Notes on the BENCHOP implementations for the FDNU method

Jari Toivanen (toivanen@stanford.edu, jari.toivanen@jyu.fi)

May 6, 2015

Abstract

This text describes the FD-NU method and its implementation for the BENCHOP-project.

1 Spatial discretizations

For example, under the Black-Scholes model European option prices u satisfy the PDE

$$u_t(s,t) + \frac{1}{2}\sigma^2 s^2 u_{ss}(s,t) + rsu_s(s,t) - ru(s,t) = 0, \quad s > 0, \quad t \in [0,T), \quad (1)$$

where σ and r are the volatility and interest rate, respectively.

We employ quadratically refined grids defined by

$$s_i = \left[\left(\frac{i}{\gamma n} - 1\right)\left|\frac{i}{\gamma n} - 1\right| + 1\right] K, \quad i = 0, 1, \ldots, n,$$

where K is the strike price. The constant γ is chosen to be $\frac{4}{10}$ except for the barrier options and under the Merton model. For the European spread options, the grids for the both spatial directions are given by the above formula with $K = 100$.

For the Heston model, the variance grid is defined by

$$v_j = \left(\frac{j}{n_v}\right)^2, \quad j = 0, 1, \ldots, n_v.$$

The spatial derivaties are mainly discretized using the central finite differences. Let the grid steps be denoted

$$\Delta s_i = s_{i+1} - s_i, \quad i = 0, 1, \ldots, n-1.$$

Then the approroximations for the first-order and second-order spatial derivatives are

$$u_s(s_i) \approx \frac{1}{\Delta s_{i-1} + \Delta s_i}\left[-\frac{\Delta s_i}{\Delta s_{i-1}} u_{i-1} + \left(\frac{\Delta s_i}{\Delta s_{i-1}} - \frac{\Delta s_{i-1}}{\Delta s_i}\right) u_i + \frac{\Delta s_{i-1}}{\Delta s_i} u_{i+1}\right]$$

and

$$u_{ss}(s_i) \approx \frac{2}{\Delta s_{i-1} + \Delta s_i} \left[\frac{1}{\Delta s_{i-1}} u_{i-1} - \left(\frac{1}{\Delta s_{i-1}} + \frac{1}{\Delta s_i} \right) u_i + \frac{1}{\Delta s_i} u_{i+1} \right],$$

respectively, where u_i denotes the grid point value $u(s_i)$.

Appying the above finite differences lead to semi-discrete system

$$\dot{\mathbf{u}} + \mathbf{A}\mathbf{u} = \mathbf{b}.$$

When the volality is low compared to the interest rate the first-order spatial derivative term dominates the second-order spatial derivative term in the PDE (1). It is well-known that in such a case the central finite differences are not accurate. In this case, we employ a semi-Lagrangian discretization [4] along the trajectory $a(s,\tau) = se^{r(\tau-t)}$ at the given time t. Thus, $a_\tau(s,\tau) = rse^{r(\tau-t)}$. The Lagrangian derivative along this trajectory is

$$\frac{Du}{Dt}(a(s,t),t) = u_t(a(s,t),t) + u_s(a(s,t),t)a_\tau(s,t).$$

Thus, we have

$$u_t(a(s,t),t) = \frac{Du}{Dt}(a(s,t),t) - u_s(a(s,t),t)a_\tau(s,t) = \frac{Du}{Dt}(a(s,t),t) - rsu_s(s,t).$$

Substituting this to (1) leads to

$$\frac{Du}{Dt}(a(s,t),t) + \frac{1}{2}\sigma^2 s^2 u_{ss}(a(s,t),t) - ru(a(s,t),t) = 0.$$

Thus, this eliminates the problematic first-order derivative term.

For the integral in the Merton model a quadrature is employed; for details see [6]. The treatment of jumps lead to matrix-vector multiplications which are performed without any FFT acceleration.

2 Temporal discretizations

Rannacher smoothed [5] Crank-Nicolson and IMEX-CNAB schemes are used for pure diffusion models and jump-diffusion models, respectively. The uniform time step $\Delta t = T/m$ is employed.

The smoothing of the Crank-Nicolson method is performed by taking the first 2 m_E half time steps by the implicit Euler method given by

$$\left(\mathbf{I} + \tfrac{\Delta t}{2}\mathbf{A}\right)\mathbf{u}^k = \mathbf{u}^{k+1/2} + \tfrac{\Delta t}{2}\mathbf{b}^k, \quad k = m - \tfrac{1}{2}, m-1, \ldots, m-m_E.$$

Four implicit Euler smoothing steps are used ($m_E = 2$) except for the spread option eight implicit Euler steps are used ($m_E = 4$). Ater that Crank-Nicolson steps

$$\left(\mathbf{I} + \tfrac{\Delta t}{2}\mathbf{A}\right)\mathbf{u}^k = \left(\mathbf{I} - \tfrac{\Delta t}{2}\mathbf{A}\right)\mathbf{u}^{k+1} + \tfrac{\Delta t}{2}\left(\mathbf{b}^k + \mathbf{b}^{k+1}\right), \tag{2}$$

$k = m - m_E - 1, \ldots, 0$, are performed.

