# Solving time-dependent PDEs with a meshless IRBFN-based Method

L. Mai-Cao[1] and T. Tran-Cong[2]

**Abstract:** *In our previous work, Indirect Radial Basis Function Network (IRBFN) method has proved to be a highly accurate tool for approximating multivariate functions and solving elliptic PDEs. A recent development of the method for solving time-dependent PDEs will be presented in this paper. Two numerical schemes combining the IRBFN method with different time integration techniques, based on either fully or semi-discrete framework, are proposed. For the problems considered in this work, the accuracy of the method is not very sensitive to the network parameters. In the particular case of parabolic PDEs, the method can tolerate a rather wide range of values of the shape parameter while yielding highly accurate results. Illustrative parabolic PDEs, hyperbolic PDEs and advection-diffusion equations are solved by the proposed method, and the results compare favourably in terms of accuracy and efficiency with those from other numerical methods such as finite difference, finite element, boundary element and direct RBFN methods.*

## 1 Introduction

In recent years, radial basis functions (RBFs) have been increasingly used as effective tools for interpoplating multivariate data and functions [1, 2, 3]. The idea of using RBFs for solving PDEs was first proposed by Kansa [4, 5] who applied a global MQ scheme to solve parabolic, hyperbolic and elliptic PDEs. This method is hereby referred to as the Direct Radial Basis Function Network (DRBFN) method. Various schemes were proposed [5, 6, 7], just to name a few, to solve transient problems.

Recently, Mai-Duy and Tran-Cong [8, 9] proposed a new method, namely the Indirect Radial Basis Function Network (IRBFN) for approximating functions and their derivatives, and solving differential equations. In the DRBFN method, a function is first approximated by the RBFN and its derivatives are then calculated by differentiating such closed form RBFN approximation. In the IRBFN method, on the other hand, the highest order derivatives are first decomposed into radial basis functions. Lower derivatives and the function

[1]University of Southern Queensland, Faculty of Engineering and Surveying, Toowoomba, QLD 4350, Australia (lanmai@usq.edu.au).

[2]University of Southern Queensland, Faculty of Engineering and Surveying, Toowoomba, QLD 4350, Australia (trancong@usq.edu.au).

itself are then successively obtained via symbolic integrations. It was reported that the indirect method performs better than the direct method in terms of accuracy for approximating both functions and their derivatives [8]. The IRBFN method has been successfully applied to solve second-order elliptic PDEs and steady Navier-Stokes equations [9, 10].

In this paper, further developments of the IRBFN method for solving transient problems are presented. In particular, the method is extended to solve time-dependent parabolic PDEs, hyperbolic PDEs, and advection-diffusion equations. Comparisons of accuracy and efficiency between the IRBFN method and other numerical methods, and sensitivity studies of the proposed method to various parameters such as the network shape parameter, grid density and time step size are also discussed.

# 2 IRBFN-based schemes for transient problems

Consider a general initial-boundary value problem

$$\begin{aligned} \frac{\partial y}{\partial t} + \mathcal{L}y = &\quad f \quad \text{in } Q_T := (0,T)\times\Omega, \\ \mathcal{B}y = &\quad g_1 \quad \text{on } \Sigma_T := (0,T)\times\partial\Omega, \quad y = g_2 \quad \text{on } \Omega,\ t=0, \end{aligned} \tag{1}$$

where $\Omega$ is a bounded domain in $\mathbb{R}^d$, $d = 1,2,3$, with boundary $\partial\Omega$; $T > 0$ is a prescribed time-level; $f = f(\mathbf{x},t)$, $g_1 = g_1(\mathbf{x},t)$ and $g_2 = g_2(\mathbf{x})$ are given functions; $\mathcal{L}$ is a differential operator, and $\mathcal{B}$ is the Dirichlet or Neumann boundary operator. The IRBFN approximant $u(\mathbf{x},t)$ to function $y(\mathbf{x},t)$ [8] is rewritten for transient problems as follows

