Comput Econ
DOI 10.1007/s10614-017-9739-8

Radial Basis Functions with Partition of Unity Method for American Options with Stochastic Volatility

Reza Mollapourasl[1,3] · **Ali Fereshtian**[1] · **Michèle Vanmaele**[2]

Accepted: 30 August 2017

Abstract In this article, we price American options under Heston's stochastic volatility model using a radial basis function (RBF) with partition of unity method (PUM) applied to a linear complementary formulation of the free boundary partial differential equation problem. RBF-PUMs are local meshfree methods that are accurate and flexible with respect to the problem geometry and that produce algebraic problems with sparse matrices which have a moderate condition number. Next, a Crank–Nicolson time discretisation is combined with the operator splitting method to get a fully discrete problem. To better control the computational cost and the accuracy, adaptivity is used in the spatial discretisation. Numerical experiments illustrate the accuracy and efficiency of the proposed algorithm.

Keywords Radial basis function · Partition of unity · Operator splitting · American option pricing · Stochastic volatility · Heston's model

✉ Reza Mollapourasl
mollapour@srttu.edu

Ali Fereshtian
a.fereshtian@srttu.edu

Michèle Vanmaele
michele.vanmaele@ugent.be

1 School of Mathematics, Shahid Rajaee Teacher Training University, Lavizan, Tehran 16788, Iran

2 Department of Applied Mathematics, Computer Science and Statistics, Ghent University, Krijgslaan 281-S9, 9000 Ghent, Belgium

3 Department of Mathematics, Oregon State University, Corvallis, OR 97331, USA

Published online: 07 September 2017

1 Introduction

The arbitrage free price of an option is determined as the analytical solution to the Black–Scholes equation introduced by Black and Scholes (1973) and extended by Merton (1973) when the underlying asset price is modelled as a geometric Brownian motion and all parameters are assumed to be constant. However, more advanced valuation models involve stochastic parameters such as interest rate and volatility. This leads to more sophisticated option pricing models. The corresponding partial differential equation (PDE) has no analytic solution and has to be solved numerically.

In this paper, we consider the option pricing problem when the volatility is stochastic. Stochastic volatility was first introduced to the traditional model in the 1980s, when Hull and White (1987) and others generalised the Black–Scholes model in order to get a better match with market prices. The model presented by Heston (1993) is the most popular one. While European options under the Heston model can be modelled by a parabolic PDE which can be solved using analytical expressions, American options result in additional inequality constraints and more sophisticated numerical methods are called for.

Many approaches to numerically solve the option pricing PDE under the Heston model are discussed in the literature. Zvan et al. (1998) viewed this problem for an American option as a nonlinear PDE, where the early exercise constraint can be imposed using a penalty method. They combine a finite element and a finite volume approach. The resulting system of nonlinear algebraic equations is solved using a Newton iteration. Apel et al. (2001) propose also a finite element approach for pricing vanilla and knock-out options while Clarke and Parrott (1999) and Oosterlee (2003) price American-style options using multigrid methods. Ikonen and Toivanen (2008, 2009) use operator splitting methods for performing time stepping after a finite difference space discretisation. The idea is to decouple the treatment of the early exercise constraint and the solution of the system of linear equations into separate fractional time steps. Haentjens and in 't Hout (2015) use alternating direction implicit (ADI) finite difference schemes to price American options under the Heston model. O'Sullivan and O'Sullivan (2013) consider Super-Time-Stepping accelerated explicit finite differencing methods.

During the past decade, increasing attention has been given to the development of meshfree methods using radial basis functions (RBFs) for the numerical solution of PDEs. The main advantages of these methods lie in their simplicity, their applicability to various PDEs, and their effectiveness in dealing with high-dimensional problems with complicated geometries since no tedious mesh generation is needed. Also for option pricing problems in one and two spatial dimensions, meshfree methods based on RBF approximations have been shown to perform better than finite difference methods (Ballestra and Pacelli 2013; Fasshauer et al. 2004). Forward Kolmogorov problems have been solved by Ballestra and Pacelli (2011, 2012) with promising results. However, the global RBF collocation methods have some disadvantages. Estimates of condition numbers of the matrices in the resulting dense linear systems indicate that the meshless method using RBFs may be unstable (Duan 2008). Also the computational cost is a main obstacle when using global RBF methods, as stated by Larsson et al. (2013).

Localised RBF approximations such as the RBF partition of unity collocation method (RBF-PUM) give an answer to deal with these issues. The matrices formed during the localised method will be sparse and, hence, will not suffer from ill-conditioning and high computational costs. PU methods for solving PDEs were studied by Babuška and Melenk (1997), combined with RBFs by Wendland (2002), implemented for interpolation on the sphere, in the plane, and in a three-dimensional space by Cavoretto and Rossi (2012, 2014). The capability of RBF-PUM for numerically solving parabolic (time-dependent) PDEs is discussed by Safdari-Vaighani et al. (2015) and illustrated for two-asset American option pricing in a Black-Scholes setting. Shcherbakov and Larsson (2016) compare the performance of the global RBF and the RBF-PUM to the finite difference penalty method for European and American double-asset options also under the Black–Scholes model.

The *contribution of this paper* is the development of an RBF-PUM based algorithm to numerically price American options under Heston's model and the study of the accuracy and efficiency of the proposed algorithm. Hereto we first apply a coordinate stretching transformation to the asset price and variance and formulate the PDE in this new coordinate system. In this way the computation of the prices can be focused on regions of real interest instead of on the whole solution domain. Then, the RBF-PUM is combined with an operator splitting technique that is applied to the Heston linear complementarity problem (LCP) obtained after space and time discretisation of the resulting PDE. To further improve the efficiency a non-uniform mesh for the spatial discretisation is used. The outline of the paper is as follows.

In the next section, we describe Heston's model and a partial differential equation for pricing European options. In Sect. 3, we formulate the American option pricing problem under the Heston model after a coordinate transformation as a linear complementary problem. The basics of radial basis functions and the spatial discretisation of the domain as well as the partition of unity method based on RBFs and the definition of the differentiation matrices are introduced and applied to the problem under consideration in Sect. 4. In Sect. 5, a time discretisation of the semidicrete LCP is combined with the operator splitting method. Finally, in the last section, the accuracy and efficiency of the proposed method is numerically investigated for European and American put options.

2 European Option Pricing Under Heston's Model

In Heston's stochastic volatility model the price process S of an asset and its variance process V have the following risk-neutral dynamics:

$$dS_t = (r - q)S_t dt + S_t\sqrt{V_t}dW_t^1 \tag{1}$$

$$dV_t = \kappa(\theta - V_t)dt + \sigma\sqrt{V_t}dW_t^2 \tag{2}$$

where r is the risk-neutral interest rate and q is the dividend yield of the asset price. The variance process V is a Cox–Ingersoll–Ross (CIR) model (Cox et al. 1985) which is a square root mean reverting process with rate of mean reversion $\kappa > 0$, long-run

mean $\theta > 0$ and volatility $\sigma > 0$. The Wiener processes W^1 and W^2 are correlated with correlation parameter $\rho \in [-1, 1]$.

By the Markov property, the fair value of a vanilla *European* (call or put) option at time t is denoted $H(t, s, v)$ if the asset price at that time is $S_t = s$ and the variance at that time is $V_t = v$. This function $H(t, s, v)$ is defined in an unbounded domain $\{(t, s, v) \mid s, v \geq 0, t \in [0, T]\}$ where T is the time of maturity and is a solution to the following parabolic PDE with convection, diffusion and reaction terms:

$$\frac{\partial H}{\partial t} + (\mathscr{L}_H - r)H = 0, \tag{3}$$

where the Heston spatial differential operator $\mathscr{L}_H$ applied to $H(t, s, v)$ is denoted by

$$\mathscr{L}_H H = \frac{1}{2}\sigma^2 v \frac{\partial^2 H}{\partial v^2} + \rho\sigma v s \frac{\partial^2 H}{\partial s \partial v} + \frac{1}{2} s^2 v \frac{\partial^2 H}{\partial s^2} + \kappa(\theta - v)\frac{\partial H}{\partial v} + (r - q)s\frac{\partial H}{\partial s}. \tag{4}$$

At maturity T, the value of the option equals the payoff. Introducing the *payoff function* for a strike price K defined by

$$P(s) = \max(\xi(s - K), 0) \tag{5}$$

with $\xi = +1$ for a call and $\xi = -1$ for a put, this terminal condition reads

$$H(T, s, v) = P(s). \tag{6}$$

In the remaining of the paper we will focus on put options, thus $\xi = -1$. As additional boundary conditions for H we impose[1]

$$\lim_{s \to 0} H(t, s, v) = K\mathrm{e}^{-r(T-t)}, \quad \lim_{s \to \infty} H(t, s, v) = 0 \tag{7}$$

$$\lim_{v \to \infty} \frac{\partial H(t, s, v)}{\partial v} = 0. \tag{8}$$

In several papers the Neumann boundary condition

$$\lim_{s \to \infty} \frac{\partial H(t, s, v)}{\partial s} = 0$$

is imposed instead of the second boundary condition in relation (7). The issue of appropriate boundary and end conditions is still subject to discussions, see e.g. Kunoth et al. (2012) and the references therein. Whether or not to impose a boundary condition for $v = 0$ depends on the parameter values in the variance process V in (2). When $2\kappa\theta \geq \sigma^2$, the Feller condition is satisfied and the boundary $v = 0$ is not attainable. Hence no boundary condition is imposed in that case. However, in many market

[1] Boundary conditions in case of a call option, i.e. $\xi = 1$, can be found in Apel et al. (2001).

situations this Feller condition is violated and a boundary condition is needed. Apel et al. (2001) impose the original Heston's boundary condition

$$\frac{\partial H}{\partial t}(t,s,0)+\kappa\theta\frac{\partial H}{\partial v}(t,s,0)+(r-q)s\frac{\partial H}{\partial s}(t,s,0)-rH(t,s,0)=0.$$

Zhu and Chen (2010), use

$$\lim_{v\to 0}H(t,s,v)=\max(K\mathrm{e}^{-r(T-t)}-s,0).$$

Further they argue that the Neumann boundary condition (8) can be converted into a Dirichlet boundary condition

$$\lim_{v\to\infty}H(t,s,v)=K\mathrm{e}^{-r(T-t)}.$$

Heston (1993) found a closed-form exact solution for the price of European-style options. Unfortunately, the approach he adopted could not be easily extended to the case of American options; no analytical solution for American options under the Heston model has been discovered yet. In the next section we will give a PDE formulation to price an American put option under Heston's model.

3 American Option Pricing Under Heston's Model

3.1 Free Boundary Problem

An American option has the early exercise feature. When the interest rate is non-zero, it is always optimal to exercise an American put even when the underlying asset pays no dividend,[2] see e.g. Kwok (1998). The optimal exercise boundary is a free boundary and separates the stopping and continuation region. Let $H(t,s,v)$ denote the fair value of an *American* put option at time t if the asset price at that time is $S_t=s$ and the variance at that time is $V_t=v$. Then, in the continuation region the option is held and this function $H(t,s,v)$ is a solution to the PDE (3) while in the stopping region the price equals the payoff function, i.e. $H(t,s,v)=P(s)$ and the option is exercised. To solve the PDE (3) for the valuation of American puts, a set of appropriate boundary conditions is needed together with the terminal condition (6). Those are similar to those for the European option and a similar discussion can be held, see (Zhu and Chen 2011):

$$\begin{aligned}&\lim_{s\to 0}H(t,s,v)=\lim_{s\to 0}P(s)=K,\quad \lim_{s\to\infty}H(t,s,v)=\lim_{s\to\infty}P(s)=0\\&\lim_{v\to 0}H(t,s,v)=P(s),\quad \lim_{v\to\infty}\frac{\partial H(t,s,v)}{\partial v}=0.\end{aligned}\tag{9}$$

Zhu and Chen (2011) further argue that the boundary condition

[2] It is well-known that an American call option equals a European call option when the underlying asset pays no dividend.

$$\lim_{v \to 0} H(t, s, v) = 0$$

is a simplified version of the corresponding boundary condition in (9) and can be imposed regardless of what the ratio $2\kappa\theta/\sigma^2$ is.

