NTNU
Kunnskap for en bedre verden

Department of Mathematics

TMA4212 - Numerical Solution of Differential Equations by Difference Methods

Project 1 - Black-Scholes Equations

Authors:
Paulius Cebatarauskas
Jasper Steinberg
Anne Asklund

February, 2022

Table of Contents

List of Figures

Workload

Jasper together with Anne showed monotonicity of the shcemes. Each group member derived one of the FDMs. Anne fixed the boundary conditions. Anne and Jasper did the FE arguments for stability and convergence. Paulius did the stability for BE and consistency. Jasper later corrected the consistency argument. Anne has created the backbone of the Jupyter Notebook with the FDMs. Paulius has chimed in and tried to help with programming and doing some of the numerical analysis. Anne programmed IMEX, while Paulius used a lot of time on 2c, to no avail. Jasper did the theory part for the nonlinear BS. Anne produced the beautiful plots. Jasper and Anne worked on the Latex file, while Paulius applied the finishing touches. Anne applied the finishing touches to the Jupyter notebook.

1 Introduction

In mathematical finance the Black-Scholes (BS) equations are used to model the price of European type options. In this project we will first investigate the 1-D linear BS equation

$$u_t - \frac{1}{2}\sigma^2 x^2 u_{xx} - rxu_x + cu = 0 \quad \text{for} \quad x \in \mathbb{R}^+, \quad t \in (0,T), \tag{1}$$

where $\sigma,\ r,\ c > 0$. Thereafter, we will treat the 1-D non-linear BS equation

$$u_t - \frac{1}{2}x^2\varphi(u_{xx})u_{xx} = 0 \quad \text{for} \quad x \in \mathbb{R}^+, \quad t \in (0,T), \tag{2}$$

where

$$\varphi(r) = \sigma_1^2 + \frac{\sigma_2^2 - \sigma_1^2}{2}\left(1 + \frac{2}{\pi}\arctan r\right). \tag{3}$$

These equations need boundary conditions (BCs) and initial conditions (ICs) in order to be well-posed. The three typical ICs from finance that we will explore are

$$u(x,0) = \max(K - x, 0) := (K - x)^+, \tag{European put}$$

$$u(x,0) = (x - K)^+ - 2(x - (K + H))^+ + (x - (K + 2H))^+, \tag{Butterfly spread}$$

$$u(x,0) = \operatorname{sgn}^+(x - K) \tag{Binary call}$$

where $K > 0$ and $H > 0$ are strike prices. As our domain is $\mathbb{R}^+$ we need BCs at $x = 0$ and $x = \infty$. Since it is impossible to have a BC at $x = \infty$ numerically, we choose an R sufficiently large such that it does not influence the behaviour of our solution near $x = 0$.The BCs we explore are $u(R,t) = u(R,0)$ for Dirichlet BCs and $u_x(R,t) = 0$ for Neumann BCs. At the boundary $x = 0$ we find our boundary condition by solving the ODE that arises from setting $x = 0$ in (1) and (2). Thus, for the linear Black-Scholes (1) and non-linear Black-Scholes (2) respectively

$$u(0,t) = u(0,0)e^{-ct} \quad \text{and} \quad u(0,t) = u(0,0). \tag{4}$$

2 Theory

2.1 Linear Black-Scholes

In the linear case, we use central differences in space to derive the following schemes for Forward Euler (FE), Backward Euler (BE) and Crank-Nicholson (CN) in time (See Appendix. A for details)

$$\vec{U}^{n+1} = (I + kA)\vec{U}^n + k\vec{b}(t_{n+1}), \tag{FE}$$

$$(I - kA)\vec{U}^{n+1} = \vec{U}^n + k\vec{b}\big(t_{n+1}\big)\big), \tag{BE}$$

$$\left(I - \frac{k}{2}A\right)\vec{U}^{n+1} = \left(I + \frac{k}{2}A\right)\vec{U}^n + \frac{k}{2}\Big(\vec{b}(t_{n+1}) + \vec{b}\big(t_{n+1}\big)\Big), \tag{CN}$$

where

$$A = \text{tridiag}\left\{\frac{\sigma^2 m^2}{2} - \frac{rm}{2}, -\sigma^2 m^2 - c, \frac{\sigma^2 m^2}{2} + \frac{rm}{2}\right\}. \tag{5}$$

We discretize with $M+1$ steps in space and $N+1$ steps in time. For Dirichlet BCs the matrices are $(M-1)\times(M-1)$, and $\vec{b}:\mathbb{R}\to\mathbb{R}^{M-1}$ is a function of time defined as $\vec{b}(t) = \left[u(0,0)e^{-ct}\left(\frac{\sigma^2}{2}-\frac{r}{2}\right), 0, \ldots, 0, u(R,0)\left(\frac{\sigma^2(M-1)^2}{2}+\frac{r}{2}(M-1)\right)\right]^T$. For Neumann BCs the matrices are of dimension $M\times M$ and A is augmented such that $A_{M,M-1} = \sigma^2 M^2$. Moreover, $\vec{b}(t) = \left[u(0,0)e^{-ct}\left(\frac{\sigma^2}{2}-\frac{r}{2}\right), 0, \ldots, 0\right]^T$.

