Finite differences scheme for Option pricing with P.I.D.E's

João Guerra

CEMAPRE and ISEG, UTL

December 3, 2014

The pricing P.I.D.E.

- In general, for European and Barrier Options, the pricing P.I.D.E. is (after introducing the new variables $\tau = T - t$, $x = \ln(S/S_0)$ and defining $h(x) = H(S_0 e^x)$ and $u(\tau, x) = e^{r\tau} C(T - \tau, S_0 e^x)$, where H is the payoff and C is the price of the option):

$$\begin{cases} \frac{\partial u}{\partial \tau} = L^X u + r\frac{\partial u}{\partial x} = Lu & \text{if } (\tau, x) \in \left]0, T\right] \times O, \\ u(0, x) = h(x) & \text{if } x \in O, \end{cases} \tag{1}$$

  where $O = \mathbb{R}$ for an European option, or $O = \left]a, b\right[$ for a Barrier option (in the case of a barrier option, appropriate boundary conditions should also be imposed outside O).
- L^X is the infinitesimal generator associated to the Lévy process X (where $S_t = \exp(rt + X_t)$, under the risk neutral measure) and $L = L^X + r\frac{\partial}{\partial x}$ is

$$Lu = \frac{\sigma^2}{2}\frac{\partial^2 u}{\partial x^2} - \left(\frac{\sigma^2}{2} - r\right)\frac{\partial u}{\partial x} + \\ + \int_{\mathbb{R}} \left[u(\tau, x + y) - u(\tau, x) - (e^y - 1)\frac{\partial u}{\partial x}\right] \nu(dy) \tag{2}$$

Numerical scheme

- Numerical scheme (for finite activity processes):
  1. Truncation of large jumps
  2. Localization
  3. Discretization
- 1. Truncation of large jumps: the domain $]-\infty, +\infty[$ is truncated to a bounded interval $]B_l, B_r[$- this removes large jumps.
- Usually, the tails of ν decrease exponentially $\Longrightarrow$ the probability of large jumps is very small $\Longrightarrow$ we can truncate these jumps.

Localization

- 2. Localization: for barrier options (when $O =]a, b[$), the barrier levels a and b are the natural limits for the domain definition.
- Localization for European options: in the absence of barriers, choose artificial bounds $]-A_l, A_r[$ and impose artificial boundary conditions

$$u(\tau, x) = g(\tau, x), \quad \forall x \notin]-A_l, A_r[\,, \tau \in [0, T]\,.$$

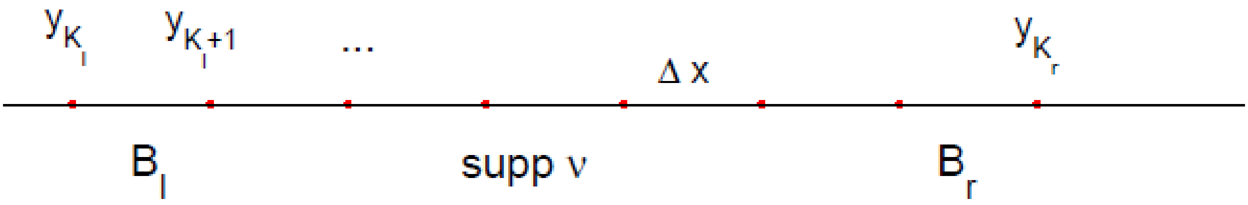
  For instance, $g(\tau, x) = h(x + r\tau)$, where h is the modified payoff function.
- Example: for a put option, we have $h(x) = (K - S_0 e^x)^+$ and therefore, we can assume the boundary conditions

$$u(\tau, x) = g(\tau, x) = h(x + r\tau) = \left(K - S_0 e^{x + r\tau}\right)^+ \quad \text{if } x \notin]-A_l, A_r[\,.$$

Discretization

- We consider the localized problem on $]-A_l, A_r[$:

$$\begin{cases} \frac{\partial u}{\partial \tau} = Lu \ \text{ if } (\tau, x) \in]0,T] \times]-A_l, A_r[\,, \\ u(0,x) = h(x) \qquad \text{if } x \in]-A_l, A_r[\,, \\ u(\tau,x) = g(\tau,x), \qquad \text{if } x \notin]-A_l, A_r[\,. \end{cases} \tag{3}$$



