iteration.

4 NUMERICAL RESULTS

In this section we present numerical results from the MRAS algorithm for both American-style call and put options, and compare them with the CE method. All the options in our numerical experiments have a finite number of early exercise opportunities, and are sometimes termed Bermudan derivatives. The stopping criteria at iteration k is 1) cov_max < 1.0, or 2) $\gamma_k = \gamma_{k-1} = \gamma_{k-2}$, or 3) $N_k > N_{max}$, where the cov_max is the maximal element in the covariance matrix of the multivariate normal distribution model and measures the convergence quality. For each test case, we use the following parameters: $\rho_0 = 0.5$, $N_0 = 100$, $\alpha = 2$, $\varepsilon = 10^{-3}$, $N_{max} = 1000$, U(x) := exp(0.1x), and smoothing parameters β = 0.8 and q = 5. The random number generator is taken from L'Ecuyer et al. (2002). The experiments were implemented with Matlab on a Pentium 4 1.5GHz computer.

The variables to be optimized are the critical prices $\{S_i^*\}$, which we obtain by optimizing over the critical price increments $\{X_i\} \sim N(\mu_k, \Sigma_k)$ at each exercise date, given a starting point S_0^*. For an option with n exercise dates, we have the following n-1 critical prices (the critical price at the last exercise date, the maturity, is known):

$S_1^* = S_0^* + X_1$;
$S_2^* = S_1^* + X_2$;
….
$S_{n-1}^* = S_{n-2}^* + X_{n-1}$.

Therefore, the initial conditions for simulation include the selection of S_0^*, the initial mean vector μ_0, and initial variance-covariance matrix Σ_0. We set the initial covariance between parameters to be 0, and the initial variance is the same for all X_i, i.e., Σ_0 is a diagonal matrix. The MRAS algorithm is not sensitive to the choice of initial mean and covariance matrix, provided that the initial sampling variance is chosen large enough.

4.1 Geometric Brownian Motion Model

We first apply MRAS algorithm to price the call option. We assume the underlying stock price follows geometric Brownian motion:

$$dS_t = (r-\delta)S_t dt + \sigma S_t dW, \quad (9)$$

where W_t is a standard Brownian motion process, δ is the dividend yield, and σ is the volatility. This leads to the discrete form used in the simulation:

$$S_{t+\Delta t} = S_t \exp((r-\delta-\sigma^2/2)\Delta t + \sigma\sqrt{\Delta t}Z, \ Z\sim N(0,1) \quad (10)$$

Table 1 illustrates the price estimates and their 95% confidence intervals based on 1,000,000 replications with obtained parameters of early exercise boundary from simulation, for a 3-year (T = 3) Bermudan call option with r = 0.05, σ = 0.2, δ = 0.04 and K=100, exercisable every 0.5yr (n = 6). We study the performance of MRAS for different initial condition settings: μ_0 = [-5, -5, -5, -5, -5] for S_0^* = 130, 140, 150, 160, 170, and 180, μ_0 = [-4, -4, -4, -4, -4] for S_0^* = 120, and μ_0 = [-2, -2, -2, -2, -2] for S_0^* = 110, that are bounded by the lower limit of the critical price at maturity. The diagonal (variance) of Σ_0 is 100 for all cases. The options considered here include in-the-money (S_0 = 110, 140), at-the-money (S_0 = 100), and out-of-the-money (S_0 = 60, 90). Results from MRAS are compared with

