M.S.Joshi, T.S.Leung, "Using Monte Carlo simulation and importance sampling to rapidly obtain jump-diffusion prices of continuous barrier options", *Journal of Computational Finance*, 10(4), 93-105, 2007

Using Monte Carlo simulation and importance sampling to rapidly obtain jump-diffusion prices of continuous barrier options

Mark S. Joshi
Centre for Actuarial Studies, Department of Economics, University of Melbourne, Victoria 3010, Australia; email: mark@markjoshi.com

Terence Leung
Department of Medical Physics and Bioengineering, Malet Place Engineering Building, University College London, Gower Street, London, WC1E 6BT, UK; email: tsl@medphys.ucl.ac.uk

The problem of pricing a continuous barrier option in a jump-diffusion model is studied. It is shown via an effective combination of importance sampling and analytic formulas that substantial speed ups can be achieved. These techniques are shown to be particularly effective for computing deltas.

1 INTRODUCTION

The Merton jump-diffusion model is almost as venerable as the Black–Scholes model; the first paper (Merton (1976)) was published in 1976. It captures a feature of equity markets which is sorely lacking in the Black–Scholes model: crashes. It allows the pricing of vanilla call and put options via an infinite sum. However, the pricing of exotic options is difficult, and other than in special cases, numerical methods must be resorted to. Here, we introduce a new approach to pricing barrier options with Monte Carlo which is predicated on the use of importance sampling and analytic formulas when there are no jumps left.

We focus here on the case of a down-and-out call option with strike above barrier for concreteness. The cases of up-and-outs and puts can be handled similarly. Knock-in options can be handled, as usual, via the observation that knock-in plus knock-out is the same as vanilla.

Linetsky (2006) has studied the pricing of equity derivatives where there is a possibility of bankruptcy represented by a jump with intensity equal to a negative power of the stock price and developed closed-form formula. This can be interpreted as a call option knocking out on jumps. Feng and Linetsky (2007) have developed an effective approach solving partial integro-differential equations (PIDEs) using finite element methods. Kou and Wang (2003) have studied the special case of barrier options for double exponential jumps, and derived a formula in terms of the Laplace transform. Lewis (2003) has developed a Fourier transform approach, and obtained formulas in the special cases that the jump does not cross the barrier and when there is a unique jump size. Penaud (2004) developed a lattice approach based on the pre-computation of densities designed to be rapid for large portfolios of derivatives but slow for a single instrument. Broadie and Yamamoto (2005) have developed pricing algorithms for discrete barrier options in the Merton jump-diffusion using the fast Gauss transform and shown them to be effective. Here, we study the case where jumps are lognormal and double exponential, however, the only properties we use of the jumps are that the cumulative distribution is easily computable, and that one can easily draw from its distribution. The method is therefore easily applicable to any case where these properties hold.

The naive approach to pricing any path-dependent derivative by Monte Carlo is simply to divide time into many small steps of size $\mathrm{d}t$, simulate the stock price across each of these small steps, and then evaluate the discounted payoff as a function of the stock-price path. We call this the short-step method. This is simple to implement and provides a useful benchmark but will be inefficient. It also neglects the possibility of barrier breaching during steps and so will be biased high. We can expect the size of the bias to be of the order of $\mathrm{d}t^{\frac{1}{2}}$, as it is of that magnitude in the Black–Scholes case (Broadie *et al* (1997)) and it is unlikely to be better in the jump-diffusion model. Metwally and Atiya (2002) examined the use of Monte Carlo simulation using a Brownian bridge technique. They successfully obtained substantial speed ups over the short-step method. The basis of their method is to first draw the times of jumps, and between jumps to compute the probability of barrier breaching by using the Brownian bridge, and then draw a indicator function to test whether knock-out has occurred. This method requires substantially fewer computations per path than short-stepping since computations need only be done at jump times. The standard deviation of the simulation is, however, roughly similar to the short-step method since the final prices are from distributions which differ only by the discretization error in the short-step method.

