Pricing European and Discretely Monitored Exotic Options under the Lévy Process Framework

Dale Olivier Roberts and Alexander Novikov

We shall consider both European and discretely monitored Exotic options (Bermudan and Discrete Barrier) in a market where the underlying asset follows a Geometric Lévy process.

First we shall briefly introduce this extended framework, then using the Variance Gamma model we shall show how to price European Options and then we will proceed to demonstrate the application of the recursive quadrature method to Bermudan and Discrete Barrier Options.

■ Introduction

It is well known that the classic Black-Scholes framework cannot capture a number of phenomena that are found in the financial markets such as the leptokurtic property found in empirical distributions of asset returns. A number of new models have been proposed, such as Stochastic Volatility that incorporates a random volatility and generalisations of the classic framework whereby the price process contains a jump component, that is, our price follows a Lévy process.

We shall first demonstrate how to price European Options when the stock price follows Variance Gamma process, and then we shall present a functional programming implementation of the quadrature method for discretely-monitored options where the stock price is modelled by Geometric Brownian Motion.

■ The Lévy Process Price Model

□ Lévy Processes

Definition

A stochastic process X_t on $(\Omega, \mathcal{F}, P)$ such that $X_0 = 0$ is called a Lévy process if it has the following properties:

- **Stationary increments:**
  The distribution (or law) of the increment $X_{t+\Delta} - X_t$ is independent of the time t.

- **Independent increments:**
  The increments of the process X_{t_1}, $X_{t_2} - X_{t_1}$, ..., $X_{t_k} - X_{t_{k-1}}$ are independent for all times $0 \le t_1 < t_2 < \cdots < t_k$.
- **Stochastically continuous paths:**
  For all $\epsilon > 0$ we have $\lim_{\Delta \to 0} \mathbb{P}\{\,|X_{t+\Delta} - X_t| \ge \epsilon\} = 0$.

Infinite Divisibility

The most distinctive property of Lévy processes is that of stationary increments, it implies the probability distribution of an increment of length Δ is the same as the distribution of an increment of length $n\,\Delta$ (the sum of n increments), this is called infinite divisibility.

This may also be expressed in terms of Characteristic functions: consider a probability measure μ on $\mathbb{R}$, and its Characteristic function $\mathcal{F}\mu(z) = \int_{\mathbb{R}} e^{ixz}\,d\mu$. The distribution is called infinitely divisible if for any positive integer k, there exists a probability measure μ_k with Characteristic function $\mathcal{F}\mu_k$ such that $\mathcal{F}\mu = (\mathcal{F}\mu_k)^k$.

This property places a restriction on the distributions that may be used for the random variables X_f but number of nice distributions with this property exist, for example: the Student t-distribution, the Log-normal distribution, the Gamma distribution, the Poisson distribution, and the Variance Gamma process.

□ Stock Price Model

We consider a market that consists of a riskless bond whose price follows the deterministic process $B_t = \exp(r\,t)$, and a non-dividend paying stock S_t with price process

$$S_t = S_0 \exp(L_t), \tag{1}$$

where $\{L_t\}_{t\ge 0}$ is a Lévy process under an appropriate risk-neutral (martingale) measure Q. Accordingly to the Lévy-Kchinchin theorem, the Lévy process $\{L_t\}_{t\ge 0}$ has the decomposition

$$L_t = \mu\,t + \sigma\,W_t + Y_t, \tag{2}$$

where W_t is a standard Wiener process and Y_t is a jump Lévy process that is independent of W_t, μ and σ are parameters.

The choice of the particular Lévy process used determines uniqueness of this measure. If the measure Q is not unique this leads the notion of an incomplete market. It is well known that Q is unique only for two special cases: (a) there is no jump component Y_t in (2) or (b) the parameter $\sigma = 0$ and Y_t has only a fixed size jump (i.e., Y_t is like a Poisson process). The choice of measure Q is usually provided by use of an utility function.

Black-Scholes Model

Setting L_t to be a Wiener process, we find ourselves in the classic Black-Scholes framework introduced in 1973 by Black, Scholes and Merton where the bond price is as before and the stock price process follows

$$dS_t = \mu_t\,S_t\,dt + \sigma\,S_t\,dW_t \tag{3}$$

where μ_t is a deterministic function of t. The Stochastic Differential Equation (SDE) in (3) has a unique solution

$$S_t = S_0 \exp\left(\int_0^t \mu_t\,ds + t\,\frac{\sigma^2}{2} + \sigma\,W_t\right). \tag{4}$$

and under the risk-neutral measure Q to obtain a martingale we must have $\mu_t = r_t$, or simply $\mu_t = r$ if we assume a constant risk-free rate r.

