

## Chapter 2

# Neutron Transport Theory and Simulation

Devoted to describing neutron movement in media and the corresponding laws, neutron transport theory is the research basis for transmutation and activation, radiation damage to materials, radiation dose, and biological safety, among other topics. There are two methods for neutron transport calculation: the Monte Carlo method (also called the probabilistic method or the stochastic method) and the deterministic method. The Monte Carlo method is a numerical method based on probability and statistical theories. It can explicitly describe the characteristics of randomly moving particles and the process of physical experiments. In contrast, in the deterministic method, a group of mathematical-physical equations is first built up to explain the physical characteristics of the target system. Then, by discretizing the variables including direction, energy, space and time in these equations, an approximate solution can be obtained with numerical calculation.

This chapter introduces the basic principles of neutron interactions with matter, neutron transport theory, simulation methods and codes for fusion neutron transport calculations.

### 2.1 Interaction of Neutrons with Matter

Interactions of neutrons with matter mainly occur between the neutrons and the atomic nuclei of the matter [1]. The relevant basic physical quantities include the reaction cross-section, differential cross-section and reaction rate.

There are two types of reaction cross-sections: microscopic cross-section and macroscopic cross-section. The microscopic cross-section represents the effective target area of a single target nucleus for an incident particle. Larger the effective area is, greater would be the reaction probability. By contrast, the macroscopic cross-section represents the effective target area of all of the nuclei contained in a volume of material. For example, consider a plate of area  $A$  and thickness  $x$ , as shown in Fig. 2.1. Suppose that the density of nuclei in the plate is  $N$  ( $\text{n/m}^3$ ). An

incident neutron beam with intensity  $I_0$  ( $\text{n/m}^2 \text{ s}$ ) travels in the direction perpendicular to the plate. Then, the number of times  $P$  that the incident neutrons interact with the nuclei in the plate per unit time is proportional to  $I_0$ ,  $N$ ,  $A$  and  $x$ , as follows:

$$P = \sigma I_0 N A x \quad (2.1)$$

where the proportionality coefficient  $\sigma$  is the microscopic cross-section, in units of  $\text{m}^2$  (in practical applications, the unit is expressed in barn “b”, where  $1\text{b} = 10^{-28} \text{ m}^2$ ). The relationship between the macroscopic cross-section ( $\Sigma$ , in units of  $\text{m}^{-1}$ ), and the microscopic cross-section  $\sigma$  is

$$\Sigma = N\sigma \quad (2.2)$$

After the collision of a neutron with a nucleus, the probability of neutron emission is related to the exit angle of the emitted neutron, which is often described by the differential cross-section, denoted by  $d\sigma/d\Omega$ , with units of  $\text{b/sr}$ . In cases where the probability of neutron emission is related to both the exit angle and the energy, the double differential cross-section, denoted by  $d^2\sigma/(d\Omega dE)$  and expressed in units of  $\text{b/(sr MeV)}$ , is used to describe the emission probability.

The reaction rate ( $\text{n/m}^3 \text{ s}$ ) is the total number of times that neutrons interact with nuclei in a unit volume per unit time and can be expressed as

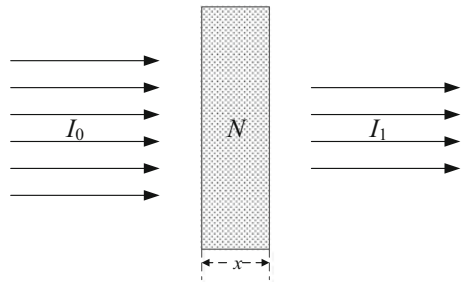
$$R = nv\Sigma \quad (2.3)$$

where  $n$  is the neutron density, which describes the number of neutrons per unit volume ( $\text{n/m}^3$ ), and  $v$  is the velocity of neutron movement ( $\text{m/s}$ ).

The interactions between neutrons and nuclei can be divided into three fundamentally different mechanisms, which are called potential scattering, compound nucleus formation and direct interaction [1].

The simplest type of nuclear reaction that occurs in a nuclear system is potential scattering, in which a neutron scatters off a nuclear potential without ever penetrating the nucleus itself. Potential scattering is a type of elastic scattering that can occur for neutrons of any energy. In this scattering mode, both the total kinetic energy and the total momentum of the system (neutron and target nucleus) are conserved.

