Notes for MATH 4665/4875/7140/7300, Fall 2019, HKBU

Location for classes: LMC 514, HKBU

Time for classes: Wednesdays 15:30–16:20; Thursdays 13:30–15:20

Instructor: Prof. Tim Sheng

FUNDAMENTALS FOR FINANCE

C. NUMERICAL SOLUTION OF PDES.

C.08 BACK TO THE PARTIAL DIFFERENTIAL EQUATIONS.

We consider the following much simplified model problem for option prices:

$$
\begin{aligned}
u_t(s,t) &= d\,u_{ss}(s,t) + c\,u_s(s,t), \quad S_{\min} < s < S_{\max},\ t > 0, && (2.6)\\
u(s,0) &= \phi(s), \quad S_{\min} < s < S_{\max}, && (2.7)\\
u(S_{\min},t) &= a, \quad t > 0, && (2.8)\\
u(S_{\max},t) &= b, \quad t > 0, && (2.9)
\end{aligned}
$$

where u is the option price function of time t and asset price s, ϕ is the initial price function given, $S_{\min}, S_{\max}$ are real values. Further, a, b, c, d are real constants and d is always assumed to be positive.

In our subsequent investigations, instead of (2.8) and (2.9), we shall often use the following "periodic boundary conditions":

$$u(S_{\min},t) = u(S_{\max},t), \quad t > 0. \tag{2.10}$$

Note that the periodic boundary condition (2.10) does not offer any particular price values on the left or right boundary of the s-domain. Instead, it implies that the values should be identical on the boundaries. That is an interesting consideration involving a geometric symmetry and possible simplicity in computations.

Of course, as what we have pointed out earlier, (2.9) may be replaced by a Neumann boundary condition

$$u_s(S_{\max},t) = c, \quad t > 0, \tag{2.11}$$

in many practical cases.

The option model (2.6)-(2.11) is, of course, a typical linear initial-boundary value problem. The derivative boundary condition (2.11) is a *Neumann boundary condition,* while boundary conditions without derivatives, such as (2.8) and (2.9), are *Dirichlet boundary conditions.*

To solve (2.6)-(2.11) via a *Method of Lines* (MOL) approach numerically, we must complete the following 3 steps.

- To obtain a semi-discretized scheme (discretization in space).
- To obtain a fully discretized scheme (discretization in time).

- To provide a necessary numerical analysis.

C.08-01 **Semi-discretizations.**

We start our investigations from examples from linear algebra.

EXAMPLE 1. Matrix vector multiplications.
Given that

$$A = \begin{bmatrix} 10 & 1 & -2 \\ 3 & 0 & -3 \\ 7 & 1 & 2 \end{bmatrix}, \; B = \begin{bmatrix} 0 & -2 & 3 \\ 1 & -1 & -1 \\ 0 & -2 & \pi \end{bmatrix}, \; v = \begin{bmatrix} 4 \\ 5 \\ 1 \end{bmatrix}.$$

Then we have

$$Av = \begin{bmatrix} 10 & 1 & -2 \\ 3 & 0 & -3 \\ 7 & 1 & 2 \end{bmatrix} \begin{bmatrix} 4 \\ 5 \\ 1 \end{bmatrix} = \begin{bmatrix} 10 \times 4 + 1 \times 5 + (-2) \times 1 \\ 3 \times 4 + 0 \times 5 + (-3) \times 1 \\ 7 \times 4 + 1 \times 5 + 2 \times 1 \end{bmatrix} = \begin{bmatrix} 43 \\ 9 \\ 35 \end{bmatrix}.$$

$$\begin{aligned} v^{\mathsf{T}} A &= \begin{bmatrix} 4 & 5 & 1 \end{bmatrix} \begin{bmatrix} 10 & 1 & -2 \\ 3 & 0 & -3 \\ 7 & 1 & 2 \end{bmatrix} = [\, 4 \times 10 + 5 \times 3 + 1 \times 7 \\ & \quad 4 \times 1 + 5 \times 0 + 1 \times 1 \quad 4 \times (-2) + 5 \times (-3) + 1 \times 2 \,] \\ &= \begin{bmatrix} 62 & 5 & -21 \end{bmatrix}. \end{aligned}$$

EXAMPLE 2. Matrix matrix multiplications.