In the Metwally–Atiya method and the short-step method a large number of paths result in zero value due to knock-out occurring. The computational time spent on these paths can be removed via the use of importance sampling. With our technique, every time a barrier can be breached, we compute the probability of its occurrence, adjust the probability measure to ensure that the breach does not occur and then multiply the final path value by that probability to compensate. Every path therefore contributes. In the case of the Black–Scholes model, Glasserman and Staum (2001) have previously suggested a similar approach.

A second way in which we achieve greater efficiency is to observe that if the next jump time is after the expiry of the product, then it is possible to derive an analytic formula for the value of the remaining product. This has three bonuses: the first is that we can immediately compute the value of the product when no jumps occur, allowing us to importance sample to require that at least one jump occurs during the simulation; the second is that by analytically integrating out the final payoff, our prices are smoother and variance is reduced; and, third, we do not need to carry out the computations for the final step.

It is not sufficient just to be able to price rapidly, one also has to be able to compute the Greeks for risk management and hedging purposes. Monte Carlo pricing of derivatives with barrier features tends to be very poor at computing deltas via bumping; the reason is that the main contribution will arise from the small fraction of paths which change from knocking-out to not knocking-out (or *vice versa*) upon bumping. We thus have a small fraction of paths of high value and a lot of paths of low value; this results in high variance. Importance sampling provides a natural way round this problem: no paths knock-out and the delta effects arise instead via an implicit change in the density arising from the changes in the importance sampling. The method therefore bears some similarities to the likelihood ratio method of Broadie and Glasserman (1996). Importance sampling in the context of discrete barrier options has previously been discussed by Jäckel (2002).

We work purely with pseudo-random numbers in this paper and compare with other methods using pseudo-random numbers. However, all the methods could deliver considerable extra speed-ups if combined with the use of low-discrepancy numbers such as Sobol sequences.

In Section 2, we review the basics of the jump-diffusion process. In Section 3, we examine the necessary expressions for importance sampling in this context. We derive the price of a barrier option conditional on no jumps occurring in Section 4. We discuss the pricing algorithm more precisely in Section 5. Modifications to the method so that it works with double exponential jumps are presented in Section 6. Numerical results are presented in Section 7.

2 THE MERTON JUMP-DIFFUSION MODEL

We briefly recapitulate the Merton jump-diffusion model. For more details, see the original paper (Merton (1976); also contained in Merton (1998)) or the exposition of Joshi (2003).

In the pricing measure, the stock, S_t, is assumed to follow a process:

$$\frac{\mathrm{d}S_t}{S_t} = (r + \lambda(1 - m))\,\mathrm{d}t + \sigma\,\mathrm{d}W_t + (J - 1)\,\mathrm{d}N_t \tag{2.1}$$

where N_t is the number of jumps up to time t, according to a Poisson process with intensity λ, r is the risk-free rate, σ the volatility, and J the jump-size distribution with m equal to the expected value of J. In this model, there are many equivalent martingale measures, and the measure is chosen via its ability to calibrate to vanilla option prices since no arbitrage is far too weak to make the measure unique. The price of a derivative is given by the discounted expectation of its payoff.

We first take the jumps to be lognormally distributed:

$$J = m\,\mathrm{e}^{-\frac{1}{2}\sigma_{\mathrm{Jump}}^2 + \sigma_{\mathrm{Jump}}\phi_n} \tag{2.2}$$

with ϕ_n a collection of independent standard $N(0, 1)$ random variables. We use $+$ to denote the value just after a jump, and $-$ to denote the value just before. We thus have:

$$S_{t_n}^{+} = J S_{t_n}^{-} \tag{2.3}$$

$$\log S_{t_n}^{+} = \log S_{t_n}^{-} + \log m - \tfrac{1}{2}\sigma_{\text{Jump}}^{2} + \sigma_{\text{Jump}}\phi_n \tag{2.4}$$