Variance-Gamma Model

Although the Black-Scholes model has become the defacto standard in the finance industry, it is well known that the fair prices it produces does not reflect what often occurs in the market for options that are deeply in or out-of-the-money, this was shown by Rubinstein in 1985 [1], and in Madan, Carr, and Chang [2].

The Variance-Gamma process (VG) introduces the notion that market information comes at random time intervals. This concept is modeled by a Wiener process with constant drift evaluated at a random time change given by a Gamma process, this leads to a pure jump process. The VG model has three parameters that allows us to control volatility, kurtosis and skewness and therefore providing a way to calibrate the model to the prices found in the market. Pricing under the Variance Gamma framework was first proposed by Madan and Seneta in 1990 [3] and was extended in 1991 [4], 1998 [2], and 2003 [5].

Under the Variance Gamma framework, the log stock price is defined in terms of a Wiener process with drift θ and volatility σ

$$B(t, \theta, \sigma) = \theta t + \sigma W(t), \tag{5}$$

where the time t follows a Gamma process $T(t, \nu) \sim \gamma(t, 1, \nu)$ with mean rate 1 per unit of time and variance ν which results in the pure jump process that has an infinite number of jumps in any interval of time:

$$X(t, \theta, \sigma, \nu) = B(T(t, \nu), \theta, \sigma), \tag{6}$$

that may be calibrated by three parameters: σ, θ, and ν. Under an equivalent martingale measure, the mean rate of return of the stock is the continuously compounded interest rate r, and the price then evolves as

$$S_t = S_0 \exp(r t + X(t, \theta, \sigma, \nu) + \omega t),$$

where $\omega = \log(1 - \sigma \nu - \frac{1}{2} \sigma^2 \nu)$ is a compensator to ensure that we have a martingale.

Madan [3] showed the characteristic function to be

$$\Phi(u) = \mathbb{E}[\exp(i u X_t)] = \left(\frac{1}{1 - i \theta \nu u + (\sigma^2 \nu / 2) u^2}\right)^{t/\nu}, \tag{7}$$

and the density $h(z)$ for the log price relative $z = \log(S_t / S_0)$ to be written in terms of the modified Bessel function of the second kind $K_n(z)$ as

$$h(z) = \frac{\left(2 e^{\frac{\theta x}{\sigma^2}}\right)}{\nu^{t/\nu} \sqrt{2 \pi}\, \sigma \Gamma(\frac{t}{\nu})} \left(\frac{x^2}{\theta^2 + \frac{2\sigma^2}{\nu}}\right)^{\frac{t}{2\nu} - \frac{1}{4}} K_{\frac{t}{\nu} - \frac{1}{2}}\left(\frac{\sqrt{x^2 \left(\theta^2 + \frac{2\sigma^2}{\nu}\right)}}{\sigma^2}\right), \tag{8}$$

where $x = z - r t - \omega t$.

◻ Simulating Variance Gamma Price Paths

The random variables of the underlying jump process $X(t, \theta, \sigma, \nu)$ may be generated by first drawing a random variable from the Gamma process for the time parameter t and then one from the Standard Normal distribution denoted n, then our random variable x from $X(t, \theta, \sigma, \nu)$ is $x = \theta t + \sigma \sqrt{t}\, n$.

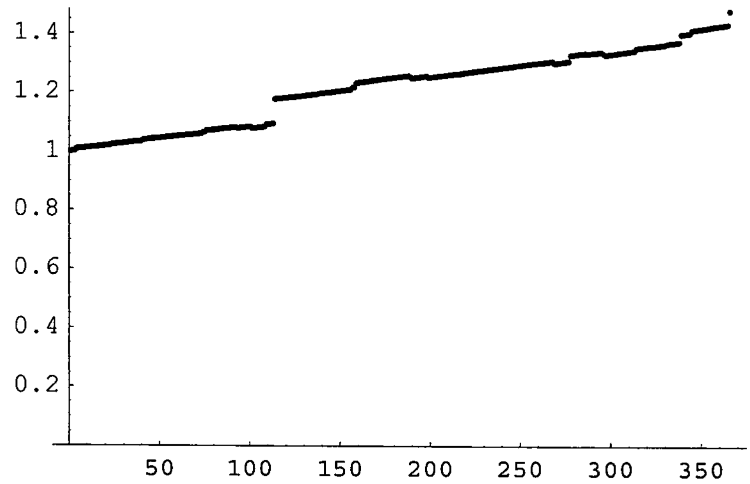
```
<< "Statistics`ContinuousDistributions`";

distGamma = GammaDistribution[tee/nu, nu];
distNormal = NormalDistribution[0, 1];

X[t_, σ_, ν_, θ_] :=
 Block[{T, N, tee = t, nu = ν, ω},
  T = Random[distGamma];
  N = Random[distNormal];
  N √T σ + θ T]
```

