

Article

An Efficient Localized RBF-FD Method to Simulate the Heston–Hull–White PDE in Finance

Tao Liu ¹, Malik Zaka Ullah ², Stanford Shateyi ^{3,*}, Chao Liu ¹ and Yanxiong Yang ⁴¹ School of Mathematics and Statistics, Northeastern University at Qinhuangdao, Qinhuangdao 066000, China² Mathematical Modeling and Applied Computation (MMAC) Research Group, Department of Mathematics, King Abdulaziz University, Jeddah 21589, Saudi Arabia³ Department of Mathematics and Applied Mathematics, School of Mathematical and Natural Sciences, University of Venda, P. Bag X5050, Thohoyandou 0950, South Africa⁴ Eighth Geological Brigade of Hebei Bureau of Geology and Mineral Resources Exploration, Qinhuangdao 066000, China

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

Abstract: The Heston–Hull–White three-dimensional time-dependent partial differential equation (PDE) is one of the important models in mathematical finance, at which not only the volatility is modeled based on a stochastic process but also the rate of interest is assumed to follow a stochastic dynamic. Hence, an efficient method is derived in this paper based on the methodology of the localized radial basis function generated finite difference (RBF-FD) scheme. The proposed solver uses the RBF-FD approximations on graded meshes along all three spatial variables and a high order time-stepping scheme. Stability is also studied in detail to show under what conditions the proposed method is stable. Computational simulations are given to support the theoretical discussions.

Keywords: graded meshes; stochastic rate of interest; stable; pricing options; Heston–Hull–White PDE

MSC: 91G60; 65N40; 91B25



Citation: 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. <https://doi.org/10.3390/math11040833>

Academic Editors: Pavol Durana and Katarina Valaskova

Received: 18 November 2022

Revised: 19 January 2023

Accepted: 31 January 2023

Published: 7 February 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

It is well-known that under the risk-neutral measure for pricing, the classic model of Heston [1] as a set of stochastic differential equations (SDEs) can show the stock option value by considering that the volatility is no longer constant. In fact, it follows the dynamic below with the state vector $X(t) = (S(t), V(t))^*$:

$$\begin{aligned}dV(t) &= \gamma \sqrt{V(t)} d\mathcal{W}_2(t) + \kappa(\eta - V(t))dt, & V(t) > 0, \\dS(t) &= \sqrt{V(t)} S(t) d\mathcal{W}_1(t) + rS(t)dt, & S(t) > 0,\end{aligned}\quad (1)$$

wherein the variance and the stock processes are $V(t)$, $S(t)$, the volatility of volatility is $\gamma > 0$, the riskless fixed interest rate is $r > 0$ and $d\mathcal{W}_1(t)d\mathcal{W}_2(t) = \rho dt$. Here $\kappa > 0$ determines the adjustment's speed of the volatility to the mean $\eta > 0$ and $\mathcal{W}_1(t)$, $\mathcal{W}_2(t)$ are two kinds of Brownian motion. In order to $V(t) > 0$, the condition of Feller should be satisfied as $2\kappa\eta > \gamma^2$.

The model (1), which is also known as a generalization of the model of Black–Scholes [2], could be extended further if the interest rate follows a special dynamic as well. There are two well-known models in the literature that take both the interest rate and the volatility to be as stochastic processes: that is, the Heston–Cox–Ingersoll–Ross (HCIR) [3] and the Heston–Hull–White (HHW) models [4]. For more information, see [5,6].

The model of HHW with the correlation parameters $\rho_{12}, \rho_{13}, \rho_{23} \in [-1, 1]$ could be given in what follows [4]:

$$\begin{aligned} dR(t) &= a(b(t) - R(t))dt + \sigma_2 d\mathcal{W}_3(t), \\ dV(t) &= \kappa(\eta - V(t))dt + \sigma_1 \sqrt{V(t)} d\mathcal{W}_2(t), \\ dS(t) &= R(t)S(t)dt + \sqrt{V(t)}S(t)d\mathcal{W}_1(t), \end{aligned} \tag{2}$$

where $R(t)$ represents the process of the rate of interest at time $T \geq t > 0$. In (2), the positive function b is given, and $\mathcal{W}_3(t)$ is a new kind of Brownian motion. Here, the parameters $\kappa, \eta, \sigma_1, \sigma_2, a$ are real positive constants.

It is necessary to explain the usefulness of the pricing model using the Heston model combined with the Hull–White model for the market and investors. In fact, as long as the interest rate follows a stochastic dynamic, then pricing would result in more accurate simulation results in equity markets at which the interest rates are no longer constant [7,8]. For further related discussions, one may refer to [8] and the references cited therein.

By employing the dynamics in (2), the value of a European option could be obtained in the form of the following PDE [9]:

$$\begin{aligned} \frac{\partial U(s, v, r, t)}{\partial t} &= \frac{1}{2} s^2 v \frac{\partial^2 U(s, v, r, t)}{\partial s^2} + \frac{1}{2} \sigma_1^2 v \frac{\partial^2 U(s, v, r, t)}{\partial v^2} + \frac{1}{2} \sigma_2^2 \frac{\partial^2 U(s, v, r, t)}{\partial r^2} \\ &+ \rho_{12} \sigma_1 s v \frac{\partial^2 U(s, v, r, t)}{\partial s \partial v} + \rho_{13} \sigma_2 s \sqrt{v} \frac{\partial^2 U(s, v, r, t)}{\partial s \partial r} + \rho_{23} \sigma_1 \sigma_2 \sqrt{v} \frac{\partial^2 U(s, v, r, t)}{\partial v \partial r} \\ &+ r s \frac{\partial U(s, v, r, t)}{\partial s} + \kappa(\eta - v) \frac{\partial U(s, v, r, t)}{\partial v} + a(b(T - t) - r) \frac{\partial U(s, v, r, t)}{\partial r} \\ &- rU(s, v, r, t), \end{aligned} \tag{3}$$

where s, v , and r are the asset price, instantaneous variance, and the rate of interest, respectively. The initial condition, which is known as the payoff function for the call case, could be written as follows:

$$U(s, v, r, 0) = (s - K)^+, \tag{4}$$

wherein K is the price of strike.

The conditions of the boundary along the spatial variables s, v, r can be written as [10]

$$U_s(s, v, r, t) = 1, \quad s = s_{\max}, \tag{5}$$

$$U(s, v, r, t) = 0, \quad s = 0, \tag{6}$$

$$U(s, v, r, t) = s, \quad v = v_{\max}, \tag{7}$$

$$U_r(s, v, r, t) = 0, \quad r = -r_{\max}, \tag{8}$$

$$U_r(s, v, r, t) = 0, \quad r = r_{\max}. \tag{9}$$

Note that for the case when $v = 0$, the PDE (3) is degenerate, and no boundaries must be imposed [11].

