Option Pricing with Radial Basis Functions: A Tutorial

Alonso Peña*
Unicredit Banca Mobiliare SpA (UBM),
95 Gresham Street,
London EC2V 7PN, UK

Abstract: In this tutorial we describe the method of radial basis functions (RBF) and how it can be used for the numerical solution of partial differential equations in finance. RBF are conceptually similar to finite difference and finite element methods. However, unlike these, they use globally-defined splines as basis for their approximation. RBF are increasingly popular in the scientific community due to their high accuracy, mesh-free characteristics and ease of implementation. Surprisingly, they are almost unknown in finance. In this contribution we present a hands-on introduction to RBF. Our objectives are: (1) to overview the RBF literature, (2) to present a step-by-step guide for solving the Black-Scholes equation using RBF, and (3) to illustrate the methodology by solving three simple option-pricing problems, including a European, an American and a Barrier option.

1 Introduction

In the past few years a new technique for the numerical solution of partial differential equations (PDEs) has gained prominence in the scientific community: the method of radial basis functions (RBF). RBF are conceptually similar to both the finite difference method (FDM) and the finite element method (FEM). However, unlike the FEM which interpolates the solution using low-order piecewise-continuous polynomials or the FDM which approximates the derivatives of the equations by finite quotients, RBF use globally-defined splines to approximate the PDE solution and its derivatives.[1]

RBFs were conceived independently by Hardy and Duchon during the 1970's as an effective multidimensional scattered interpolation method. During the 1990's, Kansa demonstrated how RBF can be combined with the classic method of collocation[2] for the numerical solution of a variety of elliptic, parabolic and hyperbolic PDEs. Since then, there have been a myriad of applications on RBF in the scientific community (both for interpolation and numerical solution to PDEs) ranging from artificial neural networks to spacecraft design, from air pollution modeling to aerial photography, from medical imaging to hydrodynamics.

Why are RBF so popular? They have a number of distinct advantages: (1) they are natively **high-dimensional**, making them easy to construct in any number of dimensions; (2) they are very **accurate**, with convergence rates increasing with the dimensionality of the problem; (3) they are **mesh-free**, requiring only a an arbitrary set of nodes rather than a full mesh, like in FDM or FEM, to compute an

*Alonso Peña has a PhD degree from the University of Cambridge on finite element analysis and the Certificate in Quantitative Finance (CQF) awarded by 7 city Financial Education plc, London. He is a quantitative analyst at UBM (London Branch). The views presented in this article do not reflect those of Unicredit Banca Mobiliare SpA (UBM). Email: **Alonso.Pena@ubm.unicredit.it**, **alonso.pena@mac.com**

approximate solution; (4) they produce smooth[3] approximations not only for the unknown function but also for its **derivatives**; finally, (5) they can be applied to sophisticated **free-boundary** and **convection-dominated** problems. Predictably, RBF also have some disadvantages: because of their global nature they generate full and sometimes ill-conditioned matrices during intermediate steps in their calculation. Nevertheless, if these are handled appropriately Hon and Kansa (2000), RBF offer an accurate and robust numerical solution method for a wide variety of PDEs.

Despite their increasing success in the scientific community, surprisingly, RBF are almost unknown in finance. With the exception of a handful of articles in the financial literature Fasshauer et al. (2004) Choi and Marcozzi (2001) Hon and Mao (1999) Hon (2002), the vast majority of publications about RBF have been in technical scientific journals. In fact, the first-ever book on RBF was published only last year by the Cambridge University Press Buhmann (2003).

With this in mind, we have prepared this tutorial as a hands-on introduction to the RBF method, with particular emphasis on its application to financial derivatives. We have focused on the application of the method rather than on its theoretical foundations, so many technical details are only covered superficially. The kind reader is encouraged to pursue the topic by following the sources described in **Further Reading** (Appendix 1).

This tutorial is divided in four sections: (1) an overview the RBF literature, including a brief history of the method and an annotated list of applications; (2) the basic mathematical concepts of RBF, both for interpolation and differential equations; (3) a step-by-step guide of how RBF can be used to solve the Black-Scholes equation; and (4) three simple option-pricing problems to illustrate the methodology.

2 An Overview

In 1971 Hardy (1971) Hardy (1990) proposed a novel type of quadratic spline (known today as Hardy's multiquadratic) as a multidimensional scattered interpolation method for modelling the earth's gravitational field. Hardy's key innovation was to suggest approximating the unknown multidimensional surface using a linear combination of basis functions of a *single* real variable, the radial distance, centered at the data points in the domain of interest. Some years later, independently but based on similar ideas, Duchon (1978) proposed a radial function called thin-plate spline.

Many years passed-by without the importance of these functions being properly appreciated. It was not until 1992 when Franke (1992) published an influential review paper evaluating twenty-nine interpolation methods, where Hardy's multiquadratic was ranked the best based on its "accuracy, visual aspect, sensitivity to parameters, execution time, storage requirements, and ease of implementation". In the following years, RBF started to be recognized by the scientific community as powerful approximation technique, with a number of attractive numerical features.

The first of these features is that they are natively **high dimensional** splines, making them easy to construct in any number of dimensions. RBF have the important property that they do not change form as a function of dimension. In addition, the interpolation matrices they generate are uniquely solvable for *any* distribution of distinct nodes[4]. This is in direct contrast with the standard method of multivariate polynomial interpolation, where there is no guarantee that the problem will be uniquely solvable.[5]

The second feature is that they are very **accurate**. In most standard methods the local truncation error of the method depends on the order of the basis function used, and thus is fixed a priori. For example, linear piecewise Lagrange interpolation polynomials, have a convergence order $O(h^2)$. In contrast, RBF have a remarkable property: for equally spaced grids, as shown by Johnson (1997), an approximation rate of $O(h^{d+1})$ is obtained with multiquadratics for suitable smooth f, where h is the density of the collocation points (i.e. the distance between consecutive nodes) and d is the spatial dimension.[6] In other words, as the spatial dimension of the problem increases, the convergence order also *increases* and hence much fewer scattered collocation points are theoretically needed to maintain the same accuracy.[7]

Finally, the third feature is that they are **mesh-fre**e. This means that RBF require only an arbitrary set of nodes rather than a full mesh, like FDM or FEM, to compute an approximate solution. Figure 1 gives a schematic representation of this concept. For an arbitrary[8] domain Ω bounded by $\partial\Omega$, a typical FDM mesh will composed of nodes arranged in an orthogonal grid, while a FEM mesh will require a tesselation of the domain (usually through a Delaunay triangulation) to define the connectivity and subdomain information. RBF, in contrast, requires only a set of nodes covering the domain in some arbitrary distribution, no connectivity or subdomain information is needed.[9] In one dimension this seems trivial, but in higher dimensions this can be extremely attractive, such as basket options or some types of barrier options.

Given all these attractive numerical features, it was natural that RBF found their way to other areas in applied mathematics, particularly the numerical solution of **partial differential equations**. Kansa (1990) Kansa (1990) was the first to suggest using RBF to solve PDEs.[10] In Kansa's method, the differential equation is approximated using globaly-defined RBF. By applying the RBF approximation to the function and its derivatives, the original differential equation is transformed into an algebraic system of equations. The convergence proofs on solving PDEs with RBFs have been given by Wu (1998) and Wu and Hon (2003).

In the last ten years, there has been a dramatic increase in the number of applications of RBF to a wide variety of fields. The Bath Information and Data Service (BIDS) and Ingenta[11], a comprehensive database of academic and professional publications available online, currently lists 453 publications on theoretical and applied research papers on RBF for the period 1994–2004. The following table lists some few examples (Table 1). Many more can be found in the internet sources mentioned in Further Reading (Appendix 1).

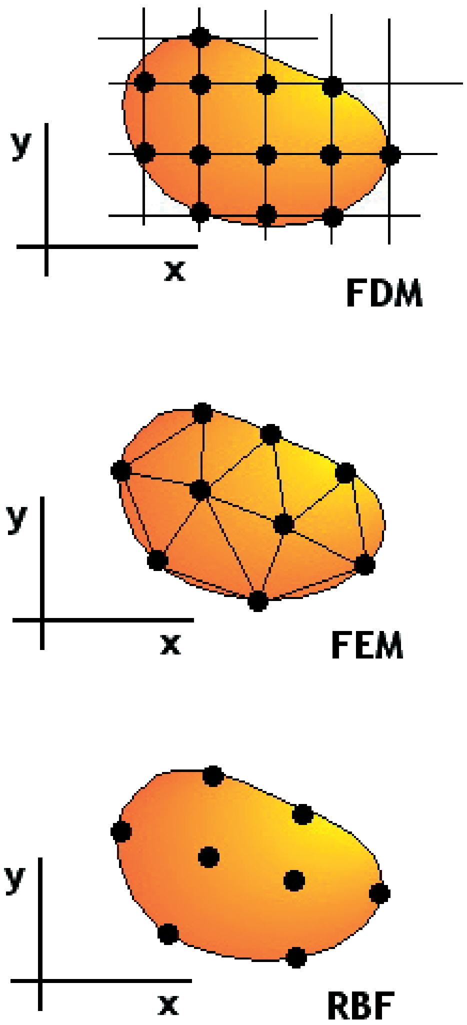

Figure 1: Schematic comparison of the various domain discretization methods discussed. For an arbitrary irregular domain Ω bounded by $\partial\Omega$, a typical FDM mesh will be composed of nodes arranged in an orthogonal grid, while a FEM mesh will require a tesselation of the domain (usually through a Delaunay triangulation) to define the connectivity and subdomain information. RBF, in contrast, requires only a set of nodes covering the domain in any arbitrary distribution, no connectivity or subdomain information is needed.