Between jumps, the process therefore follows a geometric Brownian motion with drift:

$$\mu = r + \lambda(1 - m)$$

the drift adjustment arises from the need to counteract the bias arising from the directionality of jumps. The log process between jumps is:

$$\mathrm{d}\log S_t = \nu t + \sigma\,\mathrm{d}W_t \tag{2.5}$$

with $\nu = \mu - \frac{1}{2}\sigma^2$. If we wish to simulate by stepping between jump times, we can write the solution as:

$$\begin{aligned}\log S_T = \ln S_0 &+ \sum_{n=1}^{N-1}(\nu(t_n - t_{n-1}) + \sigma(W_{t_n} - W_{t_{n-1}}) + \log J_n)\\ &+ \nu(T - t_{N-1}) + \sigma(W_T - W_{t_{N-1}})\end{aligned} \tag{2.6}$$

where the jumps occur at times $t_1, t_2, \ldots, t_N$, and t_N is the first jump time greater than T. (If $N = 1$, then the sum will be empty.) We take $t_0 = 0$.

Note that:

$$\tau_n = t_n - t_{n-1} \tag{2.7}$$

is an exponential random variable with density function:

$$\lambda\,\mathrm{e}^{-\lambda t}$$

3 IMPORTANCE SAMPLING

The fundamental idea of importance sampling is to modify the probability density being drawn from in such a way as to reduce variance. If in a probability measure, P, we have density, ϕ, we can introduce a measure, Q, with new density, ψ, and rewrite an expectation via:

$$\mathbb{E}_P(f(x)) = \int f(x)\phi(x)\,\mathrm{d}x = \int f(x)\frac{\phi(x)}{\psi(x)}\psi(x)\,\mathrm{d}x = \mathbb{E}_Q\left(f(x)\frac{\phi}{\psi}(x)\right) \tag{3.1}$$

If ψ is chosen so that $(\phi(x)/\psi(x))f(x)$ is constant or close to constant, then variance will be small.

A second related approach, which we adopt here, is to modify the density so that the area where f is zero is not sampled. For example, if an option knocks-out if a normal draw, Z, is below the value x, then denoting the normal density function by N' (which is the derivative of the cumulative normal N), we have:

$$\int f(z)N'(z)\,\mathrm{d}z = \int_{z\geq x} f(z)N'(z)\,\mathrm{d}z = \theta \int f(x)\psi(z)\,\mathrm{d}z$$

where:

$$\psi(z) = \begin{cases} \theta^{-1}N'(z) & \text{for } z \geq x \\ 0 & \text{otherwise} \end{cases}$$

with $\theta = \mathbb{P}(Z > x)$.

The algorithm to construct a normal draw above x will therefore be to draw a uniform u and set:

$$z = N^{-1}(1 - \theta u)$$

and the final value of the payoff will also be multiplied by θ.

We will also wish to importance sample for exponential distributions to ensure that at least one jump occurs. Here we simply have to multiply by $p = \mathbb{P}(t_1 < T) = 1 - \mathrm{e}^{-\lambda T}$ before applying the inverse cumulative exponential to a uniform random variable. Thus, our algorithm is:

- draw a uniform u_0;
- let $u_1 = pu_0$;
- let $t_1 = -\log(1 - u_1)/\lambda$;

and the final price will also have to be multiplied by p.

Our third method of importance sampling is very simple. Between two jump times, the probability of the Brownian motion breaching the barrier is computable from knowledge of the volatility and the value of spot at the two end points of the step. In this case, the random variable to be drawn is the indicator function of not breaching. After importance sampling, it is simply the constant 1, and we multiply the final payoff by this probability. The probability that a geometric Brownian motion will breach the level H given its values at both ends which are above H is the same problem as for its log breaching $\log H$. This is a standard problem known as the minimum of the Brownian bridge. The use of the Brownian bridge process was an important part of the approach of Metwally and Atiya (2002). From there, we have that the probability of breaching is:

$$\mathbb{P}_n^B = 1 - \exp\left(-\frac{2(\log H - \log S_{t_{n-1}}^{+})(\log H - \log S_{t_n}^{-})}{\tau_n \sigma^2}\right) \tag{3.2}$$

Each path will require a number of importance samplings from different distributions, and the final value will be multiplied by a likelihood ratio for each of these. Note that all of the ratios will be less than one so it is not possible for some paths to result in extreme values; this means that we avoid the problem of highly skewed distributions that can arise with importance sampling (Glasserman (2003)).