It is worthwhile to note that to achieve calibration targets in the market, option pricing is an important tool. To price derivative securities, different types of significant models are taken into account and investigated, leading to a set of numerical solvers. The HHW PDE (3) for the general case does not have a closed-form solution, which turns our attention to efficient numerical methods automatically.

A pioneering numerical method for pricing options is the finite difference (FD) method [12], which is mainly based on employing uniform discretization nodes, laying a uniform grid on the independent variables of the PDE problem and then transforming the continuous problem into a set of discrete ones.

Another well-performing approach that inherits both from the FD and the radial basis function (RBF) schemes is the local RBF-FD scheme [13–15]. For further reading, one may refer to [16–18].

Here, the major target is to present a new numerical solver for the simulation of (3) based on the RBF-FD methodology, which results in block sparse matrices. This is mainly because (3) is a 3D problem with variable coefficients at which there are three mixed derivatives. Thus, the numerical methods must be designed for this purpose carefully. The RBF-FD formulations are written so as to be applied on graded meshes in which there is a clear concentration on the hot area.

The rest of this article is structured as follows. First, in Section 2, we review some of the efficient ways of producing graded meshes along the spatial variables to focus more on the hot area of the 3D PDE problem (3) [19]. Then, in Section 3, the RBF-FD weighting coefficients for a new RBF are constructed and proven to converge on graded meshes. The estimations can be considered and simplified on uniform meshes as well. Section 4 proposes our RBF-FD solver with attention on the financially important area, in which the initial condition of the PDE problem has non-smoothness. The stability of the solver is given in detail in Section 5 to show how and under what conditions the proposed solver is stable for solving (3). Then, Section 6 presents a discussion of the usefulness and applicability of the proposed formulas. Some numerical tests are furnished along with numerical simulations. Ultimately, a conclusion of the work is given in Section 7.

2. Mesh Generation

The HHW 3D model is defined on $(s, v, r, t) \in [0, +\infty) \times [0, +\infty) \times (-\infty, +\infty) \times (0, T]$. The merit of the HHW model is that it can take negative rates of interest, unlike the HCIR model at which the interest rate must always be positive [20].

It is now obvious that to price under the HHW model, we first need to localize the domain as follows:

$$(s, v, r, t) \in [0, s_{\max}] \times [0, v_{\max}] \times [-r_{\max}, r_{\max}] \times (0, T], \tag{10}$$

where $s_{\max}, v_{\max}, r_{\max}$ are real positive constants.

It is not that clear what to choose exactly for the s_{\max}, v_{\max} , and r_{\max} in order to get the optimal numerical domain. However, to impose the boundary conditions as well as to reduce the error caused by truncating the domain, it would be better to choose them as far as possible and employ graded meshes at which the focus is on the important area of the problem. Here, the financially important area of the problem (3) is the region at which the underlying asset price tends to the price of the strike, the spontaneous variance tends to zero, and the interest rate tends to a special small value such as the fixed risk-free interest rate.

To proceed, we assume that $\{s_i\}_{i=1}^m$ is a partition for $s \in [s_{\min}, s_{\max}]$. Now, a well-performing and popular graded mesh could be defined as follows [21] ($i = 1, 2, \dots, m$):

$$s_i = \Psi(\omega_i), \tag{11}$$

wherein $m \gg 3$ and

$$\omega_{\max} = \omega_m > \dots > \omega_2 > \omega_1 = \omega_{\min}, \tag{12}$$

are m uniform points, while we also have

$$\begin{aligned} \omega_{\min} &= \sinh^{-1}\left(\frac{s_{\min} - s_{\text{left}}}{d_1}\right), \\ \omega_{\text{int}} &= \frac{s_{\text{right}} - s_{\text{left}}}{d_1}, \\ \omega_{\max} &= \omega_{\text{int}} + \sinh^{-1}\left(\frac{s_{\max} - s_{\text{right}}}{d_1}\right). \end{aligned} \tag{13}$$

We also take into account that $s_{\min} = 0$ and $s_{\max} = 14K$. The parameter $d_1 > 0$ controls the node density around $s = K$. Additionally, we define

$$\Psi(\omega) = \begin{cases} s_{\text{left}} + d_1 \sinh(\omega), & \omega_{\min} \leq \omega < 0, \\ s_{\text{left}} + d_1 \omega, & 0 \leq \omega \leq \omega_{\text{int}}, \\ s_{\text{right}} + d_1 \sinh(\omega - \omega_{\text{int}}), & \omega_{\text{int}} < \omega \leq \omega_{\max}. \end{cases} \tag{14}$$

where $d_1 = \frac{K}{20}$ is a proper choice in (13), while $s_{\text{left}} = \max\{0.5, e^{-0.0025T}\} \times K$, $s_{\text{right}} = K$, and $[s_{\text{left}}, s_{\text{right}}] \subset [0, s_{\max}]$.

Now, the discretized points along v , i.e., $\{v_j\}_{j=1}^n$ are given as

$$v_j = \sinh(\zeta_j)d_2, \quad j = 1, 2, \dots, n, \tag{15}$$

where $d_2 > 0$ provides a density around $v = 0$. In this paper, we employed $d_2 = \frac{v_{\max}}{500}$, where $v_{\max} = 10$. Additionally for $1 \leq j \leq n$, ζ_j stand for uniform nodes furnished as follows:

$$\Delta\zeta = (n - 1)^{-1} \sinh^{-1}\left(\frac{v_{\max}}{d_2}\right), \quad \zeta_j = (j - 1)\Delta\zeta. \tag{16}$$

Finally, the graded points alongside r are given by

$$r_k = d_3 \sinh(\zeta_k), \quad 1 \leq k \leq o, \tag{17}$$

where $r_{\max} = 1$ and $d_3 = \frac{r_{\max}}{500}$ is a positive number. It is considered $\zeta_k = (\Delta\zeta)(k - 1)$, $\Delta\zeta = \frac{1}{o-1} \sinh^{-1}\left(\frac{r_{\max}}{d_3}\right)$. Note that the variables i, j, k are local counter variables throughout the paper.

3. Constructing the RBF-FD Weights

Now, the famous inverse quadratic (IQ) RBF is considered as follows ([22], [Chapter 4]) ($i = 1, 2, \dots, \psi$):

$$\varrho(r_i) = \frac{1}{c^2 + r_i^2}, \tag{18}$$

where s is the parameter of shape and $r_i = \|\mathbf{s} - \mathbf{s}_i\|$ stands for the Euclidean distance.