3 Basic Concepts

3.1 Definitions

A radial basis function $\phi : \mathbb{R}^+ \to \mathbb{R}^+$ is a continuous spline of the form

$$\phi(\|\vec{x} - \vec{\xi}\|) \tag{1}$$

which depends upon the separation distances $r = \|\vec{x} - \vec{\xi}\|$ of a set of nodes $\Xi \in \mathbb{R}^d$, $(\vec{\xi}_j \in \Xi, j = 1, 2, 3, \ldots, N)$, also called "centres" in the literature, which might be either regularly or arbitrarily distributed, upon which the approximation is said to be "supported". Due to their spherical symmetry about the points $\vec{\xi}_j$ these functions are termed radial. The distances $\|\vec{x} - \vec{\xi}_j\|$ are usually measured using the Euclidean norm[12], although other norms are possible (Buhmann 2003).

The most common RBF are: $\phi(r) = r$ (Linear), $\phi(r) = \exp(-a\, r^2)$ (Gaussian), and $\phi(r) = \sqrt{r^2 + c^2}$ (Multiquadratic).[13] Here a and c are positive constants. Note the important feature that ϕ are **univariate** as functions of r, but **multivariate** as functions of $\vec{x}, \vec{\xi} \in \mathbb{R}^d$. If the space dimension is large, this is a significant computational advantage.

3.2 Interpolation

Consider a real-valued function $f : \mathbb{R}^d \to \mathbb{R}$ of d variables that is to be approximated by $s : \mathbb{R}^d \to \mathbb{R}$, given the values $\{f(\vec{\xi}_i) : i = 1, 2, \ldots, N\}$, where $\Xi = \{\vec{\xi}_i : i = 1, 2, \ldots, N\}$ is a set of distinct points in $\mathbb{R}^d$.

We will consider approximations of the form

$$s(\vec{x}) = p_m(\vec{x}) + \sum_{j=1}^{n} \lambda_j \phi(\|\vec{x} - \vec{\xi}_j\|_2),\ \vec{x} \in \mathbb{R}^d,\ \lambda_j \in \mathbb{R} \tag{2}$$

where $p_m(\vec{x})$ is a low degree polynomial. Therefore the radial basis function approximation s is a linear combination of translates of the single radially symmetric function ϕ plus a low degree polynomial. We will denote π_m^d the space of all polynomials of degree at most m in d variables.

The coefficients λ of the approximation s are determined by requiring s to satisfy the interpolation conditions

$$s(\vec{\xi}_i) = f(\vec{\xi}_i) \quad i = 1, \ldots, N \tag{3}$$

together with the side conditions

$$\sum_{j=1}^{n} \lambda_j q(\vec{\xi}_i) = 0 \quad q \in \pi_m^d \quad i = 1, \ldots, N \tag{4}$$

For simplicity, in this paper we will consider only one-dimensional $(d = 1)$ approximation (i.e. our field variable will be the stock price S). Also, we will also limit our analysis to the case of no polynomial term. These two aspects, however, are important characteristics that ought to be considered in a more advanced setting.

With this simplifications, our RBF approximation reduces to

$$s(x) = \sum_{j=1}^{n} \lambda_j \phi(\|x - \xi_j\|_2),\ x \in \mathbb{R},\ \lambda_j \in \mathbb{R} \tag{5}$$

based on the supporting nodes Ξ. Enforcing the interpolation condition, in other words evaluating the previous equation at the nodes, we obtain a linear algebraic system of N equations and N unknowns which can be solved for the λ coefficients.

$$[\phi]\{\lambda\} = \{f\} \tag{6}$$

where $[\phi]$ is an $N \times N$ matrix, and $\{\lambda\}$ and $\{f\}$ are $N \times 1$ vectors, with elements

TABLE 1: SOME RBF APPLICATIONS. SOURCE: BATH INFORMATION AND DATA SERVICE (BIDS) AND INGENTA (WWW.INGENTA.COM).

Field	Reference	Application
Biophysics	Donaldson et al. (1995)	muscle dynamics
Aeronautics	Zakrzhevskii (2003)	optimal spacecraft design
Medical image processing	Fornefett et al. (2001)	magnetic resonance imaging
Atmospheric dynamics	Iaco et al. (2002)	air pollution modeling
Aerial photography	Zala and Barrodale (1999)	forest management
Geomechanics	Wang et al. (2002)	solid-fluid interaction in soil
Civil engineering	Choy and Chan (2003)	rainfall and river discharges
Fluid dynamics	Shu et al. (2003)	Navier-Stokes equations
Structural engineering	Ferreira (2003)	laminated composite beams
European options	Hon and Mao (1999)	Black-Scholes equation
American options	Hon et al. (1997)	optimal exercise boundary
Interest rate derivatives	Choi and Marcozzi (2001)	foreign currency
Basket options	Fasshauer et al. (2004)	American basket options
Financial time series	VanGestel et al. (2001)	neural networks

$$[\phi] = \begin{pmatrix} \phi_{1,1} & \phi_{1,2} & \dots & \phi_{1,N} \\ \phi_{2,1} & \phi_{2,2} & \dots & \phi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N,1} & \phi_{N,2} & \dots & \phi_{N,N} \end{pmatrix} \tag{7}$$

$$\{\lambda\} = (\lambda_1, \lambda_2, \dots, \lambda_N)^T$$
$$\{f\} = (f_1, f_2, \dots, f_N)^T$$

3.3 Differential equations

Let's assume we are given a domain $\Omega \in \mathbb{R}^d$ and a linear elliptic partial differential equation

$$L\,u(\vec{x}) = f(\vec{x}),\ \vec{x} \in \partial\Omega. \tag{8}$$

where L is a differential operator, and u and f are real-valued functions. For simplicity in the description, we consider only Dirichlet boundary conditions of the form

$$u(\vec{x}) = h(\vec{x}),\ \vec{x} \in \partial\Omega. \tag{9}$$

where $h(\vec{x})$ is a given function and $\partial\Omega$ is the boundary of the domain. Thus we are trying to determine u while f and h are fixed.

We can now discuss Kansa's collocation method.[14] We concentrate again in a simple one-dimensional setting. We propose an approximate solution $\tilde{u}$ of the form,

$$\tilde{u}(x) = \sum_{j=1}^{N} \lambda_j \phi(\|x - \xi_j\|_2), \tag{11}$$

where as before the points $\Xi = \xi_1, \xi_2, \dots, \xi_N$ are a set of nodes in Ω. We now collocate $\tilde{u}(x)$ at Ξ. The collocation matrix which arises when matching the differential equation (8) and the boundary conditions (9) to the collocation points Ξ will be of the form

$$[A] = \begin{pmatrix} \phi \\ L\,[\phi] \end{pmatrix} \tag{12}$$

where the two blocks are composed of entries:

$$\phi_{ij} = \phi(\|\xi_i - \xi_j\|_2),\ \ \xi_i \in \Xi^B, \xi_j \in \Xi \tag{13}$$

$$L\phi_{ij} = L\phi(\|\xi_i - \xi_j\|_2)\ \xi_i \in \Xi^I, \xi_j \in \Xi \tag{14}$$

while the right hand side vector will be

$$\{b\} = \begin{pmatrix} f \\ h \end{pmatrix} \tag{15}$$

Note that the set Ξ is split into a set of Ξ^I interior points and Ξ^B of boundary points. The problem is well-posed if the linear system $[A]\{\lambda\} = \{b\}$ has a unique solution. We note that a change in the boundary conditions (9) is as simple as changing a few rows of matrix $[A]$ as well as on the right hand side vector $\{b\}$.

4 Radial Basis Functions for Option Pricing

We now turn our attention to finance. To simplify matters we have developed an eight step procedure, as a sort of "cooking recipe" to apply RBF to price simple financial derivatives. Obviously, this procedure will need to be modified when used for more complex PDEs, but it offers a framework that is general enough to accommodate future developments. In this section, we follow closely the work of Hon and Mao (1999). Our point of departure is the dividend-free one factor Black-Scholes equation:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0 \tag{16}$$

where $V(S,t)$ is the option price, r is the risk free interest rate, σ the volatility and S the stock price Wilmott et al. (1995). We consider a final condition (the payoff) and boundary conditions of the form $V(S,T) = V_g(S),\ \ V_a(a,t) = \alpha(t), V_b(b,t) = \beta(t)$ where T is the expiry time. We consider a truncated domain $\Omega = [a,b] \times [0,T]$.

