Journal of Mathematical Extension
Vol. 15, SI-NTFCA, (2021) (24)1-25
URL: https://doi.org/10.30495/JME.SI.2021.2148
ISSN: 1735-8299
Original Research Paper

Collocated Meshless Method for Time-Fractional Diffusion-Wave Equations

B. F. Malidareh
Babol Branch, Islamic Azad University

Abstract. This paper goals to expand a meshless method using the collocated discrete least squares meshless (CDLSM) approach for mathematical modeling of a category of time fractional diffusion-wave equation (TFDWE). First, moving least squares (MLS) method used to construct the shape function is briefly described. Then, using the shape function generated by the least squares method, the discrete shape of the TFDWE is received in the strong form. Two-dimensional test problems with different nodes and collocations distributions are studied to validate and look at the accuracy and performance of proposed method.

AMS Subject Classification: 60G22; 26A33; 65C30.
Keywords and Phrases: Meshless Method, Least Squares, Fractional, Diffusion, Wave.

1 Introduction

Numerous phenomena in different areas of applied mathematics and mechanics can be simulated by equations with fractional derivatives. Some recent research articles can be found in [33, 15, 16, 34]. A class of these equations is the time fractional wave-diffusion equation, which is used in many important physical phenomena. It is obtained by replacing $1 < \alpha < 2$ in the second-order diffusion wave equations. In this research,

Received: August 2021; Published: October 2021

our studies and investigations focus on TFDWEs with Caputo derivative as follows

$$\frac{\partial^\alpha u(\mathbf{x},t)}{\partial t^\alpha} = \mu \Delta u(\mathbf{x},t) + f(\mathbf{x},t), \tag{1}$$

accompanied by the conditions,

$$\begin{cases} u(\mathbf{x},0) = \vartheta_0(\mathbf{x}), \\ \frac{\partial u(\mathbf{x},t)}{\partial t}\Big|_{t=0} = \vartheta_1(\mathbf{x}), \end{cases} \quad \mathbf{x} \in \Omega, \tag{2}$$

and appropriate Dirichlet and Neuman boundary conditions

$$\begin{cases} u = \bar{u}, \\ \mathcal{L}u = \bar{t}, \end{cases} \quad \mathbf{x} \in \Gamma, \tag{3}$$

In which $\mathcal{L}$ is a (partial) differential operator, $u(\mathbf{x},t)$ and α are unknown variable and order of time derivative respectively. In this research, $f(\mathbf{x},t)$ is source term and $\frac{\partial^\alpha u(\mathbf{x},t)}{\partial t^\alpha}$ is the Caputo fractional derivative with repect to "t" defined as,

$$\frac{\partial^\alpha u(\mathbf{x},t)}{\partial t^\alpha} = \frac{1}{\Gamma(2-\alpha)} \int_0^t \frac{\partial^2 u(\mathbf{x},\zeta)}{\partial \zeta^2} \frac{d\zeta}{(t-\zeta)^{\alpha-1}}, \qquad 1 < \alpha < 2.$$

It must be noted that most wave-diffusion equations don't have analytical solution, and as a result, a lot of research has been done to numerically solve such equations. The method of separation of variables, Sumudu transform approch, decomposition scheme, Finite difference method, radial point interpolation method, B-spline collocation approach, Sinc-Chebyshev scheme introduce by Chen et al. [4], Darzi et al. [6], Ray [30], Huang et al.[14], Hosseini et al. [13], Esen [10], Mao [26] respectively are some of numerical schemes for solving the TFDWE problems.

In the past decades, a number of meshless approaches have been developed that have been successfully considered by mathematicians and the engineering community [22]. The meshless methods have gained more attention not only by mathematicians but also in the engineering community and numerous meshless methods have been developed due to their easy implementation on complex geometries in the fields of applied mathematics and computational mechanics. Some of meshless methods

that use nodal interpolation techniques and can therefor be widely employed in various fields can be found in [23, 3, 36, 27, 8, 29, 28]. Nowdays, method based on meshless approaches were introduced to deal with fractional diffusion-wave equation [7, 31, 17]. One of the techniques used in the meshless method is the collocation method, which was first introduced by Slater [32] and then Barta [2], Frazer et al. [12] and Lanczos [19] developed it in their works. In two past decades, a some of meshless approaches using collocation have risen in studies [1, 9, 11, 35, 24, 25]. The diffusion-wave equation solve problems that occur in the computational mechanics and mathematical modeling. To simulate time-related unusual diffusion procedure, we can use TFDWE by inserting a fractional order time derivative in the classical diffusion-wave equation.
This study put to test TFDWEs using the CDLSM method and some numerical cases are given to highlight the efficiency and accuracy of the introduced approach.

2 The MLS Approximation

The moving least squares (MLS) approach, the radial point approximation, and kriging interpolation are only a few strategies to construct meshless shape functions. Among those methods, The MLS approximation is now broadly utilized in meshless approach to generate shape functions. The MLS was inroduced by mathematicians to fit data and construct surfaces [18, 5]. It may be classified as a manner to show a set of functions. Assume the approximation of $u(\mathbf{x})$ to be $u^h(\mathbf{x})$ as shown in Figure 1 and can be defined as,

$$u^h(\mathbf{x}) = \sum_{i=1}^{m} p_i(\mathbf{x}) a(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{a}(\mathbf{x}). \tag{4}$$

In Eq.(4), coefficients vector $\mathbf{a}(\mathbf{x})$ is written as

$$\mathbf{a}^{\mathrm{T}}(\mathbf{x}) = \{a_1 \quad a_2 \quad ... \quad a_m\}.$$

Note that the coefficient vector $\mathbf{a}(\mathbf{x})$ in Eq.(4) is function of $\mathbf{x}$ and it can be obtained by minimizing the following weighted discrete L_2 norm

$$J = \sum_{i=0}^{n} W(\mathbf{x} - \mathbf{x}_i)(\mathbf{p}^{\mathbf{T}}(\mathbf{x}_i)\mathbf{a}(\mathbf{x}) - u_i)^2.$$

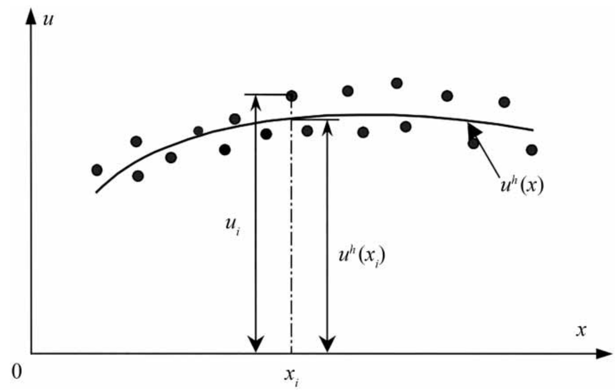


Figure 1: Approximate solutions $u(\mathbf{x})$ and nodal values u_i in the MLS interpolation

In the above equation, n is the number of nodal points in the support domain so that the weight function in this domain is opposite of zero and u_i is values of u at nodes. Since the number of nodal points n is greater than the number of unknown coefficients in the MLS interpolation, the approximation function u^h does not pass through all nodes. In this paper, the quadratic spline weighted function has been used and explained in more details in [21, 20]. The stationary of J with respect to a($\mathbf{x}$) gives

$$\begin{gathered}\frac{\partial J}{\partial \mathbf{a}} = 2\sum_{i=1}^{n} W\left(\mathbf{x}-\mathbf{x}_i\right)\mathbf{p}\left(\mathbf{x}_i\right)\left(\ \mathbf{p}^{\mathrm{T}}\left(x_i\right)\mathbf{a}\left(\mathbf{x}\right) - u_i\right) = 0 \rightarrow \\ \sum_{i=1}^{n} W_i\left(\mathbf{x}\right)\mathbf{p}\left(\mathbf{x}_i\right)\mathbf{p}^{\mathbf{T}}\left(\mathbf{x}_i\right)\mathbf{a}\left(\mathbf{x}\right) = \sum_{i=1}^{n} W_i\left(\mathbf{x}\right)\mathbf{p}\left(\mathbf{x}_i\right)u_i,\end{gathered} \tag{5}$$

where

$$W\left(\mathbf{x}-\mathbf{x}_i\right) = W_i\left(\mathbf{x}\right), \quad \mathbf{D}\left(\mathbf{x}\right) = \sum_{i=1}^{n} W_i\left(\mathbf{x}\right)\mathbf{p}\left(\mathbf{x}_i\right)\mathbf{p}^{\mathbf{T}}\left(\mathbf{x}_i\right),$$

and

$$\mathbf{E}(\mathbf{x}) = \left[W_1\left(\mathbf{x}\right)\mathbf{p}\left(\mathbf{x}_1\right) \quad W_2\left(\mathbf{x}\right)\mathbf{p}\left(\mathbf{x}_2\right) \ldots\ldots\ldots \quad W_n\left(\mathbf{x}\right)\mathbf{p}\left(\mathbf{x}_n\right)\right].$$