3.2 Coordinate Transformation

To apply numerical techniques, we replace the unbounded domain $\{(t, s, v) \mid s, v \geq 0, t \in [0, T]\}$ with a bounded one $[0, T] \times [0, S_{\max}] \times [0, V_{\max}]$ where the values $S_{\max}$ and $V_{\max}$, will be chosen based on standard financial arguments, such that the error caused by truncating the solution domain is negligible. The payoff function (5) is a non-smooth function, in particular its derivative is discontinuous at the strike price K. Therefore, to reduce the loss of accuracy in the spatial discretisation a local grid refinement near the corner $s = K$ is appropriate. Along the v-direction, we want the distribution of the grid points to be more dense in a neighbourhood of $v = V_0$, where the possible realisations of the variance process are more likely to occur or near the degenerate boundary $v = 0$. So we employ the following analytic coordinate transformation of the physical domain $\Omega = [0, S_{\max}] \times [0, V_{\max}]$ into the square $[0, 1] \times [0, 1]$ and which results in an a-priori stretching of the grid:

$$\begin{aligned} x(s) &= \frac{\sinh^{-1}(\zeta_s(s-k)) + \sinh^{-1}(\zeta_s k)}{\sinh^{-1}(\zeta_s(S_{\max}-k)) + \sinh^{-1}(\zeta_s k)}, \\ y(v) &= \frac{\sinh^{-1}(\zeta_v(v-\hat{v})) + \sinh^{-1}(\zeta_v \hat{v})}{\sinh^{-1}(\zeta_v(V_{\max}-\hat{v})) + \sinh^{-1}(\zeta_v \hat{v})}, \end{aligned} \tag{10}$$

where ζ_S and ζ_v are stretching parameters to be chosen appropriately and where k is typically set to K, and $\hat{v} = V_0$ or $\hat{v} = 0$. The above change of variables has originally been proposed by Clarke and Parrott (1999) and has been further applied by e.g. Oosterlee et al. (2005) and Ballestra and Sgarra (2010). Inverting this transformation we get

$$\begin{aligned} s(x) &= k + \frac{1}{\zeta_s} \sinh(x \sinh^{-1}(\zeta_s(S_{max}-k)) - (1-x)\sinh^{-1}(\zeta_s k)), \\ v(y) &= \hat{v} + \frac{1}{\zeta_v} \sinh(y \sinh^{-1}(\zeta_v(V_{max}-\hat{v})) - (1-y)\sinh^{-1}(\zeta_v \hat{v})). \end{aligned}$$

When selecting a set of points in $[0, 1] \times [0, 1]$, these points are mapped into points of $[0, S_{\max}] \times [0, V_{\max}]$ which tend to be concentrated near $s = k$ and $v = \hat{v}$ (the amount of refinement in the s and v direction is directly proportional to ζ_s and ζ_v, respectively). Oosterlee et al. (2005) note that instead of choosing a fixed k in the transformation (10) above for the whole duration of the option, they vary k so that the grid is refined in the vicinity of the free boundary for all t. So, it is not necessary to keep k fixed at $k = K$. We choose to fix $k = K$ and $\hat{v} = V_0$ at this point to transform the PDE problem for s and v into a PDE problem for x and y. When solving numerically

the resulting PDE problem there is an additional freedom to further refine the grid locally in certain regions of interest by using non-uniform grid points for the spatial discretisation as will be explained in Sect. 6.1.

Along similar lines as in Oosterlee et al. (2005), this coordinate transformation transforms relation (4) into the following spatial differential operator $\mathscr{L}$ applied to $F(t, x, y) = H(t, s(x), v(y))$:

$$\mathscr{L}F = f_1(x, y)\frac{\partial^2 F}{\partial y^2} + h(x, y)\frac{\partial^2 F}{\partial x \partial y} + g_1(x, y)\frac{\partial^2 F}{\partial x^2} + f_2(x, y)\frac{\partial F}{\partial y} + g_2(x, y)\frac{\partial F}{\partial x}, \tag{11}$$

where

$$\begin{aligned} f_1(x, y) &= \frac{1}{2}\sigma^2 \frac{v(y)}{(v'(y))^2}, \quad f_2(x, y) = \frac{\kappa(\theta - v(y))}{v'(y)} - f_1(x, y)\frac{v''(y)}{v'(y)} \\ g_1(x, y) &= \frac{1}{2}v(y)\frac{(s(x))^2}{(s'(x))^2}, \quad g_2(x, y) = (r - q)\frac{s(x)}{s'(x)} - g_1(x, y)\frac{s''(x)}{s'(x)} \\ h(x, y) &= \rho\sigma \frac{v(y)s(x)}{v'(y)s'(x)}. \end{aligned}$$

Next we assume that we reverse time and that t now denotes the time units before the given maturity time, i.e. we replace $T - t$ by t then the PDE (3) becomes

$$\frac{\partial F}{\partial t} - (\mathscr{L} - r)F = 0 \tag{12}$$

and the terminal condition (6) turns into an initial condition

$$F(0, x, y) = P(s(x)) \quad \text{for all } (x, y) \in [0, 1] \times [0, 1]. \tag{13}$$

For solving the free boundary problem, there are some techniques such as linear programming (Borici and Lüthi 2002) or penalty methods as in e.g. (Zvan et al. 1998; Safdari-Vaighani et al. 2015; Shcherbakov and Larsson 2016). We consider the equivalent formulation as a linear complementary problem (LCP), see e.g. Seydel (2009).

Problem 1 Solve for $F = F(t, x, y)$, $(t, x, y) \in (0, T) \times [0, 1] \times [0, 1]$ the system

$$\begin{aligned} \left(\frac{\partial F}{\partial t} - (\mathscr{L} - r)F\right)(F - P) &= 0 \\ \frac{\partial F}{\partial t} - (\mathscr{L} - r)F &\geq 0 \\ F - P &\geq 0 \end{aligned}$$

with initial condition (13) and boundary conditions

$$\lim_{x\to 0} F(t,x,y) = K, \quad \lim_{x\to 1} F(t,x,y) = 0 \tag{14}$$

$$\lim_{y\to 0} F(t,x,y) = P(s(x)), \quad \lim_{y\to 1} \frac{\partial F(t,x,y)}{\partial y} = 0. \tag{15}$$

This LCP formulation has the advantage that the free boundary does not explicitly appear in the problem formulation which enables numerical discretisation.

4 Spatial Discretisation and RBF-PUM Approximation

We will numerically solve Problem 1 using a meshfree radial basis function (RBF) collocation method. The essence of this approach consists in defining differentiation matrices to transform a PDE operator problem into an algebraic problem. We will apply a spatially local variant of more traditional global RBF methods using partition of unity (PU). This RBF-PUM has the advantage of low computational cost due to relatively sparse matrices.

The technique of RBFs is one of the most recently developed meshless methods that has been widely used for scattered data interpolation and approximation in high-dimensions. In this way mesh generation can be avoided featuring flexibility in distributing collocation points for discretisation on an irregular domain.

Hardy (1971) developed RBFs as a multidimensional scattered interpolation method for modelling the earth's gravitational field. Kansa (1990a, b) proposed to use the analytic derivatives of an RBF interpolant to approximate the spatial derivatives of a PDE. For time-dependent equations, this procedure can be combined with a time stepping technique. Kansa's method was extended to solve various ordinary and partial differential equations, for instance see (Larsson and Fornberg 2003; Hu et al. 2005; Dehghan and Shokri 2007, 2009; Tatari and Dehghan 2010; Chen et al. 2016) and the references therein.

We will shortly describe a global approximation with radial basis functions and the RBF based partition of unity.

4.1 RBF Based Approximation

Consider a spatial domain $\Omega \subset \mathbb{R}^d$ and a set of distinct points $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N\}$ in Ω. The basic idea of using RBFs in global approximation methods is to form an interpolant that approximates a function u whose values on this set X of N scattered points are given. The RBF interpolant takes the form

$$\mathscr{I}_u(\boldsymbol{x}) = \sum_{j=1}^{N} \lambda_j \phi(\|\boldsymbol{x} - \boldsymbol{x}_j\|), \tag{16}$$

where $\boldsymbol{x}$ denotes a point in $\mathbb{R}^d$, ϕ is a radial basis function, and $\|\cdot\|$ is the Euclidean norm. The function ϕ depends on the distance between points and not necessarily on their orientation. In Table 1, some globally supported RBFs which are commonly

Table 1 Some well-known functions that generate globally supported RBFs

Function name	Definition
Gaussian (GA)	$\exp(-\varepsilon^2 r^2)$
Multiquadrics (MQ)	$\sqrt{1+(\varepsilon r)^2}$
Inverse multiquadrics (IMQ)	$\frac{1}{\sqrt{1+(\varepsilon r)^2}}$
Conical splines	r^{2k+1}
Thin plate splines (TPS)	$(-1)^{k+1} r^{2k} \log(r)$

employed in the literature are listed. The positive constant ε appearing in the RBFs is called the *shape parameter* which dictates the flatness of the radial basis function and plays a key role in the convergence rate of the approximations and the condition number of the coefficient matrices. For more details about basic properties and types of radial basis functions, compactly and globally supported and also their wide applications in scattered date interpolations, the interested reader is referred to works on this topic (Powell 1992; Buhmann 2000, 2003; Fasshauer 2007).

The coefficients λ_j for $j = 1, 2, \ldots, N$ can be determined by collocating the interpolant $\mathscr{I}_u(\boldsymbol{x})$ to satisfy the interpolation condition $\mathscr{I}_u(\boldsymbol{x}_j) = u(\boldsymbol{x}_j)$, $j = 1, 2, \ldots, N$. This results in a symmetric system of linear equations

$$\boldsymbol{A}\boldsymbol{\lambda} = \boldsymbol{u}, \tag{17}$$

where the elements of $\boldsymbol{A}$ are $A_{i,j} = \phi(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|)$, $\boldsymbol{\lambda} = \left[\lambda_1\ \lambda_2 \cdots \lambda_N\right]^T$ and $\boldsymbol{u} = \left[u(\boldsymbol{x}_1)\ u(\boldsymbol{x}_2) \cdots u(\boldsymbol{x}_N)\right]^T$. For a non-singular coefficient matrix $\boldsymbol{A}$, solving (17) for $\boldsymbol{\lambda}$ and substituting in (16) leads to

$$\mathscr{I}_u(\boldsymbol{x}) = \Phi(\boldsymbol{x})\boldsymbol{A}^{-1}\boldsymbol{u} \tag{18}$$

with $\Phi(\boldsymbol{x}) = \left[\phi(\|\boldsymbol{x} - \boldsymbol{x}_1\|)\ \phi(\|\boldsymbol{x} - \boldsymbol{x}_2\|) \cdots \phi(\|\boldsymbol{x} - \boldsymbol{x}_N\|)\right]$.
When the points in X are chosen to be distinct and ϕ is positive definite the coefficient matrix $\boldsymbol{A}$ is guaranteed to be non-singular (Wendland 2005). This is the case for the RBFs GA and IMQ in Table 1. For the other RBFs a discussion on the existence of a solution to the problem (17) can be found in, e.g., Wendland (2005).

For the implementation of boundary conditions discussed later, it is preferable to express the interpolation in Lagrange form, i.e., using cardinal basis functions. The cardinal basis functions, $\psi_j(\boldsymbol{x})$, $j = 1, 2, \ldots, N$, have the property

$$\psi_j(\boldsymbol{x}_i) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad i = 1, 2, \ldots, N, \tag{19}$$

leading to the alternative formulation for the interpolant

$$\mathscr{I}_u(\boldsymbol{x}) = \Psi(\boldsymbol{x})\boldsymbol{u}, \tag{20}$$

where $\Psi(\boldsymbol{x}) = \left[\psi_1(\boldsymbol{x})\ \psi_2(\boldsymbol{x}) \cdots \psi_N(\boldsymbol{x})\right]$.