$$\begin{aligned} AB &= \begin{bmatrix} 10 & 1 & -2 \\ 3 & 0 & -3 \\ 7 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 & -2 & 3 \\ 1 & -1 & -1 \\ 0 & -2 & \pi \end{bmatrix} \\ &= \begin{bmatrix} 10 \times 0 + 1 \times 1 + (-2) \times 0 & 10 \times (-2) + 1 \times (-1) + (-2) \times (-2) & 10 \times 3 + 1 \times (-1) + (-2) \times \pi \\ 3 \times 0 + 0 \times 1 + (-3) \times 0 & 3 \times (-2) + 0 \times (-1) + (-3) \times (-2) & 3 \times 3 + 0 \times (-1) + (-3) \times \pi \\ 7 \times 0 + 1 \times 1 + 2 \times 0 & 7 \times (-2) + 1 \times (-1) + 2 \times (-2) & 7 \times 3 + 1 \times (-1) + 2 \times \pi \end{bmatrix} \\ &= \begin{bmatrix} 1 & 17 & 29 - 2\pi \\ 0 & 0 & 9 - 3\pi \\ 1 & -19 & 20 + 2\pi \end{bmatrix}. \end{aligned}$$

We may use Matlab for matrix vector or matrix matrix multiplications. However we often end at approximations if symbolic packages are not utilized. For instance, it is easy to verify that

$$AB = \begin{bmatrix} 10 & 1 & -2 \\ 3 & 0 & -3 \\ 7 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 & -2 & 3 \\ 1 & -1 & -1 \\ 0 & -2 & \pi \end{bmatrix} = \begin{bmatrix} 1.0000 & -17.0000 & 22.7168 \\ 0 & 0 & -0.4248 \\ 1.0000 & -19.0000 & 26.2832 \end{bmatrix}$$

if default short expression is used. Further,

$$BA = \begin{bmatrix} 0 & -2 & 3 \\ 1 & -1 & -1 \\ 0 & -2 & \pi \end{bmatrix} \begin{bmatrix} 10 & 1 & -2 \\ 3 & 0 & -3 \\ 7 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 15.0000 & 3.0000 & 12.0000 \\ 0 & 0 & -1.0000 \\ 15.9911 & 3.1416 & 12.2832 \end{bmatrix}.$$

Subsequently,

$$\begin{aligned}
[A,B] &= \begin{bmatrix} -14.0000 & -20.0000 & 10.7168 \\ 0 & 0 & 0.5752 \\ -14.9911 & -22.1416 & 14.0000 \end{bmatrix} \neq \Phi, \\
\exp(A) &= 1000 \begin{bmatrix} 3.1020 & 0.2505 & -1.2615 \\ -0.2853 & -0.0255 & 0.0779 \\ 3.8967 & 0.3197 & -1.5291 \end{bmatrix}, \\
\exp(B) &= \begin{bmatrix} -2.0036 & -10.6324 & 24.3849 \\ 0.6508 & 0.4559 & -0.8710 \\ -3.1103 & -11.0728 & 26.4957 \end{bmatrix}.
\end{aligned}$$

Once a long expression is used, we see the following:

$$AB = \begin{bmatrix} 1.000000000000000 & -17.000000000000000 & 22.716814692820414 \\ 0 & 0 & -0.424777960769379 \\ 1.000000000000000 & -19.000000000000000 & 26.283185307179586 \end{bmatrix}.$$

But when a bank format is used, we can only view the following, though the values stored inside the computer have no changes.

$$AB = \begin{bmatrix} 1.00 & -17.00 & 22.72 \\ 0 & 0 & -0.42 \\ 1.00 & -19.00 & 26.28 \end{bmatrix}.$$

Now, let

$$\Omega = \{s_0, s_1, \ldots, s_N, s_{N+1}\}$$

be a uniform mesh on $[S_{\min}, S_{\max}]$, where

$$s_0 = S_{\min},\ \ s_i = s_0 + ih,\ \ s_{N+1} = S_{\max};\quad h = \frac{S_{\max} - S_{\min}}{N+1}.$$

For the simplicity in mathematical discussions, let us call s as the *spatial variable* and t as the *temporal variable.*

Let $s_i \in \Omega$ be any internal mesh point and $t > 0$. We may extend our finite difference formulae given for one variable functions and their derivatives to

1. forward difference for first order partial derivatives in space.

$$\frac{u(s_{i+1},t) - u(s_i,t)}{h} = u_s(s_i,t) + \mathcal{O}(h). \tag{2.12}$$

2. backward difference for first order partial derivatives in space.

$$\frac{u(s_i,t) - u(s_{i-1},t)}{h} = u_s(s_i,t) + \mathcal{O}(h). \tag{2.13}$$

3. central difference for first order partial derivatives in space.

$$\frac{u(s_{i+1},t) - u(s_{i-1},t)}{2h} = u_s(s_i,t) + \mathcal{O}(h^2). \tag{2.14}$$

4. central difference for second order partial derivatives in space.

