Option Pricing using Radial Basis Functions

U. Pettersson,[1] E. Larsson,[2] G. Marcusson,[3] and J. Persson[4]

Abstract: *In this paper, we have implemented a radial basis function (RBF) based method for solving the Black–Scholes partial differential equation. The application we have chosen is the valuation of European call options based on several underlying assets. We have shown that by appropriate choices of the RBF shape parameter and the node point placement, the accuracy of the results can be improved by at least an order of magnitude. We have also looked at how and where to implement boundary conditions in more than one dimension.*

Keywords: Radial basis function, Black–Scholes equation, multi-dimensional, boundary conditions.

1 Introduction

The financial markets are becoming more and more complex, with trading not only of stocks, but also of numerous types of financial derivatives. The market requires updated information about the values of these derivatives every second of the day. This leads to a huge demand for fast and accurate computer simulations.

In this study, we take a special interest in European call options based on several underlying assets. The values of such options can be determined by solving the Black–Scholes equation. The number of assets then corresponds to the number of dimensions in the partial differential equation. "The curse of dimensionality" makes this task increasingly difficult in higher dimensions and it is necessary to find fast methods with very low memory requirements.

RBF approximation is a promising candidate method. With infinitely smooth RBFs the method can be spectrally accurate, meaning that the required number of node points for a certain desired accuracy is relatively small. The meshfree nature of the method makes it easy to use in higher dimensions and also allows for adaptive node placement.

[1]Uppsala University, Dept. of Information Technology, Box 337, SE-751 05 Uppsala, Sweden (ulrikap@it.uu.se).

[2]Uppsala University, Dept. of Information Technology (bette@it.uu.se).

[3]Försäkringsmatematik, Box 5148, SE-102 43 Stockholm, Sweden (Gunnar.Marcusson@forsakringsmatematik.se).

[4]Uppsala University, Dept. of Information Technology (jonasp@it.uu.se).

The RBF approach has been explored previously by other authors [4, 3, 7, 1]. Our specific angle in this context is an investigation of what can be achieved by optimizing the method parameters. Furthermore, we look at how boundary conditions should be implemented and where they are needed.

2 The multi-dimensional model problem

2.1 The Black–Scholes equation

The Black–Scholes equation is a time-dependent linear partial differential equation. Normally, it is posed as a final value problem. For ease of notation and computation, we here use a version that has been transformed to an initial value problem and also scaled to have dimensionless variables [6]. The form is

$$\begin{cases} \dfrac{\partial}{\partial \hat{t}} P(\hat{t},\underline{x}) &= \mathcal{L}P(\hat{t},\underline{x}), \quad \hat{t} \in \mathbb{R}_+, \quad \underline{x} \in \mathbb{R}_+^d, \\ P(0,\underline{x}) &= \Phi(\underline{x}), \quad \underline{x} \in \mathbb{R}_+^d, \end{cases} \tag{1}$$

where

$$\mathcal{L}P = 2\bar{r}\sum_{i=1}^{d} x_i \frac{\partial P}{\partial x_i} + \sum_{i,j=1}^{d} [\bar{\sigma}\bar{\sigma}^*]_{ij} x_i x_j \frac{\partial^2 P}{\partial x_i \partial x_j} - 2\bar{r}P. \tag{2}$$

The coefficient $\bar{r}$ is the scaled short interest rate, and $\bar{\sigma}$ is the scaled volatility. An example of a contract function for a European basket option is given by

$$\Phi(\underline{x}) = \max\left(0, \frac{1}{d}\sum_{i=1}^{d} x_i - \bar{K}\right), \tag{3}$$

where the scaled strike price in our case is $\bar{K} = 1$.

