



IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY



VOLUME : 5 ISSUE : 1 Print / Issue Publication Date: 13-Jul-2020



ISSN : 2455-2143



DOI : 10.33564/IJEAST.2020.v05i01.008

Indexed In



WWW.IJEAST.COM

editor@ijeast.com



APPROXIMATING THE SOLUTIONS OF HIGHER ORDER LINEAR TIME-DEPENDENT PARTIAL DIFFERENTIAL EQUATIONS USING RADIAL BASIS FUNCTION METHOD OF LINES (RBF-MOLS)

Tersoo Luga

Department of Mathematics/Statistics/Computer Science,
University of Agriculture, Makurdi, Nigeria

Abstract— In this paper, some globally supported radial basis functions (RBFs): multiquadrics (MQ), inverse quadratics (IQ) and generalized inverse multiquadrics (GIMQ) were combined with the fourth order Runge-Kutta method in developing radial basis function method of lines. The RBFs were used to discretize the space variable, while the fourth order Runge-Kutta method was used to integrate the system of ordinary differential equations (ODEs) emanating from the space discretization. The RBF-MOLS were implemented in MATLAB and applied to approximate the solutions of a third order, fourth order and fifth order linear time-dependent partial differential equations. The numerical results showed good approximations when compared with the exact solutions, however, the MQ-RBF-MOLS produced the best approximations.

Keywords— Radial Basis Functions, Method of Lines, Positive Definite RBFs, Globally supported RBFs and Radial Basis Functions Method of Lines, Multiquadrics, Inverse Quadratics and Generalized Inverse Multiquadrics.

I. INTRODUCTION

Time-dependent partial differential equations (PDEs) are used in modelling different evolutionary problems in engineering and mathematical sciences. Most time-dependent PDEs of interest contain higher order derivatives. Many analytical and semi-analytical methods have been developed for solving higher order time-dependent PDEs, however, it is important to develop accurate numerical methods which provide an alternative means of obtaining approximate solutions in situations where both the analytical and semi-analytical methods may fail. Over the years, numerical methods such as finite difference method (FDM), finite element method (FEM), finite volume method (FVM) and some pseudospectral/spectral

methods have been applied to approximate some higher order time-dependent PDEs.

Radial basis function (RBF) methods are meshless and spectral methods that have become viable numerical methods for solving PDEs [1, 2, 3, 4]. Kansa [5, 6] was the first to apply RBF collocation method using multiquadric RBF method to approximate the elliptic, hyperbolic and parabolic PDEs. His pioneer work paved way for a research boom in different scientific computing communities such as computer graphics, data processing [7], economics [8] etc. besides numerical solutions of PDEs. RBFs can either be independently applied to approximate problems or combined with other numerical methods to form accurate hybrid methods.

Tatari and Dehghan [9] explained that if time and space variables are discretized using RBFs, the resulting coefficient matrix may be ill-conditioned, this may affect the accuracy of the results. They suggested that time-dependent PDEs can be discretized in space using RBF methods which will reduce the PDE to a system of ordinary differential equations (ODEs), while the system of ODEs can be solved using an ODE solver. The process of solving a PDE by first discretizing the space variable(s) and integrating the resulting system of ODEs is referred to as the method of lines (MOLS) [10]. If RBF methods are used for space discretization, the MOLS is referred to as RBF-MOLS [2, 4].

Sarra and Kansa [2] formulated an RBF-MOLS for solving some time-dependent PDEs in one and two space dimensions. They used the multiquadrics (MQ) RBF method for space discretization, while the system of ODEs was integrated using the fourth order Runge-Kutta method. Luga *et al.* [11] also formulated some RBF-MOLS using some globally supported and positive definite RBFs such as the inverse multiquadrics (IMQ), inverse quadratics (IQ) and generalized inverse multiquadrics (GIMQ) RBF methods. They applied the RBF-MOLS to solve the one-dimensional cubic Schrodinger PDE and compared their results with the work of [2]. Their proposed methods compared favourably. Similarly, [12] obtained some



test problems on second order time-dependent PDEs from [2] and applied the IMQ-RBF-MOLs, IQ-RBF-MOLs and GIMQ-RBF-MOLs to approximate them, they observed that their results were in good agreement.

Our motivation comes from the works of [11, 12]. They proposed some RBF-MOLs using IMQ, IQ and GIMQ and applied them to solve second order time-dependent PDEs in one-dimension. In this paper, we wish to extend their methods to solve third order, fourth order and fifth order time-dependent PDEs. We shall also extend the RBF-MOLs of [2] to solve higher order linear time-dependent PDEs. The aim is to compare the performance of the IQ-RBF-MOLs and GIMQ-RBF-MOLs with MQ-RBF-MOLs using higher order time-dependent PDEs in one space dimension, since the IQ and GIMQ RBFs are rarely applied to solve or formulate numerical methods.

II. METHODS

In this section, the differentiation matrices of the multiquadric (MQ), inverse quadratic (IQ) and generalized multiquadric (GIMQ) radial basis functions (RBFs) used for space discretization of time-dependent PDEs are presented. Since the differentiation matrix of an RBF depends on an evaluation matrix which in turn depends on an interpolation matrix, we shall first discuss them. The algorithm for the space discretization as well as the fourth order explicit Runge-Kutta method used for integrating the ODEs that emanate from the space discretization is also stated here.

Radial Basis Function (RBF) Interpolation

Since the space discretization of one-dimensional time-dependent PDEs, we shall first make an assumption that the unknown function $u(x)$ can be approximated using the interpolant $s(x)$, i.e.

$$s(x_i) = u(x_i) \quad \text{for } 1 \leq i \leq N \quad (2.1)$$

where $s: \mathbb{R}^d \rightarrow \mathbb{R}$, $\{x_1, x_2, x_3, \dots, x_N\} \subset \mathbb{R}^d, d \geq 1$ are scattered set of data while $u: \mathbb{R}^d \rightarrow \mathbb{R}$ are function values obtained from scattered data which represents the unknown function. The interpolant is defined by

$$s(x_i) = \sum_{j=1}^N \lambda_j (\|x_i - x_j\|) + p(x), p \in \mathcal{P}_m^d \quad (2.2)$$

where $\|\cdot\|$ and $\mathcal{P}_m^d$ denote the Euclidean norm on $\mathbb{R}^d$ and a linear space containing all real valued polynomials in d variables of degree at most $m-1$.

