Nuclear Science and Techniques (2026) 37:83
https://doi.org/10.1007/s41365-026-01888-4

# Multi-antiderivative transformation alternating iterative deep learning method for solving anisotropic scattering neutron transport equations

**Dong Liu**[1,2,3] · **Bin Zhang**[1,2] · **Qi Luo**[1,3] · **Heng Zhang**[4] · **Yong Jiang**[1,2] · **Xian-Tao Cui**[1] · **Chen Zhao**[1,2]

Received: 17 May 2024 / Revised: 24 November 2024 / Accepted: 3 January 2025 / Published online: 11 February 2026


**Abstract**
Deep learning methods have achieved significant progress in solving partial differential equations. However, when applied to the widely used anisotropic scattering neutron transport equations in reactor engineering, these encounter significant challenges. To address this issue, this study introduces a multi-antiderivative transformation alternating iterative deep learning method (M-AIM). This method transforms the integral terms of the scattering and fission sources in the transport equation into multiple antiderivative functions corresponding to the integrand, converts the differential–integral form of the transport equation into an exact differential equation, and establishes the necessary constraints for a unique solution. The M-AIM uses multiple deep neural networks to map the unknown angular flux density of transport equations and represents various forms of antiderivative functions. It constructs the corresponding weighted loss functions. By alternating iterative training with deep learning methods applied to these neural networks, the loss is reduced gradually. When the loss decreases to a preset minimum, the neural network approaches a numerical solution for both angular flux density and antiderivative functions. This paper presents a numerical verification of geometries such as flat plates and spheres. It verifies the validity of the theoretical framework and associated methods. The study contributes to the development of novel technical approaches for applying deep learning to solve anisotropic scattering neutron transport equations in reactor engineering.

**Keywords** Neutron transport equations · Anisotropic scattering · Multi-antiderivative · Alternating iteration · Deep learning

## 1 Introduction

Neutron transport equations are fundamental to nuclear reactor physics. The primary objective of solving this problem is to determine the eigenvalues and spatial distribution of the neutron angular flux densities in the reactor core. These play crucial roles in reactor engineering design, development, and operational support. The conventional methods for solving this problem include deterministic and statistical methods such as the discrete ordinates (SN) method [1], method of characteristics [2–4], and Monte Carlo method [5]. Although these have been widely used with considerable success, these have limitations in terms of accuracy, resolution ratio, and efficiency. Therefore, the improvement and enhancement of numerical methods for solving neutron transport equations remain key areas of research in the industry.

In recent years, the research on artificial intelligence and deep learning methods for solving complex partial differential equations has become popular in computational

This work was supported by the National Natural Science Foundation of China (No. 12575189).

✉ Dong Liu
493159139@qq.com

1 State Key Laboratory of Advanced Nuclear Energy Technology, Nuclear Power Institute of China, Chengdu 610213, China

2 Nuclear Energy Software and Digital Reactor Engineering Research Center of CNNC, Chengdu 610213, China

3 Committee on Science and Technology of China National Nuclear Corporation, Beijing 100045, China

4 College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

physics [6–11]. Significant breakthroughs have been achieved in various fields, such as solving thermal–hydraulic equations [12–17]. In reactor physics, numerical computation methods based on deep learning have advanced significantly [18–21]. This research provides a strong foundation for a deep-learning-based numerical solution of the neutron transport equation. It is gaining increasing attention from the industry [22–27].

However, the neutron transport equation, which is a differential-integral equation, involves multiple integrals that represent neutron scattering and fission sources depending on the energy and angular flux. The application of existing deep learning methods to solve this equation presents significant challenges. A previous study adopted a discrete approach for angular variables based on the conventional SN method [28] by constructing a loss function for machine learning that approximates and discretizes definite integral terms. However, the computational results obtained using this method have limited accuracy. Furthermore, a separate study proposed the theory of differential-order transformation [29] to solve the neutron transport equation using deep learning, thereby transforming it into a higher-order differential form to enhance accuracy. However, the method presented in this study is limited to isotropic sources.

In this paper, we propose the multi-antiderivative transformation alternating iterative deep learning method (M-AIM) to solve anisotropic scattering neutron transport equations. The integral terms are transformed into multiple exact differential equations using antiderivative functions. Neural networks represent both antiderivative and angular flux functions. The numerical solution of the neutron transport equation is approximated gradually using alternating iterations of deep learning. Additionally, the accuracy of the M-AIM using examples of typical geometric transport equations is verified.

## 2 Neutron transport equations in differential-integral format

The neutron transport equation has multiple equivalent mathematical forms. The most common of these are the differential–integral, exact integral, adjoint, and even-parity forms [1, 2, 5]. The most widely studied and applied form of the neutron transport equation is the differential integral. It is expressed as follows:

$$\frac{1}{v}\frac{\partial \phi(r,\Omega,E,t)}{\partial t} + \Omega\cdot\nabla\phi(r,\Omega,E,t)+ \Sigma_{\mathrm{t}}(r,E)\phi(r,\Omega,E,t) = Q_{\mathrm{s}} + Q_{\mathrm{f}} + Q_{\mathrm{e}}, \tag{1}$$

where $\phi(r,\Omega,E,t)$ is the angular flux at position $r$ with energy $E$ at time $t$. $\Sigma_{\mathrm{t}}$ is the total reaction cross-section, and $v$ is the neutron speed. $Q_{\mathrm{s}}, Q_{\mathrm{f}}, Q_{\mathrm{e}}$ represent the scattering, fission, and external sources, respectively.

An anisotropic scattering source [5] can be expressed by Eq. 2:

$$Q_{\mathrm{s}} = \iint \Sigma_{\mathrm{s}}(r,E')f(r,\Omega',E'\to\Omega,E)\phi(r,\Omega',E',t)\mathrm{d}\Omega'\mathrm{d}E'. \tag{2}$$

The fission source is expressed by Eq. 3:

$$Q_{\mathrm{f}} = \frac{\chi(E)}{4\pi}\iint \nu\Sigma_{\mathrm{f}}(r,E')\phi(r,\Omega',E',t)\mathrm{d}\Omega'\mathrm{d}E', \tag{3}$$

where $f(r,\Omega',E'\to\Omega,E)$ is the scattering function, $\Sigma_{\mathrm{s}}$ is the scattering cross-section, $\Sigma_{\mathrm{f}}$ is the fission cross-section, $\nu$ is the neutron production number of each fission, and $\chi(E)$ is the energy spectrum distribution of fission neutrons.

The common form of the transport equation for discrete energy groups under steady-state conditions without considering external sources is [5]

$$\begin{aligned}&\Omega\cdot\nabla\phi_g(r,\Omega) + \Sigma_{\mathrm{t},g}(r,\Omega)\phi_g(r,\Omega)\\ &= \sum_{g'=1}^{G}\int \Sigma_{\mathrm{s},g'\to g}(r,\Omega'\to\Omega)\phi_g'(r,\Omega')\mathrm{d}\Omega'\\ &+ \frac{\chi_g}{4\pi k_{\mathrm{eff}}}\sum_{g'=1}^{G}(\nu(r)\Sigma_{\mathrm{f}}(r))_g'\int \phi_g'(r,\Omega')\mathrm{d}\Omega',\end{aligned} \tag{4}$$

where $k_{\mathrm{eff}}$ is the effective multiplication factor and $g$ is the energy group index.

Simultaneously, anisotropic scattering sources (Eq. 2) are usually expanded using Legendre polynomials. The multigroup forms of their anisotropic scattering sources can be represented as:

$$\begin{aligned}Q_{\mathrm{s},\,g} &= \sum_{g'=1}^{G}\int \Sigma_{\mathrm{s},g'\to g}(r,\Omega'\to\Omega)\phi_{g'}(r,\Omega')\mathrm{d}\Omega'\\ &= \sum_{g'=1}^{G}\sum_{l=0}^{\infty}\Sigma_{\mathrm{s},l,g'\to g}(r)\sum_{m=-l}^{l}Y_l^m(\Omega)\phi_{g'}^{l,m},\end{aligned} \tag{5}$$

where the neutron flux moment is represented as $\phi_{g'}^{l,m} = \int Y_l^m(\Omega')\phi_{g'}(r,\Omega')\mathrm{d}\Omega'$ and $Y_l^m(\Omega)$ is the spherical harmonic function.

In practical applications, the specific transport equation forms are usually represented by combining the geometric form of the equation, number of energy group discretization, order of anisotropic scattering sources, and presence of external sources.

The specific forms of Eq. 4 for first-order anisotropic scattering sources in a single-group flat-plate geometry are as follows:

$$\mu\frac{\partial\phi(x,\mu)}{\partial x}+\Sigma_{\mathrm{t}}(x)\phi(x,\mu)$$
$$=\sum_{g'=1}^{G}\int_{-1}^{1}(\frac{1}{2}\Sigma_{\mathrm{s0},g'\to g}(x)+\frac{3}{2}\Sigma_{\mathrm{s1},g'\to g}(x)\mu\mu')\phi_{g'}(x,\mu')\mathrm{d}\mu'$$
$$+\frac{\chi_g}{2k_{\mathrm{eff}}}\sum_{g'=1}^{G}(\nu(x)\Sigma_{\mathrm{f}}(x))_{g'}\int_{-1}^{1}\phi(x,\mu')_{g'}\mathrm{d}\mu', \tag{6}$$

where $x$ is the slab thickness. $\mu$ is the direction cosine in Cartesian coordinates.

The specific form of Eq. 4 for first-order anisotropic scattering sources in a spherical geometry is as follows:

$$\mu\frac{\partial\phi(r,\mu)}{\partial r}+\frac{1-\mu^2}{r}\frac{\partial\phi(r,\mu)}{\partial\mu}+\Sigma_{\mathrm{t}}(r)\phi(r,\mu)$$
$$=\int_{-1}^{1}\left(\frac{1}{2}\Sigma_{\mathrm{s0}}(r)+\frac{3}{2}\Sigma_{\mathrm{s1}}(r)\mu\mu'\right)\phi(r,\mu')\mathrm{d}\mu' \tag{7}$$
$$+\frac{\nu(r)\Sigma_{\mathrm{f}}(r)}{2k_{\mathrm{eff}}}\int_{-1}^{1}\phi(r,\mu')\mathrm{d}\mu',$$

where $r$ is the geometric radius of the sphere.

Therefore, by moving the terms in Eq. 4 from right to left, various specific neutron transport equations such as Eq. 6 and Eq. 7 can be described using a unified partial differential-integral equation.

$$G\Big(r,\Omega,\phi_g(r,\Omega),\Omega\cdot\nabla\phi_g(r,\Omega),\int_{\Omega'}\phi_{g'}(r,\Omega')\mathrm{d}\Omega',$$
$$\int Y_l^m(\Omega')\phi_{g'}(r,\Omega')\mathrm{d}\Omega',k_{\mathrm{eff}}\Big)=0 \tag{8}$$

## 3 Principles of multi-antiderivative transformation alternating iterative deep learning method

The M-AIM operates based on the following principle: Mathematical methods are employed to convert all the integral terms in the neutron transport equation into their corresponding antiderivative functions. This conversion transforms the neutron transport equation from its differential–integral form into an exact differential equation. Specific methods are provided for identifying unique solutions for these transformed antiderivative functions. The unknown angular flux density function and each term of the antiderivative functions are represented by separate neural networks. Substituting these neural network functions into the transport equation, a loss function is generated based on the residuals of the equation. When necessary, we formulate additional loss functions for the constraints, boundary conditions, initial conditions, and eigenvalues. These loss functions are weighted and combined into a single unified loss function. We progressively minimize the weighted loss functions by alternating iterative training by applying the M-AIM to the neural networks. After all the loss functions reduce below a set threshold, the output from the neural networks for the angular flux density function approximates the solution to the transport equation.

This section discusses in detail the transformation methods for antiderivative functions.

The angular flux $\phi(r,\Omega,E,t)$ is a continuous function of the angular variable $\Omega$ and energy variable $E$ in Eqs. 2 and 3. The integrand $\Sigma_{\mathrm{s}}(r,E')f(r,\Omega',E'\to\Omega,E)\phi(r,\Omega',E',t)$ is also continuous. If the entire integrand is discontinuous owing to the discontinuity of $\Sigma_{\mathrm{s}}(r,E')$, the integrand is considered in segments utilizing the property of integral addition, so that the integrand of each segment is continuous. According to the principles of calculus, the continuous integrand can be expressed as an integral antiderivative function $F(r,\Omega,E,\Omega',E',t)$ related to the angular variable $\Omega'$ and energy $E'$. According to the Newton–Leibniz formula, the transformation of the integral term will be discussed in detail in the following sections.

### 3.1 Antiderivative function transformation of the scattering source term

With regard to Eq. 2, the integral of the scattering sources can be calculated as follows:

$$\int_{\Omega_0}^{\Omega_1}\Sigma_{\mathrm{s}}(r,E')f(r,\Omega',E'\to\Omega,E)\phi(r,\Omega',E',t)\mathrm{d}\Omega'$$
$$=F_{\mathrm{s},\Omega}(r,\Omega,E,\Omega_1,E',t)-F_{\mathrm{s},\Omega}(r,\Omega,E,\Omega_0,E',t), \tag{9}$$

$$F_{\mathrm{s},\Omega}(r,\Omega,E,\Omega',E',t)'$$
$$=\frac{\mathrm{d}}{\mathrm{d}\Omega'}\int_{\Omega_0}^{\Omega'}\Sigma_{\mathrm{s}}(r,E')f(r,\Omega',E'\to\Omega,E)\phi(r,\Omega',E',t)\mathrm{d}\Omega'$$
$$=\Sigma_{\mathrm{s}}(r,E')f(r,\Omega',E'\to\Omega,E)\phi(r,\Omega',E',t), \tag{10}$$

where $\Omega_1$ and $\Omega_0$ are the upper and lower bounds of the angular parameters.

