Blucher Mechanical Engineering Proceedings
May 2014, vol. 1 , num. 1
www.proceedings.blucher.com.br/evento/10wccm

NUMERICAL SOLUTION OF FOKKER-PLANCK EQUATION USING THE INTEGRAL RADIAL BASIS FUNCTION NETWORKS

C.-D. Tran[1], N. Mai-Duy[1], T. Tran-Cong[1]

[1] Computational Engineering and Science Research Centre (CESRC), Faculty of Engineering and Surveying, University of Southern Queensland, Toowoomba, QLD 4350, Australia (thanh.tran-cong@usq.edu.au)

Abstract. *The Fokker Planck Equation (FPE) is a partial differential equation for the probability density and transition probability of a random process. Owing to its broad range of applications, the FPE has been an interesting research topic. Recently, Radial basis functions (RBFs) have emerged as a powerful numerical tool for solving partial differential equations and this paper reports an integrated RBFs (IRBFs) based numerical method for the solution of FPEs. The use of integration to construct RBF approximants helps avoid the reduction in convergence rate caused by differentiation [1]. Numerical experiments showed that IRBF methods can yield accurate solutions on a much coarser mesh, thus reducing the computational effort required for a given degree of accuracy.*

Keywords: *Fokker-Planck Equation, Parabolic partial differential equation, Integrated Radial Basis Functions, Collocation point.*

1. INTRODUCTION

The Fokker-Planck Equation (FPE) is used to describe a stochastic process in diverse fields, including plasma physics, biophysics, engineering, neurosciences, nonlinear hydrodynamics, and polymer physics. The FPE is a partial differential equation for the probability density and transition probability of such random processes. Owing to its broad range of applications, the FPE has attracted significant attention of several researchers.

Generally, it is difficult to obtain an analytic solution to a FPE, especially if no separation of variables is possible or if the number of variables is large. Various numerical methods for solving FPEs were devised via the transformation of a FPE to a Schrodinger equation or numerical integration methods [2]. Numerical methods for the solution of FPEs include Finite Difference Method for 2D-problems [3], Variational iteration method [4, 5], Moving Finite element method [6] and Homotopy perturbation method [7]. Recently, Radial basis functions (RBFs) have proved to be a powerful numerical tool for approximation of scattered data, and as a result, have found applications in various fields of research interests [8]. For a numerical solution of partial differential equations in engineering and sciences, RBF based collocation methods [9, 10], have increasingly been a focus of research efforts. For the direct of Kansa

approach [10], various improvements can be made for solving larger problems. For example, one can improve the condition number of the system matrix by a localisation of the RBF approximants [11] or domain composition techniques [12]. More recently, numerical schemes, based on the integrated RBFs (IRBFs) approach, for solving differential differential equations were reported [1, 13]. The use of integration to construct the RBF approximants is expected to overcome the problem of reduced convergence rate caused by differentiation [1]. Numerical experiments showed that IRBF based methods can yield accurate solutions on a coarse mesh [13], and thus have the ability to reduce the computational effort required for a given degree of accuracy. In this work, we present a collocation technique incorporating the one dimensional integrated RBFs (1D-IRBF) for a numerical solution of FPEs.

The paper is organized as follows. Section 2 gives a short review of several forms of the FPEs. In section 3, the discretization of a FPE using the 1D-IRBF method is detailed. Several numerical examples are then discussed in section 4 with a conclusion in section 5.

2. THE FOKKER PLANCK EQUATION

Consider the FPE for a field variable $u(x,t)$ of 1–D independent variables x and time t as follows [2].

$$\frac{\partial u(x,t)}{\partial t} = \left[-\frac{\partial}{\partial x}A(x) + \frac{\partial^2}{\partial x^2}B(x)\right]u(x,t) \tag{1}$$

where $A(x) > 0$ is the drift coefficient; $B(x) > 0$ the diffusion coefficient. The initial condition is given by

$$u\left(x,0\right) = f\left(x\right), \qquad x \in \Re \tag{2}$$

where $f(x)$ is a known function. If the drift and diffusion coefficients depend on time, Eq. (1) is expressed as

$$\frac{\partial u(x,t)}{\partial t} = \left[-\frac{\partial}{\partial x}A(x,t) + \frac{\partial^2}{\partial x^2}B(x,t)\right]u(x,t) \tag{3}$$