Eq.(4) and Eq.(5) yield familiar form of

$$u^h(\mathbf{x}) = \sum_{i=1}^{n} N_i(\mathbf{x})\, u_i = \mathbf{N}(\mathbf{x})\, \mathbf{u}_s, \tag{6}$$

Where $\mathbf{N}^{\mathrm{T}}(\mathbf{x})$ is the vector of MLS shape functions and $\mathbf{u}_s$ is Vector corresponding to nodal values where defined as follow

$$\mathbf{N}(\mathbf{x}) = \mathbf{p}^T \mathbf{D}^{-1}(\mathbf{x})\, \mathbf{E}(\mathbf{x}) = \left[N_1(\mathbf{x}) \quad N_2(\mathbf{x}) \quad .... \quad N_n(\mathbf{x})\right]. \tag{7}$$

As shown in Figure 1, the u_i is not equal to the values of the approximation solutions in the MLS interpolation, so it does not satisfy the Kronecker delta condition ($u^h(\mathbf{x}_i) \neq u_i$). The quartic spline function (W) is defined as follows

$$W_i(\mathbf{x}) = \begin{cases} 1 - 6\overline{r}_i^{\,2} + 8\overline{r}_i^{\,3} - 3\overline{r}_i^{\,4} & \overline{r}_i \leq 1 \\ 0 & \overline{r}_i > 1 \end{cases}$$

in which $W_i(\mathbf{x})$ has 3rd order continuity and $\overline{r} = \frac{|x - x_i|}{d_s}$. As shown in Figure 2, d_s is the size of the support domain which is as follows

$$d_s = \alpha_s d_c.$$

In the above equation, α_s is the dimensionless coefficient of the size of the support domain, and d_c is the nodal distance around the point at x_l. If the distribution of nodal points is uniform, then d_c is the distance between two adjacent points. When the distribution is non-uniform, d_c is the average distance between nodes within the support domain of $\mathbf{x}_l$. Generally, $\alpha_s = 2.0 \sim 3.0$ yields accurate results for many problems [21]. Here, $\alpha_s = 2.5$ has been considered. To obtain the spatial derivatives of the unknown variable, we must first obtain the derivatives of the shape function. For this purpose and based on Eqs.(6)-(7), we have

$$\mathbf{N}(\mathbf{x}) = \mathbf{\Theta}^{\mathbf{T}}(\mathbf{x})\, \mathbf{D}(\mathbf{x}),$$

Where $\mathbf{\Theta}(\mathbf{x})$ is determined by

$$\mathbf{E}(\mathbf{x})\, \mathbf{\Theta}(\mathbf{x}) = \mathbf{p}(\mathbf{x}).$$

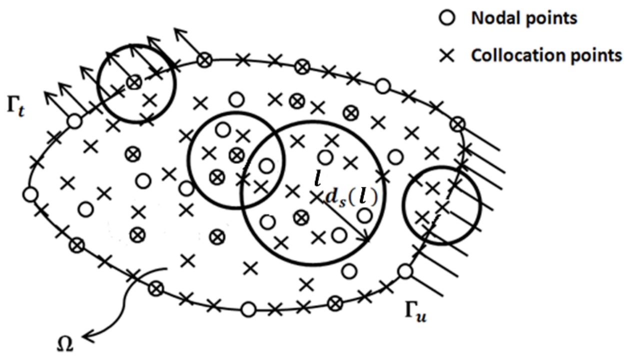


Figure 2: Field and boundary support domain

The partial derivatives of $\Theta(\mathbf{x})$ can be obtained as follows

$$\mathbf{E}\boldsymbol{\Theta}_{,x}=\mathbf{p}_{,x}-\mathbf{E}_{,x}\boldsymbol{\Theta},$$

$$\mathbf{E}\boldsymbol{\Theta}_{,xx}=\mathbf{p}_{,xx}-\mathbf{E}_{,xx}\boldsymbol{\Theta}-2\mathbf{E}_{,x}\boldsymbol{\Theta}_{,x},$$

$$\mathbf{E}\boldsymbol{\Theta}_{,y}=\mathbf{p}_{,y}-\mathbf{E}_{,y}\boldsymbol{\Theta},$$

$$\mathbf{E}\boldsymbol{\Theta}_{,yy}=\mathbf{p}_{,yy}-\mathbf{E}_{,yy}\boldsymbol{\Theta}-2\mathbf{E}_{,y}\boldsymbol{\Theta}_{,y}.$$

The numerical solution of $u(\mathbf{x},t)$ and its derivatives at a collocation point $\mathbf{x}_l$ can be obtained by the following linear combination

$$u(\mathbf{x})=\mathbf{N}(\mathbf{x})\,\mathbf{u}_{\mathrm{s}}, \tag{8}$$

$$u_x(\mathbf{x})=\mathbf{N}_x(\mathbf{x})\,\mathbf{u}_s, \tag{9}$$

$$u_y(\mathrm{x})=\mathbf{N}_y(\mathbf{x})\,\mathbf{u}_s, \tag{10}$$

$$u_{xx}(\mathrm{x})=\mathbf{N}_{xx}(\mathbf{x})\,\mathbf{u}_s, \tag{11}$$

$$u_{yy}(\mathbf{x})=\mathbf{N}_{yy}(\mathbf{x})\,\mathbf{u}_s, \tag{12}$$

where $\mathbf{N}_x$, $\mathbf{N}_{xx}$, $\mathbf{N}_y$ and $\mathbf{N}_{yy}$ are vector of first and second derivative of Eq.(8) with respect to x and y respectively as follows

$$\mathbf{N}_x(\mathbf{x})=\left[\frac{\partial N_1(\mathbf{x})}{\partial x} \quad \frac{\partial N_2(\mathbf{x})}{\partial x} \quad \cdots\cdot \quad \frac{\partial N_n(\mathbf{x})}{\partial x}\right],$$

$$\mathbf{N}_y(\mathbf{x}) = \left[\frac{\partial N_1(\mathbf{x})}{\partial y} \quad \frac{\partial N_2(\mathbf{x})}{\partial y} \quad \cdots \quad \frac{\partial N_n(\mathbf{x})}{\partial y}\right],$$

$$\mathbf{N}_{xx}(\mathbf{x}) = \left[\frac{\partial^2 N_1(\mathbf{x})}{\partial x^2} \quad \frac{\partial^2 N_2(\mathbf{x})}{\partial x^2} \quad \cdots \quad \frac{\partial^2 N_n(\mathbf{x})}{\partial x^2}\right],$$

$$\mathbf{N}_{yy}(\mathbf{x}) = \left[\frac{\partial^2 N_1(\mathbf{x})}{\partial y^2} \quad \frac{\partial^2 N_2(\mathbf{x})}{\partial y^2} \quad \cdots \quad \frac{\partial^2 N_n(\mathbf{x})}{\partial y^2}\right].$$

3 Discretization of Time Fractional Derivative

The fractional derivative with respect to time at $t{=}t_{k+1}$ is defined by

$$\frac{\partial^\alpha u(\mathbf{x},t)}{\partial t^\alpha}{=}\frac{1}{\Gamma(2{-}\alpha)}\int_0^t \frac{\partial^2 u(\mathbf{x},\zeta)}{\partial \zeta^2}\frac{d\zeta}{(t{-}\zeta)^{\alpha-1}}, \tag{13}$$

where $t_k{=}k\Delta t$, $k{=}\ 1,2,\ldots,m$ and Δt is time step
Let $\mathrm{z}\left(\mathbf{x},t\right){=}\frac{\partial u(\mathbf{x},t)}{\partial t}$, then Eq.(13) can be written as

$$w\left(\mathbf{x},t\right){=}\frac{1}{\Gamma(2{-}\alpha)}\int_0^t \frac{\partial \mathrm{z}(\mathbf{x},\zeta)}{\partial \zeta}\frac{d\zeta}{(t{-}\zeta)^{\alpha-1}}.$$