4 THE PRICE CONDITIONAL ON NO JUMPS OCCURRING

We can decompose the price according to the whether the event of a jump before maturity occurs; let $C(S, t)$ denote the price of the barrier option:

$$C(S_0, 0) = \mathbb{P}(t_1 > T)C_0^{\text{NoJump}}(S_0, 0) + \mathbb{P}(t_1 \leq T)C_0^{\text{Jump}}(S_0, 0) \tag{4.1}$$

where C_0^{NoJump} denotes the value conditional on no jumps occurring before maturity and similarly for C_0^{Jump}. A similar equation will also hold at time t.

We can analytically evaluate $C_0^{\text{NoJump}(S_0,0)}$. The stock is following a geometric Brownian motion and we have to price a barrier option. The price will not quite be the price of a barrier option in the Black–Scholes world because the drift of the stock is not the riskless rate in the pricing measure. However, the price will be the same as the price of a barrier option on a dividend-paying stock with (possibly negative) dividend, d, equal to $-\lambda(1-m)$, since that will have the same drift in the pricing measure. For a derivation of the formula in that case, see Musiela and Rutowski (1997).

Letting $\tilde{\nu} = \nu + \sigma^2$, we therefore conclude:

$$\begin{aligned} C^{\text{NoJump}}&(S_0, 0) \\ &= S_0\,\mathrm{e}^{(1-m)\lambda T}\left(N\left(\frac{\tilde{\nu}T + \log(S_0/K)}{\sigma\sqrt{T}}\right)\right. \\ &\quad \left. - \left(\frac{H}{S_0}\right)^{2\tilde{\nu}\sigma^{-2}} N\left(\frac{\log(H^2/KS_0) + \tilde{\nu}T}{\sigma\sqrt{T}}\right)\right) \\ &\quad - \mathrm{e}^{-rT}K\left(N\left(\frac{\nu T + \log(S_0/K)}{\sigma\sqrt{T}}\right)\right. \\ &\quad \left. - (H/S_0)^{2\nu\sigma^{-2}} N\left(\frac{\log(H^2/KS_0) + \nu T}{\sigma\sqrt{T}}\right)\right) \end{aligned} \tag{4.2}$$

We can therefore evaluate the first term of (4.1) analytically. Our Monte Carlo will only address the second term.

5 THE ALGORITHM

As noted above, we decompose the price into the no-jump price and the price given that one jump has occurred. We now state the algorithm for evaluating the second part. We will perform M paths and average. For each path, we proceed as follows.

(1) Set the likelihood ratio for path to 1.
(2) Draw the first jump time, t_1, using the importance sampling to ensure that it is before T.
(3) Draw the succeeding jump times, t_j, so that the increments are exponentially distributed. Repeat until the final maturity is crossed. Call the number of jumps N.

Marked Proof Ref: Joshi10(4)/35724 May 22, 2007

(4) Starting with t_0, for each jump before maturity:
   (a) draw the increment of the stock across the time step, using importance sampling to ensure that the stock is above the barrier at the end of the step; multiply the likelihood ratio for the path by the factor from this importance sampling;
   (b) compute the probability that the barrier was breached during the step using the Brownian bridge minimum formula and multiply the likelihood ratio by this;
   (c) draw the jump size with importance sampling to ensure that the stock remains above the barrier and update the likelihood ratio.

(5) Calculate the expected value of option at $(S^+_{t_{n-1}}, t_{n-1})$ using the formula for the option price conditional on no jumps, multiply by $e^{-rt_{n-1}}$ to discount, and multiply this by the accumulated likelihood ratio to get the value for the path.