In certain applications, the drift and diffusion coefficients are dependent on the distribution function $u(x,t)$ itself and the FPE can be expressed (see [2, 4] for details) by

$$\frac{\partial u(x,t)}{\partial t} = \left[-\frac{\partial}{\partial x}A(x,t,u) + \frac{\partial^2}{\partial x^2}B(x,t,u)\right]u(x,t) \tag{4}$$

Generally, Eq. (4) for a vector variable $\mathbf{x} = (x_1, x_2, ..., x_n)^T$ is rewritten as

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = \left[-\sum_{i=1}^{n}\frac{\partial}{\partial x_i}A_i\left(\mathbf{x},t,u\right) + \sum_{i,j=1}^{n}\frac{\partial^2}{\partial x_i x_j}B_{i,j}\left(\mathbf{x},t,u\right)\right]u(\mathbf{x},t) \tag{5}$$

Mathematically, FPE is a second–order parabolic partial differential equation (more details can be found in [2]). In the next sections, a 1D-IRBF based computational technique is described for the numerical solution of FPEs.

3. 1D-IRBF MESHFREE METHOD FOR THE SOLUTION OF FPEs

For the sake of explanation, consider the FPE in the form (1) with $(x,t) \in \Omega \times [0,T]$.

3.1. 1D-IRBFs for a time dependent function

At a given time t, the highest-order derivative of the dependent variable $u(x,t)$ (the second order for a FPE) is decomposed as

$$\frac{\partial^2 u(x,t)}{\partial x^2} = \sum_{i=1}^{N_x} w_i(t) G_i^{[2]}(x), \tag{6}$$

where $\{w_i(t)\}_{i=1}^{N_x}$ is the set of RBF weights; $\{G_i^{[2]}(x)\}_{i=1}^{N_x}$ the set of Multi-quadric RBFs (MQ-RBFs), generally considered as one of the best RBFs for the approximation of a function [14], and given by

$$G_i^{[2]}(x) = \left((x-c_i)^2 + a_i^2\right)^{1/2}, \tag{7}$$

where $\{c_i\}_{i=1}^{N_x}$ is a set of centres and $\{a_i\}_{i=1}^{N_x}$ a set of MQ-RBF widths [15].

The corresponding first-order derivative and the function itself are then determined through integration as follows.

$$\frac{\partial u(x,t)}{\partial x} = \sum_{i=1}^{N_x} w_i(t) G_i^{[1]}(x) + C_1(t), \tag{8}$$

$$u(x,t) = \sum_{i=1}^{N_x} w_i(t) G_i^{[0]}(x) + C_1(t)x + C_2(t), \tag{9}$$

where $G_i^{[1]}(x) = \int G_i^{[2]}(x)dx$, $G_i^{[0]}(x) = \int G_i^{[1]}(x)dx$ and C_1 and C_2 are unknown constants of integration at time t.

Collocating equations (6), (8) and (9) at a set of grid points $\{x_i\}_{i=1}^{N_x}$ yields the following set of algebraic equations

$$\frac{\partial^2 \widetilde{\mathbf{u}}(x,t)}{\partial x^2} = \widetilde{\mathbf{G}}^{[2]}(x)\widetilde{\mathbf{w}}(t), \tag{10}$$

$$\frac{\partial \widetilde{\mathbf{u}}(x,t)}{\partial x} = \widetilde{\mathbf{G}}^{[1]}(x)\widetilde{\mathbf{w}}(t), \tag{11}$$

$$\widetilde{\mathbf{u}}(x,t) = \widetilde{\mathbf{G}}^{[0]}(x)\widetilde{\mathbf{w}}(t), \tag{12}$$