Discretization

- In the finite activity case, we can write the localized version of (2) as

$$Lu = \frac{\sigma^2}{2}\frac{\partial^2 u}{\partial x^2} - \left(\frac{\sigma^2}{2} - r\right)\frac{\partial u}{\partial x} + \\ + \int_{B_l}^{B_r} u(\tau, x+y)\,\nu(dy) - \lambda u - \alpha \frac{\partial u}{\partial x},$$

  where $\lambda := \nu(\mathbb{R}) < \infty$ (in the finite activity case) and $\alpha := \int_{B_l}^{B_r} (e^y - 1)\,\nu(dy)$.
- Uniform grid on $[0,T] \times [-A_l, A_r]$:

$$\tau_n = n\Delta t, \quad n = 0,1,...,M \text{ and } \Delta t = \frac{T}{M},$$
$$x_i = -A_l + i\Delta x, \quad i = 0,1,...,N \text{ and } \Delta x = \frac{(A_l + A_r)}{N}.$$

Discretization

- Discrete values: $u_i^n := u(\tau_n, x_i)$.
- Finite difference approximations:

$$\left(\frac{\partial u}{\partial x}\right)_i \approx \frac{u_{i+1} - u_i}{\Delta x} \text{ or } \frac{u_i - u_{i-1}}{\Delta x}, \tag{4}$$

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_i \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2} \tag{5}$$

$$\int_{B_l}^{B_r} u(\tau, x_i + y)\, \nu(dy) \approx \sum_{j=K_l}^{K_r} \nu_j u_{i+j}, \tag{6}$$

where

$$\nu_j := \int_{\left(j-\frac{1}{2}\right)\Delta x}^{\left(j+\frac{1}{2}\right)\Delta x} \nu(dy).$$

The pricing P.I.D.E. for exponential Lévy models

Discretization

- The limits K_l and K_r are integers such that $[B_l, B_r] \subset \left[\left(K_l - \frac{1}{2}\right)\Delta x, \left(K_r + \frac{1}{2}\right)\Delta x\right]$.
- Using Eqs. (4), (5) and (6), we obtain

$$Lu \approx D_\Delta u + J_\Delta u,$$

where

$$(D_\Delta u)_i = \frac{\sigma^2}{2}\left(\frac{u_{i+1} - 2u_i + u_{i+1}}{(\Delta x)^2}\right) - \left(\frac{\sigma^2}{2} - r\right)\left(\frac{u_{i+1} - u_i}{\Delta x}\right),$$

$$(J_\Delta u)_i = \sum_{j=K_l}^{K_r} \nu_j u_{i+j} - \lambda u_i - \alpha\left(\frac{u_{i+1} - u_i}{\Delta x}\right).$$

Explicit scheme

- Explicit scheme:
$$\frac{u^{n+1}-u^n}{\Delta t} = D_\Delta u^n + J_\Delta u^n$$
or
$$u^{n+1} = \left[I + \Delta t\left(D_\Delta + J_\Delta\right)\right] u^n.$$

- In order for this scheme to be stable, one must impose conditions on Δt.
- A sufficient condition for stability is
$$\Delta t \leq \inf\left\{\frac{1}{\lambda}, \frac{(\Delta x)^2}{\sigma^2}\right\}.$$
The term $\frac{(\Delta x)^2}{\sigma^2}$ forces Δt to be very small and increases computation time.

Implicit scheme

- Implicit scheme:
$$\frac{u^{n+1}-u^n}{\Delta t} = D_\Delta u^{n+1} + J_\Delta u^{n+1}$$
or
$$\left[I - \Delta t\left(D_\Delta + J_\Delta\right)\right] u^{n+1} = u^n.$$

- This scheme is stable but we have to solve a linear system at each iteration.
- In the case of diffusion models ($J = 0$), the matrix $I - \Delta t D_\Delta$ is tridiagonal and the linear system is easy to solve.
- However, in the presence of jumps ($J \neq 0$), the matrix J_Δ is a dense matrix (in general, all the terms of J_Δ can be nonzero) and in order to solve the linear system we need $O\left(N^2\right)$ operations.