Table 1: Bermudan Call Option Prices on Asset under Geometric Brownian Motion

Method	S_0*	$S_0 = 60$		$S_0 = 90$		$S_0 = 100$		$S_0 = 110$		$S_0 = 140$	
		Price	C.I.	Price	C.I.	Price	C.I.	Price	C.I.	Price	C.I.
MRAS	110	0.87	0.01	8.64	0.03	13.56	0.04	19.52	0.04	42.32	0.06
	120	0.87	0.01	8.64	0.03	13.56	0.04	19.52	0.04	42.32	0.06
	130	0.86	0.01	8.65	0.03	13.57	0.03	19.52	0.04	42.33	0.06
	140	0.87	0.02	8.64	0.03	13.56	0.03	19.52	0.04	42.28	0.09
	150	0.88	0.02	8.64	0.04	13.57	0.04	19.52	0.04	42.33	0.06
	160	0.87	0.01	8.65	0.03	13.56	0.04	19.51	0.04	42.33	0.06
	170	0.87	0.01	8.64	0.03	13.56	0.04	19.52	0.04	42.32	0.06
	180	0.87	0.01	8.65	0.03	13.57	0.04	19.52	0.04	42.32	0.06
SPSA						13.69	0.04				
PASA						13.65	0.04				
DP						13.33	0.04				
Secant		0.88		8.63		13.56		19.53		42.29	
Tangent		0.87		8.63		13.55		19.53		42.29	
Eur		0.87		8.55		13.37		19.18		40.74	

those from SPSA, PASA, and sequential dynamic programming (DP) algorithms presented in Fu et al. (2001). We also compare them with the outcomes from secant and tangent methods described in Laprise et al. (2006); moreover, the corresponding European call option prices are given in the last row of the table.

Our experiments indicate that MRAS algorithm provides an accurate and efficient way to price American call options. It converges to the optimal value within 10 iterations, and the convergence is independent of the initial conditions. We achieve a cov_max less than 10 in all cases. The results are consistent with the findings of other approaches to similar accuracy, and the 95% confidence interval is about 5% of the price. The price for $S_0 = 60$ is close to the European call price, because it is deep out-of-the-money and the possibility of exercise is very small.

Table 2 displays the thresholds for $S_0 = K = 100$ at t = 0.5, 1.0, 1.5, 2.0, and 2.5. The deviation between the obtained optimal prices for various initial settings is relatively small. It is important to note that this is an at-the-money American call option example, where the fluctuation is expected to be large. Results from other scenarios suggest an even smaller critical price region dependence on S_0*.

Table 2 Thresholds of Bermudan Call Option

Method	S_0*	t = 0.5	t = 1.0	t = 1.5	t = 2.0	t = 2.5
MRAS	110	155.05	153.74	151.36	148.20	140.69
	120	153.35	151.69	148.50	146.70	133.22
	130	158.55	155.40	150.46	144.04	132.96
	140	157.06	151.27	148.67	143.94	130.80
	150	153.77	152.64	149.59	144.47	126.46
	160	158.04	154.36	147.74	145.58	129.43
	170	162.47	156.70	152.65	148.91	136.35
	180	157.11	150.56	147.60	144.28	132.88
Secant		158.43	154.06	148.68	141.70	
Tangent		158.42	154.05	148.67	141.70	

Table 3 shows the price of a 3-year (T = 3) American put option with r = 0.05, σ = 0.2, δ = 0, K = 100, and n = 6. μ_0 = [5, 5, 5, 5, 5] for S_0* = 30, 40, 50, 60, and 70, μ_0 = [4, 4, 4, 4, 4] for S_0* = 80, and μ_0 = [2, 2, 2, 2, 2] for S_0* =30, according to the upper limit of the critical price at maturity. The diagonal (variance) of Σ_0 is 100. Like the example of American call option, various scenarios of in-the-money (S_0 = 60, 90), at-the-money (S_0 = 100), and out-of-the-money (S_0 = 100, 140) are examined. The results from secant and tangent methods of Laprise et al. (2006) are listed for comparison. Analogous to Table 2, Table 4 presents the threshold estimates for at-the-money put option for each choice of S_0*.

The MRAS algorithm consistently finds the maximum values regardless of the initial choices, indicating that the true global optimum is reached in each case. The algorithm approaches the optimal value within 15 iterations for most cases. We also find the threshold bounds for the put options are tighter than for the calls, as shown in Table 4.

4.2 Merton Jump Diffusion Model

The jump-diffusion process is appealing, because it allows price discontinuities, but the presence of random jumps complicates the valuation of the American put option. The Merton (1976) jump diffusion model is written as follows:

$$S_{t+\Delta t} = S_t \exp((r-\delta-\sigma^2/2)\Delta t + \sigma\sqrt{\Delta t}Z_0 + \sum_{j=1}^{N(\Delta t)}(\gamma Z_j - \gamma^2/2)), \quad (11)$$

where Z_j ~ N(0,1) i.i.d., N(Δt) ~ Poisson (λΔt) is the number of jumps within time Δt , the jump sizes are i.i.d. log-normally distributed: LN($-\gamma^2/2$, γ^2), λ is the jump frequency, and γ is the jump volatility.

