Journal of King Saud University – Science (2015) 27, 292–301

King Saud University
Journal of King Saud University – Science
www.ksu.edu.sa
www.sciencedirect.com

ORIGINAL ARTICLE

MLPG method based on moving kriging interpolation for solving convection–diffusion equations with integral condition

S. Khankham [a,b], A. Luadsong [a,*], N. Aschariyaphotha [c]

[a] Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-utid Road, Bangmod, Toongkru, Bangkok 10140, Thailand
[b] Science Achievement Scholarship of Thailand in Mathematics, Faculty of Science at Mahidol University, 272 Rama VI Road, Ratchathewi, Bangkok 10400, Thailand
[c] Ratchaburi Learning Park, King Mongkut's University of Technology Thonburi (KMUTT), Rang Bua, Chom Bueng, Ratchaburi 70150, Thailand

Received 16 June 2014; accepted 2 March 2015
Available online 9 March 2015

KEYWORDS
Convection–diffusion equations;
Crank–Nicolson;
Moving kriging interpolation;
MLPG;
Integral condition

Abstract A formulation of the meshless local Petrov–Galerkin (MLPG) method based on the moving kriging interpolation (MK) is presented in this paper. The method is used for solving time-dependent convection–diffusion equations in two-dimensional spaces with the Dirichlet, Neumann, and non-local boundary conditions on a square domain. The method is developed based on the moving kriging interpolation method for constructing shape functions which have the Kronecker delta property. In the method, the test function in each sub-domain is chosen as the indicator function. The Crank–Nicolson method is chosen for temporal discretization. Two test problems are presented which demonstrate the easiness and accuracy of this method as shown by the relative error.

1. Introduction

In recent decades, meshless method in mathematical modeling has attracted much attention, due to their flexibility in solving practical science and engineering problems. The meshless method has been developed and achieved remarkable progress in mathematical modeling and related fields. The main reason for interest in meshless method is it can save computational time because no mesh is needed. They can easily produce high accuracy, since in areas where more refinement is required; nodes can quite easily be added. In addition, they can easily construct high-order shape functions. They can easily solve large deformation and strong nonlinear problem. The connectivity among the nodes is generated as a part of the computation and can change with time.

* Corresponding author.
E-mail address: anirut.lua@kmutt.ac.th (A. Luadsong).
Peer review under responsibility of King Saud University.

ELSEVIER Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksus.2015.03.001
1018-3647

Based on the form of system equation, the meshfree methods are classified into three categories. The first category covers the meshfree methods based on strong forms of system equations, in which discretization is performed directly from the governing differential equations, such as the general finite difference method (Liszka and Orkisz, 1980), the smooth particle hydrodynamic (SPH) method (Liu and Liu, 2003; Lucy, 1977), and other meshfree collocation methods. The second category includes meshfree methods based on weak forms of system equations, such as the element-free Galerkin (EFG) method (Belytschko et al., 1994), the meshless local Petrov–Galerkin (MLPG) method (Atluri and Zhu, 1998; Atluri and Lin, 2000; Abbasbandy and Shirzadi, 2011; Mirzaei, 2010; Mahmoodabadi et al., 2011), the point interpolation method (PIM) (Liu, 2002; Liu and Gu, 2001), etc. The third category concerns meshfree methods based on the combination of weak and strong forms, such as the meshfree weak–strong-form (MWS) method. The major difference in these meshless methods is in the choice of interpolation techniques.

The MLPG method was first discovered by Atluri and Zhu (1998). Meshless local Petrov–Galerkin (MLPG) method is a truly meshless method. This method is based on a weak form computed over a local sub-domain and the moving least squares (MLS) approximation. The MLPG method is one of the most viable methods in which the MLS approach is used to construct the shape functions. Although the MLPG method has been applied to many problems, there exists an inconvenience or disadvantage when using the MLPG because of the difficulty in implementing essential boundary conditions. This is because the MLS shape functions lack the Kronecker delta property. Therefore the MK interpolation method (Yimnak and Luadsong, 2014; Kaewumpai and Luadsong, 2014) has been proposed to overcome this problem. It uses the nodal values in the local support domain to construct shape functions with the Kronecker delta property. The MK interpolation method works well for practical problems.

Finding numerical solutions with non-local boundary conditions is important in applications such as chemical diffusion, heat conduction processes, population dynamics thermoelasticity, medical science, electrochemistry and control theory. Because of the complex problems, some analytic solutions are difficult to find in the sense of boundary conditions and geometrical shapes. So, recently much attention has been paid in the literature to the analysis and implementation of accurate methods for the numerical solution of time-dependent partial differential equations with non-local boundary conditions. Abbasbandy and Shirzadi (2010) researched on the MPLG method for the two-dimensional diffusion equation with the Neumann boundary condition and non-classical boundary condition, and a meshless method for the two-dimensional diffusion equation with an integral condition. Mohyud-Din and Yildirim (2010) employed the homotopy analysis method (HAM) for the solutions of two-dimensional diffusion equations subject to non-standard boundary specifications. The HAM method does not need transformation techniques, linearization and discretization. The approximate solutions by the HAM method were compared with exact solutions. It was shown that the method is reliable, efficient and requires less computation. Sajavicius (2013) researched optimization, conditioning and accuracy of radial basis function method for partial differential equations with nonlocal boundary conditions in case of two-dimensional Poisson equation. Techapirom and Luadsong (2013) have used an MLPG method to study the two-dimensional heat equation with Dirichlet, Neumann and non-local boundary conditions in a square domain. Their study demonstrated the good accuracy of the proposed method and indicates that it can be easily extrapolated to other problems. Sataprahm and Luadsong (2014) have used the Meshless Local Petrov–Galerkin (MLPG) method for the incompressible Navier–Stokes equations. The numerical examples presented the local symmetric weak form (LSWF) and the local unsymmetric weak form (LUSWF) with a classical Gaussian weight and an improved Gaussian weight on both regular and irregular nodes is demonstrated. It is found that LSWF1 with a classical Gaussian weight order 2 gives the most accurate result. Goodrich (2015) considers the coupled systems of boundary value problems with nonlocal boundary conditions. The result shows that this system can possess at least one positive solution even if no growth conditions. Islam et al. (2015) present two numerical methods that are analyzed for the solution of two-dimensional Poisson equation with two different types of nonlocal boundary conditions. The first numerical method is a collocation method based on Haar wavelet whereas the second numerical method is a meshless method based on different types of radial basis functions (RBFs). A two-point boundary condition and an integral boundary condition are the two types of nonlocal boundary conditions considered in the present work. Two numerical results have shown that the accuracy and efficiency wise performance is confirmed through application of the algorithms on the benchmark tests.

The purpose of this article is to present a very efficient MLPG based on MK interpolation for solving the two-dimensional time-dependent convection–diffusion equation with integral conditions. To deal with time derivatives, using the Crank–Nicolson implicit method. The solution is approximated by the MK interpolation method which holds the Kronecker delta property, thereby enhancing the nodal shape construction accuracy. It was shown that the method is reliable and efficient. Two test problems are presented illustrating the easiness and accuracy of this method as shown by the relative error.