Step 1: Construct the nodal set Ξ.

Before we start, we propose a change of variable $S = e^x$ in order to transform the Black-Scholes equation into a constant-coefficient PDE in the domain $\Omega \in [x \times t], x \in [\log(a), \log(b)], t \in [0,T]$. Our resulting log-transformed governing equation is

$$\frac{\partial U}{\partial t} + \frac{1}{2}\sigma^2\frac{\partial^2 U}{\partial x^2} + \left(r - \frac{1}{2}\sigma^2\right)\frac{\partial U}{\partial x} - rU = 0 \tag{17}$$

In the new variables, the final condition and boundary conditions can be written as

$$U(x, T) = g(x) \tag{18}$$

$$U(ln(a), t) = \alpha(t) \tag{19}$$

$$U(ln(b), t) = \beta(t) \tag{20}$$

We now discretize our domain Ω with N nodes ($N - 1$ divisions) in the x-axis and M time step in the t-axis. Therefore, in the space domain we have a grid $\Xi = (\xi_1, \xi_2, \ldots, \xi_N)$. The time domain, in turn, is discretized into a set $\Theta = (\theta_0, \theta_1, \ldots, \theta_M)$, where $\theta_0 = T$ and $\theta_M = 0$. Note that the discretisation has equidistant nodes in the x-axis, but non-equidistant nodes in the S-axis.

Step 2: Choose a specific form of ϕ and calculate its derivatives.

As mentioned in Section 2, there are many types of RBF such as Gaussians, thin-plate splines, multiquadratics and inverse multiquadratics. In our example we select Hardy's multi-quadratic (MQ), which can be written in terms of the log stock price x (the independent variable) as

$$\phi(\|x - \xi_j\|_2) = \sqrt{(x - \xi_j)^2 + c^2} \tag{21}$$

based on nodes Ξ and the user-defined parameter c.[15] The derivatives of ϕ can be easily calculated as:

$$\frac{\partial\phi}{\partial x} = \frac{(x - \xi_j)}{\sqrt{(x - \xi_j)^2 + c^2}} \tag{22}$$

$$\frac{\partial^2\phi}{\partial x^2} = \frac{1}{\sqrt{(x - \xi_j)^2 + c^2}} - \frac{(x - \xi_j)^2}{(\sqrt{(x - \xi_j)^2 + c^2})^3} \tag{23}$$

Step 3: Approximate the unknown function and its derivatives.

Using these definitions, we can now propose an approximation for the option price U as:

$$U(x, t) = \sum_{j=1}^{N}\lambda_j(t)\,\phi(\|x - \xi_j\|_2) \tag{24}$$

based on nodes Ξ. Note that in contrast to equation (9) as in Section 3.2, the coefficients λ are now function of time. The derivatives of this approximation are:

$$\frac{\partial U}{\partial x} = \sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial\phi(\|x - \xi_j\|_2)}{\partial x} \tag{25}$$

$$\frac{\partial^2 U}{\partial x^2} = \sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial^2\phi(\|x - \xi_j\|_2)}{\partial x^2} \tag{26}$$

$$\frac{\partial U}{\partial t} = \sum_{j=1}^{N}\frac{\partial\lambda}{\partial t}\phi(\|\xi_i - \xi_j\|_2) \tag{27}$$

Note how the differential operators are applied directly to ϕ. Thus, in contrast to the FDM, in the RBF we never discretize the differential operators. By direct substitution of the expansions for U, $\frac{\partial U}{\partial x}$, $\frac{\partial^2 U}{\partial x^2}$ and $\frac{\partial U}{\partial t}$ into the log-transformed Black-Scholes equation, we obtain

$$\begin{aligned}&\sum_{j=1}^{N}\frac{\partial\lambda_j}{\partial t}\phi(\|x - \xi_j\|_2) + \frac{1}{2}\sigma^2\sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial^2\phi(\|x - \xi_j\|_2)}{\partial x^2}\\ &+ \left(r - \frac{1}{2}\sigma^2\right)\sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial\phi(\|x - \xi_j\|_2)}{\partial x} - r\sum_{j=1}^{N}\lambda_j\,\phi(\|x - \xi_j\|_2) = 0\end{aligned} \tag{28}$$

based on the node set Ξ.

Step 4: Collocate.

We can now collocate the previous equation at the nodes Ξ, which in our case we have chosen to be the same as the nodes of the interpolation, and obtain the following system of N equations

$$\begin{aligned}&\sum_{j=1}^{N}\frac{\partial\lambda_j}{\partial t}\phi(\|\xi_i - \xi_j\|_2) + \frac{1}{2}\sigma^2\sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial^2\phi(\|\xi_i - \xi_j\|_2)}{\partial x^2}\\ &+ \left(r - \frac{1}{2}\sigma^2\right)\sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial\phi(\|\xi_i - \xi_j\|_2)}{\partial x} - r\sum_{j=1}^{N}\lambda_j\,\phi(\|\xi_i - \xi_j\|_2) = 0\\ &i = 1, \ldots, N\end{aligned} \tag{29}$$

which can be written in terms of the radial distances as

$$\begin{aligned}&\sum_{j=1}^{N}\frac{\partial\lambda_j}{\partial t}\phi(r_{ij}) + \frac{1}{2}\sigma^2\sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial^2\phi(r_{ij})}{\partial x^2}\\ &+ \left(r - \frac{1}{2}\sigma^2\right)\sum_{j=1}^{N}\lambda_j(t)\,\frac{\partial\phi(r_{ij})}{\partial x} - r\sum_{j=1}^{N}\lambda_j\,\phi(r_{ij}) = 0\\ &i = 1, \ldots, N\end{aligned} \tag{30}$$

where, as before, we have defined $r_{ij} = \|\xi_i - \xi_j\|_2$. The above equations can be rendered considerably clearer if we use matrix algebra. If we define the following matrix-vector products

$$[\phi]\{\lambda\} = \begin{pmatrix}\phi_{1,1} & \phi_{1,2} & \ldots & \phi_{1,N}\\ \phi_{2,1} & \phi_{2,2} & \ldots & \phi_{2,N}\\ \vdots & \vdots & \ddots & \vdots\\ \phi_{N,1} & \phi_{N,2} & \ldots & \phi_{N,N}\end{pmatrix}\begin{pmatrix}\lambda_1\\ \lambda_2\\ \vdots\\ \lambda_N\end{pmatrix} \tag{31}$$

$$[\phi']\{\lambda\} = \begin{pmatrix}\phi'_{1,1} & \phi'_{1,2} & \ldots & \phi'_{1,N}\\ \phi'_{2,1} & \phi'_{2,2} & \ldots & \phi'_{2,N}\\ \vdots & \vdots & \ddots & \vdots\\ \phi'_{N,1} & \phi'_{N,2} & \ldots & \phi'_{N,N}\end{pmatrix}\begin{pmatrix}\lambda_1\\ \lambda_2\\ \vdots\\ \lambda_N\end{pmatrix} \tag{32}$$

$$[\phi'']\{\lambda\} = \begin{pmatrix} \phi''_{1,1} & \phi''_{1,2} & \dots & \phi''_{1,N} \\ \phi''_{2,1} & \phi''_{2,2} & \dots & \phi''_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \phi''_{N,1} & \phi''_{N,2} & \dots & \phi''_{N,N} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{pmatrix} \tag{33}$$

where the elements $\phi_{i,j}$ represent the function evaluated at point $\|\xi_i - \xi_j\|_2$, and $\phi'_{i,j}$ and $\phi''_{i,j}$ stand for the first and second derivative of the RBF, respectively. We can write in matrix notation the fully disctretized log-transformed Black-Scholes equation as

$$[\phi]\{\dot{\lambda}\} - \frac{1}{2}\sigma^2[\phi'']\{\lambda\} + \left(r - \frac{1}{2}\sigma^2\right)[\phi']\{\lambda\} - r[\phi]\{\lambda\} = 0 \tag{34}$$

based on nodes Ξ, where the vector $\{\dot{\lambda}\}$ represents $\frac{\partial \lambda_j}{\partial t}$. We now put this equation in standard matrix form by re-arranging terms:

$$\{\dot{\lambda}\} = -[\phi]^{-1}\frac{1}{2}\sigma^2[\phi'']\{\lambda\} + \left(r - \frac{1}{2}\sigma^2\right)[\phi']\{\lambda\} - r[\phi]\{\lambda\} \tag{35}$$

If we define the matrix $[P]$ as

$$[P] = r[I] - \frac{1}{2}\sigma^2[\phi]^{-1}[\phi''] - \left(r - \frac{1}{2}\sigma^2\right)[\phi]^{-1}[\phi'] \tag{36}$$

we obtain

$$\{\dot{\lambda}\} = [P]\{\lambda\}. \tag{37}$$

from which it is clear that our previous equation represent a system of ODEs.