A path of a Variance Gamma process may be simulated by taking a discrete approximation of the time dimension. By plotting a simulated path we can clearly see its random jump behaviour.

```
Block[{n = 365, r = 0.10, σ = 0.12, ν = 0.20, θ = -0.14, Δ, ω},
  Δ = 1/n; ω = log(-νσ^2/2 - θν + 1)/ν;
  ListPlot[FoldList[#1 e^(rΔ+ωΔ+#2) &, 1, Array[X(Δ, σ, ν, θ) &, {n}]],
   PlotRange → {0, Automatic}]];
```



It should be noted that even though the price process looks continuous over some regions it is actually composed of many very small jumps with sudden larger jumps. The stochastic continuity condition of Lévy process means that for any given time t, the probability of seeing a jump at t is zero. The discontinuities of the path must occur at random times, this excludes process with jumps at predetermined (nonrandom) times.

■ Pricing European Options with Variance Gamma

To introduce the notion of pricing under the Variance Gamma process, we will start by pricing a simple European option where the payoff is only a function of the price at expiry. In the case of a European Put with strike price K, we have the payoff

$$f(S_T) = \max(0,\ K - S_T), \tag{9}$$

where T is the time of expiry, K is the strike price of the option, and S_T is the stock price at time T. The arbitrage free price V_t of the option at time $t = 0$ is the present value of the expectation, with respect to the risk-neutral martingale measure Q, of the option payoff

$$V_0 = D\, E_Q[\max(0, K - S_T)], \tag{10}$$

where $D = \exp(-r\,T)$ is the discounting factor.

□ Monte-Carlo Simulation

Variance Gamma options may easily be priced using Monte-Carlo simulation. To derive the expected value for a European option as in (10), we only need to simulate a large number of outcomes for the stock price at expiry and then take the average overall outcomes. The option price is then the present value of this average.

We shall first define a function to give a 95% confidence interval for the price given a list of price outcomes:

```
ConfidenceInterval[list_] :=
  Module[{σ = StandardDeviation[list], n = Length[list], μ = Mean[list]},
   Interval[{μ - 1.96 σ/√n, μ + 1.96 σ/√n}]]
```

Then define the option payoff function

```
Λ[S_, K_] := Max[0, K - S];
```

Our parameters for the Option and the market.

```
K = 100.00; S0 = 100.00; T = 1.00; r = 0.10; σ = 0.12; ν = 0.20; θ = -0.14;

ω = Log[-ν σ^2 / 2 - θ ν + 1] / ν;
```

Finally, we proceed with the simulation:

```
mcResult = Module[{outcomes, payoffs, n = 100000},
    outcomes = Array[S0 e^(r T + ω T + X(T,σ,ν,θ)) &, {n}];
    payoffs = (e^(-r T) Λ[#1, K] &) /@ outcomes;
    ConfidenceInterval[payoffs]] // Timing

{25.5543 Second, Interval[{1.8081, 1.86615}]}
```

This example may be extended to path dependant options by simulating a discrete approximation of the price process path (as performed earlier), calculating the payoff for each path, taking the average and discounting.

□ Numerical Integration

For European options, we may alternatively compute the expectation numerically using numerical integration by integrating the payoff of the price process against the density of the normal distribution and the density of the Gamma distribution.

```
𝒩[z_] = PDF[NormalDistribution[0, 1], z];
𝒢[g_] = PDF[GammaDistribution[T / ν, ν], g];

f[g_, z_] := Λ[S0 Exp[r T + ω T + g θ + √g z σ], K];

quadResult = e^(-r T) NIntegrate[f[g, z] 𝒩[z] 𝒢[g], {z, -∞, ∞}, {g, 0, ∞}] // Timing

{0.77771 Second, 1.85377}
```

Which is within the 95% confidence interval found by our Monte-Carlo approach.