General scheme

- General θ scheme

$$\frac{u^{n+1}-u^n}{\Delta t} = \theta\left(D_\Delta u^n + J_\Delta u^n\right) + (1-\theta)\left(D_\Delta u^{n+1} + J_\Delta u^{n+1}\right).$$

- For $\theta = 1$, we recover the explicit scheme, but for $\theta \neq 1$, the computational complexity is the same as for the implicit scheme.
- If $J = 0$ (diffusion model) then an implicit scheme is a good choice.
- If $D = 0$ (pure jump model) then an explicit scheme should be chosen.
- What if $J \neq 0$ and $D \neq 0$, like in the jump-diffusion case? Explicit-implicit scheme.

Explicit-implicit scheme

- Explicit-implicit scheme:

$$\frac{u^{n+1}-u^n}{\Delta t} = D_\Delta u^{n+1} + J_\Delta u^n,\ \tau_n = n\Delta t,\ n = 0, 1, ..., M.$$

This leads to the tridiagonal linear system:

$$\left[I - \Delta t D_\Delta\right] u^{n+1} = \left[I + \Delta t J_\Delta\right] u^n.$$

- Algorithm:
  1. Initialization:
  $$u_i^0 = h(x_i) \text{ if } i \in \{0, ..., N-1\},$$
  $$u_i^0 = g(0, x_i) \text{ otherwise.}$$
  2. For $n = 0, ..., M-1$,
  $$\frac{u_i^{n+1}-u_i^n}{\Delta t} = \left(D_\Delta u^{n+1}\right)_i + \left(J_\Delta u^n\right)_i, \text{ if } i \in \{0, 1, ..., N-1\}, \quad (7)$$
  $$u_i^{n+1} = g((n+1)\Delta t, x_i), \quad \text{otherwise.}$$

Explicit-implicit scheme

- The non-local operator J is treated explicitly to avoid the inversion of the dense matrix J_Δ, while the differential part D is treated implicitly.
- At each time step, we first evaluate vector $J_\Delta u^n$ where u^n is known from the previous iteration, and then solve the tridiagonal system (7) for $u^{n+1} = \left(u_0^{n+1}, u_1^{n+1}, ..., u_{N-1}^{n+1}\right)$.
- The explicit-implicit scheme is stable if

$$\Delta t < \frac{\Delta x}{|\alpha| + \lambda \Delta x}$$

- See Cont and Voltchkova (2005) or Tankov and Voltchkova (2009).

The pricing P.I.D.E. for exponential Lévy models

Explicit-implicit scheme

- Consistence: The explicit-implicit scheme is consistent with the P.I.D.E. (3).
- Monotony and stability: The explicit-implicit scheme is monotone and stable if $\Delta t < \frac{\Delta x}{|\alpha|+\lambda\Delta x}$.
- One can prove that the explicit-implicit scheme is also convergent and that the approximate solution converges uniformly to the unique solution of (1)- See Cont and Voltchkova (2005) or Cont and Tankov (2004).

Stability, consistency and monotonicity

- A scheme is stable is for any bounded initial condition, the solution u_i^n is uniformly bounded at all points of the grid, independently of Δt and Δx :

$$\exists C > 0 : \forall \Delta t > 0, \Delta x > 0, i, n, \ |u_i^n| \leq C.$$

- Stability ensures that the numerical solution at a given point does not blow up when $(\Delta t, \Delta x) \to 0$.
- A numerical scheme is consistent with the P.I.D.E. (3) if the discretized operator converges to its continuous version when applied to any test function $v \in C^{\infty}([0,T] \times \mathbb{R})$, when $(\Delta t, \Delta x) \to 0$.
- A scheme is monotone if $u^0 \geq v^0 \Longrightarrow \forall n \geq 1$, $u^n \geq v^n$.