2.1.1 Monotonicity

For a scheme to be monotone one must be able to write it in the form

$$\alpha_m^{n+1} U_m^{n+1} - \sum_{\substack{l,l'\le n+1 \\ (l',l)\neq(n+1,0)}}\sum \beta_{m+l}^{l'} U_{m+l}^{l'} + c_m^l = 0 \tag{6}$$

with $\alpha_m^n > 0$, $\beta_m^n \ge 0$, $c_m^n \in \mathbb{R}$, and $\alpha_m^n \ge \sum\sum \beta_{m+l}^{l'}$. The condition for α_m^n is satisfied for all three schemes as $1>0$ for FE, $1+k\sigma^2m^2+ck>0$ for BE and $1+\frac{k\sigma^2m^2}{2}+\frac{ck}{2}>0$ for CN. For the $\beta_{m\pm1}^{l'}$-coefficients of FE and BE, assuming that $\sigma^2 > r$,

$$\beta_{m\pm1}^{l'} = \frac{k\sigma^2m^2}{2} \pm \frac{krm}{2} \ge \frac{k\sigma^2m^2}{2} - \frac{krm}{2} \ge \frac{krm}{2}(m-1) \ge 0.$$

The same is true for CN as its $\beta_{m\pm1}^{l'}$ only differ from FE and BE by a factor of $\frac{1}{2}$. The sum of the β's in the stencil of FE is

$$\sum\sum \beta_{m+l}^{l'} = \left(\frac{k\sigma^2m^2}{2} - \frac{rmk}{2}\right) + \left(\frac{k\sigma^2m^2}{2} + \frac{rmk}{2}\right) + 1 - k\sigma^2m^2 - ck = 1 - ck \le 1 = \alpha_m^n,$$

and the sum of the β's in the stencil of BE is

$$\sum\sum \beta_{m+l}^{l'} = 1 + \left(\frac{k\sigma^2m^2}{2} - \frac{rmk}{2}\right) + \left(\frac{k\sigma^2m^2}{2} + \frac{rmk}{2}\right) = 1 + k\sigma^2m^2 \le \alpha_m^n.$$

Meanwhile, the sum of the β's in the stencil of CN is

$$\sum\sum \beta_{m+l}^{l'} = 2\left(\frac{k\sigma^2m^2}{4} - \frac{rmk}{4}\right) + 2\left(\frac{k\sigma^2m^2}{4} + \frac{rmk}{4}\right) + 1 - \frac{k\sigma^2m^2}{2} - \frac{ck}{2} = 1 + \frac{k\sigma^2m^2}{2} - \frac{ck}{2} \le \alpha_m^n.$$

Thus, for all three schemes $\alpha_m^n \ge \sum\sum \beta_{m+l}^{l'}$. Moreover, for FE $\beta_m^n = 1 - k\sigma^2m^2 - ck$ is non-negative for all m if $N \ge T(\sigma^2M^2 + c)$, which is a CFL condition that guarantees that FE is monotone. For BE, $\beta_m^n = 1$ so BE is monotone, with no CFL condition needed. For CN, $\beta_m^n = 1 - \frac{k\sigma^2m^2}{2} - \frac{ck}{2}$, which is non-negative if the CFL condition $N \ge \frac{T}{2}(\sigma^2M^2 + c)$ holds, which guarantees monotonicity for CN.

2.1.2 L^∞-stability

We will now show that BE is stable with respect to the right-hand side(RHS). Let us assume that V solves the BE scheme with a RHS, with zero initial and boundary conditions. Then our scheme is

$$\eta V_m^{n+1} + \gamma_+ V_{m-1}^{n+1} + \gamma_- V_{m+1}^{n+1} = V_m^n + kf_m^{n+1} \quad (m = 1, \ldots, M-1),$$

where

$$\eta = 1 + k\sigma^2 m^2 + kc, \qquad \gamma_\pm = \frac{\pm krm - k\sigma^2 m^2}{2}.$$

Next, we define a new grid function W as follows:

$$W_m^{n+1} = V_m^{n+1} - (V_{\max}^n + KF^{n+1}),$$

where

$$V_{\max}^n = \max_m |V_m^n|, \qquad F^{n+1} = \max_m |f_m^{n+1}|, \qquad K = \frac{k}{1+ck}.$$

Applying the BE scheme to W, rearranging and using that V satisfies the scheme yields

$$\begin{aligned}
\eta W_m^{n+1} + \gamma_+ W_{m-1}^{n+1} + \gamma_- W_{m+1}^{n+1} &= V_m^n + kf_m^{n+1} - V_{\max}^n(\eta + \gamma_+ + \gamma_-) - KF^{n+1}(\eta + \gamma_+ + \gamma_-) \\
&= \underbrace{V_m^n - V_{\max}^n(1+ck)}_{\leq 0} + kf_m^{n+1} - KF^{n+1}(1+ck) \\
&\leq F^{n+1}(k - K(1+ck)).
\end{aligned}$$