With the semi-Lagrangian discretization the Crank-Nicolson method reads

$$\left(\mathbf{I} + \tfrac{\Delta t}{2}\widetilde{\mathbf{A}}\right)\mathbf{u}^k = \left(\mathbf{I} - \tfrac{\Delta t}{2}\widetilde{\mathbf{A}}\right)\mathbf{P}\mathbf{u}^{k+1} + \tfrac{\Delta t}{2}\left(\mathbf{b}^k + \widetilde{\mathbf{b}}^{k+1}\right), \quad k = m - m_E - 1, \ldots, 0,$$

where the matrix $\mathbf{P}$ interpolates the values at the points $(s_0, s_1, \ldots, s_m)e^{r\Delta t}$ based on the grid points values at $(s_0, s_1, \ldots, s_m)$. In the Matlab implementation, we use the built-in cubical interpolation. The right-hand side $\widetilde{\mathbf{b}}^{k+1}$ is computed with a similar correction. The matrix $\widetilde{\mathbf{A}}$ is $\mathbf{A}$ without the first-order spatial term.

For models with jumps the IMEX-CNAB scheme is employed; see [7]. The jump operator is treated explicitly using the Adams-Bashfort method and the rest implicitly using the Crank-Nicolson method. For the first four half time steps the explicit and implicit Euler methods are used. For details, see [7, 8, 3].

Resulting linear systems are solved using LU decomposition which is formed once before the first time step.

The discrete dividend is handled by performing appropriate interpolation using Matlab's built-in spline interpolation at the time of dividend payment.

3 American options

For American options, the linear complementarity problem (LCP) formulation is used and it is discretized using the operator splitting method [1]; see also [2, 3, 9]. For example, the discrete LCPs corresponding to the Crank-Nicolson steps read

$$\min\left\{\left(\mathbf{I} + \tfrac{\Delta t}{2}\mathbf{A}\right)\mathbf{u}^k - \left(\mathbf{I} - \tfrac{\Delta t}{2}\mathbf{A}\right)\mathbf{u}^{k+1} - \tfrac{\Delta t}{2}\left(\mathbf{b}^k + \mathbf{b}^{k+1}\right),\ \mathbf{u}^k - \mathbf{g}\right\} = \mathbf{0},$$

where the minimum is taken componentwise and $\mathbf{g}$ is a vector containing the grid point values of the payoff function. The operator splitting method counterpart of the above LCPs have two substeps: solve the vector $\widetilde{\mathbf{u}}^k$ from the system of linear equations

$$\left(\mathbf{I} + \tfrac{\Delta t}{2}\mathbf{A}\right)\widetilde{\mathbf{u}}^k = \left(\mathbf{I} - \tfrac{\Delta t}{2}\mathbf{A}\right)\mathbf{u}^{k+1} + \tfrac{\Delta t}{2}\left(\mathbf{b}^k + \mathbf{b}^{k+1}\right) + \Delta t\boldsymbol{\lambda}^{k+1},$$

and solve the vectors $\mathbf{u}^k$ and $\boldsymbol{\lambda}^k$ from the LCP

$$\min\left\{\boldsymbol{\lambda}^k,\ \mathbf{u}^k - \mathbf{g}\right\} = \mathbf{0}, \qquad \boldsymbol{\lambda}^k = \boldsymbol{\lambda}^{k+1} + \tfrac{1}{\Delta t}\left(\mathbf{u}^k - \widetilde{\mathbf{u}}^k\right).$$

This LCP can be solved fast componentwise. The Lagrange multiplier $\boldsymbol{\lambda}^m$ required to start time stepping is chosen to be the zero vector $\mathbf{0}$.

4 Approximations for Greeks

For computing the Greeks Delta $\Delta = u_s$, Gamma $\Gamma = u_{ss}$, and Vega $\mathcal{V} = u_\sigma$, we employ the central finite differences. After computing the options prices u,

Δ and Γ are computed by first computing the corresponding spatial derivatives using the above central finite differences at grid points and then using Matlab's built-in spline interpolation Δ and Γ are obtained at desired locations. Thus, computing Δ and Γ has essentially the same computational cost as computing the option prices. For computing $\mathcal{V}$ the option prices are computed at desired locations with the volatilities $\sigma \pm 0.5 \times 10^{-4}$ and then $\mathcal{V}$ is obtained by subtracting these prices from each other and multiplying the result by 10^4. Thus, the computation of $\mathcal{V}$ is roughly two times more expensive than computing the option prices.

References

[1] S. Ikonen and J. Toivanen, *Operator splitting methods for American option pricing*, Appl. Math. Lett., 17 (2004), pp. 809–814.

[2] ——, *Operator splitting methods for pricing American options under stochastic volatility*, Numer. Math., 113 (2009), pp. 299–324.

[3] K. in 't Hout and J. Toivanen, *Application of operator splitting methods in finance*, in Operator Splitting and Alternating Direction Methods, Springer Verlag, 2015. to appear. Available in http://arxiv.org/abs/1504.01022.

[4] O. Pironneau, *On the transport-diffusion algorithm and its applications to the Navier-Stokes equations*, Numer. Math., 38 (1981/82), pp. 309–332.

[5] R. Rannacher, *Finite element solution of diffusion problems with irregular data*, Numer. Math., 43 (1984), pp. 309–327.

[6] S. Salmi and J. Toivanen, *An iterative method for pricing American options under jump-diffusion models*, Appl. Numer. Math., 61 (2011), pp. 821–831.

[7] ——, *IMEX schemes for pricing options under jump-diffusion models*, Appl. Numer. Math., 84 (2014), pp. 33–45.

[8] S. Salmi, J. Toivanen, and L. von Sydow, *An IMEX-scheme for pricing options under stochastic volatility models with jumps*, SIAM J. Sci. Comput., 36 (2014), pp. B817–B834.

[9] J. Toivanen, *Finite difference methods for early exercise options*, in Encyclopedia of Quantitative Finance, R. Cont, ed., John Wiley & Sons, 2010.