Step 5: Approximate the time derivative.

We now propose an approximation for the time derivative. Many types of time-discreizations are possible based on finite differences (e.g. Euler), Runge-Kutta, or analytical methods Iserles (1996). Using the θ-method:

$$\{\lambda^k\} = \{\lambda^{k-1}\} - \Delta t[P](\theta\lambda^{k-1} + (1-\theta)\lambda^k) \qquad k = 1, \dots, M \tag{38}$$

we obtain the linear system:

$$([I] + \Delta t(1-\theta)[P])\{\lambda^k\} = ([I] - \Delta t\theta[P])\lambda^{k-1} \qquad k = 1, \dots, M \tag{39}$$

Defining the matrices:

$$\begin{aligned} [H] &= [I] + \Delta t(1-\theta)[P] \\ [G] &= [I] - \Delta t\theta[P] \end{aligned} \tag{40}$$

we arrive at

$$[H]\{\lambda^{k-1}\} = [G]\{\lambda^k\} \qquad k = 1, \dots, M \tag{41}$$

where $[H]$ and $[G]$ are $N \times N$ matrices, and $\{\lambda^k\}$ and $\{\lambda^{k+1}\}$ are $(N \times 1)$ vectors. Note that at each iteration, the left hand side can be reduced to a vector by performing the matrix-vector product.

Step 6: Apply the boundary conditions $\alpha(t)$ and $\beta(t)$.

In our problem we have time-varying Dirichlet-type boundary conditions of the form

$$U(ln(a), t) = \alpha(t) \tag{42}$$

$$U(ln(b), t) = \beta(t) \tag{43}$$

which need to be enforced at each time step during the iteration. If you recall Section 3.3, we applied these by collocating at the boundary knots Ξ^B. In this example, however, we will follow a different approach known as "Boundary Update Procedure" (BUP) due to Hon and Mao (1999). This procedure is better adapted to a variety of numerical time-integration schemes, such as the θ-method which we follow here.

The basic idea of the BUP is to update λ^k at each timestep. To do this, we need to compute the un-adjusted RBF approximation U^* implied by the current vector λ^k.

$$\begin{pmatrix} \phi_{1,1} & \phi_{1,2} & \dots & \phi_{1,N} \\ \phi_{2,1} & \phi_{2,2} & \dots & \phi_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N,1} & \phi_{N,2} & \dots & \phi_{N,N} \end{pmatrix} \begin{pmatrix} \lambda_1^k \\ \lambda_2^k \\ \vdots \\ \lambda_N^k \end{pmatrix} = \begin{pmatrix} U_1^* \\ U_2^* \\ \vdots \\ U_N^* \end{pmatrix} \tag{44}$$

We then modify the appropriate elements from this vector, such that the boundary conditions are satisfied, e.g. $U_1^* = \alpha$ and $U_N^* = \beta$.

$$\{\alpha(t)U_2^* \dots U_{N-1}^*\beta(t)\}^T \tag{45}$$

Using this new vector U^* we then compute the adjusted λ^* implied by performing the matrix-vector multiplication $\phi^{-1}\, U^*$.

$$\{\lambda^*\} = [\phi^{-1}]\{U^*\} \tag{46}$$

We finally assign λ^* to λ^k and continue the iterations.

Step 7: Apply the initial condition $g(S)$.

To start iteration (58) we need the RBF λ coefficients for the payoff $g(S)$, as defined at the beginning of the section. As in Section 3.2, we can easily construct

$$\{g\} = [\phi]\{\lambda^0\} \tag{47}$$

from which the $\{\lambda^0\}$ coefficients can be calculated. Note that the payoff will be evaluated in terms of the log transformed variable x and not of the original variable S. For instance, for a call we would have $g(x) = \max(e^x - E, 0)$.

Step 8: Integrate the equation in time.

We are now in a position of solving equation (34). We need to integrate backward in time by iterating

$$[H]\{\lambda^{k-1}\} = [G]\{\lambda^k\} \qquad k = 1, \dots, M \tag{48}$$

This is the matrix that will be the heart of the main loop in our code. To start the time iteration we use the value of the $\{\lambda\}^0$ coefficients calculated in Step 7 and the boundary update procedure at each iteration from Step 6. At the end of this process, therefore, we will have M $\{\lambda\}$ vectors, each with N components. Note that the coefficient matrix $[G]$ does not change in time, while $[H]\{\lambda^{k-1}\}$ changes at each time step. This suggests using a factorization of matrix $[G]$ for various RHS vectors and back substitution. In our implementation we have used a LU factorisation (Crout) Flowers (2000) Press et al. (2002).

Once we have determined the $\{\lambda\}^k$ vectors, we can calculate V, Δ and Γ by directly evaluating the expressions

$$V(x^*, \theta_k) = \sum_{j=1}^{N} \lambda_j^k \, \phi(\|x^* - \xi_j\|_2) \tag{49}$$

$$\Delta(x^*, \theta_k) = \sum_{j=1}^{N} \lambda_j^k \, \frac{\partial \phi(\|x^* - \xi_j\|_2)}{\partial x} \tag{50}$$

$$\Gamma(x^*, \theta_k) = \sum_{j=1}^{N} \lambda_j^k \, \frac{\partial^2 \phi(\|x^* - \xi_j\|_2)}{\partial x^2} \tag{51}$$

for any particular value $x^* \in [a, b]$ at the times θ_k for $k = 1, \ldots, M$.

This concludes the eight step procedure. Now we apply it to some simple option pricing examples.

5 Option Pricing Examples

5.1 European Call

As a first example we consider a simple vanilla European option. This problem is useful as it will serve as a benchmark to investigate the accuracy of the RBF approximation. We want to price a European Call option with T=1, E=15, σ =0.30, and r = 0.05, for a domain $\Omega = [1, 30]$. The appropriate final and boundary conditions for this problem are:

$$g(S) = \max(S - E, 0)$$
$$\alpha(t) = 0$$
$$\beta(t) = S - Ee^{-r(T-t)}$$

The eight step procedure described before was followed, implemented in C++. The code has been tested in the following environments: Xcode Development Tools 1.1, Apple Computer Inc. (Macintosh), GNU g++ 3.1.4 (Linux), Microsoft Visual Basic 6.0 (Windows). All the computations presented in this article were done on an Apple Macintosh iBook G4. The graphs were done in (Wolfram Research) *Mathematica*, based on the results from the C++ code. The total computation time was less than one second.

Our results for $V(S, t)$, $\Delta(S, t)$ and $\Gamma(S, t)$ are presented in Figures 2, 3 and 4. They demonstrate the well-known surfaces for the option price and its derivatives. As evidenced in the graphs, even though a small number of nodes was used, the surface of these functions is smooth, with the exception of small regions near the discontinuities at $t = T$. Note that the RBF approximation is a function of both S and t (i.e. valid in the whole domain), in contrast to a finite difference solution which provides a pointwise approximation (i.e. valid only at the nodes).

In order to assess the accuracy of the numerical solution, we have calculated the absolute relative error $e(S, t)$, as

$$e(S, t) = \frac{|V_{rbf}(S, t) - V_{exact}(S, t)|}{|V_{exact}(S, t)|}$$

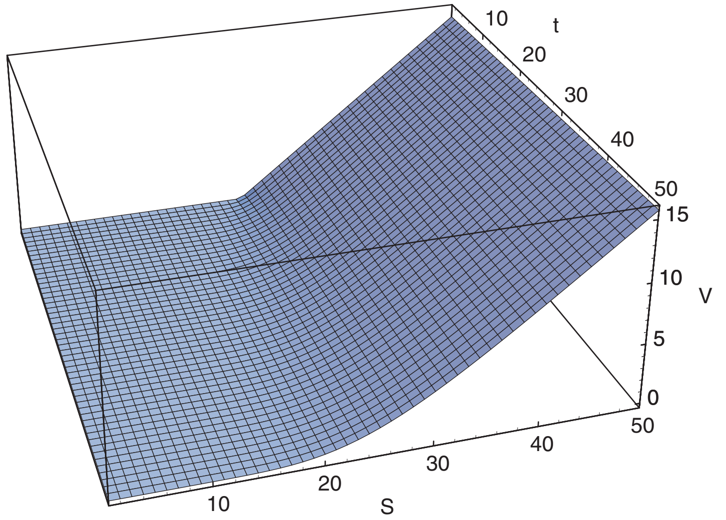

Figure 2: RBF approximation to European Call option price *V(S,t)*.