2. Moving kriging interpolation method

The kriging interpolation is well-known geostatic technique for spatial interpolation in geology and mining (Lucy, 1977). The formulation of the construction of meshless shape function by MK interpolation is introduced briefly in the following. Similar to the MLS approximation, consider the function $T(\mathbf{x})$ defined in the domain Ω discretized by a set of properly scattered nodes $\mathbf{x}_i, i = 1, 2, \ldots, N$, where N is the total number of nodes in the whole domain. It is assumed that only N nodes surrounding point $\mathbf{x}$ have the effect on $T(\mathbf{x})$. The sub-domain Ω_x that encompasses these surrounding nodes is called the interpolation domain of point $\mathbf{x}$. The MK interpolation $T^h(\mathbf{x})$ at point $\mathbf{x}$ is defined as presented in Liu and Gu (2001). Therefore, the formulation of the meshless shape function using MK interpolation is given by:

$$T^h(\mathbf{x}) = \sum_{i=1}^{N} \phi_i(\mathbf{x}) T_i = \mathbf{\Phi}(\mathbf{x})\mathrm{T}, \quad \mathbf{x} \in \Omega_x \tag{1}$$

where $\mathrm{T} = [T(\mathbf{x}_1)T(\mathbf{x}_2)\dots T(\mathbf{x}_N)]^{\mathrm{T}}$ is a vector value of the function in the domain Ω.$\mathbf{\Phi}(\mathbf{x})$ is a $1 \times N$ vector of shape functions, expressed as:

$$\mathbf{\Phi}(\mathbf{x}) = p^{\mathrm{T}}(\mathbf{x})\mathbf{A} + r^{\mathrm{T}}(\mathbf{x})\mathbf{B}, \tag{2}$$

where matrices $\mathbf{A}$ and $\mathbf{B}$ are defined as:

$$\mathbf{A} = (\mathbf{P}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{P})^{-1}\mathbf{P}^{\mathrm{T}}\mathbf{R}^{-1} \tag{3}$$

$$\mathbf{B} = \mathbf{R}^{-1}(\mathbf{I} - \mathbf{P}\mathbf{A}) \tag{4}$$

In which $\mathbf{I}$ is a unit matrix of size $N \times N$, and vector $p(\mathbf{x})$ is:

$$p^{\mathrm{T}}(\mathbf{x}) = [p_1(\mathbf{x})p_2(\mathbf{x})\dots p_N(\mathbf{x})]. \tag{5}$$

For matrix $\mathbf{P}$ with the size $N \times M$, values of the polynomial basis functions at the given set of nodes are collected:

$$\mathbf{P} = \begin{bmatrix} p_1(\mathbf{x}_1) & \cdots & p_M(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ p_1(\mathbf{x}_N) & \cdots & p_M(\mathbf{x}_N) \end{bmatrix}. \tag{6}$$

Matrices $\mathbf{R}$ and vector $r(\mathbf{x})$ are defined by the following equations:

$$\mathbf{R} = \begin{bmatrix} r(\mathbf{x}_1, \mathbf{x}_1) & \cdots & r(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ r(\mathbf{x}_N, \mathbf{x}_1) & \cdots & r(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix} \tag{7}$$

$$r^{\mathrm{T}}(\mathbf{x}) = [r(\mathbf{x}, \mathbf{x}_1)r(\mathbf{x}, \mathbf{x}_2)\dots r(\mathbf{x}, \mathbf{x}_N)], \tag{8}$$

where $r(\mathbf{x}_i, \mathbf{x}_j)$ is the correlation function between any pair of nodes located at X_i and X_j representing the covariance of the field value $T(\mathbf{x})$, i.e.

$$r(\mathbf{x}_i, \mathbf{x}_j) = E[T(\mathbf{x}_i)T(\mathbf{x}_j)]. \tag{9}$$

Similarly, the covariance $E[T(\mathbf{x}_i)T(\mathbf{x}_j)]$ can be replaced by $r(\mathbf{x}_i, \mathbf{x}_j)$. It can be seen from the foregoing formulations that the values of matrices $\mathbf{R}$ and $r(\mathbf{x})$ play important roles in the computation. A simple and frequently- used correlation function is a Gaussian function:

$$r(\mathbf{x}_i, \mathbf{x}_j) = e^{-\alpha_c\left(\frac{r_{ij}}{d_c}\right)^2} \tag{10}$$

where $r_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$, d_c and $\alpha_c > 0$ are the correlation parameters used to fit the model and are assumed to be given.

The first-order partial derivatives of the shape function $\mathbf{\Phi}(\mathbf{x})$ against the coordinates $\mathbf{x}_i, i = 1, 2$ can be easily obtained from Eq. (2)

$$\mathbf{\Phi}_{,i}(\mathbf{x}) = p^{\mathrm{T}}_{,i}(\mathbf{x})\mathbf{A} + r^{\mathrm{T}}_{,i}(\mathbf{x})\mathbf{B}, \tag{11}$$

where $(\cdot)_i$ denotes $\partial(\cdot)/\partial x^i$.

3. Formulation of the moving kriging interpolation for convection–diffusion equation

3.1. Governing equation and boundary conditions

The two-dimensional time-dependent convection–diffusion equation subject to non-local boundary conditions can be written as:

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T - \nabla \cdot (\mathbf{K}\nabla T) = f, \quad \mathbf{x} \in \Omega, \ t > 0 \tag{12}$$

where T is the unknown function, $\mathbf{K}$ is the diffusivity coefficient, $\mathbf{u}$ is the flow velocity, f is a source term. The boundary conditions are assumed to be:

– The non-local boundary condition:

$$T = h_0(\mathbf{x})\mu(t), \qquad \mathbf{x} \in \mathbf{\Gamma}_n. \tag{13}$$

– The essential boundary condition:

$$T = h_1(\mathbf{x}, t), \qquad \mathbf{x} \in \mathbf{\Gamma}_d. \tag{14}$$

– The natural boundary condition:

$$\mathbf{n} \cdot \nabla T = g(\mathbf{x}, t), \qquad \mathbf{x} \in \mathbf{\Gamma}_q. \tag{15}$$

– The initial condition assumed to be

$$T(\mathbf{x}, 0) = T_0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{16}$$

and the integral condition

$$\int_\Omega T(\mathbf{x}, t)\, d\mathbf{\Omega} = m(t), \tag{17}$$

where $\mu(t)$ is the unknown function, g, h_0, h_1 and m are given functions. The non-local boundary condition is variable-separable, with spatial dependence given by $h_0(\mathbf{x})$ and time dependence given by $\mu(t)$. $\mathbf{n}$ is the outward unit normal vector to $\partial\Omega$, where $\partial\Omega$ is the boundary of the domain. $\mathbf{\Gamma}_n, \mathbf{\Gamma}_d$ and $\mathbf{\Gamma}_q$ are the part of the boundary $\partial\Omega$, satisfying $\mathbf{\Gamma}_n \cap \mathbf{\Gamma}_d \cap \mathbf{\Gamma}_q = \phi$ and $\mathbf{\Gamma}_n \cup \mathbf{\Gamma}_d \cup \mathbf{\Gamma}_q = \partial\Omega$.