Remark 1. *The motivation behind choosing this RBF is based on the recent work [23] that states that the IQ-based RBF-FD weights could be constructed theoretically more easily than multiquadrics (MQs) RBFs, which are available in the literature [24]. Besides this, another motivation stems from the fact that the IQ is an infinitely smooth RBF; under mild conditions [25], its one-dimensional function interpolants that are analytic inside a strip exponentially converge.*

To find weights of the RBF-FD methodology on a graded mesh having three points, by considering L as a linear operator of approximation in the RBF-FD methodology, one can start with [26]

$$L[\varrho(\mathbf{s}_j)] \simeq \sum_{i=1}^{\psi} Y_i \varrho(\mathbf{s}_i), \quad j = 1, 2, \dots, \psi. \tag{19}$$

This leads to ψ unknowns for ψ equations while the solutions are Y_j . Note that L is not a differential operator and is sometimes used for the discretization of the PDE problem. To illustrate further, in this section, by (19), we only focus on approximating the (1D) function derivatives. So, a derivative of the function is approximated by a combination of (three) adjacent nodes while the weights are unknown.

Hence, for $\psi = 3$, we obtain

$$\{s_i - h, s_i, s_i + wh\}, \quad w > 0, h > 0, \tag{20}$$

and write (19) as

$$f'(s_i) \simeq Y_{i-1}f(s_{i-1}) + Y_i f(s_i) + Y_{i+1}f(s_{i+1}) = \hat{f}'(s_i), \tag{21}$$

wherein \hat{f} and f are the approximate and exact values, respectively.

Remark 2. Recalling that the error equation at the point s_i can be defined by

$$\varepsilon(s_i) = \hat{f}'(s_i) - f'(s_i).$$

The reason for mentioning the error is to say how accurate the RBF-FD estimations arising from (19) are when we approximate different function derivatives by considering only three adjacent non-uniform points.

Theorem 1. When estimating the first derivative of the function f , the equation of error by the IQ RBF-FD procedure (21) is obtained as follows:

$$\varepsilon(s_i) = w \left(\frac{2f'(s_i)}{c^2} + \frac{1}{6}f^{(3)}(s_i) \right) h^2 + \mathcal{O}(h^3), \tag{22}$$

where

$$Y_{i-1} = \frac{\omega \left(\frac{2h^2(8\omega-13)}{c^2} - 5 \right)}{5h(\omega + 1)}, \tag{23}$$

$$Y_i = \frac{\omega - 1}{h\omega} - \frac{16h(\omega - 1)}{5c^2}, \tag{24}$$

$$Y_{i+1} = \frac{\frac{2h^2(13\omega-8)}{c^2} + \frac{5}{\omega}}{5h(\omega + 1)}. \tag{25}$$

Proof. The proof is similar to [24], and hence it is omitted. \square

To compute the weights of the function's second derivative, we write for $\psi = 3$:

$$f''(s_i) \simeq \Theta_{i-1}f(s_{i-1}) + \Theta_i f(s_i) + \Theta_{i+1}f(s_{i+1}) = \hat{f}''(s_i). \tag{26}$$

Hence, we can state the following theorem.

Theorem 2. When estimating the second derivative of the function f , the equation of error by the IQ RBF-FD procedure (26) is obtained via

$$\hat{\varepsilon}(s_i) = \frac{1}{3}(w - 1) \left(\frac{12f'(s_i)}{c^2} + f^{(3)}(s_i) \right) h + \mathcal{O}(h^2), \tag{27}$$

where $\hat{\varepsilon}(s_i) = \hat{f}''(s_i) - f''(s_i)$ and

$$\Theta_{i-1} = \frac{2 \left(\frac{4\omega(4\omega-7)+26}{c^2} + \frac{5}{h^2} \right)}{5(\omega + 1)}, \tag{28}$$

$$\Theta_i = \frac{2 \left(-\frac{2(\omega(8\omega-9)+8)}{c^2} - \frac{5}{h^2} \right)}{5\omega}, \tag{29}$$

$$\Theta_{i+1} = \frac{10c^2 + 4h^2(\omega(13\omega - 14) + 8)}{5c^2h^2\omega(\omega + 1)}. \tag{30}$$

Proof. The proof is similar to [24], and hence it is omitted. \square

An effective procedure could be here used for the selection of the shape parameter in numerical implementations, as comes next:

$$c = 5 \max\{\Delta s_i\}, \quad 1 \leq i \leq m - 1, \tag{31}$$

where Δs_i are the increments along the variable mesh. The motivation behind choosing the shape parameter as (31) is that we must select c to be larger than h , i.e., $c \gg h$, and also an adaptive c that changes based on the number of discretization points will affect positively more on the numerical results than a constant shape parameter. On the other hand, after using a trial and error approach, we found that a larger value as the coefficient, i.e., 5, in (31), will tend the RBF-FD methodology towards the FD approach, and a smaller value will cause some instability issues. Thus, coefficient 5 can help us to get efficient numerical results.

Now, an introduction on how RBF-FD works is necessary to discuss the different stages of the methodology. Some remarks are in order:

- In fact, the weights we obtained by now are used in an initial step by considering three points of the stencil. Each time, three interior points of the stencil are considered (in a loop), and the weights are computed.
- These values along with the weights corresponding to the first and last nodes are grouped together in a matrix, which we call the differentiation matrix in the next section.
- Note that after computing the matrices, the final matrix will not be changed during the time stepping process, as we discuss in the next section as well.
- The computed weights change only when the three adjacent nodes of the stencil or their spacings (h or w) change.

4. Numerical Method

To construct our RBF-FD solver, first, the method of lines (MOL) is employed [27,28], in which all the spatial variables are discretized to attain a system of linear ordinary differential equations (ODEs). Hence, let us define two differentiation matrices having the weights for the RBF-FD procedures on graded meshes described of Section 2 as follows:

$$M_s = \begin{cases} Y_{i,j} \text{ using (23)} & i - j = 1, \\ Y_{i,j} \text{ using (24)} & i - j = 0, \\ Y_{i,j} \text{ using (25)} & j - i = 1, \\ 0 & \text{otherwise,} \end{cases} \tag{32}$$

and

$$M_{ss} = \begin{cases} \Theta_{i,j} \text{ using (28)} & i - j = 1, \\ \Theta_{i,j} \text{ using (29)} & i - j = 0, \\ \Theta_{i,j} \text{ using (30)} & j - i = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{33}$$

We here discuss the procedure along s , while the procedure along other variables would be similar. For the discretization nodes located on the boundaries, we point these out as follows. The formulations (23)–(25) and (28)–(29) would be fruitful for rows two to the row before the last one. Accordingly, for the nodes located on the sides, we may use sided FD approximations as follows:

$$f'(s_1) = f[s_1, s_2] + f[s_1, s_3] - f[s_3, s_2] + \mathcal{O}\left((s_1 - s_2)^2\right), \tag{34}$$

and

$$f'(s_m) = -f[s_{m-1}, s_{m-2}] + f[s_{m-2}, s_m] + f[s_{m-1}, s_m] + \mathcal{O}\left((s_m - s_{m-1})^2\right), \tag{35}$$

wherein $f[l, p] = (f(l) - f(p)) / (l - p)$.