Numerical Examples

- We illustrate the performance of the scheme proposed above in two examples. (See Cont and Tankov (2004) or Cont and Voltchkova (2005)).
- Model 1: Variance Gamma model with Lévy density

$$\nu(x) = a\frac{\exp(-\eta_{\pm}|x|)}{|x|},$$

  and two sets of parameters $a = 6.25$, $\eta_- = 14.4$, $\eta_+ = 60.2$ (VG1) and $a = 0.5$, $\eta_- = 2.7$, $\eta_+ = 7.9$.
- Model 2: Merton model (jump-diffusion) with Gaussian jumps and log-price with Lévy density (the intensity of the standard Poisson proc. is $\overline{\lambda} = 0.1$):

$$\nu(x) = 0.1\frac{e^{-x^2/2}}{\sqrt{2\pi}}$$

  and volatility $\sigma = 15\%$.
- Option: put option with maturity 1 year such that $h(x) = (1 - e^x)^+$.

Numerical Examples

- Performance of the scheme when compared to the FFT method (of Carr and Madan).
- Errors computed in terms of Black-Scholes implied volatility

$$\varepsilon(\tau, x) = \left|\Sigma^{PIDE}(\tau, x) - \Sigma^{FFT}(\tau, x)\right|,$$

where Σ denotes the Black-Scholes implied volatility computed by inverting the Black-Scholes formula with respect to the volatility parameter and applying it to the computed option price.

- We have computed both pointwise errors at $x = 0$ (i.e. forward at-themoney options) and uniform errors on the computational range $x \in [log(2/3), log(2)]$. This range contains all options prices quoted on the market.

Numerical Examples

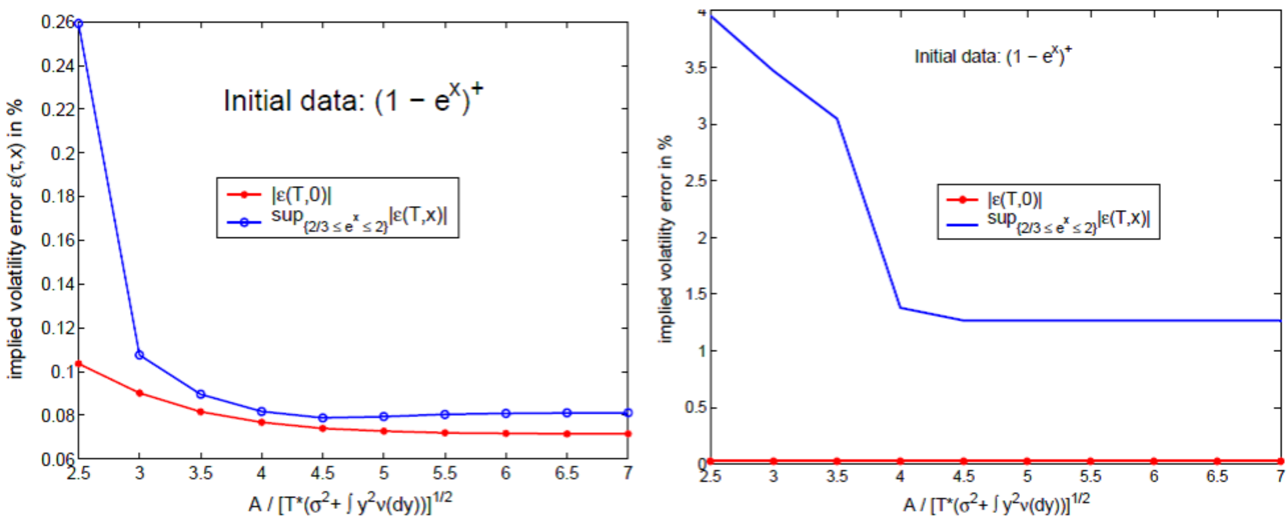


Figure 1: Influence of domain size on localization error for the explicit-implicit finite difference scheme. Left: Merton jump-diffusion model. Right: Variance Gamma model.