$w\left(\mathbf{x},t\right)$ can be obtained at $t = t_{k+1}$ as follow

$$\begin{aligned} w(\mathbf{x},t_{k+1}){=}&\frac{1}{\Gamma(2{-}\alpha)}\sum_{j=0}^{k}\int_{t_j}^{t_{j+1}} \frac{\partial \mathrm{z}(\mathbf{x},\zeta)}{\partial \zeta}\frac{d\zeta}{t(t_{k+1}{-}\zeta)^{\alpha-1}} \\ =&\frac{1}{\Gamma(2{-}\alpha)}\sum_{j=0}^{k}[\frac{\mathrm{z}(\mathbf{x},t_{j+1}){-}\mathrm{z}(\mathbf{x},t_j)}{\Delta t}+O(\Delta t)] \ \times \ \int_{t_j}^{t_{j+1}}(t_{k+1}{-}\zeta)^{1-\alpha}d\zeta. \end{aligned} \tag{14}$$

By defining $b_j = (j+1)^{2-\alpha} - j^{2-\alpha}$, then Eq.(14) can be writen as follow

$$w\left(\mathbf{x},t_{k+1}\right) = \frac{(\Delta t)^{1-\alpha}}{\Gamma(3{-}\alpha)}\sum_{j=0}^{k} b_{k-j}\left[\mathrm{z}\left(\mathbf{x},t_{j+1}\right) - \mathrm{z}\left(\mathbf{x},t_j\right)\right] + O((\Delta t)^{3-\alpha}). \tag{15}$$

By definition of $\mathrm{z}\left(\mathbf{x},t\right)$ as follow

$$\frac{\partial u(\mathbf{x},t_{j+1})}{\partial t}{=}\mathrm{z}\left(\mathbf{x},t_{j+1}\right),$$

and

$$\frac{\partial u(\mathbf{x},t_j)}{\partial t}=\mathrm{z}\left(\mathbf{x},t_j\right).$$

Then using Taylor's expansion, we have

$$u\left(\mathbf{x},t_{j-1}\right)=u\left(\mathbf{x},t_j\right)-\Delta t\frac{\partial u\left(\mathbf{x},t_j\right)}{\partial t}+\frac{\Delta t^2}{2!}\frac{\partial u\left(\mathbf{x},t_j\right)}{\partial t}-\dots$$

$$u\left(\mathbf{x},t_{j}\right)=u\left(\mathbf{x},t_{j+1}\right)-\Delta t\frac{\partial u\left(\mathbf{x},t_{j+1}\right)}{\partial t}+\frac{\Delta t^2}{2!}\frac{\partial u\left(\mathbf{x},t_{j+1}\right)}{\partial t}-\dots$$

Hence

$$\frac{\partial u(\mathbf{x},t_{j+1})}{\partial t}-\frac{\partial u(\mathbf{x},t_j)}{\partial t}=\frac{u(\mathbf{x},t_{j+1})-2u(\mathbf{x},t_j)+u(\mathbf{x},t_{j-1})}{\Delta t}+O(\Delta t),$$

or

$$\mathrm{z}\left(\mathbf{x},t_{j+1}\right)-\mathrm{z}\left(\mathbf{x},t_j\right)=\frac{u\left(\mathbf{x},t_{j+1}\right)-2u\left(\mathbf{x},t_j\right)+u\left(\mathbf{x},t_{j-1}\right)}{\Delta t}+O(\Delta t). \tag{16}$$

Substitution Eq.(16) into Eq.(15) yields

$$w(\mathbf{x},t_{k+1})=\frac{(\Delta t)^{-\alpha}}{\Gamma(3-\alpha)}\sum_{j=0}^{k}b_{k-j}[u(\mathbf{x},t_{j+1})-2u(\mathbf{x},t_j)+u(\mathbf{x},t_{j-1}) + O(\Delta t)]+O((\Delta t)^{3-\alpha}),$$

or

$$w(\mathbf{x},t_{k+1})=\frac{(\Delta t)^{-\alpha}}{\Gamma(3-\alpha)}\sum_{j=0}^{k}b_{j}[u(\mathbf{x},t_{k-j+1})-2u(\mathbf{x},t_{k-j})+u(\mathbf{x},t_{k-j-1})] + O((\Delta t)^{1-\propto})+O((\Delta t)^{3-\alpha}).$$

Now omitting truncation error terms, we have

$$w\left(\mathbf{x},t_{k+1}\right)=\frac{(\Delta t)^{-\alpha}}{\Gamma(3-\alpha)}\sum_{j=0}^{k}b_{j}\left[u\left(\mathbf{x},t_{k-j+1}\right)-2u\left(\mathbf{x},t_{k-j}\right)+u\left(\mathbf{x},t_{k-j-1}\right)\right]. \tag{17}$$

4 Numerical Scheme

Eq.(1) was discretized by the θ-method with respect to time using Eq.(17) as follows:

$$\frac{(\Delta t^{-\alpha}}{\Gamma(3-\alpha)}\sum_{j=0}^{k} b_j[u(\mathbf{x},t_{k-j+1})-2u(\mathbf{x},t_{k-j})+u(\mathbf{x},t_{k-j-1})]$$
$$-\theta\mu\Delta u(\mathbf{x},t_{k+1})-(1-\theta)\times\mu\Delta u(\mathbf{x},t_k)+F(u(\mathbf{x},t_k))-f(\mathbf{x},t_{k+1})=0. \tag{18}$$

Eq.(18) can be written as follow

$$u(\mathbf{x},t_{k+1})-\theta\mu\chi\Delta u(\mathbf{x},t_{k+1})-2u(\mathbf{x},t_k)+u(\mathbf{x},t_{k-1})+$$
$$\sum_{j=1}^{k} b_j[u(\mathbf{x},t_{k-j+1})-2u(\mathbf{x},t_{k-j})+u(\mathbf{x},t_{k-j-1})]-$$
$$(1-\theta)\mu\chi\Delta u(\mathbf{x},t_k)+\chi F(u(\mathbf{x},t_j))-\chi f(\mathbf{x},t_{j+1})=0, \tag{19}$$

where $\chi=(\Delta t)^{-\alpha}\,\Gamma\,(3-\alpha)$
It must be noted that, to evaluation unknown variable u in Eq.(19) at each time, we must calculate t_{-1}. Based on Eq.(2), using central differences method with respect to $t=0$ gives

$$\frac{u\,(\mathbf{x},t_1)-u\,(\mathbf{x},t_{-1})}{2\Delta t}=\vartheta_1(\mathbf{x}),$$

Hence

$$u\,(\mathbf{x},t_{-1})=u\,(\mathbf{x},t_1)-2\Delta t\vartheta_1(\mathbf{x}).$$

The substitution of Eqs.(8)-(12) into Eq.(19) yields residual equation (R_{Ω}) at collocations of $\mathbf{x}_l$ as follows

$$\begin{aligned}\mathbf{R}_l^{\Omega}&=\mathbf{N}(\mathbf{x}_l)\boldsymbol{u}_s-\theta\mu\chi(\mathbf{N}_{xx}(\mathbf{x}_l)+\mathbf{N}_{yy}(\mathbf{x}_l))\boldsymbol{u}_s-2u(\mathbf{x}_l,t_k)+u(\mathbf{x}_l,t_{k-1})\\&+\sum_{j=1}^{k} b_j[u(\mathbf{x}_l,t_{k-j+1})-2u(\mathbf{x}_l,t_{k-j})+u(\mathbf{x}_l,t_{k-j-1})]\\&-(1-\theta)\mu\chi\Delta u(\mathbf{x}_l,t_k)+\chi F(u(\mathbf{x}_l,t_k))-\chi f(\mathbf{x}_l,t_{k+1})\\&\qquad\qquad l=1,2,\ \ldots,n_d.\end{aligned} \tag{20}$$