2.2 Boundary conditions

For computational purposes, we need to restrict the problem to a finite domain. Because we are using a meshfree method, we have the opportunity to choose the artificial far-field boundary as we like. With the contract function above, it makes sense to use a boundary surface of the type $\sum_{i=1}^{d} x_i = C$, where the constant C is chosen to bring the surface far enough from the origin. On this surface, we can use the asymptotic solution

$$P(\hat{t},\underline{x}) \to \frac{1}{d}\sum_{i=1}^{d} x_i - \bar{K}e^{-2\bar{r}\hat{t}}, \quad \|\underline{x}\| \to \infty. \tag{4}$$

The near-field boundary can be seen as the single point $\underline{x} = \underline{0}$, and there we enforce

$$P(\hat{t},\underline{0}) = 0. \tag{5}$$

We have not specified any conditions for the parts of the boundary surface that are given by $x_j \equiv 0$ for some j. It has actually been shown that, mathematically, this is not needed [5].

2.3 Measuring the error

When measuring the error in the RBF approximation it is important to remember the real-life background of the problem we are solving. In option trading the region of interest is where the mean stock price is close to the strike price. When the mean stock price is much lower or higher than the strike price, the probability of the stock price reaching the strike price is very low. Hence in these situations the option is worth either nothing, or the difference between the mean stock price and the strike price. We define the region of interest to be all $\underline{x}$ for which $\frac{1}{d}\sum_{i=1}^{d} x_i \in \left[\frac{\bar{K}}{3}, \frac{5\bar{K}}{3}\right]$, and propose a financial error norm given by the maximum error over this region at the final time $\hat{t} = T$, i.e., the exercise time of the option.

3 RBF approximation

We approximate the solution of (1) with a time-dependent linear combination of RBFs given by

$$u(\hat{t}, \underline{x}) = \sum_{k=1}^{N} \lambda_k(\hat{t})\phi(\varepsilon||\underline{x} - \underline{x}_k||) = \sum_{k=1}^{N} \lambda_k(\hat{t})\phi_k(\underline{x}). \tag{6}$$

The equation then becomes

$$\sum_{k=1}^{N} \lambda_k'(\hat{t})\phi_k(\underline{x}) = \sum_{k=1}^{N} \lambda_k(\hat{t})\mathcal{L}\phi_k(\underline{x}), \tag{7}$$

When we solve the PDE, we work with function values at the node points. That is, we solve for the vector $\underline{u}(\hat{t}) = (u(\hat{t}, \underline{x}_1), \ldots, u(\hat{t}, \underline{x}_N))^T$ given by

$$\underline{u}(\hat{t}) = A\underline{\lambda}(\hat{t}), \tag{8}$$

where $A_{ij} = \phi_j(\underline{x}_i)$ and $\underline{\lambda}(\hat{t}) = (\lambda_1(\hat{t}), \ldots, \lambda_N(\hat{t}))^T$.

4 Time-stepping

For the time evolution of the problem, we use the BDF–2 method [2] with a constant time step k. Let $\hat{t}^n = kn$ and let $\underline{u}^n \approx \underline{u}(\hat{t}^n)$. Then the method can be written

$$\frac{\alpha_0\underline{u}^n + \alpha_1\underline{u}^{n-1} + \alpha_2\underline{u}^{n-2}}{k} = \underline{\mathcal{F}}^n,$$

where α_j are constant coefficients and $\underline{\mathcal{F}}^n = (f_1^n, \ldots, f_N^n)^T$ approximates the differential operator in space at the node points. For points $\underline{x}_i$ where we enforce the Black–Scholes equation, (7) and (8) lead to

$$f_i^n = \underline{b}_i\underline{\lambda}^n = \underline{b}_i A^{-1}\underline{u}^n, \tag{9}$$

where $\underline{b}_i = (\mathcal{L}\phi_1(\underline{x}_i), \ldots, \mathcal{L}\phi_N(\underline{x}_i))$.

