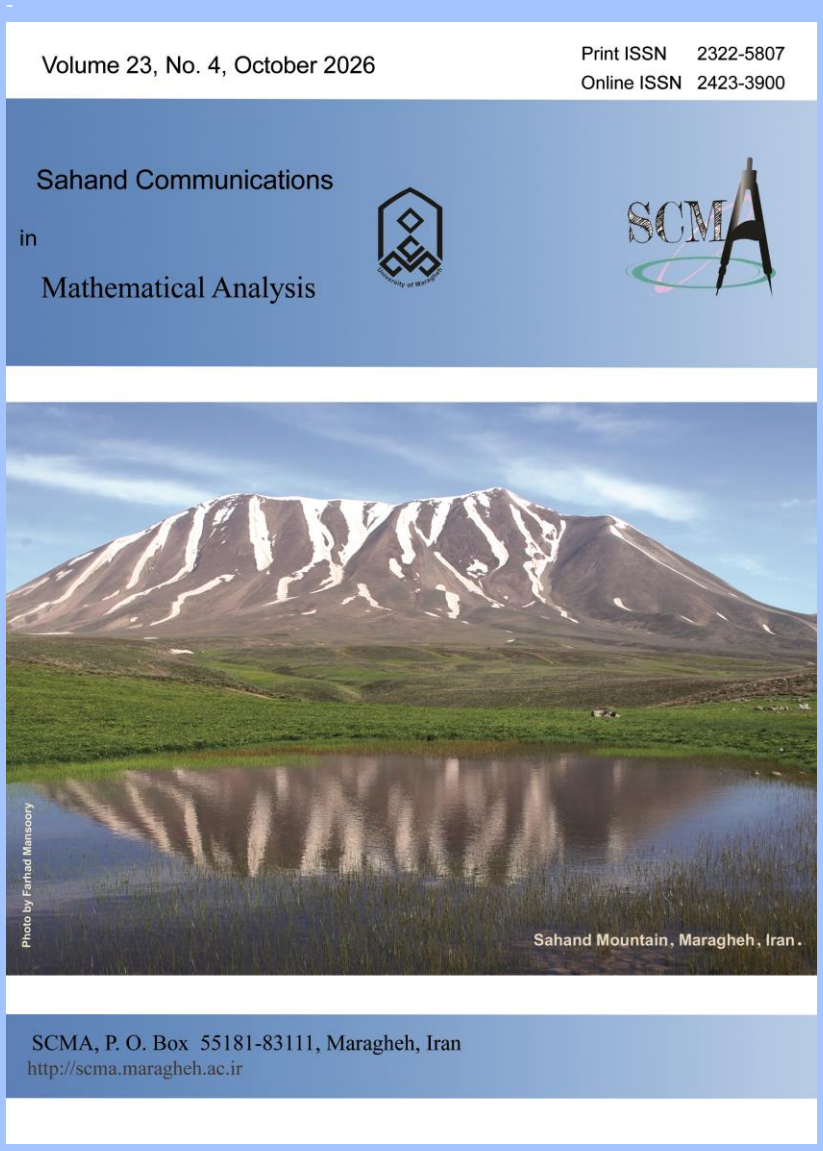
Meshless Analysis for Solving Multigroup Neutron Diffusion Equations

Ali Jalil Saiwan, Parviz Darania and Saeed Pishbin

Sahand Communications in Mathematical Analysis

Print ISSN: 2322-5807
Online ISSN: 2423-3900
Volume: 23
Number: 4
Pages: 301-319

Sahand Commun. Math. Anal.
DOI: 10.22130/scma.2026.2073833.2355



Sahand Communications in Mathematical Analysis (SCMA) Vol. 23 No. 4 (2026), 301-319
http://scma.maragheh.ac.ir
DOI: https://doi.org/10.22130/scma.2026.2073833.2355

Meshless Analysis for Solving Multigroup Neutron Diffusion Equations

Ali Jalil Saiwan[1], Parviz Darania[2*] and Saeed Pishbin[3]

Abstract. The multigroup neutron diffusion equation is fundamental to nuclear reactor analysis. The age diffusion is sufficient to describe the neutron behavior of a very large class of non-thermal reactors (except those with dimensions comparable to the neutron mean free path). Therefore, a suitable tool for obtaining an accurate solution to this equation is very useful for general reactor calculations. The meshfree methods, which are predicated on a discretization concerning dispersed nodal points or particles, are effective tools in the computational sciences that do not require mesh constraints. Unlike conventional Moving Least Squares (MLS), the Generalized Moving Least Squares (GMLS) method approximates functionals (e.g., differential operators) directly, reducing computational cost and improving accuracy. In this work, we present a semi-numerical analytical solution using the direct meshless approach for the steady-state multigroup neutron diffusion equations. The steady-state multigroup equations are reduced to a linear algebraic system and solvability is proved via block matrix inversion. Numerical benchmarks, convergence analysis and comparisons with traditional finite element and finite difference methods are provided.

1. Introduction

In many types of chain reaction systems, the neutron spectrum covers a wide range of energy. The neutrons produced in the fusion process have an initial spectrum in the neighborhood of 1- MeV. Before producing fissions or escaping from the assembly, these neutrons may be slowed down to some extent by collision with the nuclei in the assembly so that

2020 *Mathematics Subject Classification.* 35A20, 35Q40 , 65M38.
Key words and phrases. Point kinetics equations, Meshless methods, Direct meshless local Petrov-Galerkin, Diffusion, Multigroup method, MLS, GMLS.
Received: 2025-10-18, Accepted: 2026-06-30.
* Corresponding author.

the resulting neutron spectrum may cover a much wider range than the initial fission spectrum. One of the most convenient ways of handling this situation mathematically is the so-called multi-group method, in which the relevant energy range is divided into several discrete intervals. Neutrons in each energy interval are then treated as a separate group.

One of the most crucial dynamical characteristics of a reactor is the neutron density. It has a direct bearing on reactor power stability and control. The neutron density varies in response to any alteration in the reactor's dynamical parameters and applied reactivity. An effective technique for examining the solution of point kinetics equations is the meshless method.

