Probability in the Engineering and Informational Sciences, **14**, 2000, 317–326. Printed in the U.S.A.

PRICING EXOTIC OPTIONS

MONOTONICITY IN VOLATILITY AND EFFICIENT SIMULATION

SHELDON M. ROSS*

Department of Industrial Engineering and Operations Research
University of California, Berkeley, California 94720

J. GEORGE SHANTHIKUMAR

Department of Industrial Engineering and Operations Research
and Walter A. Haas School of Business
University of California, Berkeley, California 94720

We show that if the payoff of a European option is a convex function of the prices of the security at a fixed set of times, then the geometric Brownian motion risk neutral option price is increasing in the volatility of the security. We also give efficient simulation procedures for determining the no-arbitrage prices of European barrier, Asian, and lookback options.

1. WHEN THE OPTION PRICE INCREASES IN VOLATILITY

Let $S(t)$ be the price of a security at time t, and make the usual risk neutral assumption that $\{S(t), t \geq 0\}$ is a geometric Brownian motion with process volatility parameter σ and mean parameter μ that satisfies

$$\mu = r - \frac{\sigma^2}{2}.$$

*Research was supported by the National Science Foundation grant DMI-9901053 with the University of California.

 0269-9648/00 $12.50

https://doi.org/10.1017/S0269964800143037 Published online by Cambridge University Press

Here, r is the continuously compounded interest rate. Then,

$$E\left[\frac{S(t)}{S(0)}\right] = e^{rt}, \quad t \geq 0.$$

Furthermore, $X(t) := \ln\{S(t)/S(0)\}$ is normally distributed with mean

$$E[X(t)] = \left(r - \frac{\sigma^2}{2}\right)t, \quad t \geq 0,$$

and variance

$$\mathrm{Var}(X(t)) = \sigma^2 t, \quad t \geq 0.$$

Suppose $S(0) = s$ and consider an option that expires at some fixed time t and that pays P_s to its holder at time t. We assume P_s is defined as follows: For a positive integer n, a monotone time sequence $0 = t_0 < t_1 < \cdots < t_n = t$, and a function h: $R_+^n \to R$,

$$P_s = h(S(t_k), k = 1,2,\ldots,n).$$

Define g: $R_+^n \to R$, such that

$$g(x_1, x_2, \ldots, x_n) = h\left(s\prod_{l=1}^{k} x_l, k = 1,2,\ldots,n\right).$$

THEOREM 1: *Suppose g is componentwise convex. Then,* $E[e^{-rt}P_s]$ *is an increasing function of* σ.

PROOF: Define $Y_k = S(t_k)/S(t_{k-1})$, $k = 1,2,\ldots,n$. Then, Y_k, $k = 1,2,\ldots,n$, are independent log-normal random variables. Furthermore, $E[Y_k] = e^{rs_k}$ and $X_k := \ln\{Y_k\}$ is normally distributed with mean $(r - \sigma^2/2)s_k$ and variance $\sigma^2 s_k$, where $s_k = t_k - t_{k-1}$, $k = 1,2,\ldots,n$. Observing that

$$S(t_k) = s\prod_{l=1}^{k} Y_l, \; k = 1,2,\ldots,n,$$

one has

$$h(S(t_k), k = 1,2,\ldots,n) = g(Y_k, k = 1,2,\ldots,n).$$

Because Y_k, $k = 1,2,\ldots,n$, are independent, from Theorem 5.A.6 and (5.A.4) of [7], it follows that if Y_k is increasing in the convex ordering in σ {i.e., for any convex function ψ: $R_+ \to R$, $E[\psi(Y_k)]$ is increasing in σ}, then

$$E[e^{-rt}P_s] = E[e^{-rt}g(Y_k, k = 1,2,\ldots,n)]$$

is increasing in σ. Next, we shall show that Y_k is increasing in σ in the convex ordering. Let Z be a unit normal random variable. Then, for