The IQ and GIMQ RBFs are positive definite so no polynomial term is required to make their interpolation matrices invertible. Although the MQ RBFs is conditionally positive definite and requires a polynomial term of degree one to be appended to its interpolation matrix to make it invertible, [13] and [3] observed that conditionally positive definite RBFs of order 1 such as the MQ can be used without appending the constant term, yet its interpolation matrix can be non-singular. To this end, we use the interpolant of the form

$$s(x_i) = \sum_{j=1}^N \lambda_j (\|x_i - x_j\|) \quad (2.3)$$

for each $i = 1, 2, \dots, N, j = 1, 2, \dots, N$.

Interpolation Matrix of Positive Definite RBF Methods

The interpolation matrix of a positive definite RBF method can be formed by substituting equation (2.3) in equation (2.1), expanding and writing it in matrix form as shown below

$$\sum_{j=1}^N \lambda_j \phi(\|x_i - x_j\|) = s(x_i) \quad (2.4)$$

for each $i = 1, 2, \dots, N, j = 1, 2, \dots, N$. In matrix form, equation (2.4) can be written as

$$\begin{bmatrix} \phi(\|x_1 - x_1\|) & \phi(\|x_1 - x_2\|) & \dots & \phi(\|x_1 - x_N\|) \\ \phi(\|x_2 - x_1\|) & \phi(\|x_2 - x_2\|) & \dots & \phi(\|x_2 - x_N\|) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\|x_N - x_1\|) & \phi(\|x_N - x_2\|) & \dots & \phi(\|x_N - x_N\|) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \quad (2.5)$$

or

$$[A] \lambda_i = u_i \quad (2.6)$$

The entries of the matrix $[A]$ are called the basis functions of an RBF method, they are generated from a function called the basic function. The MQ, IQ and GIMQ come from a family of RBFs called generalized multiquadric (GMQ) RBFs. The basic function of the GMQ RBF is given by

$$\phi(r) = (1 + \varepsilon^2 r^2)^\beta \quad (2.7)$$

where $r = \|x_i - x_j\|$, for each $i = 1, 2, \dots, N, j = 1, 2, \dots, N$.

2.1.1 Interpolation Matrices of MQ, IQ and GIMQ RBFs

The basic functions of the MQ, IQ and GIMQ RBFs are defined by substituting $\beta = \frac{1}{2}, -1$ and -2 respectively, i.e.

$$\phi(r) = \sqrt{1 + (\varepsilon \|x_i - x_j\|)^2} \quad (2.8)$$

$$\phi(r) = \frac{1}{1 + (\varepsilon \|x_i - x_j\|)^2} \quad (2.9)$$

$$\phi(r) = \frac{1}{(1 + (\varepsilon \|x_i - x_j\|)^2)^2} \quad (2.10)$$

Substituting (2.8), (2.9) and (2.10) in (2.5) provides the following interpolation matrices for MQ, IQ and GIMQ RBFs

$$\begin{bmatrix} \sqrt{1 + (\varepsilon \|x_1 - x_1\|)^2} & \sqrt{1 + (\varepsilon \|x_1 - x_2\|)^2} & \dots & \sqrt{1 + (\varepsilon \|x_1 - x_N\|)^2} \\ \sqrt{1 + (\varepsilon \|x_2 - x_1\|)^2} & \sqrt{1 + (\varepsilon \|x_2 - x_2\|)^2} & \dots & \sqrt{1 + (\varepsilon \|x_2 - x_N\|)^2} \\ \vdots & \vdots & \ddots & \vdots \\ \sqrt{1 + (\varepsilon \|x_N - x_1\|)^2} & \sqrt{1 + (\varepsilon \|x_N - x_2\|)^2} & \dots & \sqrt{1 + (\varepsilon \|x_N - x_N\|)^2} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \quad (2.11)$$



$$\begin{bmatrix} \frac{1}{1 + (\varepsilon \|x_1 - x_1\|)^2} & \frac{1}{1 + (\varepsilon \|x_1 - x_2\|)^2} & \dots & \frac{1}{1 + (\varepsilon \|x_1 - x_N\|)^2} \\ \frac{1}{1 + (\varepsilon \|x_2 - x_1\|)^2} & \frac{1}{1 + (\varepsilon \|x_2 - x_2\|)^2} & \dots & \frac{1}{1 + (\varepsilon \|x_2 - x_N\|)^2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{1 + (\varepsilon \|x_N - x_1\|)^2} & \frac{1}{1 + (\varepsilon \|x_N - x_2\|)^2} & \dots & \frac{1}{1 + (\varepsilon \|x_N - x_N\|)^2} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \quad (2.12)$$

$$\begin{bmatrix} \frac{1}{(1 + (\varepsilon \|x_1 - x_1\|)^2)^2} & \frac{1}{(1 + (\varepsilon \|x_1 - x_2\|)^2)^2} & \dots & \frac{1}{(1 + (\varepsilon \|x_1 - x_N\|)^2)^2} \\ \frac{1}{(1 + (\varepsilon \|x_2 - x_1\|)^2)^2} & \frac{1}{(1 + (\varepsilon \|x_2 - x_2\|)^2)^2} & \dots & \frac{1}{(1 + (\varepsilon \|x_2 - x_N\|)^2)^2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{(1 + (\varepsilon \|x_N - x_1\|)^2)^2} & \frac{1}{(1 + (\varepsilon \|x_N - x_2\|)^2)^2} & \dots & \frac{1}{(1 + (\varepsilon \|x_N - x_N\|)^2)^2} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \quad (2.13)$$

Evaluation Matrices of Positive Definite RBF methods

To get the evaluation matrix of an RBF, first λ_j , $j = 1, 2, \dots, N$ is obtained from the interpolation matrix (2.5) and the interpolant (2.3) is evaluated at M point x_i , using an $M \times N$ evaluation matrix H which has the entries

$$h_{ij} = \phi(\|x_i - x_j\|) \quad (2.14)$$

for each $i = 1, 2, \dots, M, j = 1, 2, \dots, N$. According to Fasshauer (2007), symmetric matrices are invertible, thus we shall take the number of data points to coincide with the number of centres so that we shall have symmetric evaluation matrices.