3.2. Local weak-forms

In implementation of the present MK interpolation method, a weak form is first constructed over a local sub-domain Ω_s bounded by $\partial\Omega_s$, where Ω_s is located entirely inside the global domain Ω. Using the local weighted residual technique, the local weak form of Eq. (12) over a local sub-domain Ω_s can be written as:

$$\int_{\Omega_s^i} \left[\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T - \nabla \cdot (\mathbf{K}\nabla T)\right] v_i d\mathbf{\Omega} = \int_{\Omega_s^i} f v_i d\mathbf{\Omega} \tag{18}$$

where T is the trial function and v is the test function.

Using $\nabla \cdot (\mathbf{K}\nabla T)v_i = \nabla \cdot (\mathbf{K}\nabla(T)v_i) - \mathbf{K}\nabla T \cdot \nabla v_i$ and the divergence theorem, we obtain the following local weak formulation (LWF):

$$\int_{\Omega_s^i} \frac{\partial T}{\partial t} v_i d\mathbf{\Omega} + \int_{\Omega_s^i} \mathbf{u} \cdot \nabla T v_i d\mathbf{\Omega} - \int_{\partial\Omega_s^i} \mathbf{K} v_i \frac{\partial T}{\partial \mathbf{n}} d\mathbf{\Gamma} + \int_{\Omega_s^i} \mathbf{K}\nabla T \cdot \nabla v_i d\mathbf{\Omega} = \int_{\Omega_s^i} f v_i d\mathbf{\Omega}. \tag{19}$$

In general, the boundary $\partial\Omega_s^i$ of the local sub-domain Ω_s^i may intersect with the boundary of the global domain Ω. Therefore, $\partial\Omega_s^i = \mathbf{\Gamma}_{sn}^i \cup \mathbf{\Gamma}_{sd}^i \cup \mathbf{\Gamma}_{sq}^i \cup L_s^i$ where, L_s^i is a part of $\partial\Omega_s^i$. By imposing the natural boundary condition, we obtain:

$$\int_{\Omega_s^i} \frac{\partial T}{\partial t} v_i d\mathbf{\Omega} + \int_{\Omega_s^i} \mathbf{u} \cdot \nabla T v_i d\mathbf{\Omega} - \int_{\Gamma_{sn}^i} \mathbf{K} v_i \frac{\partial T}{\partial \mathbf{n}} d\mathbf{\Gamma} - \int_{\Gamma_{sd}^i} \mathbf{K} v_i \frac{\partial T}{\partial \mathbf{n}} d\mathbf{\Gamma} - \int_{L_s^i} \mathbf{K} v_i \frac{\partial T}{\partial \mathbf{n}} d\mathbf{\Gamma} + \int_{\Omega_s^i} \mathbf{K}\nabla T \cdot \nabla v_i d\mathbf{\Omega} = \int_{\Omega_s^i} f v_i d\mathbf{\Omega} + \int_{\Gamma_{sq}^i} g v_i d\mathbf{\Gamma}. \tag{20}$$

The test function in each sub-domain is chosen as the indicator function.

$$v_i(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \Omega_s^i \\ 0, & \mathbf{x} \notin \Omega_s^i \end{cases} \tag{21}$$

so, $\nabla v(\mathbf{x}) = 0, \forall \mathbf{x} \in \Omega_s^i$ and the local weak form Eq. (22) is transformed into the following simple local integral equation:

$$\int_{\Omega_s^i} \frac{\partial T}{\partial t} d\mathbf{\Omega} + \int_{\Omega_s^i} \mathbf{u} \cdot \nabla T d\mathbf{\Omega} - \int_{\Gamma_{sn}^i} \mathbf{K} \frac{\partial T}{\partial \mathbf{n}} d\mathbf{\Gamma} - \int_{\Gamma_{sd}^i} \mathbf{K} \frac{\partial T}{\partial \mathbf{n}} d\mathbf{\Gamma} - \int_{L_s^i} \mathbf{K} \frac{\partial T}{\partial \mathbf{n}} d\mathbf{\Gamma} = \int_{\Omega_s^i} f d\mathbf{\Omega}. + \int_{\Gamma_{sq}^i} g \, d\mathbf{\Gamma}. \tag{22}$$

3.3. Discretization of the weak form

As a known test function is used in the LWF, the use of the LWF for one point (and hence for one local domain) will yield only one linear equation involving $\hat{T}(t)$. Note that the trial function T within the sub-domain Ω_s, in the MK approximation, is determined by the fictitious nodal values $\hat{T}_i(t)$, within the domain of definition for all points x falling within Ω_s. The LWF in Eq. (22) gives one algebraic equation relating all these $\hat{T}_i(t)$. Thus, we obtain as many equations as the number of nodes. Therefore, we need as many local domains Ω_s as the number of nodes in the global domain to obtain as many equations as the number of unknowns. In the present implementation, the local domain is chosen as a circle, centered at a node $\mathbf{x}_i$.

To obtain the discrete equations from the LWF in Eq. (22), the MK approximation in Eq. (1), is alternatively used to approximate the trial function T. Substituting of Eq. (1) into the LWF Eq. (22) for the entire node, leads to the following discretized system of linear equations:

$$\sum_{j=1}^{N} \left(\int_{\Omega_s^i} \phi_j(\mathbf{x}) d\mathbf{\Omega} \right) \hat{T}_j'(t) - \sum_{j=1}^{N} \left[\left(\int_{\Omega_s^i} \mathbf{u} \cdot \nabla \phi_j(\mathbf{x}) d\mathbf{\Omega} - \int_{L_s^i} \mathbf{K} \phi_{j,\mathbf{n}}(\mathbf{x}) d\mathbf{\Gamma} - \int_{\Gamma_{sn}^i} \mathbf{K} \phi_{j,\mathbf{n}}(\mathbf{x}) d\mathbf{\Gamma} - \int_{\Gamma_{sd}^i} \mathbf{K} \phi_{j,\mathbf{n}}(\mathbf{x}) d\mathbf{\Gamma} \right) \hat{T}_j(t) \right] = \int_{\Omega_s^i} f d\mathbf{\Omega} + \int_{\Gamma_{sq}^i} g \, d\mathbf{\Gamma}. \tag{23}$$

where $i = 1, 2, 3, \ldots, N$.

