Mathematical Modelling and Analysis
Volume 23, Issue 1, 117–138, 2018
https://doi.org/10.3846/mma.2018.008

http://mma.vgtu.lt
ISSN: 1392-6292
eISSN: 1648-3510

A Local Radial Basis Function Method for High-Dimensional American Option Pricing Problems*

Rafael Company[a], Vera N. Egorova[b], Lucas Jódar[a] and Fazlollah Soleymani[a]

[a] *Instituto Universitario de Matemática Multidisciplinar, Universitat Politècnica de València*
Camino de Vera s/n, 46011 Valencia, Spain
[b] *BCAM – Basque Center for Applied Mathematics*
Alameda de Mazarredo 14, 48009 Bilbao, Basque Country, Spain
E-mail(*corresp.*): egorova.vn@gmail.com

Received June 18, 2017; revised December 9, 2017; accepted December 10, 2017

Abstract. In this work, we apply the local Wendland radial basis function (RBF) for solving the time-dependent multi dimensional option pricing nonlinear PDEs. Firstly, cross derivative terms of the PDE are removed with a change of spatial variables based in LDL^T factorization of the diffusion matrix. Then, it is discussed that the valuation of a multi-asset option up to 4D can be computed using a modified shape parameter algorithm. In fact, several experiments containing of three and four assets are worked out showing that the results of the presented method are in good agreement with the literature and could be much more accurate once the shape parameter is chosen carefully.

Keywords: radial basis functions, cross derivative elimination, Wendland function, multi-asset problem, American option pricing.

AMS Subject Classification: 65M70; 65D15; 91G60.

1 Introduction

Multi-dimensional option pricing problems are frequent in the industry related to real markets. After the 2008 financial crisis such as regulation, fiscal issues,

* This work has been partially supported by the European Union in the FP7-PEOPLE-2012-ITN program under Grant Agreement Number 304617 (FP7 Marie Curie Action, Project Multi-ITN STRIKE-Novel Methods in Computational Finance) and the Ministerio de Economía y Competitividad Spanish grant MTM2013-41765-P.

capitalization costs and currencies' volatilities, the number of exotic options sold has reduced. Nonetheless, the demand of competitive and reliable numerical methods for solving multidimensional problems (arising from related models) continues claiming the attention of academia.

Monte Carlo methods are popular in spite of their slow convergence because the computational cost scales linearly with the number of underlying variables, [8, chapter 8] and [18, chapter 12]. Fourier expansion-based methods do not suffer the so-called "curse" of dimensionality, but they require that the point characteristic function of the underlying asset variables be analytic, see e.g. [10]. Finite difference methods [23, chapter 7] and binomial tree methods [2] have better convergence rates but the computational cost and their manageabilities are not so convenient for high dimensions. Several techniques, including Monte Carlo, Fourier, finite difference and radial basis function methods have been used in [24].

Meshfree collocation methods and semi-discretization schemes are particularly convenient for the treatment of multidimensional partial differential equations (PDEs) because one reduces the algebraic complexity coming from the approximation of the partial derivatives and provide flexibility with respect to the geometry of the computational domain [5]. In addition, in terms of PDE solvers for high-dimensional PDEs, one may refer to [28].

To be more precise, meshfree RBF methods are of interest for solving PDEs due to alternative convergence properties [5, chapter 2], implementation advantages, etc. Since 1997 in [12] several authors started applying RBF methods to European-style financial problems and later this approach has found a wider use for valuation of options with early-exercise features, see e.g. [3, 9]. Authors in [19] presented an improved RBF method that is 20 times faster than an adaptive finite difference method in one and two space dimensions based several numerical experiments *only*.

The meshfree RBF method for the solution of PDEs is similar to the spectral method of lines approach, i.e., we approximate the solution U corresponding to M spatial variables $\mathbf{y} = (y_1, \ldots, y_M)$ and time τ as follows:

$$U(\mathbf{y}, \tau) = \sum_{j=1}^{N} \alpha_j(\tau) \phi\left(\left\|\mathbf{y} - \mathbf{y}^j\right\|\right),$$

where time and space have been decoupled. Here centers $\mathbf{y}^j$ form a discretization of the domain. The coefficients $\alpha_j(\tau)$ have to be determined by imposing that $U(\mathbf{y}, \tau)$ satisfies the equation at the centers $\mathbf{y}^j$. The radial function $\phi\left(\|\cdot\|\right)$ determines the approximation space as the span of the functions $\phi\left(\left\|\cdot - \mathbf{y}^1\right\|\right), \ldots, \phi\left(\left\|\cdot - \mathbf{y}^N\right\|\right)$.

RBFs are divided into two main categories of global ones such as multiquadric (MQ), inverse multiquadric (IMQ) and Gaussian functions; and local RBFs such as Wendland function, see for more [5, chapter 11]. The advantage of local RBFs in contrast to the global RBFs is that they result in sparse evaluation (interpolation) matrices which could be stored and handled much faster than the corresponding dense ones resulted from a global RBF. Moreover, the local RBFs in contrast to the global RBFs are less sensitive to the choice of the shape parameter. However, the global RBFs are still the main choice when a very sharp shape parameter is selected, which is practically hard to find (specially for high dimensional time-dependent PDE problems).

The most useful RBF with localized feature which lead to sparse symmetric positive definite matrices and a quicker interpolation process is Wendland's function [25]. Herein, the local Wendland RBF [27] is denoted by $\phi_{M,k}$, where M and k stand for the dimension and the smoothness. Several special cases of the Wendland functions as well as some global RBFs are given in Table 1.

An application of Wendland RBF for pricing a class of option pricing problems via the approach of partition of unity up to two dimensions has recently been discussed in [22]. Other modifications of RBF meshfree methods such as RBF-FD can be found in [7].

The aim of this paper is to introduce in the RBF method, some strategies to improve the competitiveness allowing the resolution of four spatial dimensional option pricing problems unlike the recent pessimistic forecast of prestigious practitioners of the method [13, p. 160]. In fact in Section 2, we transform the original multi-asset American option problems [23]:

$$\frac{\partial V}{\partial \tau} = \frac{1}{2}\sum_{i,j=1}^{M} \rho_{ij}\sigma_i\sigma_j S_i S_j \frac{\partial^2 V}{\partial S_i\,\partial S_j} + \sum_{i=1}^{M}(r-q_i)S_i\frac{\partial V}{\partial S_i} - rV + F(V), \qquad (1.1)$$

by eliminating the cross derivative terms that reduces the computational cost and eliminate numerical instabilities. Here τ, V, S_i, q_i, r, σ_i $\rho_{i,j}$ denote the maturity, the value of the option price, the i-th asset, the constant dividend yield of the i-th asset, risk-free interest rate, i-th volatility, and the correlation parameter between the i-th and the j-th assets, respectively. Note that $V(\mathbf{S},\tau) = V(S_1,\dots,S_M,\tau)$, $\tau = T - t$, $\rho_{ii} = 1$, $\rho_{ij} = \rho_{ji}$, $i \neq j$, and $|\rho_{ij}| \leq 1$.

In addition in (1.1), the initial condition for a basket put option is the payoff function given by

$$V(\mathbf{S},0) = \Big(E - \sum_{i=1}^{M}\omega_i S_i\Big)^{+}, \qquad (1.2)$$

where E is the strike price and ω_i is the positive weight of the corresponding i-th asset in the basket.

Note that the nonlinear term F in (1.1) is the penalty term applied by the idea of penalty method [29] in order to make the original option pricing PDE with free boundaries into a nonlinear PDE for American options. One of the ways for introducing the penalty term is:

$$F(V) = \lambda\big(V(\mathbf{S},0) - V(\mathbf{S},\tau)\big)^{+},$$

where λ is the non-negative penalty parameter.