It is important to implement boundary conditions in such a way that they are incorporated into the time scheme and not adjusted afterwards. For Dirichlet conditions $u(\hat{t},\underline{x}_i) = g(\hat{t})$, this can easily be done by defining

$$f_i^n = \frac{\alpha_0 g(\hat{t}^n) + \alpha_1 g(\hat{t}^{n-1}) + \alpha_2 g(\hat{t}^{n-2})}{k}. \tag{10}$$

Assuming that the initial boundary values are correct, this yields the desired result. Finally, combining (9) and (10), we get the overall time-stepping scheme

$$(I - \frac{k}{\alpha_0} BA^{-1})\underline{u}^n = \frac{k}{\alpha_0}\underline{g}^n - \frac{k\alpha_1}{\alpha_0}\underline{u}^{n-1} - \frac{k\alpha_2}{\alpha_0}\underline{u}^{n-2},$$

where

$$B_{i,1:N} = \begin{cases} 0, & \underline{x}_i \text{ Dirichlet point,} \\ \underline{b}_i, & \underline{x}_i \text{ Black–Scholes point,} \end{cases} \qquad g_i^n = \begin{cases} f_i^n, & \underline{x}_i \text{ Dirichlet point,} \\ 0, & \underline{x}_i \text{ Black–Scholes point.} \end{cases}$$

Remark 1: Sometimes in the literature, $d-1$ dimensional problems are solved at the boundaries where the PDE collapses. These are the boundaries where we do not use any special conditions. Our reasons are both the mathematical ones given in [5], and the fact that if we time-step these points along with the rest, we automatically use the lower dimensional differential equation where it is appropriate. This should provide the correct time evolution everywhere.

Remark 2: For second order finite difference methods, linear extrapolation conditions are often used as numerical boundary conditions, but for RBFs this simply does not make any sense.

5 Numerical experiments

We have used multiquadric RBFs in all the experiments. The far-field boundary surface was given by all $\underline{x}$ for which $\frac{1}{d}\sum_{i=1}^{d} x_i = 4\bar{K}$. The parameters were set to $\bar{r} = 5/9$, $\bar{\sigma} = 1$ in one dimension, and $\bar{\sigma} = 1$ on the diagonal and $\bar{\sigma} = 1/6$ off the diagonal in two dimensions. The time-step was chosen small enough to not affect the result and the final time was $T = 0.045$, corresponding to 1 year.

5.1 Node distribution

Because our interest is in the financial norm of the error, we can gain accuracy by using a denser node distribution in the region of interest and a sparser distribution outside this region. In other words we gain accuracy in the region where we want an accurate solution by lowering the accuracy in the region where the solution is of little interest (see Figure 1). This strategy improves the result by one order of magnitude for both the one-dimensional and the two-dimensional problem, without increasing the computational cost.

5.2 Choosing the shape parameter value

The optimal shape parameter value for an RBF method has a non-coincidental connection to the problem at hand. It is therefore likely that a formula for the best choice can be

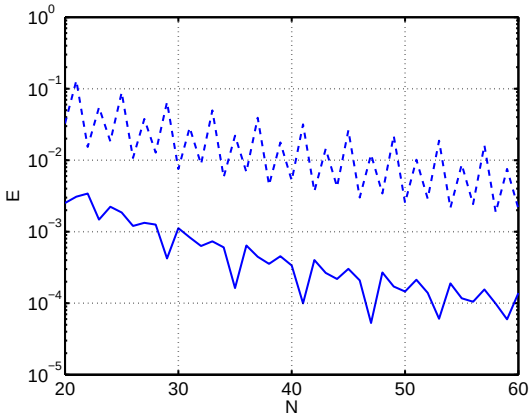

Figure 1: Errors for uniform (dashed) and non-uniform (solid) node distributions in 1D.