Evaluation Matrices of MQ, IQ and GIMQ RBFs

Using equation (2.11), (2.12) and (2.13), the evaluation matrices of the various RBFs namely MQ, IQ and GIMQ can be obtained by first finding λ_j , $j = 1, 2, \dots, N$ in each case and evaluating at x_i points for $i = 1, 2, \dots, N$ using equation (2.3) so as have symmetric evaluation matrices.

Differentiation Matrix of RBF Methods

Differentiation matrices are used to discretize the derivatives of a PDE of interest. The differentiation matrices are obtained by first differentiating the interpolant (2.3) the required number of times

$$\frac{\partial}{\partial x_i} s(x_i) = \sum_{j=1}^N \lambda_j \frac{\partial}{\partial x_i} \phi(\|x_i - x_j\|) \quad (2.15)$$

$$\frac{\partial^2}{\partial x_i^2} s(x_i) = \sum_{j=1}^N \lambda_j \frac{\partial^2}{\partial x_i^2} \phi(\|x_i - x_j\|) \quad (2.16)$$

$$\vdots \quad \vdots$$

$$\frac{\partial^k}{\partial x_i^k} s(x_i) = \sum_{j=1}^N \lambda_j \frac{\partial^k}{\partial x_i^k} \phi(\|x_i - x_j\|) \quad (2.17)$$

Evaluating equation (2.17) at the centres $\{x_j\}_{j=1}^N$ leads to

$$\frac{\partial^k}{\partial x_i^k} s(x_i) = \frac{\partial^k}{\partial x_i^k} H \lambda_j \quad (2.18)$$

where the derivatives of the evaluation matrix is an $N \times N$ matrix denoted by $\frac{\partial^k}{\partial x_i^k} H$ with the entries

$$h_{ij} = \frac{\partial^k}{\partial x_i^k} \phi(\|x_i - x_j\|) \quad (2.19)$$

for each $i = 1, 2, \dots, M, j = 1, 2, \dots, N$.

In this paper, we denote the first five partial derivatives of the evaluation matrix as $H_x, H_{xx}, H_{xxx}, H_{xxxx}$ and H_{xxxxx} respectively.

Making λ_j the subject of the formula from equation (2.6) gives $\lambda_j = [A]^{-1} u$ (2.20)

Substituting equation (2.20) into (2.18) gives

$$\frac{\partial^k}{\partial x_i^k} s(x_i) = \frac{\partial^k}{\partial x_i^k} H [A]^{-1} u \quad (2.21)$$

or

$$\frac{\partial^k}{\partial x_i^k} u(x_i) = \frac{\partial^k}{\partial x_i^k} H [A]^{-1} u \quad (2.22)$$

or

$$Du = \frac{\partial^k}{\partial x_i^k} H [A]^{-1} u \quad (2.23)$$

This implies that

$$D = \frac{\partial^k}{\partial x_i^k} H [A]^{-1} \quad (2.24)$$

Equation (2.24) represents the differentiation matrix of RBF methods. It is defined if the differentiation matrix $[A]$ is invertible. The derivatives of the unknown function $u(x)$ can be approximated using

$$\frac{\partial^k}{\partial x_i^k} s(x_i) = Du \quad (2.25)$$

First Five Derivatives of the Radial Basis Functions

The first and second derivatives for any sufficient differentiable RBF $\phi(r)$ are

$$\frac{\partial \phi}{\partial x_i} = \frac{d\phi}{dr} \frac{\partial r}{\partial x_i} \quad (2.26)$$

$$\frac{\partial^2 \phi}{\partial x_i^2} = \frac{d\phi}{dr} \frac{\partial^2 r}{\partial x_i^2} + \frac{d^2 \phi}{dr^2} \left(\frac{\partial r}{\partial x_i} \right)^2 \quad (2.27)$$

Using equations (2.26) and (2.27), we provide the derivatives for the third, fourth and fifth derivatives

$$\frac{\partial^3 \phi}{\partial x_i^3} = \frac{d\phi}{dr} \frac{\partial^3 r}{\partial x_i^3} + 3 \left(\frac{d^2 \phi}{dr^2} \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right) \right) + \frac{d^3 \phi}{dr^3} \left(\frac{\partial r}{\partial x_i} \right)^3 \quad (2.28)$$

$$\frac{\partial^4 \phi}{\partial x_i^4} = \frac{d\phi}{dr} \frac{\partial^4 r}{\partial x_i^4} + 3 \frac{d^2 \phi}{dr^2} \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 + 4 \left(\frac{d^2 \phi}{dr^2} \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^3 r}{\partial x_i^3} \right) \right)$$



$$+6 \left(\frac{d^3 \phi}{dr^3} \left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^2 r}{\partial x_i^2} \right) \right) + \frac{d^4 \phi}{dr^4} \left(\frac{\partial r}{\partial x_i} \right)^4 \quad (2.29)$$

$$\begin{aligned} \frac{\partial^5 \phi}{\partial x_i^5} &= \frac{d\phi}{dr} \frac{\partial^5 r}{\partial x_i^5} + 5 \left(\frac{d^2 \phi}{dr^2} \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^4 r}{\partial x_i^4} \right) \right) + 10 \left(\frac{d^2 \phi}{dr^2} \left(\frac{\partial^2 r}{\partial x_i^2} \right) \left(\frac{\partial^3 r}{\partial x_i^3} \right) \right) + \dots \\ &15 \left(\frac{d^3 \phi}{dr^3} \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 \right) + 10 \left(\frac{d^3 \phi}{dr^3} \left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^3 r}{\partial x_i^3} \right) \right) + 10 \left(\frac{d^4 \phi}{dr^4} \left(\frac{\partial r}{\partial x_i} \right)^3 \left(\frac{\partial^2 r}{\partial x_i^2} \right) \right) + \frac{d^5 \phi}{dr^5} \left(\frac{\partial r}{\partial x_i} \right)^5 \end{aligned} \quad (2.30)$$