Notice that the expression above is non-positive due to the definition of K. In addition, the BE scheme is monotone as shown above. Thus, we can apply the discrete maximum principle(DMP) to W. As W^{n+1} is non-positive at the boundary, DMP tells us that the maximum of W^{n+1} is less than zero. This yields that

$$\max_m V_m^{n+1} \leq V_{\max}^n + KF^{n+1}.$$

Now applying the same argument to $(-V, -f)$ yields

$$\max_m -V_m^{n+1} \leq V_{\max}^n + KF^{n+1}.$$

This gives us that

$$V_{\max}^{n+1} \leq V_{\max}^n + KF^{n+1}.$$

Using recursion and that the initial values are zero we get that

$$V_{\max}^{n+1} \leq \frac{(n+1)k}{1+ck} \max_l F^l \leq \frac{T}{1+ck} \max_l F^l \leq T \max_{l=1,\ldots,n+1} F^l.$$

2.1.3 Consistency and L^∞ convergence

To show the consistency of (BE) we note that the truncation error is

$$\begin{aligned}
\tau_m^n &= \left((u_t)_m^n - \frac{1}{k}\nabla_t u_m^n\right) + \left(-\frac{1}{2}\sigma^2 x_m^2 (u_{xx})_m^n + \frac{1}{2}\sigma^2 x_m^2 \frac{1}{h^2}\delta_x^2 u_m^n\right) + \left(-rx_m (u_x)_m^n + rx_m \frac{1}{2h}\delta_x u_m^n\right) \\
&= \frac{k}{2}u_{tt}(\epsilon_1) + \frac{1}{2}\sigma^2 x_m^2 \left(\frac{h^2}{12}u_{xxxx}(\epsilon_2)\right) + rx_m \left(\frac{h^2}{6}u_{xxx}(\epsilon_3)\right)
\end{aligned}$$

where we inserted the truncated Taylor expansion of u_t, u_x and u_{xx}. If $u \in C^4$, we can bound the derivatives and use that $x_m \leq R$ to get

$$|\tau_m^n| \leq \frac{k}{2}\|u_{tt}\|_\infty + \frac{h^2}{24}\sigma^2 R^2 \|u_{xxxx}\|_\infty + \frac{h^2}{6} rR \|u_{xxx}\|_\infty. \tag{7}$$

Thus $\tau_m^n \to 0$ when $h, k \to 0$. Hence our method is consistent.

To prove L^∞ convergence for BE we observe that the error $e := u - U$ satisfies the BE scheme with RHS $-\tau$, with zero boundary and initial conditions. It follows from stability and consistency of BE and piecewise constant interpolation that

$$\|I_h e\|_{L^\infty} = \|I_h(u-U)\|_{L^\infty} = \|\vec{u} - \vec{U}\|_\infty \leq T \max \|\vec{\tau}\|_\infty \xrightarrow[h\to 0]{k\to 0} 0.$$

Hence our method converges in L^∞. Moreover, this gives us the error-bound

$$\|I_h e\|_{L^\infty} \leq T \max \|\vec{\tau}\|_\infty \ \leq T \left(\frac{k}{2} \|u_{tt}\|_\infty + \frac{h^2}{24} \sigma^2 R^2 \|u_{xxxx}\|_\infty + \frac{h^2}{6} rR \|u_{xxx}\| \right).$$

2.2 Non-Linear Black-Scholes

For the non-linear BS equation (2) we consider the following finite-difference schemes

$$\frac{1}{k} \nabla_t U_m^{n+1} = \frac{1}{2} x_m^2 \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^n \right) \frac{1}{h^2} \delta_x^2 U_m^{n+1}, \tag{IMEX}$$

$$\frac{1}{k} \nabla_t U_m^{n+1} = \frac{1}{2} x_m^2 \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^{n+1} \right) \frac{1}{h^2} \delta_x^2 U_m^{n+1}. \tag{BE}$$

2.2.1 Monotonicity

To show monotonicity we write out the IMEX scheme in the form of (6)

$$\left(1 + \frac{k x_m^2}{h^2} \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^n \right)\right) U_m^{n+1} - \left(\frac{k x_m^2}{2h^2} \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^n \right)\right) U_{m+1}^{n+1} - \left(\frac{k x_m^2}{2h^2} \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^n \right)\right) U_{m-1}^{n+1} - U_m^n = 0.$$

We see that all the α_m^{n+1} and $\beta_{m+l}^{l'}$ are positive, as they are composed of positive expressions. Especially note that φ is positive. Furthermore,

$$\sum_{l,l'} \beta_{m+l}^{l'} = \frac{k x_m^2}{2h^2} \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^n \right) + \frac{k x_m^2}{2h^2} \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^n \right) + 1 = \frac{k x_m^2}{h^2} \varphi \left(\frac{1}{h^2} \delta_x^2 U_m^n \right) + 1 = \alpha.$$

Thus, IMEX is monotone. The proof for BE is almost identical. Since φ is positive, changing the input of φ to $\frac{1}{h^2} \delta_x^2 U_m^{n+1}$ will not impact the argument for monotonicity.