By substitution Eqs.(8)-(12) into Eq.(3)

$$\mathbf{R}_l^u = \mathbf{N}\left(\mathbf{x}_l\right)\mathbf{u}_s - \overline{u}\left(\mathbf{x}_l\right) \qquad l = 1, 2, \; \ldots . , n_u, \tag{21}$$

$$\mathbf{R}_l^t = \mathcal{L}\mathbf{N}\left(\mathbf{x}_l\right)\mathbf{u}_s - \overline{t}\left(\mathbf{x}_l\right) \qquad l = 1, 2, \; \ldots . , n_t, \tag{22}$$

where n_d, n_u and n_t are the number of collocations in the domain and on Dirichlet and Neuman boundary conditions respectively. Eqs.(20)-(22) can be written as follow

$$\mathbf{R}_\Omega = \mathbf{L}\mathbf{u}_s - \mathbf{b},$$

$$\mathbf{R}_u = \mathbf{A}\mathbf{u}_s - \mathbf{c},$$

$$\mathbf{R}_t = \mathbf{B}\mathbf{u}_s - \mathbf{d},$$

Where

$$\mathbf{L}_{lj} = \mathbf{N}\left(\mathbf{x}_l\right) - \theta\mu\chi\left(\frac{\partial^2 N_j\left(\mathbf{x}_l\right)}{\partial x^2} + \frac{\partial^2 N_j\left(\mathbf{x}_l\right)}{\partial y^2}\right)$$
$$, \; l = 1, 2, \; \ldots . , n_d \; \; and \; j = 1, 2, \; \ldots . , n,$$

$$\mathbf{b}_l = -2u(\mathbf{x}_l, t_k) + u(\mathbf{x}_l, t_{k-1}) + \sum_{j=1}^{k} b_j[u(\mathbf{x}_l, t_{k-j+1}) - 2u(\mathbf{x}_l, t_{k-j}) + u(\mathbf{x}_l, t_{k-j-1})]$$
$$- (1-\theta)\mu\chi\Delta u(\mathbf{x}_l, t_k) + \chi F(u(\mathbf{x}_l, t_k)) - \chi f(\mathbf{x}_l, t_{k+1})$$
$$, l = 1, 2, \; \ldots . , n_d,$$

$$\mathbf{A}_{lj} = N_j\left(\mathbf{x}_l\right) \qquad l = 1, 2, \; \ldots . , n_u \quad and \; j = 1, 2, \; \ldots . , n,$$
$$\mathbf{c}_l = \overline{u}(\mathbf{x}_l) \qquad l = 1, 2, \; \ldots . , n_u,$$
$$\mathbf{B}_{lj} = \mathcal{L}N_j\left(\mathbf{x}_{\boldsymbol{l}}\right) \qquad l = 1, 2, \; \ldots . , n_t \quad and \; j = 1, 2, \; \ldots . , n,$$
$$\mathbf{d}_l = \overline{t}\left(\mathbf{x}_l\right) \qquad l = 1, 2, \; \ldots . , n_t.$$

Using penalty techniques, we define discrete L_2 norm as

$$I = \|\mathbf{L}\mathbf{u}_s - \mathbf{b}\|_2^2 + \alpha\|\mathbf{A}\mathbf{u}_s - \mathbf{c}\|_2^2 + \beta\|\mathbf{B}\mathbf{u}_s - \mathbf{d}\|_2^2, \tag{23}$$

Where α and β are penalty coefficients that present the ratio of the residual on the boundary to the residual of the computational domain. Optimizing Eq.(23) with respect to u_i yields

$$\nabla I = \mathbf{L}^{\boldsymbol{T}}\left(\mathbf{L}\mathbf{u}_s - \mathbf{b}\right) + \alpha\mathbf{A}^{\boldsymbol{T}}\left(\mathbf{A}\mathbf{u}_s - \mathbf{c}\right) + \beta\mathbf{B}^{\boldsymbol{T}}\left(\mathbf{B}\mathbf{u}_s - \mathbf{d}\right) = \mathbf{0},$$

$$\mathbf{K}\mathbf{u}_s = \mathbf{F}. \tag{24}$$

The stiffness matrix $\mathbf{K}$ and load vector $\mathbf{F}$ can be calculated as

$$\mathbf{K} = \left(\mathbf{L}^{\boldsymbol{T}}\mathbf{L} + \alpha\mathbf{A}^{\boldsymbol{T}}\mathbf{A} + \beta\mathbf{B}^{\boldsymbol{T}}\mathbf{B}\right), \tag{25}$$

$$\mathbf{F} = \mathbf{b} + \alpha\mathbf{c} + \beta\mathbf{d}.$$

Matrix $\mathbf{K}$ in Eq.(25) is a symmetric and and positive-definite that can be solved using a suitable method using the direct or iterative method. Solving system Eq.(24) leads to quantities of u_i and then approximation function $u^h(x)$ at any collocation point of x_l is calculated by Eq.(6). Using collocations (additional points) can increase the accuracy of the problem without increasing the dimension of the stiffness matrix, and it causes approach being more efficient.

5 Numerical Examples

We now utilize a variety of problems to clarify the theoretical concepts that we discussed in the previous sections. To realize the accuracy and efficiency of the proposed approach, we adopt the L_2 and L_∞ norms, defined as,

$$L_2 = \sqrt{\frac{\sum_{i=0}^{M}\left(u_i^{\text{exact}} - u_i^{app}\right)^2}{\sum_{i=0}^{M}\left(u_i^{\text{exact}}\right)^2}},$$

$$L_\infty = max|u_i^{\text{exact}} - u_i^{app}|_{1\le i\le M},$$

u^{exact} and u^{app} represent exact and numerical solutions respectively and M is number of collocations used in computational domain Ω and on boundary conditions Γ.

Example 5.1. we assume following test problem

$$\frac{\partial^\alpha u(\mathbf{x},t)}{\partial t^\alpha} = \Delta u(\mathbf{x},t) + f(\mathbf{x},t).$$

We deduce the boundary and initial condition of the following analytical result
$u(\mathbf{x},t) = t^2 \sin(x+y)$.
The linear source term is assumed as $f(\mathbf{x},t) = \left(\frac{2t^{2-\alpha}}{\Gamma(3-\alpha)} + 2t^2\right)\sin(x+y)$.

The above test problem is solved in the range of $[0,1]^2$ for different numbers of Δt and α. The results can be seen in Table 1. Nodes and collocations arrangement are illustrated in Figure 3. In Figure 4, a plot of the numerical solution at $t=3$ is presented. Figure 5 shows that this method can model the problem with high accuracy. Table 2 summarizes L_2 and L_∞ between approximated results with respect to analytical solutions with different the number of points at $t=3$. As shown in Table 2, the accuracy of the problem has increased as the number of points increases.

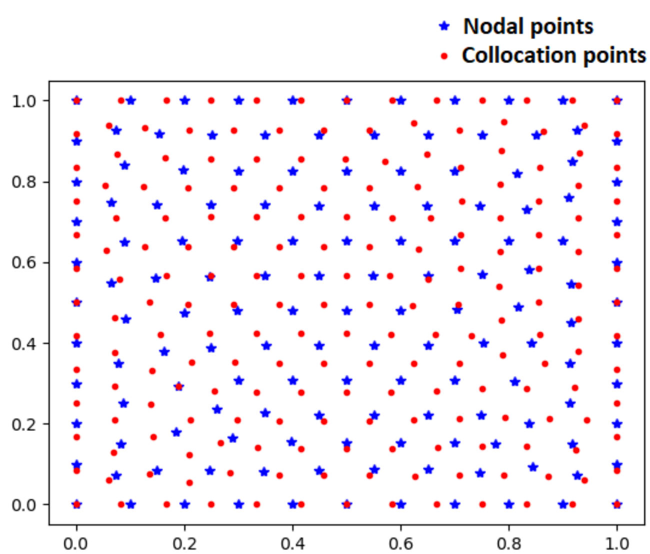
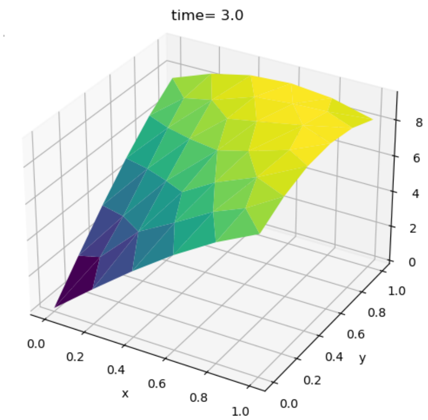
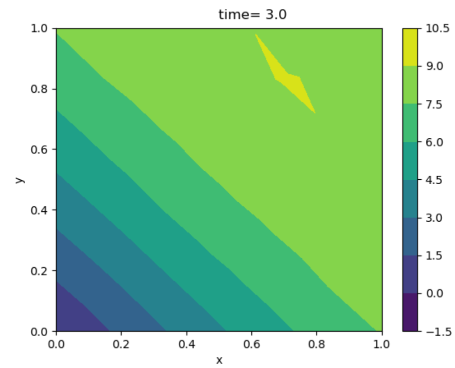


