Received: 00 Month 0000 | Revised: 00 Month 0000 | Accepted: 00 Month 0000

DOI: xxx/xxxx

ARTICLE TYPE

A study of nonlinear systems arising in the physics of liquid crystals, using MLPG and DMLPG methods

Ali Shokri*[1] | Erfan Bahmani[2]

[1]Department of Mathematics, University of Zanjan, Zanjan, Iran
[2]Department of Mathematics, University of Zanjan, Zanjan, Iran

Correspondence
*Ali Shokri, Email: a.shokri@znu.ac.ir; shokri.a@gmail.com

Summary

The study of liquid crystals is one of the active areas of research in physics. In this paper, the MLPG and direct MLPG (DMLPG) methods are used for the numerical study of the coupled nonlinear sine-Gordon equations in two dimensions arising from the modeling of some phenomena in the liquid crystals and superconductors. A comparison of numerical results between the MLPG and DMLPG methods highlight the precision and speed of the DMLPG method.

KEYWORDS:
MLPG and DMLPG methods, coupled nonlinear sine-Gordon equations, (Generalized) moving least squares

1 | INTRODUCTION

1.1 | 2D coupled nonlinear sine-Gordon equations

Nonlinear PDEs are important in various sciences, especially mathematics, engineering, and physics. Sometimes, researchers introduce many types of physical systems that require mathematics to solve them, but it's difficult to solve PDEs analytically. Hence, They use the numerical methods to approximate solutions. One of the active areas of study in physical systems is liquid crystals. In recent decades, the study of liquid crystals has been of great importance, since the liquid crystals and their optical properties are remarkable among other optical materials [1,2,3,4].

The dynamic equation of nematic liquid crystal (NLC) molecules is the nonlinear action and reaction among the liquid crystals and the external electric field. This equation is revealed as anisotropy for different physical parameters [5]. These parameters can approximately be given the permittivity anisotropy of NLCs in the direct current (DC) with

$$\Delta\xi = 4\pi nhF\left[\Delta\alpha - \frac{\mu^2 F}{3kt}\frac{3}{2}(1-3\cos^2\beta)\right]s, \tag{1}$$

where $\Delta\xi = \xi_{\|} - \xi_{\perp}$, which $\xi_{\|}$ is the parallel dirrection and $\xi_{\perp}$ is the vertical dirrection of permittivity ξ. n is the number of moecules, $h = 3\bar{\xi}/(2\bar{\xi}+1)$, $\bar{\xi} = (2\xi_{\perp}+\xi_{\|})/3$, $F = 1/(1-f\bar{\alpha})$,$f = \frac{4}{3}\pi n(2\bar{\xi}-2)/(2\bar{x}i+1)$, $\bar{\alpha} = (\alpha_{\perp}+\alpha_{\|})$ and $\Delta\alpha = \alpha_{\|} - \alpha_{\perp}$ which $\alpha_{\|}$ and $\alpha_{\perp}$ are the polarizability parallel and vertical to the direction of the molecular long axis respectively [5].

The Hamiltonian system of action and reaction energy is as follows

$$\mathcal{H} = \sum_i \left[\frac{1}{n}f_E + E_r + V_p\right], \tag{2}$$

which f_E is the potential energy, E_r is the kinetic energy and V_p is the elastic interaction energy where

$$f_E = -\frac{1}{2}\xi_\perp E^2 - \frac{1}{2}\Delta\xi(\hat{n}.\vec{E})^2, \tag{3}$$

$$E_r = \frac{1}{2}\sum_i I_i\dot{\theta}_i^2, \tag{4}$$

$$V_p = \frac{1}{2}\sum_i \left(K(\theta_{i+1}-\theta_i)^2 + K(\theta_i-\theta_{i-1})^2\right). \tag{5}$$

For more information about parameters and coefficients see[5]. Now by using (1),(3),(4) and (5) and substituting them into (2), we will have

$$\frac{\partial^2\theta_i}{\partial t^2} - \nu^2\nabla^2\theta_i + \lambda\sin 2\theta_i = 0 \tag{6}$$

where $\lambda = 2\pi hF[\Delta\alpha + F\mu^2/(kt)]E^2/I$, $\nu^2 = Ka^2/I$ which a is the distance among two adjacent liquid crystal molecules. The equation (6) holds for any θ_i so i can be omitted and therefore we have the following 2D non-linear sine-Gordon (SG) equation

$$\frac{\partial^2\theta}{\partial t^2} - \nu^2\nabla^2\theta + \lambda\sin 2\theta = 0, \tag{7}$$

This paper focuses on the numerical study of the following 2D coupled nonlinear sine-Gordon (SG) equations which earned by 2D coupled Klein-Gordon equations

$$\begin{cases} u_{tt} - \Delta u = h_u(u,v) \\ v_{tt} - c^2\Delta v = h_v(u,v) \end{cases} \tag{8}$$

where $\Delta u = \nabla.\nabla u$. The system (8) is Lagrangian with the density

$$L = \frac{1}{2}\left(u_t^2 + v_t^2 - (\nabla u)^2 - c^2(\nabla v)^2\right) + h(u,v).$$

The above Lagrangian system shows two equal equations for the protected energy and momentum

$$\frac{\partial}{\partial t}\left[\frac{1}{2}\left(u_t^2 + v_t^2 + (\nabla u)^2 + c^2(\nabla v)^2\right) - h(u,v)\right] - \nabla\left[u_t\nabla u + c^2 v_t\nabla v\right] = 0,$$
$$\frac{\partial}{\partial t}\left[u_t\nabla u + v_t\nabla v\right] - \nabla\left[\frac{1}{2}\left(u_t^2 + v_t^2 + (\nabla u)^2 + c^2(\nabla v)^2\right) + h(u,v)\right] = 0.$$

If the potential function $h(u,v) = \cos(\sigma u - v) - 1$, we have the 2D coupled nonlinear sine-Gordon (SG) equations

$$\begin{cases} u_{tt} - \nabla^2 u = -\sigma^2\sin(u-v) + f(x,y,t), \\ v_{tt} - c^2\nabla^2 v = \sin(u-v) + g(x,y,t), \end{cases} \tag{9}$$

where functions $f, g \in c^\infty(\Omega)$ and coefficients $c, \sigma > 0$ are known. This system can describe the open conditions in deoxyribonucleic acid (DNA) and also generalize the Frenkel-Kontorova dislocation model[6,7,8]. If $\sigma \to 0$ and $u = 0$, the system (9) reduces to the sine-Gordon equation (7) for the variable v.

Khusnutdinova and Pelinovsky[9] constructed a nonlinear solution for the special case $c = 1$ and they obtained weakly nonlinear solutions for the general case by reduction of the system (9) to nonlinear Schrödinger (NLS) system.

One of the applications of the coupled SG equations is that these equations can explain flexion phenomena of stacked intrinsic Josephson junction in high-temperature superconductors[10]. Also, these equations were studied widely for two junction steaks[11], and the coupled SG equations can be applied to the soliton phenomenon and ultimately leads to the expansion of liquid crystal applications, including optical performance control[5].

Many authors tried to find the exact solutions to the coupled SG equations. For example, Ekici et al.[12] used two integration tools to find the soliton solutions to these equations. Also, Salas[13] used a special rational exponential method for this purpose.

Some of the numerical methods used for 2D coupled SG equations as follows: the Modified Decomposition method (MDM)[14,15], the RBF and RBF-QR methods[16] and the regularization method[17].

1.2 | The meshless methods, MLPG and DMLPG methods

In recent years, using meshless methods in solving problems in applied sciences and engineering has grown quickly. Also, meshless methods have been known as good and powerful tools for solving boundary value problems (BVPs).