According to the principle of calculus, $F_{\mathrm{s},\Omega}(r,\Omega,E,\Omega',E',t)$ is a set of functions that differ by a real number $C$ between each function [30]. If $F_{\mathrm{c,s},\Omega}(r,\Omega,E,\Omega',E',t)$ is set as a definite function, $F_{\mathrm{s},\Omega}(r,\Omega,E,\Omega',E',t)=F_{\mathrm{c,s},\Omega}(r,\Omega,E,\Omega',E',t)+C$ . When we set $C=-F_{\mathrm{c,s},\Omega}(r,\Omega,E,\Omega_0,E',t)$, a specific antiderivative function $F_{0,\mathrm{s},\Omega}(r,\Omega,E,\Omega_0,E',t)=0$ exists. It is equal to $F_{\mathrm{c,s},\Omega}(r,\Omega,E,\Omega',E',t)-F_{\mathrm{c,s},\Omega}(r,\Omega,E,\Omega_0,E',t)$. When $\Omega'=\Omega_0$ ($F_{0,\mathrm{s},\Omega}(r,\Omega,E,\Omega_0,E',t)=0$), Eq. 9 can be expressed as:

$$\int_{\Omega_0}^{\Omega_1} \Sigma_s(r,E')f(r,\Omega',E'\to\Omega,E)\phi(r,\Omega',E',t)\mathrm{d}\Omega' \\ = F_{0,s,\Omega}(r,\Omega,E,\Omega_1,E',t). \tag{11}$$

Similarly, substituting Eq. 11 into Eq. 2, the scattering source can be expressed as follows according to the principle of repeated integral:

$$Q_s = \int_{E_0}^{E_1} F_{0,s,\Omega}(r,\Omega,E,\Omega_1,E',t)\mathrm{d}E' \\ = F_{s,E}(r,\Omega,E,\Omega_1,E_1,t) - F_{s,E}(r,\Omega,E,\Omega_1,E_0,t), \tag{12}$$

$$F_{s,E}(r,\Omega,E,\Omega_1,E',t)' \\ = \frac{\mathrm{d}}{\mathrm{d}E'}\int_{E_0}^{E'} F_{0,s,\Omega}(r,\Omega,E,\Omega_1,E',t)\mathrm{d}E' \\ = F_{0,s,\Omega}(r,\Omega,E,\Omega_1,E',t), \tag{13}$$

where $E_1, E_0$ are the upper and lower bounds of the energy parameters.

Another function $F_{0,s,E}(r,\Omega,E,\Omega_1,E',t)$ exists to make $F_{0,s,E}(r,\Omega,E,\Omega_1,E_0,t)=0$ at $E'=E_0$. Under this condition,

$$Q_s = \int_{E_0}^{E_1} F_{0,s,\Omega}(r,\Omega,E,\Omega_1,E',t)\mathrm{d}E' \\ = F_{0,s,E}(r,\Omega,E,\Omega_1,E_1,t). \tag{14}$$

### 3.2 Antiderivative function transformation of the fission source term

The fission source term in Eq. 3 can be derived as:

$$\int_{\Omega_0}^{\Omega_1} \nu\Sigma_f(r,E')\phi(r,\Omega',E',t)\mathrm{d}\Omega' \\ = F_{f,\Omega}(r,\Omega_1,E',t) - F_{f,\Omega}(r,\Omega_0,E',t), \tag{15}$$

$$F_{f,\Omega}(r,\Omega',E',t)' \\ = \frac{\mathrm{d}}{\mathrm{d}\Omega'}\int_{\Omega_0}^{\Omega'} \nu\Sigma_f(r,E')\phi(r,\Omega',E',t)\mathrm{d}\Omega' \\ = \nu\Sigma_f(r,E')\phi(r,\Omega',E',t). \tag{16}$$

Similar to the scattering source term, a certain function $F_{0,f,E}(r,\Omega_1,E',t)$ exists. It is the determined constraint solution. When $\Omega'=\Omega_0$, $F_{0,f,\Omega}(r,\Omega_0,E',t)$ is equal to zero.

$$\int_{\Omega_0}^{\Omega_1} \nu\Sigma_f(r,E')\phi(r,\Omega',E',t)\mathrm{d}\Omega' = F_{0,f,\Omega}(r,\Omega_1,E',t) \tag{17}$$

The fission source is derived as follows:

$$Q_f = \frac{\chi(E)}{4\pi}\int_{E_0}^{E_1} F_{0,f,\Omega}(r,\Omega_1,E',t)\mathrm{d}E' \\ = \frac{\chi(E)}{4\pi}(F_{f,E}(r,\Omega_1,E_1,t) - F_{f,E}(r,\Omega_1,E_0,t)) \tag{18}$$

and

$$F_{f,E}(r,\Omega_1,E',t)' = \frac{\mathrm{d}}{\mathrm{d}E'}\int_{E_0}^{E'} F_{0,f,\Omega}(r,\Omega_1,E',t)\mathrm{d}E' \\ = F_{0,f,\Omega}(r,\Omega_1,E',t). \tag{19}$$

As performed earlier, a certain function $F_{0,f,E}(r,\Omega_1,E',t)$ can be determined such that $F_{0,f,E}(r,\Omega_1,E_0,t)=0$ at $E'=E_0$.

Under this condition,

$$Q_f = \frac{\chi(E)}{4\pi}\int_{E_0}^{E_1} F_{0,f,\Omega}(r,\Omega_1,E',t)\mathrm{d}E' \\ = \frac{\chi(E)}{4\pi}F_{0,f,E}(r,\Omega_1,E_1,t). \tag{20}$$

To substitute Eq. 14 and Eq. 20 into Eq. 1, the neutron transport equation can be derived as:

$$\frac{1}{v}\frac{\partial\phi(r,\Omega,E,t)}{\partial t} + \Omega\cdot\nabla\phi(r,\Omega,E,t) + \Sigma_t(r,E)\phi(r,\Omega,E,t) \\ = F_{0,s,E}(r,\Omega,E,\Omega_1,E_1,t) + \frac{\chi(E)}{4\pi}F_{0,f,E}(r,\Omega_1,E_1,t) + Q_e. \tag{21}$$

The conditions for a definite solution are as follows:

$$\Omega'=\Omega_0, F_{0,s,\Omega}(r,\Omega,E,\Omega_0,E',t)=0, F_{0,f,\Omega}(r,\Omega_0,E',t)=0, \\ E'=E_0, F_{0,s,E}(r,\Omega,E,\Omega_1,E_0,t)=0, F_{0,f,E}(r,\Omega_1,E_0,t)=0. \tag{22}$$

To summarize, by transforming the antiderivative functions of the scattering and fission terms, the neutron transport equation (previously expressed in the differential-integral form as Eq. 1) is converted into an exact differential form. This new form comprises Eqs. 10, 13, 16, 19, 21 and 22.

### 3.3 Antiderivative function transformation of the multigroup transport equations

For the multigroup neutron transport Eq. 4, the neutron angular flux moment in the scattering source terms Eq. 5 can be transformed into an antiderivative function. Owing to the separation of the angle variables $\Omega, \Omega'$, the transformation form is simplified significantly. By setting $F^{l,m}_{s,g'}(r,\Omega')$ to the antiderivative function of $Y^m_l(\Omega')\phi_{g'}(r,\Omega')$, it can be expressed as:

$$\phi_{s,g'}^{l,m} = \int_{\Omega_0}^{\Omega_1} Y_l^m(\Omega')\phi_{g'}(r,\Omega')\mathrm{d}\Omega' = F_{s,g'}^{l,m}(r,\Omega_1) - F_{s,g'}^{l,m}(r,\Omega_0), \tag{23}$$

$$F_{s,g'}^{l,m}(r,\Omega')' = \frac{\mathrm{d}}{\mathrm{d}\Omega'}\int_{\Omega_0}^{\Omega'} Y_l^m(\Omega')\phi_{g'}(r,\Omega')\mathrm{d}\Omega' = Y_l^m(\Omega')\phi_{g'}(r,\Omega'). \tag{24}$$

As observed earlier, a certain function $F_{0,s,g'}^{l,m}(r,\Omega')$ exists. When $\Omega' = \Omega_0$, $F_{0,s,g'}^{l,m}(r,\Omega_0) = 0$. Simultaneously, Eq. 25 can be established:

$$\phi_{g'}^{l,m} = \int_{\Omega_0}^{\Omega_1} Y_l^m(\Omega')\phi_{g'}(r,\Omega')\mathrm{d}\Omega' = F_{0,s,g'}^{l,m}(r,\Omega_1). \tag{25}$$

Equation 25 can be substituted into Eq. 5. The multigroup form of the scattering source can be obtained as follows:

$$Q_{s,g} = \sum_{g'=1}^{G}\sum_{l=0}^{\infty}\Sigma_{s,l,g'\to g}(r)\sum_{m=-l}^{l}Y_l^m(\Omega)F_{0,s,g'}^{l,m}(r,\Omega_1). \tag{26}$$

For the integral term $\int \phi_{g'}(r,\Omega')\mathrm{d}\Omega'$ in the fission source term Eq. 4, $F_{f,g'}(r,\Omega')$ is set as the antiderivative function of $\phi_{g'}(r,\Omega')$.

$$Q_{f,g} = \frac{\chi_g}{4\pi k_{\mathrm{eff}}}\sum_{g'=1}^{G}(\nu(r)\Sigma_f(r))_{g'}\int \phi_{g'}(r,\Omega')\mathrm{d}\Omega' = \frac{\chi_g}{4\pi k_{\mathrm{eff}}}\sum_{g'=1}^{G}(\nu(r)\Sigma_f(r))_{g'}F_{0,f,g'}(r,\Omega_1) \tag{27}$$

Under this condition,

$$F_{0,f,g'}(r,\Omega)' = \frac{\mathrm{d}}{\mathrm{d}\Omega'}\int_{\Omega_0}^{\Omega'}\phi_{g'}(r,\Omega')\mathrm{d}\Omega' = \phi_{g'}(r,\Omega'). \tag{28}$$

The constraint condition for the definite solution is $F_{0,f,g'}(r,\Omega_0) = 0$ at $\Omega' = \Omega_0$.

Therefore, substituting Eq. 26 and Eq. 27 into the multigroup neutron transport equation Eq. 4 yields

$$\begin{aligned}\Omega\cdot\nabla\phi_g(r,\Omega) + \Sigma_{t,g}(r,\Omega)\phi_g(r,\Omega) &= \sum_{g'=1}^{G}\sum_{l=0}^{\infty}\Sigma_{s,l,g'\to g}(r)\sum_{m=-l}^{l}Y_l^m(\Omega)F_{0,s,g'}^{l,m}(r,\Omega_1)\\ &+ \frac{\chi_g}{4\pi k_{\mathrm{eff}}}\sum_{g'=1}^{G}(\nu(r)\Sigma_f(r))_{g'}F_{0,f,g'}(r,\Omega_1).\end{aligned} \tag{29}$$

The definite conditions are shown as $\Omega' = \Omega_0$, $F_{0,s,g'}^{l,m}(r,\Omega_0) = 0$, $F_{0,f,g'}(r,\Omega_0) = 0$.

Thus, Eqs. 24, 28, and 29 construct the neutron transport equations in an exact differential form under multigroup conditions.

### 3.4 Transformation of the two-dimensional angular parameter

When the angular variable $\Omega$ is a two-dimensional variable, it can be expressed using the directional cosines $\mu$ and azimuthal angle $\varphi$ [1, 2, 5] in the neutron transport equation. Thus, the angular integral of the neutron angular flux can be expressed using Eq. 30:

$$\int \phi(r,\Omega',E,t)\mathrm{d}\Omega' = \int_{\mu_0}^{\mu_1}\mathrm{d}\mu'\int_{\varphi_0}^{\varphi_1}\phi(r,\mu',\varphi',E,t)\mathrm{d}\varphi'. \tag{30}$$

Similarly, the integral term can be transformed into an antiderivative function by repeated integration and repeated transformation. Considering the isotropic multigroup fission source in Eq. 4, the integral terms in Eq. 27 can be expressed as:

$$\int \phi_{f,g'}(r,\Omega')\mathrm{d}\Omega' = \int_{\mu_0}^{\mu_1}\mathrm{d}\mu'\int_{\varphi_0}^{\varphi_1}\phi_{f,g'}(r,\mu',\varphi')\mathrm{d}\varphi' = \int_{\mu_0}^{\mu_1}F_{0,f,g',\varphi}(r,\mu',\varphi_1)\mathrm{d}\mu' = F_{0,f,g',\mu\varphi}(r,\mu_1,\varphi_1), \tag{31}$$

besides, $\frac{\partial^2 F_{0,f,g',\mu\varphi}(r,\mu',\varphi')}{\partial\mu'\partial\varphi'} = \phi_{f,g'}(r,\mu',\varphi')$.

The constraint conditions for a definite solution are as follows: $\mu = \mu_0$, $F_{0,f,g',\mu\varphi}(r,\mu_0,\varphi') = 0$; $\varphi = \varphi_0$, $F_{0,f,g',\mu\varphi}(r,\mu',\varphi_0) = 0$.

Referring to Sect. 3.3, for the high-order scattering multigroup neutron transport equation with a two-dimensional angular variable, the high-order scattering antiderivative function can be defined as $F_{0,s,g',\mu\varphi}^{l,m}(r,\mu',\varphi')$, and the flux density moment can be expressed as:

$$\phi_{s,g'}^{l,m} = \int_{\Omega_0}^{\Omega_1} Y_l^m(\Omega')\phi_{g'}(r,\Omega')\mathrm{d}\Omega' = F_{0,s,g',\mu\varphi}^{l,m}(r,\mu_1,\varphi_1). \tag{32}$$

The constraint conditions of the definite solution are $F_{0,s,g',\mu\varphi}^{l,m}(r,\mu_0,\varphi') = F_{0,s,g',\mu\varphi}^{l,m}(r,\mu',\varphi_0) = 0$.