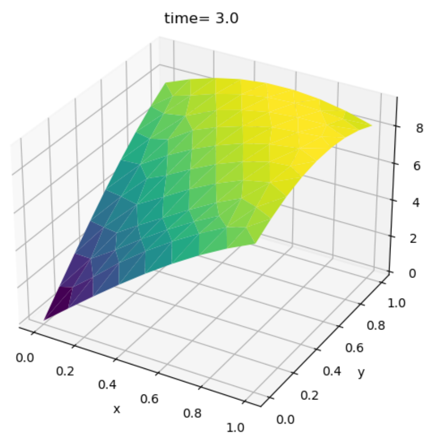
Figure 3: Nodal and collocation points arrangements.



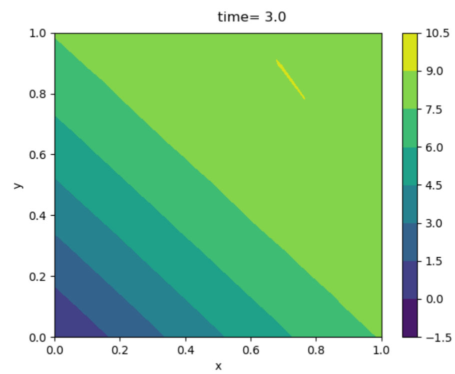
(a) Approximated result for Case A; nodes=30, collocations=44



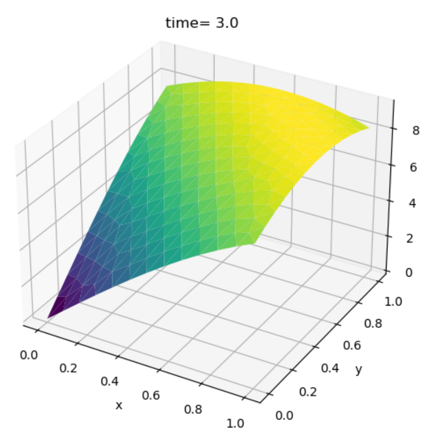
(b) Contour plot for Case A; nodes=30, collocations=44



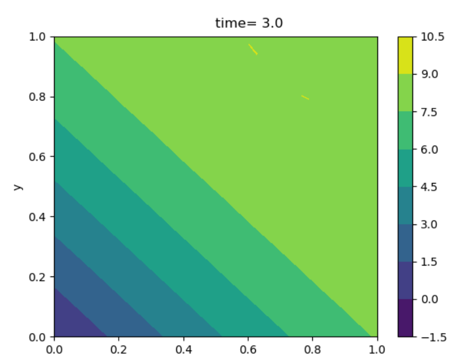
(c) Approximated result for Case B; nodes=74, collocations=98



(d) Contour plot for Case B; nodes=74, collocations=98

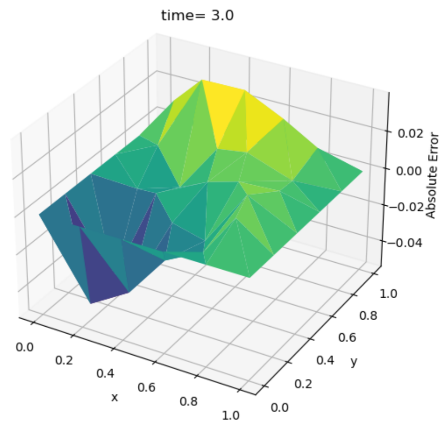


(e) Approximated result for Case C; nodes=142, collocations=198

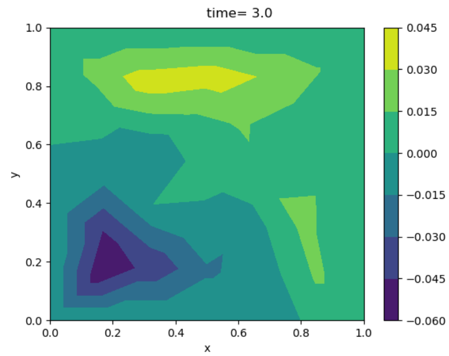


(f) Contour plot for Case C; nodes=142, collocations=198

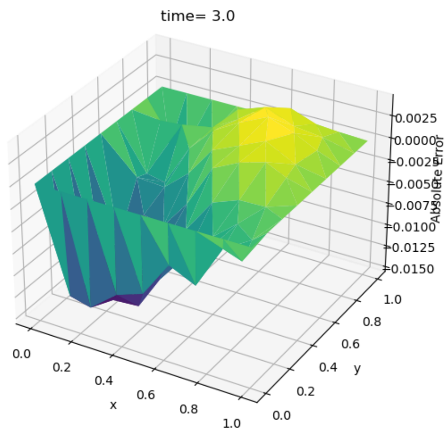
Figure 4: Approximated result and contour plot at $t = 3$ for different nodes and collocation points; $\alpha = 1.85$ and $\Delta t = 0.05$



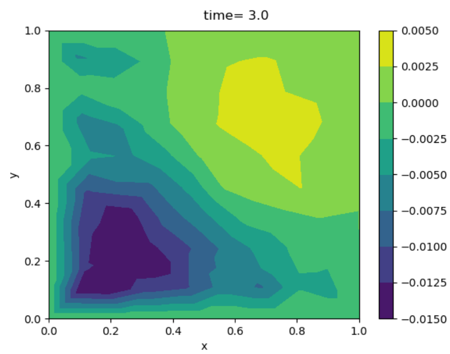
(a) Absolute error for Case A; nodes=30, collocations=44



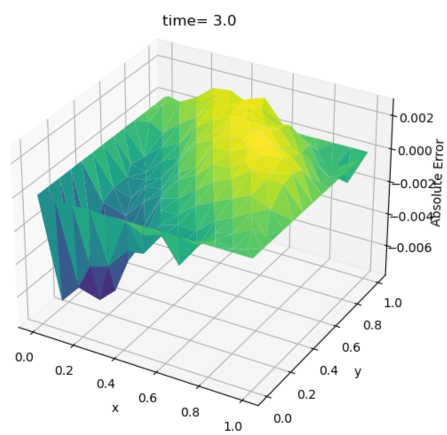
(b) Contour plot for Case A; nodes=30, collocations=44



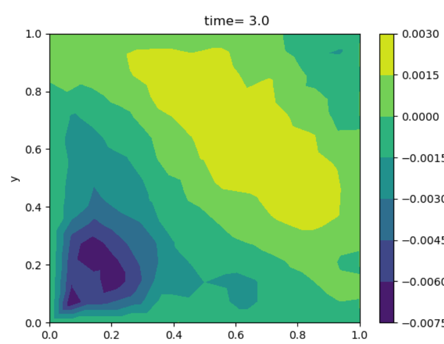
(c) Absolute error for Case B; nodes=74, collocations=98



(d) Contour plot for Case B; nodes=74, collocations=98



(e) Absolute error for Case C; nodes=142, collocations=198



(f) Contour plot for Case C; nodes=142, collocations=198

Figure 5: Absolute error and contour plot at $t = 3$ for different nodes and collocation points; $\alpha = 1.85$ and $\Delta t = 0.05$

Δt	$\alpha = 1.85$		$\alpha = 1.15$		CPU time(s)
	L_2	L_∞	L_2	L_∞	
0.01	$9.8438e-5$	$2.9576e-3$	$9.9798e-5$	$3.0345e-3$	58.46 s
0.025	$1.2101e-4$	$5.4276e-3$	$1.2767e-4$	$5.7549e-3$	28.25 s
0.05	$2.8488e-4$	$7.4516e-3$	$3.0750e-4$	$7.6618e-3$	9.84 s
0.1	$6.5234e-4$	$1.0224e-2$	$7.4335e-3$	$1.4516e-2$	5.54 s

Table 1: error values at different numbers of α and Δt at time $t = 3$, nodal points=142 and collocation points=198

Case	No. of Nodes	No. of collocations	L_2	L_∞	CPU time
A	30	44	$8.4155e-3$	$5.2187e-2$	1.99 s
B	74	98	$7.7233e-3$	$1.4969e-2$	4.29 s
C	142	198	$2.8488e-4$	$7.4516e-3$	9.85 s

Table 2: Approximated results of the problem using different points at $t = 3$ and $\Delta t = .05$.