Similarly, for the four nodes $\{s_1, f(s_1)\}, \{s_2, f(s_2)\}, \{s_3, f(s_3)\}, \{s_4, f(s_4)\}$, we can obtain (see e.g., [29])

$$f''(s_1) = \frac{2(\delta s_{1,2} + \delta s_{1,3} + \delta s_{1,4})}{\delta s_{1,2}\delta s_{1,3}\delta s_{1,4}} f(s_1) + \frac{2(\delta s_{3,1} + \delta s_{4,1})}{\delta s_{1,2}\delta s_{2,3}\delta s_{2,4}} f(s_2) + \frac{2(\delta s_{2,1} + \delta s_{4,1})}{\delta s_{1,3}\delta s_{3,2}\delta s_{3,4}} f(s_3) + \frac{2(\delta s_{2,1} + \delta s_{3,1})}{\delta s_{1,4}\delta s_{4,2}\delta s_{4,3}} f(s_4) + \mathcal{O}(h^2), \tag{36}$$

where $\delta s_{l,q} = s_l - s_q$, and h stands for the maximum space width along the considered nodes of the stencil.

Recall that using a similar spirit of logic for the point s_m , we could calculate the second-order approximate formulation and its corresponding weights as follows:

$$f''(s_m) = \frac{2(\delta s_{m-3,m} + \delta s_{m-2,m} + \delta s_{m-1,m})}{\delta s_{m-3,m}\delta s_{m,m-2}\delta s_{m,m-1}} f(s_m) + \frac{2(\delta s_{m-3,m} + \delta s_{m-2,m})}{\delta s_{m-3,m-1}\delta s_{m-1,m-2}\delta s_{m-1,m}} f(s_{m-1}) + \frac{2(\delta s_{m-3,m} + \delta s_{m-1,m})}{\delta s_{m-3,m-2}\delta s_{m-2,m-1}\delta s_{m-2,m}} f(s_{m-2}) + \frac{2(\delta s_{m-2,m} + \delta s_{m-1,m})}{\delta s_{m-2,m-3}\delta s_{m-1,m-3}\delta s_{m,m-3}} f(s_{m-3}) + \mathcal{O}(h^2). \tag{37}$$

Note here that for the sake of simplicity and since the function $b(T - t)$ does not have any fluctuations in practical consideration, we may expand it up to zeroth order in order to estimate it by a scalar, i.e., $b(T - t) \simeq \beta$. The motivation behind this is to get rid of of a time-variable coefficient function and thus a time-varying matrix. To illustrate further, by this consideration, we simplify the numerical solver as much as possible.

Now, let \otimes denote the Kronecker product, and the $N \times N$ identity matrix $I = I_s \otimes I_v \otimes I_r$, is given where $N = m \times n \times o$, I_s is the $m \times m$ identity matrix for s , and for I_v and I_r similarly. Thus, now we can encapsulate the whole numerical procedure for the 3D case as follows:

$$A = \frac{1}{2} \mathcal{S}^2 \mathcal{V} (M_{ss} \otimes I_v \otimes I_r) + \frac{1}{2} \sigma_1^2 \mathcal{V} (I_s \otimes M_{vv} \otimes I_r) + \frac{1}{2} \sigma_2^2 (I_s \otimes I_v \otimes M_{rr}) + \rho_{12} \sigma_1 \mathcal{S} \mathcal{V} (M_s \otimes M_v \otimes I_r) + \rho_{13} \sigma_2 \mathcal{S} (\mathcal{V})^{\frac{1}{2}} (M_s \otimes I_v \otimes M_r) + \rho_{23} \sigma_1 \sigma_2 (\mathcal{V})^{\frac{1}{2}} (I_s \otimes M_v \otimes M_r) + \mathcal{R} \mathcal{S} (M_s \otimes I_v \otimes I_r) + \kappa (\eta I - \mathcal{V}) (I_s \otimes M_v \otimes I_r) + a (\beta I - \mathcal{R}) (I_s \otimes I_v \otimes M_r) - r I. \tag{38}$$

The square matrices $M_s, M_v, M_r, M_{ss}, M_{vv}$, and M_{rr} are derived via the corresponding weights similarly. In addition, the diagonally sparse matrices \mathcal{R}, \mathcal{V} and \mathcal{S} can be expressed as

$$\mathcal{R} = I_s \otimes I_v \otimes \text{diag}(r_1, r_2, \dots, r_o), \tag{39}$$

$$\mathcal{V} = I_s \otimes \text{diag}(v_1, v_2, \dots, v_n) \otimes I_r, \tag{40}$$

$$\mathcal{S} = \text{diag}(s_1, s_2, \dots, s_m) \otimes I_v \otimes I_r. \tag{41}$$

Taking all the weights into matrices, it would be possible to transfigure the PDE into a system of ODEs to price (3) as follows:

$$u'(t) = Au(t), \tag{42}$$

wherein the vector of unknowns is $u(t) = \underbrace{(u_{1,1,1}(t), u_{1,1,2}(t), \dots, u_{m,n,o-1}(t), u_{m,n,o}(t))}_{N \text{ entries}}^*$.

After considering the boundary conditions (5), a system of ODEs can be attained as follows:

$$u'(t) = \bar{A}u(t) = G(t, u(t)), \tag{43}$$

where \bar{A} is the system matrix consisting of the boundary conditions. This means that the boundary conditions given in (5) must be imposed to appropriate rows of A . This can be done by writing (5) in time differentiation form, and thus we obtain a new matrix \bar{A} .

5. Stability

Now, select $k + 1$ uniform temporal points, a time step size $\xi = \frac{T}{k} > 0$, $t_{i+1} = \xi + t_i$, $0 \leq i \leq k$ and $\mathbf{u}^0 = u(0)$. Take into consideration that \mathbf{u}^i is an estimate to $\mathbf{u}(t_i)$; then, one may derive our final time-integrator method. The explicit Runge–Kutta (RK) solver with four stages [30] is given by [31] (pages 165–169)

$$\mathbf{u}^{i+1} = \mathbf{u}^i + \frac{\xi}{6}(B_1 + 2B_2 + 2B_3 + B_4), \tag{44}$$

and

$$B_1 = G(t_i, \mathbf{u}^i), \tag{45}$$

$$B_2 = G\left(t_i + \frac{\xi}{2}, \mathbf{u}^i + \frac{\xi}{2}B_1\right), \tag{46}$$

$$B_3 = G\left(t_i + \frac{\xi}{2}, \mathbf{u}^i + \frac{\xi}{2}B_2\right), \tag{47}$$

$$B_4 = G(t_i + \xi, \mathbf{u}^i + \xi B_3). \tag{48}$$

Such a higher order method over time is used to solve (43), since an explicit method that is easy to implement with an order higher than the spatial local truncation error will help us to construct an efficient fast solver for the HHW PDE. Using lower order time-marching methods will result in smaller stability regions, and on the other hand, using higher order time-marching solvers may not be necessary since they need a higher computational load per step and increase the CPU times.