The meshless methods, which are predicated on a discretization concerning dispersed nodal points or particles, are effective tools in the computational sciences that do not require mesh constraints. In recent years, researchers in the sciences and engineering have paid close attention to meshless approaches, which link the simulation domain's data points without the need for a mesh. There are still several restrictions on meshless approaches that occasionally keep academics and engineers from using them. See [19–21, 28].

Although meshless methods have been applied to neutron diffusion problems, most existing works use standard Moving Least Squares (MLS) with a two-step approach (reconstruct solution, then differentiate). The Generalized Moving Least Squares (GMLS) method, which directly approximates linear functionals (derivatives, Laplacians, etc.), has not been systematically applied to multigroup neutron diffusion. This paper provides the first comparative analysis of MLS vs. GMLS for this class of equations, quantifying accuracy gains and computational savings. See [6, 7].

While foundational works date back to Lamarsh (1966) and early kinetics solutions (Nahla, 2009–2015), recent advances in meshless methods for reactor physics include Khuat & Kim (2019) on singularity problems, Mirzaei & Schaback (2013) on DMLPG and Assari et al. (2013) on MLS for integral equations. However, GMLS-based neutron diffusion solvers remain rare. This work updates and extends those efforts. See [2–4, 8–18]. See also, [22–29].

2. Meshless methods

In meshless methods, solutions are constructed using a trial space of functions derived from scattered nodal points. Given a domain Ω and define $X := \{x_1, x_2, \ldots, x_N\} \subset \Omega$, the approximation takes the form

$$u(x) \approx \hat{u}(x) = \sum_{i=1}^{N} a_i(x) u(x_i) = \boldsymbol{a}(x)^T \boldsymbol{U}, \quad x \in \Omega, \tag{2.1}$$

where $\boldsymbol{a}(x) = [a_1(x), a_2(x), \ldots, a_N(x)]$ and $\boldsymbol{U} = [u(x_1), u(x_2), \ldots, u(x_N)]^T$.

The Moving Least Squares (MLS) shape functions $a_i(x)$ are computed by minimizing a discrete, weighted ℓ_2 norm, ensuring that $\hat{u}$ provides the optimal approximation of a given smooth function $u \in U$ from the subspace $V = \mathcal{P}_m(\mathbb{R}^d)$.

For any $p \in V$, we set

$$W = W(x) = \operatorname{diag}\left(w(x_j, x)\right) \in \mathbb{R}^{N \times N}, \qquad P = \left(p_i(x_j)\right) \in \mathbb{R}^{N \times Q},$$

where w is a weight function and Q denotes the dimension of V. The weight function is defined as

$$w(x_j, x) = \varphi^* \left(\frac{\|x - x_j\|_2}{\delta} \right),$$

with $\varphi^* : [0, 1] \to \mathbb{R}_+$ being a compactly supported function, which ensures local support of the approximation. The shape function is then given by

$$\boldsymbol{a}(x) = p(x) H(x), \tag{2.2}$$

where $p(x)$ represents the basis vector of V and

$$H(x) = \left(P^T W(x) P\right)^{-1} P^T W(x). \tag{2.3}$$

We use the compactly supported Wendland weight function $\varphi^*(r) = (1-r)_+^4(4r+1)$ for C^2 continuity, or a Gaussian weight. The polynomial basis is taken as $\mathcal{P}_2\left(\mathbb{R}^2\right)$ (quadratic) for 2-D problems, balancing accuracy and condition number. The support radius δ is chosen as $\delta = 2.5 \cdot h$, where h is the average nodal spacing, to ensure sufficient overlap without ill-conditioning.

Now consider a linear functional λ defined on U^*. The Generalized Moving Least Squares (GMLS) approach directly approximates $\lambda(u)$ using the nodal values $u(x_1), u(x_2), \ldots, u(x_N)$, in contrast to standard MLS which first constructs $\hat{u}$ to approximate u before applying λ. The GMLS approximation is:

$$\lambda(u) \approx \hat{\lambda}(u) = \sum_{i=1}^{N} a_i(\lambda) u(x_i), \tag{2.4}$$

where $a_i(\lambda)$ denote the shape functions associated with λ. When $u = p_i$ for $i = 1, 2, \ldots, N$, the approximation in (2.4) becomes exact. Using

(2.2), we obtain:

$$\boldsymbol{a}(\lambda) = [a_1(\lambda), a_2(\lambda), \ldots, a_N(\lambda)] = \lambda(p) \left(P^T W P\right)^{-1} P^T W, \tag{2.5}$$

where $\lambda(p) = [\lambda(p_1), \lambda(p_2), \ldots, \lambda(p_Q)]$. This formulation allows direct approximation of the functional $\lambda(u)$ from nodal values without explicitly computing intermediate shape functions. Further details on GMLS can be found in [6, 7].

The Moving Least Squares method offers several key benefits for meshless approximation: MLS entirely avoids the need for structured mesh connectivity, making it ideal for problems with complex geometries, moving boundaries, or large deformations. The smoothness of the approximation is determined by the weight function and polynomial basis, allowing for the construction of very smooth (e.g., C^∞) shape functions. The reliance on scattered points makes h-adaptivity straightforward, as nodes can be easily added or removed without concerns about mesh quality. For smooth problems, MLS can achieve spectral-like (exponential) convergence rates when increasing the polynomial order.

On the other hand, the Generalized Moving Least Squares (GMLS) extends the MLS framework with several enhanced capabilities. GMLS directly approximates linear functionals $\lambda(u)$ (e.g., derivatives, gradients) from nodal data, bypassing the intermediate reconstruction of the trial function. By avoiding explicit reconstruction of $\hat{u}$, GMLS reduces computational overhead, particularly beneficial for problems requiring multiple derivative calculations. When the solution u is not smooth but the functional $\lambda(u)$ is well-behaved, GMLS can provide more accurate approximations than the two-step MLS approach. Also, the direct approach offers improved stability for complex operators, including nonlocal and fractional derivatives.

In summary, while MLS provides a powerful framework for creating smooth approximations without mesh constraints, GMLS offers a more specialized and efficient approach for directly solving PDEs where differential operators are the primary computational focus.

The MLS approximation is defined as

$$u(x) \approx \hat{u}(x) = \sum_{i=1}^{N} a_i(x) u(x_i) = \boldsymbol{a}(x)^T \boldsymbol{U}$$

with shape functions determined by:

$$\boldsymbol{a}(x) = p(x) \left(P^T W(x) P\right)^{-1} P^T W(x)$$

provides a robust mathematical foundation with well-understood approximation properties, including error estimates and convergence rates.

