



Article

Numerical Investigation of the Three-Dimensional HCIR Partial Differential Equation Utilizing a New Localized RBF-FD Method

Xiaoxia Ma ¹, Malik Zaka Ullah ² and Stanford Shateyi ^{3,*}

¹ Department of Basic Courses, Jiaozuo University, Jiaozuo 454000, China

² Mathematical Modeling and Applied Computation (MMAC) Research Group, Department of Mathematics, King Abdulaziz University, Jeddah 21589, Saudi Arabia

³ Department of Mathematics and Applied Mathematics, School of Mathematical and Natural Sciences, University of Venda, P. Bag X5050, Thohoyandou 0950, South Africa

* Correspondence: stanford.shateyi@univen.ac.za

Abstract: This work is concerned with the computational solution of the time-dependent 3D parabolic Heston–Cox–Ingersoll–Ross (HCIR) PDE, which is of practical importance in mathematical finance. The HCIR dynamic states that the model follows randomness for the underlying asset, the volatility and the rate of interest. Since the PDE formulation has degeneracy and non-smoothness at some area of its domain, we design a new numerical solver via semi-discretization and the radial basis function–finite difference (RBF-FD) scheme. Our scheme is built on graded meshes so as to employ the lowest possible number of discretized nodes. The stability of our solver is proven analytically. Computational testing is conducted to uphold the analytical findings in practice.

Keywords: HCIR PDE; hot area; stochastic rate of interest; fractional Black–Scholes model; non-smoothness

MSC: 65N40; 91G60



Citation: Ma, X.; Ullah, M.Z.; Shateyi, S. Numerical Investigation of the Three-Dimensional HCIR Partial Differential Equation Utilizing a New Localized RBF-FD Method. *Fractal Fract.* **2023**, *7*, 316.
<https://doi.org/10.3390/fractalfract7040316>

Academic Editors: Stanislaw Migorski and Viorel-Puiu Paun

Received: 7 March 2023

Revised: 24 March 2023

Accepted: 29 March 2023

Published: 6 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Background

One of the fundamental and important problems in mathematical finance is to price options via the partial differential equation (PDE) of Black–Scholes (BS), see e.g., [1,2]. Several extensions to this model can be found in the literature, such as the Heston model; see, for instance, [3] and the references cited therein.

The Heston model, as an extension of the Black–Scholes model [4], can be extended further if the rate of interest pursues a stochastic dynamic as well. In literature, one of the famous models that take both the volatility and the rate of interest to follow stochastic dynamics is the model of Heston–Cox–Ingersoll–Ross (HCIR); see [5,6]. The point is that such extensions based on the randomness of the volatile and the interest of the Black–Scholes model are more favorable than some generalizations based on considering fractional Brownian motion, which leads to fractional Black–Scholes PDE or similar ones in option pricing, as discussed in [7,8].

Recalling that for the non-negative stochastic variable $R(t)$, the classic process of the Cox–Ingersoll–Ross (CIR) was given in [9] as follows:

$$\begin{aligned} R(0) &= R_0, \\ dR(t) &= \sqrt{R(t)}\theta dW(t) + (-R(t) + \eta)\kappa dt, \end{aligned} \quad (1)$$

wherein θ , κ , and η , are the volatility's speed, the adjustment, and the mean, respectively. Besides, $W(t)$ is the standard Wiener process.

The model of HCIR, combining the model of Heston and (1) with the correlation constants $\rho_{12}, \rho_{13}, \rho_{23} \in [-1, 1]$, can be provided as follows [10]:

$$\begin{aligned}dR(t) &= \sigma_2 \sqrt{R(t)} dW_3(t) + (b(t) - R(t))adt, \\dV(t) &= \sigma_1 \sqrt{V(t)} dW_2(t) + (\eta - V(t))\kappa dt, \\dS(t) &= \sqrt{V(t)} S(t) dW_1(t) + R(t) S(t) dt,\end{aligned}\quad (2)$$

whereas $R(t)$ represents the process of rate of interest. Meanwhile, $\gamma > 0$, $S(t)$ and $V(t)$ are the volatility of volatility, the stock and variance processes, respectively. Additionally, $dW_1(t)dW_2(t) = \rho dt$ and $\kappa > 0$ shows the speed of adjustment of the volatility to $\eta > 0$ as its mean, while $W_1(t)$, $W_2(t)$, and $W_3(t)$ are 3 motions of Brownian. Additionally, b is a positive. Here, the parameters σ_1, σ_2, a are real positive constants. Noting that the condition of Feller for the positivity of $V(t)$ states that $2\kappa\eta > \gamma^2$, see [11].

1.2. PDE Formulation

Considering (1), the European option valuation under the HCIR model could be attained as the following PDE (forward in time) [12]:

$$\begin{aligned}\frac{\partial U(x, y, z, t)}{\partial t} &= \frac{1}{2}x^2y \frac{\partial^2 U(x, y, z, t)}{\partial x^2} + \frac{1}{2}\sigma_1^2y \frac{\partial^2 U(x, y, z, t)}{\partial y^2} + \frac{1}{2}\sigma_2^2z \frac{\partial^2 U(x, y, z, t)}{\partial z^2} \\&+ \rho_{12}\sigma_1xy \frac{\partial^2 U(x, y, z, t)}{\partial x\partial y} + \rho_{13}\sigma_2x\sqrt{yz} \frac{\partial^2 U(x, y, z, t)}{\partial x\partial z} \\&+ \rho_{23}\sigma_1\sigma_2\sqrt{vr} \frac{\partial^2 U(x, y, z, t)}{\partial y\partial z} \\&+ zx \frac{\partial U(x, y, z, t)}{\partial x} + \kappa(\eta - y) \frac{\partial U(x, y, z, t)}{\partial y} \\&+ a(b(T - t) - z) \frac{\partial U(x, y, z, t)}{\partial z} - zU(x, y, z, t),\end{aligned}\quad (3)$$

where in z, y , and x are the interest rate, the instantaneous variance, and the asset price, respectively.

For the call case of pricing, the initial condition (the so-called payoff) can be given by:

$$U(x, y, z, 0) = (x - K, 0)^+, \quad (4)$$

where K stands for the price of the strike. The initial condition when the option is of the stated type can be furnished similarly.

The HCIR 3D model is provided on $(x, y, z, t) \in [0, +\infty) \times [0, +\infty) \times [0, +\infty) \times (0, T]$. A truncation of the unbounded domain is needed to resolve the problem numerically. This is considered as follows:

$$\Omega = [0, x_{\max}] \times [0, y_{\max}] \times [0, z_{\max}], \quad (5)$$

wherein $x_{\max}, y_{\max}$, and $z_{\max}$ are real fixed values.

Generally speaking, it is hard to choose $x_{\max}, y_{\max}$, and $z_{\max}$ as the optimal value to obtain the best numerical results. The chosen values must be large enough to make the incorporation of the boundary conditions possible. We also plan to use graded meshes to concentrate further on the hot part, in which the spontaneous variance and the interest rate tend to zero (the PDE is degenerate) and the underlying asset approaches the strike price.