Classical mesh-based methods have been used in many problems but these methods have some limitations. Due to difficulties in mesh generation and mesh refinement, these methods lead to higher errors, and the solution is not guaranteed to be accurate

enough for a long time for complex geometries [12]. Because of these limitations, researchers have focused on meshless methods to eliminate these constraints. Another reason for using meshless methods is to reduce the computational time because no mesh is used. Meshless methods have high accuracy because, for more refinement, the number of nodes can be easily added. Also, these methods can easily create high-order shape functions [18].

According to the classification in [19], the meshless methods are classified into three types. The first type is the meshless methods based on strong forms such as FDM. The second type is the meshless methods based on weak forms such as the point interpolation method (PIM) and the meshless local Petrov-Galerkin (MLPG) method. The third type is the meshless methods based on the combination of weak and strong forms, such as the meshless weak-strong-form (MWS) method (see [19] and references therein).

The MLPG methods, the most famous meshless methods based on weak forms, have been introduced by S.N. Atluri and collaborators [20]. The MLPG methods do not use any mesh or background cells for obtaining the interpolation functions or the integrations and because of this, MLPG methods are truly meshless methods. The most important difference between these methods and other meshless methods is that the local weak forms are produced around each node in local sub-domains, instead of using the global weak form.

In the MLPG methods, the moving least squares (MLS) shape functions used for approximating numerical integrals [21]. These shape functions are complicated and do not have closed forms. To get accurate results, numerical quadratures with many integration nodes are required. Thus the MLS subroutines must be called very often, leading to high computational costs.

As an improvement of the MLPG method based on MLS approximation, Mirzaei and Schaback [22] introduced the Direct meshless local Petrov–Galerkin (DMLPG) method that uses generalized moving least squares (GMLS) approximation instead of the MLS approximation. The GMLS approximation ignores MLS shape functions completely for integration in the MLPG method and uses basis polynomials for obtaining DMLPG integrals. Also, boundary conditions and local weak forms are approximated directly in the GMLS method [23]. Therefore, the DMLPG methods based on GMLS approximation are simpler, faster, and sometimes more accurate than the MLPG methods based on the MLS approximation.

There are many papers about the use of MLPG methods for numerical solution of PDEs such as the 2D sine-Gordon equation [24], the electric field integral equation (EFIE) [25], the generalized 2D nonlinear Schrödinger equation [26], the 3D Poisson problems [27] and so on. Some of the few PDEs that are numerically solved with the DMLPG methods are elliptic interface problems [28], the 2D time-fractional advection-diffusion equation [23], some non-linear time-dependent reaction-diffusion systems [29], the generalized Zakharov system [30] and the 2D complex Ginzburg–Landau equation [31].

A brief outline of this paper is as follows: Section 2 contains a review of the MLPG method and MLS approximation. Section 3 is dedicated to the generalized MLS approximation. Section 4 is about local weak forms of 2D coupled nonlinear sine-Gordon equations. Numerical implementation of the MLPG and the DMLPG methods on 2D coupled SG equations is given in Section 5. Section 6 is dedicated to the numerical outcomes and finally, a conclusion is given in Section 7.

2 | THE MLPG METHODS

The MLPG methods are impressive discretization tactics for numerical solutions of PDEs [32,33]. These methods approximate the numerical integrals which result from the local weak form of equations using the shape functions constructed through the moving least squares (MLS) method.

Generally, meshless methods use a local approximation to construct basis functions by the values of scattered nodes in the domain of the problem. Now, suppose Ω is the domain of the problem and Ω_s is a sub-domain of Ω in the around of a point $\mathbf{x}$.

The MLS approximation of a function $\Psi^{(\ell)}(\mathbf{x})$ over scattered nodes $X = \{\mathbf{x}_j\}_{j=1}^{N}$ in Ω_s can be written as

$$\widehat{\Psi}^{(\ell)}(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{a}^{(\ell)}(\mathbf{x}) \qquad \forall \mathbf{x} \in \Omega_s, \tag{10}$$

where $\mathbf{p}^T(\mathbf{x}) = [p_1(\mathbf{x}), p_2(\mathbf{x}), ..., p_m(\mathbf{x})]$ is a vector of complete monomial basis and a coefficient vector $\mathbf{a}^{(\ell)}(\mathbf{x}) = \{a_i^{(\ell)}(\mathbf{x})\}_{i=1}^{m}$ can be obtained from minimizing the following quadratic form

$$H^{(\ell)}(\mathbf{x}) = \sum_{j=1}^{N} w_j \left[\mathbf{p}^T(\mathbf{x}_j)\mathbf{a}^{(\ell)}(\mathbf{x}) - \widehat{\Psi}_j^{(\ell)}\right]^2 = \left[\mathbf{P}.\mathbf{a}^{(\ell)}(\mathbf{x}) - \widehat{\mathbf{\Psi}}^{(\ell)}\right]^T \mathbf{W} \left[\mathbf{P}.\mathbf{a}^{(\ell)}(\mathbf{x}) - \widehat{\mathbf{\Psi}}^{(\ell)}\right] \tag{11}$$

where N is the number of nodes in Ω_s and

$$\mathbf{P}^T = \left[\mathbf{p}(\mathbf{x}_1), \mathbf{p}(\mathbf{x}_2), \dots, \mathbf{p}(\mathbf{x}_N)\right]_{m\times N}^{T},$$

$$\mathbf{W} = \text{diag}\Big[w_1(\mathbf{x}), w_2(\mathbf{x}), \dots, w_N(\mathbf{x})\Big]_{N\times N}.$$

Finding a stationary point of H respect to $\mathbf{a}^{(\ell)}(\mathbf{x})$ leads to

$$B(\mathbf{x})\mathbf{a}^{(\ell)}(\mathbf{x}) = C(\mathbf{x})\widehat{\mathbf{\Psi}}^{(\ell)}, \tag{12}$$

whereby (12) is a linear relationship between $\widehat{\mathbf{\Psi}}^{(\ell)}$ and $\mathbf{a}^{(\ell)}(\mathbf{x})$, where

$$B(\mathbf{x}) = \mathbf{P}^T\mathbf{W}\mathbf{P} = \sum_{j=1}^{N} \mathbf{p}(\mathbf{x}_j)w_j(\mathbf{x})\mathbf{p}^T(\mathbf{x}_j), \tag{13}$$

and

$$C(\mathbf{x}) = \mathbf{P}^T\mathbf{W} = \left[\mathbf{p}(\mathbf{x}_1)w_1(\mathbf{x}), \mathbf{p}(\mathbf{x}_2)w_2(\mathbf{x}), \dots, \mathbf{p}(\mathbf{x}_N)w_N(\mathbf{x})\right]. \tag{14}$$

If the points $\{\mathbf{x}_j\}_1^N$ are separate then $\mathbf{P}$ is full rank and therefore the matrix B is non-singular and the MLS approximation is well-defined.

By computing $\mathbf{a}^{(\ell)}(\mathbf{x})$ from (12) and put it into (10), we will have

$$\widehat{\Psi}^{(\ell)}(\mathbf{x}) = \mathbf{\Phi}^T(\mathbf{x}).\widehat{\mathbf{\Psi}}^{(\ell)} = \sum_{j=1}^{N} \phi_j(\mathbf{x})\widehat{\Psi}_j^{(\ell)}, \qquad \mathbf{x}\in\Omega_{\mathbf{x}}, \tag{15}$$

where

$$\mathbf{\Phi}^T(\mathbf{x}) = \mathbf{P}^T(\mathbf{x})B^{-1}(\mathbf{x})C(\mathbf{x}), \tag{16}$$

or

$$\phi_j(\mathbf{x}) = \sum_{i=1}^{m} p_i(\mathbf{x})\left\{B^{-1}(\mathbf{x})C(\mathbf{x})\right\}_{ij}. \tag{17}$$

$\phi_j(\mathbf{x})$ is named the MLS shape function respect to the nodal point $\mathbf{x}_j$.