Lately in [4], it is discussed that one can transform (1.1) into a multidimensional PDE with constant coefficients using the following dimensionless logarithmic transformation

$$x_i = \frac{1}{\sigma_i}\ln\frac{S_i}{E},\ i = 1,\dots,M, \quad W(\mathbf{x},\tau) = \frac{V(\mathbf{S},\tau)}{E}, \qquad (1.3)$$

where $\mathbf{x} = [x_1, \dots, x_M]^T$, and obtain

$$\frac{\partial W}{\partial \tau} = \frac{1}{2} \sum_{i=1,j=1}^{M} \rho_{ij} \frac{\partial^2 W}{\partial x_i \partial x_j} + \sum_{i=1}^{M} \delta_i \frac{\partial W}{\partial x_i} - rW + \frac{1}{E} F(EW), \\ x_i \in \mathbb{R}, \quad i = 1, \dots, M, \quad 0 < \tau \le T, \tag{1.4}$$

while $\delta_i = (r - q_i - \sigma_i^2/2)/\sigma_i$.

In [4], it has been discussed that the LDL^T factorization of the symmetric positive semidefinite matrix $R = (\rho_{ij})$ allows to remove the cross derivative terms. In general, for the M-dimensional problem, we can remove the following number of terms $\frac{1}{2}M(M-1)$ and subsequently avoid such a number of matrices in the process of setting up meshfree RBF methods. This clearly reduces the computational load and time for executing meshfree RBF methods for solving high-dimensional option pricing problems.

By using the positive semi-definitive property of the correlation matrix $R = (\rho_{ij})$ and taking advantage of a diagonal pivoting strategy, we can apply the stable LDL^T factorization, where L is a unit lower triangular matrix and D is a diagonal matrix with positive diagonal elements D_{ii}, such that

$$PRP^T = LDL^T, \tag{1.5}$$

wherein P is a permutation matrix. By using the linear transformation

$$\mathbf{y} = [y_1, \dots, y_M]^T = C\mathbf{x}, \quad U(\mathbf{y}, \tau) = W(\mathbf{x}, \tau), \tag{1.6}$$

where $C = (c_{ij})_{1 \le i,j \le M} = L^{-1}P$, the equation (1.4) becomes

$$\frac{\partial U}{\partial \tau} = \frac{1}{2} \sum_{i=1}^{M} D_{ii} \frac{\partial^2 U}{\partial y_i^2} + \sum_{i=1}^{M} \Big(\sum_{j=1}^{M} \delta_j c_{ij} \Big) \frac{\partial U}{\partial y_i} - rU + \frac{1}{E} F(EU), \tag{1.7}$$

wherein the cross derivative terms have been removed. Under transformations (1.3) and (1.6) the initial condition (1.2) takes the following form

$$U(\mathbf{y}, 0) = \left(1 - \sum_{i=1}^{M} \omega_i e^{\sigma_i x_i} \right)^+,$$

where $\mathbf{x} = [x_1, ..., x_M]^T = C^{-1}\mathbf{y} = P^T L \mathbf{y}$.

A modification of Hardy's method [11] for the choice of the shape parameter using Wendland multidimensional RBF is proposed in Section 3. In Sections 4 and 5, we present a constructive semi-discretization procedure for obtaining the underlying time-dependent system of ordinary differential equations (ODEs). The approximate solution of the ODEs systems is obtained by using the exponential time integration method together with Simpson's quadrature formula for matrix exponential integration. Section 6 illustrates with numerical examples the improvements including comparisons with the existing competitive methods.

Table 1. Several common RBFs. The shape parameter c controls the flatness and d is the Euclidean distance, i.e., $d = \|\cdot\|_2$

Name	Definition	Degree of smoothness
Gaussian	$\exp\left(-(\frac{d}{c})^2\right)$	C^∞
Multiquadric	$\sqrt{1+(\frac{d}{c})^2}$	C^∞
Inverse Multiquadric	$\frac{1}{\sqrt{1+(\frac{d}{c})^2}}$	C^∞
Inverse quadratic	$\frac{1}{1+(\frac{d}{c})^2}$	C^∞
Matérn type 0	$\exp\left(-(\frac{d}{c})\right)$	C^0
Matérn type 2	$\exp\left(-(\frac{d}{c})\right)(1+\frac{d}{c})$	C^2
Matérn type 4	$\exp\left(-(\frac{d}{c})\right)(3+3\frac{d}{c}+(\frac{d}{c})^2)$	C^4
Cubic	$\lvert\frac{d}{c}\rvert^3$	Piecewise
Thin plate spline	$(\frac{d}{c})^2\log(\frac{d}{c})$	Piecewise
Wendland type $\phi_{3,1}$	$\left(1-\frac{d}{c}\right)_+^4\left(1+4\frac{d}{c}\right)$	C^2 for $M \leq 3$
Wendland type $\phi_{3,2}$	$\left(1-\frac{d}{c}\right)_+^6\left(3+18\frac{d}{c}+35(\frac{d}{c})^2\right)$	C^4 for $M \leq 3$
Wendland type $\phi_{5,2}$	$\left(1-\frac{d}{c}\right)_+^7\left(3+21\frac{d}{c}+48(\frac{d}{c})^2\right)$	C^4 for $M \leq 5$

2 Cross derivative elimination

It is well known that the existence of the cross derivative terms in the PDEs could cause several drawbacks such as instability and oscillation in the process of finding their solutions. Apart from this point of view, the presence of such terms will increase the computational effort and time for solving these PDEs. To discuss further, in the meshfree methods, each term in a PDE must be represented by a matrix whose entries should be calculated element-wise. As a result, a PDE with fewer terms and specially without cross derivative terms is subject to faster implementation in terms of meshfree RBF methodology. There are several approaches to remove the cross derivative terms including the algebraic and factorization ways [20].

In 2008 [15], Larsson et al. investigated several transformations on the multi-dimensional European Black-Scholes model to transform into a multidimensional parabolic time-dependent heat equation (without any mixed derivative terms), and then solve the transformed PDE with reasonable reduction of computational load in two and three dimensional spaces.

The present procedure of removing cross-derivative terms described in (1.3)–(1.7) is efficient and stable, see [4] and the references therein. In what follows we extend the results of [4] by furnishing new simplified closed forms of the coefficients in the fully transform case. This would be the contribution of this section.

For higher-dimensional problems ($M > 3$) it is not a trivial task to obtain transformation matrix C. This motivates us to search a possibility to obtain the expression of the entries c_{ij} in terms of the data in the standard case of a positive definite correlation matrix when no permutation of rows is needed and consequently $C = L^{-1}$.

Let us consider the LU factorization of the $(M-1)\times(M-1)$ correlation matrix $L_{M-1}U_{M-1}=R_{M-1}$ while

$$L_M = \left(\begin{array}{c|c} L_{M-1} & 0 \\ \hline \alpha_M & 1 \end{array}\right),$$

with $\alpha_M = [\rho_{M,1},\dots,\rho_{M,M-1}]U_{M-1}^{-1}$, [14, p. 76].