Eq. (23) can be written in the matrix form as follows:

$$\mathbf{C} \frac{\partial \mathbf{T}}{\partial t} + \mathbf{D}\mathbf{T} = \mathbf{F}(t), \tag{24}$$

where $\mathbf{C}$, $\mathbf{D}$ and $\mathbf{F}$ are matrices described as follows:

$$\mathbf{C} = [C_{ij}], \quad C_{ij} = \int_{\Omega_s^i} \phi_j(\mathbf{x}) d\mathbf{\Omega} \tag{25}$$

$$\mathbf{D} = [D_{ij}], \quad D_{ij} = \int_{\Omega_s^i} \mathbf{u} \cdot \nabla \phi_j(\mathbf{x}) d\mathbf{\Omega} - \int_{L_s^i} \mathbf{K} \phi_{j,\mathbf{n}}(\mathbf{x}) d\mathbf{\Gamma} - \int_{\Gamma_{sn}^i} \mathbf{K} \phi_{j,\mathbf{n}}(\mathbf{x}) d\mathbf{\Gamma} - \int_{\Gamma_{sd}^i} \mathbf{K} \phi_{j,\mathbf{n}}(\mathbf{x}) d\mathbf{\Gamma} \tag{26}$$

$$\mathbf{F} = [F_i], \quad F_i = \int_{\Omega_s^i} f \, d\mathbf{\Omega} + \int_{\Gamma_{sq}^i} g \, d\mathbf{\Gamma}. \tag{27}$$

Setting a time-stepping scheme to overcome the time derivative and applying the Crank–Nicolson technique of approximation to Eq. (24) yields:

$$(2\mathbf{C} + \Delta t \, \mathbf{D})\mathbf{T}^{k+1} = (2\mathbf{C} + \Delta t \, \mathbf{D})\mathbf{T}^{k} + 2\Delta t \, \mathbf{F}^{\frac{1}{2}} \tag{28}$$

where $\mathbf{T}$ is a matrix described as follows:

$$\mathbf{T} = \begin{pmatrix} \hat{\mathbf{T}} \\ \hat{\mu} \end{pmatrix} \tag{29}$$

Assuming that $\hat{T}_i^k$, for $i = 1, 2, ..., N$ and $\hat{\mu}^k$, are known, our aim is to compute $\hat{T}_i^{k+1}$, for $i = 1, 2, \ldots, N$ and $\hat{\mu}^{k+1}$. Now, we have $N + 1$ unknowns, so to compute these unknowns we need one equation, which can be obtained from the non-local boundary condition from Eq. (17). Substituting trial function Eq. (1), we obtain:

$$\int_{\Omega} (T^h)^{k+1}(\mathbf{x}) \mathbf{d\Omega} = \int_{\Omega} \sum_{j=1}^{N} \phi_j(\mathbf{x}) \hat{T}_j^{k+1} \mathbf{d\Omega} = \sum_{j=1}^{N} \left[\int_{\Omega} \phi_j(\mathbf{x}) \mathbf{d\Omega} \right] \hat{T}_j^{k+1} = m^{k+1} \tag{30}$$

which can be written in a matrix form as:

$$\mathbf{S}\hat{\mathbf{T}}^{k+1} = m^{k+1}, \tag{31}$$

where $\mathbf{S}$ is a matrix described as follows:

$$\mathbf{S} = [S_j], \quad S_j = \int_{\Omega} \phi_j(\mathbf{x}) \mathbf{d\Omega}. \tag{32}$$

For nodes on the natural, essential and non-local boundaries, we present an algorithm to impose boundary conditions as follows:

– For node $\mathbf{x}_i$ is on the natural boundary condition, $\mathbf{x}_i \in \mathbf{\Gamma}_q$

$$\sum_{j=1}^{N} \left[\int_{\Gamma_{sq}^i} \phi_{j,\mathbf{n}}(\mathbf{x}) \mathbf{d\Gamma} \right] \hat{T}_j^{k+1} = \int_{\Gamma_{sq}^i} g(\mathbf{x}^l, (k+1)\Delta t) \mathbf{d\Gamma} \tag{33}$$

– For node $\mathbf{x}_i$ is on the essential boundary condition, $\mathbf{x}_i \in \mathbf{\Gamma}_d$

$$\sum_{j=1}^{N} \left[\int_{\Gamma_{sd}^i} \phi_j(\mathbf{x}) \mathbf{d\Gamma} \right] \hat{T}_j^{k+1} = \int_{\Gamma_{sd}^i} h_1(\mathbf{x}^l, (k+1)\Delta t) \mathbf{d\Gamma} \tag{34}$$

Table 1 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10, $N = 25$, 36 and 81.

N/m	3	6	10
25	2.0163e−2	3.1541e-3	2.5064e-3
36	1.7916e-2	2.6715e-3	1.9803e-3
81	6.2939e-3	2.5497e-3	1.5738e-3

Table 2 The relative error for $\mu(t)$ at $M = 3$, 6 and 10, $N = 25$, 36 and 81.

N/m	3	6	10
25	8.8847e-3	1.0669e-3	1.0410e-3
36	7.5965e-3	7.2043e-4	6.6892e-4
81	2.5943e-3	6.5504e-4	5.9344e-4

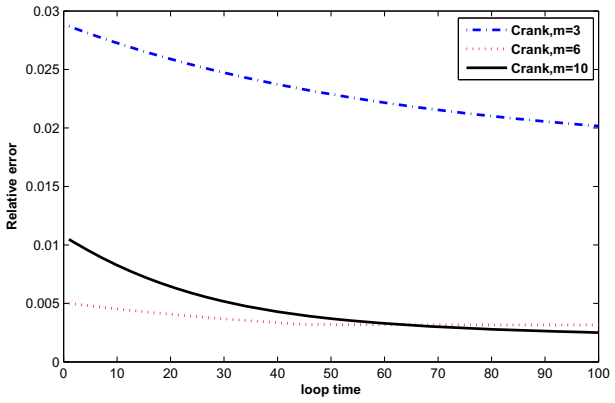

Figure 1 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10 with $N = 25$.

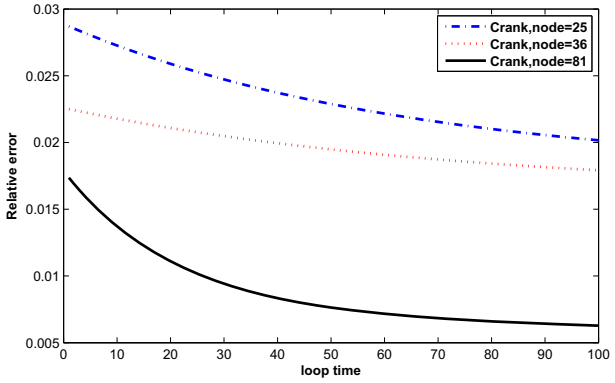

Figure 2 The relative error for $T(\mathbf{x}, t)$ at $N = 25$, 36 and 81 with $M = 3$.

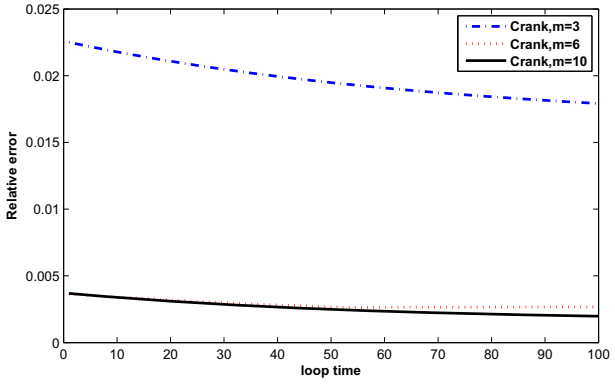

Figure 3 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10 with $N = 36$.