To find the partial derivative of $\phi_j(\mathbf{x})$ respect to a variable like x we have

$$\phi_{j,h} = \sum_{i=1}^{m}\left(p_{i,h}[A^{-1}B]_{ij} + p_i[A^{-1}B_{,h} + A^{-1}_{,h}B]_{ij}\right), \tag{18}$$

where $A^{-1}_{,h} = -A^{-1}A_{,h}A^{-1}$ and $()_{,h}$ denotes $\partial()/\partial x$. Therefore

$$\widehat{\Psi}^{(\ell)}_{,h}(\mathbf{x}) = \sum_{j=1}^{N} \phi_{j,h}(\mathbf{x})\widehat{\Psi}_j^{(\ell)}, \qquad \mathbf{x}\in\Omega_x. \tag{19}$$

3 | THE GENERALIZED MLS APPROXIMATION

To describe the GMLS approximation, we will use the GMLS method as mention in[22] and[34]. The focus of the method is on finding good estimations for functional $\lambda(\Psi)$ by nodal values $\Psi(\mathbf{x}_1), \cdots \Psi(\mathbf{x}_N)$. For this purpose, we need a generalized version of MLS, adapted from[34].

Let $\Psi \in \mathbb{C}^m(\Omega)$, $m \geq 0$, and $\{\lambda_j(\Psi)\}_{j=1}^N$ are continuous linear functionals from the dual space $\mathbb{C}^m(\Omega)^*$.

Any approximation of $\lambda(\Psi)$ as $\widehat{\lambda}(\Psi)$ should be a linear combination of the $\Psi(\mathbf{x}_j)$, i.e.

$$\widehat{\lambda}(\Psi) = \sum_{j=1}^{N} a_j(\lambda)\Psi(\mathbf{x}_j), \tag{20}$$

where the coefficients a_j should be linear in λ. Also, the same as the MLS, we consider that equation (20) be exact at least for a Q-dimensional subspace $\mathcal{P} = \text{span}\{p_1, p_2, \dots, p_Q\} \subset C^m(\Omega)$ (usually $\mathcal{P}$ is the space of d-variate polynomials of degree at most m with dimension $Q = (m+d)!/m!d!$)[31], i.e.

$$\widehat{\lambda}(\Psi) = \sum_{j=1}^{N} a_j(\lambda)p(\mathbf{x}_j), \qquad \forall p\in\mathcal{P}. \tag{21}$$

The $\widehat{\lambda}(\Psi)$ can be determined as $\widehat{\lambda}(\Psi) = \lambda(p^*)$, where p^* to be found from minimizing the weighted residual functional among all $p \in \mathcal{P}$ [34]

$$\sum_{j=1}^{N} \omega_j \left[p(\mathbf{x}_j) - \Psi(\mathbf{x}_j)\right]^2. \tag{22}$$

The optimal solution $a^*(\lambda) \in \mathbb{R}^N$ can be obtained as

$$a^*(\lambda)^T = \lambda(\mathbf{p}^T)(\mathbf{P}^T\mathbf{W}\mathbf{P})^{-1}\mathbf{P}^T\mathbf{W}, \tag{23}$$

where the matrices $\mathbf{P}$ and $\mathbf{W}$ are defined in previous section and

$$\lambda(\mathbf{p}^T) = [\lambda(p_1), \dots, \lambda(p_Q)] \in \mathbb{R}^Q.$$

It is notable that in our numerical implementation in the next section the following Gaussian weight function is used

$$\omega_j(\mathbf{x}) = \begin{cases} \dfrac{\exp\left(-\epsilon^2 r^2\right) - \exp\left(-\epsilon^2 r_0^2\right)}{1 - \exp\left(-\epsilon^2 r_0^2\right)}, & 0 \le r \le r_0, \\ 0, & \textit{elsewhere} \end{cases} \tag{24}$$

where $r = \|\mathbf{x} - \mathbf{x}_j\|_2$, r_0 is the size of the support domains and $\epsilon = 1/c$ is a constant that controls the shape of the weight function. In our Matlab codes, we set $c = 0.1$ and $r_0 = 2mh$ where h is a fill distance of nodes and m is a degree of polynomial basis functions.

4 | LOCAL WEAK FORMS OF THE COUPLED SG EQUATIONS

The local weak form of equations in meshless methods like MLPG and DMLPG is created over local subdomains such as $\Omega_s \subset \bar{\Omega} = \Omega \cup \Gamma$, which is usually a small disk taken around each point in the inner nodes of the domain and the boundary nodes. These subdomains could have different geometric shapes and sizes [22], that are considered circular in this paper.

For each $\mathbf{x} \in \Omega_s$ the local weak forms of the equations (9) can be written as

$$\int_{\Omega_s} \left(\frac{\partial^2 u}{\partial t^2} - \nabla^2 u\right) \delta \, d\Omega = \int_{\Omega_s} \left(- \sigma^2 \sin(u - v) + f(\mathbf{x}, t)\right) \delta \, d\Omega, \tag{25}$$

$$\int_{\Omega_s} \left(\frac{\partial^2 v}{\partial t^2} - c^2\nabla^2 v\right) \delta \, d\Omega = \int_{\Omega_s} \left(\sin(u - v) + g(\mathbf{x}, t)\right) \delta \, d\Omega, \tag{26}$$

Using the linearity of the integral and the Divergence theorem, these weak forms can be written as follows

$$\frac{\partial^2}{\partial t^2}\int_{\Omega_s} u\delta \, d\Omega - \int_{\partial\Omega_s} \frac{\partial u}{\partial n}\delta \, d\Gamma + \int_{\Omega_s} \nabla u \nabla \delta \, d\Omega = \; - \int_{\Omega_s} \sigma^2 \sin(u - v)\delta \, d\Omega + \int_{\Omega_s} f(\mathbf{x}, t)\delta \, d\Omega, \tag{27}$$

$$\frac{\partial^2}{\partial t^2}\int_{\Omega_s} v\delta \, d\Omega - c^2 \int_{\partial\Omega_s} \frac{\partial v}{\partial n}\delta \, d\Gamma + c^2 \int_{\Omega_s} \nabla v \nabla \delta \, d\Omega = \int_{\Omega_s} \sin(u - v)\delta \, d\Omega + \int_{\Omega_s} g(\mathbf{x}, t)\delta \, d\Omega, \tag{28}$$

where δ is an arbitrary test function. In (D)MLPG1, δ is chosen as Gaussian weight function (24) with this goal that all integrals over $\partial\Omega_s$ will vanish. But in (D)MLPG5, the Heaviside step function is used, and so all integrals that having a derivation of the test function will be zero [20,24,34,35,36,37].

5 | NUMERICAL IMPLEMENTATION

In the following two sections, we demonstrate the numerical implementation of the MLPG and DMLPG methods for 2D coupled nonlinear SG equations (9) with initial conditions

$$u(x, y, 0) = u_0(x, y), \qquad v(x, y, 0) = v_0(x, y), \tag{29}$$

$$u_t(x, y, 0) = l_1(x, y), \qquad v_t(x, y, 0) = l_2(x, y). \tag{30}$$

and Dirichlet boundary conditions.