By computing the inverse of the block matrix L_M, one gets

$$C_M = L_M^{-1} = \left(\begin{array}{c|c} L_{M-1}^{-1} & 0 \\ \hline -\alpha_M L_{M-1}^{-1} & 1 \end{array}\right).$$

Then, we can write $-\alpha_M L_{M-1}^{-1} = -[\rho_{M,1},\dots,\rho_{M,M-1}]U_{M-1}^{-1}L_{M-1}^{-1}$ and equivalently $-\alpha_M L_{M-1}^{-1} = -[\rho_{M,1},\dots,\rho_{M,M-1}]R_{M-1}^{-1}$. We obtain

$$(C_M)_{M,j} = -\sum_{k=1}^{M-1} a_{M,k}[(R_{M-1}^{-1})_{k,j}] = \sum_{k=1}^{M} a_{M,k}(-1)^{k+j+1}\frac{M_{jk}(R_{M-1})}{\varDelta_{M-1}},$$

where $1\le j\le M-1$, $\varDelta_{M-1}=\det(R_{M-1})$ and $M_{j,k}(A)$ denotes the complementary minor of the element $a_{j,k}$ of matrix A. Using a development of a determinant with elements and co-factors of a row, we have $(C_M)_{M,j} = (-1)^{M+j}M_{j,M}(R_M)/\varDelta_{M-1}$ and by construction, the non zero entries of the unit lower triangular matrix C_M take the form

$$c_{ij} = (-1)^{i+j}M_{j,i}(R_i)/\varDelta_{i-1}, \quad c_{ii}=1, \quad 1\le j<i\le M. \tag{2.1}$$

Moreover, it is easy to show that the elements D_{ii} of the diagonal matrix D are

$$D_{11}=1, \quad D_{ii}=\varDelta_i/\varDelta_{i-1}, \quad 2\le i\le M, \tag{2.2}$$

where R_i is the $i\times i$ sub-matrix of the correlation matrix R consisting of the first i-th rows and columns, $M_{j,i}(R_i)$ is a (j,i) complementary minor of the matrix R_i, and $\varDelta_i$ is the leading principal minor of order i of matrix R.

Several examples up to 4D have been included in Section 6. For this reason, now only the change of variables allowing the elimination of the cross derivative terms and the resulting PDE is explicitly presented for the four-asset case as follows. Let us consider the positive definite correlation matrix $R=R_4=(\rho_{i,j})_{4\times4}$, and consider the transformation matrix $C=(c_{ij})_{4\times4}$ whose nonzero elements are given by (2.1). After logarithmic change (1.3) together with linear transformations (1.6), the new spatial variables and the unknown take the form

$$\begin{aligned}
y_1 &= \frac{1}{\sigma_1}\ln\frac{S_1}{E}, \quad y_2 = \frac{c_{12}}{\sigma_1}\ln\frac{S_1}{E}+\frac{1}{\sigma_2}\ln\frac{S_2}{E},\\
y_3 &= \frac{c_{31}}{\sigma_1}\ln\frac{S_1}{E}+\frac{c_{32}}{\sigma_2}\ln\frac{S_2}{E}+\frac{1}{\sigma_3}\ln\frac{S_3}{E},\\
y_4 &= \frac{c_{41}}{\sigma_1}\ln\frac{S_1}{E}+\frac{c_{42}}{\sigma_2}\ln\frac{S_2}{E}+\frac{c_{43}}{\sigma_3}\ln\frac{S_3}{E}+\frac{1}{\sigma_4}\ln\frac{S_4}{E},\\
U(y_1,y_2,y_3,y_4,\tau) &= \frac{1}{E}V(S_1,S_2,S_3,S_4,\tau).
\end{aligned} \tag{2.3}$$

With respect to the new variables (2.3), the transformed PDE results

$$\begin{aligned}\frac{\partial U}{\partial \tau} =&\frac{1}{2}\frac{\partial^2 U}{\partial y_1^2}+\frac{1}{2}D_{22}\frac{\partial^2 U}{\partial y_2^2}+\frac{1}{2}D_{33}\frac{\partial^2 U}{\partial y_3^2}+\frac{1}{2}D_{44}\frac{\partial^2 U}{\partial y_4^2}\\ &+\delta_1\frac{\partial U}{\partial y_1}+\left[\delta_2+c_{21}\delta_1\right]\frac{\partial U}{\partial y_2}+\left[\delta_1 c_{31}+\delta_2 c_{32}+\delta_3\right]\frac{\partial U}{\partial y_3}\\ &+\left[\delta_1 c_{41}+\delta_2 c_{42}+\delta_3 c_{43}+\delta_4\right]\frac{\partial U}{\partial y_4}-rU,\end{aligned} \tag{2.4}$$

where D_{ii} are defined by (2.2). This formulation has six terms fewer than the original form (1.1) in the case of 4D problem and it is with constant coefficients. This advantage helps us to apply the meshfree RBF methods in solving the four dimensional option pricing problem with RBF meshfree methods.

Note that for the cases $M = 2$ and $M = 3$, both changes of variables and transformed PDEs are obtained from (2.3) and (2.4) by considering only variables y_1 and y_2 or y_1, y_2 and y_3, respectively.

3 A modification of Hardy's technique for the multi-asset case

Since the solution of the PDE (1.1) should be at least twice differentiable in space, the general Wendland's formula for any number of spatial variables M takes the following form [25]

$$\phi_{M,1}\left(\frac{d}{c}\right)=\left(1-\frac{d}{c}\right)_+^l\left(1+\frac{d}{c}\left(l\right)\right),$$

where $l = [M/2] + 3$, d is the distance between two supporting nodes (i.e., the radius), $(1 - d/c)_+ = \max(1 - d/c, 0)$, c stands for the shape parameter that should be chosen carefully and $[\cdot]$ denotes the floor function.

In the case of one spatial variable $x \in \mathbb{R}$, Wendland's function can be written as follows:

$$\phi_{1,1}\left(\frac{d}{c}\right)=\left(1-\frac{d}{c}\right)_+^3\left(1+3\frac{d}{c}\right). \tag{3.1}$$

Furthermore, the Wendland functions of the smoothness C^4 which provide better accuracies in solving practical problems can be written as follows:

$$\phi_{M,2}\left(\frac{d}{c}\right)=\left(1-\frac{d}{c}\right)_+^{l+2}\left(3+(3l+6)\frac{d}{c}+(l^2+4l+3)\left(\frac{d}{c}\right)^2\right), \tag{3.2}$$

where $l = [M/2]+3$. It is known [5, chapters 16 and 17] that there is a connection between RBF interpolation and polynomial interpolation and because of that the RBF interpolation also suffers from a phenomenon which is similar to the famous Runge phenomenon for polynomial interpolation, and accordingly a good choice of the shape parameter can alleviate this effect.

Technically speaking, the choice of the shape parameter plays an important role in the accuracy and stability of the mesh-free RBF-type methods [6, 17].

The problem of decreased stability as the shape parameter increases is well known. This is important because highest accuracy is often found at some large shape parameter, which may be in an unstable region. This relation between accuracy and stability is occasionally called [5, p. 138] as the trade-off principle.

In general, there are three main factors that could affect the optimal shape parameter c for giving the most accurate results. These three factors are the scale of supporting region, the number of supporting nodes, and the distribution of supporting nodes.

The flexibility and potential for improved accuracy offered by the shape parameter must be enforced into the algorithms when employing the RBF approach for solving PDEs.

It should be remarked that in case of choosing a shape parameter with *trial-and-error approach*, this freedom is often viewed as a disadvantage since we are forced to make a decision on the choice of the shape parameter randomly. The technique of trial-and-error is mostly used for "academic" examples [5, p. 142]. In many cases, the selection of an optimal shape parameter via trial and error will end up being a rather subjective process. Due to this, applying the existing algorithms for computing a good shape parameter or introducing a new one according to the requirements of the PDE that we are solving is indispensable.

One of the earliest techniques in order to suggest a shape parameter was proposed by Hardy in [11] and is given by

$$c = 0.815\theta, \quad \theta = \frac{1}{N}\sum_{i=1}^{N}\theta_i, \tag{3.3}$$

where θ_i is the distance of the i-th data point to its nearest neighbour.