– For node $\mathbf{x}_i$ is on the non-local boundary condition, $\mathbf{x}_i \in \mathbf{\Gamma}_n$

$$\sum_{j=1}^{N}\left[\int_{\Gamma_{sn}^{i}} \phi_j(\mathbf{x})\mathbf{d\Gamma}\right]\hat{T}_j^{k+1} - \int_{\Gamma_{sn}^{i}} h_0(\mathbf{x}^l)\mathbf{d\Gamma}\hat{\mu}^{k+1} = 0. \tag{35}$$

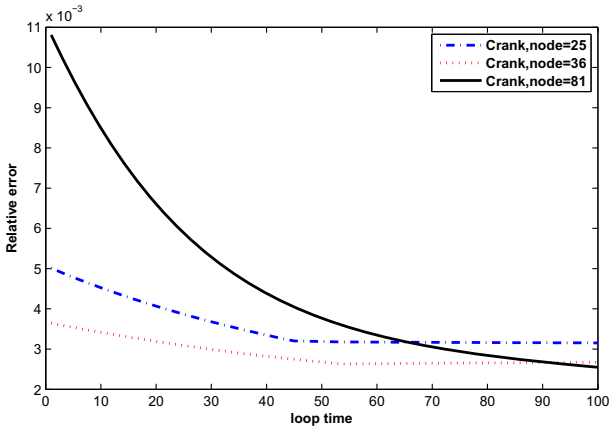

Figure 4 The relative error for $T(\mathbf{x}, t)$ at $N = 25$, 36 and 81 with $M = 6$.

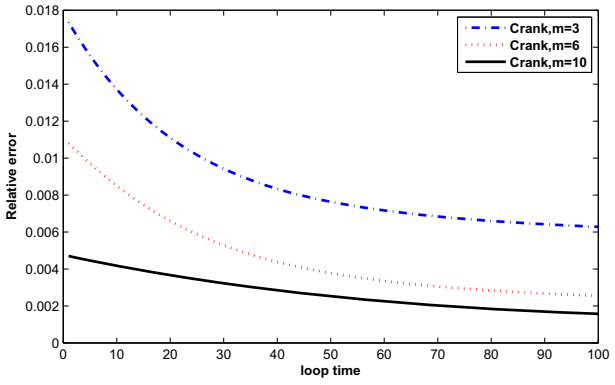

Figure 5 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10 with $N = 81$.

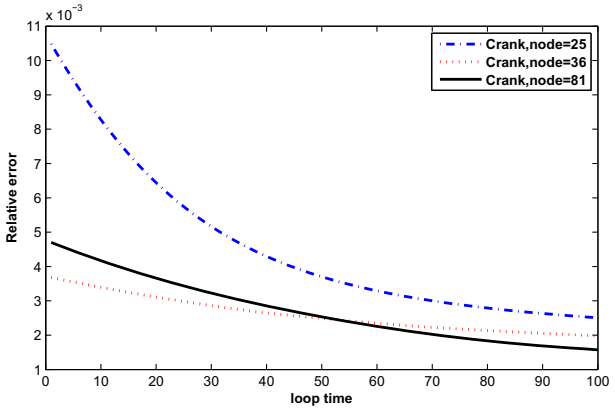

Figure 6 The relative error for $T(\mathbf{x}, t)$ at $N = 25$, 36 and 81 with $M = 10$.

4. Numerical examples

In the computation, problems are considered in a unit square domain given by $0 \leqslant \mathbf{x} \leqslant 1$.. The flow is unidirectional and constant with velocity components $u_1 = 1$ and $u_2 = 1, \mathbf{u} = (u_1, u_2)^{\mathrm{T}}$.

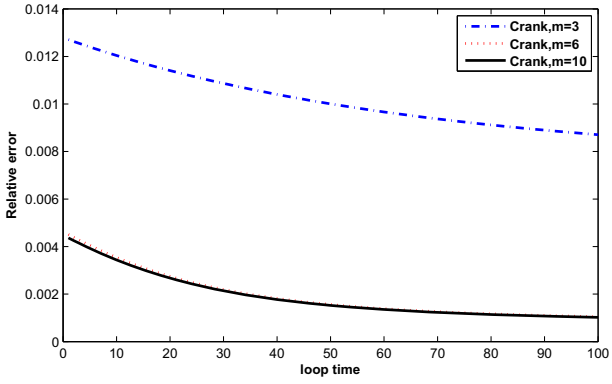

Figure 7 The relative error for $\mu(t)$ at $M = 3$, 6 and 10 with $N = 25$.

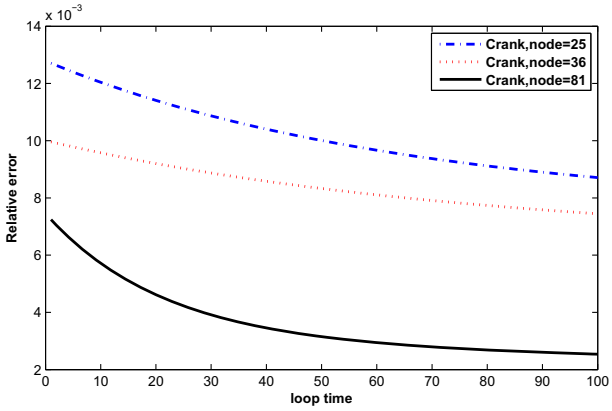

Figure 8 The relative error for $\mu(t)$ at $N = 25$, 36 and 81 with $M = 3$.

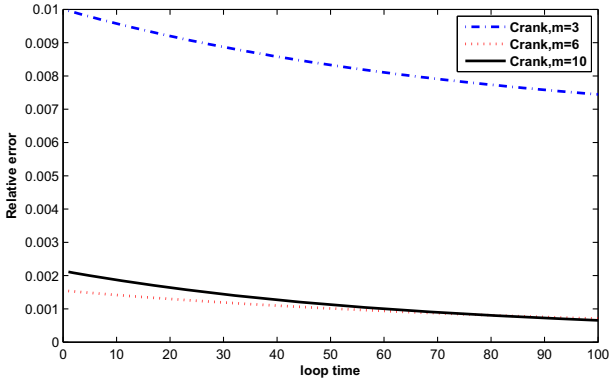

Figure 9 The relative error for $\mu(t)$ at $M = 3$, 6 and 10 with $N = 36$.

The diffusivity coefficient is 1. The radius of the local subdomain for each node is chosen as $r_0 = 0.9\,h$, with h is the distance between two consecutive nodes in each direction. The results are obtained for T and μ for several time levels with $\Delta t = 0.001$. The numbers of complete monomial basis (M) are 3, 6 and 10 and the numbers of nodal points (N) are 25, 36 and 81. The exact solutions are known for these problems and are used to test the accuracy of these numerical schemes. The numerical results in these tables show that the method converges by increasing the number of nodal points, and increasing the number of monomial basis for a fixed number of nodal points enhance accuracy.