Theorem 3. *Let us assume that (43) satisfies the Lipschitz condition; then, we have a conditional time-stable iteration process using (44) to solve (43).*

Proof. Here, the Lipschitz condition gives (43) existence and provides the uniqueness of the solution. Besides, we remind the reader that numerical time-stepping methods are A-stable as long as there are no stability restrictions for the following linear ODE problem:

$$u' = \lambda u, \quad \text{Re}(\lambda) < 0. \tag{49}$$

Imposing the solver (44) on the system of ODEs (43) gives the following relation:

$$\mathbf{u}^{i+1} = \left(I + \xi \bar{A} + \frac{(\xi \bar{A})^2}{2!} + \frac{(\xi \bar{A})^3}{3!} + \frac{(\xi \bar{A})^4}{4!} \right) \mathbf{u}^i. \tag{50}$$

Accordingly, the A-stability would be written as

$$\left| 1 + \xi \lambda_i + \frac{(\xi \lambda_i)^2}{2} + \frac{(\xi \lambda_i)^3}{6} + \frac{(\xi \lambda_i)^4}{24} \right| \leq 1, \tag{51}$$

which is due to (50) for any λ_i as the eigenvalue of \bar{A} . The stability condition can now be given by

$$\left| 1 + \xi \lambda_{\max} + \frac{(\xi \lambda_{\max})^2}{2} + \frac{(\xi \lambda_{\max})^3}{6} + \frac{(\xi \lambda_{\max})^4}{24} \right| \leq 1. \tag{52}$$

Considering $x = \zeta \lambda_{\max}$, the inequality (52) gives rise to a nonlinear scalar equation as follows:

$$x^3 + 4x^2 + 12x + 24 = 0, \tag{53}$$

which leads to the following roots $\{-2.78529, -0.607353 - 2.8719i, -0.607353 + 2.8719i\}$. Equivalently, we can find the following condition on the step size ζ using (44) when solving (43):

$$\zeta \leq \left| \frac{2.935419}{\text{Re}(\lambda_{\max})} \right|. \tag{54}$$

This inequality on the eigenvalues of \bar{A} will determine a conditional time stability bound for the proposed solver when pricing (3). The proof is ended. \square

Since the HHW PDE (3) is a linear equation, and thus as far as it is stable under (54) and is consistent, its convergence can be proved. Accordingly, its convergence can be obtained as long as the numerical method is consistent. This can be pursued when the (maximum) step sizes along space and time tend to zero. This shows the consistency of the RBF-FD formulas.

6. Simulation Results

The target of this section is to compare the efficacy of various solvers in order to solve (3) on the same numerical domain when $v_0 = 0.04$, $r_0 = 10\%$, $T = 1$ year and $K = 100\%$. The methods are summarized as follows:

- The solver proposed in [21] on graded meshes is shown by HM, which stands for Haentjens’s method. The motivation behind choosing HM is that it is one of the most fundamental and efficient methods for solving (3),
- The second-order FD method with uniform node distribution along space and the explicit first-order Euler’s method shown by FD, see e.g., [10],
- The proposed method described in Sections 3–5 denoted by RBF-FDM based on graded meshes of Section 2.

In fact, RBF-FD is a general name for the procedure, but when the specific IQ RBF is used along with the time-stepping solver (44), then we call it RBF-FDM. The reason for choosing the explicit first-order Euler’s method for the HM and FD is because this time-stepping method has been used for these solvers in [21]. All the compared solvers are written in Mathematica 12.0 [32].

Here, the CPU time is reported in seconds. Besides this, the absolute error is computed by

$$\varepsilon = \left| \frac{u_{\text{ref}} - u_{\text{num}}}{u_{\text{ref}}} \right|, \tag{55}$$

wherein u_{ref} and u_{num} are the referenced and numerical solutions, respectively. u_{ref} is selected from the already published literature [10].

Noting that the function b is normally defined as follows:

$$b(\tau) = c_1 - c_2 \exp(-c_3\tau) \simeq \beta, \quad \tau \geq 0, \tag{56}$$

where c_1, c_2, c_3 are constants, and $\tau = T - t$. Note that c_2 and c_3 are not zero. The following tests are discussed in this section.

Example 1 ([10]). *In this test, the following sets of parameters are considered: $\kappa = 3.0$, $\eta = 0.12$, $a = 0.20$, $\sigma_1 = 0.80$, $\sigma_2 = 0.03$, $\rho_{12} = 0.6$, $\rho_{13} = 0.2$, $\rho_{23} = 0.4$, $c_1 = 0.05$, $c_2 = 0$, $c_3 = 0$, where the reference value is $u_{\text{ref}}(K, v_0, r_0, T) \simeq 16.176$.*

Example 2 ([10]). *A different set of parameters is considered in this test to compare the results of different solvers: $\kappa = 0.5$, $\eta = 0.8$, $a = 0.16$, $\sigma_1 = 0.90$, $\sigma_2 = 0.03$, $\rho_{12} = -0.5$, $\rho_{13} = 0.2$, $\rho_{23} = 0.1$, $c_1 = 0.055$, $c_2 = 0$, $c_3 = 0$, where the reference value is $u_{\text{ref}}(K, v_0, r_0, T) \simeq 20.994$.*

Since the numerical domain along the variable s is longer than the other two variables, it is convenient to consider a greater number of discretization points even based on graded meshes along the underlying asset price. Clearly, the choice of the temporal step size mostly depends on the choice of the time integrator and its stability region. For HM and FD, the radius is of course lower than the time integrator for the proposed solver, which is a fourth-order method with the stability condition (54).

The numerical pieces of evidence are put together in Tables 1 and 2. Both indicate that by increasing the number of discretization points, the accuracy is improved for all solvers, but the graded meshes for the proposed solver along its efficient structure help us to obtain higher accuracies as quickly as possible. The time to compute the weights of the RBF-FD methodology are included here. Actually, the times reported here stand for the whole computational time from the very beginning of considering the input values until the final interpolations and printing the result.

For the case of $m = 24, n = 14,$ and $o = 14$ in Example 1, the numerical solution has been plotted in Figure 1 as well. It shows that the numerical solution for the considered cut is positive without oscillations. Similarly, in Figure 2 for the Example 2, two cuts of the numerical solution are provided to re-support this.

