Applied Mathematical Modelling 38 (2014) 135–144

Contents lists available at SciVerse ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

# On the selection of a good value of shape parameter in solving time-dependent partial differential equations using RBF approximation method

Marjan Uddin *

*Department of Basic Sciences, University of Engineering and Technology, Peshawar, Pakistan*

ARTICLE INFO

*Article history:*
Received 29 March 2012
Received in revised form 13 March 2013
Accepted 31 May 2013
Available online 26 June 2013

*Keywords:*
Kernel functions
Partial differential equations
Cross validation
Shape parameter

ABSTRACT

Radial basis function method is an effective tool for solving differential equations in engineering and sciences. Many radial basis functions contain a shape parameter c which is directly connected to the accuracy of the method. Rippa [1] proposed an algorithm for selecting good value of shape parameter c in RBF-interpolation. Based on this idea, we extended the proposed algorithm for selecting a good value of shape parameter c in solving time-dependent partial differential equations.



## 1. Introduction

Radial basis function method is an efficient technique for solving multidimensional problems in engineering and sciences. Kansa was the first to use RBF for solving partial differential equations [2,3]. The RBF approximation technique is truly meshless and is based on collocation in a set of scattered nodes. In the last two decades a number of researchers have developed various meshless methods using RBF and have recently been used for solving partial differential equations in engineering and sciences. Particular examples include convection–diffusion problems [4–8], elliptic problems [9–13], Poisson problems [14–17], potential problems [18,19], financial mathematics [20–23]. Many other successful application based on radial basis function method can be found in mathematics, engineering and physics journals. For examples application of RBF approximation method to Burgers equation [24–29], Korteweg–de Vries equation [30–33], RLW equation [34,35], Kuramoto–Sivashinsky equation [36,37], Coupled Korteweg–de Vries equations [38–41], etc.

Most of the RBFs used to approximate the solution of partial differential equation contain a shape parameter c which must be specified by the user. This random selection of c is a disadvantage. A number of papers have been written on choosing optimal value of RBFs shape parameter. For example Hardy [42] suggested the use of shape parameter $c = 0.815d$, where $d = 1/N\sum_{i=1}^{N} d_i$ and $d_i$ is the distance from the data point $x_i$ to its nearest neighbor. Franke [43] suggested to use $c = 1.25D/\sqrt{N}$ where D is the diameter of the minimal circle enclosing all data points. Rippa [1] proposed an algorithm for choosing an optimal value of RBFs shape parameter. G. E Fasshauer [44] suggested an algorithm for choosing optimal value of RBF shape parameter for iterated moving least squares (AMLS) approximation and for RBF pseudo-spectral (PS) methods for the solution of partial differential equations. Recently Michael Scheuerer [45] proposed another procedure

* Tel.: +92 03459556218.
*E-mail address:* marjankhan1@hotmail.com

0307-904X/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.apm.2013.05.060

for selecting good value of c in RBF-interpolation. More recently Victor Bayona, et al. [46] proposed an algorithm for selecting an optimal value of multiquadric shape parameter c in RBF-FD method.

In this paper, we extended Rippa's [1] algorithm for selecting good values of multiquadric shape parameter c in solving time-dependent partial differential equations using radial basis functions.

## 2. RBF approximation method for PDEs

Consider a spatial domain $\Omega$ and an operator $\mathcal{L}$ acting on a smooth function on $\Omega$. Suppose that the operator $\mathcal{L}$ always acts with respect to the spatial variable even when time variable $t$ is present. On a time domain $[0, T]$, we look for a scalar function $u : \Omega \times [0, T] \longrightarrow \mathcal{R}$, satisfying the time dependent partial differential equation

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} + \mathcal{L}u(\mathbf{x}, t) = f(\mathbf{x}, t), \quad \mathbf{x} \in \Omega, \tag{1}$$

along with the boundary and initial conditions

$$\mathcal{B}u(\mathbf{x}, t) = g(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega, \tag{2}$$

$$u(\mathbf{x}, t) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega. \tag{3}$$