5.1 | Discreized equations in the MLPG method

In the first step, consider a set of random nodes $X = \{x_j\}_{j=1}^{N}$ located on the domain of the problem Ω and its boundary $\partial\Omega$. In the second step, we have to consider a time-splitting method to handle the time derivatives in these equations. So, the following approximation can be used

$$\frac{\partial^2 u}{\partial t^2}(x) \simeq \frac{1}{(dt)^2}\left[u^{(k+1)}(x) - 2u^{(k)}(x) + u^{(k-1)}(x)\right], \tag{31}$$

where $u^{(k)} = u(x, kdt)$. Using (31), (27) and (28) can be written as

$$\alpha \int_{\Omega_s} u^{(k+1)}\delta \, d\Omega - \int_{\partial\Omega_s} \nabla u^{(k+1)}.n\delta \, d\Gamma + \int_{\Omega_s} \nabla u^{(k+1)}\nabla\delta \, d\Omega = 2\alpha \int_{\Omega_s} u^{(k)}\delta \, d\Omega - \alpha \int_{\Omega_s} u^{(k-1)}\delta \, d\Omega - \int_{\Omega_s} \sigma^2 \sin(u^{(k)} - v^{(k)})\delta \, d\Omega + \int_{\Omega_s} f(x, t^{(k+1)})\delta \, d\Omega, \tag{32}$$

$$\alpha \int_{\Omega_s} v^{(k+1)}\delta \, d\Omega - c^2 \int_{\partial\Omega_s} \nabla v^{(k+1)}.n\delta \, d\Gamma + c^2 \int_{\Omega_s} \nabla v^{(k+1)}\nabla\delta \, d\Omega = 2\alpha \int_{\Omega_s} v^{(k)}\delta \, d\Omega - \alpha \int_{\Omega_s} v^{(k-1)}\delta \, d\Omega + \int_{\Omega_s} \sin(u^{(k)} - v^{(k)})\delta \, d\Omega + \int_{\Omega_s} g(x, t^{(k+1)})\delta \, d\Omega, \tag{33}$$

where $\alpha = 1/(dt)^2$. Ω_s is a local subdomain which it is a circle at point x_j and $\partial\Omega_s$ is the boundary of Ω_s. Considering the Heaviside step function

$$\delta(x) = \begin{cases} 1, & x \in \Omega_s, \\ 0, & x \notin \Omega_s, \end{cases} \tag{34}$$

as a test function, the equations (32) and (33) are simplified as follows

$$\alpha \int_{\Omega_s} u^{(k+1)} \, d\Omega - \int_{\partial\Omega_s} \nabla u^{(k+1)}.n \, d\Gamma = 2\alpha \int_{\Omega_s} u^{(k)} \, d\Omega - \alpha \int_{\Omega_s} u^{(k-1)} \, d\Omega - \int_{\Omega_s} \sigma^2 \sin(u^{(k)} - v^{(k)}) \, d\Omega + \int_{\Omega_s} f(x, t^{(k+1)}) \, d\Omega, \tag{35}$$

$$\alpha \int_{\Omega_s} v^{(k+1)} \, d\Omega - c^2 \int_{\partial\Omega_s} \nabla v^{(k+1)}.n \, d\Gamma = 2\alpha \int_{\Omega_s} v^{(k)} \, d\Omega - \alpha \int_{\Omega_s} v^{(k-1)} \, d\Omega + \int_{\Omega_s} \sin(u^{(k)} - v^{(k)}) \, d\Omega + \int_{\Omega_s} g(x, t^{(k+1)}) \, d\Omega. \tag{36}$$

In the MLPG method by using the MLS approximations (15) and (19), quantities of $u^{(k)}$, $v^{(k)}$, $\nabla u^{(k)}$ and $\nabla v^{(k)}$ can be approximated by the nodal values $\widehat{u}_j^{(k)}$ and $\widehat{v}_j^{(k)}$ and then we can rewrite the equations (35) and (36) as follows

$$\sum_{j=1}^{N}\left[\alpha \int_{\Omega_s} \phi_j(x) \, d\Omega - \int_{\partial\Omega_s} \nabla\phi_j(x).n \, d\Gamma\right]\widehat{u}_j^{(k+1)} = 2\alpha \sum_{j=1}^{N}\left[\int_{\Omega_s} \phi_j(x) \, d\Omega\right]\widehat{u}_j^{(k)} - \alpha \sum_{j=1}^{N}\left[\int_{\Omega_s} \phi_j(x) \, d\Omega\right]\widehat{u}_j^{(k-1)} - \int_{\Omega_s} \sigma^2 \sin(u^{(k)} - v^{(k)}) \, d\Omega + \int_{\Omega_s} f(x, t^{(k+1)}) \, d\Omega, \tag{37}$$

$$\sum_{j=1}^{N}\left[\alpha \int_{\Omega_s} \phi_j(x) \, d\Omega - c^2 \int_{\partial\Omega_s} \nabla\phi_j(x).n \, d\Gamma\right]\widehat{v}_j^{(k+1)} = 2\alpha \sum_{j=1}^{N}\left[\int_{\Omega_s} \phi_j(x) \, d\Omega\right]\widehat{v}_j^{(k)} - \alpha \sum_{j=1}^{N}\left[\int_{\Omega_s} \phi_j(x) \, d\Omega\right]\widehat{v}_j^{(k-1)} + \int_{\Omega_s} \sin(u^{(k)} - v^{(k)}) \, d\Omega + \int_{\Omega_s} g(x, t^{(k+1)}) \, d\Omega. \tag{38}$$

Now, by calculating the integrals we get to the following system of linear equations

$$\begin{cases} A_1\widehat{\mathbf{u}}^{(k+1)} &= B_1\widehat{\mathbf{u}}^{(k)} + C_1\widehat{\mathbf{u}}^{(k-1)} - D_1 \sin(\mathbf{u}^{(k)} - \mathbf{v}^{(k)}) + F^{(k+1)}, \\ A_2\widehat{\mathbf{v}}^{(k+1)} &= B_2\widehat{\mathbf{v}}^{(k)} + C_2\widehat{\mathbf{v}}^{(k-1)} - D_2 \sin(\mathbf{u}^{(k)} - \mathbf{v}^{(k)}) + G^{(k+1)}, \end{cases} \tag{39}$$

where $\widehat{\mathbf{u}}^{(k)} = \left[\widehat{u}_1^{(k)}, \widehat{u}_2^{(k)}, \cdots, \widehat{u}_N^{(k)}\right]^T$ and $\widehat{\mathbf{v}}^{(k)} = \left[\widehat{v}_1^{(k)}, \widehat{v}_2^{(k)}, \cdots, \widehat{v}_N^{(k)}\right]^T$. Solving the systems (39) lead us to the numerical solutions of the u and v at each time step for nodal points.