Table 3: Bermudan Put Option Prices on Asset under Geometric Brownian Motion (95% C.I. half-width ≈ 0.01-0.06)

Method	S_0*	S_0=60	S_0=90	S_0=100	S_0=110	S_0=140
MRAS	30	37.31	12.96	8.39	5.52	1.51
	40	37.48	12.95	8.39	5.52	1.48
	50	37.51	12.91	8.40	5.52	1.52
	60	37.52	12.95	8.39	5.52	1.51
	70	37.52	12.94	8.39	5.50	1.51
	80	37.52	12.95	8.43	5.52	1.51
	90	37.53	12.97	8.45	5.49	1.53
Secant		37.55	12.91	8.45	5.50	1.50
Tangent		37.55	12.91	8.45	5.50	1.50
European		27.97	10.24	7.00	4.71	1.37

Table 4: Thresholds of Bermudan Put Option

Method	S_0*	t=0.5	t=1.0	t=1.5	t=2.0	t=2.5
MRAS	30	81.87	84.29	86.63	88.54	90.64
	40	81.50	84.38	87.62	88.86	90.22
	50	83.63	85.27	85.87	86.89	89.02
	60	80.30	83.38	84.78	86.07	89.61
	70	83.86	85.31	87.83	88.37	90.33
	80	80.57	81.99	84.49	88.07	89.19
	90	81.60	82.29	85.06	86.65	89.00
Secant		83.06	84.02	85.32	87.20	
Tangent		83.06	84.3	85.32	87.20	

Table 5: Bermudan Put Option Prices on Asset under Merton Jump-Diffusion (95% C.I. half-width ≈ 0.02-0.04)

Method	S_0*	n = 2	n = 3	n = 4	n = 6
MRAS	30	8.56	8.65	8.62	8.73
	40	8.56	8.65	8.63	8.72
	50	8.57	8.64	8.63	8.73
	60	8.57	8.66	8.63	8.71
	70	8.57	8.63	8.63	8.73
	80	8.58	8.65	8.62	8.71
	90	8.58	8.65	8.63	8.73
SPSA		8.49	8.62	8.70	
DP		8.57	8.88	8.73	
Secant		8.61	8.69	8.73	8.77
Tagent		8.61	8.68	8.72	8.76

Table 5 shows the results of applying the MRAS algorithm to a six-month (T = 0.5yr) put option written on a single stock modeled by the jump-diffusion model without dividend ($\delta = 0$), and $r = 0.1$, $\sigma = 0.2828$, $\lambda = 2$, $\gamma = 0.2$, $S_0 = K = 100$. The European price (n = 1) for this example is 8.393. After obtaining the early exercise thresholds, we estimate the option price using 50,000 simulation replications. The MRAS algorithm is run for 20 different seeds, giving a 95% confidence half width within 0.02 - 0.04. The table also includes prices obtained using other algorithms, including SPSA, DP, and Secant/Tangent interpolation methods. The MRAS prices results are closest to the secant/tangent algorithm prices, and the values are between those from SPSA and DP when n is small (n = 2, 3) and they are more consistent as n increases. Moreover, the Secant method provides the upper bound for the results. The MRAS algorithm converges to the optimal value within 20 iterations regardless of the initial choice of S_0*.