where

$$\widetilde{\mathbf{G}}^{[2]} = \begin{bmatrix} G_1^{[2]}(x_1) & G_2^{[2]}(x_1) & \cdots & G_{N_x}^{[2]}(x_1) & 0 & 0 \\ G_1^{[2]}(x_2) & G_2^{[2]}(x_2) & \cdots & G_{N_x}^{[2]}(x_2) & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ G_1^{[2]}(x_{N_x}) & G_2^{[2]}(x_{N_x}) & \cdots & G_{N_x}^{[2]}(x_{N_x}) & 0 & 0 \end{bmatrix},$$

$$\widetilde{\mathbf{G}}^{[1]} = \begin{bmatrix} G_1^{[1]}(x_1) & G_2^{[1]}(x_1) & \cdots & G_{N_x}^{[1]}(x_1) & 1 & 0 \\ G_1^{[1]}(x_2) & G_2^{[1]}(x_2) & \cdots & G_{N_x}^{[1]}(x_2) & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ G_1^{[1]}(x_{N_x}) & G_2^{[1]}(x_{N_x}) & \cdots & G_{N_x}^{[1]}(x_{N_x}) & 1 & 0 \end{bmatrix},$$

$$\widetilde{\mathbf{G}}^{[0]} = \begin{bmatrix} G_1^{[0]}(x_1) & G_2^{[0]}(x_1) & \cdots & G_{N_x}^{[0]}(x_1) & x_1 & 1 \\ G_1^{[0]}(x_2) & G_2^{[0]}(x_2) & \cdots & G_{N_x}^{[0]}(x_2) & x_2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ G_1^{[0]}(x_{N_x}) & G_2^{[0]}(x_{N_x}) & \cdots & G_{N_x}^{[0]}(x_{N_x}) & x_{N_x} & 1 \end{bmatrix},$$

$$
\begin{aligned}
\widetilde{\mathbf{w}}(t) &= (w_1(t), w_2(t), \cdots, w_{N_x}(t), C_1(t), C_2(t))^T, \\
\widetilde{\mathbf{u}}(x,t) &= (u_1(x,t), u_2(x,t), \cdots, u_{N_x}(x,t))^T, \\
\frac{d^k\widetilde{\mathbf{u}}(x,t)}{dx^k} &= \left(\frac{d^k u_1(x,t)}{dx^k}, \frac{d^k u_2(x,t)}{dx^k}, \cdots, \frac{d^k u_{N_x}(x,t)}{dx^k}\right)^T,
\end{aligned}
$$

where $u_i = u(x_i, t)$ with $i = (1, 2, \cdots, N_x)$.

Owing to the presence of integration constants in the IRBF based approximants, one can beneficially introduce in the algebraic equation system additional constraints such as nodal derivative values (more details can be found in [13, 1]). Thus, the algebraic equation system (12) can be reformulated as follows.

$$
\begin{pmatrix} \widetilde{\mathbf{u}} \\ \widetilde{\mathbf{f}} \end{pmatrix} = \begin{bmatrix} \widetilde{\mathbf{G}^{[0]}} \\ \widetilde{\mathbf{L}} \end{bmatrix} \widetilde{\mathbf{w}} = \widetilde{\mathbf{C}}\widetilde{\mathbf{w}},
$$

where $\widetilde{\mathbf{f}} = \widetilde{\mathbf{L}}\widetilde{\mathbf{w}}$ are additional constraints. The conversion of the network-weight space into the physical space yields

$$
\widetilde{\mathbf{w}} = \widetilde{\mathbf{C}}^{-1} \begin{pmatrix} \widetilde{\mathbf{u}} \\ \widetilde{\mathbf{f}} \end{pmatrix}, \tag{13}
$$

$\widetilde{\mathbf{C}}^{-1}$ is the conversion matrix. By substituting (13) into (6) and (8), the second and first-order derivatives of $u(x,t)$ will be expressed in terms of nodal variable values as follows.