Furthermore,

$$\frac{\partial^2 F_{0,s,g',\mu\varphi}^{l,m}(r,\mu',\varphi')}{\partial\mu'\partial\varphi'} = Y_l^m(\mu',\varphi')\phi_{g'}(r,\mu',\varphi'). \tag{33}$$

The neutron transport equations for other forms of two-dimensional angular variables can be derived in a similar manner.

To summarize, by antiderivative function transformation of the integral term, the neutron transport equation can be converted into an exact differential equation consisting of multiple antiderivative function terms. This is shown in Eq. 34:

$$G(r, \Omega, \phi_g(r, \Omega), \Omega \cdot \nabla\phi_g(r, \Omega), F^{l,m}_{0,s,g'}(r, \Omega_1), F_{0,f,g'}(r, \Omega_1), k_{\text{eff}}) = 0. \tag{34}$$

Simultaneously, the equations transformed from antiderivative functions (Eqs. 24 and 28) have a set of corresponding constraint conditions for a definite solution.

For practical problems such as Eq. 34, if several integral terms with an identical form exist, these can be combined and simplified. The discussion is presented below.

Use $[F^{l,m}_{0,s,g'}(r, \Omega_1)]$ and $[F_{0,f,g'}(r, \Omega_1)]$ to represent the parts of $F^{l,m}_{0,s,g'}(r, \Omega_1)$ and $F_{0,f,g'}(r, \Omega_1)$ that do not include constant terms, respectively. For example, in Eq. 25, if the neutron angular flux moment $\int_{\Omega_0}^{\Omega_1} Y^m_l(\Omega')\phi_{g'}(r, \Omega')\mathrm{d}\Omega'$ contains a constant term in $Y^m_l(\Omega')$, the constant term should be shifted outside of the integral term. Then, Eq. 25 is transformed into $[F^{l,m}_{0,s,g'}(r, \Omega_1)]$. Meanwhile, using $\{[F^{l,m}_{0,s,g'}(r, \Omega_1)][F_{0,f,g'}(r, \Omega_1)]\}$ to represent a non-repetitive set, if identical formal terms exist in $[F^{l,m}_{0,s,g'}(r, \Omega_1)]$ and $[F_{0,f,g'}(r, \Omega_1)]$, these should be combined. Therefore, Eq. 34 can be expressed in a simplified form as:

$$G(r, \Omega, \phi_g(r, \Omega), \Omega \cdot \nabla\phi_g(r, \Omega), \{[F^{l,m}_{0,s,g'}(r, \Omega_1)], [F_{0,f,g'}(r, \Omega_1)]\}, k_{\text{eff}}) = 0. \tag{35}$$

The following theoretical section discusses the standard form Eq. 34. Meanwhile, the numerical validation is calculated using the simplified form Eq. 35, which is a specific form of the neutron transport equation.

## 4 Generation of the loss function for machine learning

Equations 21, 29 and 34 are obtained by transforming the neutron transport equation. These can be solved based on various basic theories and methods for solving differential equations using deep learning. An important step was to use a neural network function as the trial function, which was substituted into the neutron transport control equation. This equation was derived from antiderivative functions, constraints of definite solutions, and constraints of boundary/initial conditions to generate the loss function. Considering Eq. 34 as an example, the function represented by a fully connected deep neural network can be expressed as [31]:

$$N(x) = f(x, w, b, l, n), \tag{36}$$

where $x$ is the input vector of the neutral network, $N(x)$ is the output of the neural network, $w$ is the neural network connection weight, $b$ is the neural network bias term, $l$ is the network depth, $n$ is the number of hidden neural units, and $f$ is the activation function.

Additionally, the neural network can also have multiple outputs. These can be used to represent a group of functions with similar forms. For example, for a multigroup transport equation, different outputs of the same neural network can be used to represent the angular flux densities of different energy groups.

### 4.1 Loss function of the neutron transport control equation

Set $N_{\phi_g}(x_r, x_\Omega) = \phi_g(r, \Omega)$, $N^{l,m}_{F_{0,s,g'}}(x_r, x_{\Omega_1}) = F^{l,m}_{0,s,g'}(r, \Omega_1)$, $N_{F_{0,f,g'}}(x_r, x_{\Omega_1}) = F_{0,f,g'}(x_r, x_{\Omega_1})$. The neural network vectors $x_r, x_\Omega$ represent the spatial and angular variables $r$ and $\Omega$, respectively, in the neutron transport equation. The dimensions of the vectors should be adjusted adaptively according to the specific form of the neutron transport equation.

These equivalent relationships are substituted into Eq. 34 to obtain the loss function of the transport equation in the residual form:

$$Loss_{\phi_g} = \frac{1}{|i|}\Sigma_i \Big[G(x^i_r, x^i_\Omega, N_{\phi_g}(x^i_r, x^i_\Omega), \Omega \cdot \nabla N_{\phi_g}(x^i_r, x^i_\Omega), N^{l,m}_{F_{0,s,g'}}(x^i_r, x^i_{\Omega_1}), N_{F_{0,f,g'}}(x^i_r, x^i_{\Omega_1}), k_{\text{eff}})\Big]^2, \tag{37}$$

where $i$ represents the sample points. These sample points can be generated according to a certain probability density distribution or using different strategies based on the form of the equation.

Note that when $g = g'$, the input and output of the $N_g(r, \Omega')$ -represented $\phi_g(r, \Omega')$ in Eq. 37 and the $N_{g'}(r, \Omega')$-represented $\phi_{g'}(r, \Omega')$ in Eqs. 24 and 28 are identical. Therefore, we use $N_g(r, \Omega')$ denoted $N_{g'}(r, \Omega')$ in the following discussion.

### 4.2 Loss function of the equations transformed from antiderivative functions

As for the transformation equation of the scattering source integral term, shown in Eq. 24, set $N^{l,m}_{F_{0,s,g'}}(x_r, x_\Omega)' = F^{l,m}_{0,s,g'}(r, \Omega')' = Y^m_l(\Omega)'\phi_{g'}(r, \Omega')$, $N_{\phi_{g'}}(x_r, x_\Omega) = \phi_{g'}(r, \Omega')$, the loss function of the equations transformed from antiderivative functions can be obtained:

$$Loss^{l,m}_{F,s,g'} = \frac{1}{|i|}\Sigma_i \Big[N^{l,m}_{F_{0,s,g'}}(x^i_r, x^i_{\Omega'})' - Y^m_l(\Omega')N_{\phi_{g'}}(x^i_r, x^i_\Omega)\Big]^2. \tag{38}$$

Similarly, for the transformation equation of the fission source integral term (Eq. 28), by setting $N_{F_{0,f,g'}}(x_r, x'_\Omega)' = F_{0,f,g'}(r, \Omega')' = \phi_{g'}(r, \Omega')$, the loss function can be obtained:

$$Loss_{F,f,g'} = \frac{1}{|i|}\Sigma_i\Big[N_{F_{0,f,g'}}(x_r^i, x_{\Omega'}^i)' - N_{\phi'_g}(x_r^i, x_{\Omega'}^i)\Big]^2. \tag{39}$$

## 4.3 Loss function of constraints of the definite solution for the antiderivative function

The constraint on the definite solution for the antiderivative function can be represented as a data-driven loss function.

With regard to the constraint condition $\Omega' = \Omega_0$, $F_{0,s,g'}^{l,m}(r, \Omega_0) = 0$ of the scattering source integral term Eq. 24, by setting $N_{F_{0,s,g'}}^{l,m}(x_r, x_{\Omega_0}) = F_{0,s,g'}^{l,m}(r, \Omega_0)$, the loss function can be obtained:

$$\begin{aligned} Loss_{F_0,s,g'}^{l,m} &= \frac{1}{|j|}\Sigma_j\Big[N_{F_{0,s,g'}}^{l,m}(x_r^j, x_{\Omega_0}^j) - F_{0,s,g'}^{l,m}(x_r^j, x_{\Omega_0}^j)\Big]^2 \\ &= \frac{1}{j}\Sigma_j\Big[N_{F_{0,s,g'}}^{l,m}(x_r^j, x_{\Omega_0}^j)\Big]^2, \end{aligned} \tag{40}$$

where $j$ represents the machine learning sample point in the range of the constraint term.

Similarly, for the constraint condition $\Omega' = \Omega_0$, $F_{0,f,g'}(r, \Omega_0) = 0$ for the fission source integral term Eq. 28, by setting $N_{F_{0,f,g'}}(x_r, x_{\Omega_0}) = F_{0,f,g'}(r, \Omega_0)$, the loss function can be obtained:

$$\begin{aligned} Loss_{F_0,f,g'} &= \frac{1}{|j|}\Sigma_j\Big[N_{F_{0,f,g'}}(x_r^j, x_{\Omega_0}^j) - F_{0,f,g'}(x_r^j, x_{\Omega_0}^j)\Big]^2 \\ &= \frac{1}{|j|}\Sigma_j\Big[N_{F_{0,f,g'}}(x_r^j, x_{\Omega_0}^j)\Big]^2. \end{aligned} \tag{41}$$

In Eqs. 40 and 41, $x_{\Omega_0}^j$ represents the j-th sample point of the neural network angle input variable $x_\Omega$, which is equal to $\Omega_0$. The corresponding $x_r^j$ sample point values should cover all the value domains of variable $r$ in $F_{0,s,g'}^{l,m}(r, \Omega_0), F_{0,f,g'}(r, \Omega_0)$.

## 4.4 Loss function of boundary condition

The neutron transport equation should have additional boundary and initial conditions to obtain a solution. In this case, we can refer to existing deep learning methods for solving the neutron diffusion and transport equations. Based on the various forms of boundary conditions, additional boundary/initial condition loss functions should be set for the corresponding neural network functions and their corresponding differential forms [12, 13, 18, 19]. If the equation contains a time term, the corresponding initial condition should be set. The multigroup transport equation has the following form:

$$Loss_{b,g} = \frac{1}{|k|}\Sigma_k\Big[N_{\phi_g}(x_r^k, x_\Omega^k) - K_{\phi_g}(x_r^k, x_\Omega^k)\Big]^2, \tag{42}$$

where $K_{\phi_g}(r, \Omega)$ is the boundary function of the angular flux density equation for the g-th energy group. $k$ represents the machine learning sample point on the boundary domain. The values of $r$ and $\Omega$ are extracted at the boundary of the equation. Their forms are determined based on the specific forms of the boundary condition of the neutron transport equation, such as the vacuum and reflection boundaries.

As an example, the form of a nonnegative boundary loss function can be expressed as:

$$Loss_{b,g,abs} = \frac{1}{|k|}\Sigma_k\Big[N_{\phi_g}(x_r^k, x_\Omega^k) - \Big|N_{\phi_g}(x_r^k, x_\Omega^k)\Big|\Big]^2. \tag{43}$$

## 4.5 Weighted loss function

Different neural network functions are used as objects to construct the weighted loss functions.

(1) Weighted loss function of the angular flux density

With regard to the neural network $N_{\phi_g}(x_r, x_\Omega)$ of the angular flux in the neutron transport equation, the corresponding equation and boundary condition loss functions are selected and weighted to obtain the corresponding weighted loss function. If the output of each neural network represents the angular flux of the different energy groups, the weighted loss function for each group is

$$Loss_{\phi_g,all} = P_\phi Loss_{\phi_g} + P_b Loss_{b,g}. \tag{44}$$

If multiple outputs of a neural network function are used to represent the angular flux of different energy groups, the overall loss function for the neural network is

$$Loss_{\phi_g,all} = \sum_{g'=1}^{G}(P_\phi Loss_{\phi_g} + P_b Loss_{b,g}), \tag{45}$$

where $P_\phi, P_b$ are the weights corresponding to the flux and boundary/initial condition loss functions. These can be determined based on experimental experience.

(2) Weighted loss function of the antiderivative function

Similarly, according to the different neural network functions $N^{l,m}_{F_{0,s,g'}}(x_r, x_{\Omega'})$, $N_{F_{0,f,g'}}(x_r, x_{\Omega'})$, we construct the corresponding weighted loss function.

The weighted loss function for the scattering source integral term neural network $N^{l,m}_{F_{0,s,g'}}(x_r, x_{\Omega'})$ and fission source integral term neural network $N_{F_{0,f,g'}}(x_r, x_{\Omega'})$ are shown:

$$Loss^{l,m}_{s,g',\text{all}} = P_{F,s} Loss^{l,m}_{F,s,g'} + P_{d,s} Loss^{l,m}_{F_0,s,g'}, \tag{46}$$

$$Loss_{f,g',\text{all}} = P_{F,f} Loss_{F,f,g'} + P_{d,f} Loss^{l,m}_{F_0,f,g'}, \tag{47}$$

where $P_{F,s}$, $P_{d,s}$ represent the equation transformed from the antiderivative function of the scattering source integral term and the corresponding determined solution loss function weights, respectively. $P_{F,f}$ and $P_{d,f}$ represent the equations transformed from the antiderivative function of the fission source integration term and the corresponding solution loss function, respectively.