5.2 | Discreized equations in the DMLPG method

In the beginning, like the MLPG method, consider a set of random nodes $X = \{x_j\}_{j=1}^N$ located on the domain of the problem Ω and its boundary $\partial\Omega$. By using the local weak forms (27) and (28), all integrals can be approximated using of (20):

$$\begin{aligned}
\lambda_{1,k}(\Psi) &:= \int_{\Omega_s} \Psi\delta \, d\Omega \quad \approx \quad \widehat{\lambda}_{1,k}(\Psi) = \sum_{j=1}^{N} a_{1,j}(x_k)\Psi(x_j),\\
\lambda_{2,k}(\Psi) &:= \int_{\Omega_s} \nabla\Psi.\nabla\delta \, d\Omega \quad \approx \quad \widehat{\lambda}_{2,k}(\Psi) = \sum_{j=1}^{N} a_{2,j}(x_k)\Psi(x_j),\\
\lambda_{3,k}(\Psi) &:= \int_{\partial\Omega_s} \frac{\partial\Psi}{\partial n}\delta \, d\Gamma \quad \approx \quad \widehat{\lambda}_{3,k}(\Psi) = \sum_{j=1}^{N} a_{3,j}(x_k)\Psi(x_j).
\end{aligned}$$

In DMLPG1, we used the Gaussian function (24) as a test function and therefore the functionals $\lambda_{3,k}$ are vanishes. But, by using the constant test function $\delta = 1$ in DMLPG5, the functionals $\lambda_{2,k}$ become not exist and also the functionals $\lambda_{1,k}$ have simple forms[23,38,31].

Now, using these functionals over weak forms (27) and (28), the following time-dependent systems are obtained

$$\begin{cases}
A^{(1)}\frac{\partial^2}{\partial t^2}\mathbf{u}(t) + \left(A^{(2)} - A^{(3)}\right)\mathbf{u}(t) = -D_1 \ \sin(\mathbf{u}(t) - \mathbf{v}(t)) + F,\\
B^{(1)}\frac{\partial^2}{\partial t^2}\mathbf{v}(t) + c^2\left(B^{(2)} - B^{(3)}\right)\mathbf{v}(t) = D_2 \ \sin(\mathbf{u}(t) - \mathbf{v}(t)) + G,
\end{cases} \tag{40}$$

where $\mathbf{u}(t)$ and $\mathbf{v}(t)$ are the time-dependent vectors, and D_1 and D_2 are diagonal matrices, which diagonal elements are areas of subdomains. The kth row of $A^{(\ell)}$ and $B^{(\ell)}, \ell = 1, 2, 3$ are obtained by using (23) as follow

$$a_{k,:} = b_{k,:} = \lambda_{\ell,k}(\mathbf{p}^T)(P^T W P)^{-1} P^T W,$$

where

$$\begin{aligned}
\lambda_{1,k}(\mathbf{p}) &= \Big[\int_{\Omega_s} p_1\delta \, d\Omega, \int_{\Omega_s} p_2\delta \, d\Omega, \cdots, \int_{\Omega_s} p_Q\delta \, d\Omega\Big],\\
\lambda_{2,k}(\mathbf{p}) &= \Big[\int_{\Omega_s} \nabla p_1.\nabla\delta \, d\Omega, \int_{\Omega_s} \nabla p_2.\nabla\delta \, d\Omega, \cdots, \int_{\Omega_s} \nabla p_Q.\nabla\delta \, d\Omega\Big],\\
\lambda_{3,k}(\mathbf{p}) &= \Big[\int_{\partial\Omega_s} \frac{\partial p_1}{\partial n}\delta \, d\Gamma, \int_{\partial\Omega_s} \frac{\partial p_2}{\partial n}\delta \, d\Gamma, \cdots, \int_{\partial\Omega_s} \frac{\partial p_Q}{\partial n}\delta \, d\Gamma\Big].
\end{aligned}$$

Using the time-splitting method (31) for the time-dependant ODE systems (40) we obtain

$$\begin{cases}
\Big(\alpha + A^{(2)} - A^{(3)}\Big)\mathbf{u}^{(k+1)} = 2\alpha\mathbf{u}^{(k)} + \alpha\mathbf{u}^{(k-1)} - D_1\sin(\mathbf{u}^{(k)} - \mathbf{v}^{(k)}) + F^{(k+1)},\\
\Big(\beta + c^2\left(B^{(2)} - B^{(3)}\right)\Big)\mathbf{v}^{(k+1)} = 2\beta\mathbf{v}^{(k)} + \beta\mathbf{v}^{(k-1)} + D_2\sin(\mathbf{u}^{(k)} - \mathbf{v}^{(k)}) + G^{(k+1)},
\end{cases} \tag{41}$$

where $\alpha = A^{(1)}/dt^2$ and $\beta = B^{(1)}/dt^2$. The initial conditions (30) are used for computing $\mathbf{u}^{-1}$ and $\mathbf{v}^{-1}$ at the first time step as

$$\mathbf{u}^{-1} = \mathbf{u}^1 - 2dt\mathbf{l_1}, \qquad \mathbf{v}^{-1} = \mathbf{v}^1 - 2dt\mathbf{l_2}. \tag{42}$$

TABLE 1 Maximum errors on regular nodes with $N = 121$ at $t = 1$ for example 1.

	DMLPG1		MLPG1		MQ RBF	
dt	Max_err u	Max_err v	Max_err u	Max_err v	Max_err u	Max_err v
1/20	1.67(−2)	1.39(−2)	1.67(−2)	1.39(−2)	1.34(−2)	1.93(−2)
1/80	4.33(−3)	3.93(−3)	4.34(−3)	3.93(−3)	6.77(−3)	6.48(−3)
1/320	1.06(−3)	9.90(−4)	1.06(−3)	2.88(−4)	1.78(−3)	1.97(−3)
1/1280	2.38(−4)	2.24(−4)	2.36(−4)	2.21(−4)	4.38(−4)	4.59(−4)

TABLE 2 Maximum errors on regular nodes with $N = 121$ at $t = 1$ for example 1.

	DMLPG5		MLPG5	
dt	Max_err u	Max_err v	Max_err u	Max_err v
1/20	1.63(−2)	1.36(−2)	1.65(−2)	1.38(−2)
1/80	4.02(−3)	3.64(−3)	4.10(−3)	3.70(−3)
1/320	7.53(−4)	6.99(−4)	8.04(−4)	7.40(−4)
1/1280	1.05(−4)	1.38(−4)	6.50(−5)	1.32(−4)

6 | NUMERICAL OUTCOMES

To show the suitability of the MLPG and DMLPG methods for the numerical solution of the 2D coupled SG equations (9) some computational results will provide in this section. All codes were written in Matlab© and for boundary and domain integrals a 10-point Gauss-Legendre quadrature is used.

Example 1

Consider the nonlinear sine-Gordon equations (9) in the square domain $\Omega = [0, 2] \times [0, 2]$ with the analytical solution [16]

$$\begin{cases} u(x, y, t) = t^2 \sin(x) \sin(y), \\ v(x, y, t) = t^2 \cos(x + y). \end{cases}$$

Initial and boundary conditions can be extracted from the above postulated exact solutions. For this example, we have considered two cases. In the first case, we have used regular nodes and for the second case, we have used Hammersley scattered nodes in the domain.

Case 1: Regular nodes

In this case, we consider regularly distributed nodes in the domain. Tables 1 and 2 contain maximum errors for the MLPG and DMLPG methods and RBF-QR method [16] at $t = 1$ with different time steps on 121 (11×11) uniform nodes ($h = 0.2$). By comparing the results of (D)MLPG methods with the results of the RBF-QR method in [16], it is found that the (D)MLPG methods have fewer errors.

Table 3 shows the CPU time consumed by the DMLPG and MLPG methods. As can be seen, the time consumed in the DMLPG methods is much less than the MLPG methods, and this is the main advantage of the DMLPG methods. Also, in Tables 4 and 5 , the numerical convergence rates of the MLPG and DMLPG methods are given. The results show the greater stability of the results of the DMLPG methods.

Case 2: Hammersley scattered nodes

To see the performance of the MLPG and DMLPG methods on scattered nodes, consider the Hammersley random nodes (Figure 1). Maximum errors and CPU time consumed for (D)MLPG1 and (D)MLPG5 are tabulated in Tables 6 and 7 , respectively.