1.3. Motivation and the Need for Numerical Methods

The application of time-dependent PDE-based models in finance is an obvious, since in the modeling process, each asset or the involved variables with randomness mainly yield in a new spatial variable in terms of a PDE problem. Finite difference (FD) meth-

ods as pioneer schemes of different versions have been discussed for solving financial PDEs [13]. In this work, the main goal is to propose a novel computational procedure for resolving (3) via the well-resulted methodology of the radial basis function–finite difference (RBF-FD) [14–16]. In fact, in such methods, we have the accuracy of the meshfree RBF methods and the sparsity patterns of the FD schemes simultaneously [17,18]. Hence, it will be highly suitable and not yet employed for the 3D PDE (3). Numerical pieces of evidence will support the theoretical discussions.

1.4. Layout

The remainder of this manuscript is as follows. Section 2 provides the graded meshes, i.e., the non-uniform meshes that we need to discretize (3) and (4), adaptively, with a clear focus on the financially significant part, at which the underlying asset, the spontaneous variance and the rate of interest tend to the $x = K$, and $y = z = 0$, respectively. In fact, the 3D time-dependent parabolic PDE (3) demonstrates non-smoothness at $x = K$. Section 3 describes the methodology of the RBF-FD method along the weighting coefficients of a generalized RBF for our procedure. Section 4 investigates how the proposed solver can be constructed in sparse arrays and matrix forms as elegantly as possible. Ultimately, a sparse set of linear ordinary differential equations (ODEs) is derived. Section 5 is devoted to the numerical investigation of a high-order time-stepping solver with sixth-order convergence. The theoretical stability of our solver is proven. Numerical experiments are given and discussed in detail in Section 6. A short conclusion of this work is furnished in Section 7.

2. The Graded Meshes

For these proceedings, it is assumed that $\{x_i\}_{i=1}^m$ is a partition for $x \in [x_{\min}, x_{\max}]$. A well-resulted and famous non-uniform mesh can be given as follows [19,20]:

$$x_i = \Psi(v_i), \quad 1 \leq i \leq m, \quad (6)$$

where $m \gg 3$ and

$$v_{\max} = v_m > \cdots > v_2 > v_1 = v_{\min}, \quad (7)$$

stand for m equidistant nodes, while we have:

$$\begin{aligned} v_{\text{int}} &= \frac{x_{\text{right}} - x_{\text{left}}}{d_1}, \\ v_{\min} &= \sinh^{-1} \left(\frac{x_{\min} - x_{\text{left}}}{d_1} \right), \\ v_{\max} &= v_{\text{int}} + \sinh^{-1} \left(\frac{x_{\max} - x_{\text{right}}}{d_1} \right). \end{aligned} \quad (8)$$

We, too, find that $x_{\min} = 0$ and $x_{\max} = 14K$. The parameter $v_1 > 0$ controls the nodes density on $x = K$. In addition, one defines

$$\Psi(v) = \begin{cases} x_{\text{left}} + v_1 \sinh(v), & v_{\min} \leq v < 0, \\ x_{\text{left}} + v_1 v, & 0 \leq v \leq v_{\text{int}}, \\ x_{\text{right}} + v_1 \sinh(v - v_{\text{int}}), & v_{\text{int}} < v \leq v_{\max}. \end{cases} \quad (9)$$

Here, in (8), we consider $v_1 = \frac{K}{20}$, while $x_{\text{left}} = \max\{e^{-0.0025T}, 0.5\} \times K$, $x_{\text{right}} = K$, and $[x_{\text{left}}, x_{\text{right}}] \subset [0, x_{\max}]$.

The discretized nodes for the variable y , i.e., $\{y_j\}_{j=1}^n$ are constructed by:

$$y_j = \sinh(\zeta_j) v_2, \quad j = 1, 2, \dots, n, \quad (10)$$

wherein $v_2 > 0$ furnishes a density around $y = 0$. Here, we consider $v_2 = \frac{y_{\max}}{500}$, where $y_{\max} = 10$. Additionally, ς_j are the equidistant discretized nodes furnished by:

$$\Delta\varsigma = (n-1)^{-1} \sinh^{-1}\left(\frac{y_{\max}}{v_2}\right), \quad \varsigma_j = (j-1)\Delta\varsigma, \quad 1 \leq j \leq n. \quad (11)$$

Ultimately, the non-uniform nodes for the variable z are provided as:

$$z_k = v_3 \sinh(\varsigma_k), \quad 1 \leq k \leq o, \quad (12)$$

wherein $z_{\max} = 1$ and $v_3 = \frac{z_{\max}}{500}$ is a positive value. Here, we consider that $\zeta_k = (\Delta\zeta)(k-1)$, $\Delta\zeta = \frac{1}{o-1} \sinh^{-1}\left(\frac{z_{\max}}{v_3}\right)$. We note that the variables i, j , and k are local variables in this manuscript. It is also necessary to show such graded meshes in some plots. This is illustrated in Figure 1 for different values of m, n , and o .

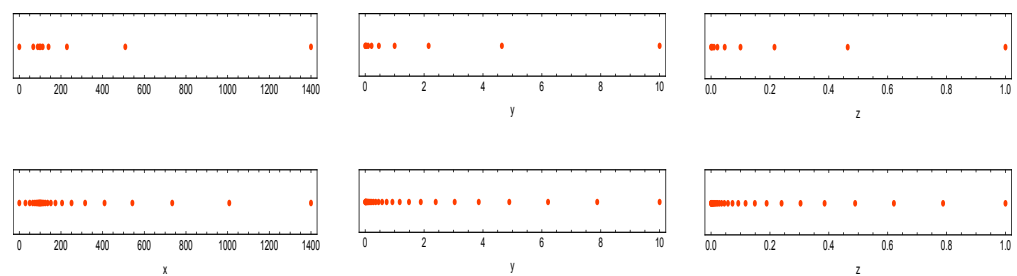


Figure 1. Distributions of the nodes along each spatial variables, for $m = n = o = 10$ at the top and $m = n = o = 30$ at the bottom.

3. The Weighting Coefficients for the RBF-FD Methodology

To compute the coefficients α_i in the RBF-FD methodology, we take into consideration that there is a linear operator, L . Then, at $\underline{y} = \underline{y}_p$, for the point locations $\underline{y}_i$, it is possible to have the following system [21]:

$$\begin{bmatrix} \phi_1(\underline{y}_1) & \phi_1(\underline{y}_2) & \cdots & \phi_1(\underline{y}_m) \\ \phi_2(\underline{y}_1) & \phi_2(\underline{y}_2) & \cdots & \phi_2(\underline{y}_m) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_m(\underline{y}_1) & \phi_m(\underline{y}_2) & \cdots & \phi_m(\underline{y}_m) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_m \end{bmatrix} = \begin{bmatrix} L\phi_1(\underline{y})|_{\underline{y}=\underline{y}_p} \\ L\phi_2(\underline{y})|_{\underline{y}=\underline{y}_p} \\ \vdots \\ L\phi_m(\underline{y})|_{\underline{y}=\underline{y}_p} \end{bmatrix}, \quad (13)$$

wherein $\underline{y}$ denotes a d-dimensional vector quantity for a set of test functions $\phi(\underline{y})$, $1 \leq k \leq m$.