4.3 Comparison between MRAS and CE Methods

Both MRAS and CE are model-based methods, which start with a parameterized probability distribution on the solution space and update the parameters at each iteration towards a 'better' solution. In MRAS, a sequence of reference distributions is adopted, and the minimum KL-divergence is achieved between the next step distribution and the current reference model, whereas in CE a single optimal importance sampling distribution is used, and the KL-divergence measures the distance between the optimal distribution and the family of parameterized distributions. The CE algorithm works as follows:

Algorithm CE

1. Initialize: Specify quantile parameter ρ and sample size N. Initialize parameters of the probabilistic model (multivariate normal distribution) μ_0 and Σ_0. Set k=0.
2. Repeat until a specified stopping rule is satisfied:
   (a) Generate N i.i.d. samples $X_1, \ldots, X_N$ from the $N(\hat{\mu}_k, \hat{\Sigma}_k)$ distribution.
   (b) Select the ρN best performing (elite) samples, and let I be the indices of the ρN best performing samples.
   (c) Update the parameters as:
   $$\mu_{k+1} = \frac{1}{\rho N}\sum_{i\in I} X_i \text{, and}$$
   $$\Sigma_{k+1} = \frac{1}{\rho N}\sum_{i\in I}(X_i - \mu_{k+1})(X_i - \mu_{k+1})^T$$
   (d) Smooth by using Equation (6), (7) and (8).
   (e) Set $k \leftarrow k+1$.

In the CE method, a fixed number (ρN) of best performing samples is selected at each iteration, where ρ and N remain constant. For MRAS, we study the sensitivity on the choice of initial ρ using a Bermudan put option written on a single asset following geometric Brownian motion, and model parameters K = 100, T = 3.0, N = 6, r = 0.05, $\delta = 0$, $\sigma = 0.2$, S_0* = [35, 40, 45, 50, 55], and initial covariance matrix with 100 on the diagonal, 0 otherwise.

Figure 1 displays the evolution of the early exercise thresholds for CE and MRAS as a function of the selection parameter ρ, where the value indicated is the initial value for a decreasing sequence in MRAS, e.g., for $\rho = 0.8$, ρ_k decreases from 0.8 down to 0.11 at the terminating point.