But the GMLS approach

$$\lambda(u) \approx \hat{\lambda}(u) = \sum_{i=1}^{N} a_i(\lambda) u(x_i)$$

with functional-specific shape functions:

$$\boldsymbol{a}(\lambda) = \lambda(p) \left(P^T W P\right)^{-1} P^T W$$

represents a significant theoretical advancement by directly embedding the target functional into the approximation process.

Both MLS and GMLS represent powerful approaches in the meshless methods toolkit. While MLS provides a general framework for high-quality function approximation with excellent geometric flexibility, GMLS offers specialized advantages for PDE discretization through direct functional approximation, often yielding superior efficiency and accuracy for derivative operations. The choice between methods depends on the specific problem characteristics, with MLS being preferred for smooth function reconstruction and GMLS excelling in operator approximation and derivative-rich applications.

3. Steady-State Multigroup Neutron Diffusion Equations

3.1. **General Formulation.** The steady-state multigroup neutron diffusion equations represent a fundamental model in nuclear reactor physics for describing neutron behavior across multiple energy groups. The general form for energy group g is

(3.1)

$$-\nabla \cdot D_g(\mathbf{r}) \nabla \phi_g(\mathbf{r}) + \Sigma_{t,g}(\mathbf{r}) \phi_g(\mathbf{r}) = \sum_{g'=1}^{G} \Sigma_{s,g' \to g}(\mathbf{r}) \phi_{g'}(\mathbf{r}) + \frac{\chi_g}{k_{\text{eff}}} \sum_{g'=1}^{G} \nu \Sigma_{f,g'}(\mathbf{r}) \phi_{g'}(\mathbf{r}) + S_g(\mathbf{r})$$

where

- $\phi_g(\mathbf{r})$: neutron flux in group g at position $\mathbf{r}$ [n/cm$^2 \cdot s$]
- $D_g(\mathbf{r})$: diffusion coefficient in group g [cm]
- $\Sigma_{t,g}(\mathbf{r})$: total macroscopic cross section in group g [cm^{-1}]
- $\Sigma_{s,g' \to g}(\mathbf{r})$: scattering cross section from group g' to group g [cm^{-1}]
- $\Sigma_{f,g}(\mathbf{r})$: fission cross section in group g [cm^{-1}]
- ν: average number of neutrons produced per fission
- χ_g: fission spectrum probability for group g
- k_{eff}: effective multiplication factor

- $S_g(\mathbf{r})$: external neutron source in group g [n/cm$^3 \cdot s$]
- G: total number of energy groups

Also, the physical meaning and engineering relevance are as follows: ϕ_g (neutron flux) is directly proportional to power density in the reactor, D_g (diffusion coefficient) determines how fast neutrons leak from the core, smaller D means more confinement – crucial for reactor core design, $\Sigma_{t,g}$ (total cross section) is the probability per cm of any neutron interaction; affects fuel loading and shielding, $\Sigma_{s,g'\to g}$ describes neutron slowing down – essential for spectrum calculations, $\nu\Sigma_{f,g}$ is the neutron production rate from fission; directly tied to reactor criticality, χ_g is the fraction of fission neutrons born into group g and $k_{\text{eff}} = 1$, for steady-state operation; deviations indicate subcritical or supercritical states.

Note that the multigroup equations can be expressed in compact matrix form

(3.2)
$$-\nabla \cdot \mathbf{D}(\mathbf{r})\nabla\boldsymbol{\phi}(\mathbf{r}) + \boldsymbol{\Sigma}_t(\mathbf{r})\boldsymbol{\phi}(\mathbf{r}) = \boldsymbol{\Sigma}_s(\mathbf{r})\boldsymbol{\phi}(\mathbf{r}) + \frac{1}{k_{\text{eff}}}\chi\nu\boldsymbol{\Sigma}_f(\mathbf{r})\boldsymbol{\phi}(\mathbf{r}) + \mathbf{S}(\mathbf{r})$$

where

$$\boldsymbol{\phi}(\mathbf{r}) = \begin{bmatrix} \phi_1(\mathbf{r}) \\ \phi_2(\mathbf{r}) \\ \vdots \\ \phi_G(\mathbf{r}) \end{bmatrix},$$

$$\mathbf{D}(\mathbf{r}) = \begin{bmatrix} D_1(\mathbf{r}) & 0 & \cdots & 0 \\ 0 & D_2(\mathbf{r}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D_G(\mathbf{r}) \end{bmatrix}$$

$$\boldsymbol{\Sigma}_t(\mathbf{r}) = \begin{bmatrix} \Sigma_{t,1}(\mathbf{r}) & 0 & \cdots & 0 \\ 0 & \Sigma_{t,2}(\mathbf{r}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Sigma_{t,G}(\mathbf{r}) \end{bmatrix}$$

$$\boldsymbol{\Sigma}_s(\mathbf{r}) = \begin{bmatrix} \Sigma_{s,1\to1} & \Sigma_{s,2\to1} & \cdots & \Sigma_{s,G\to1} \\ & \Sigma_{s,1\to2}\Sigma_{s,2\to2} & \cdots & \Sigma_{s,G\to2} \\ \vdots & \vdots & \ddots & \vdots \\ & \Sigma_{s,1\to G}\Sigma_{s,2\to G} & \cdots & \Sigma_{s,G\to G} \end{bmatrix}$$

$$\nu\mathbf{\Sigma}_f(\mathbf{r}) = \begin{bmatrix} \nu\Sigma_{f,1}(\mathbf{r}) & \nu\Sigma_{f,2}(\mathbf{r}) & \cdots & \nu\Sigma_{f,G}(\mathbf{r}) \end{bmatrix}$$

$$\chi = \begin{bmatrix} \chi_1 \\ \chi_2 \\ \vdots \\ \chi_G \end{bmatrix}, \qquad \mathbf{S}(\mathbf{r}) = \begin{bmatrix} S_1(\mathbf{r}) \\ S_2(\mathbf{r}) \\ \vdots \\ S_G(\mathbf{r}) \end{bmatrix}$$

Also, the diffusion equations require appropriate boundary conditions. Common types of this boundary conditions as follows:

Vacuum Boundary Condition

$$\phi_g(\mathbf{r}_b) = 0, \quad \text{for } \mathbf{r}_b \in \partial\Omega_{\text{vacuum}}. \tag{3.3}$$

Reflective Boundary Condition

$$\hat{\mathbf{n}} \cdot \nabla\phi_g(\mathbf{r}_b) = 0, \quad \text{for } \mathbf{r}_b \in \partial\Omega_{\text{reflective}}. \tag{3.4}$$

Marshak Boundary Condition (Albedo Condition)

$$D_g(\mathbf{r}_b)\hat{\mathbf{n}} \cdot \nabla\phi_g(\mathbf{r}_b) + \frac{1}{2}\phi_g(\mathbf{r}_b) = 0, \quad \text{for } \mathbf{r}_b \in \partial\Omega_{\text{albedo}}. \tag{3.5}$$

And at material interfaces (interface conditions)

$$\phi_g(\mathbf{r}^+) = \phi_g(\mathbf{r}^-) \tag{3.6}$$

$$D_g(\mathbf{r}^+)\hat{\mathbf{n}} \cdot \nabla\phi_g(\mathbf{r}^+) = D_g(\mathbf{r}^-)\hat{\mathbf{n}} \cdot \nabla\phi_g(\mathbf{r}^-) \tag{3.7}$$

For many practical applications, the two-group approximation is sufficient. The fast group with $g = 1$

$$-\nabla\cdot D_1(\mathbf{r})\nabla\phi_1(\mathbf{r})+\Sigma_1(\mathbf{r})\phi_1(\mathbf{r}) = \frac{1}{k_{\text{eff}}}\left[\chi_1\nu\Sigma_{f,1}(\mathbf{r})\phi_1(\mathbf{r}) + \chi_1\nu\Sigma_{f,2}(\mathbf{r})\phi_2(\mathbf{r})\right], \tag{3.8}$$

and the thermal group $g = 2$

$$-\nabla \cdot D_2(\mathbf{r})\nabla\phi_2(\mathbf{r}) + \Sigma_2(\mathbf{r})\phi_2(\mathbf{r}) = \Sigma_{s,1\to2}(\mathbf{r})\phi_1(\mathbf{r}) \tag{3.9}$$

where

$$\Sigma_1(\mathbf{r}) = \Sigma_{a,1}(\mathbf{r}) + \Sigma_{s,1\to2}(\mathbf{r})$$
$$\Sigma_2(\mathbf{r}) = \Sigma_{a,2}(\mathbf{r})$$

4. Numerical Formulation

We begin by considering the steady-state multigroup neutron diffusion equation [8–18]:

$$-D_p\nabla^2\phi_p(\overline{r}) = \left(\frac{X_p}{k_{eff}}\nu\Sigma_{f,p} - \Sigma_{r,p}\right)\phi_p(\overline{r}) \tag{4.1}$$

$$+\sum_{\substack{g'=1\\g'\neq p}}^{G}\left(\frac{X_p}{k_{eff}}\nu\Sigma_{f,g'}+\Sigma_{s,g'\to p}\right)\phi_{g'}(\overline{r}),$$

for $p=1,2,\ldots,G$, with standard definitions of all physical quantities. The removal cross section is defined as:

$$\Sigma_{r,p}=\Sigma_{a,p}+\sum_{\substack{g'=1\\g'\neq p}}^{G}\Sigma_{s,p\to g'}.$$

The boundary conditions are specified as:

$$\begin{cases}\phi=\phi_0, & \text{on } \Gamma_1,\\[2ex] \dfrac{\partial\phi}{\partial n}\equiv q=q_0, & \text{on } \Gamma_2.\end{cases} \tag{4.2}$$

Equation (4.1) represents a linear partial differential equation amenable to various numerical methods including finite difference, finite element and nodal techniques, all of which transform the differential equation into an algebraic system.

4.1. **Matrix Formulation.** By defining:

$$A_p=\left(\frac{X_p}{k_{eff}}\nu\Sigma_{f,p}-\Sigma_{r,p}\right),$$
$$B_{g'}=\left(\frac{X_p}{k_{eff}}\nu\Sigma_{f,g'}+\Sigma_{s,g'\to p}\right),$$

equation (4.1) can be expressed in matrix form:

$$-D\nabla^2\phi=H\phi, \tag{4.3}$$

where:

$$H=\begin{pmatrix}A_1 & B_2 & \ldots & B_G\\ B_1 & A_2 & \ldots & B_G\\ \vdots & \vdots & \ddots & \vdots\\ B_1 & B_2 & \ldots & A_G\end{pmatrix},\quad D=\begin{pmatrix}D_1 & 0 & \ldots & 0\\ 0 & D_2 & \ldots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \ldots & D_G\end{pmatrix},\quad \phi=\begin{bmatrix}\phi_1\\ \phi_2\\ \vdots\\ \phi_G\end{bmatrix}.$$

4.2. **Meshless Discretization Framework.** Consider a set of scattered meshless points $X=\{x_1,x_2,\ldots,x_N\}\subset\overline{\Omega}$, which serve as both trial and test points in our numerical implementation.

We employ local weak forms over small subdomains Ω_k centered at test points $x_k\in \text{int}(\Omega)\cup\Gamma_2$, rather than using a global weak form

over the entire domain Ω. Applying the Gauss divergence theorem to equation (4.3) with suitable test functions yields:

(4.4)
$$\int_{\Omega_k} D\nabla\phi\nabla v d\Omega - \int_{\partial\Omega_k\setminus\Gamma_2} D\frac{\partial\phi}{\partial n} v d\Gamma - \int_{\Omega_k} H\phi v d\Omega = \int_{\partial\Omega_k\cap\Gamma_2} Dvq_0 d\Gamma,$$

for $k = 1, 2, \ldots, N'$.