$$
\begin{aligned}
\tfrac{\partial^2 u(x,t)}{\partial x^2} &= \mathcal{D}_{2x}\tilde{\mathbf{u}}(x,t) + k_{2x}, \\
\tfrac{\partial u(x,t)}{\partial x} &= \mathcal{D}_{1x}\tilde{\mathbf{u}}(x,t) + k_{1x},
\end{aligned} \tag{14}
$$

where $\mathcal{D}_{1x}$ and $\mathcal{D}_{2x}$ are known vectors of length N_x; and k_{2x} and k_{1x} scalars. Applying (14) at each and every collocation point yields

$$
\begin{aligned}
\tfrac{\partial^2 \tilde{\mathbf{u}}(x,t)}{\partial x^2} &= \widetilde{\mathcal{D}}_{2x}\tilde{\mathbf{u}}(x,t) + \widetilde{k}_{2x}, \\
\tfrac{\partial \tilde{\mathbf{u}}(x,t)}{\partial x} &= \widetilde{\mathcal{D}}_{1x}\tilde{\mathbf{u}}(x,t) + \widetilde{k}_{1x},
\end{aligned} \tag{15}
$$

where $\widetilde{\mathcal{D}}_{2x}$ and $\widetilde{\mathcal{D}}_{1x}$ are known matrices of dimension $N_x \times N_x$; and $\widetilde{k}_{2x}$ and $\widetilde{k}_{1x}$ are known vectors of length N_x.

3.2. Temporal and spatial discretization of FPEs

The FPE (1), $(x,t) \in \Omega \times [0,T]$, is rewritten as follows

$$
\frac{\partial u}{\partial t} - B\frac{\partial^2 u}{\partial x^2} + \left(A - 2\frac{\partial B}{\partial x}\right)\frac{\partial u}{\partial x} + \left(\frac{\partial A}{\partial x} - \frac{\partial^2 B}{\partial x^2}\right)u = 0, \tag{16}
$$

where x, t, $u(x,t)$, $A(x)$ and $B(x)$ are defined as before.

Assume that the time interval $[0,T]$ is partitioned into n_T equal subintervals $[t_n, t_{n+1}]$ of length $\Delta t = T/n_T$ with $t_0 = 0$ and $t_{N_T} = T$. In fully discrete schemes, Eq. (16) is discretized with respect to both time and space variables. The discretization in time is accomplished by a time-stepping scheme, followed by the spatial discretization based on the 1D-IRBFN method.

Applying the θ-scheme to Eq. (16) yields

$$
\frac{u^{n+1} - u^n}{\Delta t} + \theta\mathcal{L}u^{n+1} + (1-\theta)\mathcal{L}u^n = 0, \tag{17}
$$

where $t^{n+1} = t^n + \Delta t$, $u^{n+1} = u(x, t^{n+1})$ and the operator $\mathcal{L}$ is given by

$$\mathcal{L}(.) = -B\frac{\partial^2(.)}{\partial x^2} + \left(A - 2\frac{\partial B}{\partial x}\right)\frac{\partial(.)}{\partial x} + \left(\frac{\partial A}{\partial x} - \frac{\partial^2 B}{\partial x^2}\right)(.). \tag{18}$$

For the Crank-Nicolson method ($\theta = 0.5$ and second-order accurate), Eq. (17) is expressed as

$$u^{n+1} + \alpha\mathcal{L}u^{n+1} = u^n - \alpha\mathcal{L}u^n, \tag{19}$$

where $\alpha = 0.5\Delta t$.

For the fully implicit method ($\theta = 1$), Eq. (17) is rewritten as

$$u^{n+1} + \alpha\mathcal{L}u^{n+1} = u^n, \tag{20}$$

where $\alpha = \Delta t$.

For the fully explicit method ($\theta = 0$), Eq. (17) is given by

$$u^{n+1} = u^n - \alpha\mathcal{L}u^n, \tag{21}$$

where $\alpha = \Delta t$.

Equations (19)-(21) together with the constraints (boundary or/and initial conditions) at time t^{n+1} are then spatially discretized using the 1D-IRBF approach described in section 3.1. The obtained solution is the values of the field variable at the grid points. Simulation is terminated when either the desired time (transient problem) or convergence (steady state problem) is reached. In the next section, the fully implicit method is used for time discretization of the FPEs.