$$\begin{aligned} u_{,jj}(\mathbf{x},t) &= \sum_{i=1}^{N} w^{(i)}(t)\tilde{h}^{(i)}(\mathbf{x}), \quad u_{,j}(\mathbf{x},t) = \sum_{i=1}^{N} w^{(i)}(t)\,\hat{h}^{(i)}(\mathbf{x}) + C_1(x_{k;k\neq j},t), \\ u(\mathbf{x},t) &= \sum_{i=1}^{N} w^{(i)}(t)\,h^{(i)}(\mathbf{x}) + C_1(x_{k;k\neq j},t)\,x_j + C_2(x_{k;k\neq j},t), \end{aligned} \tag{2}$$

where the constants of integration $C_1$ and $C_2$ as well as the set of unknown weights $\{w^{(i)}(t)\}_{i=1}^{N}$ are now dependent on time and to be found at each time step.

**Fully discrete schemes:** In fully discrete schemes, the problem (1) is discretized with respect to both time and space variables. The temporal discretisation is achieved by the standard θ-scheme, $0 \le \theta \le 1$, and the spatial discretization is based on the IRBFN method. The application of the θ-scheme to problem (1) yields

$$\frac{u^{n+1}-u^n}{\Delta t} + \theta\,\mathcal{L}u^{n+1} + (1-\theta)\,\mathcal{L}u^n = \theta\,f^{n+1} + (1-\theta)\,f^n, \quad \mathcal{B}u^{n+1} = g_1, \tag{3}$$

where $t^{n+1} = t^n + \Delta t$, $u^{n+1} \approx y(\mathbf{x},t^{n+1})$, and $u^0 = g_2(\mathbf{x})$. The time-discrete system (3) is then discretized in space by replacing $u$, $\mathcal{L}u$ and $\mathcal{B}u$ with the IRBFN approximants in (2) at each time step on the entire set of collocation points $\{\mathbf{x}^{(i)}\}_{i=1}^{N}$. Finally, the set of unknown weights $\{w^{(i)}(t)\}_{i=1}^{N}$ can be found by minimizing the corresponding sum-squared-error (SSE) at each time step in the sense of the general linear least-squares principle. For

illustrative purpose, the SSE formulation for a 2D problem is written as follows

$$SSE^{n+1} = \sum_{\mathbf{x}^{(i)}\in\Omega} \Big[u^{n+1}(\mathbf{x}^{(i)}) + \theta\,\Delta t\, Lu^{n+1}(\mathbf{x}^{(i)}) - u^{n}(\mathbf{x}^{(i)}) + (1-\theta)\,\Delta t\,\left(Lu^{n}(\mathbf{x}^{(i)}) - f^{n}\right) - \theta\,\Delta t\, f^{n+1}\Big]^2 + \sum_{\mathbf{x}^{(i)}\in\partial\Omega}\left[Bu^{n+1}(\mathbf{x}^{(i)})\right]^2 + \sum_{\mathbf{x}^{(i)}\in\Omega}\left[u_1^{n+1}(\mathbf{x}^{(i)}) - u_2^{n+1}(\mathbf{x}^{(i)})\right]^2, \quad (4)$$

where $Lu$ and $Bu$ are IRBFN-based approximations to $\mathcal{L}u$ and $\mathcal{B}u$, respectively. It should be noted that the term $u_1^{n+1}(\mathbf{x}^{(i)})$ is symbolically obtained via $u_{,11}^{n+1}(\mathbf{x}^{(i)})$ and $u_2^{n+1}(\mathbf{x}^{(i)})$ via $u_{,22}^{n+1}(\mathbf{x}^{(i)})$ in the manner of equations (2). The system (4) is solved at each time step for the set of unknown weights $\{w^{(i)}(t)\}_{i=1}^{N}$ until the prescribed time $T$ is reached.