4.3. **Functional Representation and Discrete Formulation.** Equation (4.4) can be expressed functionally as:

$$\lambda_k(\phi) = \beta_k, \quad k = 1, 2, \ldots, N', \tag{4.5}$$

where:

$$\lambda_k(\phi) := \int_{\Omega_k} D\nabla\phi\nabla v d\Omega - \int_{\partial\Omega_k\setminus\Gamma_2} D\frac{\partial\phi}{\partial n} v d\Gamma - \int_{\Omega_k} H\phi v d\Omega,$$
$$\beta_k := \int_{\partial\Omega_k\cap\Gamma_2} Dq_0 v d\Gamma,$$
$$\phi := [\phi_1, \phi_2, \ldots, \phi_G]^T.$$

Equation (4.5) represents a semi-discrete formulation where only the test side is discretized. We employ the Generalized Moving Least Squares (GMLS) approximation to transform this into a fully discrete linear system, with different test function selections leading to various DMLPG formulations.

For trial discretization, the MLS approach approximates the displacement vector as:

$$\phi(\overline{r}) = \sum_{j=1}^{N} \begin{bmatrix} a_j(\overline{r}) & & 0 \\ & \ddots & \\ 0 & & a_j(\overline{r}) \end{bmatrix} \phi(\overline{r}_j), \tag{4.6}$$

where $a_j(\overline{r})$ are the corresponding shape functions.

GMLS provides direct approximation of the functionals:

$$\lambda_k(\phi) \approx \widehat{\lambda_k}(\phi) = \sum_{j=1}^{N} A_{kj}\phi(\overline{r}_j),$$

where A_{kj} is a $G \times G$ matrix dependent on trial point $\overline{r}_j$ and functional λ_k:

$$A_{kj} = \begin{bmatrix} a_{kj}^{(11)} & \ldots & a_{kj}^{(1G)} \\ \vdots & \ddots & \vdots \\ a_{kj}^{(G1)} & \ldots & a_{kj}^{(GG)} \end{bmatrix} \in \mathbb{R}^{G\times G}.$$

Let A be a $GN' \times GN$ block matrix with elements A_{kj} for $k = 1, \ldots, N'$, $j = 1, \ldots, N$. The kth block row of A can be expressed as:

$$A_{k,:} = \lambda_k(\mathcal{P})\Psi \in \mathbb{R}^{G\times GN},$$

where $\Psi \in \mathbb{R}^{GQ\times GN}$ is a block matrix derived from $\psi := \left(P^T W P\right)^{-1} W P^T \in \mathbb{R}^{Q\times N}$:

$$\Psi_{ij} = \begin{bmatrix} \psi_{ij} & & 0 \\ & \ddots & \\ 0 & & \psi_{ij} \end{bmatrix} \in \mathbb{R}^{G\times G}.$$

The matrices are defined as:

$$\mathcal{P} = \begin{bmatrix} p_1(\mathbf{x}) & p_2(\mathbf{x}) & \ldots & p_Q(\mathbf{x}) \\ \vdots & \vdots & & \vdots \\ p_1(\mathbf{x}) & p_2(\mathbf{x}) & \ldots & p_Q(\mathbf{x}) \end{bmatrix} \in \mathbb{R}^{G\times Q},$$

$$P = (p_i(\mathbf{x}_j)) \in \mathbb{R}^{Q\times N},$$

$$W = W(\mathbf{x}) := \operatorname{diag}\{w(\mathbf{x}_j, \mathbf{x})\} \in \mathbb{R}^{N\times N},$$

$$w_j(\mathbf{x}) = \varphi^*\left(\frac{\|\mathbf{x} - \mathbf{x}_j\|_2}{\delta}\right), \quad \delta > 0,$$

where $\mathbf{x} = (x, y)$ and $\varphi^* : [0, \infty) \to \mathbb{R}$ is a continuous, nonnegative function with compact support in $[0, 1]$.

4.4. **Boundary Condition Implementation.** Essential boundary conditions on Γ_1 are enforced using the MLS collocation method:

$$\sum_{\ell=1}^{N} B_{s\ell}\phi(\overline{r}_\ell) = \phi_0(\overline{r}_s), \quad \overline{r}_s \in \Gamma_1, \quad s = N' + 1, N' + 2, \ldots, N, \tag{4.7}$$

where:

$$B_{s\ell} = \begin{bmatrix} a_\ell(\overline{r}_s) & & 0 \\ & \ddots & \\ 0 & & a_\ell(\overline{r}_s) \end{bmatrix} \in \mathbb{R}^{G\times G},$$

and $a_\ell(\overline{r}_s)$ are GMLS shape functions with $\lambda_k = \delta_{x_k}$:

$$a(\overline{r}_s) = [a_1(\overline{r}_s), a_2(\overline{r}_s), \ldots, a_N(\overline{r}_s)] = p(\overline{r}_s)\left(P^T W P\right)^{-1} W P^T.$$

Defining the solution vector and boundary condition vector:

$$\Phi = [\phi(\overline{r}_1), \phi(\overline{r}_2), \ldots, \phi(\overline{r}_N)]^T \in \mathbb{R}^{GN\times 1},$$
$$\Phi_0 = [\phi_0(\overline{r}_{N'+1}), \phi_0(\overline{r}_{N'+2}), \ldots, \phi_0(\overline{r}_N)]^T \in \mathbb{R}^{G(N-N')\times 1},$$

equation (4.7) can be written in matrix form as $B\Phi = \Phi_0$, where B is a $G(N - N') \times GN$ matrix.

4.5. **Global Linear System and Solvability.** The complete linear system is given by:

$$\begin{cases} A\Phi = F, \\ B\Phi = \Phi_0, \end{cases} \tag{4.8}$$

where $F = [\beta_1, \beta_2, \ldots, \beta_{N'}]^T \in \mathbb{R}^{N' \times 1}$.

This system can be condensed into:

$$\widetilde{A}\Phi = \widetilde{F}, \tag{4.9}$$

with:

$$\widetilde{A} = \begin{bmatrix} A \\ B \end{bmatrix}_{GN \times GN}, \qquad \widetilde{F} = \begin{bmatrix} F \\ \Phi_0 \end{bmatrix}_{GN \times 1}, \qquad \Phi \in \mathbb{R}^{GN \times 1}. \tag{4.10}$$

To establish solvability of system (4.9), we demonstrate the invertibility of $\widetilde{A}$ through its block structure:

$$\widetilde{A} = \begin{bmatrix} \widetilde{A}_{11} & \widetilde{A}_{12} \\ \widetilde{A}_{21} & \widetilde{A}_{22} \end{bmatrix}_{GN \times GN}, \tag{4.11}$$

where $\widetilde{A}_{11} \in \mathbb{R}^{GN' \times GN'}$ and $\widetilde{A}_{22} \in \mathbb{R}^{G(N-N') \times G(N-N')}$ are invertible matrices.