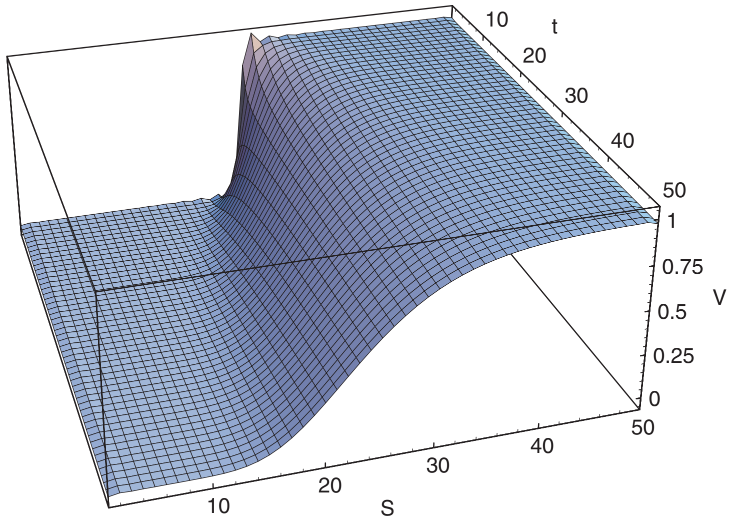

Figure 3: RBF approximation to European Call $\Delta(S, t)$.

TABLE 2: RBF EUROPEAN CALL OPTION PRICE RESULTS, $V(E,0)$ WITH T = 1, E = 15, σ = 0.30, AND r = 0.05, FOR A DOMAIN Ω = [1, 30]. $V_{EXACT}(E,0)$ = 2.1347.

N, M	30	40	50	60	70	80
$V_{RBF}(E,0)$	2.12391	2.12627	2.12766	2.13007	2.13205	2.13318
$e_{RBF}(E,0)$	0.5055%	0.3947%	0.3298%	0.2169%	0.1240%	0.0710%

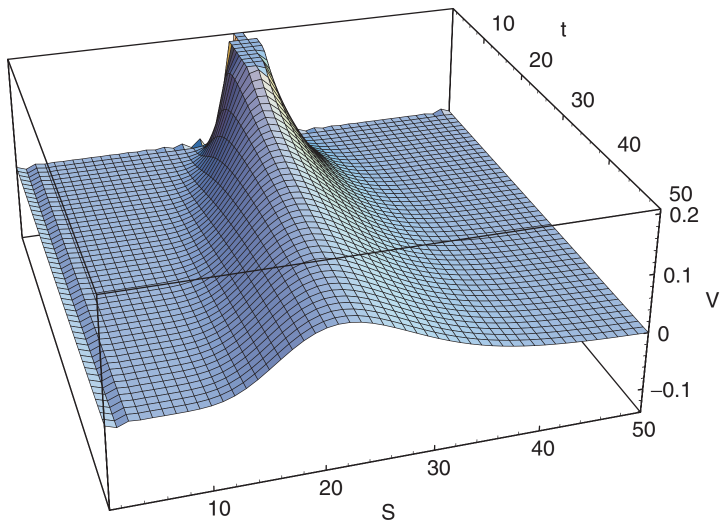

Figure 4: RBF approximation to European Call $\Gamma(S,t)$.

TABLE 3: COMPARISON OF AMERICAN PUT OPTION PRICES USING VARIOUS TECHNIQUES: THE FOUR-POINT METHOD OF GESKE AND JOHNSON (1984), THE SIX-POINT RECURSIVE SCHEME OF HUANG ET AL. (1996), THE THREE-POINT MULTI-PIECE EXPONENTIAL BOUNDARY METHOD OF JU ET AL. (1998) AND THE RBF APPROXIMATION FROM THIS PAPER. THE PARAMETERS ARE T = 3, E = 100, σ = 0.20, AND R = 0.08. THE "EXACT" VALUE, BASED ON A BINOMIAL MODEL WITH 10,000 TIMESTEPS, IS 6.9320.

Reference	Geske-Johnson	Huang et al.	Ju et al.	$V_{RBF}(E,0)$
$V(E,0)$	6.9266	6.7859	6.9346	6.9305
$e(E,0)$	0.0778%	2.1076%	0.0375%	0.0212%

and compared it with the analytical solution from the closed-form solution from the Black-Scholes formula. Our results are presented in Table 2. We have calculated the RBF approximation for various values of N and M and calculated their respective option value $V_{rbf}(E,0)$ and associated error $e(E,0)$, i.e. the approximation error at-the-money and at $t = 0$. As expected, the error decreases for larger N. Even for a modest number of nodes, $N = 30$, the RBF approximation error is only half a percent or 50 basis points (bps). When $N = 80$ nodes are used, the RBF error is only 7 bps.

5.2 American Put

Our second pricing example is an American Put option. We will use this to demonstrate how RBF can capture the early exercise feature of this type of options. We want to price an option with three years to expiry, T = 3, E = 100, $\sigma = 0.20$, and $r = 0.08$, for a domain $\Omega = [1, 200]$, N=50, M=50.

The appropriate initial and boundary conditions for this problem are:

$$g(S) = \max(E - S, 0)$$
$$\alpha(t) = Ee^{-r(T-t)}$$
$$\beta(t) = 0$$

In addition, the early exercise feature leads to the free boundary condition

$$V(S_f(t), t) = \max(E - S_f(t), 0)$$
$$\frac{\partial V(S_f(t), t)}{\partial S} = -1$$

where $S_f(t)$ is the free boundary.

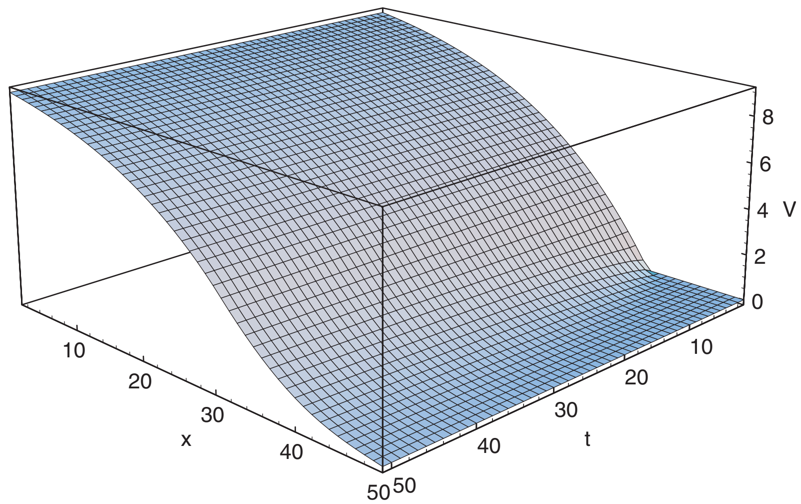

Figure 5: RBF approximation for an American Put option price $U(x,t)$. Note that the results are plotted using the log transformed price x.

The eight step procedure was again followed. The total computation time was less one second. In our C++ implementation, following Wilmott (2000) and Joshi (2004), we used an update procedure in which, at each time-step, the approximated solution is verified to be always larger than the payoff. Thus, in case that $V_{rbf}(S, t_k) > g(S)$ we simply substitute the payoff for the solution at those particular nodes.

Our RBF approximation for $V(S, t)$ is presented in Figure 5, which is plotted using the logarithmic variable x. Figure 6 illustrates the RBF solution and the payoff, which is plotted again using the logarithmic variable x. Here we can see how the free-boundary condition is satisfied by having a smooth transition at the contact point between the option price and the payoff - the so-called high-contact condition.[16]

In order to investigate the accuracy of the RBF approximation we calculated the relative error e between our approximation and the following examples from the literature: the four-point method of Geske and Johnson (1984), the six-point recursive scheme of Huang et al. (1996), the three-point multi-piece exponential boundary method of Ju (1998). They all represent the value at-the-money of an American Put option with three years to expiry, $T = 3$, $E = 100$, $\sigma = 0.20$, and $r = 0.08$. The "exact" value, based on a binomial model with 10,000 timesteps, is 6.9320. The results of the comparison are presented in Table 3. These correspond to relative errors of 7.78, 210.76, 3.75 bps, respectively. The RBF represents an error of 2.12 basis points, for $N = M = 50$ and a domain $\Omega = [1, 200]$.

Finally, in order to compare our approximation with that from another numerical method, we price an option with five months to expiry, $T = 5/12$, $E = 50$, $\sigma = 0.40$, and $r = 0.10$, for a domain $\Omega = [1, 100]$, $N = 50$, $M = 50$. The RBF value is 4.2932, while Seydel (2002) obtained 4.284 using a sophisticated Crank-Nicholson and Succesive Over Relaxation (SOR) approach.