An inquiry may arise regarding why we did not choose the same number of spatial steps and time-steps for the different methods in Tables 1 and 2. The answers lie in the fact that we use a greater number of nodes along s and then v , since their computational domains are bigger, i.e., the bigger the domain, the greater the number of nodes used. In addition, of course, smaller temporal step-sizes are used for the FD and the HM, since their time-stepping solver has a smaller stability region. The methods can simply be compared by considering their elapsed CPU times (roughly) and checking the absolute errors.

Table 1. Results of numerical simulations in Example 1.

Solver	m	n	o	N	ζ	u_{num}	ϵ	Time
FD								
	10	8	6	480	0.002	25.492	5.7×10^{-1}	0.47
	14	10	10	1400	0.001	11.098	3.1×10^{-1}	0.79
	18	12	12	2592	0.0005	17.203	6.3×10^{-2}	1.31
	24	14	14	4704	0.00025	18.731	1.5×10^{-1}	3.67
	28	16	16	7168	0.0002	13.329	1.7×10^{-1}	7.26
	45	22	22	21,780	0.00005	14.636	9.4×10^{-2}	72.65
HM								
	10	8	6	480	0.001	14.472	1.0×10^{-1}	0.56
	14	10	10	1400	0.0005	15.300	5.3×10^{-2}	1.03
	18	12	12	2592	0.00025	15.615	3.4×10^{-2}	2.76
	24	14	14	4704	0.0001	15.806	2.2×10^{-2}	9.64
	28	16	16	7168	0.0001	15.871	1.8×10^{-2}	12.69
	50	22	22	24,200	0.000025	16.006	5.9×10^{-3}	188.58
RBF-FDM								
	10	8	6	480	0.004	15.237	5.8×10^{-2}	0.75
	14	10	10	1400	0.0025	16.002	1.0×10^{-2}	1.31
	18	12	12	2592	0.002	16.029	9.0×10^{-3}	3.26
	24	14	14	4704	0.0005	16.292	7.1×10^{-3}	7.49
	28	16	16	7168	0.0004	16.246	4.3×10^{-3}	11.29
	50	22	22	24,200	0.0002	16.191	9.2×10^{-4}	123.47

Table 2. Results of numerical simulations in Example 2.

Solver	m	n	o	N	ζ	u_{num}	ε	Time
	20	10	10	2000	0.00025	22.022	4.9×10^{-2}	1.89
	24	12	12	3456	0.0002	21.436	2.1×10^{-2}	4.12
	26	14	14	5096	0.0001	19.678	6.1×10^{-2}	9.69
	28	16	16	7168	0.0001	17.376	1.7×10^{-1}	12.59
	30	18	18	9720	0.00005	17.404	1.7×10^{-1}	37.65
	36	20	20	14,400	0.000025	20.510	2.2×10^{-2}	109.47
	38	22	22	18,392	0.000025	20.275	3.3×10^{-2}	164.28
	42	22	22	20,328	0.00002	18.370	1.2×10^{-1}	231.74
HM								
	20	10	10	2000	0.00025	20.631	1.6×10^{-2}	2.14
	24	12	12	3456	0.0002	20.709	1.2×10^{-2}	4.31
	26	14	14	5096	0.0001	20.729	1.1×10^{-2}	9.17
	28	16	16	7168	0.0001	20.748	1.0×10^{-2}	13.65
	30	18	18	9720	0.00005	20.767	9.9×10^{-3}	38.12
	36	20	20	14,400	0.000025	20.810	7.8×10^{-3}	110.92
	38	22	22	18,392	0.000025	20.818	7.5×10^{-3}	170.31
	42	22	22	20,328	0.00002	20.833	6.7×10^{-3}	253.64
RBF-FDM								
	20	10	10	2000	0.004	20.859,	6.4×10^{-3}	2.54
	24	12	12	3456	0.002	20.901,	4.4×10^{-3}	5.01
	26	14	14	5096	0.001	20.931,	3.0×10^{-3}	7.58
	28	16	16	7168	0.0005	20.943,	2.4×10^{-3}	14.69
	30	18	18	9720	0.0004	20.956,	1.8×10^{-3}	37.61
	36	20	20	14,400	0.0002	20.967,	1.2×10^{-3}	103.91
	38	22	22	18,392	0.0001	20.980,	6.6×10^{-4}	172.66
	42	22	22	20,328	0.00005	20.982	5.7×10^{-4}	259.37

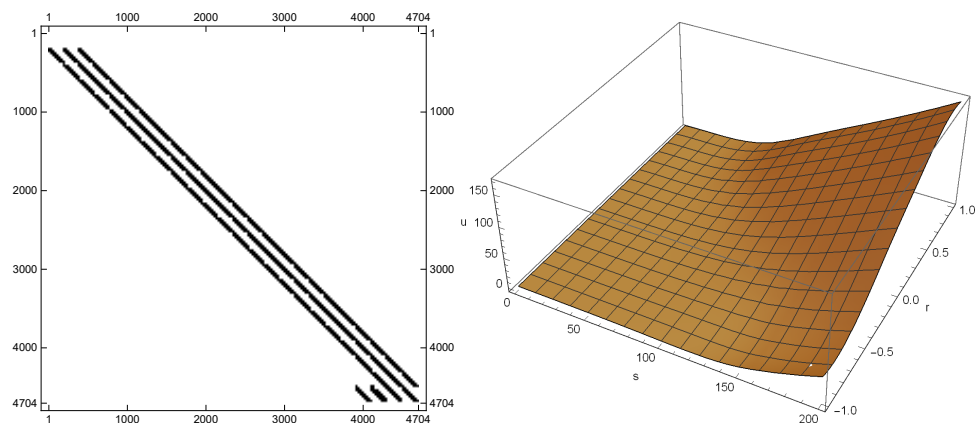


Figure 1. In Example 1 for $m = 24$, $n = 14$ and $o = 14$. (Left): The density/sparsity pattern of the system matrix \tilde{A} . (Right): The numerical solution based on RBF-FDM under the cut $U(s, 0.04, r, T)$.

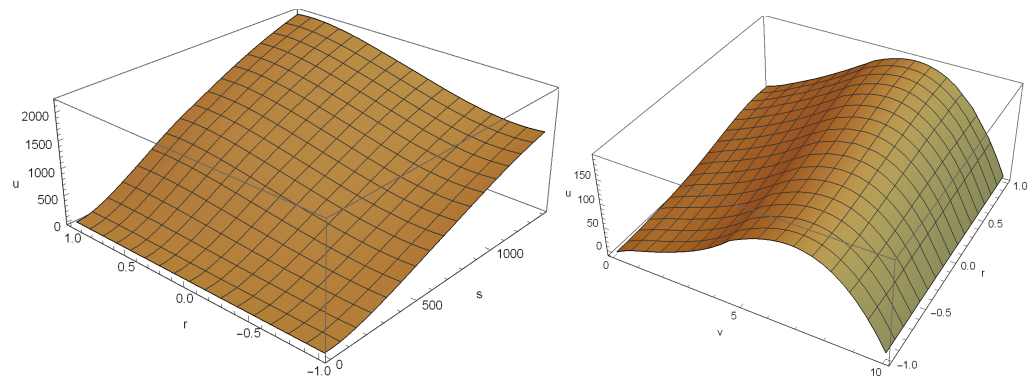


Figure 2. In Example 1 for $m = 20$, $n = 10$ and $o = 10$. (Left): The numerical solution based on RBF-FDM under the cut $U(s, 1, r, T)$. (Right): The numerical solution based on RBF-FDM under the cut $U(E, v, r, T)$.