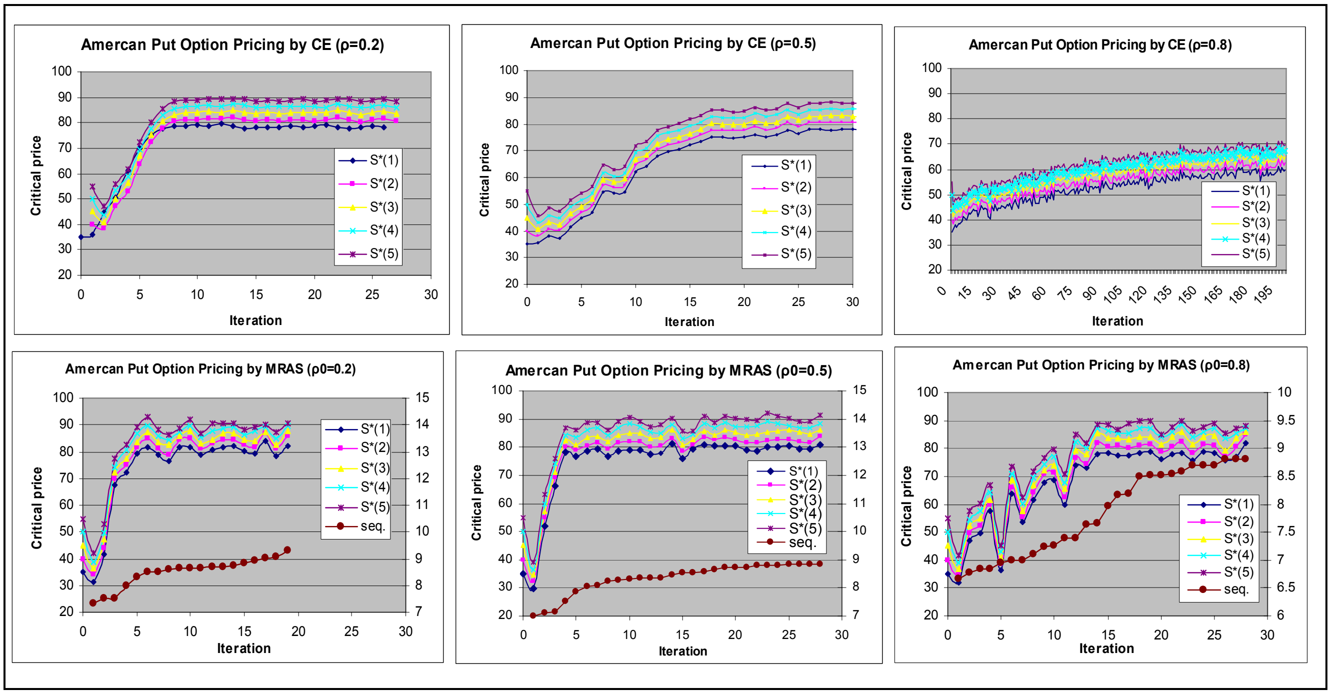

Figure 1: Evolution of Optimized Early Exercise Thresholds (Critical Prices) and MRAS Estimated Quantiles.

For MRAS, the sequence of (1- ρ_k) quantiles is also plotted (scale shown on the right side). For $\rho = 0.2$, we found CE converges more smoothly, whereas MRAS converges slightly faster. For $\rho = 0.5$, MRAS approaches the optimal value much quicker than CE, reaching near optimality by iteration 5, whereas it takes about 20 iterations for CE. For $\rho = 0.8$, CE hasn't converged even after 200 iterations, whereas MRAS reaches the optimum within 20 iterations, despite the large initial value of ρ. These results indicate that ρ assumes a critical role in the optimization process of the CE method. Unlike MRAS, where the convergence of the sequence of reference models to an optimal distribution model is guaranteed, the convergence of the sequence in CE relies on the quantile parameter ρ, which must be chosen sufficiently small. In contrast, the MRAS algorithm is relatively insensitive to the choice of initial quantile parameter and sample size.

5 CONCLUSIONS

We applied the MRAS algorithm to price American-style options written on underlying assets following geometric Brownian motion and jump-diffusion processes. MRAS optimizes the early exercise thresholds simultaneously by iterative updates via a reference model. In our simulation experiments, the global maximum is consistently found for varying initial condition settings. We demonstrate its accuracy and efficiency and also compare its performance with the CE method. We conclude that MRAS is a flexible and useful randomized optimization algorithm.

Future work includes an extensive numerical study on the choice of parameters and extending the application of MRAS to a wider range of test problems.

REFERENCES

Aarts, E., and J. Korst. 1989. *Simulated Annealing and Boltzmann Machines*. John Wiley & Sons.

Broadie, M., and P. Glasserman. 1997. Pricing American-style securities using simulation. *Journal of Economic Dynamics and Control* 21: 1323-1352.

Broadie, M., and P. Glasserman. 2004. A stochastic mesh method for pricing high-dimensional American options. *Journal of Computational Finance* 7: 35-72.

Carriere, J. F. 1996. Valuation of the early-exercise price for derivative securities using simulations and splines. *Insurance: Mathematics and Economics* 19: 19-30.

Dueck, G., and T. Scheur. 1990. Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing. *Journal of Computational Physics* 90: 161-175.

Fu, M. C., and J. Q. Hu. 1995. Sensitivity analysis for Monte Carlo simulation of option pricing. *Prob. in the Engineering and Information Sciences* 9: 417-446.

Fu, M. C., R. Wu, G. Gurkan, and A. Y. Demir. 2000. A note on perturbation analysis estimators for American-style options. *Probability in the Engineering and Informational Sciences* 14(3): 385-392.

Fu, M. C., S. B. Laprise, D. B. Madan, Y. Su, and R. Wu. 2001. Pricing American options: a comparison of

Monte Carlo simulation approaches. *Journal of Computational Finance* 4: 39-88.

Fu, M. C., Hu, J., and S. I. Marcus. 2006. Model-based randomized methods for global optimization. *Proc. of the 17th International Symposium on Mathematical Theory of Networks and Systems*, CD-ROM.

Glover, F. W. 1990. Tabu search: a tutorial. *Interfaces* 20: 74-94.

Grant, D., G. Vora, and D. Weeks. 1996. Simulation and the early-exercise option problem. *Journal of Financial Engineering* 5: 211-227.