IntervalMemberQ[Last[mcResult], Last[quadResult]]

True

Pricing Discretely Monitored Options

Discretely monitored options have payoffs that are triggered by events occurring on discrete times before expiry, for example: Bermudan options, Barrier options, and Lookback options. We shall limit ourselves to the case of Bermudan options and Discrete Barrier options.

Bermudan Options

A Bermudan option is a variation of the American option whereby the early exercise dates are restricted to a finite number throughout the life of the option. This gives the holder of a Bermudan option more rights than holding a European equivalent and less than the American equivalent, thus from an economic point of view it should be obvious that the risk-neutral price of a Bermudan is bounded above by the American and below by the European. Although uncommon in Equity and Foreign Exchange markets, it is often found with a Fixed Income underlying. For example, a Bermudan Swaption can be exercised only on the dates when swap payments are exchanged. By letting the number of exercise dates go to infinity, we may approximate the value of an American Option by a Bermudan Option.

Discrete Barrier Options

A discrete barrier option is monitored at discrete dates before maturity and is either knocked in (comes into existence) or knocked out (is terminated) if the spot price is across the barrier at the time it is monitored. As there is a positive probability of the spot price crossing (or not crossing), barrier options are generally cheaper than 'vanilla' equivalents. Analytical pricing formulas are known but assume continuous monitoring of the barrier but this may not reflect an accurate price. In the real world, barrier options are typically monitored at discrete times, for example at the close of the market. This should not be neglected as the frequency of monitoring has a strong effect on an option's price.

There are six characteristics of a barrier option that define how it should be priced: the barrier could be above or below the initial value of spot (up or down), the barrier could knock in or knock out the option and the option could be a call or a put. This leads to eight barrier options types.

The Recursive Quadrature Approach

Introduction to the Method

Quadrature is a useful tool for the probability theorist as it allows to numerically calculate the expectations in a natural manner without the need to repose the problem in terms of a differential equation or a lattice.

Discretely-monitored options may be priced by first identifying the times where a certain condition must hold and then formulating the expectation of the option in a recursive manner such that the expectation of each discrete time-step is a function of the expectation of the previous step. This technique easily applies to a range of path-dependant options such as discrete-barrier, american, and Bermudan options.

We shall present an implementation of the method proposed by Huang and Subrahmanyam [6], Sullivan [7] and Andricopoulos, Widdicks, Duck and Newton [8] who pose the value of the option at each step i in terms of the risk-neutral expectation of the step $i+1$ which gives

$$V_i = D_i E_Q[V_{i+1}],$$

where $D_i = \exp(-r(t_{i+1} - t_j))$ is the discounting factor between time steps, and V_i is the value of the option at step i. At the terminal step, we have $V_N = f$ where f is the payoff of the option. It can be noted that this method allows time steps to be non-equidistant, though in the following implementation we will take time-steps of equal length to simplify our exposition.

□ Application to a Bermudan Put

Before pricing our Bermudan Put option we must first set some parameters for the contract, the stock and the market: M is the time to expiry of the option in years, K is the strike price, r is the risk-free rate, σ is the volatility of the underlying stock and S is the current price of the stock.

```
M = 0.3333; K = 40.00; r = 0.0488; σ = 0.30; S = 40.00;
```

We shall also introduce the parameter λ which represents the number of standard deviations away from the boundary. Modifying both λ and the accuracy goal of the numerical integration allows tuning of the accuracy and speed of this method as needed.

```
λ = 10;
SetOptions[NIntegrate, AccuracyGoal → 4];
```

We shall price this option under the Black-Scholes framework, so we define the conditional PDF of the risk-neutral distribution with respect to the previous price x and the CDF of the standard normal, noting that we transform the prices so that $y = \log(S_{i+1}/K)$ and $x = \log(S_i/K)$ where S_i is the price at time step i.

```
uz = (r - σ^2/2) Δ; sz = σ √Δ;

<< "Statistics`ContinuousDistributions`";

Ψ[x_] = CDF[NormalDistribution[0, 1], x];
Φ[y_, x_] = PDF[NormalDistribution[uz + x, sz], y];
```

The risk-neutral expectation of the value is broken into two integrals at the implicit boundary b. In the case of a put option, below the boundary we have the Black-Scholes analytic solution.

```
d2[x_, y_] := (Log[y/x] - uz)/sz;
d1[x_, y_] := d2[x, y] - sz;

belowBoundary[b_Real, x_Real] :=
  K e^(-r Δ) Ψ[d2[K e^x, K e^b]] - K e^x Ψ[d1[K e^x, K e^b]]
```