4.1. Problem 1

We consider the two-dimensional time-dependent convection–diffusion equation with Dirichlet and non-local boundary condition:

$$\frac{\partial T}{\partial t} + \mathbf{u}\cdot\nabla T - \nabla\cdot(\mathbf{K}\nabla T) = f,$$
$$(\mathbf{x},t) \in [0,1]\times[0,1]\times[0,T_{\max}], \quad \mathrm{x} = (x,y)$$

subject to initial, Dirichlet boundary conditions and non-local boundary conditions:

$$T(\mathbf{x},0) = (1+y)\exp(\mathbf{x}), \quad m(t) = \frac{3}{2}\exp(t)(\exp(1)-1),$$
$$0 \le \mathbf{x} \le 1,$$

$$h_0(\mathbf{x}) = \exp(x), \quad h_2(\mathbf{x},t) = 2\exp(x+t), \quad t \ge 0.$$

$$h_3(\mathbf{x},t) = (1+y)\exp(t), \quad h_4(\mathbf{x},t) = (1+y)\exp(1+t),$$
$$t \ge 0, \quad h_1(\mathbf{x},t) = h_2(\mathbf{x},t)\cup h_3(\mathbf{x},t)\cup h_4(\mathbf{x},t)$$

The exact solution of this problem is:

$$T(\mathbf{x},t) = (1+y)\exp(x+t), \quad 0 \le \mathbf{x} \le 1, \quad t \ge 0,$$
$$\mu(t) = \exp(t).$$

The source term of this problem is:

$$f(\mathbf{x}) = (1+y)\exp(x+t) + u_1(1+y)\exp(x+t) + u_2\exp(x+t) - (1+y)\exp(x+t), \quad 0 \le \mathbf{x} \le 1.$$

The numerical results are shown in Tables 1 and 2, Figs. 1–12.

4.2. Problem 2

We consider the two-dimensional time-dependent convection–diffusion equation with Dirichlet, Neumann and non-local boundary condition:

$$\frac{\partial T}{\partial t} + \mathbf{u}\cdot\nabla T - \nabla\cdot(\mathbf{K}\nabla T) = f,$$
$$(\mathbf{x},t) \in [0,1]\times[0,1]\times[0,T_{\max}], \quad \mathbf{x} = (x,y)$$

subject to initial, Dirichlet boundary condition, Neumann boundary condition and non-local boundary condition:

$$T(\mathbf{x},0) = \exp(x+y+2t),$$
$$m(t) = \exp(2t)(\exp(2) - 2\exp(1) + 1), \quad 0 \le \mathbf{x} \le 1,$$

$$h_0(\mathbf{x}) = \exp(x), \quad h_1(\mathbf{x},t) = \exp(1+x+2t), \quad t \ge 0.$$

$$g_0(\mathbf{x},t) = \exp(y+2t), \quad g_1(\mathbf{x},t) = \exp(1+y+2t),$$
$$t \ge 0, \quad g(\mathbf{x},t) = g_0(\mathbf{x},t)\cup g_1(\mathbf{x},t)$$

The exact solution of this problem is:

$$T(\mathbf{x},t) = \exp(x+y+2t), \quad 0 \le \mathbf{x} \le 1, \quad t \ge 0,$$
$$\mu(t) = \exp(2t).$$

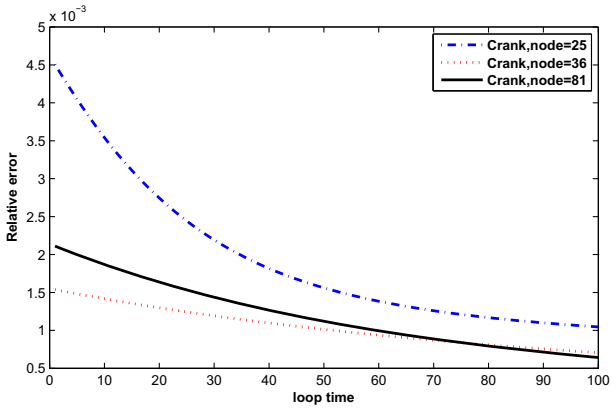

Figure 10 The relative error for $\mu(t)$ at $N = 25$, 36 and 81 with $M = 6$.

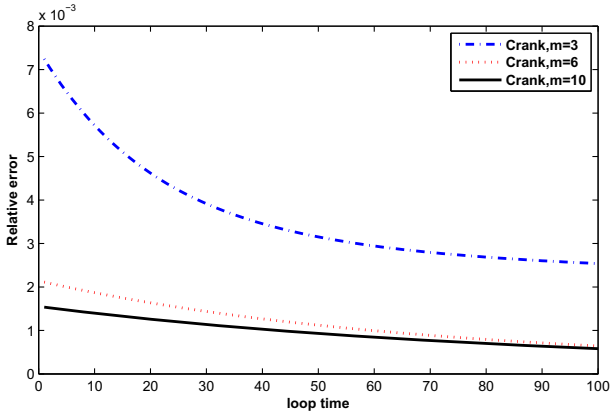

Figure 11 The relative error for $\mu(t)$ at $M = 3$, 6 and 10 with $N = 81$.

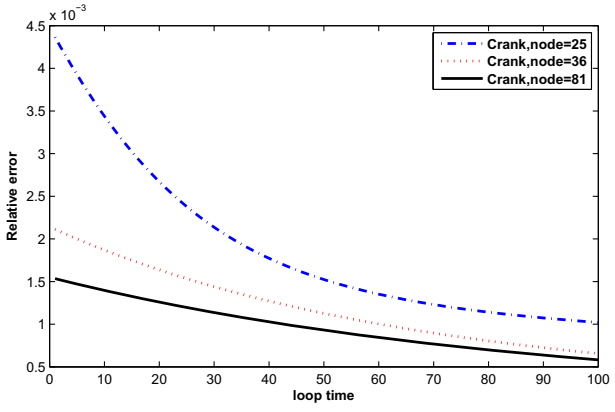

Figure 12 The relative error for $\mu(t)$ at $N = 25$, 36 and 81 with $M = 10$.

Table 3 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10, $N = 25$, 36 and 81.

N/m	3	6	10
25	1.9372e-2	5.3279e-3	4.4636e-3
36	1.6748e-2	4.8862e-3	2.7373e-3
81	5.8101e-3	2.8265e-3	1.5231e-3

Table 4 The relative error for $\mu(t)$ at $M = 3$, 6 and 10, $N = 25$, 36 and 81.

N/m	3	6	10
25	6.1612e-3	1.0348e-3	1.0079e-3
36	2.5005e-3	5.6160e-4	5.4160e-4
81	1.7075e-3	3.1049e-4	3.0039e-4

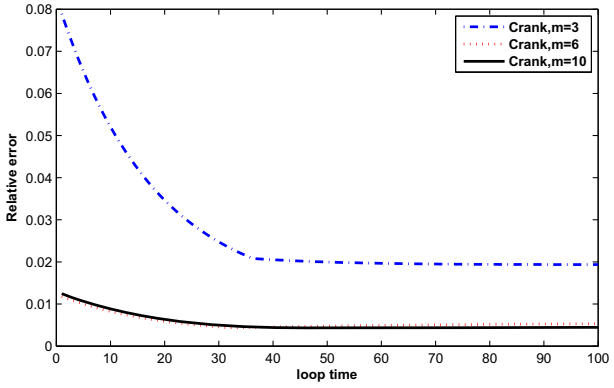

Figure 13 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10 with $N = 25$.