$$\frac{u(s_{i+1},t) - 2u(s_i,t) + u(s_{i-1},t)}{h^2} = u_{ss}(s_i,t) + \mathcal{O}(h^2). \tag{2.15}$$

Now, let us use (2.12) for approximating the first order spatial derivative at all internal mesh points in Ω for $t > 0$. To this end, we have

$$\begin{aligned}
u_s(s_1,t) &= \frac{u(s_2,t) - u(s_1,t)}{h} + \mathcal{O}(h), \\
u_s(s_2,t) &= \frac{u(s_3,t) - u(s_2,t)}{h} + \mathcal{O}(h), \\
u_s(s_3,t) &= \frac{u(s_4,t) - u(s_3,t)}{h} + \mathcal{O}(h), \\
\cdots &\quad \cdots \\
\cdots &\quad \cdots \\
u_s(s_{N_2},t) &= \frac{u(s_{N-1},t) - u(s_{N-2},t)}{h} + \mathcal{O}(h), \\
u_s(s_{N-1},t) &= \frac{u(s_N,t) - u(s_{N-1},t)}{h} + \mathcal{O}(h).
\end{aligned}$$

Apparently, unknown option prices included in approximations are

$$\begin{aligned}
&u(s_1,t), \\
&u(s_2,t), \\
&u(s_3,t), \\
&\cdots \\
&\cdots \\
&u(s_{N-2},t), \\
&u(s_{N-1},t),
\end{aligned}$$

if prices $u(s_0,t)$ and $u(s_N,t)$ are given though boundary values, that is, via conditions (2.8), (2.9).

Now, we define our unknown vector

$$u = \begin{bmatrix} u(s_1,t) \\ u(s_2,t) \\ u(s_3,t) \\ \cdots \\ \cdots \\ u(s_{N-2},t) \\ u(s_{N-1},t) \end{bmatrix} = \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \\ \cdots \\ \cdots \\ u_{N-2}(t) \\ u_{N-1}(t) \end{bmatrix} \in \mathbb{R}^{N-1}. \tag{2.16}$$

It follows that

$$u_s = \begin{bmatrix} u_s(s_1,t) \\ u_s(s_2,t) \\ u_s(s_3,t) \\ \cdots \\ \cdots \\ u_s(s_{N-2},t) \\ u_s(s_{N-1},t) \end{bmatrix} = \begin{bmatrix} (u_s)_1(t) \\ (u_s)_2(t) \\ (u_s)_3(t) \\ \cdots \\ \cdots \\ (u_s)_{N-2}(t) \\ (u_s)_{N-1}(t) \end{bmatrix} \in \mathbb{R}^{N-1}. \tag{2.17}$$

Similarly,

$$u_{ss} = \begin{bmatrix} u_{ss}(s_1,t) \\ u_{ss}(s_2,t) \\ u_{ss}(s_3,t) \\ \cdots \\ \cdots \\ u_{ss}(s_{N-2},t) \\ u_{ss}(s_{N-1},t) \end{bmatrix} = \begin{bmatrix} (u_{ss})_1(t) \\ (u_{ss})_2(t) \\ (u_{ss})_3(t) \\ \cdots \\ \cdots \\ (u_{ss})_{N-2}(t) \\ (u_{ss})_{N-1}(t) \end{bmatrix} \in \mathbb{R}^{N-1}. \tag{2.18}$$

Let's focus at the approximation of the first spatial derivative first. To this end, a combination of (2.12), (2.16) and (2.17) yields that

$$\begin{bmatrix} (u_s)_1(t) \\ (u_s)_2(t) \\ (u_s)_3(t) \\ \cdots \\ \cdots \\ (u_s)_{N-2}(t) \\ (u_s)_{N-1}(t) \end{bmatrix} = \frac{1}{h} \begin{bmatrix} u_2(t)-u_1(t) \\ u_3(t)-u_2(t) \\ u_4(t)-u_3(t) \\ \cdots \\ \cdots \\ u_{N-1}-u_{N-2}(t) \\ u_N-u_{N-1}(t) \end{bmatrix} \in \mathbb{R}^{N-1}.$$

We further notice that

$$\frac{1}{h} \begin{bmatrix} u_2(t)-u_1(t) \\ u_3(t)-u_2(t) \\ u_4(t)-u_3(t) \\ \cdots \\ \cdots \\ u_{N-1}-u_{N-2}(t) \\ u_N-u_{N-1}(t) \end{bmatrix} = \frac{1}{h} \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & 0 & \cdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \cdots & -1 & 1 \\ 0 & 0 & \cdots & \cdots & 0 & -1 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \\ \cdots \\ \cdots \\ u_{N-2}(t) \\ u_{N-1}(t) \end{bmatrix} + \frac{1}{h} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ u_N(t) \end{bmatrix}.$$