Comparing (18) and (20), it is clear that the following relation holds between the cardinal basis and the original radial basis:

$$\Psi(\boldsymbol{x}) = \Phi(\boldsymbol{x})\boldsymbol{A}^{-1}. \tag{21}$$

When we approximate a time dependent function $u(t, \boldsymbol{x})$ that is a solution to a PDE problem, we let λ_j, $j = 1, 2, \ldots, N$, be time dependent such that

$$\mathscr{I}_u(t, \boldsymbol{x}) = \sum_{j=1}^{N} \lambda_j(t)\phi(\|\boldsymbol{x} - \boldsymbol{x}_j\|), \quad \boldsymbol{x} \in \Omega, \ t \geq 0,$$

or, equivalently, when using the Lagrange form, we use the interpolant

$$\mathscr{I}_u(t, \boldsymbol{x}) = \sum_{j=1}^{N} \psi_j(\boldsymbol{x})u_j(t) = \Phi(\boldsymbol{x})\boldsymbol{A}^{-1}\boldsymbol{u}(t). \tag{22}$$

By interpolating the initial condition (or a final condition) such that $\boldsymbol{u}(0) = \left[u(0, \boldsymbol{x}_1)\ u(0, \boldsymbol{x}_2) \cdots u(0, \boldsymbol{x}_N)\right]^T$ we get $\mathscr{I}_u(0, \boldsymbol{x}_k) = u(0, \boldsymbol{x}_k)$ for all k while for $t > 0$ we have $\boldsymbol{u}(t) \approx \left[u(t, \boldsymbol{x}_1)\ u(t, \boldsymbol{x}_2) \cdots u(t, \boldsymbol{x}_N)\right]^T$ and hence $\mathscr{I}_u(t, \boldsymbol{x}_k) \approx u(t, \boldsymbol{x}_k)$ for all k.

4.2 RBF-PUM Based Approximation

Matrices obtained by a global RBF interpolant such as those in Table 1 produce dense matrices due to the fact that all N nodes in the domain are used to generate the interpolant. In this section, we introduce a partition of unity (PU) method which constitute a given set of local approximation spaces together to produce a conforming global approximation. Hereto, some weight functions are used.

Let $\Omega \subset \mathbb{R}^d$ be a bounded set. Let $\{\Omega_j\}_{j=1}^{M}$ be an open and bounded covering of Ω. This means all Ω_j are open and bounded and Ω is contained in their union. Further, we define

$$\forall \boldsymbol{x} \in \Omega \quad I(\boldsymbol{x}) := \{j \mid \boldsymbol{x} \in \Omega_j\}, \quad \text{card}(I(\boldsymbol{x})) \leq K,$$

where the constant K is independent of the number of patches M. For each subdomain we define the weight function $w_j : \Omega_j \to \mathbb{R}$ constructed by using Shepard's method (Shepard 1968) as follows:

$$w_j(\boldsymbol{x}) = \frac{\phi_j(\boldsymbol{x})}{\sum_{k \in I(\boldsymbol{x})} \phi_k(\boldsymbol{x})}, \quad j = 1, 2, \ldots, M \tag{23}$$

where $\phi_j(\boldsymbol{x})$ are compactly supported functions with support on Ω_j. Clearly, the partition of unity property

$$\sum_{j=1}^{M} w_j(\boldsymbol{x}) = \sum_{j \in I(\boldsymbol{x})} w_j(\boldsymbol{x}) = 1 \quad \text{for } \boldsymbol{x} \in \Omega,$$

is satisfied since $w_j(x) = 0$ for all $j \notin I(\boldsymbol{x})$. To guarantee non-negativity and compact support in Ω_j, we define in (23)

$$\phi_j(\boldsymbol{x}) = \phi\left(\frac{\|\boldsymbol{x} - \boldsymbol{c}_j\|}{r_j}\right), \quad j = 1, 2, \ldots, M, \tag{24}$$

where $\{\boldsymbol{c}_j\}_{j=1}^{M}$ and $\{r_j\}_{j=1}^{M}$ are the centres and radii of the circular, spherical or hyper-spherical patches $\{\Omega_j\}_{j=1}^{M}$ and where ϕ is one of the compactly supported functions with minimal degree described in (Wendland 1995, Corollary 9.14). Numerical considerations ask for a polynomial of the lowest possible degree. But the smoothness of the basis function determines that of the approximant. Given the space dimension $d = 2$ in the numerical section, we will choose there the function $\phi(r) = (1-r)_+^4(4r+1)$ which belongs to C^2.

The global approximation function $\mathscr{I}_u(\boldsymbol{x})$, with $\boldsymbol{x} \in \Omega$, to the function $u(\boldsymbol{x})$ is in the RBF-PUM constructed as

$$\mathscr{I}_u(\boldsymbol{x}) = \sum_{j=1}^{M} w_j(\boldsymbol{x}) \mathscr{I}_{u,j}(\boldsymbol{x}) = \sum_{j \in I(\boldsymbol{x})} w_j(\boldsymbol{x}) \mathscr{I}_{u,j}(\boldsymbol{x}),$$

where $\mathscr{I}_{u,j}$ are local interpolants such that $\mathscr{I}_{u,j}(\boldsymbol{x}_i) = u(\boldsymbol{x}_i)$ for each node $\boldsymbol{x}_i \in \Omega_j$. Then, the global PU approximant inherits the interpolation property, i.e. $\mathscr{I}_u(\boldsymbol{x}_i) = u(\boldsymbol{x}_i)$. Using the cardinal basis functions (19) the local interpolant $\mathscr{I}_{u,j}(\boldsymbol{x})$ is an RBF-approximant of type (20) on Ω_j.

When we are dealing with a time dependent function $u(t, \boldsymbol{x})$, we construct the global approximant built up from local RBF interpolants of type (22). For $j \in \{1, \ldots, M\}$, let $J(\Omega_j) := \{k \mid \boldsymbol{x}_k \in \Omega_j\}$ be the set of indices of the node points that belong to the patch Ω_j. For such patch Ω_j, the local RBF approximation is given by

$$\mathscr{I}_{u,j}(t, \boldsymbol{x}) = \sum_{k \in J(\Omega_j)} \psi_k(\boldsymbol{x}) u_k(t)$$

where $\mathscr{I}_{u,j}(t, \boldsymbol{x}_k) = u_k(t)$ for all nodes $\boldsymbol{x}_k \in \Omega_j$ and ψ_k are cardinal basis functions. Hence, in the RBF-PUM, we find for the global approximant for the time-dependent function $u(t, \boldsymbol{x})$

$$\mathscr{I}_u(t, \boldsymbol{x}) = \sum_{j \in I(\boldsymbol{x})} w_j(\boldsymbol{x}) \mathscr{I}_{u,j}(t, \boldsymbol{x}) = \sum_{j \in I(\boldsymbol{x})} \sum_{k \in J(\Omega_j)} w_j(\boldsymbol{x}) \psi_k(\boldsymbol{x}) u_k(t). \tag{25}$$

4.3 Differentiation Matrices

The aim is to use the RBF-PU approximant (25) for solving PDE-problems. The essence of this approach consists in defining differentiation matrices to transform a linear PDE operator problem into a linear algebra problem. Hereto, we need to apply a spatial differential operator $\mathscr{L}$, as e.g. the one in (11), to the global approximant (25).

Using Leibniz's rule, we find for first and second order partial derivatives with respect to x_ℓ or/and x_m, $\ell, m = 1, \ldots, d$,

$$\frac{\partial \mathscr{I}_u}{\partial x_\ell}(t, \boldsymbol{x}) = \sum_{j \in I(x)} \sum_{k \in J(\Omega_j)} \left(\frac{\partial w_j}{\partial x_\ell}(\boldsymbol{x}) \psi_k(\boldsymbol{x}) + w_j(\boldsymbol{x}) \frac{\partial \psi_k}{\partial x_\ell}(\boldsymbol{x}) \right) u_k(t), \tag{26}$$

$$\begin{aligned} \frac{\partial^2 \mathscr{I}_u}{\partial x_\ell \partial x_m}(t, \boldsymbol{x}) = \sum_{j \in I(x)} \sum_{k \in J(\Omega_j)} \Bigg(& \frac{\partial^2 w_j}{\partial x_\ell \partial x_m}(\boldsymbol{x}) \psi_k(\boldsymbol{x}) + \frac{\partial w_j}{\partial x_\ell}(\boldsymbol{x}) \frac{\partial \psi_k}{\partial x_m}(\boldsymbol{x}) \\ & + \frac{\partial w_j}{\partial x_m}(\boldsymbol{x}) \frac{\partial \psi_k}{\partial x_\ell}(\boldsymbol{x}) + w_j(\boldsymbol{x}) \frac{\partial^2 \psi_k}{\partial x_\ell \partial x_m}(\boldsymbol{x}) \Bigg) u_k(t), \end{aligned} \tag{27}$$

where the first and second order partial derivatives of the basis functions ψ_k are elements in the matrices $\Psi_{x_\ell}(\boldsymbol{x})$ and $\Psi_{x_\ell x_m}(\boldsymbol{x})$ which are obtained by means of relation (21), i.e.

$$\Psi_{x_\ell}(\boldsymbol{x}) = \Phi_{x_\ell}(\boldsymbol{x}) \boldsymbol{A}^{-1} \quad \text{resp.} \quad \Psi_{x_\ell x_m}(\boldsymbol{x}) = \Phi_{x_\ell x_m}(\boldsymbol{x}) \boldsymbol{A}^{-1}$$

with $\Psi_{x_\ell}(\boldsymbol{x}) = \left[\frac{\partial}{\partial x_\ell} \psi_1(\boldsymbol{x}) \cdots \frac{\partial}{\partial x_\ell} \psi_N(\boldsymbol{x}) \right]$ and $\Phi_{x_\ell}(\boldsymbol{x}) = \left[\frac{\partial}{\partial x_\ell} \phi(\|\boldsymbol{x} - \boldsymbol{x_1}\|) \cdots \frac{\partial}{\partial x_\ell} \phi(\|\boldsymbol{x} - \boldsymbol{x_N}\|) \right]$ and analogously for the second order partial derivatives.

By evaluating at the nodes $\boldsymbol{x}_i$, $i = 1, \ldots, N$, the terms between brackets in (26) and in (27) deliver for each j the ik-elements of the local differentation matrices corresponding to the first and second order respectively. Putting these elements at the corresponding places of the $N \times N$ differentiation matrices we get the global differentiation matrices $\mathbf{D}_{x_\ell}$ and $\mathbf{D}_{x_\ell x_m}$ such that relations (26) and (27) evaluated at the nodes $\boldsymbol{x}_i$, $i = 1, \ldots, N$ can be expressed as the matrix product $\mathbf{D}_{x_\ell} \boldsymbol{u}(t)$ and $\mathbf{D}_{x_\ell x_m} \boldsymbol{u}(t)$ respectively. For a composite linear operator such as $\mathscr{L}$ in (11), we combine these matrices to get the global differentation matrix under this operator.

Note that in the context of a PDE-problem the formation and assembling of the differentiation matrices can be done beforehand and in parallel for the different patches. Moreover, these differentiation matrices are sparse due to the compact support of the weight functions and of the basis functions.