$$\hat{Y}_k(\sigma) := e^{-(\sigma^2/2)s_k + \sigma\sqrt{s_k}Z},$$

we have

$$Y_k \stackrel{d}{=} E[Y_k]\hat{Y}_k.$$

For $\sigma_2 > \sigma_1 > 0$ and ϕ: $R_+ \to R$, given by

$$\phi(x) = (e^{-\sigma_2(\sigma_2-\sigma_1)s_k/2})x^{\sigma_2/\sigma_1},$$

we have

$$\hat{Y}_k(\sigma_2) = \phi(\hat{Y}_k(\sigma_1)).$$

Observing that ϕ is non-negative, increasing, and star-shaped [meaning that $\phi(x)/x$ is nondecreasing in x for $x > 0$], it follows from Theorem 2.A.10 and (2.A.16) of [7] that $\hat{Y}_k$ is increasing in σ in the convex ordering. Because $E[Y_k]$ is independent of σ, it immediately follows that Y_k too is increasing in σ in the convex ordering. ■

Remark 1: If h is convex or for any $x, y \in R_+^n$ such that $x_k \le y_k$, $k = 1, 2, \dots, n$,

$$h(\alpha x + (1-\alpha)y) \le \alpha h(x) + (1-\alpha)h(y), \quad 0 < \alpha < 1, \tag{1}$$

then g is componentwise convex. Hence, we have the following corollary.

COROLLARY 1: *Suppose h is convex or satisfies* (1). *Then,* $E[e^{-rt}P_s]$ *is an increasing function of* σ.

Some examples of options that satisfy the condition of Theorem 1 are as follows:

1. European vanilla option with strike price K

$$P_{E-V} = (S(t) - K)^+.$$

2. European option with a convex payoff function h,

$$P_{E-C} = h(S(t)).$$

3. Asian option with average end-of-day strike price

$$P_{A-A} = \left(S_d(n) - \frac{1}{n}\sum_{k=1}^{n} S_d(k)\right)^+,$$

where $S_d(k)$ is the price at the end of trading day k.

4. Asian option with minimum end-of-day strike price,

$$P_{A-\mathrm{MIN}} = S_d(n) - \min\{S_d(k), k = 1, 2, \dots, n\}.$$

5. With payoff maximum of end-of-day prices and a strike price of K,

$$P_{\mathrm{MAX}} = (\max\{S_d(k), k = 1, 2, \dots, n\} - K)^+.$$

6. Maximum deviation payoff,

$$P_{\mathrm{MAX}-D} = \max\{S_d(k), k = 1, 2, \dots, n\} - \min\{S_d(k), k = 1, 2, \dots, n\}.$$

https://doi.org/10.1017/S0269964800143037 Published online by Cambridge University Press

Remark: Part 2 of Corollary 1, the case of a European option with a convex payoff function, was proven in [1] and [4] by other methods.

2. EFFICIENT SIMULATION OF BARRIER OPTIONS

To define a European barrier call option with strike price K and exercise time t, a barrier value v is specified; depending on the type of barrier option, the option either becomes alive or is killed when this barrier is crossed. A *down and in* barrier option becomes alive only if the security's price goes below v before time t, whereas a *down and out* barrier option is killed if the security's price goes below v before time t. In both cases, v is a specified value that is less than the initial price s of the security. In addition, in most applications, the barrier is only considered to be breached if an end-of-day price is lower than v; that is, a price below v that occurs in the middle of a trading day is not considered to breach the barrier. Now, if one owns both a down and in option and a down and out option, both having the same values of K and t, then exactly one will be in play at time t; hence, owning both is equivalent to owning a vanilla option with exercise time t and exercise price K. As a result, if $D_i(s,t,K)$ and $D_o(s,t,K)$ represent respectively the risk neutral present values of owning the down and in and the down and out options, then

$$D_i(s,t,K) + D_o(s,t,K) = C(s,t,K)$$

where $C(s,t,K)$ is the Black–Scholes valuation of the standard call option. As a result, determining either one of the values $D_i(s,t,K)$ or $D_o(s,t,K)$ automatically yields the other.