Subsequently, we have

$$\begin{bmatrix} (u_s)_1(t) \\ (u_s)_2(t) \\ (u_s)_3(t) \\ \cdots \\ \cdots \\ (u_s)_{N-2}(t) \\ (u_s)_{N-1}(t) \end{bmatrix} = \frac{1}{h} \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & 0 & \cdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \cdots & -1 & 1 \\ 0 & 0 & \cdots & \cdots & 0 & -1 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \\ \cdots \\ \cdots \\ u_{N-2}(t) \\ u_{N-1}(t) \end{bmatrix} + \frac{1}{h} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ u_N(t) \end{bmatrix}. \tag{2.19}$$

We also note that under boundary condition (2.9), we have

$$\begin{bmatrix} (u_s)_1(t) \\ (u_s)_2(t) \\ (u_s)_3(t) \\ \cdots \\ \cdots \\ (u_s)_{N-2}(t) \\ (u_s)_{N-1}(t) \end{bmatrix} = \frac{1}{h}\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & 0 & \cdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \cdots & -1 & 1 \\ 0 & 0 & \cdots & \cdots & 0 & -1 \end{bmatrix}\begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \\ \cdots \\ \cdots \\ u_{N-2}(t) \\ u_{N-1}(t) \end{bmatrix} + \frac{1}{h}\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ b \end{bmatrix}. \tag{2.20}$$

A more compact matrix form of the above can be

$$u_s = M_1 u + \beta_2, \tag{2.21}$$

where

$$M_1 = \frac{1}{h}\text{tridiag}\,\{0, \ \ -1, \ 1\}, \ \beta_2 = \frac{1}{h}[0, 0, \ldots, 0, b]^\intercal \in \mathbb{R}^{N-1}.$$

Does (2.21) look familiar?

But what can be the semi-discretization if the periodic condition (2.10) instead of (2.9) is used?

An option is to extend the domain from $(0,1)$ to $[0,1)$ for the derivative u_s. Thereafter, a combination of (2.12), (2.16) and (2.17) yields the following.

$$\begin{bmatrix} (u_s)_0(t) \\ (u_s)_1(t) \\ (u_s)_2(t) \\ \cdots \\ \cdots \\ (u_s)_{N-2}(t) \\ (u_s)_{N-1}(t) \end{bmatrix} = \frac{1}{h}\begin{bmatrix} u_1(t) - u_0(t) \\ u_2(t) - u_1(t) \\ u_3(t) - u_2(t) \\ \cdots \\ \cdots \\ u_{N-1} - u_{N-2}(t) \\ u_N - u_{N-1}(t) \end{bmatrix} \in \mathbb{R}^{N-1}.$$

Now,

$$
\frac{1}{h}\begin{bmatrix} u_1(t)-u_0(t)\\ u_2(t)-u_1(t)\\ u_3(t)-u_2(t)\\ \cdots\\ \cdots\\ u_{N-1}-u_{N-2}(t)\\ u_N-u_{N-1}(t)\end{bmatrix}
= \frac{1}{h}\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0\\ 0 & -1 & 1 & 0 & \cdots & 0\\ 0 & 0 & -1 & 1 & 0 & \cdots\\ 0 & 0 & \cdots & \cdots & 0 & 0\\ 0 & 0 & \cdots & \cdots & -1 & 1\\ 0 & 0 & \cdots & \cdots & 0 & -1\end{bmatrix}\begin{bmatrix} u_0(t)\\ u_1(t)\\ u_2(t)\\ \cdots\\ \cdots\\ u_{N-2}(t)\\ u_{N-1}(t)\end{bmatrix}
+\frac{1}{h}\begin{bmatrix}0\\0\\0\\0\\0\\u_N(t)\end{bmatrix}
$$

$$
= \frac{1}{h}\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0\\ 0 & -1 & 1 & 0 & \cdots & 0\\ 0 & 0 & -1 & 1 & 0 & \cdots\\ 0 & 0 & \cdots & \cdots & 0 & 0\\ 0 & 0 & \cdots & \cdots & -1 & 1\\ 0 & 0 & \cdots & \cdots & 0 & -1\end{bmatrix}\begin{bmatrix} u_0(t)\\ u_1(t)\\ u_2(t)\\ \cdots\\ \cdots\\ u_{N-2}(t)\\ u_{N-1}(t)\end{bmatrix}
+\frac{1}{h}\begin{bmatrix}0\\0\\0\\0\\0\\u_0(t)\end{bmatrix}
$$

$$
= \frac{1}{h}\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0\\ 0 & -1 & 1 & 0 & \cdots & 0\\ 0 & 0 & -1 & 1 & 0 & \cdots\\ 0 & 0 & \cdots & \cdots & 0 & 0\\ 0 & 0 & \cdots & \cdots & -1 & 1\\ 1 & 0 & \cdots & \cdots & 0 & -1\end{bmatrix}\begin{bmatrix} u_0(t)\\ u_1(t)\\ u_2(t)\\ \cdots\\ \cdots\\ u_{N-2}(t)\\ u_{N-1}(t)\end{bmatrix}.
$$