4.4 Spatial Discretisation

Assume that the set of nodes $X = X_I \cup X_B$ where X_I and X_B are subsets consisting of N_I interior nodes and $N_B = N - N_I$ boundary nodes respectively, where the

indices are ordered such that N_I interior nodes are followed by N_B boundary node. Further, assume that the local nodal values coincide with the global nodal values of the unknown function u and that the unknown function satifies some Dirichlet boundary condition such that $u_i(t) = g(t, \boldsymbol{x}_i), i = N_I+1, \ldots, N$. Then, the global approximant (25) can also be split up in two parts

$$\begin{aligned}\mathscr{I}_u(t, \boldsymbol{x}) = &\sum_{j \in I(\boldsymbol{x})} \sum_{k \in J(\Omega_j \cap X_I)} w_j(\boldsymbol{x}) \psi_k(\boldsymbol{x}) u_k(t) \\ &+ \sum_{j \in I(\boldsymbol{x})} \sum_{k \in J(\Omega_j \cap X_B)} w_j(\boldsymbol{x}) \psi_k(\boldsymbol{x}) g(t, \boldsymbol{x}_k)\end{aligned}$$

Applying the partial derivatives w.r.t. x_ℓ or/and x_m, for $\ell, m = 1, \ldots, d$ to both sides and collocating at the interior nodal points $\boldsymbol{x}_i$, $i = 1, \ldots, N_I$, we obtain the block matrix products

$$\begin{aligned}\frac{\partial}{\partial x_\ell} \boldsymbol{u}_I(t) &= \begin{bmatrix}\mathbf{D}_{x_\ell, I} & \mathbf{D}_{x_\ell, B}\end{bmatrix} \begin{bmatrix}\boldsymbol{u}_I(t) \\ \boldsymbol{g}(t)\end{bmatrix} \quad \text{and} \\ \frac{\partial^2}{\partial x_\ell \partial x_m} \boldsymbol{u}_I(t) &= \begin{bmatrix}\mathbf{D}_{x_\ell x_m, I} & \mathbf{D}_{x_\ell x_m, B}\end{bmatrix} \begin{bmatrix}\boldsymbol{u}_I(t) \\ \boldsymbol{g}(t)\end{bmatrix}\end{aligned}$$

where the $N_I \times N_I$ differentation matrices $\mathbf{D}_{x_\ell, I}$, $\mathbf{D}_{x_\ell x_m, I}$ and the vectors $\boldsymbol{u}_I(t) = \begin{bmatrix}u_1(t) \cdots u_{N_I}(t)\end{bmatrix}^T$ correspond to the interior nodes, while the $N_I \times N_B$ differentation matrices $\mathbf{D}_{x_\ell, B}$, $\mathbf{D}_{x_\ell x_m, B}$ and the vector $\boldsymbol{g}(t) = \begin{bmatrix}g(t, \boldsymbol{x}_{N_I+1}) \cdots g(t, \boldsymbol{x}_N)\end{bmatrix}^T$ correspond to the boundary nodes.

The above relations can also be expressed using $N \times N$ matrices as follows:

$$\begin{aligned}\begin{bmatrix}\frac{\partial \boldsymbol{u}_I}{\partial x_\ell}(t) \\ \boldsymbol{g}(t)\end{bmatrix} &= \begin{bmatrix}\mathbf{D}_{x_\ell, I} & \mathbf{D}_{x_\ell, B} \\ \mathbf{O}_{N_B \times N_I} & \mathbf{I}_{N_B \times N_B}\end{bmatrix} \begin{bmatrix}\boldsymbol{u}_I(t) \\ \boldsymbol{u}_B(t)\end{bmatrix} \quad \text{and} \\ \begin{bmatrix}\frac{\partial^2 \boldsymbol{u}_I}{\partial x_\ell \partial x_m}(t) \\ \boldsymbol{g}(t)\end{bmatrix} &= \begin{bmatrix}\mathbf{D}_{x_\ell x_m, I} & \mathbf{D}_{x_\ell x_m, B} \\ \mathbf{O}_{N_B \times N_I} & \mathbf{I}_{N_B \times N_B}\end{bmatrix} \begin{bmatrix}\boldsymbol{u}_I(t) \\ \boldsymbol{u}_B(t)\end{bmatrix},\end{aligned}$$

with $\boldsymbol{u}_B(t) = \begin{bmatrix}u_{N_I+1}(t) \cdots u_N(t)\end{bmatrix}^T$.

Assume that in certain boundary nodes a Neumann boundary condition has to be satisfied. Then, we number those nodes such that for the nodes $N_I + 1$ up to $N_I + N_{BD}$ the Dirichlet boundary conditions hold and for the remaining N_{BN} nodes numbered $N_I + N_{BD} + 1$ up to N the Neumann boundary condition $\frac{\partial u}{\partial x_n}(t, \boldsymbol{x}) = h(t, \boldsymbol{x})$ for $n \in \{1, \ldots, d\}$. Then we get

$$\begin{bmatrix}\frac{\partial \boldsymbol{u}_I}{\partial x_\ell}(t) \\ \boldsymbol{g}(t) \\ \boldsymbol{h}(t)\end{bmatrix} = \begin{bmatrix}\mathbf{D}_{x_\ell, I} & \mathbf{D}_{x_\ell, BD} & \mathbf{D}_{x_\ell, BN} \\ \mathbf{O}_{N_{BD} \times N_I} & \mathbf{I}_{N_{BD} \times N_{BD}} & \mathbf{O}_{N_{BD} \times N_{BN}} \\ \mathbb{D}_{x_n, I} & \mathbb{D}_{x_n, BD} & \mathbb{D}_{x_n, BN}\end{bmatrix} \begin{bmatrix}\boldsymbol{u}_I(t) \\ \boldsymbol{u}_{BD}(t) \\ \boldsymbol{u}_{BN}(t)\end{bmatrix} \tag{28}$$

where the differentiation matrices $\mathbb{D}_{x_n,I}$, $\mathbb{D}_{x_n,BD}$ and $\mathbb{D}_{x_n,BN}$ are obtained by collocation of (26) at the N_{BN} boundary nodes $\boldsymbol{x}_i$, $i = N_I + N_{BD} + 1, \ldots, N$.

We now apply this to discretise Problem 1 where $d = 2$ and the option values $F(t, x, y)$ are approximated at the spatial grid points $\boldsymbol{x}_i$, $i = 1, \ldots, N$ by $F_i(t) \approx F(t, \boldsymbol{x}_i)$. We will number the grid points such that we have first the N_I interior points, followed by the N_{BD} boundary points on the sides $x = 0$, $x = 1$ and $y = 0$ of the square $[0, 1] \times [0, 1]$ where a Dirichlet boundary holds, and closed by the N_{BN} boundary points on the side $y = 1$ where a Neumann condition holds. We recall that the boundary condition on the boundary $y = 0$ will only be imposed when the Feller condition is not satisfied.

Using the RBF-PUM approximation (25) and collocating at the interior nodal points the PDE for F with $\mathscr{L}F$ given by (11), we obtain the semidiscrete LCP

$$\begin{cases} (\boldsymbol{F}_I(t) - \boldsymbol{P}_I)^T (\boldsymbol{F}_I'(t) - \mathbf{L}\boldsymbol{F}(t) + r\boldsymbol{F}_I(t)) = 0, \\ \boldsymbol{F}_I'(t) - \mathbf{L}\boldsymbol{F}(t) + r\boldsymbol{F}_I(t) \geq \mathbf{0}, \qquad 0 < t \leq T \\ \boldsymbol{F}_I(t) - \boldsymbol{P}_I \geq \mathbf{0}, \end{cases} \tag{29}$$

with $\boldsymbol{P}_I = \left[P(s(x_1)) \cdots P(s(x_{N_I}))\right]^T$, and with the vector $\boldsymbol{F}(t)$ consisting of three blocks $\boldsymbol{F}_I(t) = \left[F_1(t) \cdots F_{N_I}(t)\right]^T$, $\boldsymbol{F}_{BD}(t) = \left[F_{N_I+1}(t) \cdots F_{N_I+N_{BD}}(t))\right]^T$ and $\boldsymbol{F}_{BN}(t) = \left[F_{N_I+N_{BD}+1}(t) \cdots F_N(t)\right]^T$, and where the matrix $\mathbf{L}$ can also be split up in corresponding blocks $\mathbf{L}_I, \mathbf{L}_{BD}, \mathbf{L}_{BN}$ such that

$$\mathbf{L}\boldsymbol{F}(t) = \mathbf{L}_I\boldsymbol{F}_I(t) + \mathbf{L}_{BD}\boldsymbol{F}_{BD}(t) + \mathbf{L}_{BN}\boldsymbol{F}_{BN}(t).$$

These block matrices $\mathbf{L}_\star$ with $\star = I, BD, BN$ are composed of differentiation matrices of order $N_I \times N_\star$ introduced in Sect. 4.3 and at the beginning of this section as follows

$$\mathbf{L}_\star = f_1(\boldsymbol{x}_I)\mathbf{D}_{yy,\star} + h(\boldsymbol{x}_I)\mathbf{D}_{xy,\star} + g_1(\boldsymbol{x}_I)\mathbf{D}_{xx,\star} + f_2(\boldsymbol{x}_I)\mathbf{D}_{y,\star} + g_2(\boldsymbol{x}_I)\mathbf{D}_{x,\star}.$$

$f_1(\boldsymbol{x}_I)$, $f_2(\boldsymbol{x}_I)$, $g_1(\boldsymbol{x}_I)$, $g_2(\boldsymbol{x}_I)$, $h(\boldsymbol{x}_I)$ are short hand notations for diagonal matrices with on the diagonal the function value in the interior nodes $\boldsymbol{x}_i, i = 1, \ldots, N_I$. Further, inequalities are to be interpreted component wise.

The Dirichlet boundary conditions can be included by substituting the matrix $\boldsymbol{G}$ for $\boldsymbol{F}_{BD}(t)$. The elements of $\boldsymbol{G}$ corresponding to the nodes on the boundary $x = 0$ are the value K, while those corresponding to the nodes on the boundary $x = 1$ are zero and those corresponding to the nodes on the boundary $y = 0$ (when the Feller condition is not satisfied) are the values of the payoff function at those nodes.

To account for the Neumann boundary condition at $y = 1$, an additional system of equations has to be added to (29) as in (28), namely

$$\mathbb{D}_y\boldsymbol{F}(t) := \mathbb{D}_{y,I}\boldsymbol{F}_I(t) + \mathbb{D}_{y,BD}\boldsymbol{F}_{BD}(t) + \mathbb{D}_{y,BN}\boldsymbol{F}_{BN}(t) = \mathbf{0}, \tag{30}$$

where also $\boldsymbol{G}$ can be substituted for $\boldsymbol{F}_{BD}(t)$.

5 Time Discretisation

In this section, we consider the discretisation of the time derivative. Hereto, let $\triangle t = T/N_t$ be the time step for a given integer $N_t \geq 1$. Then the temporal grid points are given by $t_k = k\triangle t$, $k = 0, \ldots, N_t$. We apply the Crank-Nicolson scheme to the semidiscrete LCP (29) including the Dirichlet and Neumann boundary conditions and get a fully discrete problem in terms of the approximation vectors $\boldsymbol{F}^k \approx \boldsymbol{F}(t_k)$ for $k = 1, \ldots, N_t$,

$$\begin{cases} (\boldsymbol{F}_I^k - \boldsymbol{F}_I^0)^T((\mathbf{I} - \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I}))\boldsymbol{F}_I^k - (\mathbf{I} + \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I}))\boldsymbol{F}_I^{k-1} - \triangle t\mathbf{L}_{BD}\boldsymbol{G} - \triangle t\mathbf{L}_{BN}\boldsymbol{F}_{BN}^{k-1}) = 0, \\ (\mathbf{I} - \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I}))\boldsymbol{F}_I^k - (\mathbf{I} + \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I}))\boldsymbol{F}_I^{k-1} - \triangle t\mathbf{L}_{BD}\boldsymbol{G} - \triangle t\mathbf{L}_{BN}\boldsymbol{F}_{BN}^{k-1} \geq \mathbf{0}, \\ \boldsymbol{F}_I^k - \boldsymbol{F}_I^0 \geq \mathbf{0}, \\ \mathbb{D}_y \boldsymbol{F}^k = \mathbf{0}, \end{cases}$$

where $\boldsymbol{F}^0 = \boldsymbol{P}$ is the initial condition.