Let $S_d(i)$ denote the price of the security at the end of day i and let

$$X(i) = \log\left(\frac{S_d(i)}{S_d(i-1)}\right).$$

Because successive daily price ratio changes are independent under geometric Brownian motion, it follows that $X(1),\ldots,X(n)$ are independent normal random variables, each having mean $(r-\sigma^2/2)/N$ and variance σ^2/N, where N is the number of trading days in a year. Now, suppose that we want to find the risk neutral valuation of a down and in barrier option with strike price K, barrier value v, whose initial value is $S(0)=s$, and whose exercise time is at the end of day n. To do so, generate n independent normal random variables with mean $(r-\sigma^2/2)/N$ and variance σ^2/N; set them equal to $X(1),\ldots,X(n)$ and use them to determine the sequence of end-of-day prices. Letting

$$I = \begin{cases} 1 & \text{if } S_d(i) < v \text{ for some } i = 1,\ldots,n \\ 0 & \text{if } S_d(i) \geq v \text{ for all } i = 1,\ldots,n, \end{cases}$$

then

$$\text{Present value payoff of the down and in option} = e^{-rn/N} I(S_d(n)-K)^+.$$

https://doi.org/10.1017/S0269964800143037 Published online by Cambridge University Press

Now, let $\mathbf{X}_i = (X(1), \ldots, X(i))$ and let f be the joint density function of $\mathbf{X}_n$. If g is another joint density, then the importance sampling identity yields that

$$E_f[I(S_d(n) - K)^+] = E_g\left[\frac{If(\mathbf{X}_n)(S_d(n) - K)^+}{g(\mathbf{X}_n)}\right].$$

Let T equal $n+1$ if the option never becomes alive, and let it equal i if the option first becomes alive at the end of day i. Then, for $T \le n$,

$$\begin{aligned}
E_g&\left[\frac{If(\mathbf{X}_n)(S_d(n) - K)^+}{g(\mathbf{X}_n)}\middle| T, \mathbf{X}_T\right] \\
&= \frac{f(\mathbf{X}_T)}{g(\mathbf{X}_T)} E_g\left[\frac{f(X_{T+1}, \ldots, X_n)}{g(X_{T+1}, \ldots, X_n)} (S_d(n) - K)^+ | T, \mathbf{X}_T\right] \\
&= \frac{f(\mathbf{X}_T)}{g(\mathbf{X}_T)} E_f[(S_d(n) - K)^+ | T, \mathbf{X}_T] \\
&= \frac{f(\mathbf{X}_T)}{g(\mathbf{X}_T)} e^{rn/N} C(S_d(T), (n - T)/N, K).
\end{aligned}$$

Defining $C(s, t, K)$ to equal 0 when $t < 0$, the preceding also holds when $T = n+1$. Hence, combining importance sampling and conditional expectation, the X_i can be generated according to a density that makes it more likely that the barrier is crossed; once the barrier is crossed, that simulation run ends with the following estimator of the risk neutral price:

$$\frac{f(\mathbf{X}_T)}{g(\mathbf{X}_T)} C(S_d(T), (n - T)/N, K). \tag{2}$$

If we generate the X_i as normal random variables with mean $(r - \sigma^2/2)/N - b$ and variance σ^2/N, then the estimator from that run is

$$C(S_d(T), (n - T)/N, K) \exp\left\{\frac{Tb^2 N}{2\sigma^2} + \frac{Nb}{\sigma^2}\sum_{i=1}^{T} X_i - \frac{Tb}{\sigma^2}\left(r - \frac{\sigma^2}{2}\right)\right\}. \tag{3}$$

Implementation requires an appropriate choice of b, which can be arrived at empirically. However, in an importance sampling application that did not utilize the conditional expectation improvement, it was noted in [3] that the choice

$$b = \frac{(\mu + \sigma^2/2)}{N} - \frac{2\log(S(0)/v) + \log(K/S(0))}{n}$$

works well.

Remark: Variance reduction by conditional expectation and by importance sampling were both suggested in [2]. That these procedures could be simultaneously utilized was, however, not noted. The estimator (3) has a smaller variance than the importance sampling estimator suggested in [3] and also requires less simulation time.