Another technique which is efficient for 1D and 2D problems was proposed by Rippa in [21]. This technique can be illustrated as follows. By fixing c and for any $\ell = 1, \ldots, N$, we first compute

$$e_\ell = |f(y_\ell) - P_f^{[\ell]}(y_\ell)|,$$

wherein

$$P_f^{[\ell]}(y_\ell) = \sum_{j=1}^{N-1} \alpha_j^{[\ell]} \varphi\left(\|y - y_j^{[\ell]}\|\right).$$

Now, a good shape parameter can be deduced by minimizing the cost function, viz, $\mathbf{e} = [e_1, \cdots, e_N]^T$, via $\|\mathbf{e}\|$. It is reminded that the whole of this procedure was simplified by Rippa via proposing the following final formula:

$$e_\ell = \alpha_\ell / P_{\ell\ell}^{-1}, \tag{3.4}$$

where P is the interpolation matrix and α is the coefficient vector of the RBF interpolation. In addition, the initial condition of the PDE, i.e., the payoff is used as the vector in the construction of a good shape parameter. Note that in this technique, the choice of the shape parameter also depends on the choice of the input interval whereas the good value of the shape parameter is predicted to be in that area.

For high-dimensional problems, the techniques which are mostly based on the distance of the nodes, such as (3.3) are quicker rather than methods based on matrix-inverse computation. In order to propose a new technique for high-dimensional problems, we can apply a modification of the Hardy's technique (3.3) in finding a good shape parameter for the M dimensional multi-asset option pricing problem in what follows:

$$c = M(0.815\theta). \tag{3.5}$$

The technique (3.5) extends (3.3) and can consider more centers in the RBF schemes for high dimensions.

To justify the reason of such a proposed variant of (3.3), it should be reminded that the local Wendanld RBF results in sparse interpolation matrices, but it does not possess an exponential rate of convergence in like the global RBFs which possessed dense ill-conditioned matrices. In between, the main effort is to choose the shape parameter to get both benefits as much as possible, i.e., sparsity of the matrices and higher rate of convergence. Hence, the factor M, which arises from the dimension problem is multiplied to the Hardy's shape parameter. Based on this, more supporting nodes are involved in the process of filling the interpolation matrices which would cause in higher rate of convergence speed while the whole interpolation matrix is still sparse.

4 Application for one asset options

In the case of one asset, the option price is the solution of the following well-known PDE problem for $S > 0,\ 0 < \tau \le T$:

$$\frac{\partial V}{\partial \tau} = \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + (r-q) S \frac{\partial V}{\partial S} - rV + \lambda \left(V(S,0) - V(S,\tau) \right)^+ , \tag{4.1}$$

where λ is the penalty parameter. Note that the case $\lambda = 0$ corresponds to European option, while $\lambda \gg 1$ leads to American option. Here, the initial condition is defined by the payoff function: $V(S,0) = (E-S)^+$.

Let us consider the dimensionless transformation $x = \ln \frac{S}{E}$, $U(x,\tau) = \frac{V(S,\tau)}{E}$. Then the problem (4.1) takes the following form:

$$\frac{\partial U}{\partial \tau} = \frac{\sigma^2}{2} \frac{\partial^2 U}{\partial x^2} + \left(r - q - \frac{\sigma^2}{2} \right) \frac{\partial U}{\partial x} - rU + \lambda \left(U(x,0) - U(x,\tau) \right)^+ , \tag{4.2}$$
$$U(x,0) = (1 - e^x)^+.$$

Problem (4.2) can now be solved by using the Wendland RBF approach.

Let us choose N centers $x^1, x^2, \ldots, x^N$ in some fixed domain $[x_{min}, x_{max}]$, not necessary uniformly distributed, while the numerical solution at these points is denoted by $u_j(\tau)$, $1 \le j \le N$. All N values form a vector solution $u(\tau) = [u_1(\tau), \ldots, u_N(\tau)]^T$. Let $p_j(x)$ be the function defined by

$$p_j(x) = \phi_{1,1} \left(\frac{|x - x_j|}{c} \right)$$

and $P_{ij} = p_j(x^i)$, $1 \leq i, j \leq N$. From (3.1) one gets

$$P_{ij} = \left(1 - \frac{|x_i - x_j|}{c}\right)_+^3 \left(1 + 3\frac{|x_i - x_j|}{c}\right) \geq 0 \tag{4.3}$$

and $P = (P_{ij})_{N\times N}$ is the so-called interpolation matrix. Applying RBF method, we assumed that solution of (4.2) is approximated by

$$u(x,\tau) = \sum_{j=1}^{N} \alpha_j(\tau) p_j(x). \tag{4.4}$$

Let P'_{ij} and P''_{ij} denote the first and second derivatives of $P_j(x)$ at point $x = x^i$ respectively. From (3.1) one gets

$$P'_{ij} = -\frac{12}{c^2}\left(1 - \frac{|x_i - x_j|}{c}\right)_+^2 (x_i - x_j), \tag{4.5}$$

$$P''_{ij} = -\frac{12}{c^2}\left(1 - \frac{|x_i - x_j|}{c}\right)_+ \left(1 - 3\frac{|x_i - x_j|}{c}\right). \tag{4.6}$$

Matrices P' and P'' are defined by

$$P' = (P'_{ij})_{N\times N}, \quad P'' = (P''_{ij})_{N\times N}. \tag{4.7}$$

From (4.2), (4.4)–(4.7), one gets the system of ODEs

$$\frac{du(\tau)}{d\tau} = \varPhi P^{-1} u(\tau) + \lambda \left(u(0) - u(\tau)\right)^+, \tag{4.8}$$

where $\varPhi \in \mathbb{R}^{N\times N}$ is the following matrix

$$\varPhi = \frac{\sigma^2}{2} P'' + \left(r - q - \frac{\sigma^2}{2}\right) P' - rP$$

and vector of unknown coefficients $\alpha(\tau) = [\alpha_1(\tau), \ldots, \alpha_N(\tau)]^T$ has been eliminated by using $u(\tau) = P\alpha(\tau)$.

In this work and just like [5, p. 114] in order to avoid any difficulties with the computation of ill-conditioned matrices, we use the pseudo-inverse known as Moore-Penrose inverse in our implementations.

From (4.3), (4.5) and (4.6) it follows that matrices P and P'' are symmetric, Toeplitz and Hermitian, while P' is skew-symmetric. Moreover, since entries of P are non-negative from (4.3), so P^{-1} is an M-matrix.

System (4.8) can be solved by exponential time integration. Let us denote $\tau^n = n\varDelta\tau$, where $\varDelta\tau = \frac{T}{N_\tau}$, $0 \leq n \leq N_\tau$ and N_τ is the given number of time steps. Then, the solution on the next time level can be calculated as follows:

$$u(\tau^{n+1}) = e^{\varPhi P^{-1}\varDelta\tau} u(\tau^n) + \lambda \int_0^{\varDelta\tau} e^{\varPhi P^{-1}s} \left(u(0) - u(\tau^{n+1} - s)\right)^+ ds.$$

Assuming the first order approximation $u(\tau^{n+1}-s)$ is approximated by $u(\tau^n)$ for $0 \le s \le \varDelta\tau$. We apply the Simpson's rule for the numerical evaluation of the integral using

$$\int_0^{\varDelta\tau} e^{As} ds \simeq \varphi(A, \varDelta\tau) = \frac{\varDelta\tau}{6}\left(I + 4e^{A\frac{\varDelta\tau}{2}} + e^{A\varDelta\tau}\right),$$

where the coefficient matrix is $A = \varPhi P^{-1}$ and I is the $N \times N$ identity matrix. Finally, by denoting $u(\tau^n) = u^n$, the numerical scheme for the solution of the problem (4.2) is given by:

$$u^{n+1} = e^{A\varDelta\tau}u^n + \varDelta\tau\lambda\varphi(A, \varDelta\tau)\left(u^0 - u^n\right)^+. \tag{4.9}$$