TABLE 3 CPU time consumed in the MLPG and DMLPG methods for example 1.

	DMLPG		MLPG	
N	1	5	1	5
121	1.2	1.2	13.4	5.1
441	8.4	7.9	76.5	30.3
1681	73.8	71.4	446.1	191.4

TABLE 4 Numerical convergence rates for example 1.

		DMLPG1		MLPG1	
N	dt	u	v	u	v
121 ($h = 0.2$)	0.01	–	–	–	–
441 ($h = 0.1$)	$0.01/2$	1.38	1.33	1.38	1.33
1681 ($h = 0.05$)	$0.01/4$	1.41	1.41	NaN	NaN

TABLE 5 Numerical convergence rates for example 1.

		DMLPG5		MLPG5	
N	dt	u	v	u	v
121 ($h = 0.2$)	0.01	–	–	–	–
441 ($h = 0.1$)	$0.01/2$	1.28	1.22	1.30	1.24
1681 ($h = 0.05$)	$0.01/4$	1.37	1.36	NaN	NaN

The results indicate good accuracy of both methods and greater stability of DMLPG results. In Figures 2 and 3 the position density of u and v are shown for DMLPG1 and DMLPG5.

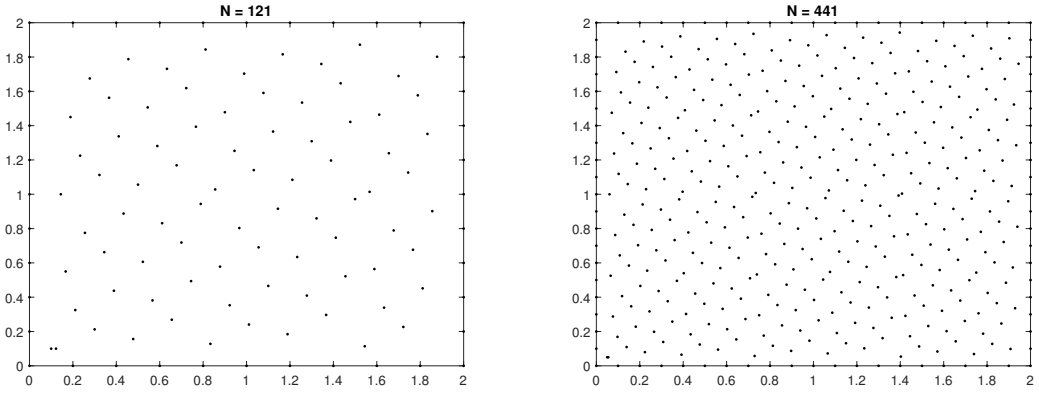

FIGURE 1 Hammersley random nodes.

TABLE 6 Maximum errors and CPU time consumed on Hammersley nodes for example 1.

		DMLPG1			MLPG1		
N	dt	Max_err u	Max_err v	CPU	Max_err u	Max_err v	CPU
36	0.01	3.14(−3)	2.75(−3)	0.3	1.76(−2)	4.98(−2)	1.5
121	0.01/2	1.74(−3)	1.63(−3)	1.8	1.76(−3)	2.01(−3)	16.8
441	0.01/4	8.92(−4)	8.32(−4)	16.2	2.10(+5)	4.70(+5)	93.3

TABLE 7 Maximum errors and CPU time consumed on Hammersley nodes for example 1.

		DMLPG5			MLPG5		
N	dt	Max_err u	Max_err v	CPU	Max_err u	Max_err v	CPU
36	0.01	1.86(−3)	1.66(−3)	0.3	3.65(−3)	4.82(−3)	0.6
121	0.01/2	1.44(−3)	1.34(−3)	1.8	1.48(−3)	1.37(−3)	5.6
441	0.01/4	8.15(−4)	7.57(−4)	15.5	4.45(+29)	8.02(+29)	37.5

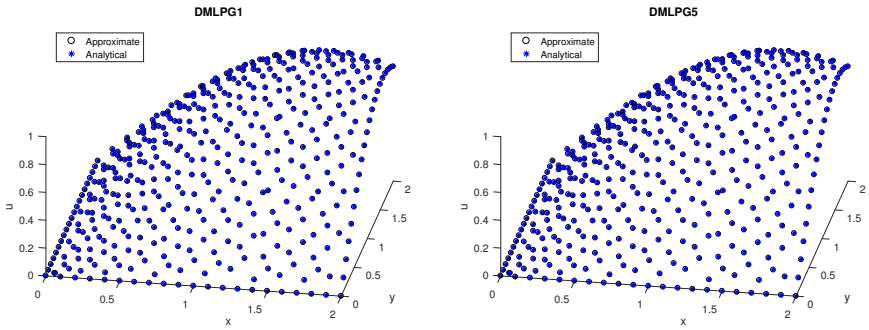

FIGURE 2 The position density of u by the DMPLG1 and DMLPG5 at $t = 1$ for example 1.

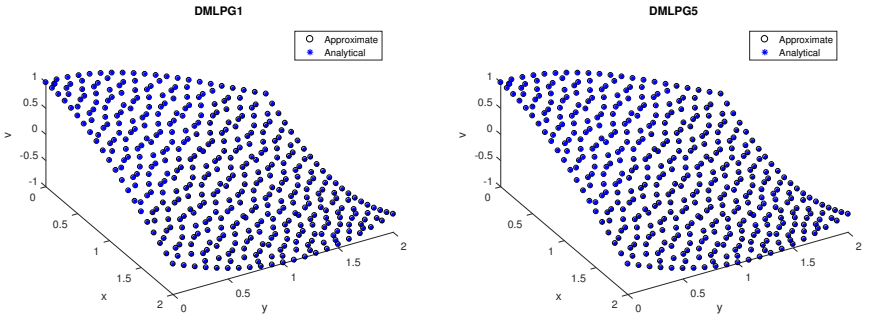

FIGURE 3 The position density of v by the DMPLG1 and DMLPG5 at $t = 1$ for example 1.

Example 2

Now consider the nonlinear sine-Gordon equations (9) with parameters $\sigma = 1$ and $c = 1$ in the square domain $\Omega = [0, 1] \times [0, 1]$ with the postulated analytical solution [16]

$$\begin{cases} u(x, y, t) = \operatorname{sech}(x + y - t), \\ v(x, y, t) = \tanh(x + y - t). \end{cases}$$

Initial and boundary conditions and right-hand side functions f and g can extract from the exact solutions. The same as the previous example, we have considered two cases of regular nodes and Hammersley random nodes.

TABLE 8 Maximum errors on regular nodes with $N = 121$ at $t = 2$ for example 2.

	DMLPG1		MLPG1		MQ RBF	
dt	Max_err u	Max_err v	Max_err u	Max_err v	Max_err u	Max_err v
1/10	3.51(−3)	2.99(−3)	3.51(−3)	3.99(−3)	2.74(−3)	6.47(−3)
1/40	1.04(−3)	9.49(−4)	1.03(−3)	9.52(−4)	1.27(−3)	1.21(−3)
1/160	3.12(−4)	2.77(−4)	3.02(−4)	2.81(−4)	4.05(−4)	4.00(−4)
1/640	9.32(−5)	6.82(−5)	8.40(−5)	7.27(−5)	1.20(−4)	1.31(−4)

TABLE 9 Maximum errors on regular nodes with $N = 121$ at $t = 2$ for example 2.