If Eqs. 46 and 47 use multiple outputs of a neural network function, these can be represented as:

$$Loss^{l,m}_{s,G,\text{all}} = \sum_{g'=1}^{G} \left( P_{F,s} Loss^{l,m}_{F,s,g'} + P_{d,s} Loss^{l,m}_{F_0,s,g'} \right), \tag{48}$$

$$Loss_{f,G,\text{all}} = \sum_{g'=1}^{G} \left( P_{F,f} Loss_{F,f,g'} + P_{d,f} Loss^{l,m}_{F_0,f,g'} \right). \tag{49}$$

The various weight values of the loss function $P$ included in Eqs. 44–49, as well as the number of layers $l$ and number of hidden neurons $n$ in Eq. 36, are hyperparameters in the deep learning process. Their values are typically determined and optimized using experimental methods. With regard to the selection of various loss function weights, this paper recommends the following operation: set $P_\phi = 1$ as a fixed benchmark, with other weights $P_b, P_{F,s}, P_{d,s}, P_{F,f}, P_{d,f}$ ranging between 10 and 50. During the machine learning training process, these weights are adjusted dynamically based on a decrease in the loss function. An effective dynamic adjustment strategy formulated based on the experiments in this study is as follows: During the training process of machine learning, it is advantageous to sequentially decrease the weight values of $P$ according to the decrease in loss function, excluding $P_\phi$. Their values decrease gradually from large to small. This approach can improve the performance by reducing the loss function, thereby enhancing the efficiency of the M-AIM.

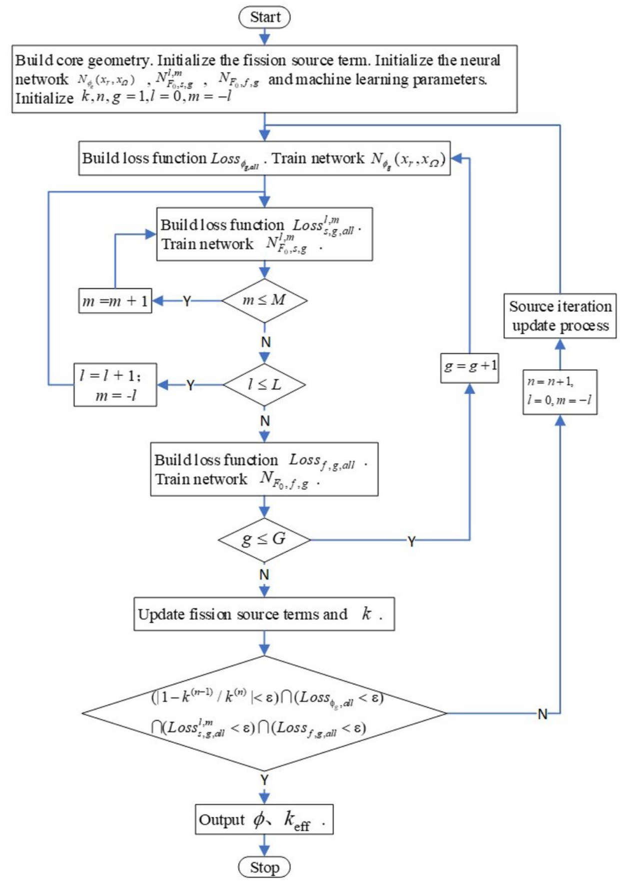



**Fig. 1** Calculation process of M-AIM for neutron transport equations. The figure illustrates the calculation process of M-AIM, which contains three neural networks $N_{\phi_g}(x_r, x_\Omega)$, $N^{l,m}_{F_{0,s,g}}$, and $N_{F_{0,f,g}}$ through alternating iterations

## 5 Framework of deep learning

After transforming the transport equation into an exact differential equation using an antiderivative function and defining the loss functions for each neural network, we can apply automatic differentiation techniques [32] to perform iterative deep learning in an alternating manner. This process is based on standard architectures such as PINN [12] and extends to improved architectures designed for solving various differential equations [33, 34]. It systematically minimizes the loss functions of all the neural networks involved. Consequently, the neural network output, which represents the neutron flux density, serves as a numerical solution to the neutron transport equation.

Please note that the eigenvalues in Eq. 34 can be derived using conventional source-iteration methods. The approach presented in this study is designed to be integrated into an iterative process, specifically within the inner loop. Furthermore, we may utilize specialized deep-learning-based indirect methods to determine the eigenvalues [19]. In addition, the direct search method proposed in [35] was designed to concurrently optimize the eigenvalues and weights of the neural network. In the following discussion, we use the conventional source-iteration method to solve for $k_{\mathrm{eff}}$.

In the multigroup transport equation, the deep learning process utilizes weighted loss functions, as detailed in Eqs. 44–49. It involves iteratively adjusting the neural network functions $N_{\phi_g}(x_r, x_\Omega)$, $N^{l,m}_{F_{0,s,g'}}(x_r, x_{\Omega'})$, and $N_{F_{0,f,g'}}(x_r, x_{\Omega'})$ related to the terms $w$, $b$ in Eq. 36. This is aimed at minimizing $Loss_{\phi_g,\mathrm{all}}$, $Loss^{l,m}_{\mathrm{s},g',\mathrm{all}}$, $Loss_{\mathrm{f},g',\mathrm{all}}$. At this point, the output of $N_{\phi_g}(x_r, x_\Omega)$ is approximately $\phi_g(r, \Omega)$. In practice, specific convergence criteria for machine learning are generally established based on the loss function values or number of iterations. For example, if all the loss functions fall below a predefined minimum threshold $\varepsilon$, the machine learning process is considered complete. Alternatively, the maximum number of iterations $N$ can be set for the machine learning process. Training ceases when this limit is exceeded.

In the case of a neural network with a single output, the convergence criterion dictates that all the loss functions and iterative parameters of the source-iteration method should be below a predefined threshold $\varepsilon$. When this condition is satisfied, the training procedure is illustrated in Fig. 1.

In this figure, the loss functions for the unknown function of the angular flux, as well as for the antiderivative function of the scattering and fission sources, are defined by Eqs. 44, 46 and 47.

## 6 Numerical results

### 6.1 Critical homogeneous single-group problem of slab geometry

To facilitate a comparison with the theoretical values, we first verified the critical form of the single-group planar geometric first-order anisotropic scattering source Eq. 6. $k_{\mathrm{eff}}$ was set equal to 1.0. A set of geometric and cross-sectional parameters that are theoretically critical for verification were identified.

The isotropic form of Eq. 6 for a single group is

$$\mu \frac{\partial \phi(x,\mu)}{\partial x} + \Sigma_{\mathrm{t}}(x)\phi(x,\mu) = \frac{1}{2}\Sigma_{\mathrm{s0}}(x)\int_{-1}^{1}\phi(x,\mu')\mathrm{d}\mu' + \frac{\nu(x)\Sigma_{\mathrm{f}}(x)}{2k_{\mathrm{eff}}}\int_{-1}^{1}\phi(x,\mu')\mathrm{d}\mu'. \tag{50}$$

Analyzing and comparing Eqs. 6 and 50 for a single group revealed the difference to be in Eq. 6. This equation includes a first-order scattering term $\frac{3}{2}\Sigma_{\mathrm{s1}}(x)\mu\int_{-1}^{1}\mu'\phi(x,\mu')\mathrm{d}\mu'$. The integral of the first-order scattering source is symmetric at approximately zero. Therefore, this scattering source term affects only the shape of the flux and not the eigenvalue of the equation. It is independent of the value of the first-order scattering cross-section $\Sigma_{\mathrm{s1}}(x)$. Therefore, if a set of geometric cross-sectional parameters is identified, Eq. 50 satisfies the critical condition, and the corresponding Eq. 6 is critical.

Problem setup: We assumed slab material properties $\Sigma_{\mathrm{t}} = 0.050\,\mathrm{cm}^{-1}$, $\Sigma_{\mathrm{s0}} = 0.030\,\mathrm{cm}^{-1}$, $\Sigma_{\mathrm{s1}} = 0.001705\,\mathrm{cm}^{-1}$, and $\nu\Sigma_{\mathrm{f}} = 0.0225\,\mathrm{cm}^{-1}$, and vacuum boundary conditions on both sides of the plate. According to the literature [2], the parameter $C = (\Sigma_{\mathrm{s0}} + \nu\Sigma_{\mathrm{f}})/\Sigma_{\mathrm{t}}$ related to zeroth-order scattering cross-sectional data is equal to 1.05. The critical geometric half-thickness of the problem is $b = 3.300263772\,\mathrm{cm}$, $\lambda = 66.00527544\,\mathrm{cm}$. Here, $\lambda = \frac{1}{\Sigma_{\mathrm{t}}}$ is the neutron mean free path. At this time, $k_{\mathrm{eff}} = 1$. This indicates a critical state.

For the single-group form of the first-order scattering equations (Eq. 6) according to the M-AIM described in Sect. 3.3 and the simplified form Eq. 32, we obtain

$$\mu \frac{\partial \phi(x,\mu)}{\partial x} + \Sigma_{\mathrm{t}}(x)\phi(x,\mu) = \frac{1}{2}\Sigma_{\mathrm{s0}}(x)[F^{0,0}_{0,\mathrm{s}}(x,1)] + \frac{3}{2}\Sigma_{\mathrm{s1}}(x)\mu[F^{1}_{0,\mathrm{s}}(x,1)] + \frac{\nu(x)\Sigma_{\mathrm{f}}(x)}{2k_{\mathrm{eff}}}[F_{0,\mathrm{f}}(x,1)]. \tag{51}$$

At this point, the constraint of definite solution for the antiderivative function is $\mu_0 = -1$, $[F^{0,0}_{0,\mathrm{s}}(x,-1)] = [F^{1}_{0,\mathrm{s}}(x,-1)] = F_{0,\mathrm{f}}(x,-1) = 0$, and $[F^{1}_{0,\mathrm{s}}(x,1)] = \sum_{m=-1}^{1}[F^{1,m}_{0,\mathrm{s}}(x,1)]$, $[F^{0,0}_{0,\mathrm{s}}(x,\mu')]' = [F_{0,\mathrm{f}}(x,\mu')]' = \phi(x,\mu')$, $[F^{1}_{0,\mathrm{s}}(x,\mu')]' = \mu'\phi(x,\mu')$.

It is observed that in Eq. 51, both zeroth-order scattering and fission sources are isotropic, and $[F^{0,0}_{0,\mathrm{s}}(x,\mu')] = [F_{0,\mathrm{f}}(x,\mu')]$. Therefore, when neural networks $N_\phi(x,\mu)$ are used to represent $\phi(x,\mu)$, the zeroth-order scattering source neural network $N^{0,0}_{F_{0,\mathrm{s}}}(x,\mu')$ is equivalent to the fission source neural network $N_{F_{0,\mathrm{f}}}(x,\mu')$ (which simplifies to a single neural network $N_{F_{0,\mathrm{s,f}}}(x,\mu')$), whereas the first-order scattering neural network $N_{F^{1}_{0,\mathrm{s}}}(x,\mu')$ is represented separately. This forms a total of three neural networks. The machine learning process shown in Fig. 1 can be simplified.

The machine learning method and parameter selection were as follows: the neural network was set to ($n = 20$) and ($l = 9$), the tanh activation function was applied, the network was initialized with a Gaussian random distribution, and the ADAM machine learning algorithm was employed [12, 19, 29]. The learning rate started at $1 \times 10^{-4}$ and decreased gradually to $5 \times 10^{-7}$ as the number of learning iterations increased. It stopped after $1 \times 10^{-4}$ iterations. To accelerate the convergence, referring to the literature [19, 29], a feature value constraint $\phi(0, \pm 1) = 0.2$ was set. The number of sample points for the control equation was 8000, and 100 sample points were considered under the boundary conditions. The weight coefficient of the loss function is an empirical parameter. The strategy of this study was to use $P_{\phi}$=1 as a benchmark and periodically replace different values of $P_{\mathrm{b}}$, $P_{F,\mathrm{s}}$, $P_{F,\mathrm{f}}$, and $P_{d,\mathrm{s}}$ with values ranging from 10 to 50 during the training process. The hardware was equipped with an Intel Core i9-11900K CPU and NVIDIA RTX A2000 GPU.

Figure 2 shows the flux density equation $\phi(x, \mu)$, antiderivative function of the zeroth-order scattering source/fission source $[F_{0,\mathrm{s}}^{0,0}(x, \mu)]/[F_{0,\mathrm{f}}(x, 1)]$, antiderivative function of the first-order scattering source $F_{0,\mathrm{s}}^{1}(x, \mu)$, and numerical calculation results $\mu[F_{0,\mathrm{s}}^{1}(x, 1)]$. Figure 3 compares the center-normalized scalar flux and the reference solution, including the theoretical values, calculation results from OpenMC [36], variational nodal methods (VNM) [37], and residual network (ResNet) [38].

As shown above, the application of deep learning techniques successfully produces flux solutions that align well with the theoretical values. Furthermore, it provides a continuous flux distribution under anisotropic scattering conditions. It is noteworthy that the impact of the higher-order scattering terms $\mu[F_{0,\mathrm{s}}^{1}(x, 1)]$ on the flux density is symmetric about zero. This demonstrates that these scattering terms affect only the flux shape and not the equation $k_{\mathrm{eff}}$. Additionally, the experiment revealed that the M-AIM can be applied to various deep learning frameworks such as fully connected neural networks and residual neural networks. This demonstrates its good applicability (Table 1).