The upper integral takes a function approximation of the previous step (working backwards) and computes the expectation numerically. Using function approximation allows us to not indulge in a recursive calculation at each step.

```
aboveBoundary[func_, b_Real, ymax_Real, x_Real] :=
  e^(-r Δ) NIntegrate[func[y] Φ[y, x], {y, b, ymax}]
```

Thus the value at each step is the sum of these integrals.

```
putValue[func_, b_Real, x_Real] :=
  belowBoundary[b, x] + aboveBoundary[func, b, q, x]
```

The difficulty of Bermudan and American options is the implicit or moving boundary, at each step we must numerically identify the price where we are indifferent to holding the option or exercising the option. Again, as finding this point requires a number of iterations of the value function, function approximation simplifies this greatly.

At each step we must find the boundary of the previous calculated step, calculate the expectation, and create a new function approximation to pass along to the next step.

```
valueStep[data_] :=
  Block[{h},
   h = Interpolation[data];
   b = z /. FindRoot[h[z] == K (1 - e^z), {z, 0.0}];
   generateData[h, b]]
```

Our function approximation is created by sampling the value at evenly spaced points within λ standard deviations distance from the boundary.

```
Ys := Range[Log[S / K] - q, Log[S / K] + q, 2 q / N];
generateData[func_, b_] := {#, putValue[func, b, #]} & /@ Ys
```

To find our option value, we now simply step backwards through time to the present day which gives us a function approximation for a range of stock prices. The function takes two arguments, the first is the number of exercise dates and the second is the number of evenly spaced sampling points for each step. Our option value is equal to the value for our current stock price.

```
Off[InterpolatingFunction::"dmval"];

bermudanPut[nDates_, qPoints_] :=
 Block[{v0, b = Log[S / K], Δ, T, N, Y},
  T = nDates; N = qPoints; Δ = M / T; q = λ σ √Δ;
  v0 = {#, belowBoundary[b, #]} & /@ Ys;
  Interpolation[Nest[valueStep, v0, T - 1]][0.0]]
```

```
bermudanPut[16, 32] // Timing
```

```
{1.40309 Second, 2.47801}
```

We can compare these results against the paper by Sullivan, where the number of points $q = 32$.

Exercise Dates	Mathematica	Sullivan	RMSE
16	2.47801	2.4775	0.0005
64	2.47288	2.4812	0.0083

Table 1. Comparison of results to Sullivan with $q = 32$.

□ Application to a Discrete Barrier "Down-And-Out" Call

We shall now apply this quadrature technique to a discrete "down-and-out" barrier call option with parameters for time to maturity M, strike price K, risk-free rate r, stock volatility σ, and current price S.

```
M = 0.3333; K = 40.00; r = 0.0488; σ = 0.30; S = 40.00;
```

In the case of a discrete barrier option, we now also have a list of stock prices for which the option is knocked out, or in other words rendered useless. Again for simplicity, we shall restrict ourselves to the case where the times are equally spaced but it should be known that this method works equally well for arbitrary times. We start with one knock-out price, and append our strike price at expiry to the list.

```
𝓑 = {35.00}; AppendTo[𝓑, K];
```

We now know how many equally-spaced steps are needed to value this option, and we set our number of sample points to be N.

```
T = Length[𝓑]; N = 32;
```

And, as before, we may tune the accuracy and speed as necessary.

```
λ = 10;
SetOptions[NIntegrate, AccuracyGoal → 4];
```

We now transform the boundaries and define the time step Δ, and again define q as the price change from the boundary.

$$\mathcal{B} = \text{Log}\left[\frac{\mathcal{B}}{K}\right];\ \Delta = \frac{M}{T};\ q = \lambda\,\sigma\sqrt{\Delta};$$

We define the conditional PDF of the stock price change under the transformation $y = \log(S_{i+1}/K)$ and $x = \log(S_i/K)$.

$$\text{uz} = (r - \sigma^2/2)\,\Delta;\ \ \text{sz} = \sigma\sqrt{\Delta};$$

```
<< Statistics`ContinuousDistributions`

Φ[y_, x_] = PDF[NormalDistribution[uz + x, sz], y];
```

Discrete barrier options are somewhat simpler than Bermudan as we know the location of the boundary and for the down-and-out call below the boundary the option is worth zero. This leaves us with only the upper part of the integral to calculate.

```
callValue[func_, b_Real, x_Real] := e^(-r Δ) NIntegrate[func[y] Φ[y, x], {y, b, b + q}];
```