For solving the above LCP problem, we apply a method which makes use of an auxiliary vector $\boldsymbol{\lambda}^k$ and which is known as the *operator splitting method* introduced by Ikonen and Toivanen (2004). Hence the LCP is reformulated as

$$\begin{cases} (\mathbf{I} - \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I}))\boldsymbol{F}_I^k - (\mathbf{I} + \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I}))\boldsymbol{F}_I^{k-1} - \triangle t\mathbf{L}_{BD}\boldsymbol{G} - \triangle t\mathbf{L}_{BN}\boldsymbol{F}_{BN}^{k-1} = \triangle t\boldsymbol{\lambda}^k, \\ (\boldsymbol{F}_I^k - \boldsymbol{F}_I^0)^T\boldsymbol{\lambda}^k = 0, \\ \boldsymbol{\lambda}^k \geq \mathbf{0}, \\ \boldsymbol{F}_I^k - \boldsymbol{F}_I^0 \geq \mathbf{0}, \\ \mathbb{D}_y \boldsymbol{F}^k = \mathbf{0} \\ \boldsymbol{F}^0 = \boldsymbol{P}. \end{cases}$$

The operator splitting method contains two stages. In the first stage, an intermediate approximation $\widetilde{\boldsymbol{F}}_I^k$ on the subinterval $[t_{k-1}, t_k]$ is computed by solving

$$\left(\mathbf{I} - \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I})\right)\widetilde{\boldsymbol{F}}_I^k = \left(\mathbf{I} + \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I})\right)\boldsymbol{F}_I^{k-1} + \triangle t\mathbf{L}_{BD}\boldsymbol{G} + \triangle t\mathbf{L}_{BN}\boldsymbol{F}_{BN}^{k-1} + \triangle t\widetilde{\boldsymbol{\lambda}}^k \tag{31}$$

where $\widetilde{\boldsymbol{F}}_I^0 = \boldsymbol{F}_I^0$, and the vector $\widetilde{\boldsymbol{\lambda}}^k$ is given at the start of each time step.

The second stage is concerned with determining the approximation solutions $\boldsymbol{F}_I^k$ and $\boldsymbol{\lambda}^k$ on the subinterval $[t_{k-1}, t_k]$ by solving the problem

$$\boldsymbol{F}_I^k - \widetilde{\boldsymbol{F}}_I^k = \triangle t(\boldsymbol{\lambda}^k - \widetilde{\boldsymbol{\lambda}}^k)$$

with two inequality constraints and one equality constraint

$$\boldsymbol{\lambda}^k \geq \mathbf{0}, \quad \boldsymbol{F}_I^k \geq \boldsymbol{F}_I^0, \quad (\boldsymbol{F}_I^k - \boldsymbol{F}_I^0)^T\boldsymbol{\lambda}^k = 0.$$

The above problem can be performed very fast at each spatial grid point independently with the formulas

$$\boldsymbol{F}_I^k = \max\left(\boldsymbol{F}_I^0, \widetilde{\boldsymbol{F}}_I^k - \triangle t\widetilde{\boldsymbol{\lambda}}^k\right), \quad \boldsymbol{\lambda}^k = \max\left(\widetilde{\boldsymbol{\lambda}}^k + \frac{\boldsymbol{F}_I^0 - \widetilde{\boldsymbol{F}}_I^k}{\triangle t}, 0\right).$$

Finally, knowing $\boldsymbol{F}_I^k$ and $\boldsymbol{F}_{BD}^k = G$, $\boldsymbol{F}_{BN}^k$ is according to (30) solved from

$$\mathbb{D}_{y,BN}\boldsymbol{F}_{BN}^k = -\mathbb{D}_{y,I}\boldsymbol{F}_I^k - \mathbb{D}_{y,BD}\boldsymbol{G}. \tag{32}$$

Note that in this procedure two linear algebraic systems have to be solved, namely one with the $N_I \times N_I$ coefficient matrix $\mathbf{I} - \frac{1}{2}\triangle t\,(\mathbf{L}_I - r\mathbf{I})$ in (31) and one with the $N_{BN} \times N_{BN}$ coefficient matrix $\mathbb{D}_{y,BN}$ in (32). Alternatively, one could merge those linear systems as in (28) to a system with one $N \times N$ coefficient matrix. The choice of the present procedure is motivated by numerical experiments as in Sect. 6 that show that the condition numbers of the smaller matrices are much smaller compared with the condition number of the $N \times N$ matrix. Moreover, the condition number of $\mathbf{I} - \frac{1}{2}\triangle t\,(\mathbf{L}_I - r\mathbf{I})$ decreases with an increasing $\triangle t$.

6 Numerical Results

In this section we will present some numerical results to test the accuracy and efficiency of the proposed method based on RBF-PUM. We will first price a European option under the Heston model since in that case an analytical solution exists. For American options there is no analytical solution and we will compare our results to those found in literature. We will test different parameter sets corresponding to whether the Feller condition is satisfied or not. Since we put the dividend yield equal to zero we will focus on American put options. We will also compute the corresponding European put options.

For the numerical experiments we truncate the unbounded domain for s and v by limiting the asset values s by $S_{max} = 2K$ and the variance v by $V_{max} = 1$ or 0.5 depending on whether the Feller condition is satisfied or not, thus

$$[0, S_{max}] \times [0, V_{max}] = \begin{cases} [0, 2K] \times [0, 1] & \text{Feller condition satisfied,} \\ [0, 2K] \times [0, 0.5] & \text{Feller condition not satisfied.} \end{cases}$$

6.1 Uniform Versus Non-uniform Spatial Discretisation

According to Sect. 3.2 we transform the spatial domain $[0, 2K] \times [0, V_{max}]$ for (s, v) into the square $[0, 1] \times [0, 1]$ for (x, y). Hereto, we choose $\zeta_s = 0.1$ and $\zeta_v = 0.01$ in (10) which corresponds to a finer grid for (s, v) near the discontinuity of the payoff function at $s = K$ when using a *uniform* discretisation of the domain $[0, 1] \times [0, 1]$ as in Fig. 1 with m grid points in x-direction and n in y-direction. Thus the total number of nodal points equals $N = m \times n$. In the experiments we choose $m = n$.

Further, we will investigate the effect on the accuracy of the approximation when using a *nonuniform* node placement in x and y direction compared to a uniform one. We apply a nonuniform discretisation as in in 't Hout and Foulon (2010), Safdari-Vaighani

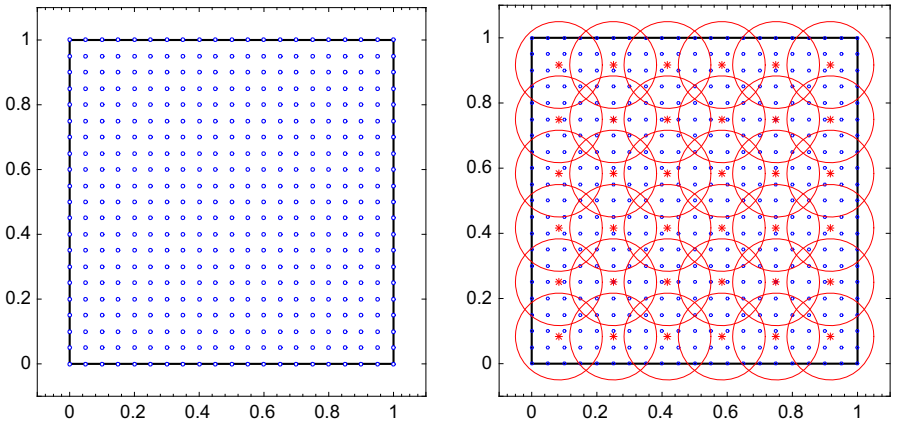

Fig. 1 *Left* Uniform node distributions, *right* partitioning of the square domain with circular patches for uniform node distributions

et al. (2015), Tavella and Randall (2000) and Haentjens and in 't Hout (2015). The coordinates of the nodes are defined by

$$x_i = x(K) + l^x \sinh\left(\zeta_i^x\right), \quad 0 \le i \le m,$$
$$y_j = y(V_0) + l^y \sinh\left(\zeta_j^y\right), \quad 0 \le j \le n,$$

with $x(K)$ en $y(V_0)$ given by (10), and where $\zeta_j^x \in [\zeta_0^x, \zeta_m^x]$ and $\zeta_j^y \in [\zeta_0^y, \zeta_n^y]$ are equidistant values and l^x and l^y are parameters that determine the distribution of the points in the area. Since the nodes should belong to the region of the problem, the range of ζ is computed as follow

$$\zeta_0^x = \sinh^{-1}\left(-\frac{x(K)}{l^x}\right), \quad \zeta_m^x = \sinh^{-1}\left(\frac{1-x(K)}{l^x}\right),$$
$$\zeta_0^y = \sinh^{-1}\left(-\frac{y(V_0)}{l^y}\right), \quad \zeta_n^y = \sinh^{-1}\left(\frac{1-y(V_0)}{l^y}\right).$$

Figure 2 shows a nonuniform grid which is obtained by choosing $l^x = x(K)/2$ and $l^y = y(V_0)/2$ in the transformation formulae above. In the left figure a finer grid can clearly be observed around $y = 0.25$ which corresponds to the value $V_0 = 0.25$ of interest for the variance and in the right figure around $y = 0.1$ corresponding to $V_0 = 0.0625$.

6.2 Patches, Partition of Unity Weights

In the RBF-PU method we need to cover the domain by patches. The centres of those patches are either uniformly or nonuniformly distributed similarly as the nodes. In Fig. 1(right) the domain is covered by circular patches where the centres are uniformly

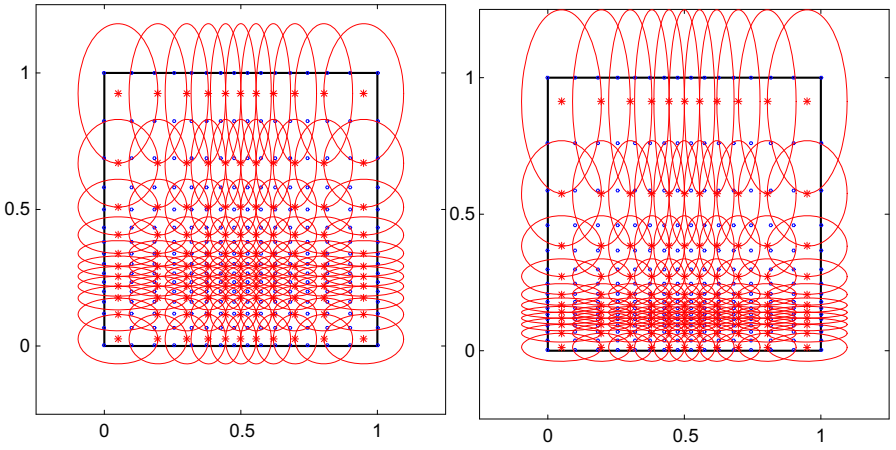

Fig. 2 Discretisation of $[0, 1] \times [0, 1]$ with nonuniform nodes and elliptic patches for $S_0 = 10$ and $V_0 = 0.25$ (*left*) respectively $V_0 = 0.0625$ (*right*)

distributed as the nodes. However when the centres of patches are nonuniformly distributed as in Fig. 2, circular patches are not suitable and elliptic ones are used instead. In our numerical experiments we have chosen the clustering parameters $l^x = x(K)/3$ and $l^y = y(V_0)/3$ for the centres of the patches.