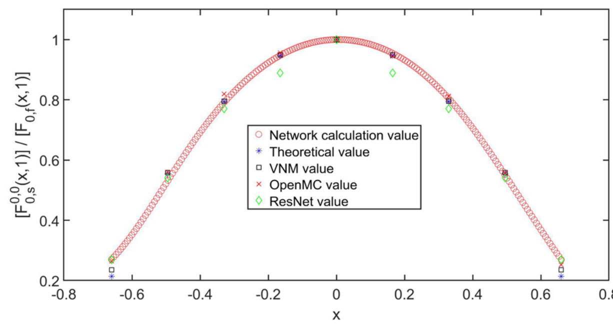



**Fig. 3** (Color online) Results of center-normalized scalar flux, which contains the scalar flux at different positions of the critical single-group slab geometric anisotropic transport problem

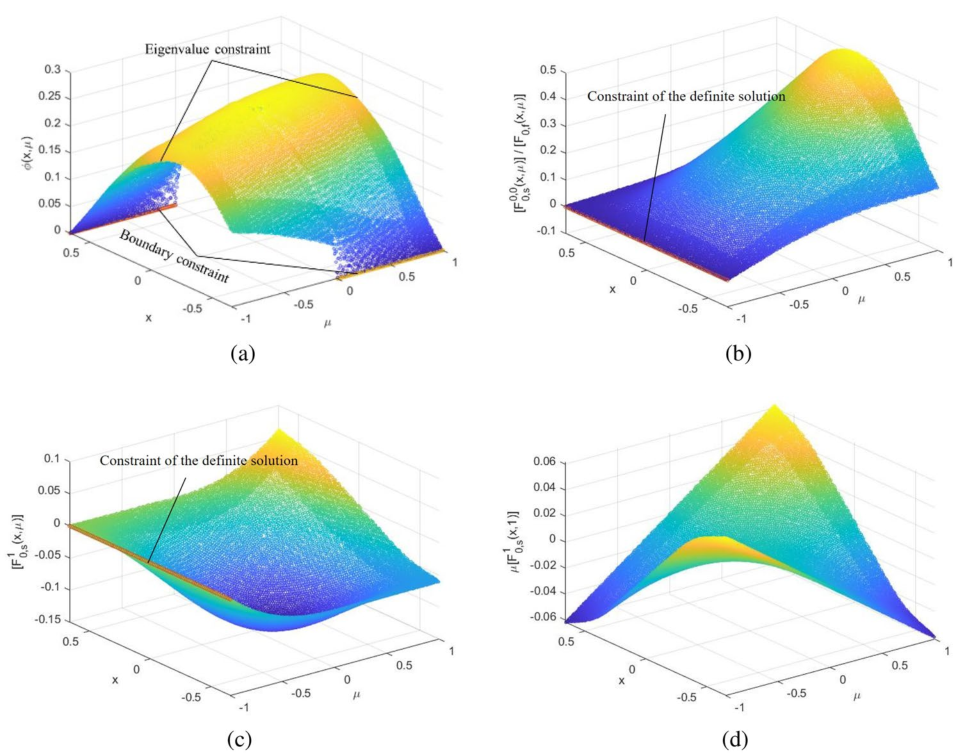



**Fig. 2** (Color online) Result of single-group slab geometric anisotropic transport problem. It shows the numerical calculation results of the critical single-group slab geometric anisotropic transport problem, including the antiderivative functions of different orders (zeroth-order scattering source/zero-order fission source/first-order scattering source) and continuous numerical solutions for angular flux. **a** Angular flux $\phi(x, \mu)$ distribution; **b** distribution of the antiderivative function $[F_{0,\mathrm{s}}^{0,0}(x, \mu)]/[F_{0,\mathrm{f}}(x, \mu)]$; **c** distribution of the antiderivative function $[F_{0,\mathrm{s}}^{1}(x, \mu)]$; and **d** distribution of the antiderivative function $\mu[F_{0,\mathrm{s}}^{1}(x, 1)]$

**Table 1** Flux results of different locations for single-group slab geometry

| Method | Different values for *x*/*b* | | | | | |
|---|---|---|---|---|---|---|
| | 0 (center) | 0.25 | 0.50 | 0.75 | 1.00 (boundary) | Time (min) |
| Actual calculation value (M-AIM) | 0.49994469 | 0.47717363 | 0.40185571 | 0.27500153 | 0.13455909 | 505 |
| Normalized calculation value (M-AIM) | 1 | 0.95445284 | 0.80380034 | 0.55006390 | 0.26914796 | / |
| Normalized theoretical value | 1 | 0.94714400 | 0.79372641 | 0.55329025 | 0.21419206 | / |
| Normalized VNM value | 1 | 0.94965777 | 0.79561143 | 0.55840989 | 0.23590973 | 0.03 |
| Normalized OpenMC value | 1 | 0.94440551 | 0.81270187 | 0.55785209 | 0.25272043 | 2.5 |
| Normalized ResNet value | 1 | 0.88963412 | 0.77045493 | 0.54105856 | 0.26854225 | 450 |
| M-AIM: mean-squared error relative to theoretical value | $7.59151071 \times 10^{-4}$ | | | | | |
| M-AIM: mean-squared error relative to VNM value | $3.58673270 \times 10^{-4}$ | | | | | |
| M-AIM: mean-squared error relative to OpenMC value | $2.56903029 \times 10^{-4}$ | | | | | |
| ResNet: mean-squared error relative to theoretical value | $1.54500122 \times 10^{-3}$ | | | | | |

Theoretical values are available in reference [2]

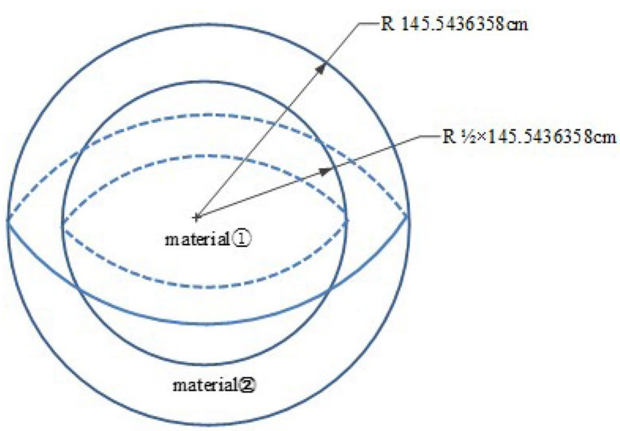



**Fig. 4** (Color online) Geometric diagram for non-critical single-group multi-material problem

## 6.2 Non-critical multi-material single-group problem of spherical geometry

Problem setup: Two materials are used for spherical fuels (Fig. 4) with vacuum boundary conditions on the outer surface.

The spherical geometry consists of two regions composed of two materials. The material parameters are listed in Table 2.

The basic equation for the anisotropic transport problem of a single-group spherical scattering source is given by Eq. 7. Applying the antiderivative function transformation method described in Sect. 3.3, we obtain

**Table 2** Material properties of non-critical single-group multi-material region of spherical geometry

| Material number | $\Sigma_t$ (cm$^{-1}$) | $\Sigma_{s,0}$ (cm$^{-1}$) | $\Sigma_{s,1}$ (cm$^{-1}$) | $\nu\Sigma_f$ (cm$^{-1}$) |
|---|---|---|---|---|
| ① | 0.05 | 0.03 | 0.001705 | 0.0255 |
| ② | 0.05 | 0.03 | 0.001705 | 0.023625 |

$$\mu\frac{\partial\phi(r,\mu)}{\partial r}+\frac{1-\mu^2}{r}\frac{\partial\phi(r,\mu)}{\partial\mu}\Sigma_t(r)\phi(r,\mu) = \frac{1}{2}\Sigma_{s0}(r)[F^{0,0}_{0,s}(x,1)]+\frac{3}{2}\Sigma_{s1}(r)\mu[F^{1}_{0,s}(r,1)] + \frac{\mu(r)\Sigma_f(r)}{2k_{eff}}[F_{0,f}(r,1)]. \tag{52}$$

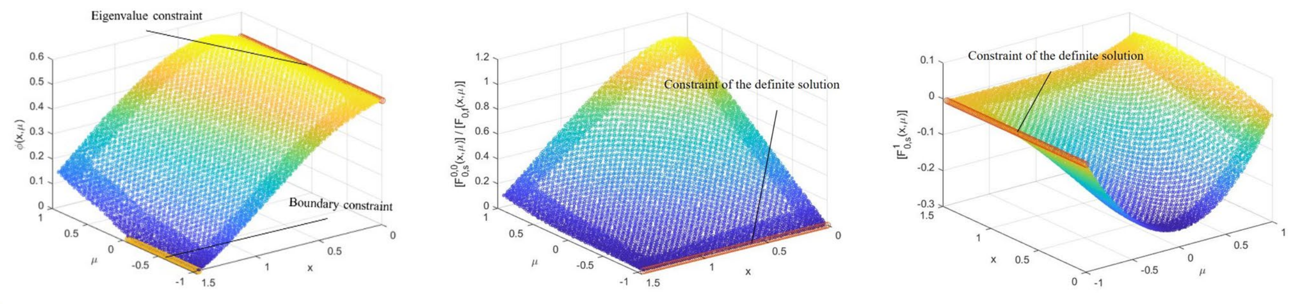
The constraints of the definite solution for the antiderivative function are $\mu_0=-1, [F^{0,0}_{0,s}(x,-1)]=[F^{1}_{0,s}(x,-1)]=[F_{0,f}(x,-1)]=0$ . Similar to Sect. 6.1, three neural networks are used to represent the flux, first-order scattering source antiderivative function, and zeroth-order scattering and fission source antiderivative functions. Because the zero-order scattering and fission source integral terms have an identical form, these can be represented using the same neural network.

The machine learning method and parameter selection were as follows: the neural network was set to $n=40$ and $l=9$, and the eigenvalue constraint was $\phi(0,\mu)=0.5$, and the other parameters were identical to those in Sect. 6.1.

**Table 3** Calculation result for single-group spherical geometry

| Method | $k_{\text{eff}}$ | $k_{\text{eff}}$ /relative error | Mean-squared error of scalar flux | Time (min) | $Loss_{\phi,\text{all}}$ | $Loss^{0,0}_{\text{s,all}}$ | $Loss^{1}_{\text{s,all}}$ |
|---|---|---|---|---|---|---|---|
| M-AIM | 1.0218 | −0.0053 | $1.9200 \times 10^{-3}$ | 764 | $6.7260 \times 10^{-4}$ | $3.2938 \times 10^{-6}$ | $1.3420 \times 10^{-9}$ |
| OpenMC | 1.0272 | / | / | 2.0 | / | / | / |

The relative error and MSE of flux were obtained using the results obtained by the M-AIM and conventional methods




(a) (b) (c)

**Fig. 5** (Color online) Result of single-group spherical geometric transport problem. It shows the numerical calculation results of the non-critical single-group spherical geometric anisotropic transport problem with two materials. These include the antiderivative functions of different orders (zeroth-order scattering source/zeroth-order fission source/first-order scattering source) and continuous numerical solutions for angular flux. **a** Angular flux density $\phi(x, \mu)$ distribution; **b** distribution of the antiderivative function $[F^{0,0}_{0,\text{s}}(x, \mu)]/[F_{0,\text{f}}(x, \mu)]$; **c** antiderivative function $[F^{1}_{0,\text{s}}(x, \mu)]$ distribution of the first-order scattering source

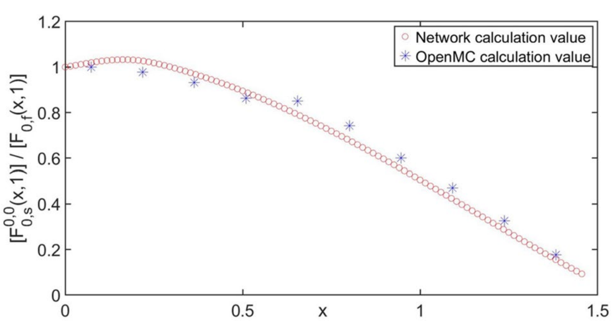



**Fig. 6** (Color online) Normalized flux at the spherical geometric center of a sphere. It contains the scalar flux at different positions of the non-critical single-group spherical geometric anisotropic transport problem with two materials

Figure 5 presents the numerical results of the equation's flux density $\phi(x, \mu)$, zeroth-order scattering source/fission source antiderivative function $[F^{0,0}_{0,\text{s}}(x, \mu)]/[F_{0,\text{f}}(x, 1)]$, and first-order scattering source antiderivative function $[F^{1}_{0,\text{s}}(x, \mu)]$. Figure 6 shows a scatter plot of the normalized scalar flux in the central region for the spherical geometry. Table 3 presents the comparative calculation results for the eigenvalues between the M-AIM and OpenMC, as well as the final loss function. The numerical calculation results validate the accuracy of the M-AIM while also providing a continuous distribution of the angular flux density.