where

$$\begin{aligned} \frac{\partial r}{\partial x_i} &= \frac{x_i}{r}; \\ \frac{\partial^2 r}{\partial x_i^2} &= \frac{1 - \left(\frac{\partial r}{\partial x_i} \right)^2}{r}; \\ \frac{\partial^3 r}{\partial x_i^3} &= \frac{\left(\frac{\partial r}{\partial x_i} \right)^3 - \left(\frac{\partial r}{\partial x_i} \right) - 2r \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right)}{r^2}; \\ \frac{\partial^4 r}{\partial x_i^4} &= \frac{A_1 - A_2}{r^4}; \end{aligned}$$

where

$$A_1 = r^2 \left(3 \left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^2 r}{\partial x_i^2} \right) - \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 - 2 \left(\left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^2 r}{\partial x_i^2} \right) + r \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 + r \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^3 r}{\partial x_i^3} \right) \right)$$

and

$$A_2 = 2r \left(\left(\frac{\partial r}{\partial x_i} \right)^4 - \left(\frac{\partial r}{\partial x_i} \right)^2 - 2r \left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^2 r}{\partial x_i^2} \right) \right)$$

$$\frac{\partial^5 r}{\partial x_i^5} = \frac{B_1 - B_2}{r^4} - \frac{B_3}{r^6};$$

Furthermore

$$B_1 = r^2 \left(3 \left(\frac{\partial^2 r}{\partial x_i^2} \right) \left(\frac{\partial^3 r}{\partial x_i^3} \right) - \left(\frac{\partial^3 r}{\partial x_i^3} \right)^2 - 4r \left(\frac{\partial^2 r}{\partial x_i^2} \right) \left(\frac{\partial^3 r}{\partial x_i^3} \right) + \left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^3 r}{\partial x_i^3} \right) + r \left(\frac{\partial^2 r}{\partial x_i^2} \right) \left(\frac{\partial^3 r}{\partial x_i^3} \right) + r \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^4 r}{\partial x_i^4} \right) \right)$$

$$B_2 = 2r \left(\left(\frac{\partial r}{\partial x_i} \right)^3 \left(\frac{\partial^2 r}{\partial x_i^2} \right) - \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 - 2r \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 - 2r \left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^3 r}{\partial x_i^3} \right) \right)$$

and

$$\begin{aligned} B_3 &= 2 \left(r^3 \left(2 \left(\frac{\partial r}{\partial x_i} \right)^3 \left(\frac{\partial^2 r}{\partial x_i^2} \right) - 2 \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 - 4r \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right)^2 - 2r \left(\frac{\partial r}{\partial x_i} \right)^2 \left(\frac{\partial^3 r}{\partial x_i^3} \right) \right) - 3r^3 \left(\left(\frac{\partial r}{\partial x_i} \right)^5 - \left(\frac{\partial r}{\partial x_i} \right)^3 \right. \right. \\ &\quad \left. \left. - 2r \left(\frac{\partial r}{\partial x_i} \right) \left(\frac{\partial^2 r}{\partial x_i^2} \right) \right) \right) \end{aligned}$$

If equations (2.8), (2.9) and (2.10) are substituted in equations (2.26) – (2.30), then the various derivatives of the MQ, IQ and GIMQ RBFs can be obtained.

Existence and Uniqueness of Interpolation Matrix of Radial Basis Functions

The existence and uniqueness of an RBF interpolation problem is guaranteed if its interpolation matrix is invertible [2], however, it is difficult to characterize the class of all basis functions $\phi(r)$ that can generate non-singular interpolation matrices. An alternative is to show that the basis functions form positive definite matrices [14]. Schoenberg and Fourier series methods can be used to characterize positive definite matrices, nevertheless, these methods are not easy to compute. To this end, two criteria (i) Completely Monotone Functions and (ii) Multiply Monotone Functions are used to determine that a function is strictly positive definite and radial on $\mathbb{R}^d$.

Definition 2.1 Completely Monotone Functions (Fasshauer, 2007)

A function $\phi: [0, \infty) \rightarrow \mathbb{R}$ that is in $C[0, \infty) \cap C^\infty(0, \infty)$ and satisfies

$$(-1)^\ell \phi^{(\ell)}(r) \geq 0, \quad \ell = 0, 1, 2, \dots \quad (2.31)$$

is completely monotone on $[0, \infty)$.

Theorem 2.1 (Fasshauer, 2007)

A function $\phi: [0, \infty) \rightarrow \mathbb{R}$ is completely monotone but not constant if and only if $\phi(\|\cdot\|^2)$ is strictly positive definite and radial on $\mathbb{R}^d$ for any d .

Definition 2.2 Multiply Monotone Functions (Fasshauer, 2007)

A function $\phi: (0, \infty) \rightarrow \mathbb{R}$ which is in $C^{k-2}(0, \infty)$, $k \geq 2$ and for which $(-1)^\ell \phi^{(\ell)}(r)$ is non-negative, non-increasing and convex for $\ell = 0, 1, 2, \dots, k-2$ is called k -times monotone on $(0, \infty)$. In case $k = 1$, we require that $\phi \in C(0, \infty)$ be non-negative and non-increasing

Theorem 2.2 (Sarra and Kansa, 2009)

Let $\phi(r) = \phi(\sqrt{r}) \in C[0, \infty)$ and $\phi(r) > 0$ for $r > 0$, let $\phi'(r)$ be completely monotone and non-constant on $(0, \infty)$. Then for any set of N distinct centres $\{x_j\}_{j=1}^N$, then the $N \times N$ matrix $[A]$ with entries $a_{ij} = \phi(\|x_i - x_j\|)$ is invertible, such a function is said to be conditionally positive definite of order 1.