The RBF approximation to the solution $u(\mathbf{x}, t)$ of Eq. (1) is given as

$$u^n(\mathbf{x}) = \sum_{j=1}^{N} \lambda_j^n \psi(\|\mathbf{x} - \mathbf{x}_j\|), \quad \mathbf{x} \in \Omega, \tag{4}$$

where $u(\mathbf{x}, t_n)$ is denoted by $u^n(\mathbf{x})$. The grid points in the time interval $[0, T]$ are labeled as $t_n = n\delta t, \delta t = 1/M, n = 0, 1, 2, \ldots, T \times M, \delta t$ is the time step size, $\psi(\|x - x_j\|)$ is a radial basis function, and $\|\cdot\|$ is Euclidian norm. Eq. (4) can be written in the matrix–vector form as

$$\mathbf{u}^n = \mathbf{A}\lambda^{\mathbf{n}}. \tag{5}$$

The entries of the matrix $\mathbf{A}$ are $A_{ij} = \psi(\|x_i - x_j\|)$, and $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_N]$ is the expansion coefficient vector. Applying $\theta$-weighted scheme to (1) we get

$$\frac{1}{\delta t}[u^{n+1}(\mathbf{x}) - u^n(\mathbf{x})] + \theta\mathcal{L}u^{n+1}(\mathbf{x}) + (1 - \theta)\mathcal{L}u^n(\mathbf{x}) = f(\mathbf{x}, t^{n+1}). \tag{6}$$

Using Eq. (4) in Eq. (1) we can write

$$\frac{1}{\delta t}\left[\sum_{j=1}^{N}\lambda_j^{n+1}\psi(\|\mathbf{x} - \mathbf{x}_j\|)(\mathbf{x}) - \sum_{j=1}^{N}\lambda_j^{n}\psi(\|\mathbf{x} - \mathbf{x}_j\|)(\mathbf{x})\right] + \theta\left[\sum_{j=1}^{N}\lambda_j^{n+1}\mathcal{L}\psi(\|\mathbf{x} - \mathbf{x}_j\|)(\mathbf{x})\right] + (1 - \theta)\left[\sum_{j=1}^{N}\lambda_j^{n}\mathcal{L}\psi(\|\mathbf{x} - \mathbf{x}_j\|)(\mathbf{x})\right]$$
$$= f(\mathbf{x}, t^{n+1}) \tag{7}$$

and from Eq. (2) we have

$$\sum_{j=1}^{N}\lambda_j^{n+1}\mathcal{B}\psi(\|\mathbf{x} - \mathbf{x}_j\|)(\mathbf{x}) = g(\mathbf{x}, t^{n+1}). \tag{8}$$

The above system of equations can be written in matrix–vector form as

$$\mathbf{G}\lambda^{\mathbf{n}+1} = \mathbf{b}^{n+1}, \tag{9}$$

where

$$\mathbf{G} = \begin{bmatrix} \psi(\|\mathbf{x} - \mathbf{x}_j\|) + \delta t\theta\mathcal{L}\psi(\|\mathbf{x} - \mathbf{x}_j\|), j = 1, \ldots, N, \mathbf{x} \in \mathcal{I} \\ \mathcal{B}\psi(\|\mathbf{x} - \mathbf{x}_j\|), j = 1, \ldots, N, \mathbf{x} \in \partial\Omega \end{bmatrix},$$

$$\mathbf{b}^{n+1} = \begin{bmatrix} u^n(\mathbf{x}) - \delta t(1 - \theta)\mathcal{L}u^n(\mathbf{x}) + f(\mathbf{x}, t^{n+1}), \mathbf{x} \in \mathcal{I} \\ g(\mathbf{x}, t^{n+1}), \mathbf{x} \in \partial\Omega \end{bmatrix}.$$

It should be noted that $\mathbf{G}$ is $N \times N$ matrix, $\mathbf{b}^{n+1}, \lambda^{n+1}$ are $N \times 1$ vectors respectively. If the operator $\mathcal{L}$ is not linear we can linearize the nonlinear terms involved in $\mathbf{G}$. The values of $\lambda^n$ at any time level $n$ can be obtained from Eq. (9), and then RBF approximate solution from Eq. (5). We are using $\theta = 1/2$ in all our computations.