2.2.2 Truncation error

By the definition of truncation error for IMEX we get

$$\tau_m^{n+1} = -\left(\frac{1}{k} \nabla_t u_m^{n+1} - (u_t)_m^{n+1} \right) + \frac{1}{2} x_m^2 \varphi \left(\frac{1}{h^2} \delta_x^2 u_m^n \right) \frac{1}{h^2} \delta_x^2 u_m^{n+1} - \frac{1}{2} x_m^2 \varphi \Big((u_{xx})_m^n \Big) (u_{xx})_m^{n+1}.$$

Using the identities

$$\varphi \left(\frac{\delta_x^2}{h^2} u_m^n \right) = \varphi \left((u_{xx})_m^n + O(h^2) \right) = \varphi \left((u_{xx})_m^n \right) + O(h^2) \varphi'(\epsilon)$$

$$\frac{1}{k} \nabla_t (u)_m^{n+1} = (u_t)_m^{n+1} + O(k)$$

$$\frac{1}{h^2} \delta_x^2 u_m^{n+1} = (u_{xx})_m^{n+1} + O(h^2)$$

where we use big-O notation quite freely for clearer calculations. This results in

$$\tau_m^{n+1} = O(k) + \frac{x_m^2}{2}\left(O(h^2)\varphi((u_{xx})_m^n) + O(h^2)\varphi'(\epsilon)(u_{xx})_m^{n+1} + O(h^2)O(h^2)\varphi'(\epsilon)\right).$$

Now assuming u is sufficiently regular, then all the derivatives of u are bounded. Furthermore, φ' is bounded, x_m^2 does not impact order since $x_m^2 \leq R^2$, and $\varphi(u_{xx})$ is bounded by a constant (independent of h) since φ is bounded. This shows that the truncation error for IMEX behaves like $O(k + h^2)$. The proof for BE is similar, since we only change u_{xx} to be evaluated at $n + 1$ instead of n in φ. Hence, the order of truncation error for BE is the same as for IMEX. Both schemes have the following bound for the truncation error

$$|\tau_m^{n+1}| \leq O(k) + O(h^2)\frac{R^2}{2}\left(\max\varphi(u_{xx}) + \max\varphi' \max u_{xx} + O(h^2)\max\varphi'\right).$$

The maximums being evaluated on our domain.

2.2.3 L^∞-stability

We first look at the IMEX scheme with RHS f. We define $W_m^{n+1} = V_m^{n+1} - t_{n+1}\|\vec{f}\|_\infty$, where V solves our scheme with zero boundary and initial values. Then applying our scheme on W_m^{n+1} yields

$$\frac{1}{k}\nabla_t V_m^{n+1} + \frac{1}{2}x_m^2\varphi\left(\frac{1}{h^2}\delta_x^2 V_m^n\right)\frac{1}{h^2}\delta_x^2 V_m^{n+1} - \|\vec{f}\|_\infty = f_m^{n+1} - \|\vec{f}\|_\infty \leq 0,$$

where we used that $\frac{1}{k}\nabla_t t_{n+1} = 1$ and $\delta_x^2 t_{n+1} = 0$. Since IMEX is monotone we can use DMP, which yields

$$\max_{m=1,..,M-1}\left\{V_m^{n+1} - t_{n+1}\|\vec{f}\|_\infty\right\} \leq \max_{m=0,M}\left\{V_m^{n+1} - t_{n+1}\|\vec{f}\|_\infty, 0\right\} = \max_{m=0,M}\left\{-t_{n+1}\|\vec{f}\|_\infty, 0\right\} = 0.$$

Moving $t_{n+1}\|\vec{f}\|_\infty$ to the other side and using that $t_n \leq T$ gives

$$\max_{m=1,..,M-1} V_m^{n+1} \leq t_{n+1}\|\vec{f}\|_\infty \leq T\|\vec{f}\|_\infty.$$

Now repeating the same proof for $-V$ and $-f$ gives

$$\max_{m=1,..,M-1}\left(-V_m^{n+1}\right) \leq T\|\vec{f}\|_\infty.$$

Hence, we can conclude that

$$\max_{m=1,..,M-1}|V_m^{n+1}| \leq T\|\vec{f}\|_\infty.$$

In other words, IMEX is stable. The proof for BE is similar, just with V_m^{n+1} inside of φ.