**Table 4** Material property in the calculation region of two-group slab geometry

| $g$ | $\Sigma_{\text{t}}$ (cm$^{-1}$) | $v\Sigma_{\text{f}}$ (cm$^{-1}$) | $\chi_g$ | $\Sigma_{\text{s},0,g-1}$ (cm$^{-1}$) | $\Sigma_{\text{s},0,g-2}$ (cm$^{-1}$) | $\Sigma_{\text{s},1,g-1}$ (cm$^{-1}$) | $\Sigma_{\text{s},1,g-2}$ (cm$^{-1}$) |
|---|---|---|---|---|---|---|---|
| 1 | 0.33285210 | 0.01266922 | 1 | 0.31876071 | 0.00093413 | 0.00730600 | 0.00004500 |
| 2 | 0.44925527 | 0.22110247 | 0 | 0 | 0.31466259 | 0 | 0.00129800 |

### 6.3 Non-critical homogeneous two-group problem of slab geometry

Problem setup: For the two-group single-material region problem in a slab geometry, with geometric parameters as in Sect. 6.1 and the material parameters listed in Table 4, the basic forms of the two-group slab theory and transport equations are given by Eq. 6 and a detailed explanation in Refs. [1, 2, 5]. Referring to Eq. 29 for the antiderivative function transformation equation, we obtain

$$\begin{cases} \mu\frac{\partial\phi_1(x,\mu)}{\partial x}+\Sigma_{t,1}(x)\phi_1(x,\mu)= \\ \frac{1}{2}(\Sigma_{s,0,1-1}(x)[F^{0,0}_{0,s,1}(x,1)]+\Sigma_{s,0,2-1}(x)[F^{0,0}_{0,s,2}(x,1)]) \\ +\frac{3}{2}(\Sigma_{s,1,1-1}(x)\mu[F^{1}_{0,s,1}(x,1)]+\Sigma_{s,1,2-1}(x)\mu[F^{1}_{0,s,2}(x,1)]) \\ +\frac{1}{2k_{\mathrm{eff}}}((\nu\Sigma_f(x))_1[F_{0,f,1}(x,1)]+(\nu\Sigma_f(x))_2[F_{0,f,2}(x,1)]) \\ \mu\frac{\partial\phi_2(x,\mu)}{\partial x}+\Sigma_{t,2}(x)\phi_2(x,\mu) \\ =\frac{1}{2}(\Sigma_{s,0,1-2}(x)[F^{0,0}_{0,s,1}(x,1)]+\Sigma_{s,0,2-2}(x)[F^{0,0}_{0,s,2}(x,1)]) \\ +\frac{3}{2}(\Sigma_{s,1,1-2}(x)\mu[F^{1}_{0,s,1}(x,1)]+\Sigma_{s,1,2-2}(x)\mu[F^{1}_{0,s,2}(x,1)]) \end{cases} \tag{53}$$

At this point, the constraints of definite solution for the antiderivative function $[F^{0,0}_{0,s,1}(x,-1)]$, $[F^{0,0}_{0,s,2}(x,-1)]$, $[F^{1}_{0,s,1}(x,-1)]$,$[F^{1}_{0,s,2}(x,-1)]$,$[F_{0,f,1}(x,-1)]$,$[F_{0,f,2}(x,-1)]$ were set to be equal to zero. Different neural networks were used to represent the flux densities of the subgroups and the antiderivative functions of the scattering source/fission source in Eq. 53. Similar to Sect. 6.1, the zeroth-order scattering source antiderivative function is equivalent to the fission source antiderivative function. These can be combined and simplified to yield a total of six neural networks.

The machine learning method and parameter selection were as follows: The eigenvalue constraint for the fast group was $\phi_1(0,-1)=\phi_1(0,1)=1$. Furthermore, the eigenvalue updating method adopted the source-iteration method with other network parameters, loss functions, and sample generation methods, as described in Sect. 6.1.

Because this problem does not have an analytical solution or theoretical estimate, the results of the M-AIM were compared with the normalized scalar flux calculation results obtained using conventional methods (both normalized based on the scalar flux value at $x = 0$ for the fast group). This is shown in Fig. 7 and Table 5. The results of the loss functions are listed in Table 6.

**Table 6** Results of loss functions for non-critical two-group scalar flux of slab geometry

| Energy group | $Loss_{g,\mathrm{all}}$ | $Loss^{0,0}_{s,G,\mathrm{all}}$ | $Loss^{1}_{s,G,\mathrm{all}}$ |
|---|---|---|---|
| Fast($g = 1$) | $5.0392\times10^{-4}$ | $4.7992\times10^{-7}$ | $6.6793\times10^{-9}$ |
| Thermal($g = 2$) | $1.4554\times10^{-5}$ | $3.5532\times10^{-4}$ | $1.4946\times10^{-8}$ |

### 6.4 Fixed-source homogeneous single-group problem of infinite cylindrical geometry

Problem setup: For the fixed-source single-group infinite cylindrical geometry single-material region problem, the parameter radius $R = 1.08225766\,\mathrm{m}$. The material parameters are identical to those described in Sect. 6.1 under the zero-flux boundary condition. According to literature [1, 2, 5], the basic form of a transport equation with first-order anisotropy is as follows:

$$\sqrt{1-\mu^2}\left(\cos\varphi\frac{\partial}{\partial r}-\frac{\sin\varphi}{r}\frac{\partial}{\partial\varphi}\right)\phi(r,\mu,\varphi) + \Sigma_t\phi(r,\mu,\varphi)=Q_s+Q_f, \tag{54}$$

where $Q_s=\frac{1}{4\pi}\Sigma_{s,0}(r)\int_{-1}^{1}\int_0^{2\pi}\mathrm{d}\varphi'\mathrm{d}\mu'+\frac{3}{4\pi}\Sigma_{s,1}(r)\mu\int_{-1}^{1}\int_0^{2\pi}\phi(r,\mu',\varphi')\mu'\mathrm{d}\varphi'\mathrm{d}\mu'+\frac{3}{4\pi}\Sigma_{s,1}(r)(1-\mu^2)^{1/2}\sin\varphi\int_{-1}^{1}\int_0^{2\pi}\phi(r,\mu',\varphi')(1-\mu'^2)^{1/2}\sin\varphi'\mathrm{d}\varphi'\mathrm{d}\mu'+\frac{3}{4\pi}\Sigma_{s,1}(r)(1-\mu^2)^{1/2}\cos\varphi\int_{-1}^{1}\int_0^{2\pi}\phi(r,\mu',\varphi')(1-\mu'^2)^{1/2}\cos\varphi'\mathrm{d}\varphi'\mathrm{d}\mu'$ .

$Q_f=0.2\cos\left(\frac{\pi r}{2R}\right)$.

The above equation was transformed using the method described in Sect. 3.4 and the simplified transformation form of Eq. 34 to obtain:

$$\sqrt{1-\mu^2}\left(\cos\varphi\frac{\partial\phi(r,\mu,\varphi)}{\partial r}-\frac{\sin\varphi}{r}\frac{\partial\phi(r,\mu,\varphi)}{\partial\varphi}\right) + \Sigma_t\phi(r,\mu,\varphi)=Q_s+Q_f, \tag{55}$$

where $Q_s=\frac{1}{4\pi}\Sigma_{s,0}(r)\left[F^{0,0}_{0,s,\mu\varphi}(r,1,2\pi)\right]+\frac{3}{4\pi}\Sigma_{s,1}(r)\mu\left[F^{1,0}_{0,s,\mu\varphi}(r,1,2\pi)\right]+\frac{3}{4\pi}\Sigma_{s,1}(r)(1-\mu^2)^{1/2}\sin\varphi\left[F^{1,-1}_{0,s,\mu\varphi}(r,1,2\pi)\right]+\frac{3}{4\pi}\Sigma_{s,1}(r)(1-\mu^2)^{1/2}\cos\varphi\left[F^{1,1}_{0,s,\mu\varphi}(r,1,2\pi)\right]$ .

$Q_f=0.2\cos\left(\frac{\pi r}{2R}\right)$.

**Table 5** Calculation result for non-critical two-group scalar flux of slab geometry

| Method | $k_{\mathrm{eff}}$ | $k_{\mathrm{eff}}$/relative error | MSE of flux | | Time (min) |
|---|---|---|---|---|---|
| | | | Fast group | Thermal group | |
| OpenMC | 0.9710 | −0.0076 | $5.1988\times10^{-4}$ | $1.3964\times10^{-3}$ | 6 |
| VNM | 0.9700 | −0.0037 | $4.9479\times10^{-4}$ | $1.3614\times10^{-3}$ | 0.06 |
| M-AIM | 0.9636 | / | / | / | 2360 |

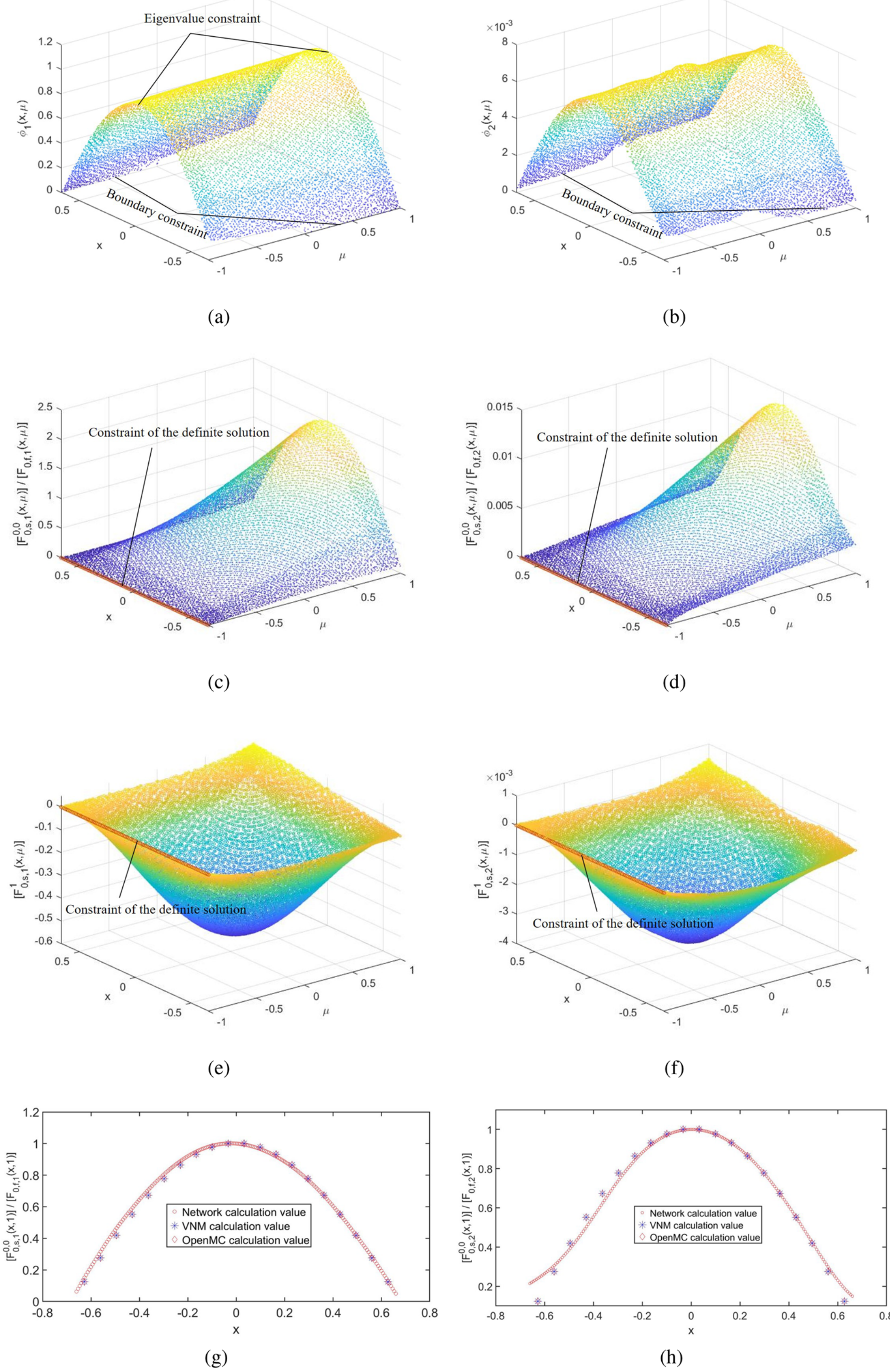



**Fig. 7** (Color online) Flux result for fast/thermal group at different positions of non-critical single-material two-group slab geometry scattering source anisotropic transport problem. It shows the numerical calculation results including the antiderivative functions of different orders (zeroth-order scattering source/zeroth-order fission source/first-order scattering source) and continuous numerical solutions for angular flux. **a** Fast group flux density $\phi_1(x,\mu)$; **b** thermal group flux density $\phi_2(x,\mu)$; **c** distribution of the fast group antiderivative function $[F^{0,0}_{0,\mathrm{s},1}(x,\mu)]/[F_{0,\mathrm{f},1}(x,\mu)]$; **d** distribution of the thermal group antiderivative function $[F^{0,0}_{0,\mathrm{s},2}(x,\mu)]/[F_{0,\mathrm{f},2}(x,\mu)]$; **e** distribution of the antiderivative function $[F^{1}_{0,\mathrm{s},1}(x,\mu)]$ of the first-order scattering source in the fast group; **f** distribution of the antiderivative function $[F^{1}_{0,\mathrm{s},2}(x,\mu)]$ of the first-order scattering source in the thermal group; **g** center-normalized scalar flux of fast group; and **h** center-normalized scalar flux of thermal group

In Eq. 55, it is given that $\frac{\partial^2[F^{0,0}_{0,\mathrm{s},\mu\varphi}(r,\mu',\varphi')]}{\partial\mu'\partial\varphi'} = \phi(r,\mu',\varphi')$, $\frac{\partial^2[F^{1,0}_{0,\mathrm{s},\mu\varphi}(r,\mu',\varphi')]}{\partial\mu'\partial\varphi'} = \phi(r,\mu',\varphi')\mu'$, $\frac{\partial^2[F^{1,-1}_{0,\mathrm{s},\mu\varphi}(r,\mu',\varphi')]}{\partial\mu'\partial\varphi'} = \phi(r,\mu',\varphi')(1-\mu'^2)^{1/2}\sin\varphi'$, $\frac{\partial^2[F^{1,1}_{0,\mathrm{s},\mu\varphi}(r,\mu',\varphi')]}{\partial\mu'\partial\varphi'} = \phi(r,\mu',\varphi')(1-\mu'^2)^{1/2}\cos\varphi'$.