**Fig. 2.1** Neutron beam passing through a plate

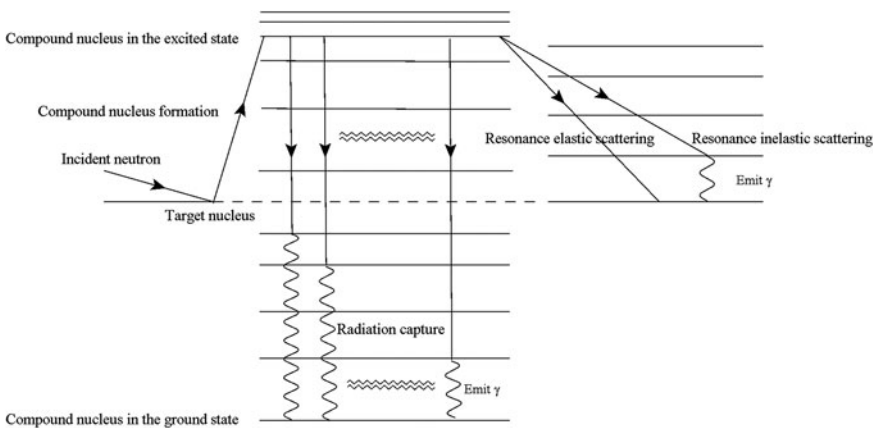


In the formation of a compound nucleus, an incident neutron is absorbed by a target nucleus to form a compound nucleus in an excited state with energy (above the ground state) equal to the sum of the kinetic energy of both the neutron and target nucleus and their binding energy, as shown in Fig. 2.2. The excited compound nucleus may decay or disintegrate through various pathways, such as resonance elastic scattering, resonance inelastic scattering or fission.

Neutrons of comparatively high energy can also interact with nuclei via direct interaction. In this process, an incident neutron collides directly with the nucleons inside a nucleus. As a result, nucleons can be knocked out of the nucleus, whereas the incident neutron may be retained by the target nucleus. After absorbing the kinetic energy of the incident neutron, the target nucleus is excited to some higher energy state. Eventually, this excited nucleus will de-excite by releasing gamma rays.

Depending on the results of the reactions, the interactions between neutrons and nuclei can be classified into two types: neutron scattering and neutron absorption. In neutron scattering, a neutron is released after the reaction, and the difference in energy between the incident and outgoing neutrons becomes internal or kinetic energy of the target nucleus. Neutron scattering (elastic and inelastic scattering) is the main reaction process for neutron moderation. In neutron absorption, the incident neutron is absorbed by the target nucleus, and particles such as photons, protons are released thereafter. Such reactions include  $(n, \gamma)$ ,  $(n, p)$ ,  $(n, d)$ ,  $(n, t)$ ,  $(n, {}^3\text{He})$  and  $(n, \alpha)$ .

The interaction cross-section of a neutron with a nucleus strongly depends on the incident neutron's energy. For an incident neutron with energy between 1 eV and 1 MeV, the cross-section for neutron interaction with a heavy nucleus or medium-mass nucleus may oscillate with the neutron energy. Many large peaks, which are called resonance peaks, appear in the cross-section; this phenomenon is called resonance. During the interaction process, the resonance peaks broaden with increasing temperature; this effect is called Doppler broadening.



**Fig. 2.2** Formation and decay of a compound nucleus

## 2.2 Foundation of Neutron Transport Theory

Transport theory has a history that goes back more than 100 years. In 1872, L. Boltzmann derived the conservation expression for the transport of microscopic molecules in a medium. Similar conservation equations for neutrons, photons and some other particles were derived, and these relations are known as the particles transport equations or Boltzmann equations. The neutron transport equation is the basic equation used to study the neutron transport process in a medium. It is essential to understand neutron transport theory and perform reactor physics analysis.

Seven reasonable assumptions are made to simplify the transport problems when deriving the neutron transport equation [2]: (1) neutrons are considered as points; (2) neutrons are without charge and thus travel in straight lines between point collisions; (3) neutron-neutron interactions are neglected; (4) collisions are instantaneous; (5) the material properties are assumed to be isotropic; (6) the nuclear properties and compositions of the materials under consideration are assumed to be known and time independent unless explicitly stated otherwise; and (7) only the expected or mean value of the neutron density distribution is considered.