The generalized multiquadric RBF (GMQ RBF) is used for numerical implementations from now on, as follows ([22], Chapter 4):

$$\phi(r_i) = (p^2 + r_i^2)^\varsigma, \quad 1 \leq i \leq m, \quad (14)$$

wherein ς stands for an appropriate constant, the shape parameter is p , and $r_i = \|\mathbf{x} - \mathbf{x}_i\|$ stands for the Euclidean distance.

It concentrates on calculating the weighting coefficients for the GMQ RBF (for the one-dimensional case without loss of generality). To calculate the coefficients of the RBF-FD methodology, we consider three non-uniform points in a stencil and write for the first derivative [23]:

$$L[\phi(x_j)] \simeq \sum_{i=1}^{\psi} \alpha_i \phi(x_i), \quad j = 1, 2, \dots, \psi, \quad (15)$$

wherein L as an operator. In fact, for $\psi = 3$, we attain: $\{x_i + \theta h, x_i, x_i - h\}$, $\theta > 0$, $h > 0$, and compute (15) as follows:

$$f'(x_i) \simeq \alpha_{i+1}f(x_{i+1}) + \alpha_i f(x_i) + \alpha_{i-1}f(x_{i-1}). \quad (16)$$

In approximating the first derivative of a sufficiently differentiable function, the analytical weights corresponding to this RBF can be determined as follows [24]:

$$\alpha_{i-1} = \frac{\theta(p^2(9 - 6\zeta) - h^2(\zeta - 1)(4(\zeta - 5)\theta - 10\zeta + 29))}{3p^2h(2\zeta - 3)(\theta + 1)}, \quad (17)$$

$$\alpha_i = \frac{(\theta - 1)(p^2(6\zeta - 9) + 4h^2(\zeta - 5)(\zeta - 1)\theta)}{3p^2h(2\zeta - 3)\theta}, \quad (18)$$

$$\alpha_{i+1} = \frac{p^2(6\zeta - 9) - h^2(\zeta - 1)\theta(2\zeta(5\theta - 2) - 29\theta + 20)}{3p^2h(2\zeta - 3)\theta(\theta + 1)}. \quad (19)$$

In a similar way, to estimate the function's second derivative, one has the following

$$f''(x_i) \simeq \sum_{j=i-1}^{i+1} \Theta_j f(x_j), \quad (20)$$

and we obtain the following weighting coefficients:

$$\Theta_{i-1} = \frac{2(p^2(6\zeta - 9) - h^2(\zeta - 1)(4(\zeta - 5)\theta^2 + (34 - 8\zeta)\theta + 10\zeta - 29))}{3p^2h^2(2\zeta - 3)(\theta + 1)}, \quad (21)$$

$$\Theta_i = \frac{2(p^2(9 - 6\zeta) + h^2(\zeta - 1)(4(\zeta - 5)\theta^2 + (25 - 2\zeta)\theta + 4(\zeta - 5)))}{3p^2h^2(2\zeta - 3)\theta}, \quad (22)$$

$$\Theta_{i+1} = \frac{2(p^2(6\zeta - 9) - h^2(\zeta - 1)(2\zeta(\theta(5\theta - 4) + 2) + \theta(34 - 29\theta) - 20))}{3p^2h^2(2\zeta - 3)\theta(\theta + 1)}. \quad (23)$$

The procedure for the discretization nodes which are on the boundaries is given along the spatial variable x , while it is similar for the other spatial variables of the PDE problem (3). The formulations (17)–(19) and (21)–(22) are fruitful from the second row to the second-to-last row, while for the first and final rows of (28) and (29), the weights are not valid on boundaries and sided estimations must be incorporated. Thus, using [25] for the stencil $\{x_1, x_2, x_3\}$, we have:

$$f'(x_1) = f[x_2, x_1] - f[x_2, x_3] + f[x_3, x_1] + \mathcal{O}\left((x_1 - x_2)^2\right), \quad (24)$$

and

$$f'(x_m) = -f[x_{m-2}, x_{m-1}] + f[x_{m-2}, x_m] + f[x_{m-1}, x_m] + \mathcal{O}\left((x_{m-1} - x_m)^2\right), \quad (25)$$

where $f[w, q] = (f(w) - f(q))(w - q)^{-1}$. Although we have used these approximation from the work [25], it is recalled that they are obtained by considering three unstructured points in a stencil and constructing the unique interpolating polynomial of the second degree. We then compute its first derivative in x_1 to obtain (24) or in x_m to obtain (25).

In a similar manner, for the four points $\{x_4, f(x_4)\}, \{x_3, f(x_3)\}, \{x_2, f(x_2)\}, \{x_1, f(x_1)\}\}$, we can obtain

$$\begin{aligned} f''(x_1) = & \frac{2(\delta x_{1,2} + \delta x_{1,3} + \delta x_{1,4})}{\delta x_{1,2}\delta x_{1,3}\delta x_{1,4}} f(x_1) + \frac{2(\delta x_{3,1} + \delta x_{4,1})}{\delta x_{1,2}\delta x_{2,3}\delta x_{2,4}} f(x_2) \\ & + \frac{2(\delta x_{2,1} + \delta x_{4,1})}{\delta x_{1,3}\delta x_{3,2}\delta x_{3,4}} f(x_3) + \frac{2(\delta x_{2,1} + \delta x_{3,1})}{\delta x_{1,4}\delta x_{4,2}\delta x_{4,3}} f(x_4) + \mathcal{O}(h^2), \end{aligned} \quad (26)$$

where $\delta x_{\zeta,q} = x_\zeta - x_q$, h is for the maximum space width along the stencil's points. Recalling that we can also obtain:

$$\begin{aligned} f''(x_m) = & \frac{2(\delta x_{m-3,m} + \delta x_{m-2,m} + \delta x_{m-1,m})}{\delta x_{m-3,m} \delta x_{m,m-2} \delta x_{m,m-1}} f(x_m) + \frac{2(\delta x_{m-3,m} + \delta x_{m-2,m})}{\delta x_{m-3,m-1} \delta x_{m-1,m-2} \delta x_{m-1,m}} f(x_{m-1}) \\ & + \frac{2(\delta x_{m-3,m} + \delta x_{m-1,m})}{\delta x_{m-3,m-2} \delta x_{m-2,m-1} \delta x_{m-2,m}} f(x_{m-2}) + \frac{2(\delta x_{m-2,m} + \delta x_{m-1,m})}{\delta x_{m-2,m-3} \delta x_{m-1,m-3} \delta x_{m,m-3}} f(x_{m-3}) \\ & + \mathcal{O}(h^2). \end{aligned} \quad (27)$$