At each step we identify the upper and lower bounds of our price range and then generate a function approximation for the next step, and since we explicitly know the boundary points we no longer need to find them.

```
valueStep[data_, b_] :=
  Block[{h},
    h = Interpolation[data];
    generateData[h, b]]
```

Again, our function approximation is created by sampling the value at evenly spaced points within λ standard deviations distance from the boundary.

```
Ys := Range[Log[S / K] - q, Log[S / K] + q, 2 q / N];
generateData[func_, b_] := {#, callValue[func, b, #]} & /@ Ys
```

To value the option we step through each time-step and find the value of the expectation with respect to the previous step, ensuring that below the barrier the option is worth zero.

```
v0 = {#, Max[0, K (Exp[#] - 1)]} & /@ Ys;
downOutCallResult =
  Interpolation[Fold[valueStep, v0, Reverse[𝓑]]][0.0] // Timing

{0.323439 Second, 3.03237}
```

We may verify this result using a Monte Carlo simulation.

```
DownOutCallMC[S_, K_, M_, σ_, r_, Bs_List, m_] :=
  Block[{Ps, Ss, s, dt, n, Path, rv, μ, Σ},
   n = Length[Bs] + 1; dt = M / n;

   Ss := S Drop[FoldList[#1 Exp[(r - σ^2 / 2) dt + σ Sqrt[dt] #2] &,
         1, RandomArray[NormalDistribution[0, 1], n]], 1];

   Ps = Table[s = Ss; If[Min[Drop[s, -1] - Bs] < 0.0,
        0.0, Exp[-r M] Max[Last[s] - K, 0.0]], {m}];

   ConfidenceInterval[Ps]
  ]

downOutCallInterval = DownOutCallMC[S, K, M, σ, r, {35}, 100000] // Timing

{30.1354 Second, Interval[{2.99638, 3.05439}]}

IntervalMemberQ[Last[downOutCallInterval], Last[downOutCallResult]]

True
```

■ Conclusion

In this paper we have shown how to quickly price European Options under the Variance Gamma process and then we have implemented the recursive quadrature technique, a powerful method that is often forgotten in the literature on option pricing and lacking the needed working examples to allow a quick implementation by industry practitioners.

■ References

[1] M. Rubinstein, "Nonparametric Tests of Alternative Option Pricing Models Using all Reported Trades and Quotes on the 30 Most Active CBOE Option Classes from August 23, 1976 through August 31, 1978," *Journal of Finance*, **40**, 1985 pp. 455–480.

[2] D. Madan, P. Carr and E. Chang, "The Variance Gamma Process and Option Pricing," *European Finance Review*, **2**, 1998 pp. 79–105.

[3] D. Madan and E. Seneta, "The Variance Gamma (V.G.) Model for Share Market Returns," *Journal of Business*, **63**(4), 1990 pp. 511–524.

[4] D. Madan and F. Milne, "Option Pricing with VG Martingale Components," *Mathematical Finance*, **1**(4), 1991 pp. 39–55.

[5] P. Carr, H. Geman, D. Madan, and M. Yor, "Stochastic Volatility for Lévy Processes," *Mathematical Finance,* **13**(3), 2003 pp. 345–382.

[6] J. M. Huang and G. Y. Subrahmanyam, "Pricing and Hedging American Options: A Recursive Integration Method," *Review of Financial Studies*, **9**, 1996 pp. 277–300.

[7] M. A. Sullivan, "Valuing American Put Options using Gaussian Quadrature," *Review of Financial Studies*, **13**, 2000 pp. 75–94.

[8] A. D. Andricopoulos, M. Widdicks, P. W. Duck, and D. P. Newton, "Universal Option Valuation using Quadrature Methods," *Journal of Financial Economics*, **67**, 2003 pp. 447–471.

About the Authors

Dale Olivier Roberts is a Honours student at the Department of Mathematical Sciences at the University of Technology, Sydney (UTS).

Alexander Novikov is Professor of Probability at the same department and Senior Researcher at Steklov Mathematical Institute, Moscow.