At this point, the following are the constraints of the definite solution for the antiderivative function: $[F^{0,0}_{0,\mathrm{s},\mu\phi}(r,-1,\varphi)] = [F^{0,0}_{0,\mathrm{s},\mu\phi}(r,\mu,0)] = [F^{0,0}_{0,\mathrm{s},\mu\phi}(r,0,0)] = 0$, $[F^{1,0}_{0,\mathrm{s},\mu\phi}(r,-1,\varphi)] = [F^{1,0}_{0,\mathrm{s},\mu\phi}(r,\mu,0)] = [F^{1,0}_{0,\mathrm{s},\mu\phi}(r,0,0)] = 0$, $[F^{1,-1}_{0,\mathrm{s},\mu\phi}(r,-1,\varphi)] = [F^{1,-1}_{0,\mathrm{s},\mu\phi}(r,\mu,0)] = [F^{1,-1}_{0,\mathrm{s},\mu\phi}(r,0,0)] = 0$, $[F^{1,1}_{0,\mathrm{s},\mu\phi}(r,-1,\varphi)] = [F^{1,1}_{0,\mathrm{s},\mu\phi}(r,\mu,0)] = [F^{1,1}_{0,\mathrm{s},\mu\phi}(r,0,0)] = 0.$

The angular flux density $\phi(r, \mu, \varphi)$ is represented by neural network $N_\phi(x_r, x_\mu, x_\varphi)$. Let $N^{0,0}_{F_{0,s,\mu\varphi}}$ represent the zeroth-order scattering source antiderivative function and $N^{1,m}_{F_{0,s,\mu\varphi}}$ represent the first-order scattering source antiderivative functions for different values of $m$. $N^{l,m}_{F_{0,s,\mu\varphi}}(x_r, x_{u'}, x_{\varphi'}) = \left[F^{l,m}_{0,s,\mu\varphi}(r, u', \varphi')\right]$. Machine learning was performed according to the complete process illustrated in Fig. 1.

The machine learning method and parameter selection were as follows: The neural network was set with $n = 100$ and $l = 5$. The other parameters were identical to those in Sect. 6.1.

The zeroth-order scattering source antiderivative function $F^{0,0}_{0,s,\mu\varphi}(r, \mu, \varphi)$, first-order scattering source antiderivative function $F^{1,0}_{0,s,\mu\varphi}(r, \mu, \varphi)$, $F^{1,-1}_{0,s,\mu\varphi}(r, \mu, \varphi)$, $F^{1,1}_{0,s,\mu\varphi}(r, \mu, \varphi)$, and the angular flux density $\phi(r, \mu, \varphi)$ at $r = 0.5$ m are presented in Fig. 8. Additionally, the results for the angular flux density $\phi(r, \mu, \varphi)$ and the flux in the cylindrical geometry for variable $r$ are also provided. In addition, Fig. 8 compares the results of the DL method [39, 40], OpenMC [36], and discrete ordinate discontinuous finite element method (SN-DFEM) [41]. The DL method uses a fully connected neural network to solve the conventional transport equation in a differential-integral form (Eq. 54). The integral terms in Eq. 54 adopts the conventional numerical integration method for finite integral sums as follows: $\int_a^b \phi(x)\mathrm{d}x = \sum_{j=1}^{n} w_j \phi(x_j)$.

Table 7 compares the results and final loss function. $\mathrm{MSE}_{\mathrm{MC}}$ represents the mean-squared error with respect to OpenMC. $\mathrm{MSE}_{\mathrm{SN}}$ represents the mean-squared error with respect to the SN-DFEM. The results indicate that the M-AIM and DL method display comparable accuracies. However, the training time of the M-AIM was approximately 93% shorter than that of the DL method. Compared with the DL method for solving the original differential-integral form of the transport equation (Eq. 54), the M-AIM for solving Eq. 55 has a significant advantage in terms of computational efficiency.

To analyze the performance of the adjustment strategies, several numerical experiments were conducted. We set the weight of the governing equation loss term to one. The other loss terms used different weight values. The first five sets used fixed weight values, whereas the last three sets employed a dynamic adjustment strategy. All the cases were trained 500 times. Table 8 presents the results for different PB weight values. Figure 9 shows the weight values of the PB in the fixed-source homogeneous single-group problem with an infinite cylindrical geometry. The results indicate that the weight values of the loss terms affect the accuracy of the model. Experiments 1–5 could not achieve optimal computational results using fixed weight values. In contrast, experiments 6–8, which employed adaptive adjustment strategies, dynamically adjusted the weight values within the range of 10–50. This yielded a higher precision. Therefore, in actual hyperparameter settings, a dynamic adjustment strategy can be adopted to achieve better computational accuracy. The actual dynamic adjustment strategies depend on specific problems, and the hyperparameters can be determined and optimized based on experiments and experience. No "one-size-fits-all" adjustment strategy exists for any problem.

To summarize, the above verification demonstrates that the proposed M-AIM is applicable to various critical/non-critical neutron transport problems under different conditions including different energy groups, typical geometries (slab, sphere, and cylinder), and one/two-dimensional angular variables. It is also suitable for fixed-source and eigenvalue problems that involve higher-order scattering. It provided continuous numerical solutions for the neutron transport equation under anisotropic scattering conditions. The numerical calculation process was standardized, and the results showed good precision.

However, the aforementioned examples indicate that compared with conventional numerical methods, deep learning requires more time. This is a common challenge confronted by the current numerical methods using deep learning. The directions for improvement include using higher-performance GPUs and adopting large-scale parallel technology to reduce the training time. In addition, improvements can be achieved using transfer learning, training differential operators, and optimizing the structure of deep neural networks to reduce the training time.

## 7 Summary

This study addressed the challenges confronted by the current deep learning techniques when solving the neutron transport equation with anisotropic scattering sources. It introduced the multi-antiderivative transformation alternating iterative deep learning method (M-AIM). The method employs multiple antiderivative functions to transform the transport equation. This transformation converts the integral–differential form of the transport equation into an exact differential equation suitable for deep-learning-based solutions. Subsequently, multiple neural network functions were used to map the unknown angular flux and antiderivative functions. Deep learning was then performed using an alternating iterative method to obtain numerical solutions for each antiderivative function and unknown angular flux density function. Finally, the correctness of the theory and related methods was verified through the numerical results of typical problems. This research has laid the foundation

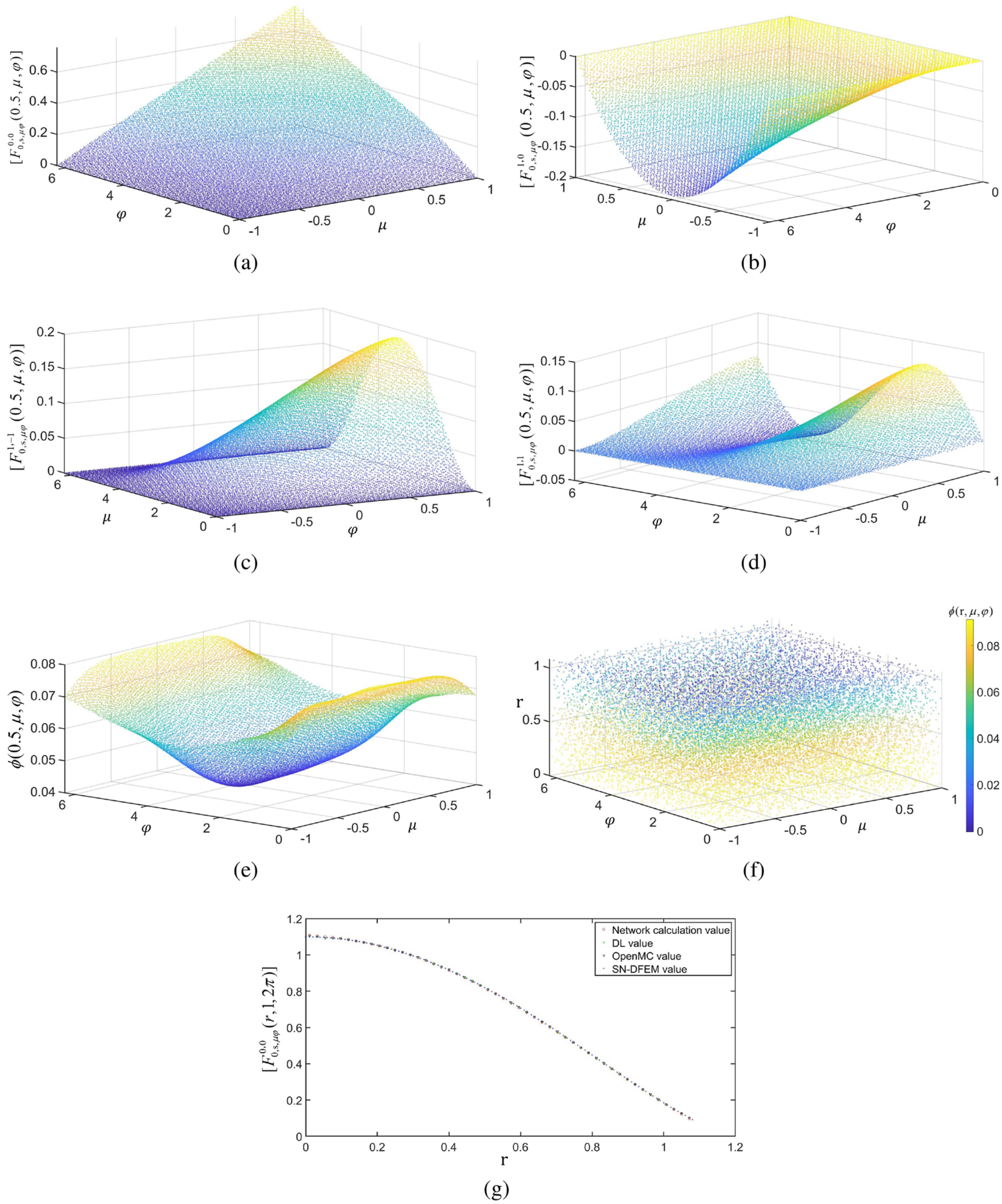



**Fig. 8** (Color online) Flux result of fixed-source single-material cylindrical geometry scattering source anisotropic transport problem. It shows the numerical calculation results including the antiderivative functions of different orders (zeroth-order scattering source/first-order scattering source) and continuous numerical solutions for angular flux. In Fig. 8(g), $[F^{0,0}_{0,\mathrm{s},\mu\varphi}(r,1,2\pi)] = \int_{-1}^{1}\mathrm{d}\mu' \int_{0}^{2\pi} \phi(r,\mu',\varphi')\mathrm{d}\varphi'$. **a** Zeroth-order scattering source antiderivative function $F^{0,0}_{0,\mathrm{s},\mu\varphi}(0.5,\mu,\varphi)$; **b** first-order scattering source antiderivative function $F^{1,0}_{0,\mathrm{s},\mu\varphi}(0.5,\mu,\varphi)$; **c** first-order scattering source antiderivative function $F^{1,-1}_{0,\mathrm{s},\mu\varphi}(0.5,\mu,\varphi)$; **d** first-order scattering source antiderivative function $F^{1,1}_{0,\mathrm{s},\mu\varphi}(0.5,\mu,\varphi)$; **e** angular flux density $\phi(0.5,\mu,\varphi)$; **f** angular flux density $\phi(r,\mu,\varphi)$; and **g** comparison results with DL method and OpenMC and SN-DFEM

**Table 7** Calculation result for fixed-source single-group problem of cylindrical geometry

| Method | $\mathrm{MSE}_{\mathrm{MC}}$ | $\mathrm{MSE}_{\mathrm{SN}}$ | Time (min) | $Loss_{\mathrm{s,all}}^{0,0}$ | $Loss_{\mathrm{s,all}}^{1,0}$ | $Loss_{\mathrm{s,all}}^{1,-1}$ | $Loss_{\mathrm{s,all}}^{1,1}$ | $Loss_{\phi,\mathrm{all}}$ |
|---|---|---|---|---|---|---|---|---|
| M-AIM | $6.4125\times10^{-6}$ | $6.6431\times10^{-6}$ | 144.2 | $1.0242\times10^{-6}$ | $1.1836\times10^{-6}$ | $1.0453\times10^{-6}$ | $2.0745\times10^{-6}$ | $1.8186\times10^{-5}$ |
| DL | $7.9087\times10^{-6}$ | $5.7365\times10^{-6}$ | 2005.7 | / | / | / | / | $1.7098\times10^{-5}$ |

**Table 8** Results of different weight values of PB in the fixed-source homogeneous single-group problem of infinite cylindrical geometry

| Number | Weight | MSEmc | MSEsn | $Loss_{\phi,\mathrm{all}}$ |
|---|---|---|---|---|
| 1 | 100 | $1.1176\times10^{-4}$ | $1.1459\times10^{-4}$ | $4.0336\times10^{-6}$ |
| 2 | 80 | $3.3023\times10^{-4}$ | $3.2613\times10^{-4}$ | $2.0588\times10^{-6}$ |
| 3 | 50 | $1.0796\times10^{-4}$ | $1.0657\times10^{-4}$ | $1.9826\times10^{-6}$ |
| 4 | 30 | $2.0696\times10^{-5}$ | $1.5756\times10^{-5}$ | $1.0382\times10^{-6}$ |
| 5 | 10 | $4.1371\times10^{-5}$ | $4.0113\times10^{-5}$ | $7.6040\times10^{-7}$ |
| 6 | strategy 1 | $1.3919\times10^{-5}$ | $1.0784\times10^{-5}$ | $1.4721\times10^{-6}$ |
| 7 | strategy 2 | $1.6479\times10^{-5}$ | $1.1058\times10^{-5}$ | $7.1233\times10^{-7}$ |
| 8 | strategy 3 | $1.5255\times10^{-5}$ | $1.0176\times10^{-5}$ | $7.6950\times10^{-7}$ |

for solving more complex neutron transport equations in geometry and energy groups using deep learning methods. New technical approaches for numerical calculation methods of anisotropic scattering neutron transport equations were evaluated.