	DMLPG5		MLPG5	
dt	Max_err u	Max_err v	Max_err u	Max_err v
1/10	3.42(−3)	3.10(−3)	3.48(−3)	3.99(−3)
1/40	1.12(−3)	9.90(−4)	1.05(−3)	1.00(−3)
1/160	4.23(−4)	3.19(−4)	3.38(−4)	3.35(−4)
1/640	2.40(−4)	1.10(−4)	1.39(−4)	1.29(−4)

TABLE 10 Maximum errors and CPU time consumed on Hammersley nodes for example 2.

		DMLPG1			MLPG1		
N	dt	Max_err u	Max_err v	CPU	Max_err u	Max_err v	CPU
36	0.01	2.73(−2)	2.36(−2)	0.9	1.83(+1)	2.59(+2)	1.9
121	0.01/2	1.72(−3)	1.66(−3)	5.8	7.54(−2)	6.59(−2)	16.5
441	0.01/4	5.77(−4)	8.28(−4)	48.3	3.26(+22)	7.17(+22)	117.4

Case1: Regular nodes

In this case, we consider the regular nodes in the domain. Maximum errors for (D)MLPG1 and (D)MLPG5 methods with $N = 121$ $h = 0.1$ and at $t = 2$ for different time steps are given in Table 8 and Table 9 . The results show a similar accuracy of both methods. Also, the accuracy of (D)MLPG methods can be compared with the QR RBF method, which is presented in [16]. This comparison shows better accuracy of the (D)MLPG methods.

The plot of L_∞ errors of MLPG and DMLPG methods for $h = 0.2,\ 0.1$ and 0.05 are given in Figures 4 and 5 . As can be seen in these figures, in $h = 0.05$, the MLPG1 and MLPG5 are divergent.

Case 2: Hammersley scattered nodes

Again to see the performance of the MLPG and DMLPG methods on scattered nodes, consider the Hammersley random nodes with $N = 36,\ 121$ and 441 nodes. Maximum errors and CPU time consumed for (D)MLPG1 and (D)MLPG5 are tabulated in Tables 10 and 11 , respectively. The results show that the DMLPG methods are more accurate and the MLPG methods are divergent for $N = 441$. The positions density of u and v for DMLPG1 and DMLPG5 are demonstrated in Figures 6 and 7 .

7 | CONCLUSIONS

In this paper, we used the MLPG and DMLPG methods to numerically solve the 2D coupled nonlinear Sine-Gordon equations. By comparing the results from numerical examples, it can be seen that the DMLPG method is significantly faster than the MLPG method because

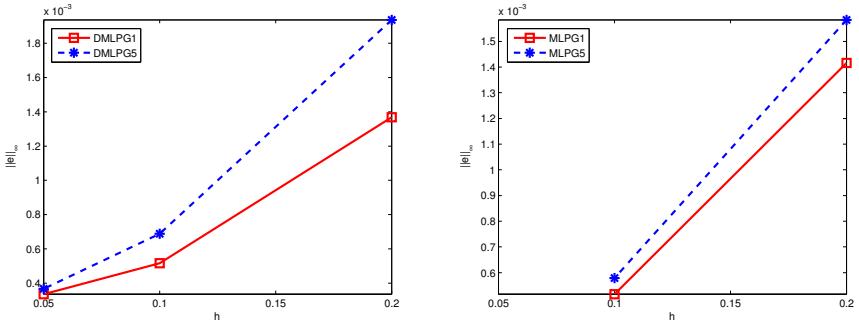

FIGURE 4 L_∞ errors for u on regular nodes at $t = 2$ for example 2.

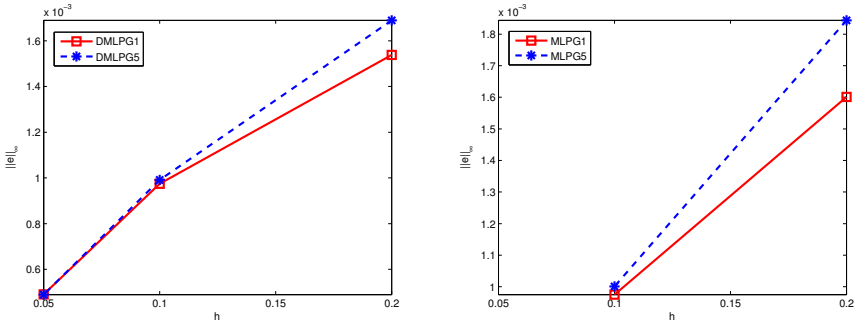

FIGURE 5 L_∞ errors for v on regular nodes at $t = 2$ for example 2.

TABLE 11 Maximum errors and CPU time consumed on Hammersley nodes for example 2.

		DMLPG5			MLPG5		
N	dt	Max_err u	Max_err v	CPU	Max_err u	Max_err v	CPU
36	0.01	1.69(−2)	1.65(−2)	0.5	5.09(−2)	4.68(−2)	1.1
121	0.01/2	1.76(−3)	2.27(−3)	5.8	3.17(−3)	2.45(−3)	9.4
441	0.01/4	6.71(−4)	9.77(−4)	47.4	5.87(+70)	2.48(+70)	68.1

1. The DMLPG methods use direct approximations to estimate boundary conditions and the local weak form of the differential equations.
2. These methods use basic polynomials instead of complex shape functions to compute numerical integrals on local subdomains.

Also by comparing the convergence rates of the two methods, it is observed that in some cases the MLPG methods are divergent while the DMLPG methods are convergent. So the DMLPG methods are numerically more stable than the MLPG methods. In addition, due to the reduced computational volume in the DMLPG methods and reduced the computational errors, the results of the DMLPG methods are more accurate than the MLPG methods. Therefore, in many cases, DMLPG methods have the potential to replace MLPG methods.

References

1. Srivatsa S.K., Ranganath. G.S.. New nonlinear optical processes in liquid crystal. *Journal of Optical Communications.* 2000;180:349-359.

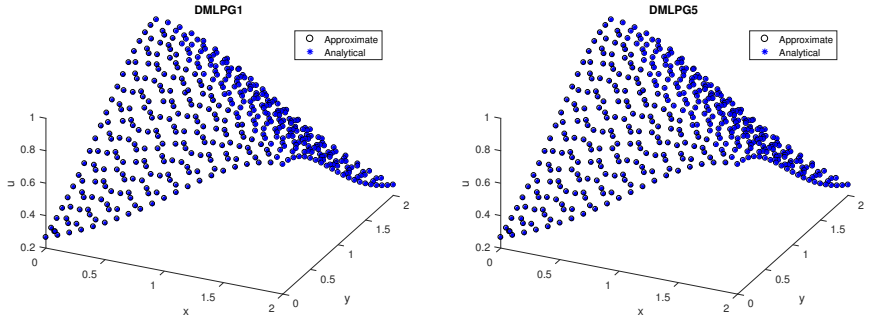

FIGURE 6 The position density of u by the DMPLG1 and DMLPG5 at $t = 2$ for example 2.

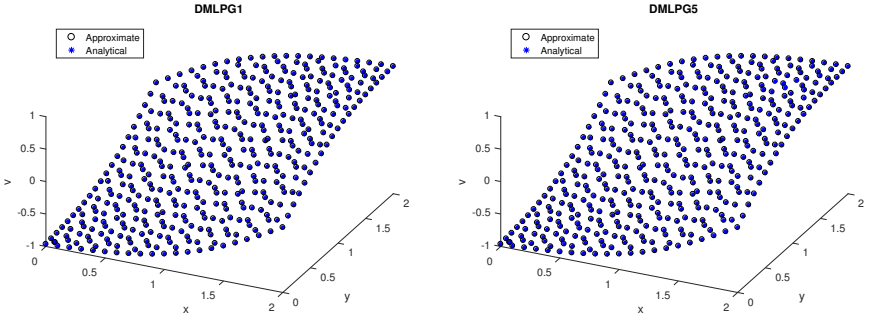

FIGURE 7 The position density of v by the DMPLG1 and DMLPG5 at $t = 2$ for example 2.