6 DOUBLE EXPONENTIAL JUMPS

Whilst we have focused this far on lognormal jumps, it is also possible to use double exponential jumps for the change in value of the log stock price. These are popular for jump-diffusion models because of the ability to develop analytic formulas (Kou (2000); Kou and Wang (2003)). We discuss the modifications necessary to encompass that case. The only change is to the distribution of the jumps, and therefore the changes to the algorithm appear via its cumulative distribution function and its inverse.

The jump size is:

$$J = m\, e^{x-\kappa} \tag{6.1}$$

with $x - \kappa$ double exponentially distributed.

The double exponential density is defined by:

$$f_X(x) = \frac{1}{2\eta} e^{-|x-\kappa|/\eta}, \quad 0 < \eta < 1 \tag{6.2}$$

It can also be represented by:

$$X - \kappa = \begin{cases} \xi & \text{with probability } 0.5 \\ -\xi & \text{with probability } 0.5 \end{cases}$$

where ξ is an exponential random variable with mean η and variance η^2.

We repeat how to simulate from an exponential with parameter η, as we will need this for simulating from the double exponential:

$$F_E(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 - e^{-x/\eta} & \text{for } x \geq 0 \end{cases} \tag{6.3}$$

with inverse:

$$F_E^{-1}(v) = -\eta \log(1 - v) \tag{6.4}$$

To simulate from a double exponential, we need to get a downwards jump with probability 0.5 and an upwards one otherwise. We therefore divide into two cases, we

- take a uniform $u \in (0, 1)$;
- if $u < 0.5$, we set $v = 1 - 2u$ in the above, and take a jump of size $\kappa - \xi$;
- if $u > 0.5$ we use set $v = 2u - 1$ in the above, and take a jump of size $\kappa + \xi$;
- if $u = 0.5$, we take a jump of size κ.

Note that with this algorithm the jump will be monotone increasing in u.

To carry out the importance sampling for jumps, it is then simply a question of computing the u that makes the jump land on the barrier, call this u_0, and rescaling the original u to ensure that it is in the range $(u_0, 1)$. The rest of our algorithm is unchanged.

7 NUMERICAL RESULTS

In this section, we present numerical results. Since we have an analytic price when no jumps occur, we can expect greater accuracy for a given number of simulations when the jump intensity is small. Indeed, as the jump intensity goes to zero, the price will converge to the no-jump price which will become the Black–Scholes price, whereas when λ is large, the gain from pricing the no-jump part analytically will become insignificant as very few paths will have no jumps.

We compare our method with the method of Metwally and Atiya (2002). We do not carry out comparisons with the short-step method, since it has already been demonstrated by Metwally and Atiya (2002) that it is much slower. We do not just wish to improve the standard deviation of the simulation, but also to achieve better computation times. We therefore present results for both. Note that our method requires additional computations in terms of cumulative normals in order to compute the no-jump expectation and likelihood ratios, and so the timing per path can be expected to be slower. On the other hand, the use of importance sampling reduces variance, as does the use of the no-jump formula. As well as comparing prices, we also examine the convergence of the Delta obtained by symmetrical bumping.

The computer used to generate the results has a 2.8 GHz Xenon processor and the two methods have been implemented in C++ (Visual C++ 6.0). The values of the parameters in the model are spot = 100, strike = 110, barrier = 95, volatility = 25%, $r = 0.05$, $T = 1$, and the parameters for the jump distribution are $\sigma_{\text{jump}} = 0.1$ and $m = 1.005$. These values have been chosen to be the same as those given by Metwally and Atiya (2002) for comparison. It should be noted that while rebate has been set to US$1 dollar in Metwally and Atiya (2002), rebate is zero here. We also present results for the double exponential model with $\kappa = 0.005$ and $\eta = 0.071$. These parameters have been chosen to give a similar calibration of the model to that of the lognormal jump-diffusion model.