Dale Olivier Roberts and Alexander Novikov
Department of Mathematical Sciences
University of Technology Sydney
P O Box 123
Broadway NSW 2007
Australia
dale.o.roberts@uts.edu.au
alex.novikov@uts.edu.au

International Mathematica Symposium 2005

On behalf of the Editors

This year, 44 abstracts were submitted to IMS. After reviewing the abstracts, 33 authors were invited to submit full papers. These full papers were reviewed by two referees and, after review, 31 papers were accepted to appear in the electronic proceedings, published as a CD with ISBN and, simultaneously, at physics.uwa.edu.au/pub/IMS/2005/Proceedings. We would like to thank the Program Committee for their excellent work on reviewing the submitted papers.

Proceedings record what was said at a conference. Electronic proceedings should enhance this record—being not just a static collection of papers, but a dynamic set of documents that encourage interaction by the reader. With *Mathematica* Notebooks as the medium of all presentations, achieving this goal was straightforward. We would like to thank Glenn Scholebo from The *Mathematica* Journal for his assistance in formatting all papers consistently, and to help make best use of the extensive features that the Notebook environment provides: such as animation, automatic numbering, hyperlinking, and support of multiple screen and print environments. Special effort has gone into editing the papers and the production of the electronic proceedings, and we hope you'll agree that the quality of this year's papers is the best of any IMS thus far.

The argument about whether proceedings should be printed or provided in electronic-only format is likely to continue for a few more years. This year, again, we have compromised by arranging for selected best papers to be published as a special issue of *The Mathematica Journal.*

To conclude, there is now a true integration of material traditionally presented using a static medium (paper) with a dynamic medium (the *Mathematica* notebook). Very few technical problems were encountered in the production of these proceeding, and it is fair to claim that we now have a fully workable dynamic *eProceedings*.

Paul Abbott and Shane McCarthy
School of Physics
The University of Western Australia
Crawley, Australia

2 August 2005

From the Host

Welcome to the 7th International *Mathematica* Symposium. I would like to thank all international participants for making the long journey to Perth, Western Australia—the most isolated capital city in the world!

The main goal of IMS is to bring together *Mathematica* users from all over the world. Past IMS conferences have been held at widely dispersed locations—Southampton (1995), Rovaniemi (1997), Linz (1999), Tokyo (2001), London (2003), and Banff (2004)—but IMS 2005 is the first conference to be held in the southern hemisphere, and in our Winter.

The main reason why we all come to IMS is our shared passion for *Mathematica*, which, for most of us, is the key scientific computing environment used in our daily research. IMS has always provided a wonderful forum for *Mathematica* users across a wide range of disciplines, and this is exactly why the symposium has always been a very special conference. The multidisciplinary nature of IMS encourages interdisciplinary discussion and the exchange of ideas. Interdisciplinary research requires tools for communicating our ideas—and *Mathematica* is the best such communication tool.

IMS is now run annually. This change poses several organizational challenges, principally the reduction of time for paper preparation. However, this did not significantly affect the number of papers that will be presented at this year's symposium and there are many high-quality contributions in the IMS proceedings. I would like to thank the program committee for reviewing all submissions. And I am very grateful to all the authors for their submissions and their timely replies to our requests regarding their final papers.

Again this year, the electronic proceedings are on CD. The electronic medium is the only way to fully utilize the potential of *Mathematica* notebooks, enabling the move towards 'live proceedings', where one can not only read about the research, but can immediately try out the examples and modify them—a true live experience.

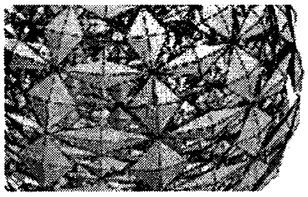
A conference like this cannot be organized without help from many people. I would like to thank:

- Shane McCarthy (content) and Glen Scholebo (technical editing) who spent countless hours editing the IMS 2005 Proceedings.
- my good friend and colleague, John Brookes from *Analytica*, who handled all the tedious administrative details, including general organisation, the budget and accounts, registration, accommodation, venue hire, events, buses, and other tasks too numerous to mention. Thank you!
- the School of Physics. My head of school, Ian McArthur allowed me the time-off, and Lydia Brazzale, our secretary, helped with ordering the badges, satchels, and CDs.
- Nick Loh for the design and production of the conference T-shirt and for creating the local area map showing cafes and restaurants.
- Michael Eilon for setting up internet access and configuration of *Mathematica* in the teaching laboratories.
- Julie Harrison of The University Club for helping with all issues related to the venue hire.
- Lyn Ellis from the Raine Foundation and Janette Atkinson from the Australia College of Physical Scientist and Engineers in Medicine for partial funding of Bart M ter Haar Romenij's visit.
- Terri-ann White from the Institute of Advanced Studies for arranging Sarah Flannery's workshop.
- Jean Buck and several other WRI staff members for their vital support of IMS.
- this year's sponsors: Wolfram Research (for funding 4 invited speakers), The University of Western Australia (for covering the venue hire), the Faculty of Life and Physical Sciences, the Perth Convention Bureau,, and *Analytica*.