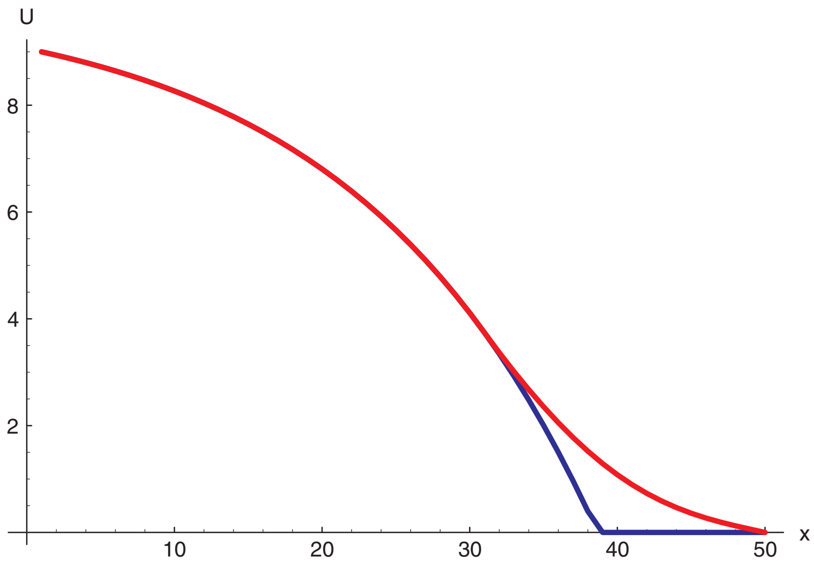

Figure 6: RBF approximation for an American Put option price, demonstrating the smooth transition between the option price $U(x,0)$ (red) and the payoff $g(x)$ (blue). Note that the results are plotted using the log transformed price x.

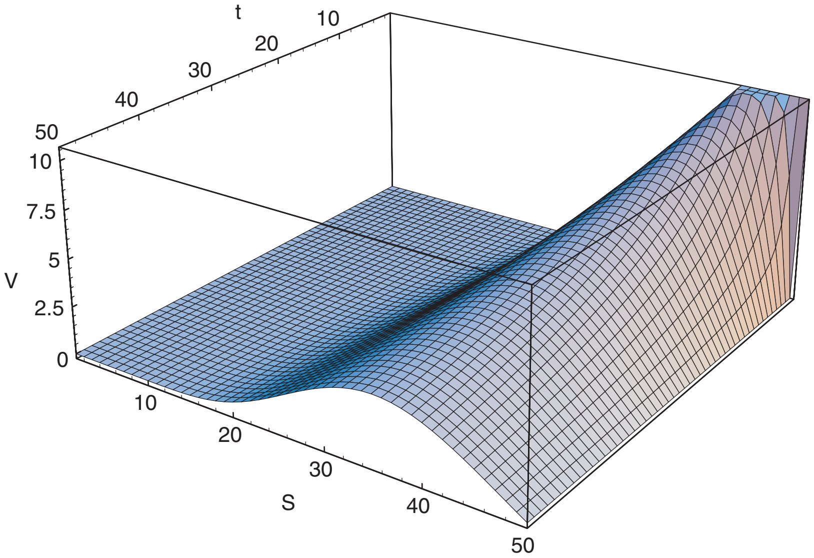

Figure 7: RBF approximation for Barrier Up-and-Out Call option price $V(S,t)$.

5.3 Up-and-out Barrier

Our final example is a continuous Up-and-Out Call Barrier option. This example is interesting as it will illustrate how RBF can capture the dramatic discontinuities due to the payoff and boundary conditions. This type of option has a payoff that is zero if the asset crosses some predefined barrier $B > S_0$ at some time in $[0, T]$. If the barrier is not crossed then the payoff becomes that of a European call (Higham 2004, p. 190).

Even though in practice the asset price is only observable at some discrete times, usually on a daily basis, making barrier options discrete rather than continuous (See Joshi, 2004). It is also interesting to note that up-and-out barriers have a limited upside - the payoff cannot exceed $B - E$, and hence can be bought for much less than the European version. So, if a speculator has very strong views of on how he/she believes an asset will move, he/she can make more money by purchasing an option which expresses a priori those views. (See Joshi, 2004).

We want to price an option with one year to expiry, $T = 1$, $E = 15$, $\sigma = 0.30$, and $r = 0.05$, for a domain $\Omega = [1, 30]$, $N = 40$, $M = 40$ and the barrier level at B = 30. The appropriate initial and boundary conditions for this problem are:

$$g(S) = \max(S - E, 0)$$
$$\alpha(t^+) = 0 \qquad t^+ < T$$
$$\beta(t^+) = 0 \qquad t^+ < T$$

The eight step procedure was followed. The total computation time was less than one second. Our results for $V(S,t)$, $\Delta(S,t)$ and $\Gamma(S,t)$ are presented in Figures 7, 8 and 9. They demonstrate smooth surfaces for the option price surface and the Greeks.

For the option price, note in particular the dramatic change in curvature just before T, when the "out" boundary condition was applied, and how at $t = 0$ the option price is largest in the midpoint between the strike E and the barrier B. For the Δ, note how the function changed from highly discontinuous at $t = T$ into a smooth surface at $t = 0$, from which sensible hedging parameters can be obtained. For $S < E$ the Δ is zero or positive, becoming negative close to the barrier B. Finally, for the Γ note the large discontinuity at the strike, just like a European option. When it approaches $t = 0$ the Γ is positive away from the barrier and negative close to it.

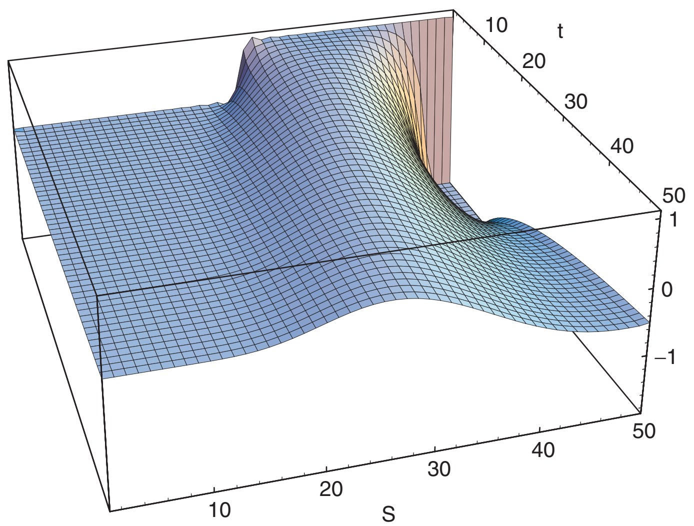

Figure 8: RBF approximation for Barrier Up-and-Out Call option $\Delta(S,t)$.

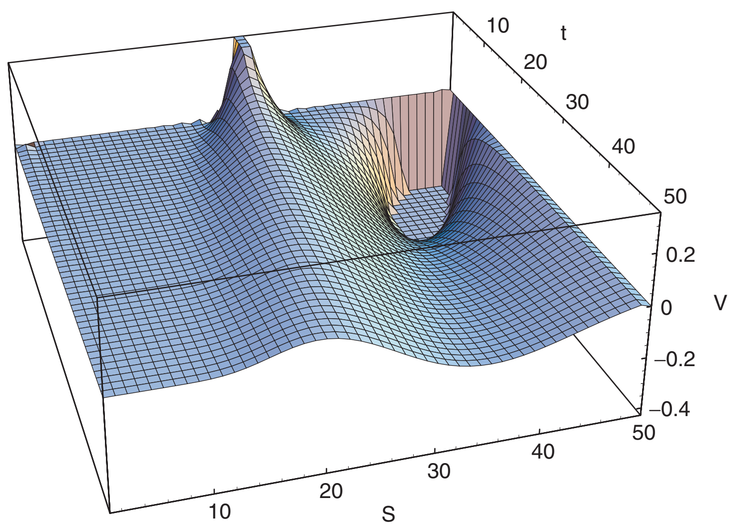

Figure 9: RBF approximation for Barrier Up-and-Out Call option $\Gamma(S,t)$.

In order to assess the accuracy of our numerical approximation with compare it with that from the analytical solution (Wilmott 2000) and with a Monte Carlo with antithetic variates as implemented in the recent textbook by Higham (2004). The exact Black-Scholes value at the money and $t = 0$ is 1.8187, while that of the MC (with $\Delta t = 10^-3$) a confidence interval of [1.7710, 1.8791] for $M = 10^4$, [1.8161, 1.8505] for $M = 10^5$, and [1.8229, 1.8338] for $M = 10^6$. The RBF approximation ($N = 40$, $M = 40$), in turn, was 1.82173, representing a relative error less than 17 bps.

6 Discussion

The preceding examples illustrate how RBF can produce accurate approximations for the Black-Scholes equation in a variety of situations. Even though our results do not constitute a formal numerical comparison between RBF and other techniques, our findings suggest that accurate option prices can be obtained with a small number of nodes. In particular, we have seen how for the European option example, the option price were computed to an accuracy of 7 bps with $N = 80$. We have also seen how, by a simple modification of the code, the early exercise feature of an American option can be easily captured and its free boundary calculated. Finally, in the case of continuous Up-and-Out Barrier option, we have seen how the large discontinuities in the initial and boundary conditions can be captured successfully and an stable option price computed to an accuracy of less than 17 bps.