For details about the invertibility of the MQ RBF, [2] may be consulted, while the invertibility of the IQ and GIMQ RBFs is shown in [3]. Any function of the form

$$\phi(r) = \frac{1}{(1+r)^\beta}, \quad \beta \geq 0 \quad (2.32)$$

is completely monotone on $[0, \infty)$ since

$$(-1)^\ell \phi^{(\ell)}(r) = (-1)^{2\ell} \beta(\beta+1) \dots (\beta+\ell-1)(1+r)^{-\beta-\ell} \geq 0, \quad \ell = 0, 1, 2, \dots \quad (2.33)$$



If $\beta = 1$ and 2 in equations (2.32) and (2.33), then IQ and GIMQ are shown to be invertible.

2.6 An Algorithm for Discretizing Space Variable(s) via RBF-MOLs

Consider a PDE of the form

$$\frac{\partial u}{\partial t} + L(u) = 0, \quad x \in \Omega, t \geq 0 \quad (2.34)$$

where

$u = u(x, t)$, L = spatial derivative operator.

Let $x_1, x_2, \dots, x_N \in \Omega \subset \mathbb{R}^d$ ($d = 1, 2$ and 3) be centres obtained from the given domain (Ω). We restrict $d = 1$, since we are dealing with one-dimensional problems.

The approximate solution $u(x, t)$ for a time-dependent PDE can first be expressed as

$$u(x_i) = \sum_{j=1}^N \lambda_j(t) \phi(\|x_i - x_j\|) \quad (2.35)$$

where $\lambda_j(t)$, $j = 1, 2, \dots, N$ represent coefficients to be determined, ϕ represent a radial basis function and x_j are the centres. Applying the operator L on both sides of equation (2.35) yields

$$L(u(x_i)) = \sum_{j=1}^N \lambda_j(t) (L\phi(\|x_i - x_j\|)) \quad (2.36)$$

in matrix form, equations (2.35) and (2.36) can be written as

$$[A]\lambda = u, u = [u_1(t), u_2(t), \dots, u_N(t)]^T \quad (2.37)$$

and

$$[B]\lambda = L(u) \quad (2.38)$$

where

$$[A] = \phi(\|x_i - x_j\|), i, j = 1, 2, 3, \dots, N$$

and

$$[B] = L\phi(\|x_i - x_j\|), i, j = 1, 2, 3, \dots, N$$

From equations (2.37) and (2.38), we get

$$L(u) = ([B][A]^{-1})u$$

$$L(u) = Du \quad (2.39)$$

After the space discretization with RBFs, equation (2.35) is transformed into a semi-discretized system defined by

$$\frac{du}{dt} = Du. \quad (2.40)$$

Next, the resulting system of ODEs from equation (2.40) are integrated using the fourth order explicit Runge-Kutta method.

Algorithm for Fourth Order Explicit Runge-Kutta Method

The fourth order explicit Runge-Kutta method for integrating the system of ODEs from equation (2.40) is given below.

Suppose $u(t_n, x_n)$, $x_n \in \mathbb{R}$, then

$$k_1 = \Delta t F(t_n, x_n),$$

$$k_2 = \Delta t F\left(t_n + \frac{\Delta t}{2}, x_n + \frac{k_1}{2}\right),$$

$$k_3 = \Delta t F\left(t_n + \frac{\Delta t}{2}, x_n + \frac{k_2}{2}\right),$$

$$k_4 = \Delta t F(t_n + \Delta t, x_n + k_1),$$

$$u_{n+1} = u_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

III. RESULTS

In this Section, the RBF-MOLs developed in Section 2 will be applied to approximate a third, fourth and fifth order one-dimensional time-dependent PDEs. The methods are implemented in MATLAB 2007b on Windows 8 operating system and the results are displayed on Graphs and Tables for comparison and discussion. All the test problems are drawn from [15].

Example 1: The linear KdV equation

The linear KdV equation with a third order space derivative is defined by

$$\frac{\partial u}{\partial t} - \frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0, \quad (3.1)$$