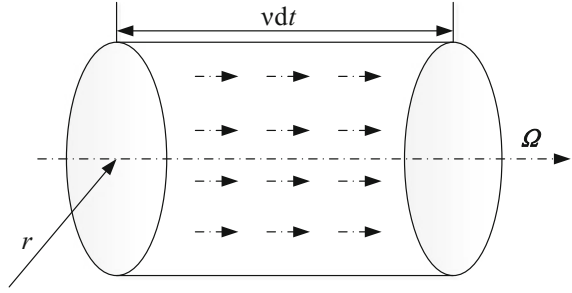
Under these assumptions, the main research aims of neutron transport theory are to calculate the neutron distribution in the medium and to develop corresponding models and methods. To describe the neutron distribution in a medium, it is necessary to obtain the neutron distribution with space coordinates  $\mathbf{r}(x, y, z)$ , energy  $E$ , direction  $\boldsymbol{\Omega}(\theta, \varphi)$  and time  $t$ . This neutron distribution is usually described by the neutron angular density  $n(\mathbf{r}, E, \boldsymbol{\Omega}, t)$ , neutron angular flux  $\phi(\mathbf{r}, E, \boldsymbol{\Omega}, t)$  and neutron current  $\mathbf{J}(\mathbf{r}, E)$ .

The neutron angular density  $n(\mathbf{r}, E, \boldsymbol{\Omega}, t)$  is defined as the probable (or expected) number of neutrons at position  $\mathbf{r}$  with direction  $\boldsymbol{\Omega}$  and energy  $E$  at time  $t$  per unit volume per unit solid angle per unit energy. The commonly used unit of neutron angular density is  $\text{n}/(\text{cm}^3 \text{ sr})$ .

The neutron angular flux  $\phi(\mathbf{r}, E, \boldsymbol{\Omega}, t)$  is defined by the product of the neutron angular density  $n(\mathbf{r}, E, \boldsymbol{\Omega}, t)$  and neutron velocity  $v$ . The unit  $\text{n}/(\text{cm}^2 \text{ sr s})$  is typically used in practical applications. It can be expressed as the number of neutrons at position  $\mathbf{r}$  with energy  $E$  travelling through the unit surface area perpendicular to the direction  $\boldsymbol{\Omega}$  at time  $t$ , as shown in Fig. 2.3.  $\phi(\mathbf{r}, E, \boldsymbol{\Omega}, t)dVd\boldsymbol{\Omega}dEdt$  describes the total path length traveled during  $dt$  by all particles in the incremental phase space volume  $dVd\boldsymbol{\Omega}dE$ .

The neutron flux  $\phi(\mathbf{r}, E, t)$  is obtained by integrating the neutron angular flux over all directions, as shown in Eq. (2.4). The unit then becomes  $\text{n}/(\text{cm}^2 \text{ s})$ . The neutron fluence is then obtained by integrating the neutron flux over time, and the unit is  $\text{n}/\text{cm}^2$ . The neutron flux (i.e., the term commonly used in fusion neutronics) equals the neutron fluence rate according to the following definition.

**Fig. 2.3** Differential neutron beam



$$\phi(\mathbf{r}, E, t) = v n(\mathbf{r}, E, t) = \int_{4\pi} \phi(\mathbf{r}, E, \boldsymbol{\Omega}, t) d\boldsymbol{\Omega} = \int_{4\pi} v n(\mathbf{r}, E, \boldsymbol{\Omega}, t) d\boldsymbol{\Omega} \quad (2.4)$$

The neutron current  $\mathbf{J}(\mathbf{r}, E, t)$  in units of  $\text{n}/(\text{cm}^2 \text{ s})$  is obtained by integrating the quantity  $\boldsymbol{\Omega}\phi(\mathbf{r}, E, \boldsymbol{\Omega}, t)$  over all directions.

$$\mathbf{J}(\mathbf{r}, E, t) = \int_{4\pi} \boldsymbol{\Omega}\phi(\mathbf{r}, E, \boldsymbol{\Omega}, t) d\boldsymbol{\Omega} \quad (2.5)$$