That is all very good, but so far we have been focusing in simple examples in option pricing. Where do we go from here? As we said before, our eight step procedure is *very* basic and many important extensions ought to be incorporated to adapt it to a real-world setting. We suggest three.

First, to use a different type of ϕ. In our analysis we have relied solely on Hardy's multiquadratic function. However, there are many other types of RBF. Amongst the family of globally-supported RBF we have: MQ, Gaussians, Thin-plate splines. But, there is a large family of compactly-supported RBF Schaback (1995) with the nice property of avoiding the problem of having full matrices. In fact they generate sparse and well-conditioned matrices, though at the expense of loosing the spectral convergence rates. They provide some guidelines on the proper choice of the support radius and the smoothness of the function based on numerical experiments.

Second, use matrix preconditioning techniques. As we have already seen, the radial basis function interpolation matrix [A] is usually ill-conditioned when Ξ is a large set and ϕ is the multiquadratic. In fact, the matrix condition number may be improved before beginning the computation of the interpolation coefficients, which is the standard approach when linear systems become numerically intractable, such as in spline interpolation and in finite element methods Braess (1997).

Finally, instead of using Kansa's straight collocation method, we suggest experimenting with a promising new approach called Hermite collocation Fasshauer (1996), which has the benefit of generating symmetric matrices. Power and Barraco (2002) present a detailed numerical comparison between unsymmetric and symmetric RBF collocation methods for the numerical solution of PDEs. Even though, the unsymmetric method is simplier to implement, the authors show that the symmetric method coefficient matrix is in general easier to solve than the unsymmetric method.

7 Conclusion

RBF is a young field and therefore many questions about it don't have yet clear-cut answers—a sort of *terra incognita* with all its promises and dangers. Where exactly will RBF fit in the great map of numerical analysis is still unclear. At worst they are just one more numerical method. At best they open the door to intriguing new opportunities for the numerical solution of multidimensional PDEs—a field which we tend to disregard as impractical due to the increase in computation time as function of dimension—the so-called "curse of dimensionality". As Wilmott (2000) has explained:

> "The hardest problems to solve are those with both high dimensionality, for which we would like to use Monte Carlo simulation, and early exercise, for which we would like to use finite-difference methods. There is currently no numerical method that copes well with such a problem." (p. 177)

As we have seen RBF work with both early exercise and higher dimensions. Maybe in the future RBF could play a role, either on their own or combined with other techniques, to exorcise the "curse of dimensionality".

8 Acknowledgements

I would like to express my gratitude to Dr Paul Wilmott, Editor-in-Chief, for offering me the opportunity to publish this contribution.

I would like to thank Wouter Hendrickx, University of Antwerp, Belgium for his expert advice on RBF. Sebastien Lleo, Imperial College of Science, Technology and Medicine, London, for kindly reviewing the manuscript in great detail. Also John de Courcy-Bower, Barclays Capital, Tokyo, Japan, for several valuable discussions during the initial stages of this project. The staff of the Moore Library, Department of Applied Mathematics and Theoretical Physics, University of Cambridge. My friend and colleague Eric Schmidt without whom this work might not have come to light. UBM for offering me a stimulating environment to learn and think, particularly Douglas Muirden. And to Marcella *mi amor*, *mi complice y todo*.

9 Appendix 1: Further Reading

The best place to start reading about RBF is the **book** by Prof Martin Buhmann (Buhmann 2003), recently published by the Cambridge University Press. This same author has an excellent **review chapter** in the numerical analysis series *Acta Numerica* (Buhmann 2000) edited by Prof Arieh Iserles, University of Cambridge. Many technical articles, both on theory and applications, are available to download from the **internet**, for example from CiteSeer (www.citeseer.com), JStor (www.jstor.org) and Ingenta (www.ingenta.org). Of particular note is the series of articles in the special issue of *Computers and Mathematics with Applications*, Volume 43, Numbers 3–5 (2002). Edward Kansa's (kansa.irisinternet.net) website contains a large bibliography. Finally, the Cambridge University Press (www.cup.cam.ac.uk) has just published a book on Approximation Theory (and RBF) by Wendland

FOOTNOTES & REFERENCES

1. RBF are an instance of a new class of numerical methods in which the numerical solution of PDEs is obtained using globally-defined basis functions. Another example of such methods is the "spectral method" (see Trefethen 2000).
2. A method for determining the coefficients α_i in an expansion $y(x) = y_0(x) + \sum_{i=1}^{q} \alpha_i y_i(x)$ so as to nullify the values of a differential equation $L[y(x)] = 0$ at prescribed points. The method has been used extensively in the finite element literature in the context of the methods of weighted residuals (e.g. see Zienkiewicz and Taylor 2000).
3. Of course, the degree of smoothness will depend on the particular RBF used. Some RBF, such as Gaussians are infinitely differentiable, while others like thin-plate splines are discontinuous at the origin Buhmann (2003).
4. Micchelli (1986) proved that for a case when the knots are all distinct, the matrix resulting from the above radial basis function interpolation is always non-singular.
5. In the case of multidimensional polynomial interpolation, we can always find a finite set of sites $\Xi \in \mathbb{R}^d$ that causes the interpolation problem to become singular, whenever the dimension is greater than one and the data sites can be varied. This is a standard result in multivariate interpolation theory. As a consequence of this, we need either to impose special requirements on the placement of Ξ or preferably to make the space of polynomials depend on Ξ. The easiest cases for multivariate polynomial interpolation with prescribed geometries of data points are the tensor-product approach (which is useless in most practical cases when the dimension is large because of the exponential increase of the required number of data and basis functions) Buhmann (2003).
6. See also, Madych and Nelson (1988), and Madych (1992) who showed that interpolation with the MQ basis is exponentially convergent.
7. Further convergence proofs in applying the RBFs for scattered data interpolation was given by Wu (1993) and Wu and Schaback (1998).
8. The archeotypical "bean" used in finite element analysis to represent a domain of irregular shape.
9. Multidimensional spline interpolation methods usually require a triangulation of the set Ξ in order to define the space from which we approximate, unless the data sites are in very special positions, gridded or regularly distributed. The reason for this is that it has to be decided where the pieces of the polynomials lie and where they are joined together. Moreover, it has then need to be decided with what smoothness they are joined together at common vertices, edges, and hos this is done. This is not at all trivial in more than one dimension and it is highly relevant in connection with the dimension of the space.
10. The method is now known as "Kansa's (asymmetric) collocation method" and will form the basis of this paper, as it is the clearest and easiest to implement. However, there

are some interesting variations that ought to be considered in a more advanced setting, such as the symmetric Hermite collocation method of Fasshauer (1996).
11. www.ingenta.com, Currently listing 16,962,300 articles from 28,707 publications. Search done on 12 Sep 2004. Period covered by database: March 1995-Sep 2004. Keywords used "radial basis function.
12. The Euclidean norm of the vector $\vec{x}$ is defined as $\|\vec{x}\|_2 = [\sum_{i=1}^{d} x_i^2]^{1/2}$. The distance between two points $\vec{x} = (x_1, x_2, \ldots, x_N)^T$ and $\vec{y} = (y_1, y_2, \ldots, y_N)^T$ in $\mathbb{R}^d$ can be found by taking their difference elementwise and then obtaining the Euclidean norm of the resulting vector, $\|\vec{x} - \vec{y}\|_2 = [\sum_{i=1}^{d} (x_i - y_i)^2]^{1/2}$. See e.g. Faires and Burden, (1993).
13. The Thin Plate Spline $r^2 \log r$ was originally defined for dimension $d = 2$ as the surface minimizing of a certain bending energy.
14. Collocation can be alternatively explained in terms of residuals. We start by defining the residual function

$$R(\vec{x}) = L\,u(\vec{x}) - f(\vec{x}),\ \in \Omega, \tag{10}$$

We now consider a family of tests functions $\tilde{u}$, which can be used to approximate the exact solution u. Our criteria to determine how "good" each of these functions are to solve the differential equation, will be based on $R(\vec{x})$. In particular, for a set of points Ξ in Ω, we will enforce the condition $R(\vec{\xi}_i) = 0$. For details see Zienkiewicz (2000).
15. The constant c is the so-called "shape parameter", whose optimum value is problem-dependent, we follow Hardy and use the value c=4dS (see Kansa 1990). However, there is currently no formal method for setting it. This is one of the big open questions in RBF theory.
16. Because RBF provides a solution for the whole domain, it is straight-forward to calculate the locus of the exercise boundary, $B(t)$ by simply taking the difference between the payoff $g(S)$ and the approximation $V_{American}\ (S, t)$, without the need of using special techniques.