3 Results

For the error analysis of both the linear and non-linear BS, we have used an exact solution

$$u_{\text{exact}} = e^{-ct}\sin(x), \tag{8}$$

which solves the BS equation with a RHS. For the linear BS, the RHS is

$$f_1(x,t) = e^{-ct}\left(\frac{1}{2}\sigma^2 x^2 \sin(x) - rx\cos(x)\right), \tag{9}$$

while for the nonlinear BS, the RHS becomes

$$f_2(x,t) = e^{-ct}\sin(x)\left(-c + \frac{1}{2}\varphi\left(-\sin(x)e^{-ct}\right)\right). \tag{10}$$

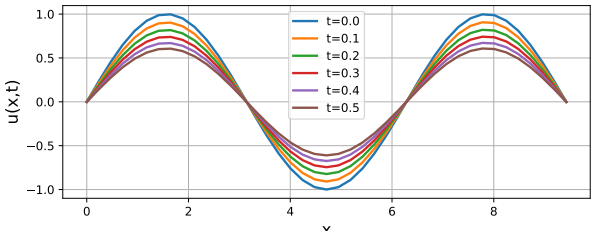

Figure 1: An approximation of (8) found by solving the non-linear BS with RHS f_2.

Figure 1 shows the solution of the nonlinear BS with f_2 as the RHS. We observe that the approximation behaves like our exact solution. At $t = 0$ we get a sine function, while for increasing t, the amplitude of the sine function gets smaller due to the exponential decay of the time variable.

3.1 Linear Black-Scholes

Figure 2 shows the solution of the linear BS for (Binary call) ICs with Dirichlet and Neumann BCs. More experiments for linear BS with other schemes and ICs can be found in Appendix B.

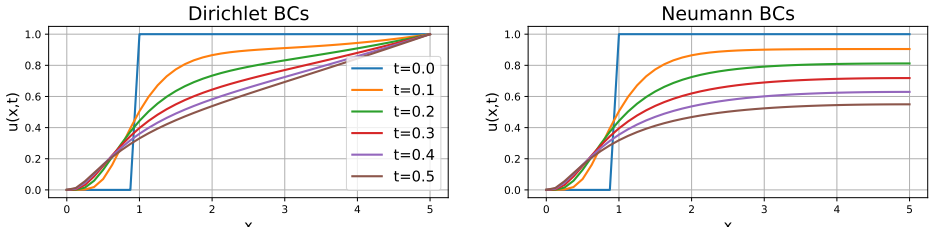

Figure 2: Solution of the linear BS with the BE scheme using binary call ICs. The plot on the left illustrates the solution with Dirichlet BCs, while the plot on the right shows the solution for Neumann BCs.

In the theory part we found a CFL condition for FE and CN. We tried testing this condition numerically for FE. Our CFL condition was $N \geq T(\sigma^2 M^2 + c)$, e.g. if this condition holds then our method should be stable. For some fixed values of T, M, σ, we get that $N \geq 400$, but according to our numerical tests this method broke down for $N = 282$, but for $N > 282$ it was still stable. So it seems that the CFL condition is sufficient to guarantee stability. But in some cases even though the CFL condition does not hold, the solution may still be stable.

Figure 3 shows that the error in space for the BE with both Dirichlet and Neumann BCs is second order, while the error in time is first order for both the BCs. This agrees with

our theoretical estimates derived in the theory part. We have confirmed that the error of CN is second order in time. For that error plot and the convergence of other schemes, see Appendix C.

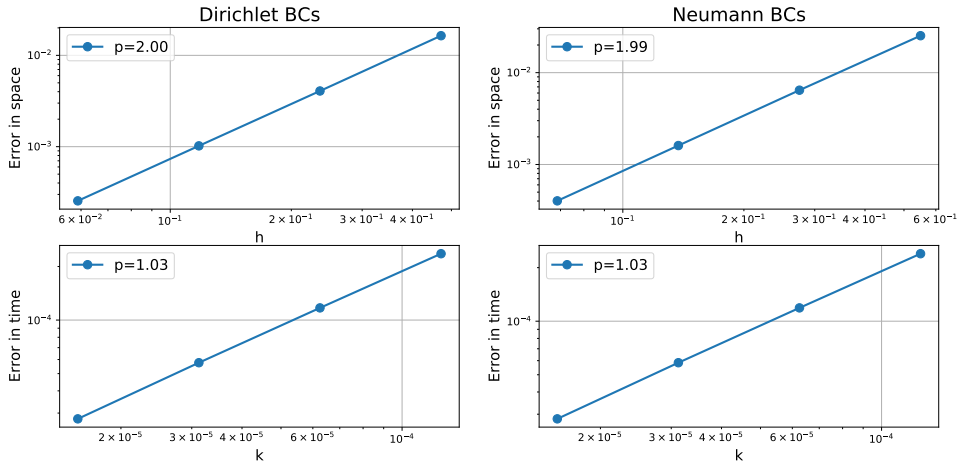

Figure 3: Errors in space and time for Dirichlet and Neumann BCs for the BE scheme.

Timing our schemes with European put ICs and $M = 40,\ N = 1700,\ T = 0.5,$ and $R = 10$ we get the computational time shown in the table below. As expected FE is significantly faster than BE and CN. This is because FE is an explicit method, so we can simply iterate to get new solutions. On the other hand, BE and CN are implicit schemes, which means that we need to solve linear systems in order to obtain new solutions. The process of solving linear systems is computationally heavier than simply iterating, hence implicit methods are generally slower than explicit methods.