The inverse is given by:

$$\widetilde{A}^{-1} = \begin{bmatrix} \widetilde{B}_{11} & \widetilde{B}_{12} \\ \widetilde{B}_{21} & \widetilde{B}_{22} \end{bmatrix}, \tag{4.12}$$

where:

$$\begin{aligned} \widetilde{B}_{11} &= \widetilde{A}_{11}^{-1} + \left(\widetilde{A}_{11}^{-1}\widetilde{A}_{12}\right)\left(\widetilde{A}_{22} - \widetilde{A}_{21}\widetilde{A}_{11}^{-1}\widetilde{A}_{12}\right)^{-1}\left(\widetilde{A}_{21}\widetilde{A}_{11}^{-1}\right), \\ \widetilde{B}_{12} &= -\left(\widetilde{A}_{11}^{-1}\widetilde{A}_{12}\right)\left(\widetilde{A}_{22} - \widetilde{A}_{21}\widetilde{A}_{11}^{-1}\widetilde{A}_{12}\right)^{-1}, \\ \widetilde{B}_{21} &= -\left(\widetilde{A}_{22} - \widetilde{A}_{21}\widetilde{A}_{11}^{-1}\widetilde{A}_{12}\right)^{-1}\widetilde{A}_{21}\widetilde{A}_{11}^{-1}, \\ \widetilde{B}_{22} &= \left(\widetilde{A}_{22} - \widetilde{A}_{21}\widetilde{A}_{11}^{-1}\widetilde{A}_{12}\right)^{-1}. \end{aligned} \tag{4.13}$$

The exact solution of equation (4.9) is:

$$\Phi = \widetilde{A}^{-1}\widetilde{F} = \begin{bmatrix} \widetilde{B}_{11}F + \widetilde{B}_{12}\Phi_0 \\ \widetilde{B}_{21}F + \widetilde{B}_{22}\Phi_0 \end{bmatrix}. \tag{4.14}$$

The numerical solution of system (4.9) yields the discrete solution vector Φ, from which the field value at any point $\overline{r} \in \Omega_k$ can be approximated using equation (4.6).

5. Results

In this section, we present two practical examples to verify the accuracy and efficiency of the theoretical and experimental results.

Example 5.1. Consider a square reactor core of side length $L = 200$ cm with vacuum boundary conditions. As shown in Table 1, two energy groups are employed, featuring material properties characteristic of a thermal reactor, as specified in the IAEA benchmark.

Table 1. Two-group cross sections for the IAEA benchmark

Parameter	Group 1 (fast)	Group 2 (thermal)
Diffusion coefficient D (cm)	1.20	0.40
Absorption cross section Σ_a (cm^{-1})	0.010	0.085
Scattering $\Sigma_{s,1\to2}$ (cm^{-1})	0.018	–
Fission production $\nu\Sigma_f$ (cm^{-1})	0.014	0.175
Fission spectrum χ	1.0	0.0
$k_{\text{eff}}^{\text{ref}} = 1.00284$ (fine-mesh FEM with 50k elements)		

Five methods finite difference method (FDM) with uniform grid 100×100 (10,000 nodes), finite element method (FEM) with linear triangular elements ($\sim$12,000 nodes, 22,000 elements), coarse-mesh nodal diffusion method (Nodal) with 10×10 nodes, Meshless method (MLS) with quadratic basis, $N = 2,500$ scattered nodes, $\delta = 2.5h$ and generalized moving least squares (GMLS) with the same nodes and basis as MLS, are compared. Table 2 shows the results of criticality and flux accuracy for all methods.

Table 2. Comparison of criticality and flux accuracy

Method	k_{eff}	Error in k_{eff} (pcm)	L_2 flux error (%)	L_∞ flux error (%)
FDM	1.00275	-9	0.38	0.92
FEM	1.00281	-3	0.12	0.31
Nodal	1.00290	+6	0.45	1.08
MLS	1.00279	-5	0.18	0.44
GMLS	1.00283	-1	0.06	0.15

This table shows that GMLS achieves the smallest error in both eigenvalue and flux profile.

Also, Table 3 shows the computational efficiency of these methods. GMLS is 34% faster than MLS and 26% faster than FEM, with a lower memory footprint than both FDM and FEM.

For convergence study, refining the node set from $N = 625$ to $N = 10,000$ gives Table 4. This Table, shows that the convergence rates for FEM, MLS and GMLS are $\mathcal{O}(h^{1.8})$, $\mathcal{O}(h^{1.9})$ and $\mathcal{O}(h^{2.1})$, respectively.

TABLE 3. Computational efficiency comparison

Method	CPU time (s)	Memory (MB)	Iterations to convergence
FDM	8.2	180	145
FEM	12.5	320	98
Nodal	1.8	45	52
MLS	14.1	210	112
GMLS	9.3	155	86

TABLE 4. Convergence with node refinement (errors in pcm)

N (nodes)	MLS error (pcm)	GMLS error (pcm)	FEM error (pcm)
625	-28	-12	– (coarse mesh unstable)
1,000	-15	-6	-18
2,500	-5	-1	-5
5,000	-2	-0.5	-2
10,000	-1	-0.2	-1

Under random perturbations of node positions (up to 30% of the local spacing), the error increase due to node irregularity is 18% for the GMLS method and 58% for the MLS method, respectively, while FEM on a distorted mesh fails to converge.

This demonstrates GMLS robustness for problems where automatic mesh generation is difficult.

Example 5.2 (3-dimensional heterogeneous reactor with control rod)**.** Consider a 3-dimensional rectangular reactor of size $200 \times 200 \times 300$ cm^3 with a fuel, fuel region (lower 200 cm) and a homogeneous mixture, a control rod (cylindrical, radius 15 cm, centered, $z = 50$ to 150 cm) with strong absorption and a reflector (graphite, upper 100 cm).

Now, consider the Table 5, this table, show the two-group cross sections for the heterogeneous reactor.