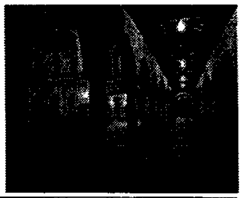
I am looking forward to a successful and enjoyable IMS 2005 and hope that all overseas attendees will enjoy their visit to Australia.

Paul Abbott
School of Physics
The University of Western Australia
Crawley, Australia

August 2, 2005



INTERNATIONAL MATHEMATICA SYMPOSIUM 2005



Home
Call For Abstracts
Speakers
Registration
Program
Events
Proceedings
Training Courses
Venue
Accommodation
Program Committee
Sponsors
Contact Us

IMS2005
7th International *Mathematica* Symposium
5-8 August 2005
The University Of Western Australia
Perth, Western Australia

Introduction

The International *Mathematica* Symposium is an interdisciplinary conference for *Mathematica* users in mathematics, natural and life sciences, social sciences and law, engineering, graphics and design, arts and music, education, industry, and commerce.

Background and Focus

If you use *Mathematica* in research or teaching, or if you are developing products based on *Mathematica*, then the Symposium is an opportunity to share your results with like-minded colleagues. IMS has also built up a deserved reputation as an exceptionally convivial and friendly gathering.

Call for Abstracts

The Symposium is refereed, and a published Proceedings (book or CD with ISBN) is produced for each conference.

The closing date for abstracts is 28 February 2005. Please submit your proposals by email, preferably as *Mathematica* Notebooks, to Paul Abbott [**paul@physics.uwa.edu.au**]. Abstracts will be reviewed by the Program Committee and full papers will be requested for accepted abstracts. All full papers will appear on the CD proceedings and selected best papers will appear in the edited printed proceedings.

Call for Exhibits

The IMS 2005 Organizing Committee also seeks proposals for

- Discussion Panels and Workshops
- Art Exhibitions & Installations
- Poster Presentations
- Software Demonstrations, and
- Displays / Exhibition Booths by vendors and sponsors

The closing date for such submissions is 30 April 2005.

Important Dates
Abstract Submission Due Date: 28 February 2005
Notification of Acceptance: 15 March 2005
Closing Date for Exhibits and Workshops: 30 April 2005
Submission of Full Papers: 30 May 2005
Conference Dates: 5-8 August 2005

Report on
"Pricing discretely monitored exotic options under the Levy process framework"

Summary:

Roberts and Novikov have produced an interesting paper on a timely topic. It's goal is to demonstrate the use of Mathematica and recursive integration for the evaluation of certain discretely monitored exotic options, under Levy processes. The overall approach (recursive integration for the expectation) is well-known and sound. The novelty here is applying the method to certain Levy processes beyond Geometric Brownian motion. (GBM). The Mathematica code is well-structured.

To apply the method to (non-GBM) Levy processes, you either need (i) a closed-form expression for the transition density or (ii) you have to derive this density from characteristic function. In the first case, the presentation is fairly convincing that the method works effectively (at least for the Variance Gamma model). But, even in this case (and throughout), the paper is lacking either error estimates or comparison with other researcher's numerical results. In case (ii), the paper is much less convincing because the only application is to simple GBM. My recommendation is that the paper can be accepted if these fairly major revisions can be made successfully: (I) errors estimates or referenced comparisons to others for *all* numerical results, and (II) a 'penny accurate' (3 digit) demonstration of a non-GBM example for the characteristic function case.

In addition, I suggest below a number of other revisions that should improve the presentation:

Other suggestions.

1. pg 1: Abstract needs revision: neither lookbacks, nor time-dependent parameters are treated.

2. pg 2: Infinitely divisibility is defined differently, through characteristic functions.

3. typo in (4)

4. pg 3: typos in S(t) and omega; sentence surrounding (8) is very awkward.

5. pg 4: would be more interesting to simulate VG and show the results with the market parameters used later.

6. pg 5: Monte Carlo routines should always report std. errors.

7. pg. 7: "below the boundary" really should be elaborated upon. A Plot of the critical exercise points with some annotation might help explain the recursion better.

8. pg 9: Bermudan put; switched from parameter theta to mu; should stick with one.