Method	Computational time (s)
Forward Euler	0.0559 s
Backward Euler	0.2326 s
Crank-Nicholson	0.2368 s

In Figure 4 we computed the error in space for increasing values of R. The error was measured on a fixed interval in the max-norm between our exact solution u_{exact} and an approximate solution which was acquired using FE. The plot shows that the error decreases as R increases. In addition we see that R need not be too large before giving a decently low error.

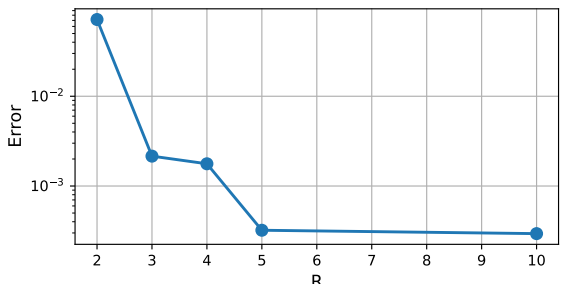

Figure 4: Errors in the interval $[0, 2]$ for changing $R \geq 2$.

3.2 Non-Linear Black-Scholes

Figure 5 shows solutions of the nonlinear BS using the IMEX scheme with Dirichlet BCs for the (European put), (Butterfly spread), and (Binary call) ICs. Note that the solutions of the linear and nonlinear BS are similar for the same initial and BCs.

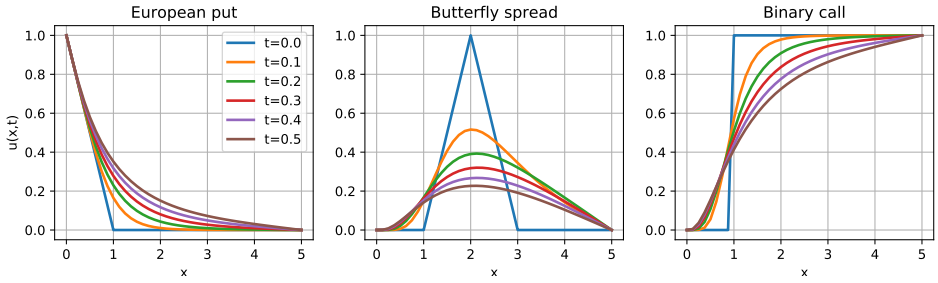

Figure 5: Experiments for IMEX for the three ICs from finance.

Figure 6 shows the error in space and time for IMEX solving the nonlinear BS with RHS (10). As expected, IMEX is second order in space and first order in time.

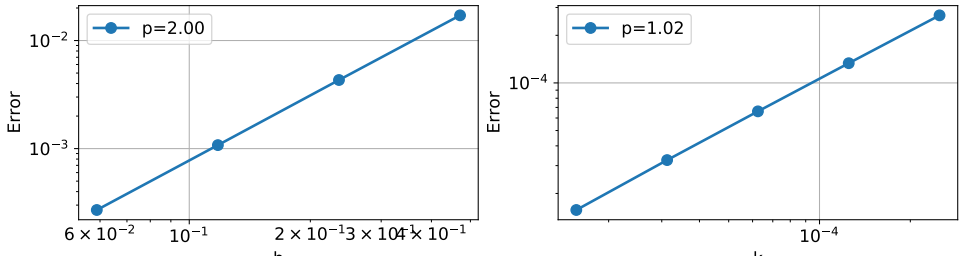

Figure 6: Error in space (left) and time (right) for IMEX with Dirichlet BCs

3.2.1 BE for nonlinear BS

We tried implementing the BE for the nonlinear BS, it did not work properly and we were not able to fix it. A more detailed explanation is in Appendix D and we would appreciate if you would read it, as we have spent a lot of time trying to make it work. Using the nonlinear BE, gives rise to a nonlinear system that needs to be solved. We tried to solve this system using Newton's method. Comparing the results with IMEX we quickly figured out that something was wrong. In particular, when we made Newton's method iterate more times the solution got worse and worse.

4 Conclusion

In this project we have investigated the Black-Scholes equations in both linear and non-linear form. For the linear Black-Scholes equation we compared the Forward Euler, Backward Euler, and Crank-Nicholson finite difference methods. For the non-linear Black-Scholes equation we compared the IMEX, and Backward Euler finite difference methods. We have shown theoretical results for monotonicity, stability, consistency, and convergence for different schemes, and done numerical tests verifying our theoretical results.