The Greek characters are significant to calculate in order to show the stability of a numerical method in option pricing. Hence, Figure 3 shows the delta and gamma and confirms a stable numerical solution for the proposed approach.

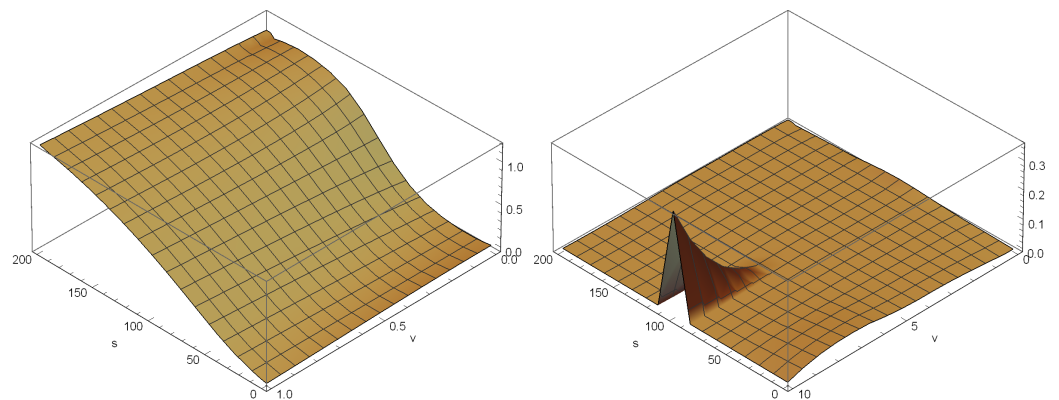


Figure 3. The Greek characters delta (Left) and gamma (Right) at $u(s, v, 0.024, 1)$ via the RBF-FDM method in Example 1 for $m = 28$, $n = 16$, and $o = 16$.

It is commented that the two methods (FD and HM) have some more stability issues in contrast to the RBF-FDM. This is because of the use of a time-stepping method with a larger stability region. The methods FD and HM use the explicit first-order Euler’s method, which of course has a smaller stability region in contrast to (44).

In Tables 1 and 2, it is noticeable that the proposed approach has a slower calculation time as well as higher accuracies. The improved accuracy in terms of absolute errors is mainly due to the application of the RBF-FDM scheme with an appropriate shape parameter, and the lower computational time is mainly due to employing lower temporal step sizes because of using (44) with larger stability regions.

With the numerical results given in this section, we believe that our fast and stable solver opens a new possibility for efficient and accurate derivative pricing.

We end this section by repeating Example 1 and only by changing $a = 0.85$ in order to check the numerical results when the Feller’s condition is unsatisfied. Considering RBF-FDM with $m = 30$, $n = 18$, and $o = 18$, the numerical results are given in Figure 4, confirming the applicability and usefulness of the proposed approach even when the Feller condition is unsatisfied.

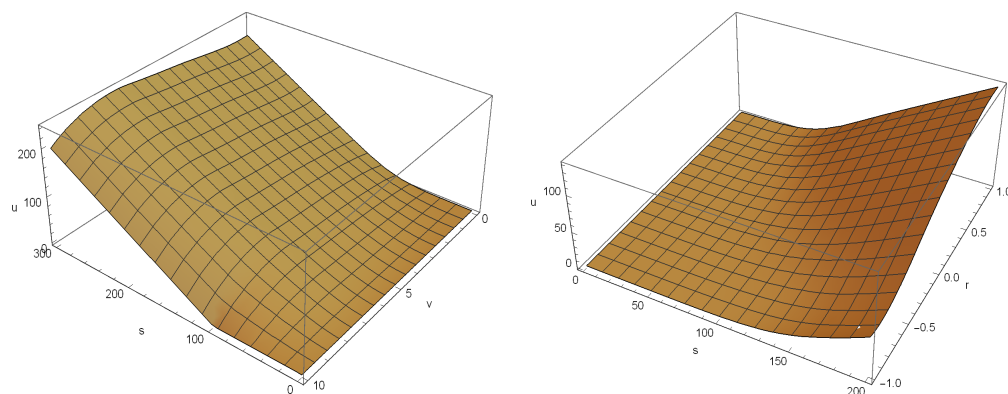


Figure 4. Results for the unsatisfied Feller condition: Left: The numerical solution based on RBF-FDM under the cut $U(s, v, 0.024, T)$. Right: The numerical solution based on RBF-FDM under the cut $U(s, 0.04, r, T)$.

7. Summary

Trading based on options can occur in daily market routines. Hence, pricing them when both the interest rate and the volatility follow some stochastic dynamics is of practical interest for risk managers and traders. This work focuses on the significant financial model of HHW as a 3D time-dependent PDE problem and proposes a new solver to expand the relevant literature's conclusions. In fact, our methodology is to consider graded meshes on all the involved spatial variables in order to concentrate on the financially hot area more. Then, the weighting coefficients for an RBF were constructed on such meshes and proved to have second and first orders to approximate the first and second derivatives of a function. MOL is then applied through extensive compact matrix notations to build up the new solver as a system of ODEs. An explicit but fast time-stepping method was used and the stability of the solver discussed in detail. Computational simulations have validated the applicability of the presented RBF-FD sparse solver for solving the HHW equation.

Investigating how to compute an optimal value for the shape parameter as well as to employ the new efficient method furnished in this paper for other kinds of derivative securities could be conducted in future works in this field of research. Besides, it is recalled that the generalized Stein–Stein model [7], which is also known as the model of Schöbel–Zhu [33], can be expressed as a system of two SDEs in a similar way to (1) but differs in some perspectives. Applying such a consideration along with stochastic interest rates can yield another generalized hybrid model, known as the hybrid Schöbel–Zhu–Hull–White PDE, which is discussed deeply in [34]. Pricing such a PDE [35,36] can be investigated by the localized meshless methods in forthcoming works.

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

Funding: This research was funded by the Natural Science Foundation of Hebei Province of China (A2020501007), the Fundamental Research Funds for the Central Universities (N2123015), and the Technical Service Project of Eighth Geological Brigade of Hebei Bureau of Geology and Mineral Resources Exploration (KJ2022-021).