5 Application for the multi-asset case

For the multi-asset case, first we apply the transformation (1.5) in order to eliminate the cross derivative terms to obtain (1.7). In the truncated numerical domain $\varOmega \subset \mathbb{R}^M$ we choose N nodes $\mathbf{y}^i = (y_1^i, \ldots, y_M^i)$, $i = 1, \ldots, N$. Then the Eucledean distance d_{ij} between two nodes $\mathbf{y}^i$ and $\mathbf{y}^j$ is calculated by

$$d_{ij} = d_{ji} = \left\|\mathbf{y}^i - \mathbf{y}^j\right\| = \sqrt{\sum_{m=1}^{M}\left(y_m^i - y_m^j\right)^2}.$$

Here the numerical solution at center j is denoted by $u_j(\tau)$ and the vector solution is $u(\tau) = [u_1(\tau), \ldots, u_N(\tau)]^T$ as in the 1D-case. Let $p_j(\mathbf{y}) = \phi_{M,2}\left(\frac{\|\mathbf{y}-\mathbf{y}^j\|}{c}\right)$, and denote $P_{ij} = p_j(\mathbf{y}^i)$, $1 \le i, j \le N$. From (3.2) one gets

$$P_{ij} = \left(1 - \frac{d_{ij}}{c}\right)_+^{l+2}\left[(l^2 + 4l + 3)\frac{d_{ij}^2}{c^2} + (3l + 6)\frac{d_{ij}}{c} + 3\right],$$

where $P = (P_{ij})_{N\times N}$ is the interpolation matrix. Applying RBF method, we assume that solution of (1.7) is approximated by $u(\mathbf{y}, \tau) = \sum_{j=1}^{N}\alpha_j(\tau)p_j(\mathbf{y})$. Let $(P'_m)_{ij}$ and $(P''_m)_{ij}$ denote the first and second partial derivative of $p_j(\mathbf{y})$ with respect to $\mathbf{y}_m$ at the point $\mathbf{y} = \mathbf{y}^i$, respectively. From (3.2) one gets

$$\begin{aligned}(P'_m)_{ij} =& \frac{\partial p_j}{\partial y_m}(\mathbf{y}^i) = -\frac{(l{+}3)(l{+}4)}{c^2}(y_m^i{-}y_m^j)\left(1{-}\frac{d_{ij}}{c}\right)_+^{l+1}\left((l+1)\frac{d_{ij}}{c} + 1\right),\\ (P''_m)_{ij} =& \frac{\partial^2 p_j}{\partial (y_m)^2}(\mathbf{y}^i) = -\frac{(l+3)(l+4)}{c^2}\left(1 - \frac{d_{ij}}{c}\right)_+^{l}\\ &\times\left(\left(1 - \frac{d_{ij}}{c}\right)_+\left((l+1)\frac{d_{ij}}{c} + 1\right) - \frac{(y_m^i - y_m^j)^2}{c^2}(l+1)(l+2)\right).\end{aligned}$$

For the sake of obtaining better accuracies as will be shown by numerical experiments, we use the $\phi_{5,2}$ for higher dimensions up to 4D. The formulation of $\phi_{5,2}$ was given in Table 1. Similarly, the matrices P and P''_m are symmetry while P'_m is skew-symmetry.

By denoting $A = \mathbf{\Phi} P^{-1}$, where

$$\mathbf{\Phi} = \frac{1}{2}\sum_{m=1}^{M} D_{mm} P''_m + \sum_{m=1}^{M}\left(\sum_{l=1}^{M}\delta_l c_{ml}\right) P'_m - rP$$

and $u^n = u(n\varDelta\tau)$, where $\varDelta\tau = \frac{T}{N_\tau}$, $0 \le n \le N_\tau$ for a given number of time steps N_τ, then the numerical solution can be found similarly by (4.9).

Table 2. Several notations used in the paper and their descriptions based on the computation of the shape parameter.

Notation	Description
$V_R(\mathbf{S}^*, T)$	Price using Wendland scheme with (3.4)
$V_H(\mathbf{S}^*, T)$	Price using Wendland scheme with (3.3)
$V_{MH}(\mathbf{S}^*, T)$	Price using Wendland scheme with (3.5)
$V_{WTE}(\mathbf{S}^*, T)$	Price using Wendland scheme with trail-and-error
$V_G(\mathbf{S}^*, T)$	Price using Gaussian scheme with guessing
$V_{GTE}(\mathbf{S}^*, T)$	Price using Gaussian scheme with trail-and-error
$V_{FD(E)}(\mathbf{S}^*, T)$	Price using FD with equidistant grid [16]
$V_{FD(NE)}(\mathbf{S}^*, T)$	Price using FD with non-equidistant grid [16]
$V_{KM}(\mathbf{S}^*, T)$	Price using FD with equidistant grid [26]
$\text{Size}_{FD(E)} = N = 2^{3n}$	Size of the equidistant grid with FD for 3D
$\text{Size}_{FD(NE_3)} = N = 16(2^n) \times 4(2^n) \times 4(2^n)$	Size of the non-equidistant grid with FD for 3D
$\text{Size}_{FD(NE_4)} = N = 16(2^n) \times 4(2^n) \times 4(2^n) \times 4(2^n)$	Size of the non-equidistant grid with FD for 4D

6 Computational experiments

In this section, the RBF strategy is employed for solving high-dimensional option pricing problems by constructing the corresponding semi-discretized systems of ODEs.

We compare the results of the proposed Wendland RBF approach with competitive existing alternative methods up to 4D. It is also required to remark that most of the existing numerical reports in the literature for three assets or more are mostly based on the parallelization technique of sparse grid discretization, while our approach is quite direct and without any parallelization. This means that the computation of our new modified RBF schemes with a good shape parameter and parallelization could even yield in much more quicker results.

In this section, κ_P and κ_A stand for the condition numbers of the interpolation and the coefficient matrices, respectively.

Here we check and illustrate that meshfree RBF methods are competitive with respect to than the FD method for option pricing problems. They can achieve equal or better accuracy by using fewer number of nodes in contrast to the FD methods. In addition, it is confirmed by experiments that global RBFs depend too much to the choice of a good shape parameter and they tend not to provide useful numerical results since the evaluation of the coefficient matrices become so ill-conditioned. On the contrary, the local Wendland RBF is illustrated to be efficient for option pricing problem whenever the shape parameter is chosen by (3.5).

Throughout the implementation of our meshfree RBF method, we do not impose any boundaries and in fact we consider that the discretized equations at the boundaries are the best approximations for nodes located on them. In our simulations we apply an equidistant grid of nodes for the original coordinates which after incorporating the transformations would not be anymore equidistant in the final computational domain. In addition, several notations which will be used later are provided in Table 2.

Example 1. In this experiment, we consider one-asset option with the following parameters:

$$r = 0.1,\ \sigma = 0.2,\ T = 1,\ q = 0,\ E = 10. \tag{6.1}$$

Results for the European and American put option pricing problems are reported in Table 3.

Table 3. Numerical results in 1D case for European and American option.