### 2.2.1 Neutron Transport Equation

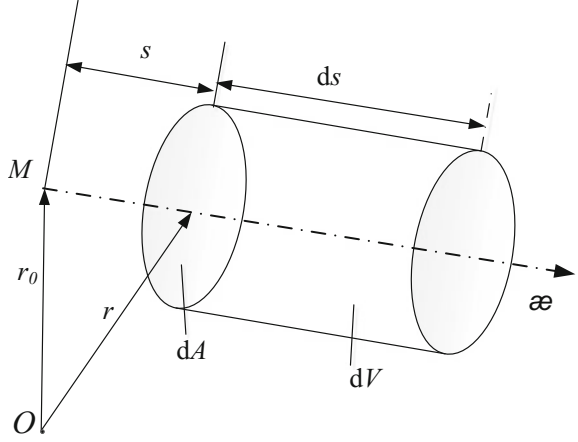
The conservation of the neutron number is a basic principle in the study of neutron transport [2] and means that the rate of change of the neutron angular density with time is equal to the neutron production rate minus the leakage and removal rate in a given micro-element, as shown in Fig. 2.4. In a steady-state system, the rate of change of the neutron angular density with time,  $\partial n/\partial t$ , is zero.

$$\frac{\partial n}{\partial t} = \frac{1}{v} \frac{\partial \phi}{\partial t} = \text{Production rate}(Q) - \text{Leakage rate}(L) - \text{Removal rate}(R) \quad (2.6)$$

#### 1. Production rate

There are three sources of neutrons in  $dVdEd\boldsymbol{\Omega}$ : (1) the scattering source ( $Q_s$ ), which refers to neutrons are produced after incident neutrons scatter from energy  $E'$  and direction  $\boldsymbol{\Omega}'$  to energy  $E$  and direction  $\boldsymbol{\Omega}$ ; (2) the fission source ( $Q_f$ ); and (3) the independent extraneous source ( $S$ ), such as a spontaneous fission source, natural radioactive source and others. Thus, the production rate is

**Fig. 2.4** Neutron conservation in micro-element  $dV$



$$Q = Q_s + Q_f + S = dV dE d\Omega \left[ \int_0^\infty dE' \int_{\Omega'} \Sigma_s(\mathbf{r}, E') f(\mathbf{r}, E' \rightarrow E, \Omega' \rightarrow \Omega) \right. \\ \left. \times \phi(\mathbf{r}, E', \Omega', t) d\Omega' + \frac{\chi(E)}{4\pi} \int_0^\infty dE' \int_{\Omega'} v \Sigma_f(\mathbf{r}, E') \phi(\mathbf{r}, E', \Omega', t) d\Omega' + S(\mathbf{r}, E, \Omega, t) \right] \quad (2.7)$$

where  $f(\mathbf{r}, E' \rightarrow E, \Omega' \rightarrow \Omega)$  is the scattering function, which is defined as the probability of a neutron being scattered from  $(E', \Omega')$  to  $(E, \Omega)$ ;  $\Sigma_s$  is the macroscopic scattering cross-section;  $\Sigma_f$  is the macroscopic fission cross-section;  $\chi(E)$  is the fission spectrum; and  $v$  is the mean number of fission neutrons released per fission. It should be noted that the time-delaying effects of delayed neutrons are neglected in Eq. (2.7).

## 2. Leakage rate

The neutron leakage rate is defined as the difference between the number of neutrons exiting the elemental volume ( $dV$ ) and the number of neutrons entering the elemental volume ( $dV$ ) per unit time. As shown in Fig. 2.4, the number of neutrons entering the volume  $dV$  from the surface  $dA$  at position  $\mathbf{r}_0 + s\Omega$ , with energy from  $E$  to  $E + dE$  having a direction interval  $d\Omega$  at  $\Omega$  per unit time is  $\phi(\mathbf{r}_0 + s\Omega, E, \Omega, t) dA dE d\Omega$ . The number of neutrons streaming out from another surface at position  $\mathbf{r}_0 + (s + ds)\Omega$  of volume  $dV$  with energy from  $E$  to  $E + dE$  having a direction interval  $d\Omega$  at  $\Omega$  per unit time is  $\phi(\mathbf{r}_0 + (s + ds)\Omega, E, \Omega, t) dA dE d\Omega$ . Thus, the number of neutrons leaking out from element  $dV$  per unit time can be expressed as in Eq. (2.8), where the condition of  $ds \rightarrow 0$  should be applied.