The current numerical solution methods for transport equations using PINN are still in the early stages of research. Numerous unknown challenges need to be addressed urgently. The areas that deserve particular attention include the following:

- Enhancing the training efficiency of deep learning models: As described in the numerical experiments in this paper, the current training time for deep learning models is relatively long, and the convergence speed should be improved significantly. Adopting advanced neural network structures, machine learning optimization algorithms, and large-scale parallel technology based on specialized GPU hardware is recommended to enhance the computational efficiency.
- Strengthening the generalization capability of deep learning models: Currently, deep learning models exhibit good generalization capabilities for non-training data samples within the solution space. However, for regions outside the solution space, particularly for problems with different cross-sections and geometries, the generalization capability of the model is weak. It is necessary to achieve improvements through methods such as transfer learning and the training of differential operators.
- There is an urgent need to perform uncertainty analyses and assessments of deep learning models to enhance their interpretability. Numerical solution methods and the related software for neutron transport, which play a role in reactor engineering, are crucial for nuclear safety. Thus, these require rigorous verification and uncertainty analysis. However, the "black box" characteristics of current deep neural networks complicate this process. This necessitates a thorough uncertainty analysis and assessment before these can be applied in practice. A feasible approach is to provide a comprehensive uncertainty assessment for deep learning models by referring to the

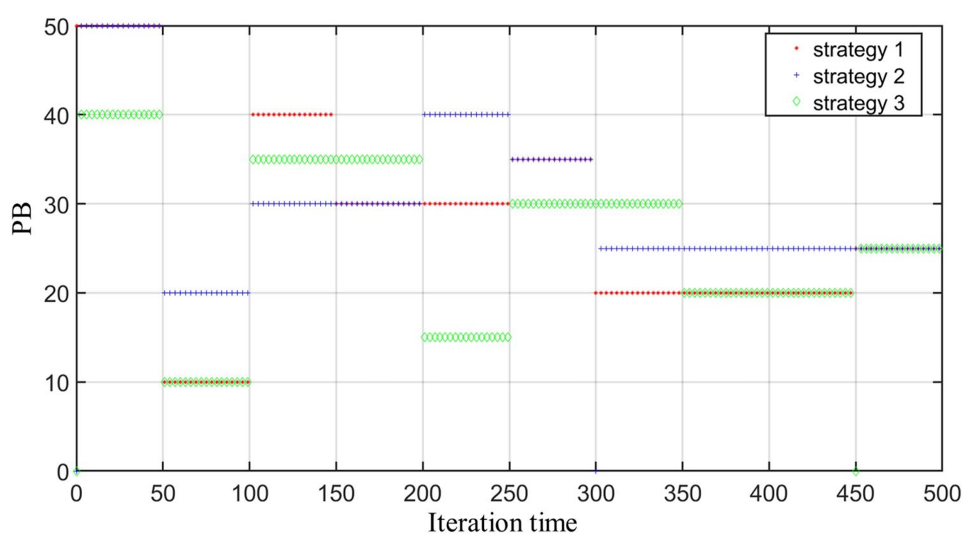



**Fig. 9** (Color online) Different weight values of PB in the fixed-source homogeneous single-group problem of infinite cylindrical geometry

evaluation methods of conventional numerical methods. Furthermore, it is important to develop new methods that can improve the interpretability of deep learning models and ensure their reliability and safety.

**Author contributions** All authors contributed to the study conception and design. Theoretical research was performed by Dong Liu and Bin Zhang. The project support was performed by Qi Luo. Experimental data collection was performed by Bin Zhang, Xian-Tao Cui, and Chen Zhao. The experiment was performed by Yong Jiang, Bin Zhang, Dong Liu, and Heng Zhang. The article revision was performed by Yong Jiang, Bin Zhang and Heng Zhang. The first draft of the manuscript was written by Dong Liu and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

**Data availability** The data that support the findings of this study are openly available in Science Data Bank at https://cstr.cn/31253.11.sciencedb.j00186.00849 and https://www.doi.org/10.57760/sciencedb.j00186.00849.

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

## References


1. H.C. Wu, Y.Q. Zheng, L.Z. Cao, *Deterministic Numerical Methods for Neutron Transport Equations* (Atomic Energy Press, Beijing, 2018). (**in Chinese**)
2. S.H. Du, *Computer Simulation of Transportation Problems* (Hunan Science and Technology Press, Changsha, 1988). (**in Chinese**)
3. H. Zou, Y. Chen, Q. Zhang et al., Heterogeneous parallel high-order scattering MOC and its application to simulation of critical experiment. At. Energy Sci. Technol. **58**(01), 135–143 (2024). https://doi.org/10.7538/yzk.2023.youxian.0099. (**in Chinese**)
4. H. Zou, L. Liang, Q. Zhang et al., Development and verification of 2D kinetic MOC code based on GPU. At. Energy Sci. Technol. **56**(1), 87–95 (2022). https://doi.org/10.7538/yzk.2021.youxian.0502. (**in Chinese**)
5. Z.S. Xie, L. Deng, *Numerical Calculation Method for Neutron Transport Theory*, 2nd edn. (Northwest Polytechnical University Press, Xian, 2005). (**in Chinese**)
6. J. Sirignano, J.F. MacArt, J.B. Freund et al., DPM: a deep learning PDE augmentation method with application to large-eddy simulation. J. Comput. Phys. **423**, 109811 (2020). https://doi.org/10.1016/j.jcp.2020.109811
7. M. Bichuch, J.H. Hou, Deep PDE solution to BSDE. Digit. Financ. **6**, 727–758 (2024). https://doi.org/10.1007/s42521-023-00098-6
8. Z.H. Zhang, M. Xu, Solving multi-dimensional optimal stochastic control problems with deep learning and evolution algorithm. Adv. Appl. Math. **11**(3), 1222–1241 (2022). https://doi.org/10.12677/AAM.2022.113133. (**in Chinese**)
9. J. Pan, Z.L. Guo, S.Z. Chen, A compound neural network for learning partial differential equations from noisy data. Chin. J. Comput. Phys. **39**(2), 223–232 (2022). https://doi.org/10.19596/j.cnki.1001-246x.8364. (**in Chinese**)
10. Q.H. Yang, H. Yang, T. Cui et al., FDM-PINN: physics-informed neural network based on fictitious domain method. Int. J. Comput. Math. **100**(3), 511–524 (2023). https://doi.org/10.1080/00207160.2022.2128674
11. J.T. Xin, H.E. Jia, R.P. Niu, Physics-informed encoder-decoder network based on ConvLSTM for solving time-dependent partial differential equations. Adv. Appl. Math. **13**(2), 714–722 (2024). https://doi.org/10.12677/AAM.2024.132069. (**in Chinese**)
12. M. Raissi, P. Perdikaris, G.E. Karniadakis et al., Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. **378**(1), 686–707 (2019). https://doi.org/10.1016/j.jcp.2018.10.045
13. L. Lu, P.Z. Jin, G.F. Pang et al., Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nat. Mach. Intell. **3**(3), 218–229 (2021). https://doi.org/10.1038/s42256-021-00302-5
14. S.Q. Chun, H. Feng, A.N. Zhang et al., Method of predicting transient thermal hydraulic parameters of the core based on the gated recurrent unit model of soft attention. Nucl. Tech. **47**(1), 010603 (2024). https://doi.org/10.11889/j.0253-3219.2024.hjs.47.010603. (**in Chinese**)
15. J.K. Li, T. Huang, M. Lin et al., Study on calculation method of Courant limit in thermal hydraulic system analysis code. Nucl. Power Eng. **42**(4), 63–67 (2021). https://doi.org/10.13832/j.jnpe.2021.04.0063. (**in Chinese**)
16. W.X. Tian, M.J. Wang, C.J. Zeng et al., Development and application of three-dimensional nuclear steam generator thermal-hydraulic analysis code STAF. At. Energy Sci. Technol. **56**(11), 2239–2252 (2022). https://doi.org/10.7538/yzk.2022.youxian.0590. (**in Chinese**)
17. H. Chen, M.M. Guo, Y. Tian et al., Progress of convolution neural networks in flow field reconstruction. Chin. J. Mech. **54**(9), 2343–2360 (2022). https://doi.org/10.6052/0459-1879-22-130. (**in Chinese**)
18. Y.C. Xie, Y.H. Wang, Y. Ma et al., Neural network based deep learning method for multi-dimensional neutron diffusion problems with novel treatment to boundary. J. Nucl. Eng. **2**(4), 533–552 (2021). https://doi.org/10.3390/jne2040036
19. D. Liu, Q. Luo, L. Tang, Solving multi-dimensional neutron diffusion equation using deep machine learning technology based on PINN model. Nucl. Power Eng. **43**(2), 1–8 (2022). https://doi.org/10.13832/j.jnpe.2022.02.0001. (**in Chinese**)
20. Y. Yang, H.L. Gong, S.Q. Zhang et al., A data-enabled physics-informed neural network with comprehensive numerical study on solving neutron diffusion eigenvalue problems. Ann. Nucl. Energy **183**, 109656 (2023). https://doi.org/10.1016/j.anucene.2022.109656
21. J.Y. Wang, X.J. Peng, Z. Chen et al., Surrogate modeling for neutron diffusion problems based on conservative physics-informed neural networks with boundary conditions enforcement. Ann. Nucl. Energy **176**, 109234 (2022). https://doi.org/10.1016/j.anucene.2022.109234
22. H.L. Gong, S.B. Cheng, Z. Chen et al., Data-enabled physics-informed machine learning for reduced-order modeling digital twin: application to nuclear reactor physics. Nucl. Sci. Eng. **196**(6), 668–693 (2022). https://doi.org/10.1080/00295639.2021.2014752
23. Y. Yang, H.L. Gong, Q.L. He et al., On the uncertainty analysis of the data-enabled physics-informed neural network for solving neutron diffusion eigenvalue problem. Nucl. Sci. Eng. **198**(5), 1075–1096 (2024). https://doi.org/10.1080/00295639.2023.2236840
24. Q.H. Yang, Y. Yang, Y.T. Deng et al., Physics-constrained neural network for solving discontinuous interface K-eigenvalue problem with application to reactor physics. Nucl. Sci. Tech. **34**, 161 (2023). https://doi.org/10.1007/s41365-023-01313-0

25. Y.C. Xie, Y. Ma, Y.H. Wang, Automatic boundary fitting framework of boundary dependent physics-informed neural network solving partial differential equation with complex boundary conditions. Comput. Method. Appl. M. **414**, 116139 (2023). https://doi.org/10.1016/j.cma.2023.116139
26. D. Liu, Z.Z. Qin, A. AN et al., Neural network study of the nuclear ground-state spin distribution within a random interaction ensemble. Nucl. Sci. Tech. **35**, 64 (2024). https://doi.org/10.1007/s41365-024-01424-2
27. Y.S. Hao, Z. Wu, Y.H. Pu et al., Research on inversion method for complex source-term distributions based on deep neural networks. Nucl. Sci. Tech. **34**, 195 (2023). https://doi.org/10.1007/s41365-023-01327-8
28. Z. Chen, L. Liu, L. Mu, Solving the linear transport equation by a deep neural network approach. Discrete. Cont. Dyn-S. **15**(4), 669–686 (2022). https://doi.org/10.3934/dcdss.2021070
29. D. Liu, X.Q. Wang, B. Zhang et al., Differential transform order theory for solving neutron transport equation by deep learning method. At. Energy Sci. Technol. **57**(05), 946–959 (2023). https://doi.org/10.7538/yzk.2023.youxian.0002. (**in Chinese**)
30. Q.X. Cheng, L.S. Wu, X.C. Pang, *Mathematical Analysis* (Higher Education Press, Beijing, 2001). (**in Chinese**)
31. M.T. Hagan, H.B. Demuth, M.H. Beale, *Neural Network Design*, 2nd edn. (China Machine Press, Beijing, 2014). (**in Chinese**)
32. A.G. Baydin, B.A. Pearlmutter, A.A. Radul et al., Automatic differentiation in machine learning: a survey. J. Mach. Learn. Res. **18**(153), 1–43 (2018)
33. L. Yang, X.H. Meng, G.E. Karniadakis, B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. J. Comput. Phys. **425**, 109913 (2021). https://doi.org/10.1016/j.jcp.2020.109913
34. L.D. McClenny, U.M. Braga-Neto, Self-adaptive physics-informed neural networks. J. Comput. Phys. **474**, 111722 (2023). https://doi.org/10.1016/j.jcp.2022.111722
35. D. Liu, L. Tang, P. An et al., The deep learning method to search effective multiplication factor of nuclear reactor directly. Nucl. Power Eng. **44**(05), 6–14 (2023). https://doi.org/10.13832/j.jnpe.2023.05.0006. (**in Chinese**)
36. P.K. Romano, B. Forget, The openMC Monte Carlo particle transport code. Ann. Nucl. Energy **51**, 274–281 (2013). https://doi.org/10.1016/j.anucene.2012.06.040
37. H. Yin, X.J. Liu, T.F. Zhang, An efficient parallel algorithm of variational nodal method for heterogeneous neutron transport problems. Nucl. Sci. Tech. **35**, 69 (2024). https://doi.org/10.1007/s41365-024-01430-4
38. C.X. Wu, M. Zhu, Q.Y. Tan et al., A comprehensive study of non-adaptive and residual-based adaptive sampling for physics-informed neural networks. Comput. Method. Appl. M. **403**, 115671 (2023). https://doi.org/10.1016/j.cma.2022.115671
39. Z. Chen, L. Liu, L. Mu, Solving the linear transport equation by a deep neural network approach. Discrete Cont. Dyn-S. **15**(4), 669–686 (2022). https://doi.org/10.3934/dcdss.2021070
40. Y. Guan, T.T. Fang, D.K. Zhang et al., Solving Fredholm integral equations using deep learning. Int. J. Appl. Comput. Math. **8**(2), 87 (2022). https://doi.org/10.1007/s40819-022-01288-3
41. G.C. Zhang, H. Zhang, TARS: a parallel tetrahedral discontinuous finite element code for the solution of the discrete ordinates neutron transport equation. Ann. Nucl. Energy **196**, 110242 (2024). https://doi.org/10.1016/j.anucene.2023.110242