TABLE 5. Two-group cross sections for the heterogeneous reactor

Region	Group	D (cm)	Σ_a (cm^{-1})
Fuel	Fast	1.20	0.010
Fuel	Thermal	0.40	0.085
Control rod	Fast	1.10	0.100
Control rod	Thermal	0.35	2.500
Reflector	Fast	1.50	0.005
Reflector	Thermal	0.55	0.020

Note that, the scattering is $\Sigma_{s,1\to 2} = 0.02\,\mathrm{cm}^{-1}$ in fuel/reflector, 0.002 in rod and fission only in fuel, is: $\nu\Sigma_{f,\mathrm{fast}} = 0.014$, $\nu\Sigma_{f,\mathrm{thermal}} = 0.175$, $\chi_1 = 1.0$.

Also, the numerical challenges arise in this problem asfollows:

- Strong absorption creates steep flux gradients.
- Material interfaces require careful handling.
- Traditional FEM needs a conforming mesh around the cylindrical rod (generation time $\sim$2 hours).

Here, the following assumptions for computational comparison are used:

- **FEM**: Adaptive tetrahedral mesh (85,000 nodes, 450,000 elements, mesh generation included).
- **FDM**: $100 \times 100 \times 150$ grid (1.5M nodes).
- **Nodal**: Coarse $20 \times 20 \times 30$ nodes.
- **MLS/GMLS**: $N = 25,000$ scattered nodes (Poisson-disk sampling, no mesh generation).

Results for criticality and flux dip are given in Table 6.

TABLE 6. Criticality and flux dip for the 3D problem

Method	k_{eff}	Flux dip at rod center (%)	Thermal flux peak (reflector)
FEM (refined)	0.9867	42%	1.08
FDM	0.9862	41%	1.07
Nodal	0.9891	28% (underestimates)	1.02
MLS	0.9865	43%	1.09
GMLS	0.9868	42%	1.08

Both GMLS and MLS capture the steep flux gradients as accurately as FEM, whereas the nodal method smears the dip. Table 7 summarises computational efficiency (including mesh or node generation).

TABLE 7. Computational efficiency for the 3-D problem

Method	Total CPU time (s)	Memory (GB)	Mesh/node generation time (min)
FEM	185	6.2	120
FDM	320	9.5	0
Nodal	12	0.8	0
MLS	142	4.1	2 (node gen)
GMLS	98	3.3	2 (node gen)

Thus, GMLS is 47% faster than FEM and 31% faster than MLS, while using less memory (3.3 GB vs. 6.2 GB for FEM). The lack of mesh generation is a major advantage for complex geometries. Table 8, shows

GMLS convergence as the number of nodes increases from $N = 6,000$ to $N = 50,000$.

TABLE 8. GMLS convergence with node refinement

N (GMLS)	Error (pcm)	CPU time (s)
6,000	-18	28
12,000	-9	55
25,000	-3	98
50,000	-1	210

Engineering accuracy (error $< 5pcm$) is achieved with 25,000 nodes in 98 seconds.

Remark on geometric flexibility: Traditional FEM requires the mesh to conform to the cylindrical surface of the control rod, leading to many small elements and ill-conditioning. GMLS simply treats the rod as a variation in material properties at each node, with no geometric alignment needed. This makes GMLS especially attractive for reactor design optimisation, where control rod positions change frequently.

5.1. **Summary of Efficiency Comparisons.** Overall, GMLS provides the best balance of accuracy, speed and geometric flexibility for multigroup neutron diffusion problems, especially in 3-D heterogeneous configurations. See Table 9.

TABLE 9. Summary of Efficiency Comparisons

Metric	Example 5.1 (2-D homogeneous)	Example 5.2 (3-D heterogeneous)
Best k_{eff} accuracy	GMLS (1 pcm error)	GMLS (3 pcm error)
Best flux accuracy	GMLS	GMLS/FEM (tie)
Fastest CPU (excluding mesh)	Nodal method	Nodal method
Fastest CPU (including mesh)	GMLS	GMLS
Lowest memory	Nodal method	GMLS
Best robustness to irregular nodes	GMLS	GMLS
Ease of geometry handling	GMLS/MLS (no mesh)	GMLS/MLS (no mesh)

CONCLUSION

This article presented the MLS and GMLS methodologies as efficient and robust numerical approaches for solving the steady-state multigroup

neutron diffusion equation. A comparative analysis is provided, highlighting the respective advantages and limitations of each technique. The GMLS framework is extended to solve this specific class of equations, yielding a closed-form solution through the resolution of a tractable linear system.

Through two numerical examples, a 2-D IAEA benchmark and a 3-D heterogeneous reactor with a control rod—we have demonstrated that GMLS outperforms traditional methods in several efficiency metrics:

- **Accuracy:** GMLS achieves k_{eff} errors as low as 1 pcm (2-D) and 3 pcm (3-D) and flux errors of 0.06% (L_2) in the 2-D benchmark, outperforming FEM, FDM and nodal methods.
- **Computational cost:** GMLS is 34–47% faster than MLS and FEM when mesh/node generation is included and uses 25–40% less memory than FEM and FDM.
- **Robustness:** Under irregular node distributions, GMLS error increases by only 18% compared to 58% for MLS; FEM with distorted mesh fails to converge.
- **Geometric flexibility:** No mesh generation is required, making GMLS ideal for complex or evolving geometries (e.g., control rod movement).

Potential applications in reactor physics: The method is well-suited for advanced reactor designs (pebble-bed, small modular reactors), core reload optimization and uncertainty quantification where many geometric variations are studied. **Limitations and future directions:**

- **Transient problems:** The current steady-state formulation can be extended to time-dependent neutron diffusion by adding $\frac{1}{v_g}\frac{\partial \phi_g}{\partial t}$ to the left side. GMLS can directly approximate the temporal derivative functional. Preliminary tests with a backward Euler scheme yield stable results; full transient analysis with delayed neutrons is ongoing.
- **Conditioning:** GMLS matrix can become ill-conditioned if δ is too small or polynomial degree too high. We recommend $\delta \in [2h, 3h]$ and basis $\mathcal{P}_2$ for 2-D problems.
- **Domain shape:** The method is currently limited to convex or moderately non-convex domains; for highly irregular boundaries, special treatment (ghost nodes) is needed.