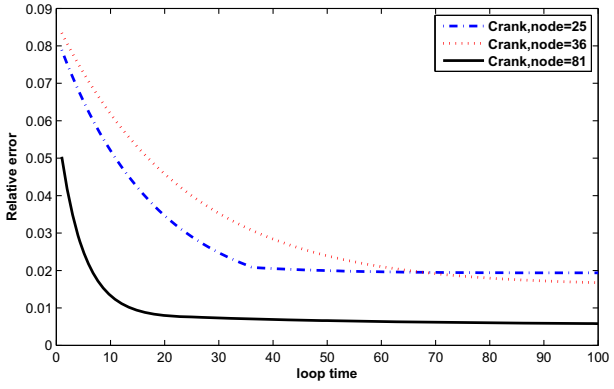

Figure 14 The relative error for $T(\mathbf{x}, t)$ at $N = 25$, 36 and 81 with $M = 3$.

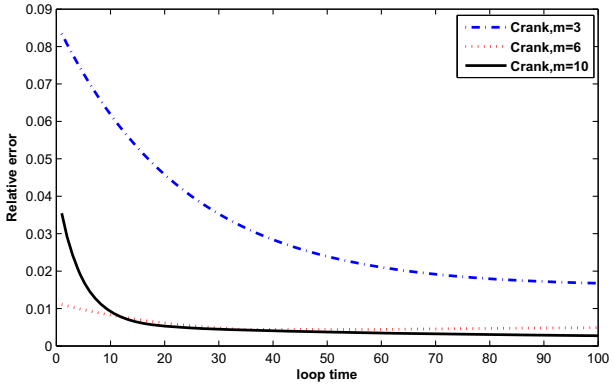

Figure 15 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10 with $N = 36$.

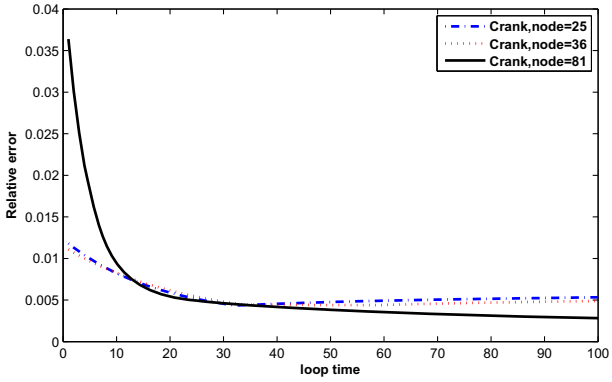

Figure 16 The relative error for $T(\mathbf{x}, t)$ at $N = 25$, 36 and 81 with $M = 6$.

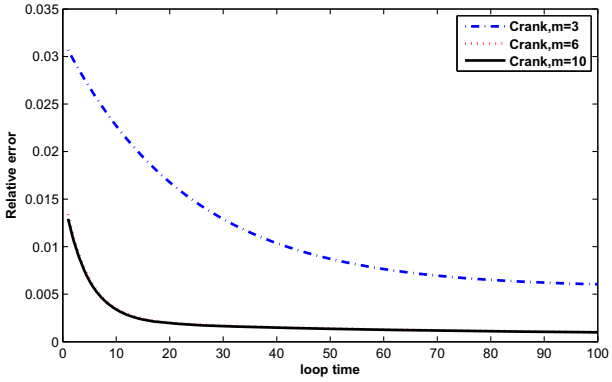

Figure 19 The relative error for $\mu(t)$ at $M = 3$, 6 and 10 with $N = 25$.

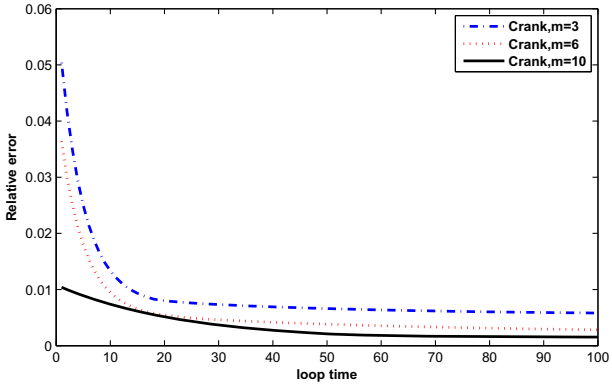

Figure 17 The relative error for $T(\mathbf{x}, t)$ at $M = 3$, 6 and 10 with $N = 81$.

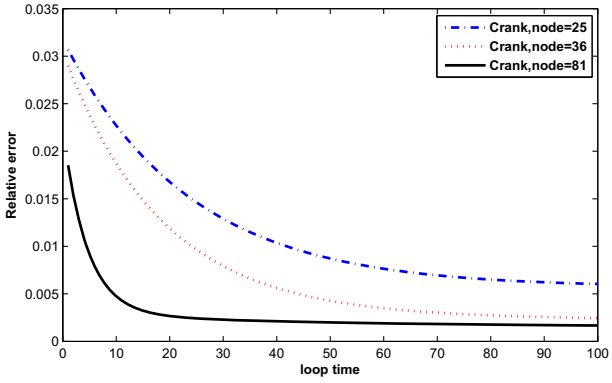

Figure 20 The relative error for $\mu(t)$ at $N = 25$, 36 and 81 with $M = 3$.

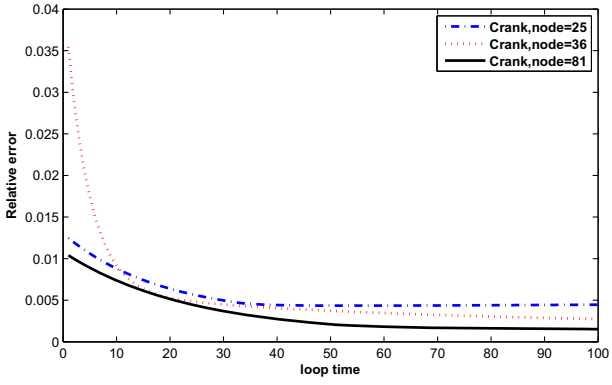

Figure 18 The relative error for $T(\mathbf{x}, t)$ at $N = 25$, 36 and 81 with $M = 10$.

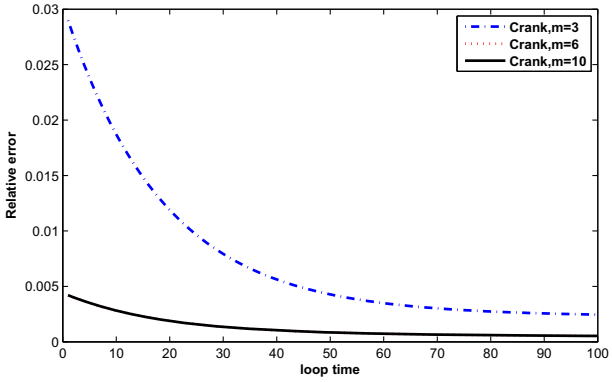

Figure 21 The relative error for $\mu(t)$ at $M = 3$, 6 and 10 with $N = 36$.