2.1. An Estimator with Smaller Variance

The estimator given by Eq. (2) can be improved upon by using an approach that we will first describe in more generality. Let $M_i, i \geq 0$, be a Markov chain with given initial state $M_0 = 0$; let L denote a fixed set of states and set

$$T = \min\{n+1, \min(i = 1, \ldots, n \colon M_i \in L)\}.$$

Suppose that we want to use simulation to estimate

$$\theta \equiv E[I(T \leq n)H(M_1, \ldots, M_T) + I(T = n+1)G(M_1, \ldots, M_n)],$$

where H and G are specified functions defined on partial sequences of states, and $I(B)$ is the indicator variable for the event B. Although θ can be estimated by stopping a simulation run either when it hits L or after n states have been simulated, consider the following way of doing the simulation. Generate the sequence $U_1, L_1, U_2, L_2, \ldots, U_n, L_n$ as follows: Let $U_0 = 0$; then, whenever the value of U_i has been determined, say it is equal to u, do the following:

1. Generate L_{i+1} by letting its distribution be the conditional distribution of the next state of the Markov chain given that the present state is u and given that this next state is in L.
2. Generate U_{i+1} by letting its distribution be the conditional distribution of the next state of the Markov chain given that the present state is u and given that this next state is not in L.
3. Generate a random variable J_{i+1} such that

$$P\{J_{i+1} = 1\} = P(u) = 1 - P\{J_{i+1} = 0\},$$

where

$$P(u) = P\{M_{i+1} \in L | M_i = u\}.$$

The preceding can be used to generate the successive states of the Markov chain, stopping either when it hits L or after n states have been generated, by letting

$$T = \min\{n+1, \min(i \colon J_i = 1)\}$$

and letting the states be $U_1, \ldots, U_{T-1}, L_T$ if $T \leq n$, or $U_1, \ldots, U_n$ if $T = n+1$. In other words, J is an indicator for the event that the next state is in L; if $J = 1$, we use L as the final state in that run, and if $J = 0$, we use U as the next state. The estimator from that run is the raw simulation estimator

$$\mathcal{E} \equiv I(T \leq n)H(U_1, \ldots, U_{T-1}, L_T) + I(T = n+1)G(U_1, \ldots, U_n).$$

https://doi.org/10.1017/S0269964800143037 Published online by Cambridge University Press

However, an improved estimator (one with the same mean and smaller variance) can be obtained by taking the conditional expectation of $\mathcal{E}$ given the values $\mathbf{V} \equiv (U_1, L_1, \ldots, U_n, L_n)$. Calling this new estimator $\mathcal{E}_1$, we have

$$\mathcal{E}_1 = E[\mathcal{E}|\mathbf{V}] = \sum_{i=1}^{n+1} E[\mathcal{E}|\mathbf{V}, T = i]\, P\{T = i|\mathbf{V}\}$$

$$= H(L_1)P(0) + \sum_{i=2}^{n} H(U_1, \ldots, U_{i-1}, L_i)P(U_{i-1}) \prod_{j=0}^{i-2} Q(U_j)$$

$$+ G(U_1, \ldots, U_n) \prod_{j=0}^{n-1} Q(U_j),$$

where

$$Q(u) = 1 - P(u).$$

Note that the preceding estimator only requires that $\mathbf{V}$, and not the J_i, be generated.

One difficulty with the estimator $\mathcal{E}_1$ is that it generates values L_i and computes $H(U_1, \ldots, U_{i-1}, L_i)$ even in situations where the probability that $M_i \in L$ is extremely small. From a practical point of view, the technique used in this estimator should only be employed when transitions into L begin having non-negligible probabilities; that is, one should employ some stopping time $N \le n$, which will depend on the states generated, and then start the simulation by generating the successive states $M_1, \ldots, M_{\min(T,N)}$. If $T \le N$, then the estimator $H(M_1, \ldots, M_T)$ is used. If $T > N$, then the new simulation procedure (of generating the U_i and L_i) begins with the states $M_1, \ldots, M_N$; that is, it sets $U_N = M_N$ and then generates L_{N+1} and U_{N+1}, and so on. The final estimator is

$$\sum_{i=N+1}^{n} H(U_1, \ldots, U_{i-1}, L_i)P(U_{i-1}) \prod_{j=N}^{i-2} Q(U_j) + G(U_1, \ldots, U_n) \prod_{j=N}^{n-1} Q(U_j),$$

where

$$U_i = M_i, \quad i = 1, \ldots, N, \quad \text{and} \quad \prod_{j=N}^{N-1} Q(U_j) \equiv 1.$$

Another variation that can improve the time (not variance) efficiency of the estimator is to switch back to the raw simulation estimator when $\prod_{j=0}^{i} Q(U_j)$ becomes very small. If we switch back at time N^*, then with

$$\beta = \prod_{j=N}^{N^*-1} Q(U_j),$$

https://doi.org/10.1017/S0269964800143037 Published online by Cambridge University Press

the estimator is

$$\sum_{i=N+1}^{N^*} H(U_1,\ldots,U_{i-1},L_i)P(U_{i-1})\prod_{j=N}^{i-2} Q(U_j) + \beta\mathcal{E},$$

where $\mathcal{E}$ is the raw simulation estimator.