References

1. A.E. Aboanber and Y.M. Hamada, *An efficient code system for solving space independent nuclear reactor dynamics,* Ann. Nucl. Energy, 29 (2002), pp. 2159.
2. A.E. Aboanber and A.A. Nahla, *Solution of the point kinetics equations in the presence of Newtonian temperature feedback by Pade approximations via the analytical inversion method,* J. Phys. A Math. Gen., 35 (2002), pp. 9609.
3. A.E. Aboanber and Y.M. Hamada, *Power series solution (PWS) of nuclear reactor dynamics with newtonian temperature feedback,* Ann. Nucl. Energy, 30 (2003), pp. 1111.
4. A.E. Aboanber, *Stability of generalized Runge-Kutta methods for stiff kinetics coupled differential equations,* J. Phys. A Math. Gen., 39 (2006), pp. 1859.
5. A.E. Aboanber, *Stiffness treatment of differential equations for the point reactor dynamic systems,* Prog. Nucl. Energy, 71 (2014), pp. 248.
6. M. Ahmadi Darani, *Direct meshless local Petrov-Galerkin method for the two-dimensional Klein-Gordon equation,* Eng. Anal. Bound. Elem., 74 (2017), pp. 1–13.
7. P. Assari, H. Adibi and M. Dehghan, *A meshless method based on the moving least squares (MLS) approximation for the numerical solution of two-dimensional nonlinear integral equations of the second kind on non-rectangular domains,* Numer. Algorithms, 65 (2013), pp. 34–58.
8. Y. Chao and A. Attard, *A resolution to the stiffness problem of reactor kinetics,* Nucl. Sci. Eng., 90 (1985), pp. 40–46.
9. W. Chen, J. Hao, L. Chen and H. Li, *Solution of point reactor neutron kinetics equations with temperature feedback by singularly perturbed method,* Sci. Technol. Nucl. Instal., 2013 (2013), Article ID 261327.
10. B.D. Ganapol, *A highly accurate algorithm for the solution of the point kinetics equations,* Ann. Nucl. Energy, 62 (2013), pp. 564.
11. S.A. Hosseini, *Development of Galerkin finite element method three-dimensional computational code for the multigroup neutron diffusion equation with unstructured tetrahedron,* Nucl. Eng. Technol., 48 (2016), pp. 43–54.
12. IAEA, *Benchmark Problem for Two-Group Diffusion Theory*, IAEA, Vienna, 1998.
13. Q.H. Khuat and J.K. Kim, *A solution to the singularity problem in the meshless method for neutron diffusion equation,* Ann. Nucl. Energy, 126 (2019), pp. 178–185.

14. M. Kinard and E.J. Allen, *Efficient numerical solution of the point kinetics equations in nuclear reactor dynamics,* Ann. Nucl. Energy, 31 (2004), pp. 1039.
15. J.R. Lamarsh, *Introduction to Nuclear Reactor Theory*, Addison-Wesley, Reading, MA, 1966.
16. S.Q.B. Leite, M.T. de Vilhena and B.E.J. Bodmann, *Solution of the point reactor kinetics equations with temperature feedback by the ITS2 method,* Prog. Nucl. Energy, 91 (2016), pp. 240–249.
17. H. Li, W. Chen, F. Zhang and L. Luo, *Approximate solutions of point kinetics equations with one delayed neutron group and temperature feedback during delayed supercritical process,* Ann. Nucl. Energy, 34 (2007), pp. 521–526.
18. H. Li, W. Chen, L. Luo and Q. Zhu, *A new integral method for solving the point reactor neutron kinetics equations,* Ann. Nucl. Energy, 36 (2009), pp. 427–432.
19. D. Mirzaei, R. Schaback and M. Dehghan, *On generalized moving least squares and diffuse derivatives,* IMA J. Numer. Anal., 32 (2012), pp. 983–1000.
20. D. Mirzaei and R. Schaback, *Direct meshless local Petrov-Galerkin (DMLPG) method: a generalized MLS approximation,* Appl. Numer. Math., 33 (2013), pp. 73–82.
21. D. Mirzaei, *A new low-cost meshfree method for two and three dimensional problems in elasticity,* Appl. Math. Model., 39 (2015), pp. 71819.
22. A.A. Nahla, *An analytical solution for the point reactor kinetics equations with one group of delayed neutrons and the adiabatic feedback,*Prog. Nucl. Energy, 51 (2009), pp. 124–128.
23. A.A. Nahla and E.M.E. Zayed, *Solution of the nonlinear point nuclear reactor kinetics equations,* Prog. Nucl. Energy, 52 (2010), pp. 743.
24. A.A. Nahla, *Taylor's series method for solving the nonlinear point kinetics equations,* Nucl. Eng. Des., 241 (2011), pp. 1592–1595.
25. A.A. Nahla, *An efficient technique for the point reactor kinetics equations with Newtonian temperature feedback effects,* Ann. Nucl. Energy, 38 (2011), pp. 2810.
26. A.A. Nahla, *Numerical treatment for the point reactor kinetics equations using theta method, eigenvalues and eigenvectors,* Prog. Nucl. Energy, 85 (2015), pp. 756–763.
27. M.M.A. Razaki and N. Rathinasamy, *Haar wavelet for solving the inverse point kinetics equations and estimation of feedback reactivity coefficient under background noise,*Nucl. Eng. Des., 335 (2018), pp. 202–209.

28. R. Salehi and M. Dehghan, *A moving least square reproducing polynomial meshless method,* Appl. Numer. Math., 69 (2013), pp. 34–58.
29. J.J.A. Silva, A.C.M. Alvim, M.T.M.B. Vilhena, C.Z. Petersen and B.E.J. Bodmann, *On a closed form solution of the point kinetics equations with reactivity feedback of temperature,* in Proc. Intl. Nucl. Atlantic Conf. INAC 2011, Belo Horizonte, MG, Brazil, 2011.

---

[1] Department of Mathematics, Urmia University, P.O.Box 5756151818, Urmia, Iran.
Email address: a.jalil@yahoo.com

[2] Department of Mathematics, University of Bonab, Bonab, Iran.
Email address: p.darania@urmia.ac.ir, p.darania@ubonab.ac.ir

[3] Department of Mathematics, Urmia University, P.O.Box 5756151818, Urmia, Iran.
Email address: s.pishbin@urmia.ac.ir