■ Braess D, Finite elements: theory, fast solvers, and applications in solid mechanics. Cambridge Univeristy Press, 1997.
■ Buhmann M, Radial Basis Functions, Cambridge University Press, 2003.
■ Choi S, Marcozzi MD, "A Numerical Approach to American Curreny Option Valuation", Journal of Derivatives, 19–28, 2001.
■ Choy KY, Chan CW, Modelling of river discharges and rainfall using radial basis function networks based on support vector regression. International Journal of Systems Science, January 2003, vol. 34, no. 1, pp. 763–773.
■ Donaldson NN, Gollee H, Hunt KJ, Jarvis JC, Kwende MKN, "A radial basis function model of muscle stimulated with irregular inter-pulse intervals", Medical Engineering and Physics, September 1995, vol. 17, no. 6, pp. 431–441(11)
■ Duchon J, "Sur l'erreur d'interpolation des functions de plusiers variables pas les D^m splines". Rev. Francaise Automat. Informat. Rech. Oper. Anal. Numer. 12 (4): 325–334, 1978.
■ Faires JD, Burden RL, Numerical Methods, PWS, Boston, 1993.
■ Fasshauer GE, Khaliq A ,Voss D, "Using Meshfree Approximation for Multi-Asset American Option Problems", J. Chinese Institute Engineers (Vol.27 No.4), 563–571, 2004.
■ Fasshauer GE, "Solving partial differential equations by collocation with radial basis functions", In Proceedings of Chamonix (Ed. A Le Mechaute et al.), 1–8, Vanderbilt University Press, Nashville, USA, 1996.
■ Fasshauer GE, "Solving differential equations with radial basis functions: multilevel methods and smoothing", Advances in Computational Mathematics, Volume 11, Issue 2-3, 1999, Pages 139–159
■ Ferreira AJM, "Thick Composite Beam Analysis Using a Global Meshless Approximation Based on Radial Basis Functions", Mechanics of Composite Materials and Structures, Jul-Sep vol. 10, no. 3, pp. 271–284(14), 2003.
■ Flowers B H, An introduction to numerical methods in C++. Oxford University Press, 2000.
■ Fornefett M, Rohr K, Stiehl HS, "Radial basis functions with compact support for elastic registration of medical images", Image and Vision Computing, January 2001, vol. 19, no. 1, pp. 87–96(10).
■ Franke R, "Scattered data interpolation: tests of some methods", Math. Comput. 38, 181–199 (1992) .
■ Geske R, Johnson HE, "The American Put Valued Analytically", Journal of Finance, 39: 1511–1524, 1984.
■ Golub GH, Van Loan CF, Matrix Computations. The Johns Hopkins University Press, 1996.
■ Hardy RL, "Multiquadratic equations of topography and other irregular surfaces", J Geophys Res (1971), 76: 1905–15.
■ Hardy RL, "Theory and applications of the mutiquadratic-biharmonic method: 20 years of discovery", Comput. Math. Applic., 19 (8/9), 163–208, (1990).
■ Higham DJ. An introduction to financial option valuation: mathematics, stochastics and computation. Cambridge University Press, 2004.
■ Hon YC, Kansa EJ, "Circumventing the ill-conditioning problem with multiquadric radial basis functions: applications to elliptic partial differential equations", Comput. Math. Applic., 39:123–127, 2000.
■ Hon YC, Mao XZ, "A Radial Basis Function Method for Solving Options Pricing Models", Financial Engineering, 8, (1), 31–49, (1999). International Association of Financial Engineers (IAFE), http://www.iafe.org/
■ Hon YC, Lu MW, Xue WM, Zhu YM, "Multiquadratic method for the numerical solution of a biphasic mixture model", Appl Math Comput 1997, 88:153–75.
■ Hon YC "A quasi radfial basis function method for American Option Pricing", Computers and Mathematics with Applications 43 (2002), 513–524.
■ Huang JM, Subrahmanyam M, and Yu G, "Pricing and Hedging American Options: A Recursive Integration Method", Review of Financial Studies, 9, 277–300. 1996.
■ De Iaco S, Myers DE, Posa D. "Space-time variograms and a functional form for total air pollution measurements", Computational Statistics and Data Analysis, 28 December 2002, vol. 41, no. 2, pp. 311–328(18)
■ Iserles A, A first course in the numerical analysis of differential equations, Cambridge University Press, 1996.
■ Johnson MJ (1997) "An upper bound on the approximation power of principal shift invariant spaces", Constructive Approximation, 13: 155–176.
■ Joshi MS, The concepts and practice of mathematical finance. Cambridge University Press, 2003.
■ Ju N. Pricing an American Option by Approximating its early exercise boundary as a multipiece exponential function, Review of Financial Studies, 11(3), 627–646, 1998.
■ Kansa EJ, "Multiquadratics - a scattered data approximation scheme with applications to computational fluid dynamics. I. Surface approximations and partial derivastive estimated", Comput. Math. Applic. 19 (8/9), 127–145 (1990).
■ Kansa EJ, "Multiquadratics - a scattered data approximation scheme with applications to computational fluid dynamics. II. Solutions to hyperbolic, parabolic, and elliptic partial differential equations", Comput. Math. Applic. 19 (8/9), 147–161 (1990).
■ Madych WR, Nelson SA, "Multivariate Interpolation and Conditionally Positive Definite Functions", Approx. Theory Appl., 4, 77–89, (1988).
■ Madych WR, "Miscellaneous Error Bounds for Multiquadric and Related Interpolators", Comput. Math. Appl., 24, 121–138, (1992).
■ Micchelli CA, "Interpolation of scattered data: distance matrices and conditionally positive definite functions", Constr. Approx.,1, 11–22 (1986).
■ Power H, Barraco V, "A comparison analysis between unsymmetric and symmetric radial basis function collocation methods for the numerical solution of partial differential equations", Comp. Math. Appl. 43 (2002) 551–583.

■ Press WH et al. Numerical recipes in C++ : the art of scientific computing. Cambridge University Press, 2002.
■ Schaback R, Wendland H, "Using compactly supported radial basis functions to solve partial differential equations", in: C.S. Chen, C.A. Brebbia and D.W. Pepper (eds.), Boundary Element Technology XIII, WitPress, Southampton, Boston, 1999, 311–324.
■ Seydel R. Tools for computational finance. Springer-Verlag,2002.
■ Shu C, Ding H, Yeo KS, "Local radial basis function-based differential quadrature method and its application to solve two-dimensional incompressible Navier-Stokes equations. Computer Methods in Applied Mechanics and Engineering, 14 February 2003, vol. 192, no. 7, pp. 941–954(14).
■ Schaback R, "Error Estimates and Condition Numbers for Radial Basis Function Interpolation", Adv. Comput. Math., 3, 251–264, (1995).
■ Trefethen LN, Spectral Methods in MATLAB, Society of Industrial and Applied Mathematics (SIAM), 2000.
■ Van Gestel T, Suykens J, Baestaens D, Lambrechts A, Lanckriet G, Vandaele B, De Moor B, Vandewalle J, IEEE Transactions on Neural Networks, Special Issue on Neural Networks in Financial Engineering, 12(4):809–821, 2001.
■ Wang JG, Liu GR, Lin P, "Numerical analysis of Biot's consolidation process by radial point interpolation method", International Journal of Solids and Structures, March 2002, vol. 39, no. 6, pp. 1557–1573(17)
■ Wilmott P, Howison S, Dewynne J. The Mathematics of Financial Derivatives: A Student Introduction. Cambridge University Press, 1995.
■ Wilmott P. Paul Wilmott on Quantitative Finance, Volume 1, John Wiley and Sons, 2000
■ Wu ZM,"Hermite-Bikhoff interpolation of scattered data by radial basis functions", Approximation Theory and its Application, 8, 1–10, (1992).
■ Wu ZM, Schaback R, "Local Error Estimates for Radial Basis Function Interpolation of Scattered Data", IMA J. Numer. Anal., 13, 13–27, (1993).
■ Wu ZM,"Solving PDE with radial basis functions and the error estimation", Advances in Computational Mathematics, Lecture Notes on Pure and Applied Mathematics, 202, eds. Z. Chen, Y. Li, CA Micchelli, Y Xu, M. Dekker, GuangZhou (1998).
■ Wu ZM, Hon YC, "Convergence error estimate in solving free boundary diffusiuon problems by radial basis functions method", Engineering Analysis with Boundary Elements 27 (2003): 73–79.
■ Zala CA, Barrodale I, "Warping aerial photographs to orthomaps using thin plate splines", Advances in Computational Mathematics, 1999, vol. 11, no. 2/3, pp. 211–227(17)
■ Zakrzhevskii AE, "Optimal Slewing of a Flexible Spacecraft", International Applied Mechanics, October 2003, vol. 39, no. 10, pp. 1208–1214(7)
■ Zerroukat M, Power H, Chen CS. "A numerical method for heat transfer problems using collocation and radial basis functions", Int J Numer Meth Engng 1998, 42: 1263–78.
■ Zienkiewicz OC, Taylor RL. The finite element method. Oxford, Butterworth-Heinemann, 2000.