N	European ($\lambda = 0$)			American ($\lambda = 100$)		
	RMSE	min	max	RMSE	min	max
16	0.1567	-1.3298	-0.1	0.1595	-1.3359	-0.1949
31	0.0782	-6.0621	-0.1	0.0382	-6.0613	-0.2047
61	0.0211	-23.9371	-0.1	0.0092	-23.9379	-0.1915
121	0.0058	-95.9372	-0.1	0.0023	-95.9379	-0.1883
241	0.0018	-383.9373	-0.1	0.0005	-383.9379	-0.1875
481	0.0009	-1535.9	-0.1	0.0001	-1535.9	-0.1873

First of all, we consider the computational domain $[-5, 1] \times [0, T]$ with the spatial uniform grid and various step sizes h. The parameter c is chosen to be 1. We consider European ($\lambda = 0$) and American ($\lambda = 100$) options. In Table 3 the root mean square error (RMSE) and maximum and minimum of real part of eigenvalues of matrix A are reported.

Note that for computing RMSE for European option, the exact solution is used as the reference value. Since the exact solution of an American option pricing problem is not known, as the reference value to the solution on the refined grid ($N = 961$) is used. In both cases of European and American cases, real parts of eigenvalues are strongly negative.

Table 4. Accuracy of the solution and matrix properties depending on the parameter c.

c	RMSE	min	max
0.25	1.2265	-96.0475	-0.05
0.5	0.0838	-95.9739	-0.05
1	0.0070	-95.9781	-0.05
1.5	0.0028	-95.9823	-0.05
2	0.0019	-95.9849	-0.05
4	0.0014	-96.0024	-0.05
12	0.0009	-95.9894	-0.05
12.5	0.0076	-95.9894	-0.05

In Table 4 the results for another interest rate ($r = 0.05$) with the remaining parameters from (6.1) and $\lambda = 0$ (European option) are presented by varying the shape parameter c. As it is seen from the results, the accuracy is increasing with increasing c until certain value ($c = 12$). For higher values of c, the accuracy becomes decreasing.

Example 2. The American basket put option of two assets is considered with the following parameters and zero dividend yields [2]:

$$\sigma_1 = 0.3,\ \sigma_2 = 0.2,\ r = 0.05,\ \rho = 0.6,\ \omega_1 = 0.7,\ \omega_2 = 0.3,\ T = 1,\ E = 50.$$

As a reference value 3.9751 at the point $\mathbf{S}^* = (50, 50)$, the result of the binomial tree method of [2] is used. The numerical solution is presented in Figure 1.

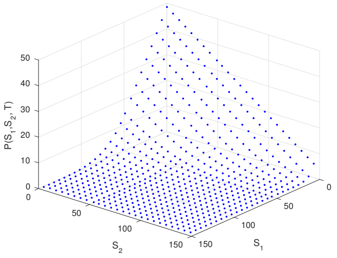

Figure 1. Numerical solution of the 2D-basket put option.

Moreover, the comparison of C^2 and C^4 Wendland's functions is presented in Table 5 for the fixed time step $\varDelta\tau = 0.01$ and several numbers N of centers, denoted by n^2 of the grid. Here, the good shape parameter for each set of parameters is found by using Rippa's algorithm. The CPU time in seconds is presented to prove the efficiency of the proposed method. In the experiment, the specifications of the PC are CPU Intel Core i5 (4th Gen) 4310U / 2.00GHz × 4 , 8GB RAM.

It can be observed from Table 5 that although C^2 functions are enough for solving our option pricing problems but for the sake of having better accuracies with fewer number of nodes, we can apply C^4 Wendland function for higher dimensions.

Table 5. Results of comparisons for Example 2.

Size (n^2)	C^2			C^4		
	$V_R(\mathbf{S}^*, T)$	c	CPU	$V_R(\mathbf{S}^*, T)$	c	CPU
25	3.69	34.04	0.31	3.72	28.84	0.41
100	3.97	44.18	0.71	3.97	34.26	0.76
400	3.98	54.78	4.89	3.98	38.76	4.92

The Rippa's algorithm (3.4) works based on minimizing a function which its output is a number but one of its argument is the computation of a Moore-Penrose inverse. Due to this, for high-dimensional cases, or the cases at which the total number of nodes are so many, e.g. more than 4000, the use of the algorithm is useless due to failure by making the kernel shut down or further tremendous computational time. As a result, using (3.4) for the 3D and 4D cases is not recommended and we restrict the use of (3.4) up to the 2D time-dependent PDEs. On the other hand, we focus on the use of Hardy's algorithm (3.3) and its modification (3.5) which provide quite good results for our high-dimensional PDEs.

In the following experiments, we compare the results of different methods for the 3D equation in terms of European call pricing. For computational purposes, we consider the original computational domain to be $[1, 3E]^M$. This domain is discretized uniformly which will not be uniform in the final transformed computational domain. In all further experiments, the specifications of the PC are four Intel Xeon processors E5-4600 and a memory capacity of up to 1.5 terabytes via 48 memory sockets in a dense 2U rack system.

Table 6. Comparison of different schemes for valuing 3D option pricing problem at the point $(\mathbf{S}^*, T)$.

H and MH				EFDM			NEFDM		
n	Size	V_H	V_{MH}	n	Size	V	n	Size	V
12	1728	13.1973	13.1950	3	2^9	12.862	1	2^{11}	13.098
16	4096	13.2163	13.2211	4	2^{12}	13.150	2	2^{14}	13.207
20	8000	13.2277	13.2341	5	2^{15}	13.221	3	2^{17}	13.236
24	13824	13.2331	13.2398	6	2^{18}	13.239	4	2^{20}	13.243
28	21952	13.2358	13.2425	7	2^{21}	13.243			

Example 3. We consider the European basket call option with no dividends and the following parameters (see [16, p. 76])

$$\sigma_1 = 0.3, \ \sigma_2 = 0.35, \ \sigma_3 = 0.4, \ r = 0.04, \ \rho_{ij} = 0.5, \ \omega_i = \frac{1}{3}, \ T = 1, \ E = 100.$$

The results are presented in Table 6 comparing RBF method with Hardy (H) and modified Hardy (MH) shape parameters and FD methods of [16, chapter 3] with equidistant (EFDM) and non equidistant grids (NEFDM). The reference solution value is 13.245 computed by using accurate FFT technique [16, chapter 4]. The $V_{FD(NE)}(\mathbf{S}^*, T)$ is based on the sparse grid solution mimics and no *stretching.* The numerical solution in this case is brought forward in Figure 2 using the Wendland RBF semi-discretization and modified Hardy's technique by applying $n = 12$ uniform nodes in each dimension. The graph shows an stable behavior of RBF application for pricing multi-dimensional problems. The computational domain for this test is $[1, 3E] \times [1, 3E] \times [1, 3E]$.

Table 7. Behavior of the Gaussian RBF for valuing 3D option pricing problem.

n	Size (n^3)	κ_P	κ_A	$V_G(\mathbf{S}^*, T)$
5	125	54.6	19.8	5.4628
6	216	879.7	42.6	9.42048
7	343	17847.5	80.2	10.3154
8	512	470427.0	331.6	13.3034
9	729	1.62×10^7	1797.7	12.3784
10	1000	7.73×10^8	13936.8	13.485
11	1331	4.84×10^{10}	197295.0	12.7355
12	1728	3.66×10^{12}	1.73×10^7	13.3328
13	2197	1.11×10^{14}	2.18×10^9	-1.48×10^6
14	2744	1.82×10^{15}	4.29×10^{12}	2.99×10^{35}

Table 7 contains the convergence history of the meshfree Gaussian RBF with $c = 1$ for pricing the 3D option which clearly shows that ill-conditioning result in the blowup and divergence of the numerical scheme.

Practically speaking, it could be observed from Tables 6–7 that meshfree local Wendland RBF method with the choice of a good shape parameter extracted from (3.5) can converge to the true solution efficiently. Furthermore, the total number of nodes for converging (the size of the semi-discretized system of ODEs) is quite lower than the size of the systems obtained in the process of applying finite difference schemes.