Appendix

A Derivation of Finite Differences Schemes for Different BCs

For the 1-D linear Black-Scholes equation (1), the following semi-discretization can be made with central differences in space

$$\begin{aligned}\dot{v}_m &= \frac{\sigma^2}{2h^2}x_m^2\delta_{x,h}^2 v_m + \frac{r}{2h}x_m\delta_{x,2h}v_m - cv_m \\ &= \frac{\sigma^2}{2h^2}x_m^2(v_{m-1} - 2v_m + v_{m+1}) + \frac{r}{2h}x_m(-v_{m-1} + v_{m+1}) - cv_m \\ &= v_{m-1}\left(\frac{1}{2h^2}\sigma^2 x_m^2 - \frac{rx_m}{2h}\right) - v_m\left(\frac{1}{h^2}\sigma^2 x_m^2 + c\right) + v_{m+1}\left(\frac{1}{2h^2}\sigma^2 x_m^2 + \frac{rx_m}{2h}\right)\end{aligned}$$

Letting $x_m = mh$ where $h = \frac{R}{M}$ and $m \in \{1, \ldots, M-1\}$, if $\dot{\vec{v}} = A\vec{v} + \vec{b}$

$$A = \text{tridiag}\left\{\frac{1}{2h^2}\sigma^2 m^2 h^2 - \frac{rmh}{2h}, -\frac{1}{h^2}\sigma^2 m^2 h^2 - c, \frac{1}{2h^2}\sigma^2 m^2 h^2 + \frac{rmh}{2h}\right\} \tag{11}$$

$$= \text{tridiag}\left\{\frac{1}{2}\sigma^2 m^2 - \frac{rm}{2}, -\sigma^2 m^2 - c, \frac{1}{2}\sigma^2 m^2 + \frac{rm}{2}\right\}. \tag{12}$$

For Dirichlet boundary conditions,

$$\vec{b} = \left[g_0(t)\left(\frac{1}{2}\sigma^2 - \frac{r}{2}\right), 0, \ldots, 0, g_1(t)\left(\frac{1}{2}\sigma^2(M-1)^2 + \frac{r}{2}(M-1)\right)\right]^T. \tag{13}$$

After discretizing in time such that $0 \le t \le T$ and $k = \frac{T}{N}$ where $N \in \mathbb{N}$ is the number of time steps; for the Forward Euler (FE), Backward Euler (BE), and Crank-Nicholson (CN) schemes

$$\vec{U}^{n+1} = (I + kA)\vec{U}^n + k\vec{b}(nk) \tag{FE}$$

$$(I - kA)\vec{U}^{n+1} = \vec{U}^n + k\vec{b}\big((n+1)k\big) \tag{BE}$$

$$\left(I - \frac{k}{2}A\right)\vec{U}^{n+1} = \left(I + \frac{k}{2}A\right)\vec{U}^n + \frac{k}{2}\Big(\vec{b}(nk) + \vec{b}\big((n+1)k\big)\Big) \tag{CN}$$

B Experiments

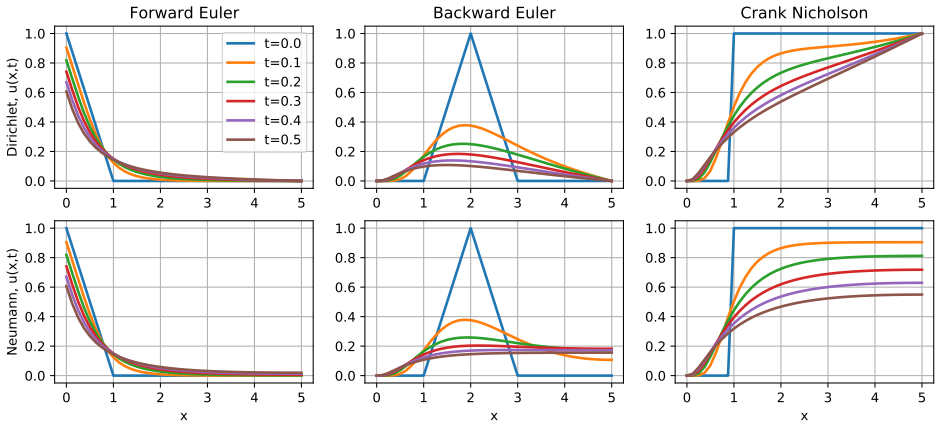

Figure 7: Experiments with the different schemes for the three ICs from finance with Dirichlet BCs (top) and Neumann BCs (bottom).

C Error Plots

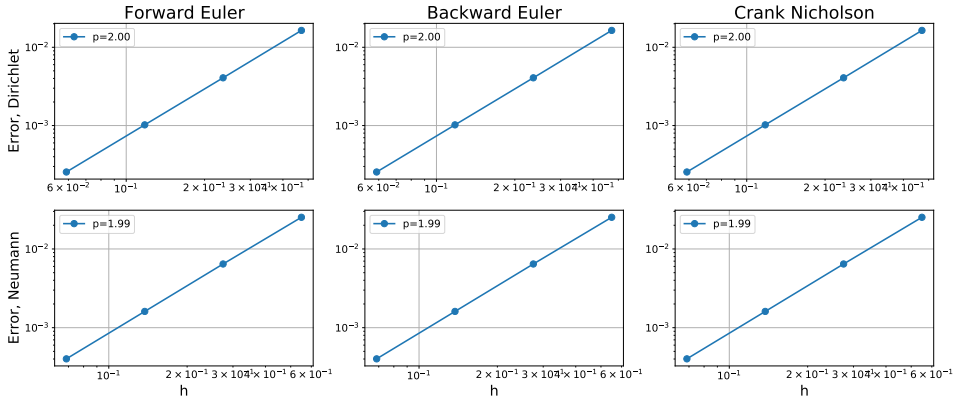

Figure 8: Errors in space for the different schemes solving linear BS with different BCs.