$$\begin{aligned}
L &= \phi(\mathbf{r}_0 + (s + ds)\mathbf{\Omega}, E, \mathbf{\Omega}, t) dA dE d\mathbf{\Omega} - \phi(\mathbf{r}_0 + s\mathbf{\Omega}, E, \mathbf{\Omega}, t) dA dE d\mathbf{\Omega} \\
&= \frac{d\phi(\mathbf{r}_0 + s\mathbf{\Omega}, E, \mathbf{\Omega}, t)}{ds} dV dE d\mathbf{\Omega} = \mathbf{\Omega} \cdot \nabla \phi dV dE d\mathbf{\Omega}
\end{aligned} \tag{2.8}$$

### 3. Removal rate

Neutrons can be removed from  $dV dE d\mathbf{\Omega}$  in two ways: (1) neutrons can be absorbed in  $dV$ , and (2) neutrons can be scattered out from energy  $E$  and direction  $\mathbf{\Omega}$  in  $dV$ . Therefore, the neutron number removed from  $dV dE d\mathbf{\Omega}$  per unit time is

$$R = (\Sigma_s + \Sigma_a)\phi(\mathbf{r}, E, \mathbf{\Omega}, t) dV dE d\mathbf{\Omega} = \Sigma_t \phi(\mathbf{r}, E, \mathbf{\Omega}, t) dV dE d\mathbf{\Omega} \tag{2.9}$$

where  $\Sigma_s$  is the macroscopic scattering cross-section,  $\Sigma_a$  is the macroscopic absorption cross-section, and  $\Sigma_t$  is the total macroscopic cross-section.

Substituting Eqs. (2.7), (2.8) and (2.9) into Eq. (2.6), the conservation equation of neutron in phase space ( $\mathbf{r} \times E \times \mathbf{\Omega}$ ) at any instant  $t$  is derived as shown in Eq. (2.10), which is a classical time-dependent differential-integral neutron transport equation. The steady-state neutron transport equation can be obtained by setting  $\partial\phi/\partial t$  to zero.

$$\begin{aligned}
\frac{1}{v} \frac{\partial \phi}{\partial t} + \mathbf{\Omega} \cdot \nabla \phi + \Sigma_t(\mathbf{r}, E)\phi &= \int_0^\infty dE' \int_{\mathbf{\Omega}'} \Sigma_s(\mathbf{r}, E') f(\mathbf{r}, E' \rightarrow E, \mathbf{\Omega}' \rightarrow \mathbf{\Omega}) \\
&\quad \phi(\mathbf{r}, E', \mathbf{\Omega}', t) d\mathbf{\Omega}' + \frac{\chi(E)}{4\pi} \int_0^\infty dE' \\
&\quad \int_{\mathbf{\Omega}'} v \Sigma_f(\mathbf{r}, E') \phi(\mathbf{r}, E', \mathbf{\Omega}', t) d\mathbf{\Omega}' + S(\mathbf{r}, E, \mathbf{\Omega}, t)
\end{aligned} \tag{2.10}$$

The time-dependent integral neutron transport equation can be obtained by integrating the Eq. (2.10) along the characteristic curve [2].

$$\phi(\mathbf{r}, E, \mathbf{\Omega}, t) = \int_0^\infty \exp\left[-\int_0^l \Sigma_t(\mathbf{r} - l'\mathbf{\Omega}, E) dl'\right] Q\left(\mathbf{r} - l\mathbf{\Omega}, E, \mathbf{\Omega}, t - \frac{l}{v}\right) dl \tag{2.11}$$

where  $Q$  is the neutron source term, including the extraneous neutron source, scattering source and fission source. For a pure fusion reactor, the source term contains a scattering source and a neutron source generated from the fusion reaction, but it does not contain any fission source (because there is no fissile or fissionable material). For a fusion-driven subcritical system, which is also known as a hybrid nuclear energy system, fissionable materials in the breeding or transmutation blanket introduce a fission source term in addition to the scattering source and extraneous fusion source.