**Semi-discrete schemes:** In semi-discrete schemes, problem (1) is first discretized in space while still continuously dependent on time. As a result, a system of ordinary differential equations (ODE) of the unknown weight vector $\mathbf{w}$ is obtained

$$\mathbf{H}\frac{d\mathbf{w}}{dt} = \mathbf{G}(t,\mathbf{w}), \quad (5)$$

where $\mathbf{G}$ is the vector obtained by applying IRBFN approximants to $\mathcal{L}y$ and $f$ in (1) on the entire set of collocation points $\{\mathbf{x}^{(i)}\}_{i=1}^{N}$; $\mathbf{H}$ and $\mathbf{w}$ are the matrix and vector defined by (2) in a compact matrix form $\mathbf{u} = \mathbf{H}\mathbf{w}$. Any high-order ODE solver can be then applied to discretize the system (5). For illustrative purpose, the fourth-order Runge-Kutta scheme is described here. The ODE system (5) can be rewritten as $\frac{d\mathbf{w}}{dt} = \mathbf{F}(t,\mathbf{w})$, where $\mathbf{F} = \mathbf{H}^{-1}\mathbf{G}$. In the case of multiquadric RBF (which is used here), it is known that the coefficient matrix $\mathbf{H}$ is always invertible for a set of distinct collocation points [11]. At each time step, four function evaluations of $\mathbf{F}$ are performed taking into account the boundary conditions described in (1). The updated value of $\mathbf{w}$ at time step $t^{n+1}$ is therefore calculated as follows

$$\begin{aligned}
\mathbf{K}_1 &= \Delta t\,\mathbf{F}\left(t^n,\mathbf{w}^n\right), \quad \mathbf{K}_2 = \Delta t\,\mathbf{F}\left(t^{n+1/2},\mathbf{w}^n + \mathbf{K}_1/2\right),\\
\mathbf{K}_3 &= \Delta t\,\mathbf{F}\left(t^{n+1/2},\mathbf{w}^n + \mathbf{K}_2/2\right), \quad \mathbf{K}_4 = \Delta t\,\mathbf{F}\left(t^{n+1},\mathbf{w}^n + \mathbf{K}_3\right),\\
\mathbf{w}^{n+1} &= \mathbf{w}^n + \frac{\mathbf{K}_1 + 2\mathbf{K}_2 + 2\mathbf{K}_3 + \mathbf{K}_4}{6},
\end{aligned} \quad (6)$$

where $t^{n+1} = t^n + \Delta t$, $\Delta t$ is the time step size; $\mathbf{w}^n$ is the weight vector at time $t^n$. It should be noted that other one-step and multi-step schemes such as modified Rosenbrock, Adams-based predictor-corrector methods, and backward differentiation formula can also be used for both non-stiff and stiff problems [12].

# 3 Numerical Examples

Five examples are used to demonstrate the performance of the present IRBFN method, namely the 1D and 2D diffusion equations, the 1D and 2D wave equations, and the 1D advection-diffusion equation. Naturally, the method is not limited to two dimensions, however, the examples are chosen because of the availability of the corresponding exact solutions. The results are compared with those obtained by FDM, FEM, BEM and

DRBFN method in terms of accuracy and efficiency. Sensitivity studies of the proposed method to the shape parameter, grid density and the time step size are performed to investigate the influence of these parameters on the accuracy of the solutions. For sensitivity studies, root-mean-squared-error throughout the time domain is used, which is given by $RMSE = \sqrt{\frac{1}{N_T}\sum_{k=1}^{N_T}\left(u_a^{(k)} - u^{(k)}\right)^2}$, where $u$ and $u_a$ are numerical and analytical solutions, respectively; $N_T$ is the total number of time steps. Fully discrete schemes are used for the diffusion equations, and semi-discrete schemes for the wave and advection-diffusion equations. In general, the present IRBFN method yields superior accuracy and efficiency. Specifically, better accuracy is obtained even with coarser grids and larger time step sizes. For parabolic PDEs, the method can tolerate a wide range of values of the shape parameter ($\beta$ ranges from 6 to 12). Following is a more detailed discussion of one of the examples.