4. Construction of Our Solver

Now, the method of lines (MOL) is employed [26,27] to semi-discretize the PDE problem. In this procedure, the spatial variables are discretized with the RBF-FD formulations we have obtained in Section 3, and we will obtain a system of ODEs. Thus, we can obtain the following differentiation matrices for the first and second derivatives of the function based on the non-uniform stencils provided in Section 2, as follows:

$$M_x = \begin{cases} \alpha_{i,j} \text{ using (17)} & i - j = 1, \\ \alpha_{i,j} \text{ using (18)} & i - j = 0, \\ \alpha_{i,j} \text{ using (19)} & j - i = 1, \\ 0 & \text{otherwise,} \end{cases} \quad (28)$$

and

$$M_{xx} = \begin{cases} \Theta_{i,j} \text{ using (21)} & i - j = 1, \\ \Theta_{i,j} \text{ using (22)} & i - j = 0, \\ \Theta_{i,j} \text{ using (23)} & j - i = 1, \\ 0 & \text{otherwise.} \end{cases} \quad (29)$$

We come up with the matrices (28) and (29) form the weights in Section 3. To be more precise, we must consider a loop on the set of discretization points. Then, each time, a stencil with three non-uniform points is considered and the corresponding weights are computed. These weights are written in the corresponding rows of the differentiation matrix. For instance, if we take the second node of the set of discretization points, then the three-point stencil consists of the first, second the third nodes. After we have obtained the weights, they will be written in the second row (corresponding to the second node of the discretization points) of the differentiation matrix.

Now, we denote the Kronecker product by $\otimes$ and consider $I = I_x \otimes I_y \otimes I_z$, as an $N \times N$ identity matrix where $N = o \times n \times m$, I_x is the $m \times m$ unit matrix for x , and a similar operation is conducted for I_y and I_z . The PDE (3) has several diffusions, reactions and advection terms. Let us consider the first term of this PDE as follows:

$$\frac{1}{2} x^2 y \frac{\partial^2 U(x, y, z, t)}{\partial x^2}, \quad (30)$$

which can be semi-discretized along space using matrix notations, as follows:

$$\frac{1}{2} \mathcal{X}^2 \mathcal{Y} (M_{xx} \otimes I_y \otimes I_z), \quad (31)$$

where the square matrices M_x , and M_{xx} are derived via the corresponding weights (28) and (29), respectively. Furthermore, $\mathcal{X}$ is a diagonal and sparse matrix, as follows:

$$\mathcal{X} = \text{diag}(x_1, x_2, \dots, x_m) \otimes I_y \otimes I_z. \quad (32)$$

Hence, the MOL procedure on (3) leads to the following matrix:

$$\begin{aligned}
 B = & \frac{1}{2}\mathcal{X}^2\mathcal{Y}(M_{xx} \otimes I_y \otimes I_z) + \frac{1}{2}\sigma_1^2\mathcal{Y}(I_x \otimes M_{yy} \otimes I_z) \\
 & + \frac{1}{2}\sigma_2^2\mathcal{Z}(I_x \otimes I_y \otimes M_{zz}) + \rho_{12}\sigma_1\mathcal{X}\mathcal{Y}(M_x \otimes M_y \otimes I_z) \\
 & + \rho_{13}\sigma_2\mathcal{X}(\mathcal{Y}\mathcal{Z})^{\frac{1}{2}}(M_x \otimes I_y \otimes M_z) + \rho_{23}\sigma_1\sigma_2(\mathcal{Y}\mathcal{Z})^{\frac{1}{2}}(I_x \otimes M_y \otimes M_z) \\
 & + \mathcal{Z}\mathcal{X}(M_x \otimes I_y \otimes I_z) + \kappa(\eta I - \mathcal{Y})(I_x \otimes M_y \otimes I_z) \\
 & + a(\beta_0 I - \mathcal{Z})(I_x \otimes I_y \otimes M_z) - zI.
 \end{aligned} \tag{33}$$

In addition, the square matrices M_y , M_z , M_{yy} , and M_{zz} , are similarly derived via the corresponding weights. Furthermore, $\mathcal{Z}$, and $\mathcal{Y}$ are diagonal and sparse matrices, as follows:

$$\mathcal{Z} = I_x \otimes I_y \otimes \text{diag}(z_1, z_2, \dots, z_n), \tag{34}$$

$$\mathcal{Y} = I_x \otimes \text{diag}(y_1, y_2, \dots, y_n) \otimes I_z, \tag{35}$$

The derivation of (33) is based on the differentiation matrices and the structure of the 3D PDE (3). The function $b(T - t)$ can be defined by:

$$b(\tau) = c_1 - c_2 \exp(-c_3\tau) \simeq \beta_0, \tag{36}$$

where c_1 , c_2 , and c_3 are fixed parameters, and $\tau = T - t$. The advantage of the approximation in (36) when c_2 and c_3 are not zero is that, with such an approximation, we can get rid of the time-dependent function b , and thus the system matrix B is constant.

Now, finally, we determine that:

$$u'(t) = Bu(t). \tag{37}$$

The boundaries for the spaces x , y and z are defined by [12]:

$$U(x, y, z, t) = x, \quad y = y_{\max}, \tag{38}$$

$$\frac{\partial U}{\partial x}(x, y, z, t) = 1, \quad x = x_{\max}, \tag{39}$$

$$U(x, y, z, t) = 0, \quad x = 0, \tag{40}$$

$$\frac{\partial U}{\partial z}(x, y, z, t) = 0, \quad z = z_{\max}. \tag{41}$$

Once $y = 0$ and $z = 0$, the problem (3) is degenerate, and boundary conditions should not be incorporated.

By imposing such boundaries, finally, a set of ODEs is deduced by:

$$u'(t) = \bar{B}u(t) = F(t, u(t)), \tag{42}$$

wherein $\bar{B}$ is the coefficient matrix including all the conditions of the boundary. The boundary conditions (38)–(41) of the PDE are non-homogeneous. However, the system ODE (42) which comes from the semi-discretization is homogeneous. This is because the boundaries are imposed by differentiating over time, so for the Neumann conditions, we write the second-order estimation along each boundary and then they are differentiated over time. The coefficients will then be replaced as new rows to the matrix B in order to obtain $\bar{B}$.

5. The Time-Stepping Solver

The solution in the closed-form for the system of stiff ODEs (42) can be given by:

$$u(t) = e^{t\bar{B}}u(0), \tag{43}$$

wherein $u(0)$ is initial condition (payoff) based on the type of the option. Finding the final approximated solution based on (43) needs the calculation of matrix exponential as a matrix function, which is a time-consuming procedure in a general setting.