on the domain

$$\Omega = [0, 2.5\pi],$$

with the initial condition

$$u(x, 0) = \sin(x),$$

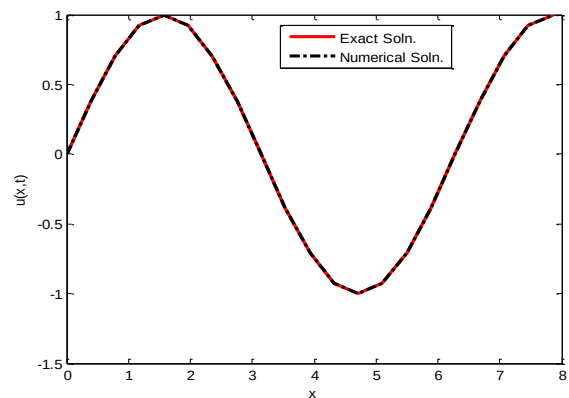
and boundary conditions

$$u(0, t) = u(2.5\pi, t).$$

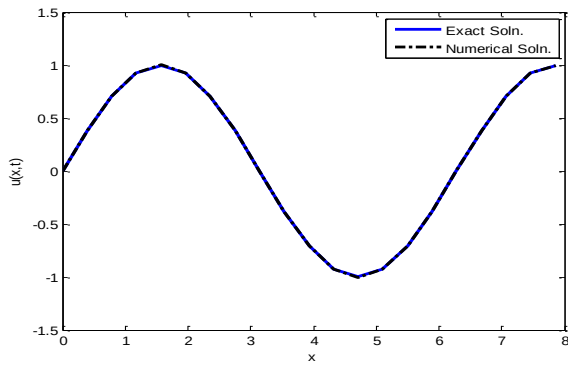
The exact solution is given by

$$u(x, t) = \sin(x + 2.5t).$$

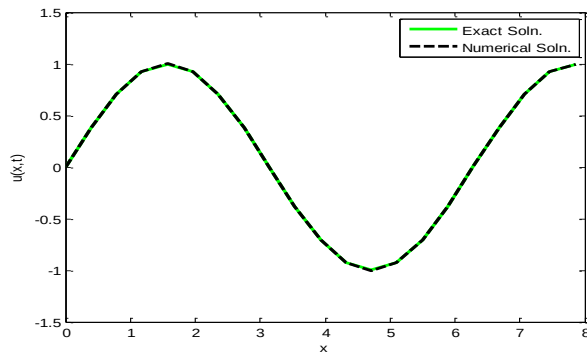
Equation (3.1) is discretized in space using the MQ, IQ and GIMQ RBFs with $N = 21$ uniformly distributed centres. The resulting systems of ODEs are solved using the fourth order Runge-Kutta method with a small time-step $\Delta t = 5.0 \times 10^{-7}$. The results for the various RBF-MOLs are provided in Table 1, while Fig. 1 is the graphical representation of the results.



(a)



(b)



(c)

Fig. 1: The Exact Solution versus the Numerical Solution using (a) IMQ RBF-MOLs, (b) IQ RBF-MOLs, (c) GIMQ RBF-MOLs for Example 1

Table 1: Summary of MQ, IQ and GIMQ RBF-MOLs for Example 1

S/ N	RBF- MOLs	N	Δt	FT	ϵ	MPE
1	MQ	21	5.0×10^{-7}	1.0×10^{-4}	7.0	4.6810×10^{-4}
2	IQ	21	5.0×10^{-7}	1.0×10^{-4}	2.0	1.0000×10^{-3}
3	GIMQ	21	5.0×10^{-7}	1.0×10^{-4}	0.9	8.9400×10^{-4}

Example 2: Linear Biharmonic Equation

Consider the linear biharmonic equation

$$\frac{\partial u}{\partial t} + \frac{\partial^4 u}{\partial x^4} = 0, \quad (3.2)$$

on the domain

$$\Omega = [0, 2\pi],$$

with the initial condition

$$u(x, 0) = \sin(x),$$

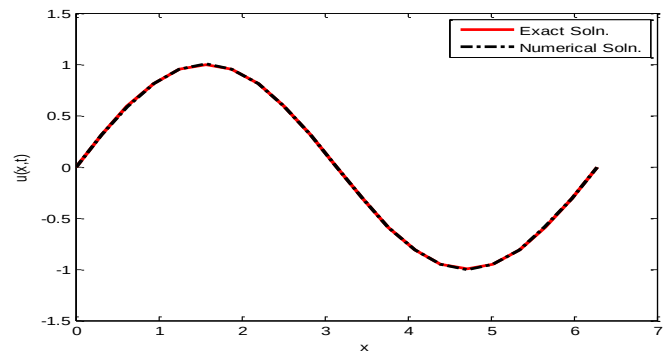
and the boundary conditions

$$u(0, t) = u(2\pi, t).$$

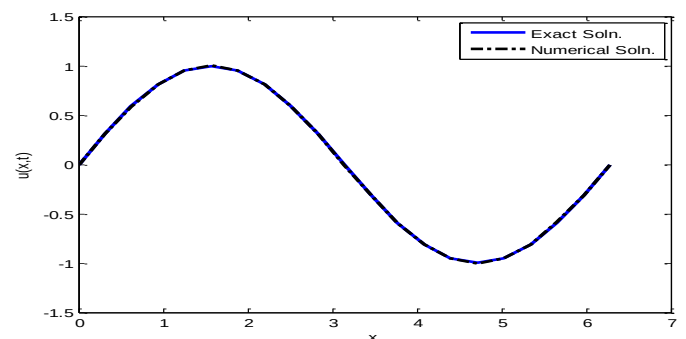
The exact solution is given by

$$u(x, t) = e^{-t} \sin(x).$$

The MQ, IQ and GIMQ RBFs were used to discretize Example 2 in space on a uniformly spaced node, $N = 21$. The resulting systems of ODEs are integrated using the forth order Runge-Kutta method with a time step of $\Delta t = 5.0 \times 10^{-7}$. Fig. 2 shows the graphical results at the 200th time step while Table 2 displays the numerical results of the various RBF-MOLs



(a)



(b)

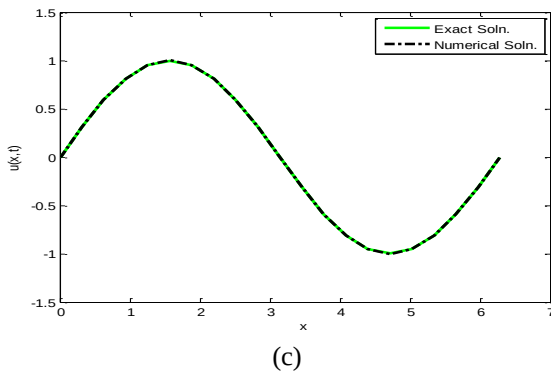


Figure 2: The Exact Solution versus the Numerical Solution using (a) IMQ RBF-MOLs, (b) IQ RBF-MOLs, (c) GIMQ RBF-MOLs for Example 2

Table 2: Summary of MQ, IQ and GIMQ RBF-MOLs for Example 2

S/N	RBF-MOLs	N	Δt	FT	ϵ	MPE
1	MQ	21	5.0×10^{-7}	1.0×10^{-4}	1.9	4.7100×10^{-4}
2	IQ	21	5.0×10^{-7}	1.0×10^{-4}	1.0	1.1300×10^{-3}
3	GIMQ	21	5.0×10^{-7}	1.0×10^{-4}	0.9	1.4000×10^{-4}

Example 3: A fifth order linear time-dependent PDE

Consider the linear time-dependent PDE that contains a fifth order derivative in space

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial^5 u}{\partial x^5} = 0, \quad (3.3)$$

on the domain

$$\Omega = [0, 4\pi],$$

with the initial condition

$$u(x, 0) = \sin(x),$$

and the boundary conditions

$$u(0, t) = u(4\pi, t).$$

The exact solution is given by

$$u(x, t) = \sin(x - 4t).$$

The space discretization of Example 3 is obtained via the MQ, IQ and GIMQ RBFs on uniformly distributed collocation points, $N = 31$. A small time-step $\Delta t = 5.0 \times 10^{-7}$ is used to advance the system of ODEs emanating from the space discretization using the fourth order Runge-Kutta method.