Figure 1 depicts the performance of the present IRBFN method in the case of the 2D diffusion equation, which is given by $\frac{\partial u}{\partial t} = \nabla^2 u + g(x,y,t)$ defined on $0 \le x \le 1,\ 0 \le y \le 1$. The forcing function is given by $g(x,y,t) = \sin x \sin y(2\sin t + \cos t)$ The initial and boundary conditions are appropriate to the analytical solution $u = \sin x \sin y \sin t$. In this example, a fully discrete scheme is used with $\theta = 0.5$. Using the variation of parameters technique, Ingber and Phan-Thien [13] obtained a particular solution which involves integration of the modified Bessel's functions. In the present IRBFN method, on the other hand, integrals involved in solving PDEs are obtained in closed form and therefore only the function needs be evaluated numerically. Figure 1 shows that the IRBFN method is both accurate and efficient in comparison with the BEM. The IRBFN method yields more accurate result even with a coarser grid M1 ($4 \times 4$ interior, $9 \times 9$ boundary points) than the BEM does with a finer mesh M2 ($9 \times 9$ interior points, $17 \times 17$ boundary points). Furthermore, the figure also shows that the IRBFN method maintains its higher accuracy compared to the BEM throughout the whole time domain. Both methods use the same time step size $\Delta t = 0.25$. The figure shows that the maximum absolute error of the IRBFN method does not change significantly within a range of $\beta$'s values from 6 to 12. The accuracy of the fully discrete scheme based on the $\theta$-scheme is strongly influenced by the number of time-steps. As the grid size increases from Figures 1(a)(c) to more than 3 times denser in Figures 1(b)(d), the maximum error did not decrease noticeably, indicating mesh convergence. However, as the time step size decreases from Figures 1(a)(b) to half the value in Figures 1(c)(d), the error decreases by an order of magnitude. This means that errors by the $\theta$ time-stepping scheme used in the fully discrete scheme dominate those due to spatial discretization. Thus, to improve the accuracy of the solution by the fully discrete scheme, an increase in the number of time steps, instead of denser grid, should be used.

# 4 Conclusions

The combination of the high order IRBFN method for spatial discretisation with two time integration schemes based on either fully or semi-discrete framework has resulted in a highly accurate and efficient approach for solving time-dependent PDEs. The method also exhibits tolerance to the RBF shape parameter. In the present case of semi-implicit temporal discretisation, while the fully discrete scheme is unconditionally stable, the semi-discrete scheme offers the ability of using higher order ODE solvers.