Now, consider that $\mathbf{u}^l$ as an approximate to $\mathbf{u}(t_l)$. We construct the explicit time-integrating scheme as follows. Consider $k + 1$ equidistant points along time, a temporal step size $\zeta = \frac{T}{k} > 0$, $t_{l+1} = \zeta + t_l$, $\mathbf{u}^0 = u(0)$ and $0 \leq l \leq k$, then the Runge–Kutta (RK) method with s stages [28] can be written as follows:

$$\begin{aligned} g_i &= \mathbf{u}^l + \zeta \sum_{j=1}^s k_j a_{i,j}, \\ k_i &= F(\zeta c_i + t_l, g_i), \\ \mathbf{u}^{l+1} &= \mathbf{u}^l + \zeta \sum_{i=1}^s q_i k_i, \end{aligned} \quad (44)$$

wherein $F(\cdot, \cdot) = \tilde{B}u(t)$. Note that basically, we consider:

$$c_i = \sum_{j=1}^s a_{i,j}, \quad 1 \leq i \leq s. \quad (45)$$

Explicit RK solvers are a specific case when

$$\Lambda = [a_{i,j}],$$

to be lower in a strictly triangular sense. Hence, we can attain a solver with sixth-order convergence by considering [29]:

$$\Lambda = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{3}{8} & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 \\ \frac{8}{27} & \frac{2}{27} & \frac{8}{27} & 0 & 0 & 0 & 0 \\ \frac{3(3p_1-7)}{392} & \frac{(p_1-7)}{49} & \frac{-6(p_1-7)}{49} & \frac{3(p_1-21)}{392} & 0 & 0 & 0 \\ \frac{-3(17p_1+77)}{392} & \frac{(-p_1-7)}{49} & \frac{-(8p_1)}{49} & \frac{3(121p_1+21)}{1960} & \frac{(p_1+6)}{5} & 0 & 0 \\ \frac{(7p_1+22)}{12} & \frac{2}{3} & \frac{2(7p_1-5)}{9} & \frac{-7(3p_1-2)}{20} & \frac{-7(9p_1+49)}{90} & \frac{-7(p_1-7)}{18} & 0 \end{pmatrix}, \quad (46)$$

with $p_1 = 21^{1/2}$, $C = (1, 1/2, 2/3, (7 - p_1)/14, (7 + p_1)/14, 1)$, and $q = (9/180, 0, 64/180, 0, 49/180, 49/180, 9/180)$.

Here, the choice $c_1 = 0$ in (45) led to an explicit solver having sixth order of convergence along time with seven stages. This means that the total accumulated error is $\mathcal{O}(\zeta^6)$ and the local truncation error is $\mathcal{O}(\zeta^7)$, ([30], Chapter 8). It is remarked that (44) is just an application of literature in the process of resolving (3), which is considered the main result in this work.

Such a higher-order scheme is employed over time to solve (42), since the spatial order is four and there is a non-smoothness at the strike price for the payoff function. This automatically affects the accuracy of the numerical solution when marching along time. Hence, an explicit easy-to-implement solver of higher convergence rate help derive an effective numerical method.

Theorem 1. Consider that (42) reads the well-known Lipschitz condition. Hence, one has a conditional temporal-stable iterative method according to (44)–(46) for solving (42).

Proof. The existence and uniqueness can be obtained by the assumption of satisfying in the Lipschitz condition. Thus, by incorporating the solver (44)–(46) on the set of ODEs, (42) provides the relation below [31]:

$$\mathbf{u}^{t+1} = \left(I + \zeta \tilde{B} + \frac{(\zeta \tilde{B})^2}{2!} + \frac{(\zeta \tilde{B})^3}{3!} + \frac{(\zeta \tilde{B})^4}{4!} + \frac{(\zeta \tilde{B})^5}{5!} + \frac{(\zeta \tilde{B})^6}{6!} - \frac{(\zeta \tilde{B})^7}{2160} \right) \mathbf{u}^t. \quad (47)$$

Hence, the A-stability is

$$\left| 1 + \zeta \lambda_i + \frac{(\zeta \lambda_i)^2}{2} + \frac{(\zeta \lambda_i)^3}{6} + \frac{(\zeta \lambda_i)^4}{24} + \frac{(\zeta \lambda_i)^5}{120} + \frac{(\zeta \lambda_i)^6}{720} - \frac{(\zeta \lambda_i)^7}{2160} \right| \leq 1, \quad (48)$$

which is due to (47) for any λ_i as the eigenvalue of $\tilde{B}$. The condition of stability can be now expressed as follows:

$$\left| 1 + \zeta \lambda_{\max} + \frac{(\zeta \lambda_{\max})^2}{2} + \frac{(\zeta \lambda_{\max})^3}{6} + \frac{(\zeta \lambda_{\max})^4}{24} + \frac{(\zeta \lambda_{\max})^5}{120} + \frac{(\zeta \lambda_{\max})^6}{720} - \frac{(\zeta \lambda_{\max})^7}{2160} \right| \leq 1. \quad (49)$$

Considering $\Gamma = \zeta \lambda_{\max}$, the inequality (49) furnishes a non-linear scalar equation, as follows:

$$\Gamma^7 - 3\Gamma^6 - 18\Gamma^5 - 90\Gamma^4 - 360\Gamma^3 - 1080\Gamma^2 - 2160\Gamma - 4320 = 0, \quad (50)$$

with $\{-2.61051 - 1.23959i, -2.61051 + 1.23959i, -0.720604 - 2.58827i, -0.720604 + 2.58827i, 0.900758 - 2.88184i, 0.900758 + 2.88184i, 7.86071\}$ as its roots. Equivalently, one could calculate the following condition on ζ using (44)–(46) when solving (42):

$$\zeta \leq \left| \frac{7.86}{\text{Real}(\lambda_{\max})} \right|. \quad (51)$$

This inequality on the eigenvalues of $\tilde{B}$ will determine the conditional temporal stability bounds of the presented method when solving (3). Thus, it completes the proof. $\square$

Now, an inquiry might arise. The inequality (51) depends on the real part of the largest eigenvalues of matrix $\tilde{B}$. On the other hand, on the proposed graded mesh, the largest eigenvalue of $\tilde{B}$ is usually large, so the choice of the time-step ζ could be restrictive. Thus, it seems the proposed numerical method is not quite stable! In fact, the real part of the largest eigenvalue of $\tilde{B}$ might be larger due to employing the graded mesh, but because of this, we need to employ a higher-order explicit method which has large numerical stability region. To illustrate this further, in Figure 2, we have drawn the numerical stability region for different time step sizes. It is clear that when ζ is changed at a smaller size, it has a very big impact on the numerical stability region, and this upholds our choice for the time stepping solver. It is also pointed out that the application of the graded mesh leads to a lower number of discretization points to obtain the required accuracy, and this means that the resulting $\tilde{B}$ is basically of lower dimension when we want to obtain the same accuracy on uniform meshes.