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: The second author (M.Z.U.) states the following: The Deanship of Scientific Research (DSR) at King Abdulaziz University (KAU), Jeddah, Saudi Arabia has funded this project, under grant no. (RG-11-130-43). The authors thanks all four referees, particularly referee 2, for many comments and corrections on an earlier version of this manuscript that helped to improve the paper.

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

References

- Heston, S.L. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Rev. Financ. Stud.* **1993**, *6*, 327–343. [\[CrossRef\]](#)
- Adhikari, R. Foundations of Computational Finance. *Math. J.* **2020**, *22*, 1–59. [\[CrossRef\]](#)
- Cox, J.C.; Ingersoll, J.E.; Ross, S.A. A theory of the term structure of interest rates. *Econometrica* **1985**, *53*, 385–407. [\[CrossRef\]](#)
- Hull, J.; White, A. Using Hull-White interest rate trees. *J. Deriv.* **1996**, *4*, 26–36. [\[CrossRef\]](#)
- 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. The general Hull-White model and supercalibration. *Financ. Anal. J.* **2001**, *57*, 34–43. [\[CrossRef\]](#)
- Stein, J.C.; Stein, E.M. Stock price distributions with stochastic volatility: An analytic approach. *Rev. Financ. Stud.* **1991**, *4*, 727–752. [\[CrossRef\]](#)
- Djeutcha, E.; Aime Fono, L. Pricing for options in a Hull-White-Vasicek volatility and interest rate model. *Appl. Math. Sci.* **2021**, *15*, 377–384. [\[CrossRef\]](#)
- Grzelak, L.A.; Oosterlee, C.W. On the Heston model with stochastic interest rates. *SIAM J. Financ. Math.* **2011**, *2*, 255–286. [\[CrossRef\]](#)
- Ullah, M.Z. Numerical solution of Heston-Hull-White three-dimensional PDE with a high order FD scheme. *Mathematics* **2019**, *7*, 704. [\[CrossRef\]](#)
- Wong, H.Y.; Zhao, J. An artificial boundary method for the Hull-White model of American interest rate derivatives. *Appl. Math. Comput.* **2011**, *217*, 4627–4643. [\[CrossRef\]](#)
- Fornberg, B. *A Practical Guide to Pseudospectral Methods*; Cambridge University Press: Cambridge, UK, 1996.
- Soleymani, F.; Zhu, S. On a high-order Gaussian radial basis function generated Hermite finite difference method and its application. *Calcolo* **2021**, *58*, 50. [\[CrossRef\]](#)
- Kadalbajoo, M.K.; Kumar, A.; Pati Tripathi, L. Application of the local radial basis function-based finite difference method for pricing American options. *Int. J. Comput. Math.* **2015**, *92*, 1608–1624. [\[CrossRef\]](#)
- Yousuf, M.; Khaliq, A.Q.M. Partial differential integral equation model for pricing American option under multi state regime switching with jumps. *Numer. Meth. Part. Diff. Equ.* **2023**, *39*, 890–912. [\[CrossRef\]](#)
- Milovanović, S.; von Sydow, L. Radial basis function generated finite differences for option pricing problems. *Comput. Math. Appl.* **2018**, *75*, 1462–1481. [\[CrossRef\]](#)
- 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. [\[CrossRef\]](#)
- Soleymani, F. Finding an efficient machine learning predictor for lesser liquid credit default swaps in equity markets. *Iran. J. Numer. Anal. Optim.* **2023**, *13*, 19–38.
- Kluge, T. Pricing Derivatives in Stochastic Volatility Models Using the Finite Difference Method. Ph.D. Thesis, TU Chemnitz, Chemnitz, Germany, 2002.
- Soleymani, F.; Saray, B.N. Pricing the financial Heston-Hull-White model with arbitrary correlation factors via an adaptive FDM. *Comput. Math. Appl.* **2019**, *77*, 1107–1123. [\[CrossRef\]](#)
- 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. [\[CrossRef\]](#)
- Fasshauer, G.E. *Meshfree Approximation Methods with MATLAB*; World Scientific Publishing Co.: Singapore, 2007.
- Rahimi, A.; Shivanian, E.; Abbasbandy, S. Analysis of new RBF-FD weights, calculated based on inverse quadratic functions. *J. Math.* **2022**, *2022*, 3718132. [\[CrossRef\]](#)
- Soleymani, F. An efficient numerical scheme for the solution of a stochastic volatility model including contemporaneous jumps in finance. *Int. J. Comput. Methods* **2022**, *19*, 2141021. [\[CrossRef\]](#)
- Platte, R.B. How fast do radial basis function interpolants of analytic functions converge? *IMA J. Numer. Anal.* **2011**, *31*, 1578–1597. [\[CrossRef\]](#)
- Wright, G.B.; Fornberg, B. Scattered node compact finite difference-type formulas generated from radial basis functions. *J. Comput. Phys.* **2006**, *212*, 99–123. [\[CrossRef\]](#)
- Knapp, R. A method of lines framework in Mathematica. *J. Numer. Anal. Indust. Appl. Math.* **2008**, *3*, 43–59.
- Meyer, G.H. *The Time-Discrete Method of Lines for Options and Bonds: A PDE Approach*; World Scientific Publishing: Hackensack, NJ, USA, 2015.

29. Soleymani, F.; Akgül, A.; Karatas Akgül, E. On an improved computational solution for the 3D HCIR PDE in finance. *An. Stiintifice Ale Univ. Ovidius Constanta—Ser. Mat.* **2019**, *27*, 207–230. [[CrossRef](#)]
30. Sofroniou, M.; Knapp, R. *Advanced Numerical Differential Equation Solving in Mathematica*; Wolfram Mathematica, Tutorial Collection: Champaign, IL, USA, 2008.
31. Butcher, J.C. *Numerical Methods for Ordinary Differential Equations*, 2nd ed.; Wiley: Oxford, UK, 2008.
32. Wellin, P.R.; Gaylord, R.J.; Kamin, S.N. *An Introduction to Programming with Mathematica*; Cambridge University Press: Cambridge, UK, 2005.
33. Schöbel, R.; Zhu, J. Stochastic volatility with an Ornstein-Uhlenbeck process: An extension. *Eur. Financ. Rev.* **1999**, *3*, 23–46. [[CrossRef](#)]
34. Grzelak, L.A.; Oosterlee, C.W.; Van Weeren, S. Extension of stochastic volatility equity models with the Hull-White interest rate process. *Quant. Finance* **2012**, *12*, 89–105. [[CrossRef](#)]
35. van Haastrecht, A.; Lord, R.; Pelsser, A.; Schrager, D. Pricing long-maturity equity and FX derivatives with stochastic interest rates and stochastic volatility. *Insur. Math. Econ.* **2009**, *45*, 436–448. [[CrossRef](#)]
36. 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](#)]

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.