Remark: Another estimator is

$$\mathcal{E}_2 \equiv H(L_1)P(0) + \sum_{i=2}^{\min(T,n)} H(U_1,\ldots,U_{i-1},L_i)P(U_{i-1}) + I(T > n-1)Q(U_{n-1})G(U_1,\ldots,U_n).$$

Rewriting $\mathcal{E}_2$ as

$$\mathcal{E}_2 = H(L_1)P(0) + \sum_{i=2}^{n} I(T \geq i)H(U_1,\ldots,U_{i-1},L_i)P(U_{i-1}) + I(T > n-1)Q(U_{n-1})G(U_1,\ldots,U_n)$$

shows that

$$E[\mathcal{E}_2|\mathbf{V}] = E[\mathcal{E}|\mathbf{V}] = \mathcal{E}_1.$$

The preceding shows that although $\mathcal{E}_2$ is also an unbiased estimator, its variance is larger than that of $\mathcal{E}_1$. Its advantage, however, is that it stops a simulation run at T, which may save quite a bit of computational time and thus offset the increased variance.

The estimator $\mathcal{E}_2$ can be obtained as follows. Initially, let $M_0 = U_0 = L_0 = 0$; when M_{i-1}, U_{i-1}, and L_{i-1} have been determined, generate M_i according to the transition probabilities of the Markov chain given that the preceding state was M_{i-1}. If $M_i \notin L$, let $U_i = M_i$ and generate L_i by letting its distribution be the conditional distribution of the next state of the Markov chain from M_{i-1} given that it is in L. If $M_i \in L$, let $L_i = M_i$ and end that simulation run. In addition, the time efficiency improvements noted for the estimator $\mathcal{E}_1$ can also be implemented for $\mathcal{E}_2$.

In applying the preceding to the barrier simulation model, we can let

$$M_i = \sum_{j=1}^{i} X(j),$$

$$L = \{m: se^m < v\},$$

$$H(M_1,\ldots,M_T) = \frac{f(\mathbf{X}_T)}{g(\mathbf{X}_T)} C(se^{M_T}, (n-T)/N, K),$$

$$G \equiv 0.$$

To implement the method, at each step we will need to generate normal random variables conditioned to be either smaller or larger than some value, which can be

https://doi.org/10.1017/S0269964800143037 Published online by Cambridge University Press

accomplished by generating a standard normal conditioned to be greater than some positive value. An efficient approach for generating a standard normal conditioned to exceed $b > 0$ is to use the acceptance rejection technique either with the distribution of an exponential random variable that is conditioned to exceed b (and so is equal to b plus the exponential) or with the distribution function

$$G(x) = 1 - e^{-(x^2 - b^2)}, \quad x > b.$$

Details are given in [6].

3. EFFICIENT SIMULATION OF ASIAN AND LOOKBACK OPTION PRICES

Consider an Asian option whose strike price is the average end-of-day price; that is, if the option expires at the end of n trading days, then the present value of its payoff is

$$\begin{aligned} Pe^{-rn/N} &\equiv e^{-rn/N}\left(S_d(n) - \sum_{i=1}^{n} \frac{S_d(i)}{n}\right)^+ \\ &= \frac{n-1}{n} e^{-rn/N}\left(S_d(n) - \frac{\sum_{i=1}^{n-1} S_d(i)}{n-1}\right)^+. \end{aligned}$$