Example 5.2. Consider following test problem

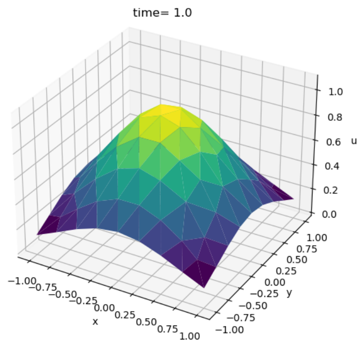
$$\frac{\partial^\alpha u(\mathbf{x},t)}{\partial t^\alpha} = \nabla u(\mathbf{x},t) + f(\mathbf{x},t).$$

The boundary and initial conditions are obtained from the following equation $u(\mathbf{x},t) = t^2 e^{-(x^2+y^2)}$. The source terms is $f(\mathbf{x},t) = (\frac{2t^{2-\alpha}}{\Gamma(3-\alpha_1)} + 4t^2(x^2+y^2) - 4t^2)e^{-(x^2+y^2)}$. The above problem has been solved in range $[-1,1]^2$ with $\alpha = 1.85$. The numerical results at $t = 1$ with $\Delta t = 0.05$ are listed in Table 3. In Figures 6 and 7, we illustrated the approximated solution and absolute error for different number of nodes and collocations.

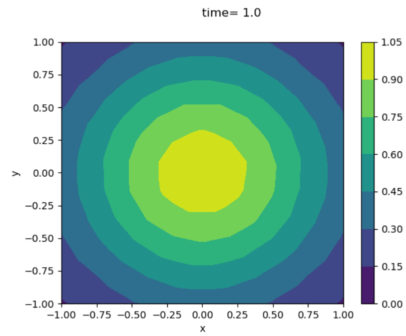
Example 5.3. Assume following test problem

$$\frac{\partial^\alpha u(\mathbf{x},t)}{\partial t^\alpha} = \Delta u(\mathbf{x},t) + f(\mathbf{x},t).$$

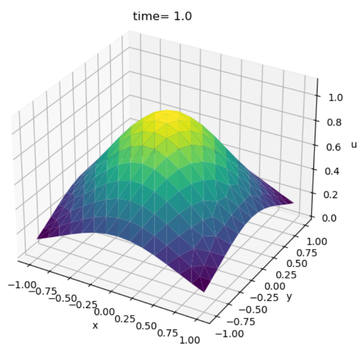
This example is adopted from [17]. The analytical solution is used to deduce initial and boundary conditions, $u(\mathbf{x},t) = \cos(\pi x)\cos(\pi y)t^{3+\alpha}$.



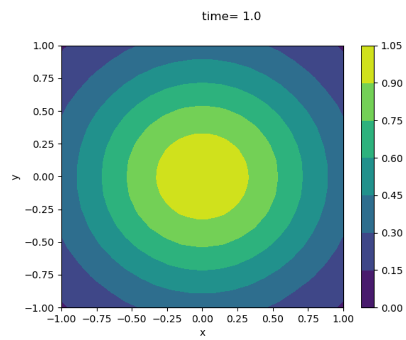
(a) Approximated result for Case A; nodes=58, collocations=98



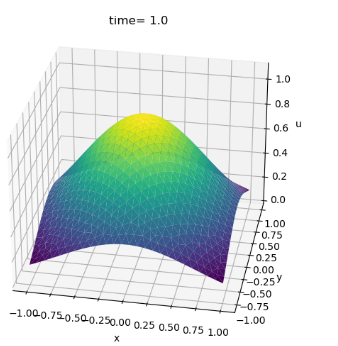
(b) Contour plot for Case A; nodes=58, collocations=98



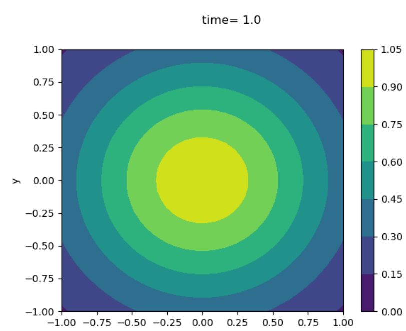
(c) Approximated result for Case B; nodes=144, collocations=258



(d) Contour plot for Case B; nodes=144, collocations=258

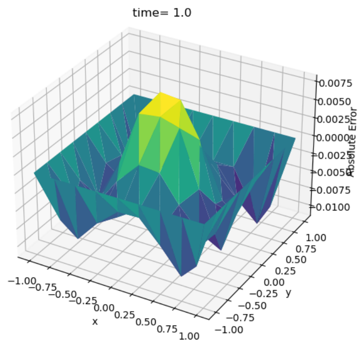


(e) Approximated result for Case C; nodes=514, collocations=790

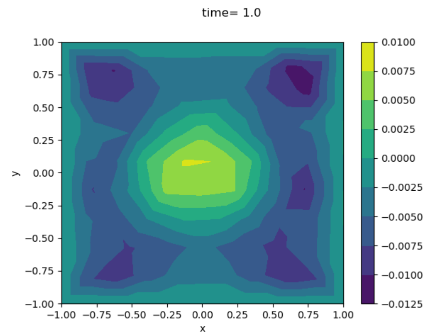


(f) Contour plot for Case C; nodes=514, collocations=790

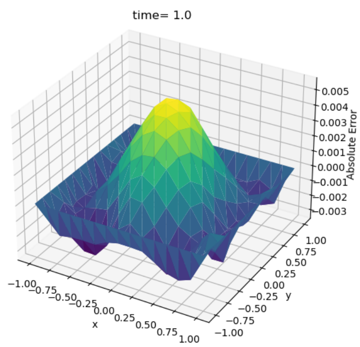
Figure 6: Approximated result and contour plot at $t = 1$ for different nodes and collocation points; $\alpha = 1.85$ and $\Delta t = 0.05$



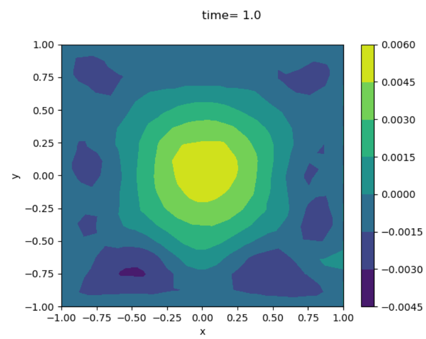
(a) Absolute error for Case A; nodes=58, collocations=98



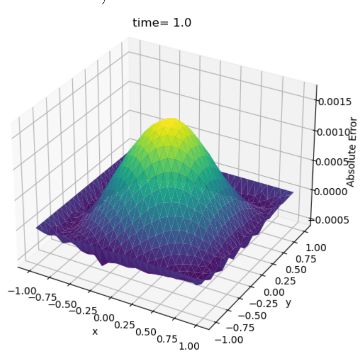
(b) Contour plot for Case A; nodes=58, collocations=98



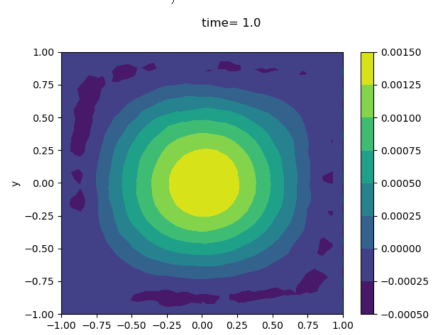
(c) Absolute error for Case B; nodes=144, collocations=258



(d) Contour plot for Case B; nodes=144, collocations=258



(e) Absolute error for Case C; nodes=514, collocations=790



(f) Contour plot for Case C; nodes=514, collocations=790

Figure 7: Absolute error and contour plot at $t = 1$ for different nodes and collocation points; $\alpha = 1.85$ and $\Delta t = 0.05$

Case	No. of Nodes	No. of collocations	L_2	L_∞	CPU time
A	58	98	$9.1945e-3$	$1.0901e-2$	1.96 s
B	144	258	$3.3013e-3$	$5.3810e-3$	4.87 s
C	514	790	$8.9051e-4$	$1.4972e-3$	22.45 s

Table 3: Approximated results of the problem using different points at $t = 1$ and $\Delta t = 0.05$.