4. NUMERICAL EXAMPLES

The present method is verified with some numerical experiments which have been described in [4, 16, 17]. It is worth noting that the problems are solved on a bounded interval which is uniformly discretized. The approximate solutions obtained are compared with the analytic ones using the following error norms

$$\begin{aligned} N_e &= \sqrt{\frac{\sum_{i=1}^{N}\left(u^n(x_i) - u_{ex}(x_i)\right)^2}{\sum_{i=1}^{N}(u_{ex}(x_i))^2}} \\ RMSE &= \sqrt{\frac{\sum_{i=1}^{N}\left(u^n(x_i) - u_{ex}(x_i)\right)^2}{N}} \end{aligned}$$

where $u(x_i)$, $u_{ex}(x_i)$ are the numerical and exact solutions for u respectively at a collocation point x_i and time t^n and N the total number of test points.

4.1. Example 1

Consider the FPE (1)

$$\frac{\partial u(x,t)}{\partial t} = \left[-\frac{\partial}{\partial x}A(x) + \frac{\partial^2}{\partial x^2}B(x)\right]u(x,t) \tag{22}$$

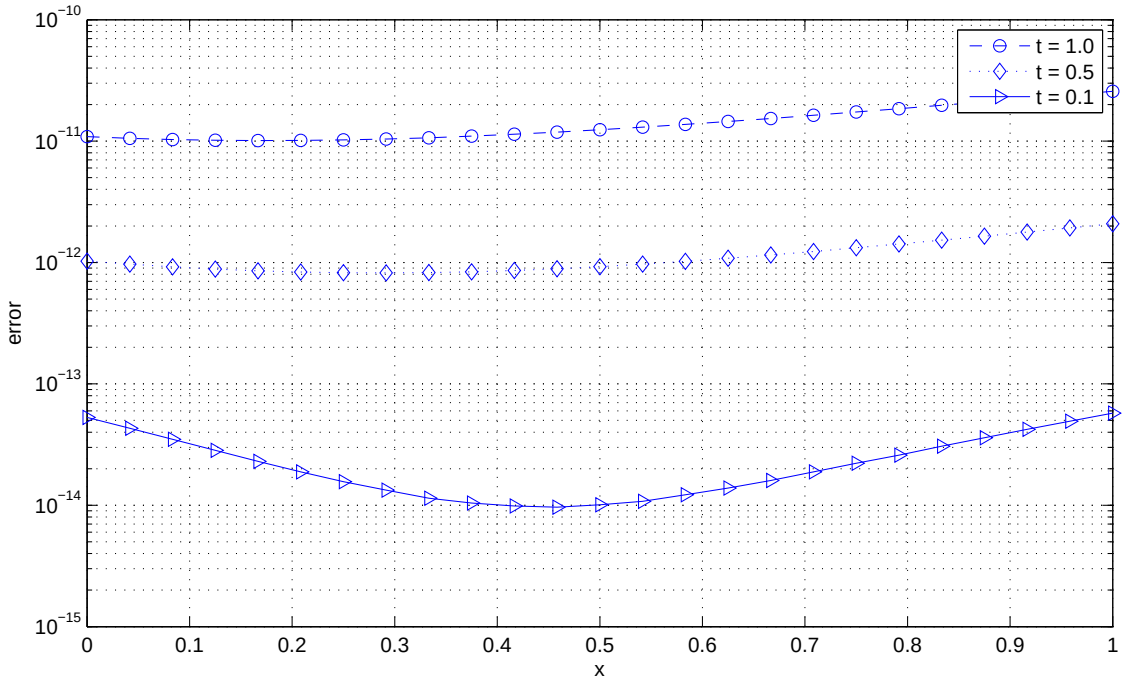

Figure 1. The absolute error of u with respect to x at times $t = 0.1$, $t = 0.5$ and $t = 1$.

with $A(x) = -1$, and $B(x) = 1$ and initial condition $u(x, 0) = x, \quad (x \in [0, 1])$. The exact solution to the problem is $u(x, t) = x + t$.