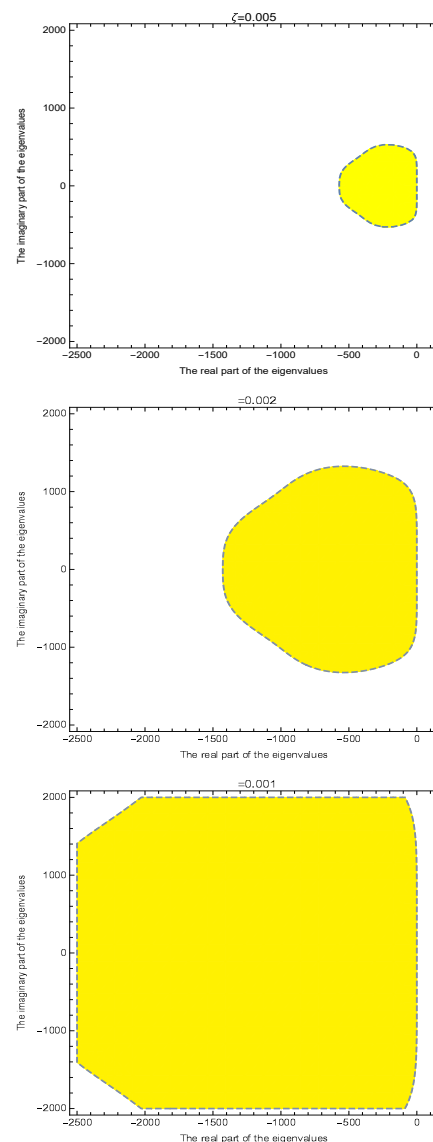


Figure 2. The numerical stability region of (44)–(46) for different values of ζ , i.e., $\zeta = 0.005$, $\zeta = 0.002$ and $\zeta = 0.001$.

6. Financial Experiments

The goal here is to calculate the efficiency of different methods to numerically resolve (3) on a same computational domain when $K = 100\%$, $T = 1$ year, $y_0 = 0.04$, $z_0 = 10\%$. The outcome of tests must be compared with the associated outcome of existing solvers to arrive at a logical and valid conclusion. The following solvers are used:

- The quadratically convergent FD method on uniform meshes and the first-order explicit Euler's scheme denoted by FDS.
- The scheme with non-equally spaced node distribution (via the Douglas time-stepping method) given in [19], (shown by THM).
- The method presented by Soleymani et al. in [25] and shown by SAM.
- The presented method of Sections 2–5 shown by RBF-FD-PM.

We used Mathematica 12.0 [32] for programming, with double-precision arithmetic. The absolute error is given by:

$$\varepsilon = \left| \frac{u_{\text{ref}} - u_{\text{num}}}{u_{\text{ref}}} \right|, \quad (52)$$

wherein u_{ref} and u_{num} are the referenced and numerical solutions, respectively. u_{ref} was obtained from the literature.

One effective procedure with which to calculate the parameter of the shape is $p = 5 \max\{\Delta x_i\}$, where $1 \leq i \leq m - 1$, and Δx_i are the increments along the variable mesh.

Three different types of parameters have been considered in Table 1. Tables 2–4 show the convergence history of different solvers in Cases I–III, while also revealing that RBF-FD-PM is better than the FDS and the THM methods.

Table 1. The values of the constants in PDE (3).

	Case I	Case II	Case III
c_3	0	0	2.10
c_2	0	0	0.014
c_1	0.05	0.055	0.034
a	0.20	0.16	0.22
σ_2	0.03	0.03	0.11
σ_1	0.04	0.90	1.00
η	0.12	0.04	0.09
ρ_{23}	0.4	0.1	−0.2
ρ_{13}	0.2	0.2	−0.5
ρ_{12}	0.6	−0.5	−0.3
κ	3.0	0.3	1.0
K	100	100	100
T	1	1	0.25

Table 2. Computational results for Case I.

Method	m	n	o	N	ζ	u	ε	Time
FDS								
	10	10	10	1000	0.001	21.187	5.75×10^{-1}	0.49
	16	12	12	2304	0.0005	5.887	5.62×10^{-1}	1.07
	30	16	16	7680	0.0001	7.542	4.38×10^{-1}	11.23
	40	20	20	16,000	0.00005	10.698	2.04×10^{-1}	52.96
	54	22	22	26,136	0.00002	10.738	2.01×10^{-1}	382.17
THM								
	10	10	10	1000	0.001	12.216	9.12×10^{-2}	0.68
	16	12	12	2304	0.0005	13.046	2.95×10^{-2}	1.81
	30	16	16	7680	0.0001	13.325	8.82×10^{-3}	15.31
	40	20	20	16,000	0.00005	13.376	4.99×10^{-3}	99.67
	54	22	22	26,136	0.00002	13.404	2.92×10^{-3}	473.52
SAM								
	10	10	10	1000	0.001	14.944	1.11×10^{-1}	0.63
	16	12	12	2304	0.0005	13.804	2.68×10^{-2}	1.68
	30	16	16	7680	0.0001	13.515	5.28×10^{-3}	21.54
	40	20	20	16,000	0.00005	13.477	2.49×10^{-3}	107.49
	54	22	22	26,136	0.00002	13.457	9.87×10^{-4}	499.67
RBF-FD-PM								
	10	10	10	1000	0.002	14.846	1.04×10^{-1}	0.62
	16	12	12	2304	0.001	13.762	2.36×10^{-2}	1.62
	30	16	16	7680	0.0004	13.499	4.09×10^{-3}	20.81
	40	20	20	16,000	0.0001	13.471	2.00×10^{-3}	101.19
	54	22	22	26,136	0.00004	13.455	8.18×10^{-4}	477.28

Table 3. Computational results and comparisons for the Case II.

Method	m	n	o	N	ζ	u	ε	Time
FDS								
	8	8	8	512	0.002	47.829	5.99×10^0	0.27
	14	10	10	1400	0.0005	5.469	2.00×10^{-1}	0.71
	20	14	12	3360	0.00025	14.786	1.16×10^{-1}	2.13
	24	16	14	5376	0.0001	12.960	8.95×10^{-1}	6.91
	32	18	18	10,368	0.00005	8.599	2.57×10^{-1}	28.64
	45	24	20	19,800	0.000025	6.456	5.59×10^{-2}	178.43
THM								
	8	8	8	512	0.002	5.010	2.67×10^{-1}	0.30
	14	10	10	1400	0.0005	6.440	5.83×10^{-2}	0.69
	20	14	12	3360	0.00025	6.672	2.43×10^{-2}	1.97
	24	16	14	5376	0.0001	6.729	1.59×10^{-2}	6.58
	32	18	18	10,368	0.00005	6.797	6.02×10^{-3}	32.44
	45	24	20	19,800	0.000025	6.830	1.28×10^{-3}	180.59
SAM								
	8	8	8	512	0.002	5.794	1.52×10^{-1}	0.35
	14	10	10	1400	0.0005	6.628	3.07×10^{-2}	0.96
	20	14	12	3360	0.00025	6.759	1.16×10^{-2}	3.11
	24	16	14	5376	0.0001	6.776	9.07×10^{-3}	13.37
	32	18	18	10,368	0.00005	6.809	4.30×10^{-3}	57.16
	45	24	20	19,800	0.000025	6.833	8.52×10^{-4}	232.76
RBF-FD-PM								
	8	8	8	512	0.004	5.861	1.43×10^{-1}	0.33
	14	10	10	1400	0.001	6.501	4.94×10^{-2}	0.90
	20	14	12	3360	0.0004	6.760	1.15×10^{-2}	3.03
	24	16	14	5376	0.0002	6.786	7.74×10^{-3}	12.69
	32	18	18	10,368	0.0001	6.826	1.90×10^{-3}	55.84
	45	24	20	19,800	0.00004	6.834	7.31×10^{-4}	224.21

The reference solutions for the Cases I-II-III are 13.444, 6.839 and 3.890, respectively [25]. A numerical solution is presented for Case II in Figure 3, showing the stability of the numerical solution using RBF-FD-PM.