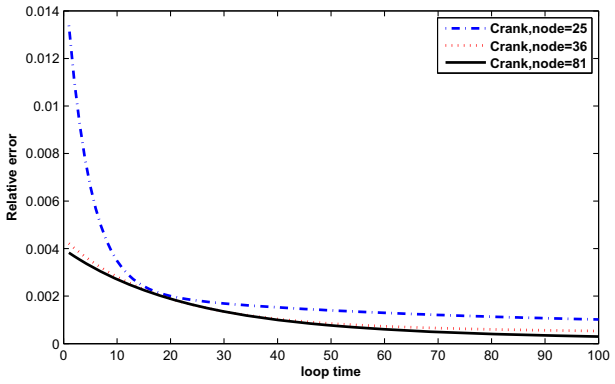

Figure 22 The relative error for $\mu(t)$ at $N = 25$, 36 and 81 with $M = 6$.

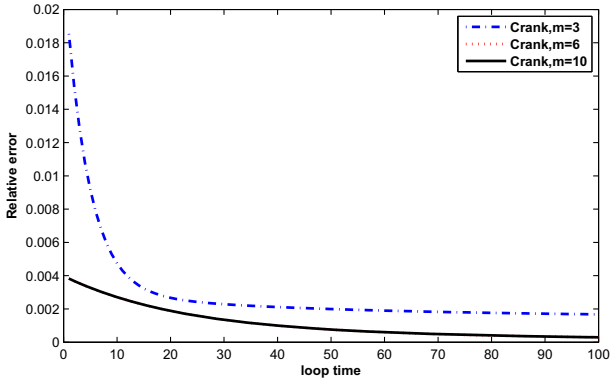

Figure 23 The relative error for $\mu(t)$ at $M = 3$, 6 and 10 with $N = 81$.

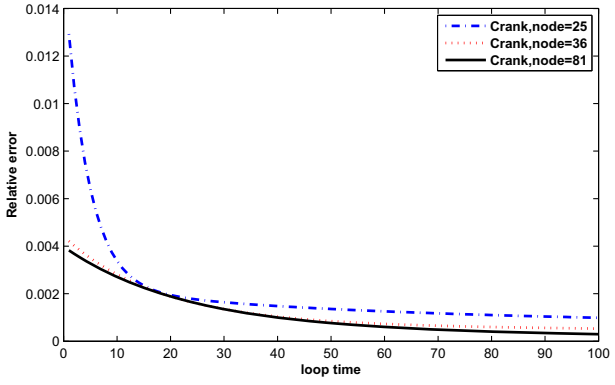

Figure 24 The relative error for $\mu(t)$ at $N = 25$, 36 and 81 with $M = 10$.

The source term of this problem is:

$$f(\mathbf{x}) = u_1 \exp(x + y + 2t) + u_2 \exp(x + y + 2t), \quad 0 \leq \mathbf{x} \leqslant 1.$$

The numerical results are shown in Tables 3 and 4, Figs. 13–24.

5. Conclusions

A formulation of meshless local Petrov–Galerkin (MLPG) has been presented in the present work. We have proposed an idea of MK interpolation demonstrating how it can be used to construct the meshless shape functions for solving the two-dimensional time-dependent convection–diffusion equation with integral conditions. To deal with time derivatives, the Crank–Nicolson implicit method was used. The key feature of the MK interpolation is that the shape function possesses the Kronecker delta property which makes it easy to directly impose Dirichlet boundary conditions. The numerical results show a decrease in relative error with increasing number of nodal points. Similarly, the relative error decreases when the size of the monomial basis increases. Moreover, it can be concluded that the numerical results of two test problems have been used to verify the efficiency, easiness and accuracy of the method.

Acknowledgements

This research is partially supported by the Science Achievement Scholarship of Thailand in Mathematics.

References

Abbasbandy, S., Shirzadi, A., 2010. A meshless method for two-dimensional diffusion equation with an integral condition. Eng. Anal. Bound. Elem. 34, 1031–1037.

Abbasbandy, S., Shirzadi, A., 2011. MLPG method for two-dimensional diffusion equation with Neumann's and non-classical boundary conditions. Appl. Numer. Math. 61, 170–180.

Atluri, S.N., Lin, H., 2000. Meshless local Petrov–Galerkin (MLPG) method for convection–diffusion problems. CMES 1 (2), 45–60.

Atluri, S.N., Zhu, T.L., 1998. A new meshless local Petrov–Galerkin (MLPG) approach in computational mechanics. Comput. Mech. 22, 117–127.

Belytschko, T., Lu, Y.Y., Gu, L., 1994. Element-free Galerkin methods. Int. J. Numer. Methods Eng. 37 (1), 229–256.

Goodrich, C.S., 2015. Coupled systems of boundary value problems with nonlocal boundary conditions. Appl. Math. Lett. 41, 17–22.

Islam, S.U., Aziz, I., Ahmad, M., 2015. Numerical solution of two-dimensional elliptic PDEs with nonlocal boundary conditions. Comput. Math. Appl. 69, 180–205.

Kaewumpai, S., Luadsong, A., 2014. Two-field-variable meshless method based on moving kriging interpolation for solving simply supported thin plates under various loads. J. King Saud Univ. Sci., 1018–3647

Liszka, T., Orkisz, J., 1980. The finite difference method on arbitrary irregular grid and its application in applied mechanics. Comput. Struct. 11, 83–95.

Liu, G.R., 2002. A point assembly method for stress analysis for two-dimensional solids. Int. J. Solids Struct. 39, 261–276.

Liu, G.R., Gu, Y.T., 2001. A point interpolation method for two-dimensional solid. Int. J. Numer. Methods Eng. 50, 937–951.

Liu, G.R., Liu, M.B., 2003. Smoothed Particle Hydrodynamics – A Meshfree Practical Method. World Scientific, Singapore.

Lucy, L., 1977. A numerical approach to testing the fission hypothesis. Astrophys. J. 82, 1013–1024.

Mahmoodabadi, M.J., Abedzadeh, R.M., Bagheri, A., Baradaran, G.H., 2011. Meshless local Petrov–Galerkin method for 3D steady-state heat conduction problems. Adv. Mech. Eng. http://dx.doi.org/10.1155/2011/251546 (10 pages).

Mirzaei, D., 2010. Meshless local Petrov–Galerkin (MLPG) approximation to the two dimensional sine-Gordor equation. Appl. Math. Comput. 233, 2737–2754.

Mohyud-Din, S., Yildirim, A., 2010. On two-dimensional diffusion with integral condition. J. King Saud Univ. Sci. 23 (2), 121–125.

Sajavicius, S., 2013. Optimization, conditioning and accuracy of radial basis function method for partial differential equations with nonlocal boundary condition-A case of two-dimensional Poisson equation. Eng. Anal. Bound. Elem. 37, 788–804.

Sataprahm, C., Luadsong, A., 2014. The meshless local Petrov–Galerkin method for simulating unsteady incompressible fluid flow. J. Egypt. Math. Soc. 23 (3), 501–510.

Techapirom, T., Luadsong, A., 2013. The MLPG with improved weight function for two-dimensional heat equation with non-local boundary condition. J. King Saud Univ. Sci. 25, 341–348.

Yimnak, K., Luadsong, A., 2014. A local integral equation formulation based on moving kriging interpolation for solving coupled nonlinear reaction–diffusion equations. Adv. Math. Phys.