Table 8. Convergence history of the Gaussian RBF for valuing 3D option pricing problem using trial and error.

n	Size (n^3)	Optimal shape parameter	κ_P	κ_A	$V_{GTE}(\mathbf{S}^*, T)$
4	64	2.48433	1182.2	23.0	13.245
5	125	2.17781	237760.0	122.2	13.245
6	216	2.39314	4.97×10^8	20249.9	13.245

In Tables 8–9, we report the results of pricing the Example 3 in a very efficient way by applying only a few nodes which resulted in a system of ODE of low size with an optimal selection of the shape parameter. This can only be derived by employing the trial-and-error strategy discussed in [5, p. 142].

To discuss further on the other aspects, when we discretize in the spatial variables throughout this section we can check that all the eigenvalues have

Table 9. Convergence history of the Wendland RBF for valuing 3D option pricing problem using trial and error.

n	Size (n^3)	Optimal shape parameter	κ_P	κ_A	$V_{WTE}(\mathbf{S}^*, T)$
3	27	13.1949	195.0	9.2	13.245
4	64	9.662	7130.2	55.4	12.666
5	125	6.87231	14326.6	92.3	13.245
6	216	138.361	2.01×10^{12}	66765.2	13.245

negative real parts which ensure the minimum requirements for the stability of the time-stepping ODE solvers.

The numerical solution is presented in Figure 2.

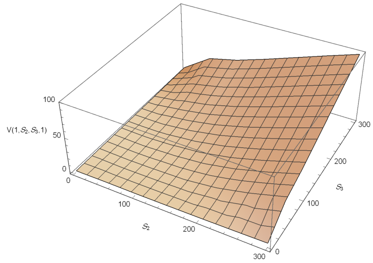

Figure 2. Numerical solution of the 3D-basket call option via modified Hardy's technique and Wendland RBF in Example 3.

Example 4. We test different methods for 3-dimensional American put option using the following data with no dividends and the following parameters [1]

$$\sigma_1 = 0.4,\ \sigma_2 = 0.4,\ \sigma_i = 0.4,\ r = 0.05,\ \omega_i = 1,\ T = 1,$$

where the correlation matrix is

$$R = \begin{pmatrix} 1.0 & 0.3 & 0.5 \\ 0.3 & 1.0 & -0.3 \\ 0.5 & -0.3 & 1.0 \end{pmatrix},$$

with the strike price $E = 300$.

The results are provided in Table 10 at $\mathbf{S}^* = (100, 100, 100)^T$ using the reference solution 26.213 provided by a very refined grid [1]. Note also that we considered $\lambda = 1000$ and $\Delta\tau = 2.5 \times 10^{-3}$ while the computational domains for the Yousuf et al. method [26] is $[0, 2.5E] \times [0, 2.5E] \times [0, 2.5E]$ and for the proposed modified Wendland's method is $[1, 2.5E] \times [1, 2.5E] \times [1, 2.5E]$. As can be seen from the results, the RBF Wendland scheme using the modified Hardy algorithm converge to the reference solution while the finite difference technique of [26] requires more number of nodes.

Table 10. Convergence history of different methods for valuing 3D American option pricing problem.

n	Size (n^3)	$V_{MH}(\mathbf{S}^*,T)$	$V_{KM}(\mathbf{S}^*,T)$
16	4096	25.656	24.425
18	5832	26.453	25.916
20	8000	26.427	26.465
22	10648	26.362	26.592
24	15552	26.234	26.421
26	17576	26.204	26.199

Example 5. In this experiment, we compare the results of applying the meshfree Wendland function for solving the four-dimensional time-dependent PDE (2.4) with the following inputs and zero dividend yields (see [16, p. 76])

$$\sigma_1 = 0.3,\ \sigma_2 = 0.35,\ \sigma_3 = 0.4,\ \sigma_4 = 0.45,\ r = 0.04,\ \omega_i = \frac{1}{4},\ T = 1,\ E = 100,$$

while the correlation matrix is

$$R = \begin{pmatrix} 1 & 0.5 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 & 0.5 \\ 0.5 & 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 0.5 & 1 \end{pmatrix}.$$

The results of comparisons in this case are given in Table 11 and we used (3.5) for constructing the constant shape parameter to the reference solution 13.659. The computational domain is $[1, 3E]^4$. Based on Table 11 numerous nodes are required even with a non-equidistant nodal points in the parallelized FD method to converge while with fewer number of nodes, we obtained good results based on our proposed RBF procedure.

Table 11. Comparison of different schemes for valuing 4D option pricing problem at the point $(\mathbf{S}^*, T)$.

n	Size (n^4)	$V_H(\mathbf{S}^*,T)$	V_{MH}	n	$Size_{FD(NE_4)}$	$V_{FD(NE)}(\mathbf{S}^*,T)$
6	1296	13.4084	13.4167	1	2^{14}	13.672
8	4096	13.6905	13.7458	2	2^{18}	13.662
10	10000	13.5984	13.6101	3	2^{22}	13.661
12	20736	13.6756	13.6834	4	2^{26}	13.659
14	38416	13.6437	13.6559			

Results in Table 11 once again show that the meshfree local Wendland RBF method along with the technique (3.5) is a good choice for solving even 4D time-dependent option pricing problems. The numerical solution in this case is brought forward in Figure 3 using the Wendland RBF semi-discretization and modified Hardy's technique by applying $n = 7$ uniform nodes in each dimension. The graph shows an stable behavior of RBF application for pricing multi-dimensional problems. In Figures 2–3, the main point is to show that the numerical solution even for the 3D and 4D cases do not have any spurious or instability. In Figure 2 the solution of the 3D problem is plotted by fixing $S_1 = 1$ and in Figure 3 the solution of the 4D problem is plotted for $S_1 = S_2 = 1$. In both cases the current date $\tau = T = 1$ is considered.

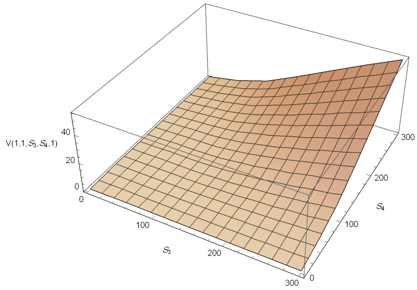

Figure 3. Numerical solution of the 4D case via modified Hardy's technique and Wendland RBF in Example 5.

Apart from the efficiency of the new RBF method, an equally important matter is the accuracy of the proposed procedures for pricing options which can simply be seen in the solved examples.

Once again by applying the strategy of trial-and-error for the Gaussian and Wendland RBFs, we may obtain the optimal value of the shape parameter for pricing 4D option pricing PDE as given in Tables 12–13. The numerical results using this strategy are promising and fast due to considering a few nodes in each direction. This suggests that our RBF implementations with a choice of a good shape parameter can even be used for pricing high-dimensional time-dependent PDEs, e.g., the 5D option pricing problems.

Table 12. Convergence history of the Gaussian RBF for valuing 4D option pricing problem using trial and error.

n	Size (n^4)	Optimal shape parameter	κ_P	κ_A	$V_{GTE}(\mathbf{S}^*, T)$
3	81	3.58671	1831.02	9.70313	13.659
4	256	2.25897	46819.2	28.2716	13.659
5	625	2.11515	3.06×10^7	909.193	13.659

Overall, it could be observed from the computational comparisons that the new local Wendland RBF scheme along with the semi-discretization technique and modified Hardy's technique for the shape parameter is competitive for pricing multi-dimensional time-dependent PDE problems.