Table 3 shows the numerical results while Fig. 3 displays the graphical results of the exact and the numerical solutions.

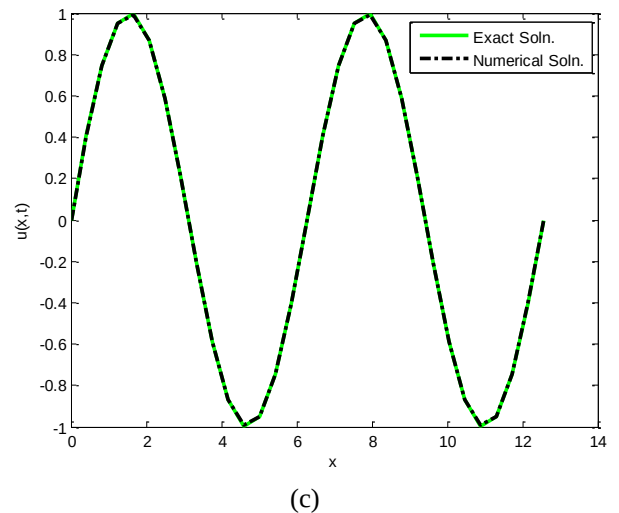
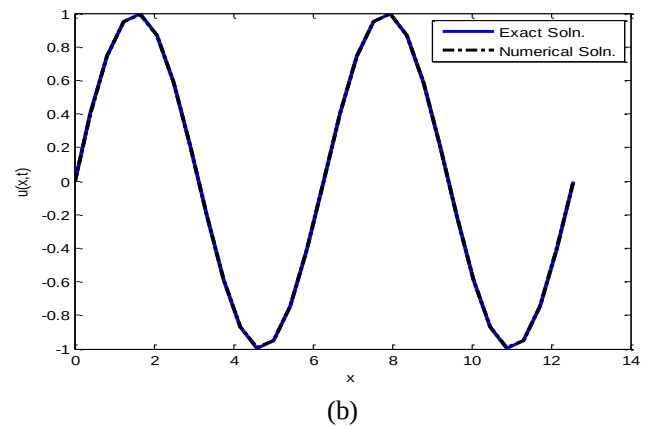
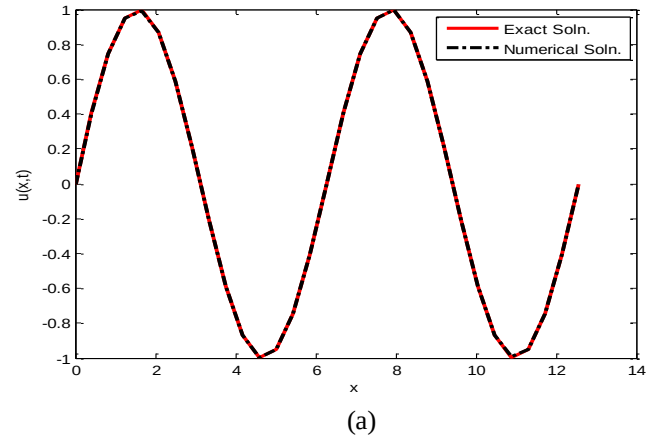


Figure 3: The Exact Solution versus the Numerical Solution using (a) MQ RBF-MOLs, (b) IQ RBF-MOLs, (c) GIMQ RBF-MOLs for Example 3

Table 3: Summary of MQ, IQ and GIMQ RBF-MOLs for Example 3



S/ N	RBF- MOLs	N	Δt	FT	ϵ	MPE
1	MQ	31	5.0×10^{-7}	1.0×10^{-4}	0.9	3.3100×10^{-3}
2	IQ	31	5.0×10^{-7}	1.0×10^{-4}	0.5	5.6000×10^{-3}
3	GIMQ	31	5.0×10^{-7}	1.0×10^{-4}	0.4	5.7000×10^{-3}

IV. DISCUSSION

Example 1 is a linear KdV time-dependent PDE with a third order space derivative. The numerical solution of this problem was obtained by first discretizing the space derivatives using MQ, IQ and GIMQ RBFs with equally spaced centres, $N = 21$ on the domain $[0, 2.5\pi]$. A suitable estimate for the shape parameter $\epsilon = 7.0, 2.0$ and 0.9 were found for the MQ, IQ and GIMQ RBFs respectively using the brute force method. A single differentiation matrix was used to discretize the entire space derivatives since this is a linear problem. After an extensive experimentation using different time-steps, a time step of $\Delta t = 5.0 \times 10^{-7}$ was used to advance the solution of the resulting systems of ODEs in time up to the 200th time level using the fourth order Runge-Kutta method. Table 1 shows the numerical values of the results while Fig. 1 displays the plots of the numerical solution and the exact solution. We observed that the point-wise errors for the three RBF-MOLs are within the same range, i.e. at a final time, $FT = 1.0 \times 10^{-4}$, the maximum point-wise error of the MQ, IQ and GIMQ RBFs are given as 4.681×10^{-4} , 1.0×10^{-3} and 8.940×10^{-4} respectively. Fig. 1 which provides the plots of both the exact and numerical solutions show that the various RBF-MOLs for this problem gave accurate approximations.