#### 4. Boundary and initial conditions

To solve the neutron transport equation, boundary and initial conditions are necessary. Commonly used boundary conditions are as follows:

- (1) The neutron angular flux must be a finite, nonnegative real number in regions where the equation applies.
- (2) The neutron angular flux is continuous across the boundary between two different media if there is no source at the interface.
- (3) The outer surface boundary conditions (i.e., explicit boundary conditions) can be divided into two types: (i) If no neutrons enter from external regions, or if a neutron cannot return once it leaves the surface, then the surface is called a free surface, in which the incident neutron angular flux is zero. A convex surface that meets the vacuum is one example. (ii) The second type is a source surface with a given incident flux. Usually, the incident flux is treated as an imaginary surface source, making the source surface being treated as a free surface.
- (4) Implicit boundary conditions can be categorized into three types according to the relationship between the incident and outgoing neutron fluxes: (1) The incident neutron flux in one direction is equal to the outgoing neutron flux in the reflected direction on the reflective boundary that appears on the symmetry plane of the symmetry system. (2) The incident flux on the albedo boundary, is equal to a known isotropic albedo,  $\alpha(E)$ , times the outgoing flux on the same boundary in the direction corresponding to spectral reflection. The special case of  $\alpha = 1.0$  is referred to as a reflective boundary condition since all outgoing particles are “reflected” back. (3) All particles passing out of volume  $V$  through a surface element return with an isotropic distribution on the white boundary.

To solve the time-dependent neutron transport equation, the initial condition (i.e., the angular neutron flux distribution in the phase space at the initial moment) must be given.

$$\phi(\mathbf{r}, E, \boldsymbol{\Omega}, t)|_{t=0} = \phi_0(\mathbf{r}, E, \boldsymbol{\Omega}) \quad (2.12)$$

When performing deep-penetration simulations for radiation shielding, sensitivity and uncertainty analysis, and neutron kinetics analysis, among others, the adjoint neutron flux is often used to characterize the contribution of neutrons with position  $\mathbf{r}$ , time  $t$ , energy  $E$  and direction  $\boldsymbol{\Omega}$  to the target response. The following is a brief introduction. The adjoint neutron flux is also known as the neutron importance function.

The adjoint neutron transport equation, which is also known as the neutron importance conservation equation, can be derived according to the mathematical conjugation of the neutron transport equation (also called the neutron importance conservation equation) [2].



$$\begin{aligned}
-\frac{1}{v} \frac{\partial \phi^*}{\partial t} - \boldsymbol{\Omega} \cdot \nabla \phi^* + \Sigma_t \phi^* &= \int_0^\infty dE' \int_{\boldsymbol{\Omega}'} \Sigma_s(\mathbf{r}, E) \\
f(\mathbf{r}, E \rightarrow E', \boldsymbol{\Omega} \rightarrow \boldsymbol{\Omega}') \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) d\boldsymbol{\Omega}' & \\
+ \frac{v \Sigma_f(\mathbf{r}, E)}{4\pi} \int_0^\infty dE' \int_{\boldsymbol{\Omega}'} \chi(E') \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) d\boldsymbol{\Omega}' + S^* &
\end{aligned} \tag{2.13}$$

By solving Eq. (2.13), the adjoint neutron flux can be obtained.

Because the neutrons leaving the surface of the system no longer contribute to the system, the boundary condition at boundary  $\Gamma$  of the region is

$$\phi^*(\mathbf{r}, E, \boldsymbol{\Omega}, t) = 0, \quad \text{if } (\boldsymbol{\Omega} \cdot \mathbf{n}) > 0 \quad \text{and} \quad \mathbf{r} \in \Gamma \tag{2.14}$$

The physical meaning of the time-independent adjoint function can be understood by considering a steady-state system containing an arbitrary steady source  $S(\mathbf{r}, E, \boldsymbol{\Omega})$ .  $S^*$  characterizes a certain response. For instance, in a fission reaction,  $S^* = \Sigma_f$  and  $\phi^*(\mathbf{r}, E, \boldsymbol{\Omega})$  represents the contribution of a neutron with position  $\mathbf{r}$ , energy  $E$ , and direction  $\boldsymbol{\Omega}$  to a certain response.

For time-independent multiplying media without an extraneous neutron source,  $S^* = 0$  and  $\phi^*(\mathbf{r}, E, \boldsymbol{\Omega})$  denotes the response of a neutron with position  $\mathbf{r}$ , energy  $E$ , and direction  $\boldsymbol{\Omega}$  to nuclear power.