It can be inferred from the findings of performance outcomes, displayed in Tables 2–4, that the presented solver is numerically more effective compared to the available methods. Similar computational tests, conducted on different parameter settings, largely uphold the above conclusions.

We note that implicit schemes might be attractive for solving the system of ODEs (42), since they mostly do not rely on the selection of the step size. However, this topics needs further investigation, since the problem with implicit solvers is that a large set of non-linear algebraic equations must be solved via a Newton-type method, which increases the computational elapsed time.

Table 4. Computational results and comparisons for Case III.

Method	m	n	o	N	ζ	u	ε	Time
FDS								
	8	8	8	512	0.002	0.857	7.79×10^{-1}	0.10
	14	10	10	1400	0.0005	2.124	4.53×10^{-1}	0.27
	20	14	12	3360	0.00025	2.976	2.34×10^{-1}	1.08
	24	16	14	5376	0.0001	3.214	1.73×10^{-1}	3.54

Table 4. Cont.

Method	m	n	o	N	ζ	u	ε	Time
THM								
	8	8	8	512	0.002	3.210	1.74×10^{-1}	0.30
	14	10	10	1400	0.0005	3.528	9.30×10^{-2}	1.01
	20	14	12	3360	0.00025	3.604	7.35×10^{-2}	1.51
	24	16	14	5376	0.0001	3.694	5.03×10^{-2}	4.92
SAM								
	8	8	8	512	0.002	3.329	1.44×10^{-1}	0.26
	14	10	10	1400	0.0005	3.539	9.02×10^{-2}	0.87
	20	14	12	3360	0.00025	3.719	4.39×10^{-2}	1.48
	24	16	14	5376	0.0001	3.924	8.74×10^{-3}	4.76
RBF-FD-PM								
	8	8	8	512	0.0025	3.413	1.22×10^{-1}	0.23
	14	10	10	1400	0.000625	3.610	7.19×10^{-2}	0.78
	20	14	12	3360	0.0004	3.816	1.90×10^{-2}	1.39
	24	16	14	5376	0.00025	3.918	7.19×10^{-3}	4.67

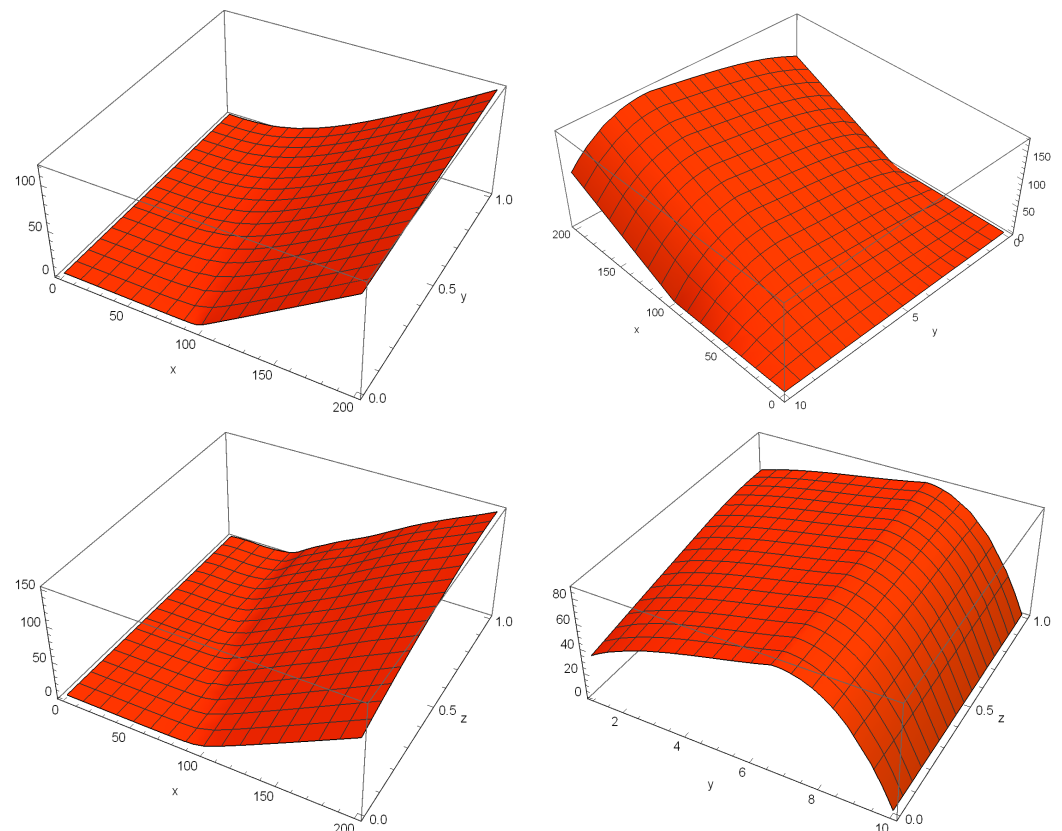


Figure 3. Numerical solution of the HCIR PDE in Case II. Top-left: for $u(x, y, 0.024, 1)$ on $0 \leq x \leq 200$, $0 \leq y \leq 1$, Top-right: for $u(x, y, 0.024, 1)$ on $0 \leq x \leq 200$, $0 \leq y \leq 10$, Bottom-left: for $u(x, 0.04, z, 1)$ on $0 \leq x \leq 200$, $0 \leq z \leq 1$ and Bottom-right: for $u(100, y, z, 1)$ on $1 \leq y \leq 10$, $0 \leq z \leq 1$.

7. Conclusions

To computationally and efficiently solve the PDE problem (3), this work began by considering the RBF-FD formulations corresponding to the GMQ RBF in Section 3 based on the graded meshes discussed in Section 2. Then, the MOL for (3) was performed using sparse arrays in order to maintain and develop the sparsity as the dimension of the problem grew. Then in Section 4, a sixth-order explicit solver from the RK family of methods was

introduced. It was shown that the contributed scheme is fast and stable, with error bounds $\mathcal{O}(\zeta^6)$. The numerical results given in Section 6 overwhelmingly upheld the theoretical discussions of the paper. Hence, the proposed solver can be used to solve the important 3D time-dependent PDE (3) in practice.