The problem is solved in the time period $[0, 1]$ with time step size $\Delta t = 0.01$. Figure 1 shows the error of u with respect to x at times $t = 0.1$, $t = 0.5$ and $t = 1$ using the present method with 25 collocation points in comparison with the exact solution.

The problem is also solved using 15 and 36 collocation points. The values of N_e and RMSE using 50 test points with 15, 25 and 36 collocation points are given in table 1. The results show that the method yields a higher degree of accuracy while using relatively coarse grids in comparison with others, including the Kansa RBF (table 2) and Hermite RBF (table 3) approaches.

Table 1. Values of N_e and $RMSE$ for numerical examples 1,2,3 for several sets of collocation points (15, 25 and 36) using the 1D-IRBF collocation method. The number of test points is $N = 50$ and $\Delta t = 0.01$.

	Example 1		Example 2		Example 3	
N	N_e	RMSE	N_e	RMSE	N_e	RMSE
15	$3.45e-6$	$1.16e-6$	$3.62e-4$	$6.74e-4$	$4.16e-4$	$7.86e-4$
25	$8.25e-7$	$2.89e-7$	$2.22e-5$	$3.54e-5$	$9.28e-6$	$1.13e-5$
36	$5.48e-9$	$1.94e-9$	$7.62e-7$	$2.37e-7$	$5.12e-7$	$6.38e-7$