We study several values of λ, and see that there is considerable difference according to its value in comparative performance of the models. This is not

TABLE 1 The standard error, time taken and time taken to achieve a standard error of 0.01 for the importance sampling method for lognormal jumps.

Intensity	Price	Importance standard error	Importance time	Importance time 0.01 standard error
0.1	4.0396	0.0006	4.0930	0.0143
0.2	4.0651	0.0011	4.1870	0.0548
0.5	4.1384	0.0026	4.3600	0.2936
1.0	4.2475	0.0044	4.7500	0.9405
2.0	4.4623	0.0069	5.7180	2.7288
4.0	4.8480	0.0096	7.9850	7.4153
8.0	5.4585	0.0127	13.0310	21.0178

TABLE 2 The standard error, time taken and, the estimated time to achieve a standard error of 0.01 for the Metwally–Atiya method. The final column is the ratio of the time to get 0.01 for the importance sampling to the Metwally–Atiya method for lognormal jumps.

Jump intensity	Standard error	Time	Time 0.01 standard error	Ratio of times
0.1	0.0127	0.7970	1.2877	0.0111
0.2	0.0128	0.8440	1.3904	0.0394
0.5	0.0131	0.9220	1.5932	0.1843
1.0	0.0137	1.0620	1.9939	0.4717
2.0	0.0147	1.3750	2.9740	0.9176
4.0	0.0166	1.9060	5.2823	1.4038
8.0	0.0202	3.0780	12.5327	1.6770

particularly surprising in that for small λT almost all of the value is contained in the analytic no-jump price for the method presented here, whilst with large λT only a small fraction of paths will have no jumps, and this will have little impact.

In the first numerical experiment, we aim to find the time, M, required for each method to achieve a pre-defined standard error for different values of λ. The target standard error is 0.01. The range of λ analysed here are 0.1, 0.2, 0.5, 1, 2, 4 and 8. We run one million paths, compute the standard error and then scale to get the time taken to achieve a standard error of 0.01 using the fact that standard error is equal to $\sigma/\sqrt{n}$ where σ is the sample standard deviation.

We present numerical results for the two methods. (See Tables 1–6.) The importance sampling method results in a price with a lower standard error for the same number of paths. The crucial test of a numerical method is, however, speed. If a lot of computational effort is required to reduce the variance, then the gains may well be outweighed by the costs. We see that the importance sampling method is much faster for small values of λ and of comparable speed for large values. Much of the extra time is devoted to computing cumulative normal functions, and it is perhaps possible that the method could be made more rapid by using an

Marked Proof Ref: Joshi10(4)/35724 May 22, 2007

TABLE 3 The standard error, time taken and time taken to achieve a standard error of 0.01 for the importance sampling method for double-exponential jumps.

Intensity	Price	Importance standard error	Importance time	Importance time 0.01 standard error
0.1	4.0398	0.001	3.906	0.014
0.2	4.0618	0.001	3.969	0.053
0.5	4.1354	0.003	4.156	0.285
1.0	4.2439	0.005	4.547	0.926
2.0	4.4630	0.007	5.484	2.694
4.0	4.8308	0.010	7.688	7.356
8.0	5.4582	0.013	12.531	21.055

TABLE 4 The standard error, time taken and the estimated time to achieve a standard error of 0.01 for the Metwally–Atiya method for double-exponential jumps. The final column is the ratio of the time to get 0.01 for the importance sampling to the Metwally–Atiya method.

Jump intensity	Standard error	Time	Time 0.01 standard error	Times
0.1	0.0127	0.7970	1.2900	0.0108
0.2	0.0129	0.8290	1.3699	0.0384
0.5	0.0132	0.9060	1.5729	0.1812
1.0	0.0138	1.0470	1.9883	0.4655
2.0	0.0148	1.3120	2.8898	0.9322
4.0	0.0168	1.8910	5.3339	1.3791
8.0	0.0204	3.0780	12.7588	1.6502

TABLE 5 The standard error, time taken and the estimated time to achieve an error 0.01 for the importance sampling method for the Delta with lognormal jumps.