2. Madani A., Beeckmanc J., Neytsc K.. An experimental observation of a spatial optical soliton beam and self splitting of beam into two soliton beams in chiral nematic liquid crystal. *Journal of Optical Communications.* 2013;298:222-226.

3. Minzoni A.A., Sciberras L.W., Smyth N.F., Worthy A.L.. Optical vortex solitary wave in a bounded nematic-liquid-crystal cell. *Physical Review A.* 2013;87:013810.

4. Minzoni A.A., Sciberras L.W., Smyth N.F., Worthy A.L.. Elliptical optical solitary waves in a finite nematic liquid crystal cell. *Journal of Physics D.* 2015;301-302:59-73.

5. Li Y., Lu X., Hou C.. The Sine-Gordon solitons in nematic liquid crystals under the external electric field. *Results in Physics.* 2018;10:297-300.

6. Braun O.M., Kivshar Yu.S.. Nonlinear dynamics of the Frenkel-Kontorova model. *Physics Reports.* 1998;306:1-108.

7. Kontorova T.A., Frenkel Ya.I.. On the theory of plastic deformation and twinning I, II. *Journal of Experimental and Theoretical Physics.* 1938;8:89-95.

8. Yomosa S.. Soliton excitations in deoxyribonucleic acid (DNA) double helices. *Physical Review A.* 1983;27:2120-2125.

9. Khusnutdinova K.R., Pelinovsky D.E.. On the exchange of energy in coupled Klein-Gordon equations. *Wave Motion.* 2003;38:1-10.

10. Sakai S., Bodin P., Pedersen N.F.. Fluxons in thin-film superconductor-insulator superlattices. *Journal of Applied Physics.* 1993;73:2411-2418.

11. Ustinov A.V.. Solitons in Josephson junctions. *Physica D.* 1998;123:315-329.

12. Ekici M., Zhou Q., Sonmezoglu A., Mirzazadeh M.. Exact solitons of the coupled sine-Gordon equation in nonlinear system. *Optik.* 2017;136:435-444.

13. Salas A.H.. Exact solutions of coupled sine-Gordon equations. *Nonlinear Analysis: Real World Applications.* 2010;11:3930-3935.

14. Ray S. Saha. A numerical solution of the coupled sine-Gordon equation using the modified decomposition method. *Physical Review A*. 2006;175:1046-1054.

15. Kaya D.. A numerical solution of the sine-Gordon equation using the modified decomposition method. *Applied Mathematics and Computation.* 2003;143:309-317.

16. Ilati M., Dehghan M.. The use of radial basis functions (RBFs) collocation and RBF-QR methods for solving the coupled nonlinear sine-Gordon equations. *Engineering Analysis with Boundary Elements.* 2015;52:99-109.

17. Khoaa V.A., Truongb M.T.N., Duyc N.H.M, Tuand N.H.. The Cauchy problem of coupled elliptic sine-Gordon equations with noise: Analysis of a general kernel-based regularization and reliable tools of computing. *Computers and Mathematics with Applications.* 2017;73:141-162.

18. Dai B., Zheng B., Liang Q., Wang L.. Numerical solution of transient heat conduction problems using improved meshless local Petrov-Galerkin method. *Applied Mathematics and Computation.* 2013;219:10044-10052.

19. Khankham S., Luadsong A., Aschariyaphotha N.. MLPG method based on moving kriging interpolation for solving convection-diffusion equations with integral condition. *Journal of King Saud University Science.* 2015;27:292-301.

20. Atluri S.N.. *The Meshless Method (MLPG) for Domain and BIE Discretizations*. Tech Science Press; 2004.

21. Salehi R., Dehghan M.. A moving least square reproducing polynomial meshless method. *Applied Numerical Mathematics.* 2013;69:34-58.

22. Mirzaei D., Schaback R.. Direct meshless local petrov-galerkin (dmlpg) method: A generalized mls approximation. *Applied Numerical Mathematics.* 2013;68:73-82.

23. Ramezani M., Mojtabaei M., Mirzaei D.. DMLPG solution of the fractional advection-diffusion problem. *Engineering Analysis with Boundary Elements.* 2015;59:36-42.

24. Mirzaei D., Dehghan M.. Meshless local Petrov-Galerkin (MLPG) approximation to the two dimensional sine-Gordon equation. *Journal of Computational and Applied Mathematics.* 2010;233:2737-2754.

25. Honarbakhsh B.. Numerical solution of EFIE using MLPG methods. *Engineering Analysis with Boundary Elements.* 2017;80:199-217.

26. Dehghan M., Mirzaei D.. The meshless local Petrov-Galerkin (MLPG) method for the generalized two-dimensional non-linear Schrodinger equation. *Engineering Analysis with Boundary Elements.* 2008;32:747-756.

27. Sartoretto F., Mazzia A., Pini G.. The dmlpg meshless technique for poisson problems. *Applied Mathematical Sciences.* 2014;164:8233-8250.

28. A.Taleei , Dehghan M.. Direct meshless local Petrov-Galerkin method for elliptic interface problems with applications in electrostatic and elastostatic. *Computer Methods in Applied Mechanics and Engineering.* 2014;278:479-498.

29. Ilati M., Dehghan M.. Application of direct meshless local Petrov-Galerkin (dmlpg) method for some turing-type models. *Engineering with Computers.* 2017;33:107.

30. Dehghan M., Abbaszadeh M.. Numerical investigation based on direct meshless local Petrov-Galerkin (direct MLPG) method for solving generalized zakharov system in one and two dimensions and generalized gross-pitaevskii equation. *Engineering with Computers.* 2017;33:983-996.

31. Shokri A., Bahmani E.. Direct meshless local PetrovâĂŞGalerkin (DMLPG) method for 2D complex GinzburgâĂŞLandau equation. *Engineering Analysis with Boundary Elements.* 2019;100:195-203.

32. Atluri S.N., Shen S.. The meshless local Petrov-Galerkin (MLPG) method: a simple less-costly alternate to the finite element and boundary element methods. *Computer Modeling in Engineering and Sciences.* 2002;3:11-51.

33. Honarbakhsh B., Tavakoli A.. The meshless local boundary equation method. *The Applied Computational Electromagnetics Society.* 2012;27:550-559.

34. Mirzaei D., Schaback R., Dehghan M.. On generalized moving least squares and diffuse derivatives. *IMA Journal of Numerical Analysis.* 2012;32:983-1000.

35. Mirzaei D.. Error bounds for gmls derivatives approximations of sobolev functions. *Journal of Computational and Applied Mathematics.* 2016;294:93-101.

36. Atluri S.N., Zhu T.. A new meshless local petrov galerkin approach in computational mechanics. *Computational Mechanics.* 1998;22:17-27.

37. Atluri S.N., Shen S.P.. The meshless local petrov-galerkin (mlpg) method: A simple and less-costly alternative to the finite element methods. *Computer Modeling in Engineering and Sciences.* 2002;3:11-51.

38. Mirzaei D., Schaback R.. Solving heat conduction problems by the direct meshless local Petrov-Galerkin (DMLPG) method. *Numerical Algorithms.* 2014;65:275-291.