4.1.1 Example 2

Consider the Fokker-Planck equation (1) with

$$A(x) = x, \quad B(x) = \frac{x^2}{2}, \tag{23}$$

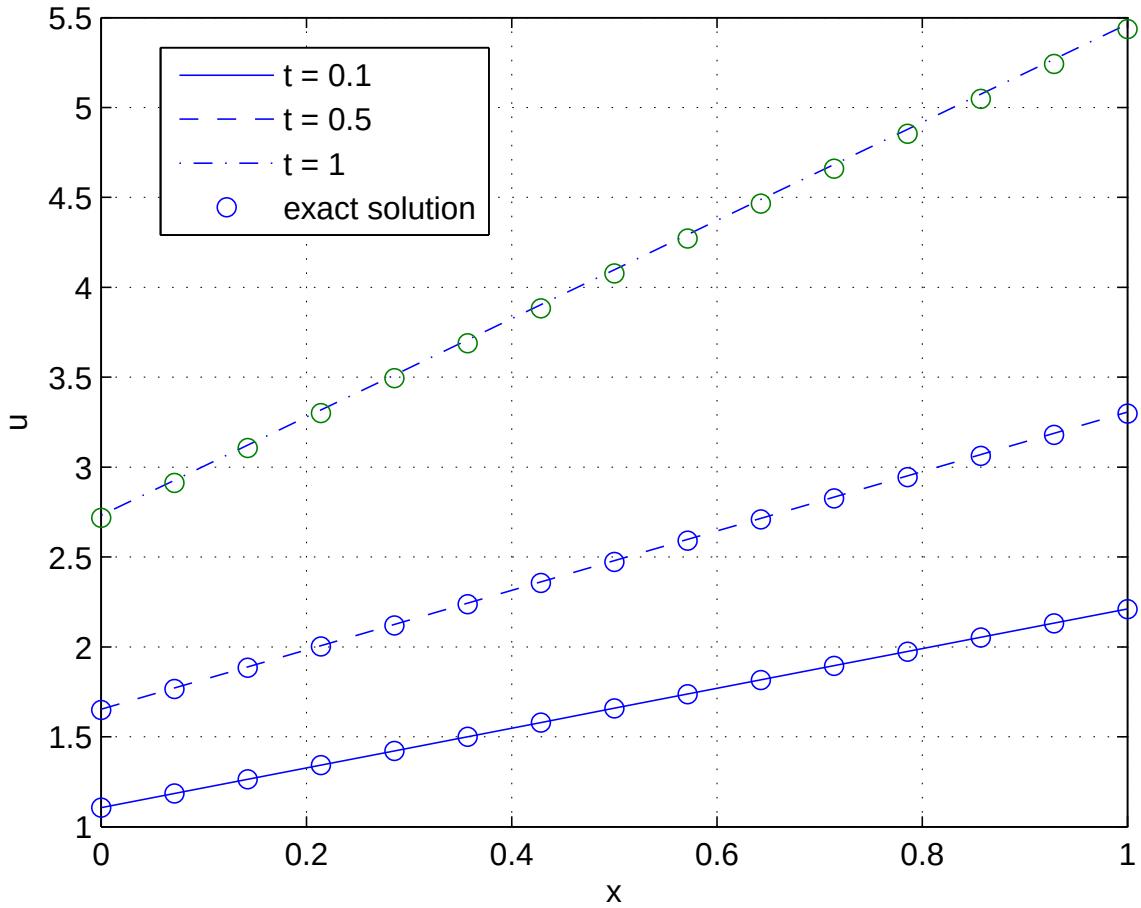

Figure 2. Exact and approximate solutions for u are plotted against x at times $t = 0.1$, $t = 0.5$ and $t = 1$.

$t \in [0, 1]$, $x \in [0, 1]$, and initial condition $u(x, 0) = f(x) = x$. The exact solution is given by $u(x, t) = x \exp(t)$.

Similarly, the problem is solved with time step size $\Delta t = 0.01$ and three sets of collocation points as in example 1. Errors are shown table 1 using 50 test points as in example 1. A comparison with the other results given in tables 2 and 3 confirms higher accuracy of the present solution.

4.1.2 Example 3

Consider the backward Kolmogorov FPE (see [2] for more details) as follows

$$\frac{\partial u(x,t)}{\partial t} = -\left[A(x,t)\frac{\partial}{\partial x} + B(x,t)\frac{\partial^2}{\partial x^2}\right] u(x,t) \tag{24}$$

where the drift and diffusion coefficients depend on both time and space as follows

$$A(x,t) = -(x+1), \quad B(x,t) = x^2 \exp(t). \tag{25}$$

With the initial condition

$$u(x,0) = f(x) = x + 1, \quad x \in [0,1], \tag{26}$$

Table 2. Values of N_e and $RMSE$ for numerical examples 1,2,3 for several sets of collocation points (25, 36, 49 and 81) using the Kansa's RBF approach, $\Delta t = 0.01$ and the number of test points $N = 50$. Data are obtained from source [16]

	Example 1		Example 2		Example 3	
N	N_e	RMSE	N_e	RMSE	N_e	RMSE
25	$2.81e-2$	$1.22e-2$	$3.37e-2$	$4.40e-2$	$2.74e-2$	$7.94e-2$
36	$1.50e-3$	$6.52e-4$	$1.91e-2$	$2.50e-2$	$3.94e-3$	$1.12e-2$
49	$1.34e-4$	$5.83e-5$	$8.68e-4$	$1.13e-3$	$7.23e-4$	$2.04e-3$
81	$9.20e-6$	$4.00e-6$	$1.94e-5$	$2.53e-5$	$2.18e-5$	$6.18e-5$

Table 3. Values of N_e and $RMSE$ for numerical examples 1,2,3 for several sets of collocation points (25, 36, 49 and 81) using the Hermite RBF without time discretization scheme. The number of test points is $N = 50$. Data are obtained from source [16].

	Example 1		Example 2		Example 3	
N	N_e	RMSE	N_e	RMSE	N_e	RMSE
25	$1.48e-4$	$6.41e-5$	$2.12e-4$	$1.21e-4$	$1.73e-4$	$2.23e-4$
36	$2.30e-5$	$9.92e-6$	$7.11e-5$	$5.43e-5$	$1.13e-4$	$1.61e-4$
49	$2.81e-6$	$1.21e-6$	$8.63e-6$	$6.86e-6$	$8.29e-5$	$8.61e-5$
81	$3.83e-8$	$1.65e-8$	$4.93e-8$	$3.12e-8$	$1.35e-7$	$1.76e-7$

the exact solutionto the problem is $u(x,t) = (x+1)\exp(t)$.