We further use the Wendland function $\phi(r) = (1 - r)_+^4(4r + 1)$ in (24) for constructing the partition of unity weights (23) with compact support in the patches Ω_j, $j = 1, \ldots, M$ in case of uniform distributed nodes and centres of circular patches. When working with elliptic patches we use the same Wendland function but the argument is not as in (24), it is scaled as follows for a point $(x, y) \in \Omega_j$, $j = 1, \ldots, M$,

$$\phi_j(x, y) = \phi\left(\sqrt{\frac{(x - c_{x,j})^2}{r_{x,j}^2} + \frac{(y - c_{y,j})^2}{r_{y,j}^2}}\right)$$

with $\boldsymbol{c}_j = (c_{x,j}, c_{y,j})$ the centre of the elliptic patch and with $r_{x,j}$ and $r_{y,j}$ the radii in the two directions.

6.3 RBF and Shape Parameter

As radial basis function for the local approximations we select the inverse multiquadric (IMQ) one (see Table 1). Experiments with the multiquadric RBF lead to similar results as the IMQ RBF while those with Gaussian RBFs produce less accurate results. Therefore we only report those for the IMQ RBFs.

The shape parameter for the radial basis function is of importance for the accuracy of the RBF method. We scale the shape parameter ε with respect to the node density in the patch such that $\varepsilon_j = \varepsilon h/\delta_j$ is a shape parameter for Ω_j where h is the uniform node distance corresponding to the number of nodes used, and δ_j is the actual minimum node distance within the patch. To find the best size of ε for our problem, we plot the

Table 2 Parameter setup for put options

	Feller condition $2\kappa\theta \geq \sigma^2$	
	Satisfied	Not satisfied
Time of maturity T	0.25	0.25
Strike price K	10	100
Volatility σ	0.9	0.39
Mean reversion rate κ	5	1.15
Correlation ρ	0.1	−0.64
Risk free interest rate r	0.1	0.04
Mean reversion level θ	0.16	0.0348
Dividend yield of the asset q	0	0

L_2-error for different values of the shape parameter for the different parameter sets of Table 2 with $V_0 \in \{0.065, 0.25\}$ when the Feller condition is satisfied and with $V_0 = 0.0348$ when the Feller condition is not satisfied. The L_2-error is computed for a fixed V_0 as follows

$$L_2\text{-error} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(H(0, s_i, V_0) - H_{\text{exact}})^2}. \tag{33}$$

As reference value H_{exact} we use the exact one for the European option while we choose the solution of Ikonen and Toivanen (2009) reported in Table 6 for American put options when the Feller condition is satisfied and the one of Haentjens and in 't Hout (2015) reported in Table 8 when the Feller condition is not satisfied. According to Fig. 3, we choose $\varepsilon = 2$ for the European option pricing experiments and $\varepsilon = 3$ for the American ones.

6.4 Sparsity and Refinement Strategy

A global RBF approximation leads to a dense coefficient matrix $\mathbf{I} - \frac{1}{2}\triangle t(\mathbf{L}_I - r\mathbf{I})$ in the resulting linear algebraic problem (31) after discretisation. Therefore, we use a partition of unity method of RBFs which introduces sparsity in the discretisation matrices. The sparsity pattern of the matrix depends on the number of patches as can be seen in Fig. 4. More patches lead to more sparsity for the same number of nodes. The question is which number of patches leads to a good efficiency since more patches means more sparsity but also lower accuracy? So a trade-off has to be made. For a fixed number of nodal points we compute the absolute error for different numbers of patches. The number of patches corresponding to the the minimal error is then used in the numerical results. So when increasing the number of nodal points to refine the grid the number of patches will be changed accordingly.

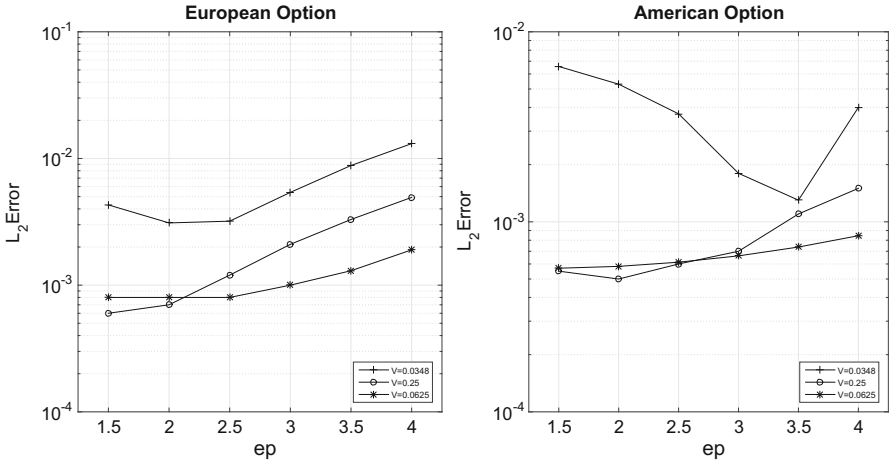

Fig. 3 L_2-error with respect to ε and nonuniform discretisation with 38×38 number of nodes for European option (*left*) respectively for American option (*right*)

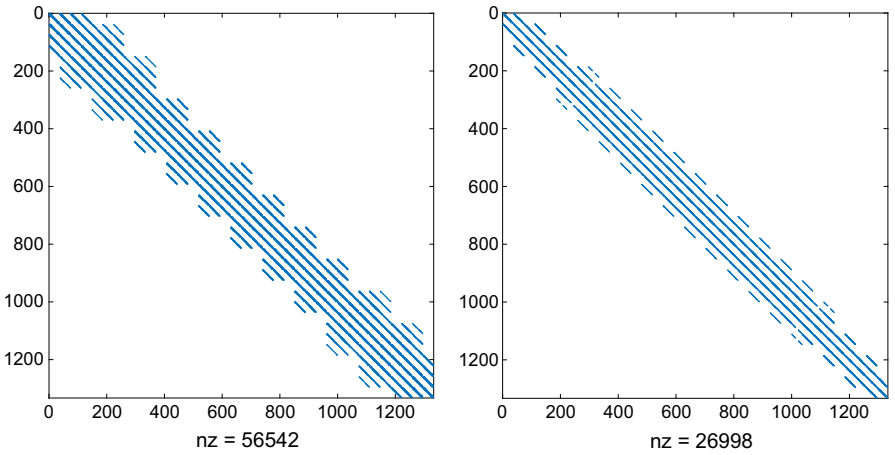

Fig. 4 Sparsity of stiffness matrix with 38×38 uniform nodes and 12×12 patches (*left*) respectively 18×18 patches (*right*)

6.5 European Options

We first illustrate the proposed method for the pricing of European options since an analytical solution exists in that case.

In Tables 3, 4 and 5 we compare the quality of the approximation when using a uniform grid and a nonuniform grid. We distinguish three cases, two where the Feller condition is satisfied but where we further differentiate between high (0.25) and low (0.0625) variance V_0 and one where the Feller condition is not satisfied ($V_0 = 0.0348$). We recall that we consider m nodes in the x-direction (and hence in the s-direction) and n in the y-direction (and hence in the v-direction). The second column displays the

Table 3 European put option prices under the Heston model with the parameter set of Table 2 with satisfied Feller condition, $V_0 = 0.25$ and $N_t = 25$ for uniform and nonuniform discretisation

$m = n$	$\sqrt{M}$	$S_0 = 8$	$S_0 = 9$	$S_0 = 10$	$S_0 = 11$	$S_0 = 12$	CPU (s)	L_2-error
Uniform								
30	10	1.97347	1.27408	0.76281	0.43026	0.23349	0.03181	5.4e−3
34	13	1.97349	1.27238	0.76078	0.42891	0.23315	0.04780	6.6e−3
38	18	1.97965	1.28205	0.77203	0.43918	0.24087	0.05575	2.8e−3
Nonuniform								
30	10	1.97510	1.27632	0.76529	0.43195	0.23390	0.03817	3.6e−3
34	13	1.97762	1.27972	0.76942	0.43668	0.23852	0.05194	6.7e−4
38	18	1.97811	1.28087	0.77024	0.43620	0.23678	0.05716	6.2e−4
Exact price		1.97731	1.28000	0.76969	0.43605	0.23726		

Table 4 European put option prices under the Heston model with the parameter set of Table 2 with satisfied Feller condition, $V_0 = 0.0625$ and $N_t = 25$ for uniform and nonuniform discretisation

$m = n$	$\sqrt{M}$	$S_0 = 8$	$S_0 = 9$	$S_0 = 10$	$S_0 = 11$	$S_0 = 12$	CPU (s)	L_2-error
Uniform								
30	10	1.83590	1.03939	0.48804	0.19986	0.07673	0.03093	8.4e−3
34	13	1.83546	1.03732	0.48806	0.19877	0.07666	0.03093	9.1e−3
38	18	1.84063	1.04992	0.50383	0.21125	0.08373	0.05515	2.5e−3
Nonuniform								
30	10	1.83812	1.04574	0.49658	0.20561	0.07898	0.03911	2.8e−3
34	13	1.83992	1.04733	0.49939	0.20910	0.08248	0.05168	1.5e−3
38	18	1.83879	1.04830	0.50200	0.20690	0.07945	0.05568	7.6e−4
Exact price		1.83887	1.04835	0.50147	0.20819	0.08043		

Table 5 European put option process under the Heston model with the parameter set of Table 2 with Feller condition not satisfied, $V_0 = 0.0348$ and $N_t = 20$ for uniform and nonuniform discretisation

$m = n$	$\sqrt{M}$	$S_0 = 90$	$S_0 = 100$	$S_0 = 110$	CPU (s)	L_2-error
Uniform						
30	10	9.40569	2.99034	0.88641	0.02544	8.6e−2
34	13	9.46082	3.02240	0.90659	0.03648	8.3e−2
38	18	9.39125	3.16383	0.93617	0.04110	2.4e−2
Nonuniform						
30	10	9.44949	3.13290	0.91889	0.02801	4.7e−2
34	13	9.44810	3.12303	0.92411	0.03928	4.6e−2
38	18	9.37385	3.13326	0.91826	0.04439	3.0e−3
Exact price		9.36868	3.13248	0.91752		

Table 6 American put option prices under the Heston model for the parameter set of Table 2 with satisfied Feller condition and $V_0 = 0.25$ for uniform and nonuniform discretisation

$m = n$	$\sqrt{M}$	N_t	$S_0 = 8$	$S_0 = 9$	$S_0 = 10$	$S_0 = 11$	$S_0 = 12$	CPU (s)
Uniform								
30	10	25	2.07851	1.33331	0.79785	0.45316	0.25045	0.03163
		50	2.07704	1.33064	0.79419	0.44893	0.24598	0.05694
		75	2.07664	1.32987	0.79307	0.44758	0.24451	0.08445
34	14	25	2.08264	1.34028	0.80519	0.46073	0.25705	0.04100
		50	2.08113	1.33768	0.80169	0.45666	0.25271	0.07854
		75	2.08072	1.33692	0.80061	0.45536	0.25130	0.11694
38	18	25	2.08381	1.34197	0.80700	0.46141	0.25675	0.05511
		50	2.08228	1.33940	0.80352	0.45734	0.25241	0.10855
		75	2.08183	1.33862	0.80242	0.45603	0.25099	0.16679
Nonuniform								
30	10	25	2.07699	1.33118	0.79266	0.44482	0.23969	0.03644
		50	2.07678	1.33101	0.79258	0.44480	0.23971	0.06821
		75	2.07671	1.33096	0.79256	0.44480	0.23972	0.11355
34	14	25	2.07905	1.33493	0.79824	0.45140	0.24618	0.04980
		50	2.07877	1.33477	0.79816	0.45139	0.24620	0.09548
		75	2.07871	1.33472	0.79814	0.45139	0.24620	0.14237
38	18	25	2.07955	1.33463	0.79602	0.44721	0.24170	0.05867
		50	2.07929	1.33455	0.79608	0.44735	0.24185	0.11374
		75	2.07926	1.33456	0.79614	0.44742	0.24193	0.15527
Ikonen and Toivanen (2009)			2.0785	1.3336	0.7959	0.4482	0.2427	
Zvan et al. (1998)			2.0784	1.3337	0.7961	0.4483	0.2428	
Oosterlee (2003)			2.079	1.334	0.796	0.449	0.243	
Haentjens and in 't Hout (2015)			2.0788	1.3339	0.7962	0.4486	0.2433	

number of patches in the x-direction and in the y-direction with a total of $\sqrt{M} \times \sqrt{M}$, and the number of time steps is denoted by N_t. The L_2-error is computed according to (33) with $N = 5$ in Tables 3 and 4 and with $N = 3$ in Table 5. It is clear that the results for a nonuniform discretisation are more accurate than for a uniform discretisation and are obtained in a comparable computational time. Increasing the number of nodal points and correspondingly the number of patches also leads to an improvement of the approximation. Since an increase in the number of time steps N_t leads to comparable European option prices we do not report those here.