Intensity	Delta	Importance standard error	Importance time	Importance time 0.01 standard error
0.1	0.7854	0.0001	7.4060	0.0006
0.2	0.7889	0.0002	7.7970	0.0025
0.5	0.7988	0.0004	7.9690	0.0134
1.0	0.8130	0.0007	8.6880	0.0428
2.0	0.8400	0.0011	10.5000	0.1230
4.0	0.8877	0.0015	14.7340	0.3294
8.0	0.9512	0.0019	24.5000	0.8866

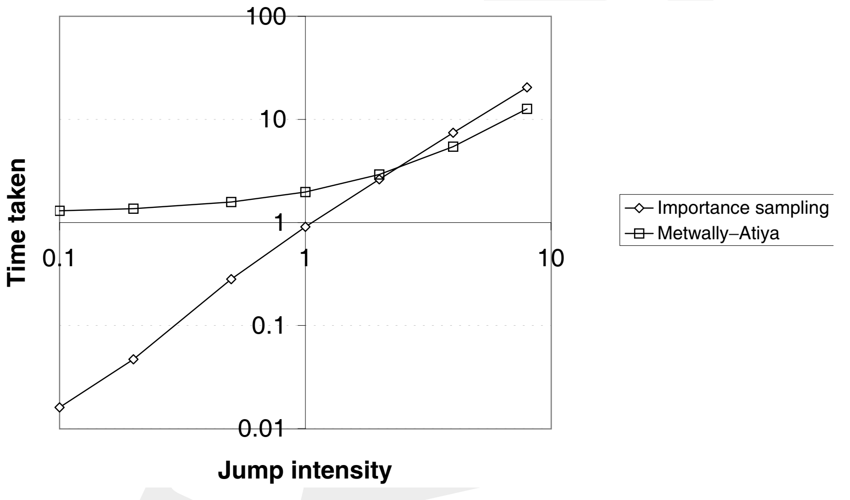
alternative method of computing the cumulative normal function, for example, a table.

This suggests that for modeling barrier options on underlyings such as equity indices where one would expect a jump intensity much smaller than one, the importance sampling method is the best choice.

TABLE 6 The standard error, time taken and the estimated time to achieve an error 0.01 for the Metwally–Atiya method for the Delta with lognormal jumps. The final column is the ratio of the time to get 0.01 for the importance sampling to the Metwally–Atiya method.

Jump intensity	Standard error	Time	Time 0.01 standard error	Ratio of times
0.1	0.033	1.141	12.537	0.00005
0.2	0.036	1.187	15.664	0.00016
0.5	0.035	1.313	15.667	0.00085
1.0	0.038	1.516	21.815	0.00196
2.0	0.040	1.890	30.662	0.00401
4.0	0.044	2.672	51.203	0.00643
8.0	0.049	4.297	103.046	0.00860

FIGURE 1 The amount of time required to achieve a standard error of 0.01 for the price.