Grant, D., G. Vora, and D. Weeks. 1997. Path-dependent options: extending the Monte Carlo simulation approach. *Management Science* 43: 1589-1602.

Hu, J., M. C. Fu, and S. I. Marcus. 2006. A model reference adaptive search algorithm for global optimization. *Operations Research* (to appear).

Hu, J., M. C. Fu, and S. I. Marcus. 2005. Simulation optimization using model reference adaptive search. *Proceedings of the 2005 Winter Simulation Conference*, 811-818. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Kroese, D., R. Y. Rubinstein, and S. Porotsky. 2004. The cross-entropy method for continuous multi-extremal optimization. Submitted to *Operations Research*.

Laprise, S. B., M. C. Fu, S. I. Marcus, A.E.B. Lim, and H. Zhang. 2006. Pricing American-style derivatives with European call options. *Management Science* 52: 95-110.

Larranaga, P., R. Etxeberria, J. A. Lozano, B. Sierra, I. Inza, and J. Pena. 1999. A review of the cooperation between evolutionary computation and probabilistic graphical models. *Proceedings of the Second Symposium on Artificial Intelligence. Adaptive Systems. CIMAF 99. Special Session on Distributions and Evolutionary Computation,* 314-324.

L'Ecuyer, P., R. Simard, E. Chen, and W. D. Kelton. 2002. An objected-oriented random-number package with many long streams and substreams. *Operations Research* 50: 1073—1075.

Longstaff, F., and E. Schwartz. 2001. Valuing American options by simulation: a simple least-squares approach. *The Review of Financial Studies* 14: 113-148.

Merton, R. C. 1976. Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics* 3: 125-144.

Mühlenbein, H., and G. Paaβ. 1996. From recombination of genes to the estimation of distributions: I. binary parameters. In Lecture Notes in Computer Science 1411: *Parallel Problem Solving from Nature – PPSN IV*: 178-187.

Paul, T., and H. Iba. 2002. Linear and combinatorial optimizations by estimation of distribution algorithms. *9th MPS Symposium on Evolutionary Computation*, *IPSJ*, Japan.

Rubinstein R. Y., and D. Kroese. 2004. *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*. Springer-Verlag, New York.

Spall, J. 1992. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transaction on Automatic Control* 37: 332-341.

Srinivas, M., and L. Patnaik. 1994. Genetic algorithms: a survey. *IEEE Computer* 27: 17-26.

Tilley, J. 1993. Valuating American options in a path simulation model. *Transactions of the Society of Actuaries* 45: 83-104.

AUTHOR BIOGRAPHIES

HUIJU ZHANG is a Ph.D. student of Decision and Information Technologies Department in the Robert H. Smith School of Business at University of Maryland, College Park. Her research interests include financial modeling and Monte Carlo simulation. She is a member of INFORMS. She is currently working at the World Bank, Washington D. C., focusing on risk management. Her e-mail address is <huizhang@rhsmith.umd.edu>.

MICHAEL C. FU is Ralph J. Tyser Professor of Management Science in the Robert H. Smith School of Business, with a joint appointment in the Institute for Systems Research and affiliate faculty appointment in the Department of Electrical and Computer Engineering, all at the University of Maryland. He received degrees in mathematics and EE/CS from MIT, and a Ph.D. in applied mathematics from Harvard University. His research interests include simulation optimization and applied probability, with applications in supply chain management and financial engineering. He was named a University of Maryland Distinguished Scholar-Teacher for 2004-2005. He is co-author (with J.Q. Hu) of the book, *Conditional Monte Carlo: Gradient Estimation and Optimization Applications*, which received the INFORMS College on Simulation Outstanding Publication Award in 1998. He currently serves as Stochastic Models and Simulation Department Editor for *Management Science*, serves as an Associate Editor for *Mathematics of Operations Research*, served as Simulation Area Editor of *Operations Research* 2000-2005, and also served on the editorial boards of *INFORMS Journal on Computing*, *IIE Transactions*, and *Production and Operations Management*. He was also Guest co-Editor of a 2003 special issue on simulation optimization for the *ACM Transactions on Modeling and Computer Simulation*. His e-mail address is <mfu@rhsmith.umd.edu>.