The reported CPU times in seconds for the experiments do not include the setup cost for the computations of the differentiation matrices. In the RBF-PU method, these computations can easily be processed beforehand and be parallelised. They will not greatly affect the overall time. A 3.6 GHz Corei3 processor was used.

Table 7 American put option prices under the Heston model for the parameter set of Table 2 with satisfied Feller condition and $V_0 = 0.0625$ for uniform and nonuniform discretisation

$m = n$	$\sqrt{M}$	N_t	$S_0 = 8$	$S_0 = 9$	$S_0 = 10$	$S_0 = 11$	$S_0 = 12$	CPU (s)
Uniform								
30	10	25	2.00052	1.10466	0.51581	0.21694	0.08987	0.03154
		50	2.00021	1.10218	0.51178	0.21230	0.08501	0.05599
		75	2.00024	1.10151	0.51056	0.21080	0.08339	0.08637
34	14	25	1.99974	1.11115	0.52861	0.22881	0.09780	0.04115
		50	1.99857	1.10885	0.52484	0.22436	0.09311	0.07672
		75	1.99835	1.10821	0.52367	0.22292	0.09155	0.11135
38	18	25	2.00058	1.10925	0.53018	0.22640	0.09538	0.05609
		50	2.00024	1.10697	0.52647	0.22196	0.09068	0.11010
		75	2.00015	1.10627	0.52529	0.22050	0.08913	0.16411
Nonuniform								
30	10	25	2.00001	1.10450	0.51439	0.20964	0.07952	0.03805
		50	1.99998	1.10453	0.51433	0.20969	0.07959	0.06924
		75	1.99997	1.10455	0.51432	0.20970	0.07961	0.11418
34	14	25	1.99820	1.10704	0.52065	0.21620	0.08490	0.04801
		50	1.99860	1.10684	0.52062	0.21623	0.08495	0.09673
		75	1.99860	1.10680	0.52062	0.21625	0.08496	0.14222
38	18	25	1.99968	1.10848	0.52036	0.21226	0.08142	0.11360
		50	1.99969	1.10839	0.52051	0.21249	0.08160	0.12769
		75	1.99970	1.10839	0.52057	0.21258	0.08167	0.15645
Ikonen and Toivanen (2009)			2.0000	1.1076	0.5199	0.2135	0.0820	
Zvan et al. (1998)			2.0000	1.1076	0.5202	0.2138	0.0821	
Oosterlee (2003)			2.00	1.107	0.517	0.212	0.0815	
Haentjens and in 't Hout (2015)			2.0000	1.1081	0.5204	0.2143	0.0827	

6.6 American Options

We now discuss the proposed RBF-PU method for pricing of American put options under the Heston model with parameters according to Table 2. As for the European options we distinguish three cases, namely the Feller condition is satisfied and the variance V_0 takes high (0.25) and low (0.0625) values, and the Feller condition is not satisfied and $V_0 = 0.0348$. Since there exists no exact solution for the American option pricing problem under the Heston model in closed form, we compare our solution with solutions in the literature.

From Tables 6, 7 and 8 we observe that our option prices are nicely in line with those obtained, by different discretisation techniques and corresponding to the finest grid utilised, in the literature. As for European options, it is clear that a nonuniform spatial discretisation leads to a better accuracy than a uniform spatial discretisation in a comparable computational time. Besides a refinement of the grid with a corresponding

Table 8 American put option prices under the Heston model for the parameter set of Table 2 with Feller condition not satisfied and $V_0 = 0.0348$ for uniform and nonuniform discretisation

$m = n$	$\sqrt{M}$	N_t	$S_0 = 90$	$S_0 = 100$	$S_0 = 110$	CPU (s)
Uniform						
30	9	20	10.00394	3.21297	0.93283	0.10353
		40	10.01954	3.20786	0.92938	0.10423
		60	10.01486	3.20619	0.92832	0.10575
34	14	20	10.01864	3.27294	0.96736	0.11151
		40	10.02248	3.26686	0.96235	0.11295
		60	10.02174	3.26507	0.96064	0.11415
38	18	20	10.00670	3.27436	0.96424	0.16270
		40	10.01647	3.26872	0.95899	0.16291
		60	10.01456	3.26676	0.95715	0.16418
Nonuniform						
30	9	20	10.03654	0.17166	0.89000	0.11853
		40	10.03619	3.17516	0.89351	0.11975
		60	10.03614	3.17616	0.89469	0.12285
34	14	20	10.01043	3.22189	0.94367	0.14184
		40	10.00966	3.22181	0.94399	0.14298
		60	10.00945	3.22170	0.94405	0.14399
38	18	20	10.00419	3.20914	0.93029	0.14767
		40	10.00394	3.20945	0.93068	0.15737
		60	10.00382	3.20956	0.93083	0.15794
Fang and Oosterlee (2011)			9.9958	3.2079	0.9280	
Haentjens and in 't Hout (2015)			10.0039	3.2126	0.9305	

increase in the number of patches we increase the number of time steps to get a better accuracy.

To discuss the convergence of the time discretisation, we numerically investigate the behaviour of the global temporal errors as a function of $\triangle t$ which is defined by

$$\text{Temporal error} = \sqrt{\frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(H_{N_t}(0, s_i, v_j) - H_{10\,000}(0, s_i, v_j)\right)^2}, \tag{34}$$

where H_{N_t} is the numerical solution at the spatial nodes $(s_i, v_j) \in [0, 2K] \times [0, V_{max}]$ after N_t time steps, and $H_{10\,000}$ is the corresponding solution for $N_t = 10\,000$ used as an approximant for the exact solution. Figure 5 displays the global temporal errors versus $\triangle t$ for a sequence of eight increasing time steps N_t, namely 25, 50, 100, 200, 400, 800, 1600, 3200 and $N = m \times n = 38 \times 38$ nodal points.

As a first main observation, the global temporal errors are comparable for both cases (Feller condition satisfied and not satisfied). They decrease monotonically as N_t increases or equivalently $\triangle t$ decreases. Concerning the actual convergence behaviour,

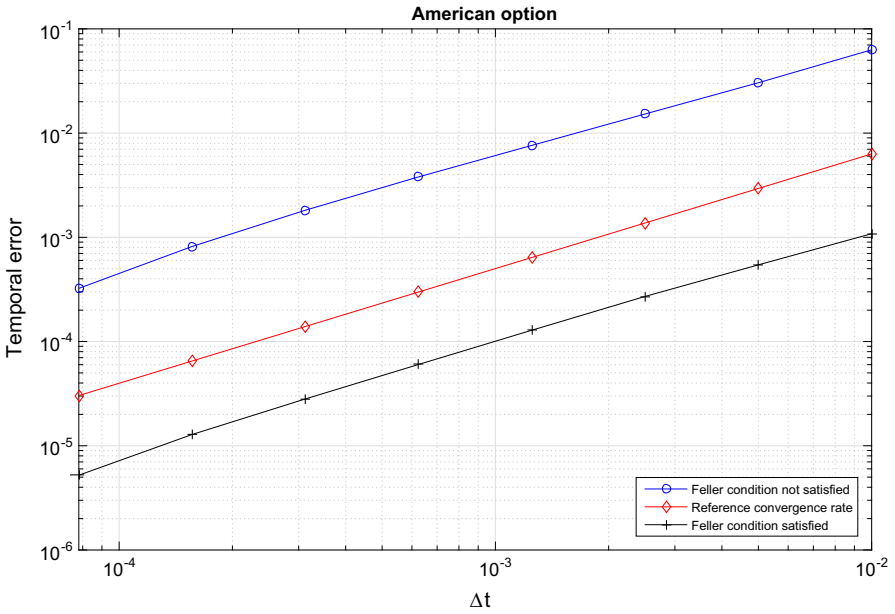

Fig. 5 Temporal error versus $\triangle t$ with 38×38 nonuniform spatial nodes for American put options under the Heston model for the parameter set of Table 2 with Feller condition satisfied and not satisfied

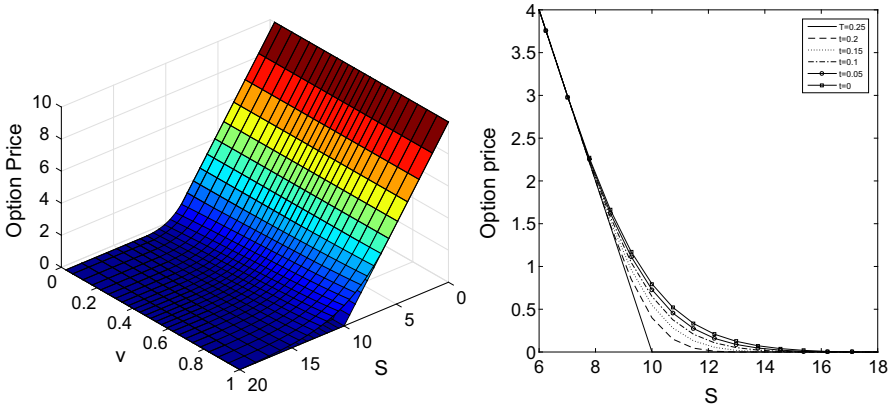

Fig. 6 *Left* Visualisation of the computed option value surface at $t = 0$ (*upper plot*) and the payoff function P (*lower plot*) for the parameter set of Table 2 when the Feller condition is satisfied, *right* evolution of the option price for constant variance $V_0 = 0.25$

the reference line in Fig. 5 represents the function $(\triangle t)^p$ with $p = 1.1$. Hence, it is readily seen that the temporal errors as a function of $\triangle t$ are bounded from above in both cases by $C(\triangle t)^p$ with $p = 1.1$ with some moderate constant $C \approx 0.25$ when the Feller condition is satisfied but a larger value of $C \approx 10$ when the Feller condition is not satisfied. From the observed order of convergence p we conclude that

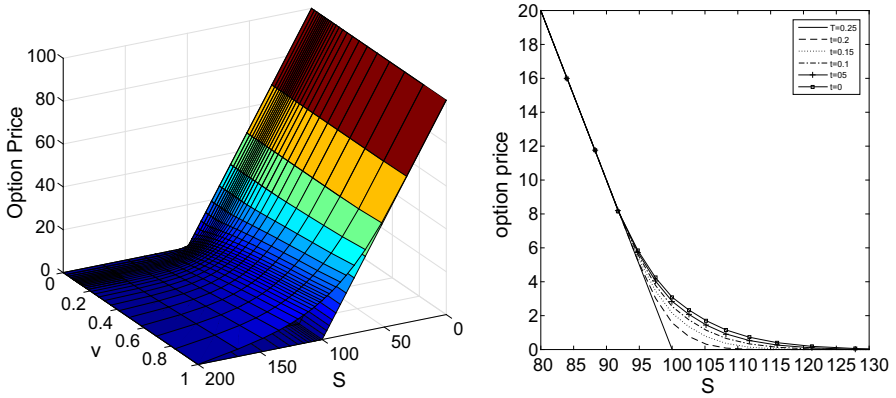

Fig. 7 *Left* Visualisation of the computed option value surface at $t = 0$ (*upper plot*) and the payoff function P (*lower plot*) for the parameter set of Table 2 when the Feller condition is not satisfied, *right* evolution of the option price for constant variance $V_0 = 0.0348$

the combination of the operator splitting method with the Crank–Nicolson scheme is a first-order method in time.