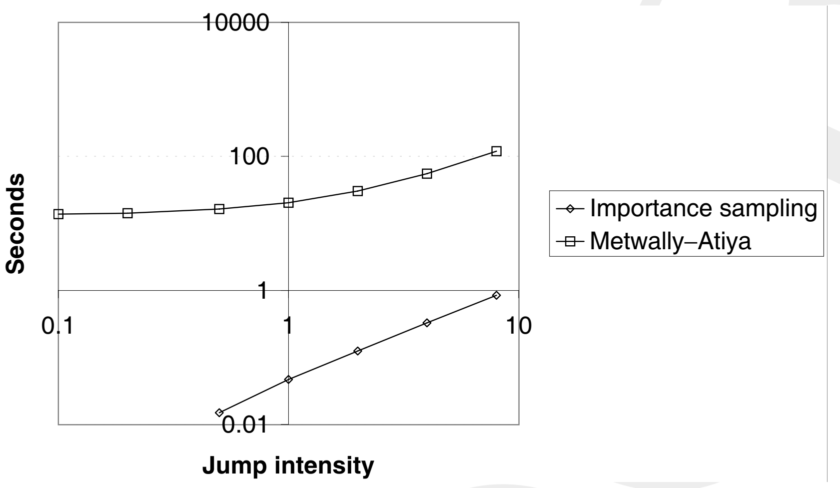


It is not sufficient for a numerical method to be good at computing the price, it must also be fast for computing Greeks; we therefore also examine the number of paths and time required to achieve a standard error of 0.01 ± 0.0001 for the Delta. This is computed by symmetric differencing with $\epsilon = 0.01$:

$$\Delta = \frac{f(S+\epsilon) - f(S-\epsilon)}{2\epsilon}$$

The amount of time required is plotted in Figure 2. The importance sampling method requires very few paths and is consistently faster by a factor of about 100 than the Metwally–Atiya method. Q1

FIGURE 2 The amount of time required to achieve a standard error of 0.01 for the Delta. Values not shown for importance sampling for small jump intensities as the value is less than 1 clock tick.



8 CONCLUSION

We have presented a new method for computing the price of a continuous barrier option in jump-diffusion models. It relies upon using a mixture of importance sampling and an analytic formula conditioned on no further jumps occurring. The method has been show to be effective for two different choices of jump distribution: it is faster at computing prices for low jump intensities, and consistently much faster at computing the Delta across all jump intensities.

REFERENCES Q2

Broadie, M., and Glasserman, P. (1996). Estimating security derivative prices by simulation. *Management Science* **42**, 269–285.

Broadie, M., Glasserman, P., and Kou, S. (1997). A continuity correction for barrier options. *Mathematical Finance* **7**(4), 325–348.

Broadie, M., and Yamamoto, Y. (2005). A double exponential fast Gauss transform algorithm for pricing discrete path-dependent options. *Operations Research* **53**(5), 764–779.

Feng, L., and Linetsky, V. (2007). Pricing options in jump-diffusion models: an extrapolation approach. *Operations Research*, to appear.

Glasserman, P. (2003). *Monte Carlo Methods in Financial Engineering*. Springer, Berlin.

Glasserman, P., and Staum, J. (2001). Conditioning on one-step survival for barrier option simulations. *Operations Research* **49**(6), 923–927.

Jäckel, P. (2002). *Monte Carlo Methods in Finance*. Wiley, New York.

Joshi, M. (2003). *The Concepts and Practice of Mathematical Finance*. Cambridge University Press, Cambridge.

Kou, S. G. (2000). A jump-diffusion model for option pricing with three properties: leptokurtic feature, volatility smile and analytical tractability. *Contributed Paper to the Econometric Society World Congress*.

Kou, S. G., and Wang, H. (2003). First passage times of a jump diffusion process. *Advances in Applied Probability* **35**(2), 504–531.

Lewis, A. (2003). Path-dependent options under jump-diffusions. *Invited Talk U.S.C. Mathematical Finance Seminar*, 22nd April 2003. http://www.optioncity.net/publications.htm.

Linetsky, V. (2006). Pricing equity derivatives subject to bankruptcy. *Mathematical Finance* **16**(2), 255–282.

Merton, R. (1998). *Continuous-Time Finance*. Blackwell, Oxford.

Merton, R. (1976). Option pricing when underlying stock returns are discontinuous. *J. Financial Economics* **3**, 125–144.

Metwally, S., and Atiya, A. (2002). Using Brownian bridge for fast simulation of jump-diffusion processes and barrier options. *Journal of Derivatives* (Fall), 43–54.

Musiela, M., and Rutowski, M. (1997). *Martingale Methods in Financial Modelling*. Springer, Berlin.

Penaud, A. (2004). Fast valuation of a portfolio of barrier options under options Merton's jump-diffusion hypothesis. *Wilmott Magazine*, September.