Numerical Examples

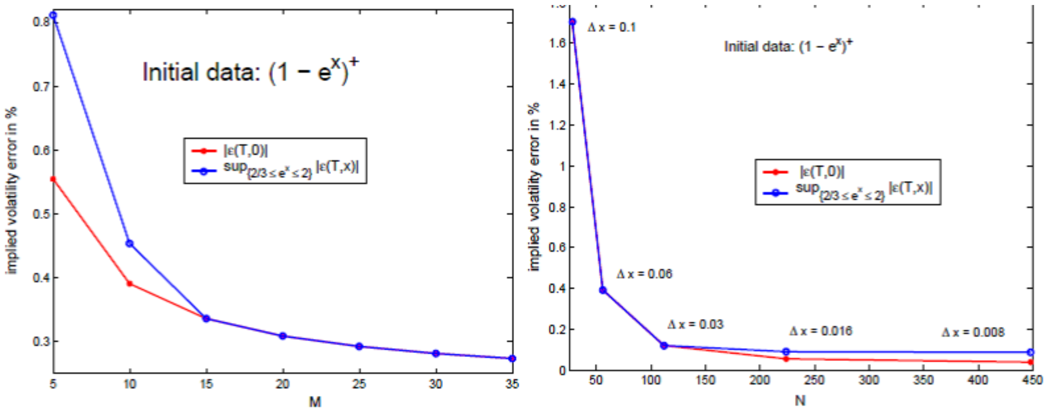


Figure 2: Numerical accuracy for a put option in the Merton model. Left: Influence of number of time steps M. $\Delta x = 0.05$, $\Delta t = T/M$. Right: influence of number of space steps N. $\Delta x = 2A/N$, $\Delta t = 0.02$.

Numerical Examples

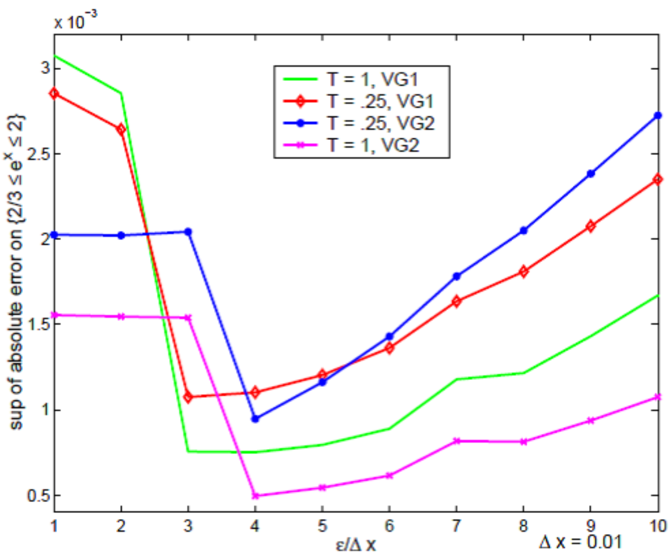


Figure 4: Influence of truncation of small jumps on numerical error in various Variance Gamma models. Put option.

Numerical Examples

- The localization error is shown in Figure 1: domain size A is represented in terms of its ratio to the standard deviation of X_T. An acceptable level is obtained for values of order 5.
- Figure 2 illustrates the decay of numerical error when $\Delta t, \Delta x \to 0$.
- Figure 4 confirms that, for a given $\Delta x > 0$, the minimal error is obtained for a finite ε which in this case is larger than Δx. The optimal choice of ε depends on the growth of the Lévy density near zero.
- In the Table, some examples of option values obtained with the numerical scheme are listed

Model	Put	t sec.	Up-and-out call $H = 120$	t sec.	Double-barrier put $L = 80,\ H = 120$	t sec.
VG1	6.72	0.5	2.73	0.2	2.42	0.1
VG2	8.38	0.9	3.34	0.5	1.68	0.1
Merton	11.04	1.2	1.17	0.5	3.35	4

Cont, R. and P. Tankov (2004). Financial modelling with jump processes. Chapman and Hall/CRC Press, Chapter 12.

Cont, R. and Voltchkova, E. (2005). Finite difference methods for option pricing in jump-diffusion and exponential Lévy models, SIAM Journal on Numerical Analysis 43(4), 1596–1626.

Tankov, P. and Voltchkova, E. (2009). Jump-diffusion models: a practitioner's guide, Banque et Marchés, No. 99, March-April 2009. Available in http://www.proba.jussieu.fr/pageperso/tankov/tankov_voltchkova.pdf