Figure 2 shows the approximate solution with respect to x using a very coarse grid of 15 collocation points at times $t = 0.1$, $t = 0.5$ and $t = 1$. The problem is also solved with 25 and 36 collocation points. Similar comments as in examples 1 and 2 can be made here regarding results in tables 1, 2 and 3.

In general, although the accuracy of solutions tends to decrease with time, the results claimed that the present method can reach high order accuracy using coarse grids.

5. CONCLUSION

The 1D-IRBF based meshfree method has been successfully developed for the computation of FPEs. The advantages of the present approach include (i) to yield a meshless discretisation of FPEs; (ii) to improve the approximation accuracy by avoiding the reduction in convergence rate caused by differentiation; (ii) to reduce the noise in the approximation via the use of integration as a smoothing operator to construct the approximants. The present method is demonstrated with several forms of FPEs. The obtained results show that the present method yields a high degree of accuracy with relatively coarse grids.

Acknowledgements

This work was supported by the Australian Research Council.

6. REFERENCES

[1] Mai-Duy N., Tran-Cong T., "Numerical solution of differential equations using multi-quadric radial basis function networks". *Neural Networks* 14, 185-199, 2001.

[2] Risken, H., "The Fokker-Planck equation: method of solution and applications", 1989, Springer Verlag Berlin.

[3] Zorzano M. P., Mais H., Vazquez L., "Numerical solution of two-dimensional Fokker-Planck equations". *Appl. Math. Comput.* 98, 109-17, 1999.

[4] Dehghan M., Tatari M., "The use of He's variational iteration method for solving a Fokker-Planck equation". *Phys. Scr.* 74, 310-6, 2006.

[5] Odibat Z., Momani S., "Numerical solution of Fokker-Planck equation with space and time fractional derivatives". *Phys. Lett. A* 369, 349358, 2007.

[6] Harrison G. Numer. Meth. "Numerical solution of the Fokker-Planck equation using moving finite elements". *Part. Diff. Eqs.* 4, 219-32, 1988.

[7] Jafari M. A., Aminataei A. "Application of homotopy perturbation method in the solution of Fokker-Planck equation". *Phys. Scr.* 80, 055001, 2009.

[8] Fasshauer G. E., Meshfree approximation methods with Matlab (Interdisciplinary Mathematical Sciences Vol. 6, World Scientific, Singapore, 2007.

[9] Kansa E., "Multiquadrics a scattered data approximation scheme with applications to computational fluid dynamics-I: Surface approximations and partial derivatives estimates". *Computers & Mathematics with Applications* 19, 127145, 1990.

[10] Kansa E. J., "A scattered data approximation scheme with applications to computational fluid-dynamics. II. Solutions to parabolic, hyperbolic and elliptic partial differential equations'. *Comput. Math. Appl.* 19, 147-161, 1990.

[11] Sarler B., Vertnik R., "Meshfree explicit local radial basis function collocation method for diffusion problems". *Comput. Math. Appl.* 51, 1269-1282, 2006.

[12] Divo E., Kassab A., "Iterative domain decomposition mesh-less method modeling of incompressible viscous flows and conjugate heat transfer". *Eng. Anal. Boundary Elements* 30, 465-478, 2006.

[13] Mai-Duy N., Tran-Cong T., "A Control Volume Technique Based on Integrated RBFNs for the Convection-Diffusion Equation". *Numerical methods for partial differential equations* 26, 426447, 2010.

[14] Franke R., "Scattered data interpolation: tests of some methods". *Mathematics of Computation* 38, 181200, 1982.

[15] Haykin S., *Neural networks: A comprehensive foundation*. New Jersey:Prentice Hall, 1999.

[16] Kazem S., Rad J. A., Parand K., "Radial basis functions methods for solving Fokker-Planck equation". *Engineering Analysis with Boundary Elements* 36, 181-189, 2012.

[17] Tatari M., Dehghan M., Razzaghi M., "Application of the Adomian decomposition method for the Fokker-Planck equation". *Math Comput Model* 45, 639-50, 2007.