The same remark holds as in the European case that the reported CPU times for the experiments do not include the setup cost for the computations of the differentiation matrices and are obtained using a 3.6 GHz Corei3 processor.

Finally, Figs. 6 and 7 contain on the left the option value surface which lays above the payoff function. On the right we consider a cut of the option value surface at $V_0 = 0.25$ respectively $V_0 = 0.0348$ when time evolves. At maturity the discontinuity at the value of the strike K is clearly visible. This discontinuity is smoothed as time goes backward from $T = 0.25$ to $t = 0$.

7 Conclusion

We proposed a RBF-PU based method to price American options under the Heston model. The free boundary problem formulated as a PDE was transformed into an LCP problem. The RBF-PUM was used for the spatial discretisation. Next, a Crank–Nicolson time discretisation was combined with an operator splitting method. This results in a linear algebraic system with a sparse matrix that has a moderate condition number. We made use of the local adaptivity of the RBF-PUM. Patches are locally refined and have shapes adapted to the local solution behaviour. Further, the node density in each partition is locally adjusted to increase the accuracy. The shape parameter in the RBF affects the accuracy and stability of the method. When the Feller condition is not satisfied the influence of this parameter is larger than when the Feller condition is satisfied. The numerical experiments confirm that a nonuniform discretisation performs better than a uniform nodal distribution. An increase of the number of nodal points and correspondingly an increase of the number of patches also leads to an improvement of the approximation. The effect of the time discretisation is measured by studying the temporal error. For the American option case where the

Crank–Nicolson scheme is combined with an operator splitting method we conclude that we observe a rate of convergence of order one. Finally, the numerical results are in accordance with those found in literature using other numerical methods.

Acknowledgements The first author would like to thank the Department of Applied Mathematics, Computer Science and Statistics of Ghent University and the FWO Scientific Research Network Stochastic Modelling with Applications in Financial Markets for some financial support during his scientific research stay at that department. The authors thank professor in 't Hout for some fruitful discussion.

References

Apel, T., Winkler, G., & Wystup, U. (2001). Valuation of options in Heston's stochastic volatility model using finite element methods. In J. Hakala & U. Wystup (Eds.), *Foreign exchange risk* (pp. 283–303). London: Risk Books.

Babuška, I., & Melenk, J. M. (1997). The partition of unity method. *International Journal for Numerical Methods in Engineering*, *40*(4), 727–758.

Ballestra, L. V., & Pacelli, G. (2011). Computing the survival probability density function in jump-diffusion models: A new approach based on radial basis functions. *Engineering Analysis with Boundary Elements*, *35*(9), 1075–1084.

Ballestra, L. V., & Pacelli, G. (2012). A radial basis function approach to compute the first-passage probability density function in two-dimensional jump-diffusion models for financial and other applications. *Engineering Analysis with Boundary Elements*, *36*(11), 1546–1554.

Ballestra, L. V., & Pacelli, G. (2013). Pricing European and American options with two stochastic factors: A highly efficient radial basis function approach. *Journal of Economic Dynamics and Control*, *37*(6), 1142–1167.

Ballestra, L. V., & Sgarra, C. (2010). The evaluation of American options in a stochastic volatility model with jumps: An efficient finite element approach. *Computers & Mathematics with Applications*, *60*(6), 1571–1590.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, *81*(3), 637–654.

Borici, A., & Lüthi, H.-J. (2002). Pricing American put options by fast solutions of the linear complementarity problem. In E. J. Kontoghiorghes, B. Rustem, & S. Siokos (Eds.), *Computational methods in decision-making, economics and finance* (pp. 325–338). US: Springer.

Buhmann, M. D. (2000). Radial basis functions. *Acta Numerica*, *9*, 1–38. 01.

Buhmann, M. D. (2003). *Radial basis functions: theory and implementations*. New York: Cambridge University Press.

Cavoretto, R., & De Rossi, A. (2012). Spherical interpolation using the partition of unity method: An efficient and flexible algorithm. *Applied Mathematics Letters*, *25*(10), 1251–1256.

Cavoretto, R., & De Rossi, A. (2014). A meshless interpolation algorithm using a cell-based searching procedure. *Computers & Mathematics with Applications*, *67*(5), 1024–1038.

Chen, Y., Gottlieb, S., Heryudono, A., & Narayan, A. (2016). A reduced radial basis function method for partial differential equations on irregular domains. *Journal of Scientific Computing*, *66*(1), 67–90.

Clarke, N., & Parrott, K. (1999). Multigrid for American option pricing with stochastic volatility. *Applied Mathematical Finance*, *6*(3), 177–195.

Cox, J. C., Ingersoll, J. E., & Ross, S. A. (1985). A theory of the term structure of interest rates. *Econometrica*, *53*(2), 385–408.

Dehghan, M., & Shokri, A. (2007). A numerical method for two-dimensional Schrödinger equation using collocation and radial basis functions. *Computers & Mathematics with Applications*, *54*(1), 136–146.

Dehghan, M., & Shokri, A. (2009). Numerical solution of the nonlinear Klein-Gordon equation using radial basis functions. *Journal of Computational and Applied Mathematics*, *230*(2), 400–410.

Duan, Y. (2008). A note on the meshless method using radial basis functions. *Computers & Mathematics with Applications*, *55*(1), 66–75.

Fang, F., & Oosterlee, C. W. (2011). A Fourier-based valuation method for Bermudan and barrier options under Heston's model. *SIAM Journal on Financial Mathematics*, *2*(1), 439–463.

Fasshauer, G. E. (2007). *Meshfree approximation methods with Matlab*. Singapore: World Scientific Publishing Co.

Fasshauer, G. E., Khaliq, A. Q. M., & Voss, D. A. (2004). Using meshfree approximation for multiasset American options. *Journal of the Chinese Institute of Engineers*, *27*(4), 563–571.

Haentjens, T., & in 't Hout, K. J. (2015). ADI schemes for pricing American options under the Heston model. *Applied Mathematical Finance*, *22*(3), 207–237.

Hardy, R. L. (1971). Multiquadric equations of topography and other irregular surfaces. *Journal of Geophysical Research*, *76*(8), 1905–1915.

Heston, S. L. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, *6*(2), 327–343.

Hu, H.-Y., Li, Z.-C., & Cheng, A. H.-D. (2005). Radial basis collocation methods for elliptic boundary value problems. *Computers & Mathematics with Applications*, *50*(1), 289–320.

Hull, J., & White, A. (1987). The pricing of options on assets with stochastic volatilities. *The Journal of Finance*, *42*(2), 281–300.

Ikonen, S., & Toivanen, J. (2004). Operator splitting methods for American option pricing. *Applied Mathematics Letters*, *17*(7), 809–814.

Ikonen, S., & Toivanen, J. (2008). Efficient numerical methods for pricing American options under stochastic volatility. *Numerical Methods for Partial Differential Equations*, *24*(1), 104–126.

Ikonen, S., & Toivanen, J. (2009). Operator splitting methods for pricing American options under stochastic volatility. *Numerische Mathematik*, *113*(2), 299–324.

in 't Hout, K. J., & Foulon, S. (2010). ADI finite difference schemes for option pricing in the Heston model with correlation. *International Journal of Numerical Analysis and Modeling*, *7*(2), 303–320.

Kansa, E. J. (1990a). Multiquadrics—A scattered data approximation scheme with applications to computational fluid-dynamics-I: surface approximations and partial derivative estimates. *Computers & Mathematics with Applications*, *19*(8), 127–145.

Kansa, E. J. (1990b). Multiquadrics—A scattered data approximation scheme with applications to computational fluid-dynamics-II: solutions to parabolic, hyperbolic and elliptic partial differential equations. *Computers & Mathematics with Applications*, *19*(8), 147–161.

Kunoth, A., Schneider, C., & Wiechers, K. (2012). Multiscale methods for the valuation of American options with stochastic volatility. *International Journal of Computer Mathematics*, *89*(9), 1145–1163.

Kwok, Y. K. (1998). *Mathematical models of financial derivatives*. Singapore: Springer.

Larsson, E., & Fornberg, B. (2003). A numerical study of some radial basis function based solution methods for elliptic PDEs. *Computers & Mathematics with Applications*, *46*(5), 891–902.

Larsson, E., Gomes, S. M., Heryudono, A.,& Safdari-Vaighani, A. (2013). Radial basis function methods in computational finance. In *Proceedings of the 13th international conference on computational and mathematical methods in science and engineering, CMMSE 2013*.

Merton, R. C. (1973). Theory of rational option pricing. *Bell Journal of Economics and Management Science*, *4*(1), 141–183.

Oosterlee, C. W. (2003). On multigrid for linear complementary problems with application to American-style options. *Electronic Transactions on Numerical Analysis*, *15*, 165–185.

Oosterlee, C. W., Leentvaar, C. C. W.,& Huang, X. (2005). Accurate American option pricing by grid stretching and high order finite differences. Technical report, DIAM, Delft University of Technology.

O'Sullivan, C., & O'Sullivan, S. (2013). Pricing European and American options in the Heston model with accelerated explicit finite differencing methods. *International Journal of Theoretical and Applied Finance*, *16*(03), 1350015.

Powell, M. J. D. (1992). The theory of radial basis functions approximation in 1990. In W. A. Light (Ed.), *Advances in numerical analysis. Vol. II. Wavelets, subdivision algorithms and radial basis functions* (pp. 105–210). London: Oxford Univ. Press.

Safdari-Vaighani, A., Heryudono, A., & Larsson, E. (2015). A radial basis function partition of unity collocation method for convection-diffusion equations arising in financial applications. *Journal of Scientific Computing*, *64*(2), 341–367.

Seydel, R. (2009). *Tools for Computational Finance* (4th ed.). Berlin, Heidelberg: Springer.

Shcherbakov, V., & Larsson, E. (2016). Radial basis function partition of unity methods for pricing vanilla basket options. *Computers & Mathematics with Applications*, *71*(1), 185–200.

Shepard, D. (1968). A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM national conference, ACM '68* (pp. 517 – 524). ACM: New York, NY, USA.

Tatari, M., & Dehghan, M. (2010). A method for solving partial differential equations via radial basis functions: Application to the heat equation. *Engineering Analysis with Boundary Elements*, *34*(3), 206–212.

Tavella, D., & Randall, C. (2000). *Pricing financial instruments: The finite difference method*. New York: Wiley.

Wendland, H. (1995). Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, *4*(1), 389–396.

Wendland, H. (2002). Fast evaluation of radial basis functions: Methods based on partition of unity. In C. K. Chui, L. L. Schumaker, & J. Stöckler (Eds.), *Approximation theory X: Wavelets, splines, and applications* (pp. 473–483). Nashville: Vanderbilt University Press.

Wendland, H. (2005). *Scattered Data Approximation*. Number 17 in Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, New York.

Zhu, S.-P., & Chen, W.-T. (2010). A new analytical approximation for European puts with stochastic volatility. *Applied Mathematics Letters*, *23*(6), 687–692.

Zhu, S.-P., & Chen, W.-T. (2011). A predictor-corrector scheme based on the ADI method for pricing American puts with stochastic volatility. *Computers & Mathematics with Applications*, *62*(1), 1–26.

Zvan, R., Forsyth, P. A., & Vetzal, K. R. (1998). Penalty methods for American options with stochastic volatility. *Journal of Computational and Applied Mathematics*, *91*(2), 199–218.