The linear source term is assumed as

$$f(\mathbf{x}, t) = (\frac{\Gamma(4+\alpha)}{6}t^3 + 2\pi^2 t^{3+\alpha})\cos(\pi x)\cos(\pi y).$$

To show that this method can model irregular domain as well as Newman boundaries with high accuracy, the computational domain shown in Figure 8 is used. We approximate the solution of this problem with $\alpha = 1.50$ and $\Delta t = 0.05$ at time $t = 1$, the results are summarized in Table 4. In Figures 9 and 10, we display the results for different number of nodes and collocations.

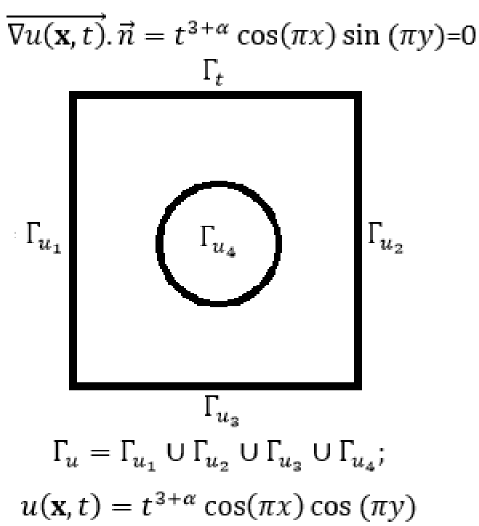
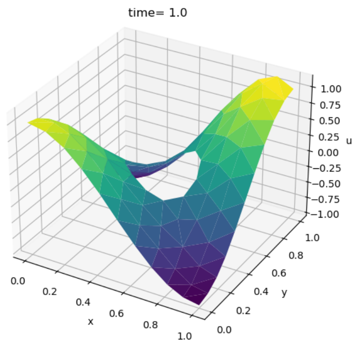
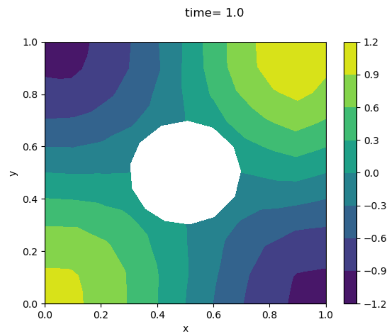


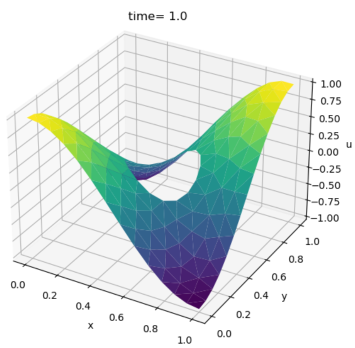
Figure 8: computational domain with Dirichlet and Neumann boundary conditions.



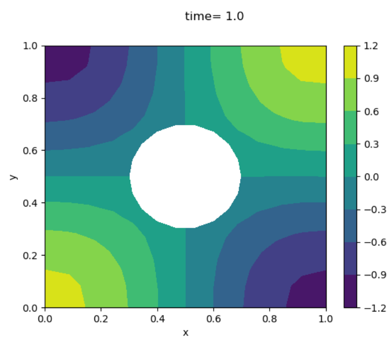
(a) Approximated result for Case A; nodes=42, collocations=106



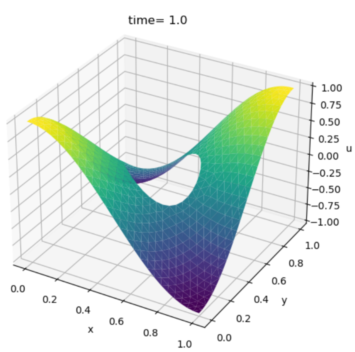
(b) Contour plot for Case A; nodes=42, collocations=106



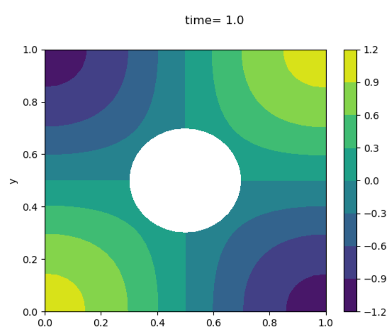
(c) Approximated result for Case B; nodes=106, collocations=172



(d) Contour plot for Case B; nodes=106, collocations=172

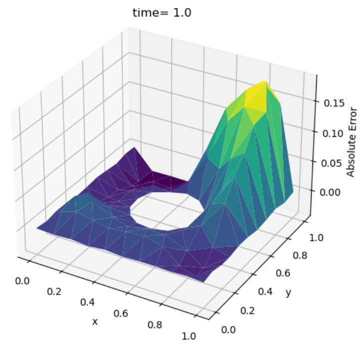


(e) Approximated result for Case C; nodes=420, collocations=658

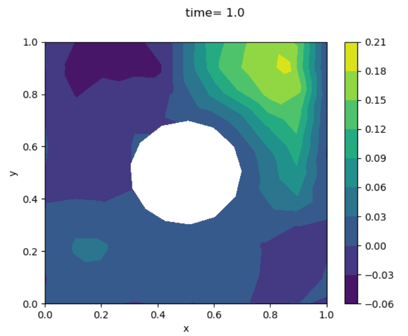


(f) Contour plot for Case C; nodes=420, collocations=658

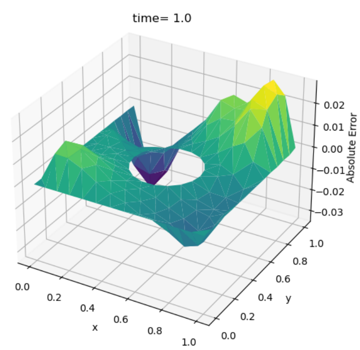
Figure 9: Approximated result and contour plot at $T = 1$ for different nodes and collocation points; $\alpha = 1.50$ and $\Delta t = 0.05$



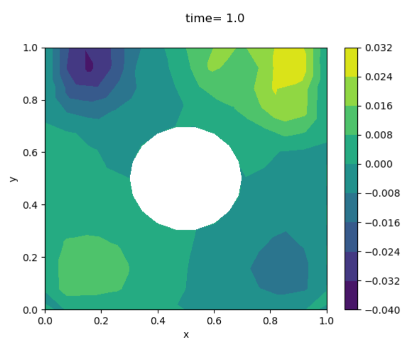
(a) Absolute error for Case A; nodes=42, collocations=106



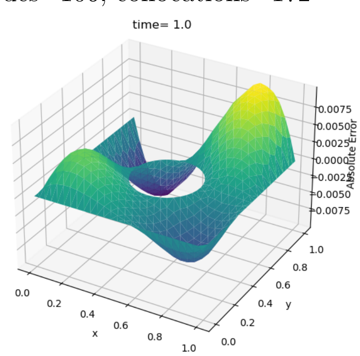
(b) Contour plot for Case A; nodes=42, collocations=106



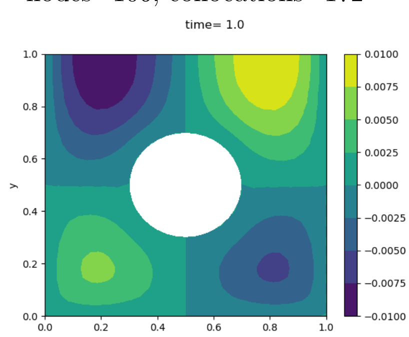
(c) Absolute error for Case B; nodes=106, collocations=172



(d) Contour plot for Case B; nodes=106, collocations=172



(e) Absolute error for Case C; nodes=420, collocations=658



(f) Contour plot for Case C; nodes=420, collocations=658

Figure 10: Absolute error and contour plot at $T = 1$ for different nodes and collocation points; $\alpha = 1.50$ and $\Delta t = 0.05$

Case	No. of Nodes	No. of collocations	L_2	L_∞	CPU time
A	42	106	$9.4163e-2$	$1.8758e-1$	1.82 s
B	106	172	$1.7614e-2$	$3.3777e-2$	4.89 s
C	420	658	$7.3304e-3$	$9.9280e-3$	19.75 s

Table 4: Approximated results of the problem using different points at $t = 1$ and $\Delta t = 0.05$.