7 Conclusions

The aim of this paper has been the construction of competitive high-dimensional RBF methods for valuation of nonlinear multiasset options problems. This

Table 13. Convergence history of the Wendland RBF for valuing 4D option pricing problem using trial and error.

n	Size (n^4)	Optimal shape parameter	κ_P	κ_A	$V_{WTE}(\mathbf{S}^*, T)$
3	81	14.3889	1926.87	18.0281	13.659
4	256	15.4241	367008.	315.704	13.659
5	625	8.34669	136911.	202.963	13.659

has been performed by combining two strategies. First, by using an elimination of the cross-derivative terms, we have saved the computational cost and amount of memory involved in the calculation of the RBF interpolation task. And second, we have used the Wendland compact support functions allowing stability when the dimension and the number of grid points grow.

As the accuracy of the RBF method is very sensible to the choice of the shape parameter, we have suggested here an improvement of the Hardy's technique (3.5) where the number of centers considered in the support domain grows linearly with the dimension.

These advantages have been illustrated with numerical examples comparing the results with other acknowledged existing methods for higher dimensions. We can conclude that the proposed procedure can also be used for dimensions higher than four when the choice of a good shape parameter is available.

Acknowledgement

We wish to express our gratitude to anonymous reviewers for their helpful comments which improved the quality of the paper.

References

[1] J. Benk, H.-J. Bungartz, A.-E. Nagy and S. Schraufstetter. Variants of the combination technique for multi-dimensional option pricing. *Progress in Industrial Mathematics at ECMI 2010*, **17**:231–237, 2012. https://doi.org/10.1007/978-3-642-25100-9_27.

[2] S. Borovkova, F. Permana and J. V. der Weide. American basket and spread option pricing by a simple binomial tree. *The J. Derivatives*, **19**(4):29–38, 2012. https://doi.org/10.3905/jod.2012.19.4.029.

[3] S. Choi and M. Marcozzi. A numerical approach to American currency option valuation. *The J. Derivatives*, **9**(2):19–29, 2001. https://doi.org/10.3905/jod.2001.319172.

[4] R. Company, V.N. Egorova, L. Jódar and F. Soleymani. A mixed derivative terms removing method in multi-asset option pricing problems. *Appl. Math. Lett.*, **60**:108–114, 2016. https://doi.org/10.1016/j.aml.2016.04.011.

[5] G. Fasshauer. *Meshfree Approximation Methods with MATLAB*. World Scientific Publishing Co., Singapore, 2007. https://doi.org/10.1142/6437.

[6] G. Fasshauer and J. Zhang. On choosing optimal shape parameters for RBF approximation. *Numer. Algor.*, **45**(1–4):345–368, 2007. https://doi.org/10.1007/s11075-007-9072-8.

[7] B. Fornberg and N. Flyer. Solving PDEs with radial basis functions. *Acta Numerica*, **24**:215–258, 2015. https://doi.org/10.1017/S0962492914000130.

[8] P. Glasserman. *Monte Carlo Methods in Financial Engineering.* Springer-Verlag, New York, NY, USA, 2004.

[9] A. Golbabai, D. Ahamadian and M. Milev. Radial basis functions with application to finance: American put option under jump diffusion. *Math. Comput. Modell.*, **55**(3–4):1354–1362, 2012. https://doi.org/10.1016/j.mcm.2011.10.014.

[10] H. Gzyl, M. Milev and A. Tagliani. Discontinuous payoff option pricing by Mellin transform: A probabilistic approach. *Finance Res. Lett.*, **20**:281–288, 2017. https://doi.org/10.1016/j.frl.2016.10.011.

[11] R. Hardy. Multiquadric equations of topography and other irregular surfaces. *J. Geophys. Res.*, **176**(8):1905–1915, 1971. https://doi.org/10.1029/JB076i008p01905.

[12] Y. Hon and X. Mao. A radial basis function method for solving options pricing models. *Finan. Eng.*, **8**:31–49, 1999.

[13] S. U. Islam and I. Ahmad. A comparative analysis of local meshless formulation for multi-asset option models. *Eng. Anal. Bound. Elem.*, **65**:159–176, 2016. https://doi.org/10.1016/j.enganabound.2015.12.020.

[14] P. Lancaster and M. Tismenetsky. *The Theory of Matrices with Applications.* Academic Press, USA, 2nd ed. edition, 1985.

[15] E. Larsson, A. Krister and H. Andreas. Multi-dimensional option pricing using radial basis functions and the generalized Fourier transform. *J. Comput. Appl. Math.*, **222**(1):175–192, 2008. https://doi.org/10.1016/j.cam.2007.10.039.

[16] C. Leentvaar. *Pricing multi-asset options with sparse grids.* PhD thesis, TU Delft, 2008.

[17] M. Mongillo. Choosing basis functions and shape parameters for radial basis function methods. Technical report, SIAM Undergraduate Research Online (SIURO), 2011.

[18] A. Pascucci. *PDE and Martingale Methods in Option Pricing.* Springer-Verlag, Italia, 2011. https://doi.org/10.1007/978-88-470-1781-8.

[19] U. Pettersson, E. Larsson, G. Marcusson and J. Persson. Improved radial basis function methods for multi-dimensional option pricing. *J. Comput. Appl. Math.*, **222**(1):82–93, 2008. https://doi.org/10.1016/j.cam.2007.10.038.

[20] C. Reisinger and G. Wittum. Efficient hierarchial approximation of high-dimensional option pricing problems. *SIAM J. Sci. Comput.*, **29**(1):440–458, 2007. https://doi.org/10.1137/060649616.

[21] S. Rippa. An algorithm for selecting a good value for the parameter c in radial basis function interpolation. *Adv. Comput. Math.*, **11**(2–3):193–210, 1999. https://doi.org/10.1023/A:1018975909870.

[22] V. Shcherbakov and E. Larsson. Radial basis function partition of unity methods for pricing vanilla basket options. *Comput. Math. Appl.*, **71**(1):185–200, 2016. https://doi.org/10.1016/j.camwa.2015.11.007.

[23] D. Tavella. *Quantitative methods in derivatives pricing.* John Wiley & Sons, California, 2012.

[24] L. von Sydow, L.J. Höök, E. Larsson, E. Lindström, S. Milovanović, J. Persson, V. Shcherbakov, Y. Shpolyanskiy, S. Sirén, J. Toivanen, J. Waldén, M. Wiktorsson, J. Levesley, J. Li, C.W. Oosterlee, M.J. Ruijter, A. Toropov and Y. Zhao. BENCHOP – The BENCHmarking project in option pricing. *Int. J. Comput. Math.*, **92**(12):2361–2379, 2015. https://doi.org/10.1080/00207160.2015.1072172.

[25] H. Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Adv. Comput. Math.*, **4**(1):389–396, 1995. https://doi.org/10.1007/BF02123482.

[26] M. Yousuf, A.-Q.M. Khaliq and R. Liu. Pricing American options under multi-state regime switching with an efficient L-stable method. *Int. J. Comput. Math.*, **92**(12):2530–2550, 2015. https://doi.org/10.1080/00207160.2015.1071799.

[27] S. Zhu. Compactly supported radial basis functions: how and why? OCCAM report. Technical report, The Mathematical Institute, University of Oxford, 2006.

[28] H.B. Zubair, C.W. Oosterlee and R. Wienands. Multigrid for high-dimensional elliptic partial differential equations on non-equidistant grids. *SIAM J. Sci. Comput.*, **29**(4):1613–1636, 2007. https://doi.org/10.1137/060665695.

[29] R. Zvan, P. Forsyth and K. Vetzal. Negative coefficients in two-factor option pricing models. *J. Comput. Finance*, **7**(1):37–73, 2003. https://doi.org/10.21314/JCF.2003.096.