To estimate $E[Pe^{-rn/N}]$, first condition on the data values $\mathbf{X}_{n-1} = (X(1), \ldots, X(n-1))$ to obtain

$$\begin{aligned} E[Pe^{-rn/N} | \mathbf{X}_{n-1}] &= \frac{n-1}{n} e^{-rn/N} E\left[\left(S_d(n) - \frac{\sum_{i=1}^{n-1} S_d(i)}{n-1}\right)^+ \middle| \mathbf{X}_{n-1}\right] \\ &= \frac{n-1}{n} e^{-rn/N} E\left[\left(S_d(n-1)e^{X(n)} - \frac{\sum_{i=1}^{n-1} S_d(i)}{n-1}\right)^+ \middle| \mathbf{X}_{n-1}\right] \\ &= \frac{n-1}{n} e^{-rn/N} C\left(S_d(n-1), \frac{1}{N}, \sum_{i=1}^{n-1} \frac{S_d(i)}{n-1}\right). \end{aligned} \tag{4}$$

Hence, we can estimate $E[e^{-rn/N}P]$ by generating $\mathbf{X}_{n-1}$ to obtain $S_d(1), \ldots, S_d(n-1)$, and then using the estimator given by Eq. (4), where $C(s, t, K)$ is the Black–Scholes risk neutral call option valuation.

This estimator can be improved by first noting that

$$C\left(S_d(n-1), \frac{1}{N}, \sum_{i=1}^{n-1} \frac{S_d(i)}{n-1}\right) \approx \left(S_d(n-1) - \sum_{i=1}^{n-1} \frac{S_d(i)}{n-1}\right)^+.$$

https://doi.org/10.1017/S0269964800143037 Published online by Cambridge University Press

Hence, as a simulation run consists of generating $X(1),\ldots,X(n-1)$, independent normal random variables with mean $(r-\sigma^2/2)/N$ and variance σ^2/N, and then setting

$$S_d(i) = S(0)e^{X(1)+\cdots+X(i)}, \quad i = 1,\ldots,n-1,$$

it follows that $C(S_d(n-1), 1/N, \sum_{i=1}^{n-1} S_d(i)/(n-1))$ will be large if the latter values of the sequence $X(1), X(2),\ldots,X(n-1)$ are among the largest, and small if the reverse is true. Consequently, one could try a control variable of the type $\sum_{i=1}^{n-1} w_i X(i)$, where the weights w_i are increasing in i. However, a better approach is to let the simulation itself determine the weights, by using all of the variables $X(1), X(2),\ldots, X(n-1)$ as control variables; that is, from each run consider the estimator

$$C\left(S_d(n-1), \frac{1}{N}, \sum_{i=1}^{n-1} \frac{S_d(i)}{n-1}\right) + \sum_{i=1}^{n-1} c_i \left(X(i) - \frac{(r-\sigma^2/2)}{N}\right).$$

The values of the constants $c_1,\ldots,c_n$ can then be determined from the simulation runs (see [5]). (An important technical point is that because the suggested control variables are independent random variables, there is not much additional computation needed to determine the values of the c_i.)

We also suggest first conditioning on $X(1), X(2),\ldots,X(n-1)$, and then using these variables as control variables in the lookback options 4, 5, and 6 of Corollary 1. When the payoff is a convex function of $S(t)$, then $S(t)$ itself can be used as a control variable.

Remark: Another possibility is to use $e^{X(i)}, i = 1,\ldots,n$, rather than the $X(i)$ themselves, as the control variates.

References

1. Bergman, Y.Z., Grundy, B.D., & Wiener, Z. (1996). General properties of option prices. *Journal of Finance* 51: 1573–1610.
2. Boyle, P. (1977). Options, a Monte Carlo approach. *Journal of Financial Economics* 4: 323–338.
3. Boyle, P., Broadie, M., & Glasserman, P. (1997). Monte Carlo methods for security pricing. *Journal of Economic Dynamics and Control* 21: 1267–1321.
4. Jagannathan, R.K. (1984). Call options and the risk of underlying securities. *Journal of Financial Economics* 13: 425–434.
5. Ross, S.M. (1997). *Simulation*, 2nd ed. New York: Academic Press.
6. Ross, S.M. (1999). Average run lengths for moving average control charts. *Probability in the Engineering and Informational Sciences* 14(2): 209–220.
7. Shaked, M. & Shanthikumar, J.G. (1994). *Stochastic orders and their applications.* New York: Academic Press.