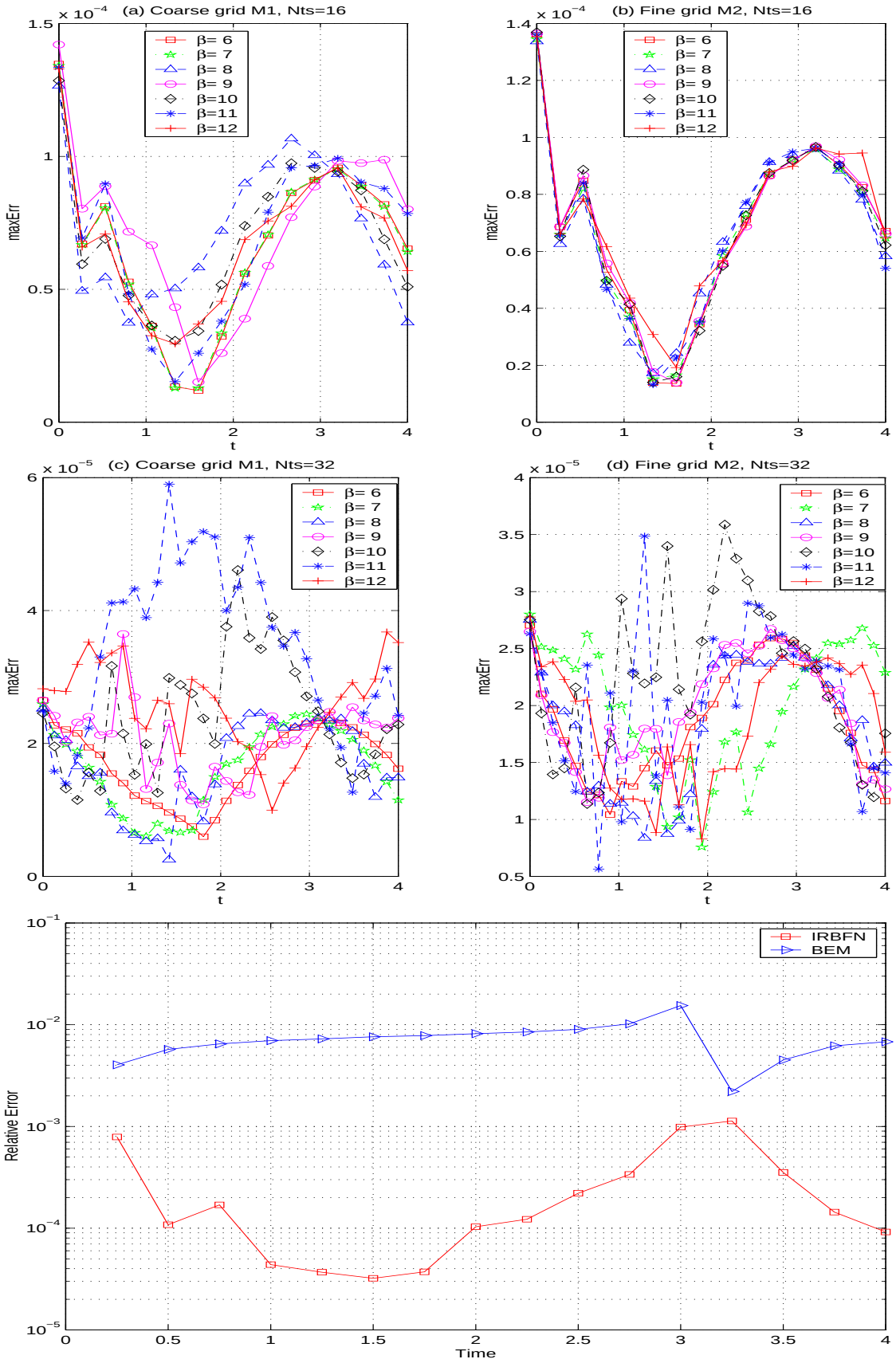


Figure 1: Example 2: 2D diffusion equation. The influence of time step size, shape parameter and grid density on the solution accuracy were investigated. The bottom plot shows that the IRBFN results are superior to the BEM results [13].

**Acknowledgements:** L. Mai-Cao is supported by a USQ and a Faculty scholarship.

# References

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

[2] R. Franke. Scattered data interpolation: Test of some methods. *Mathematics of Computation*, 38(157):181–200, 1982.

[3] P. Lancaster and K. Salkauskas. *Curve and Surface Fitting: An Introduction*. Academic Press, New York, USA, 1986.

[4] E.J. Kansa. Multiquadrics - A scattered data approximation scheme with applications to computational fluid-dynamics–I. Surface approximations and partial derivative estimates. *Computers and Mathematics with Applications*, 19(8/9):127–145, 1990.

[5] E.J. Kansa. 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, 1990.

[6] M. Zerroukat, H. Power, and C.S. Chen. A numerical method for heat transfer problems using collocation and radial basis functions. *International Journal for Numerical Methods in Engineering*, pages 1263–1278, 1998.

[7] M. Zerroukat, K. Djidjeli, and A. Charafi. Explicit and implicit meshless methods for linear advection-diffusion-type partial differential equations. *International Journal for Numerical Methods in Engineering*, 48:19–35, 2000.

[8] N. Mai-Duy and T. Tran-Cong. Approximation of function and its derivatives using radial basis function networks. *Applied Mathematical Modelling*, 27:197–220, 2003.

[9] N. Mai-Duy and T. Tran-Cong. Numerical solution of differential equations using multiquadric radial basis function networks. *Neural Networks*, 14:185–199, 2001.

[10] N. Mai-Duy and T. Tran-Cong. Numerical solution of Navier-Stokes equations using multiquadric radial basis function networks. *International Journal for Numerical Methods in Fluids*, 37:65–86, 2001.

[11] C.A. Micchelli. Interpolation of scattered data: Distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2:11–22, 1986.

[12] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, USA, 1988.

[13] M.S. Ingber and N. Phan-Thien. A boundary element approach for parabolic differential equations using a class of particular solutions. *Applied Mathematical Modelling*, 16:124–132, 1992.