Author Contributions: Conceptualization, X.M.; formal analysis, X.M. and S.S.; funding acquisition, M.Z.U.; investigation, X.M., M.Z.U. and S.S.; methodology, X.M. and S.S.; supervision, M.Z.U.; validation, X.M., M.Z.U. and S.S.; writing—original draft, X.M. and S.S.; writing—review and editing, X.M., M.Z.U. and S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: For the data availability statement, we state that data sharing is not applicable to this article, as no new data were created in this study.

Acknowledgments: This research work was funded by Institutional Fund Projects under grant no. (IFPIP: 1417-130-1443). The authors gratefully acknowledge technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no known competing financial interest or personal relationships that could have appeared to influence the work reported in this paper.

References

- Adhikari, R. Foundations of Computational Finance. *Math. J.* **2020**, *22*, 1–59. [\[CrossRef\]](#)
- Itkin, A. *Pricing Derivatives Under Lévy Models: Modern Finite-Difference and Pseudo-Differential Operators Approach*; Birkhäuser: Basel, Switzerland, 2017.
- Guo, S.; Grzelak, L.; Oosterlee, C.W. Analysis of an affine version of the Heston-Hull-White option pricing partial differential equation. *Appl. Numer. Math.* **2013**, *72*, 143–159. [\[CrossRef\]](#)
- Heston, S.L. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Rev. Finan. Stud.* **1993**, *6*, 327–343. [\[CrossRef\]](#)
- Cao, J.; Lian, G.; Roslan, T.R.N. Pricing variance swaps under stochastic volatility and stochastic interest rate. *Appl. Math. Comput.* **2016**, *277*, 72–81. [\[CrossRef\]](#)
- Hull, J.; White, A. Using Hull-White interest rate trees. *J. Deriv.* **1996**, *4*, 26–36. [\[CrossRef\]](#)
- Ampun, S.; Sawangtong, P.; Sawangtong, W. An analysis of the fractional-order option pricing problem for two assets by the generalized Laplace variational iteration approach. *Fractal Fract.* **2022**, *6*, 667. [\[CrossRef\]](#)
- Zhang, X.; Yang, J.; Zhao, Y. Numerical solution of time fractional Black-Scholes model based on Legendre wavelet neural network with extreme learning machine. *Fractal Fract.* **2022**, *6*, 401. [\[CrossRef\]](#)
- Cox, J.C.; Ingersoll, J.E.; Ross, S.A. A theory of the term structure of interest rates. *Econometrica* **1985**, *53*, 385–407. [\[CrossRef\]](#)
- Grzelak, L.A.; Oosterlee, C.W. On the Heston model with stochastic interest rates. *SIAM J. Finan. Math.* **2011**, *2*, 255–286. [\[CrossRef\]](#)
- Djeutcha, E.; Fono, L.A. Pricing for options in a Hull-White-Vasicek volatility and interest rate model. *Appl. Math. Sci.* **2021**, *15*, 377–384. [\[CrossRef\]](#)
- Haentjens, T. Efficient and stable numerical solution of the Heston-Cox-Ingersoll-Ross partial differential equation by alternating direction implicit finite difference schemes. *Int. J. Comput. Math.* **2013**, *90*, 2409–2430. [\[CrossRef\]](#)
- Fornberg, B. *A Practical Guide to Pseudospectral Methods*; Cambridge University Press: Cambridge, UK, 1996.
- Kadalbajoo, M.K.; Kumar, A.; Tripathi, L.P. Radial-basis-function-based finite difference operator splitting method for pricing American options. *Int. J. Comput. Math.* **2018**, *95*, 2343–2359. [\[CrossRef\]](#)
- Vanani, S.K.; Hafshejani, J.S.; Soleymani, F.; Khan, M. Radial basis collocation method for the solution of differential-difference equations. *World Appl. Sci. J.* **2011**, *13*, 2526–2530.
- Farahmand, G.; Lotfi, T.; Ullah, M.Z.; Shateyi, S. Finding an efficient computational solution for the Bates partial integro-differential equation utilizing the RBF-FD scheme. *Mathematics* **2023**, *11*, 1123. [\[CrossRef\]](#)
- Liu, T.; Ullah, M.Z.; Shateyi, S.; Liu, C.; Yang, Y. An efficient localized RBF-FD method to simulate the Heston-Hull-White PDE in finance. *Mathematics* **2023**, *11*, 833. [\[CrossRef\]](#)
- Milovanović, S.; von Sydow, L. A high order method for pricing of financial derivatives using radial basis function generated finite differences. *Math. Comput. Simul.* **2020**, *174*, 205–217. [\[CrossRef\]](#)
- Haentjens, T.; In't Hout, K.J. Alternating direction implicit finite difference schemes for the Heston-Hull-White partial differential equation. *J. Comput. Financ.* **2012**, *16*, 83–110. [\[CrossRef\]](#)

20. Kluge, T. Pricing Derivatives in Stochastic Volatility Models Using the Finite Difference Method. Ph.D. Thesis, Technische Universität Chemnitz, Chemnitz, Germany, 2002.
21. Milovanović, S.; von Sydow, L. Radial basis function generated finite differences for option pricing problems. *Comput. Math. Appl.* **2018**, *75*, 1462–1481.
22. Fasshauer, G.E. *Meshfree Approximation Methods with MATLAB*; World Scientific Publishing: Singapore, 2007.
23. Wright, G.B.; Fornberg, B. Scattered node compact finite difference-type formulas generated from radial basis functions. *J. Comput. Phys.* **2006**, *212*, 99–123. [[CrossRef](#)]
24. Soleymani, F.; Zhu, S. RBF-FD solution for a financial partial-integro differential equation utilizing the generalized multiquadric function. *Comput. Math. Appl.* **2021**, *82*, 161–178. [[CrossRef](#)]
25. Soleymani, F.; Akgül, A.; Karatas Akgül, E. On an improved computational solution for the 3D HCIR PDE in finance. *Analele Stiintifice Ale Univ. Ovidius Constanta-Ser. Mat.* **2019**, *27*, 207–230. [[CrossRef](#)]
26. Knapp, R. A method of lines framework in Mathematica. *J. Numer. Anal. Indust. Appl. Math.* **2008**, *3*, 43–59.
27. Meyer, G.H. *The Time-Discrete Method of Lines for Options and Bonds, A PDE Approach*; World Scientific Publishing: Singapore, 2015.
28. Sofroniou, M.; Knapp, R. *Advanced Numerical Differential Equation Solving in Mathematica*; Wolfram Mathematica, Tutorial Collection; Wolfram: Champaign, IL, USA, 2008.
29. Luther, H.A. An explicit sixth-order Runge-Kutta formula. *Math Comput.* **1967**, *22*, 434–436. [[CrossRef](#)]
30. Keskin, A.Ü. *Ordinary Differential Equations for Engineers*; Springer International Publishing: Cham, Switzerland, 2019.
31. Shampine, L.F. *Numerical Solution of Ordinary Differential Equations*; Chapman and Hall: London, UK, 1994.
32. Wellin, P.R.; Gaylord, R.J.; Kamin, S.N. *An Introduction to Programming with Mathematica*; Cambridge University Press: Cambridge, UK, 2005.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.