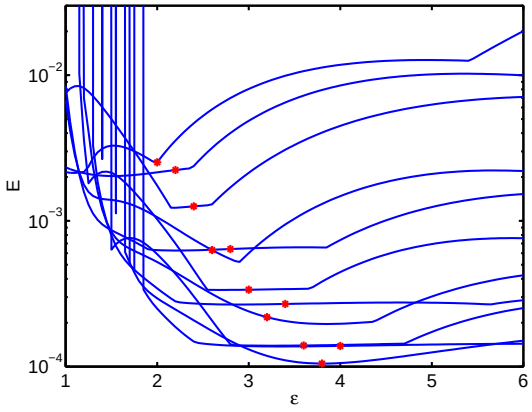

Figure 2: Error as function of ε for different N. The stars show $\varepsilon = 1 + N/20$.

found. However, in practise, the global optimum of the shape parameter value is often hidden by ill-conditioning. This means that we must at the present time settle for finding a choice of the shape parameter that gives good results for any number of points, although it might not be the optimal choice theoretically. We have found that in one dimension the formula $\varepsilon = 1 + N/20$, gives a rather good result for the values of N that we have tested (see Figure 2). For two-dimensional problems it seems that the best results are obtained for the smallest possible ε-value for which the problem is not too ill-conditioned.

5.3 Accuracy in space

One of the main advantages of the RBF method is that it can give spectral accuracy. This has been shown theoretically for some types of problems and numerically for a wider range of problems. We have tested the accuracy in space for the one-dimensional Black–Scholes problem, but have unfortunately found that the accuracy does not seem to be spectral but rather algebraic (see Figure 3). The error is approximately $E(N) \approx 59N^{-3}$. There may be compound reasons for this, but one likely source is the discontinuous derivative in the initial data. An analysis of the results for a European put option in [7] shows a similar convergence rate.

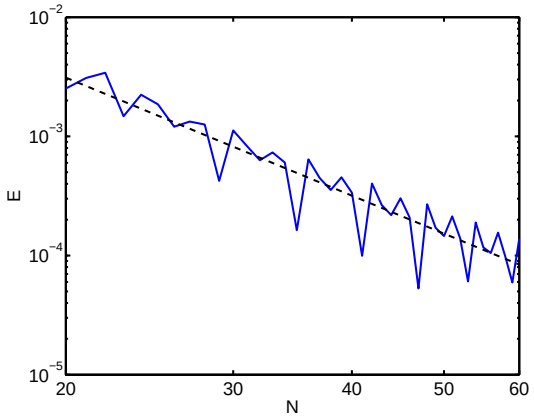

Figure 3: The error as a function of N (logarithmic axes).

6 Work in progress

We have not shown any high dimensional results here. We have some two-dimensional results and will produce more before the conference. Higher dimensions will be looked into later. A thorough comparison of the computational costs and memory requirements of the RBF method and the finite difference method described in [6] is under way, but not yet finished. We are also looking at ways to recover the spectral accuracy, including multiscale approximation and least squares.

References

[1] Fasshauer G. E., Khaliq A. Q. M., Voss D. A., Using meshfree approximation for multi-asset American option problems, J. Chinese Institute Engineers 27 (2004), pp. 563–571.

[2] Hairer E., Nørsett S. P., Wanner G., Solving ordinary differential equations I, 2nd ed. Springer-Verlag, Berlin 1993.

[3] Hon Y. C., A quasi-radial basis functions method for American options pricing, Comput. Math. Appl., 43 (2002), pp 513–524.

[4] Hon Y. C., Mao X. Z., A radial basis function method for solving options pricing model, J. Financial Engineering, 8 (1999), pp.1–24.

[5] Janson S., Tysk J., Feynman–Kac formulas for Black–Scholes type operators, preprint (2004).

[6] Persson J., von Sydow L., Pricing European multi-asset options using a space-time adaptive FD-method, Comput. Vis. Sci., submitted (2003).

[7] Wu Z., Hon Y. C., Convergence error estimate in solving free boundary diffusion problem by radial basis functions method, Engrg. Anal. Bound. Elem., 27 (2003), pp. 73–79.