6 Conclusion

A numerical scheme based on the CDLSM method was investigated to solve TFDWEs. We described how we construct shape functions by applying the least squares method. It is evident from the problems considered that if the number of collocation points exceeds the field point numbers, then the suggested approach yields improved results and ascertains stability. Finally, three test examples were implemented to highlight the efficiency and accuracy of the introduced method.

References

[1] H. Arzani, M. H Afshar, Solving Poisson's equations by the discrete least square meshless method, *WIT Transactions on Modelling and Simulation*, 42 (2006) 23-31.

[2] J. Barta, Über die näherungsweise Lösung einiger zweidimensionaler Elastizitätsaufgaben, *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 17, no. 3 (1937) 184-185.

[3] T. Belytschko, Y. Y. Lu, L. Gu, M. Tabbara, Element-free Galerkin methods for static and dynamic fracture, *International Journal of Solids and Structures*, 32, no. 17-18 (1995) 2547-2570.

[4] J. Chen, F. Liu, V. Anh, S. Shen, Q. Liu, C. Liao, The analytical solution and numerical solution of the fractional diffusion-wave

equation with damping, *Applied Mathematics and Computation*, 219, no. 4 (2012) 1737–1748.

[5] WS. Cleveland, *Visualizing Data*, Hobart Press, 1993.

[6] R. Darzi, B. Mohammadzade, S. Mousavi, R. Beheshti, Sumudu transform method for solving fractional differential equations and fractional diffusion-wave equation, *Journal of Mathematics and Computer Science*, 2013, no. 6 (2013) 79–84.

[7] M. Dehghan, M. Abbaszadeh, A. Mohebbi, An implicit RBF meshless approach for solving the time fractional nonlinear sine-Gordon and Klein–Gordon equations, *Engineering Analysis with Boundary Elements*, 50 (2015) 412-434.

[8] C. A. Duarte, J. T. Oden, A new meshless method to solve Boundary-Value Problems. *In Proceedings of the XVI CILAMCE-Iberian Latin American Conference on Computational methods for engineering, Curitiba, Brazil*, (1995) 90-99.

[9] Z. El Zahab, E. Divo, A. J. Kassab, A localized collocation meshless method (LCMM) for incompressible flows CFD modeling with applications to transient hemodynamics, *Engineering analysis with boundary elements*, 33, no. 8-9 (2009) 1045-1061.

[10] A. Esen, O. Tasbozan, Y. Ucar, N. M. Yagmurlu, A B-spline collocation method for solving fractional diffusion and fractional diffusion-wave equations, *Tbilisi Mathematical Journal*, 8, no. 2 (2015) 181–193.

[11] A. R. Firoozjaee, M. H. Afshar, Steady-state solution of incompressible Navier–Stokes equations using discrete least-squares meshless method, *International Journal for Numerical Methods in Fluids*, 67, no. 3 (2011) 369-382.

[12] R. A. Frazer, W. P. Jones, S. W.Skan, Approximations to Functions and to the Solutions of Differential Equations, *Britain Aerospace Research Council London*. Report and Memo 1799 (1937).

[13] V. R. Hosseini, E. Shivanian, W. Chen, Local radial point interpolation (MLRPI) method for solving time fractional diffusion-wave equation with damping, *Journal of Computational Physics*, 312 (2016) 307–332.

[14] J. Huang, Y. Tang, L. V azquez, J. Yang, Two finite difference schemes for time fractional diffusion-wave equation, *Numerical Algorithms*, 64, no. 4 (2013) 707–720.

[15] H. Jafari, H. Tajadodi, New method for solving a class of fractional partial differential equations with applications, *Thermal Science*, 22(Suppl. 1) (2020) 277-286.

[16] H. Jafari, H. Tajadodi, and B. Dumitru, A numerical approach for fractional order Riccati differential equation using B-spline operational matrix, *Fractional Calculus and Applied Analysis*, 18, no. 2 (2015) 387-399.

[17] A. Kumar, A. Bhardwaj, A local meshless method for time fractional nonlinear diffusion wave equation, *Numerical Algorithms*, 85, no. 4 (2020) 1311-1334.

[18] P. Lancaster, K. Salkauskas, Surfaces generated by moving least squares methods, *Mathematics of computation*, 37, no. 155 (1981) 141-158.

[19] C. Lanczos, Trigonometric interpolation of empirical and analytical functions. *Journal of Mathematics and Physics*, 17, no. 1-4 (1938) 123-199.

[20] GR. Liu, *Meshfree Methods: Moving Beyond the Finite Element Method*, CRC press, 2009.

[21] GR. Liu, YT. Gu, *An Introduction to Meshfree Methods and Their Programming*, Springer Science & Business Media, 2005.

[22] GR. Liu, YT. Gu, K. Y. Dai, Assesment and applications of point interpolation methods for computational mechanics, *International Journal for Numerical methods in engineering*, 59, no. 10 (2004) 1373-1397.

[23] W. K. Liu, S. Jun, Y. F. Zhang, Reproducing kernel particle methods, *International journal for numerical methods in fluids*, 20 (1995) 1081-1106.

[24] B. F. Malidareh, S. A. Hosseini, Collocated discrete subdomain meshless method for dam-break and dam-breaching modelling, Thomas Telford Ltd, *In Proceedings of the Institution of Civil Engineers-Water Management*, 172, no.2 (2019) 68-85.

[25] B. F. Malidareh, S. A. Hosseini, J. Jabbari , Discrete mixed subdomain least squares (DMSLS) meshless method with collocation points for modeling dam-break induced flows, *Journal of Hydroinformatics*, 18, no. 4 (2016) 702-723.

[26] Z. Mao, A. Xiao, Z. Yu, L. Shi, Sinc-chebyshev collocation method for a class of fractional diffusion-wave equations, *The Scientific World Journal*, 2014 (2014).

[27] J. M. Melenk, I. Babuška, The partition of unity finite element method: basic theory and applications. *Computer methods in applied mechanics and engineering*, 139, no. 1-4 (1996): 289-314.

[28] B. Nayroles, G. Touzot, P. Villon, The diffuse elements method, *Comptes Rendus De L Academie Des Sciences Serie Ii*, 313, no. 2 (1991) 133-138.

[29] E. Oñate, S. Idelsohn, O. C. Zienkiewicz, R. L. Taylor, A finite point method in computational mechanics, Applications to convective transport and fluid flow, *International journal for numerical methods in engineering*, 39, no. 22 (1996) 3839-3866.

[30] S. S. Ray, Exact solutions for time-fractional diffusion-wave equations by decomposition method, *Physica Scripta*, 75, no. 1 (2007) 53–61.

[31] Y. Shekari, A. Tayebi, M. H. Heydari, A meshfree approach for solving 2D variable-order fractional nonlinear diffusion-wave equation, *Computer Methods in Applied Mechanics and Engineering*, 350 (2019) 154-168.

[32] J. C. Slater, Electronic energy bands in metals, *Physical Review*, 45, no. 11 (1934) 794.

[33] H. Tajadodi, A Numerical approach of fractional advection-diffusion equation with Atangana–Baleanu derivative, *Chaos, Solitons& Fractals*, 130 (2020) 109527.

[34] H. Tajadodi, A. Khan, J. Francisco Gómez-Aguilar, H. Khan, Optimal control problems with Atangana-Baleanu fractional derivative, *Optimal Control Applications and Methods*, 42, no. 1 (2021) 96-109.

[35] X. Zhang, X. H. Liu, K. Z. Song, M. W. Lu, Least-squares collocation meshless method, *International Journal for Numerical Methods in Engineering*, 51, no.9 (2001) 1089-1100.

[36] T. Zhu, S. N. Atluri, A modified collocation method and a penalty formulation for enforcing the essential boundary conditions in the element free Galerkin method, *Computational Mechanics*, 21, no. 3 (1998) 211-222.

Babak Fazli Malidareh
Department of Civil Engineering
Assistant Professor of civil engineering
Babol Branch, Islamic Azad University
Babol, Iran
E-mail: fazli.babak@baboliau.ac.ir