A linear biharmonic equation containing a fourth order derivative in space forms Example 2. To find an approximate solution to this example using RBF-MOLs, the MQ, IQ and GIMQ RBFs were used for the discretization of the space derivatives on the domain $[0, 2\pi]$ which was divided into $N = 21$ equally spaced centres. The brute force method was used to estimate the shape parameters for the various RBFs as $1.9, 1.0$ and 0.9 respectively. As a linear time-dependent PDE, the entire space derivatives were discretize using a single differentiation matrix. we choose a time-step of 5.0×10^{-7} and used it to integrate the systems of ODEs at 200th time level using the 4th order Runge-Kutta method. The results are

displayed in Table 2 and Fig. 2. Table 2 reveals that the maximum point-wise error for the three RBF-MOLs at the different time level were observed to be within the same range. For instance, the maximum point-wise error of the MQ, IQ and GIMQ RBF-MOLs are given as 4.711×10^{-4} , 1.1×10^{-3} and 1.4×10^{-3} respectively. Fig. 2 show that the various RBF-MOLs provide good approximation of Example 2.

The efficiency of the MQ, IQ and GIMQ RBF-MOLs was tested on a linear time-dependent PDE that contains a fifth order space derivative. $N = 31$ equally spaced centres were used to subdivide the domain $[0, 4\pi]$ while the various RBFs were applied to discretize the space derivatives. Fig. 3 shows that the various shape parameter obtained by the brute force method were $0.9, 0.5$ and 0.4 respectively. The Runge-Kutta fourth order method was used to integrate the system of ODEs arising from the space discretization using a time step of $\Delta t = 5.0 \times 10^{-7}$. Fig. 3 shows the various plots of the exact solution and the numerical solutions as well as the point-wise error for all the three RBF-MOLs. The following maximum point-wise error, 3.3×10^{-3} , 5.6×10^{-3} and 5.7×10^{-3} for the IMQ, IQ and GIMQ RBF-MOLs at the 200th time level recorded in Table 3 showed that the maximum point-wise error lie within the same range.

V. CONCLUSION

Some radial basis function method of lines (RBF-MOLs) namely, the MQ-RBF-MOLs, IQ-RBF-MOLs and the GIMQ-RBF-MOLs were developed for solving some higher order one-dimensional time-dependent PDEs. The MQ-RBF-MOLs is the extension of the work of Sarra and Kansa (2009), while the IQ-RBF-MOLs, and GIMQ-RBF-MOLs is the extension of the work of Luga *et al.* (2017a & 2017b) to solve higher order dimensional problems. Three test problems on a third, fourth and fifth order time-dependent PDEs in one-dimension were used to validate the performance of the developed RBF-MOLs. The point-wise error from the results revealed that the three RBF-MOLs provided good numerical approximations of the exact solutions, however, the MQ-RBF-MOLs provided the best approximations.

VI. REFERENCES

- [1] Chen, W., Fu, Z.-J. and Chen, C. S. (2014). *Recent Advances in Radial Basis Function Collocation Method*. New York, Springer Heidelberg.
- [2] Sarra, S. A. and Kansa, E. J. (2009). *Multiquadric Radial Basis Function Approximation Methods for the Numerical Solution of Partial Differential Equations*, California, Technical Science Press.
- [3] Fasshauer, G. E. (2007). *Meshfree Approximation Methods with MATLAB*, Singapore, World Scientific Publishing Co. Ltd.



- [4] Fasshauer, G. and McCourt, M. (2015). *Kernel-based Approximation Methods using MATLAB*. Singapore, World Scientific Publishing Co. Ltd, Singapore.
- [5] Kansa, E. J. (1990a). Multiquadrics-A Scattered Data Approximation Scheme with Application to Computational Fluid Dynamics-I: Surface Approximations and Surface Estimates *Computer and Mathematics with Applications*, 19(8-9): 127-145.
- [6] 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 and Mathematics with Applications*, 19(8-9): 147-161.
- [7] Zitova, B. and Flusser, J. (2003). Image Registration Methods: A Survey. *Image Vis. Comput.*, 21(11): 977-1000
- [8] Tsai, C. C., Young, D. L., Chiang, J. H. and Lo, D. C. (2006). The Method of Fundamental Solutions for Solving Option Pricing Models. *Appl. Math. Comput.*, 181(1): 390-401.
- [9] Tatari, M. and Dehghan, M. (2009). On the Solution of the Non-Local Parabolic Partial Differential Equations via Radial Basis Functions. *Applied Mathematics Modelling*, 33:1729-1738.
- [10] Schiesser, W. E. (1991). *The Numerical Method of Lines: Integration of Partial Differential Equations*, San Diego, Academic Press.
- [11] Luga, T., Aboiyar, T. and Adey, S. O. (2017a). Radial Basis Function Methods for Approximating Time-Dependent Partial Differential Equations in One Space Dimension. *Journal of Pure and Applied Sciences*, 9(1):182-200.
- [12] Luga, T., Aboiyar, T. and Onah, E. S. (2017b) Radial Basis Function Methods for Approximating the Cubic Nonlinear Schrödinger Equation in One Space Dimension. *Journal of the Nigerian Association of Mathematical Physics*, 43: 325-332.
- [13] Micchelli, C. (1986). Interpolation of Scattered Data: Distance Matrices and Conditionally Positive Definite Functions. *Constructive Approximation*, 2 (2,3):11-22.
- [14] Iske, A. (2003). Radial Basis Functions: Basics Advanced Topics and Meshfree Methods for Transport Problems. *Rendicoti Del Seminario Matematica*, 61(3):247-286.
- [15] Yingda, C. and Chi-Wang, S. (2008). A Discontinuous Galerkin Finite Element Method for Time-Dependent Partial Differential Equations with Higher Order Derivatives. *Mathematics of Computations*, 77(262):699-730.

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

ABOUT IJEAST

International Journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high-quality research papers in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

FOCUS AREAS

- Engineering
- Applied Science
- Technology
- Innovation & Development
- Interdisciplinary Studies



PEER REVIEWED

All submissions are rigorously peer reviewed to ensure quality.



OPEN ACCESS

Free and unrestricted access to research for all.



GLOBAL REACH

Connecting researchers and professionals worldwide.



TIMELY PUBLICATION

We ensure a swift and efficient publication process.



For more information, visit our website

www.ijeast.com



INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

✉ editor@ijeast.com

🌐 www.ijeast.com

📍 India



2455-2143