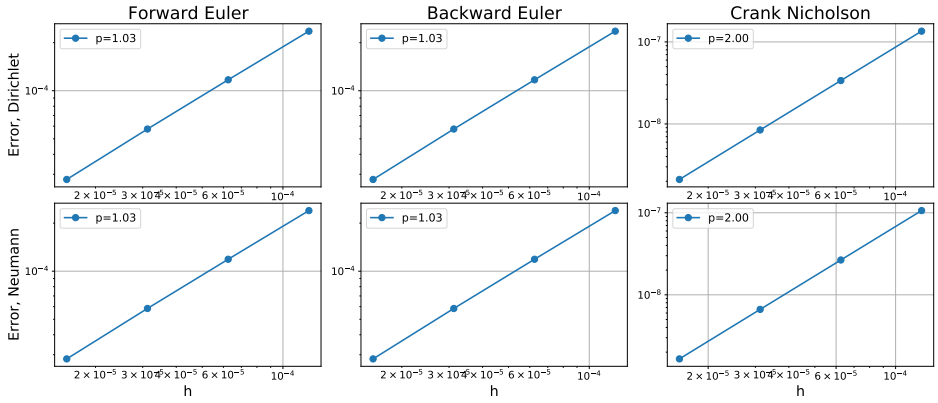

Figure 9: Errors in time for the different schemes solving linear BS with different BCs.

D BE for nonlinear BS

Our goal is to solve the nonlinear BS equation using the following FDM scheme

$$\frac{1}{k}\nabla_t U_m^{n+1} - \frac{1}{2}x_m^2\varphi\left(\frac{1}{h^2}\delta_x^2 U_m^{n+1}\right)\frac{1}{h^2}\delta_x^2 U_m^{n+1} = 0. \tag{BE}$$

We define g as

$$g(U^{n+1};U^n) = \begin{pmatrix} g_1(U^{n+1};U^n) \\ \vdots \\ g_{M-1}(U^{n+1};U^n) \end{pmatrix},$$

where

$$g_m(U^{n+1};U^n) = \frac{1}{k}\nabla_t U_m^{n+1} - \frac{1}{2}x_m^2\varphi\left(\frac{1}{h^2}\delta_x^2 U_m^{n+1}\right)\frac{1}{h^2}\delta_x^2 U_m^{n+1}.$$

Note that U^{n+1} is the variable, while U^n is a parameter which is known. In addition, $U^{n+1} = (U_0^{n+1},\dots,U_M^{n+1})$, but U_0^{n+1} and U_M^{n+1} are known. Hence, the only unknown variables in g are $U_1^{n+1},\dots,U_{M-1}^{n+1}$. We wish to use Newton's method to find the zero of g, as $g(U^{n+1};U^n)$ should equal to zero, if U^{n+1} satisfies BE for the nonlinear BS. We thus need to compute the Jacobian of g, J_g. To simplify the notation, let $U = U^{n+1}$. Then

$$J_g = \begin{pmatrix} \partial_{U_1}g_1 & \partial_{U_2}g_1 & & & & & \\ \partial_{U_1}g_2 & \partial_{U_2}g_2 & \partial_{U_3}g_2 & & & & \\ & \partial_{U_2}g_3 & \partial_{U_3}g_3 & \partial_{U_4}g_3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \partial_{U_{M-3}}g_{M-2} & \partial_{U_{M-2}}g_{M-2} & \partial_{U_{M-1}}g_{M-2} \\ & & & & \partial_{U_{M-2}}g_{M-1} & \partial_{U_{M-1}}g_{M-1} \end{pmatrix}.$$

So J_f is a tridiagonal matrix. Computing the nonzero elements yields

$$\partial_{U_{m-1}}g_m = -\frac{m^2k}{2}\left(\varphi'\left(\frac{1}{h^2}\partial_x^2 U\right)\frac{1}{h^2}\delta_x^2 U_m + \varphi\left(\frac{1}{h^2}\partial_x^2 U\right)\right) \tag{14}$$

$$\partial_{U_m}g_m = 1 + \partial_{U_{m-1}}g_m \tag{15}$$

$$\partial_{U_{+1}}g_m = \partial_{U_{m-1}}g_m. \tag{16}$$

We see that J_g is diagonally dominant. Hence, it is invertible. Now we are ready to use Newton's method

$$x_{k+1}[1:-1] = x_k[1:-1] - J_g^{-1}(x_k[1:-1])g(x_k; U^n).$$

This is a hybrid notation, half mathematical, half Python. When we write x_k we mean the whole vector, while $x_k[1:-1]$ means the whole x_k vector without the boundary values. Then our goal is that $U^{n+1} \approx x_k$ for big enough values for k. When we initiate we use $x_0 = U^n$. This seems perfectly logical, but just does not work. So we wonder whether something with our method is flawed, or if our implementation is wrong.