Therefore we acquire the following matrix form:

$$
\tilde{u}_s = \tilde{M}_1\tilde{u}, \tag{2.22}
$$

where

$$
\tilde{M}_1 = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0\\ 0 & -1 & 1 & 0 & \cdots & 0\\ 0 & 0 & -1 & 1 & 0 & \cdots\\ 0 & 0 & \cdots & \cdots & 0 & 0\\ 0 & 0 & \cdots & \cdots & -1 & 1\\ 1 & 0 & \cdots & \cdots & 0 & -1\end{bmatrix} \in \mathbb{R}^{N\times N},\ \ \tilde{u} = \frac{1}{h}\begin{bmatrix} u_0\\ u_1\\ \vdots\\ u_{N_2}\\ u_{N-1}\end{bmatrix} \in \mathbb{R}^N.
$$

Now, $\tilde{M}_1$ is neither tridiagonal or banded! But it stays as a sparse matrix. Is that interesting?

Well, we may question that if we must extend the spatial domain from $(0,1)$ to $[0,1)$ for approximating derivative u_s under a periodic boundary condition. Do we have other options so that original good properties of M_1, such as the tridiagonality, can be preserved?

The option two is to extend $(0,1)$ to $(0,1]$ for approximating the derivative u_s under (2.10). May you work out the details (next homework)?

Now, how about if the Neumann boundary condition (2.11) is used instead of (2.9)?

A natural thought may be using a backward finite difference at the right boundary, that is,

$$u_s(S_{\max}, t) = \frac{u(S_{\max}, t) - u(S_{\max} - h, t)}{h} + \mathcal{O}(h) = c.$$

Drop the truncation error and we arrive at the approximation

$$u(S_{\max}, t) \approx u(S_{\max} - h, t) + ch$$

which means that

$$u_N(t) = u_{N-1}(t) + ch. \tag{2.23}$$

Recall (2.19). We have

$$
\begin{aligned}
\begin{bmatrix} (u_s)_1(t) \\ (u_s)_2(t) \\ (u_s)_3(t) \\ \cdots \\ \cdots \\ (u_s)_{N-2}(t) \\ (u_s)_{N-1}(t) \end{bmatrix}
&= \frac{1}{h}\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & 0 & \cdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \cdots & -1 & 1 \\ 0 & 0 & \cdots & \cdots & 0 & -1 \end{bmatrix}
\begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \\ \cdots \\ \cdots \\ u_{N-2}(t) \\ u_{N-1}(t) \end{bmatrix} \\
&\quad + \frac{1}{h}\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ u_N(t) \end{bmatrix} \\
&= \frac{1}{h}\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & 0 & \cdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \cdots & -1 & 1 \\ 0 & 0 & \cdots & \cdots & 0 & -1 \end{bmatrix}
\begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \\ \cdots \\ \cdots \\ u_{N-2}(t) \\ u_{N-1}(t) \end{bmatrix} \\
&\quad + \frac{1}{h}\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ u_{N-1}(t) + ch \end{bmatrix} \\
&= \frac{1}{h}\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & 1 & 0 & \cdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \cdots & -1 & 1 \\ 0 & 0 & \cdots & \cdots & 0 & 0 \end{bmatrix}
\begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \\ \cdots \\ \cdots \\ u_{N-2}(t) \\ u_{N-1}(t) \end{bmatrix} \\
&\quad + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ c \end{bmatrix}. \qquad (2.24)
\end{aligned}
$$

We may observe that the above coefficient matrix has now become singular! That is not very desirable. So, should we use a different finite difference formula instead of (2.23)? Your suggestions and mathematical verifications? Should any interval extension, such as those being utilized in periodic boundary condition cases, be introduced and used?

References

[1] K. in 't Hout, *Numerical Partial Differential Equations in Finance Explained,* Springer, Antwerp, Belgium, 2017 (pages 15-35).

[2] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations,* Cambridge University Press, Cambridge and New York, 2004 (pages 139-146, 251-269, 349-380).