### 2.2.2 Fixed Source and Eigenvalue Problems

The neutron transport equation can be expressed in operator notation as follows:

$$\frac{1}{v} \frac{\partial \phi}{\partial t} + \mathbf{M} \phi = \mathbf{F} \phi + \mathbf{S} \tag{2.15}$$

where  $\phi = \phi(\mathbf{r}, E, \boldsymbol{\Omega}, t)$ ,  $\mathbf{M}$  is the transport operator and  $\mathbf{F}$  is the fission operator.

$$\begin{aligned}
\mathbf{M} \phi &= \boldsymbol{\Omega} \cdot \nabla \phi + \Sigma_t(\mathbf{r}, E) \phi - \int_0^\infty dE' \int_{\boldsymbol{\Omega}'} \Sigma_s(\mathbf{r}, E') \\
&\quad f(\mathbf{r}, E' \rightarrow E, \boldsymbol{\Omega}' \rightarrow \boldsymbol{\Omega}) \phi(\mathbf{r}, E', \boldsymbol{\Omega}', t) d\boldsymbol{\Omega}'
\end{aligned} \tag{2.16}$$

$$\mathbf{F} \phi = \frac{\chi(E)}{4\pi} \int_0^\infty dE' \int_{\boldsymbol{\Omega}'} v \Sigma_f(\mathbf{r}, E') \phi(\mathbf{r}, E', \boldsymbol{\Omega}', t) d\boldsymbol{\Omega}' \tag{2.17}$$

Solutions to the neutron transport equation can be divided into fixed source problems and eigenvalue problems according to whether an external neutron source exists. If neutrons from a time-independent external source are supplied, the subcritical system will eventually reach an equilibrium state characterized by a time-independent flux distribution in which the production rate of external and fission neutrons is in equilibrium with the absorption and leakage. However, if the system is critical or supercritical, no such equilibrium can exist in the presence of an external source; thus, the neutron flux distribution will be an increasing function of time. This is the so-called fixed source problem. Eigenvalue problems need to be solved (i.e., via criticality calculation) to determine whether a system is subcritical or supercritical and at which level.

A key step in fixed source problems is obtaining the neutron flux. When the introduced external source is time independent, the system will eventually reach a steady state for a subcritical multiplying system. For system without a multiplying medium, such as a fusion reactor, the fission neutron source  $Q_f$  is zero. Then, the fixed source problem is expressed using the following inhomogeneous equation:

$$\mathbf{M}\phi = \mathbf{S} \quad (2.18)$$

For systems with a multiplying medium, such as fusion-fission hybrid reactors, the fixed source problem is expressed by the following inhomogeneous equation:

$$\mathbf{M}\phi = \mathbf{F}\phi + \mathbf{S} \quad (2.19)$$

Eigenvalue problems can be categorized into  $k$ -eigenvalue problems and  $\alpha$ -eigenvalue problems. Taking a  $k$ -eigenvalue problem as an example, it is necessary to solve the eigenvalue and the eigenfunction of the homogeneous equation (2.20). For instance, the effective multiplication factor  $k_{\text{eff}}$  and neutron flux of the breeding or transmutation blanket must be determined for fusion-fission hybrid reactors.

$$\mathbf{M}\phi = \frac{1}{k_{\text{eff}}} \mathbf{F}\phi \quad (2.20)$$

In a fusion-fission hybrid reactor that contains an external neutron source generated by the fusion reaction, the effective multiplication factor  $k_{\text{eff}}$  is not sufficient to characterize the breeding property of the system. Therefore, the source multiplication factor  $k_s$  is introduced to demonstrate the breeding property of the system with  $k_{\text{eff}}$ . Thus,  $k_s$  characterizes the breeding property of the system by considering the external neutron source.

$$k_s = \frac{\mathbf{F}\phi}{\mathbf{M}\phi} = \frac{\mathbf{F}\phi}{\mathbf{F}\phi + \mathbf{S}} \quad (2.21)$$

## 2.3 Neutron Transport Computational Methods

The neutron transport equation contains many independent variables, such as space coordinates  $\mathbf{r}(x, y, z)$ , energy  $E$ , neutron motion direction  $\mathbf{\Omega}(\theta, \varphi)$  and time  $t$ . In practical problems, the complexity of geometry and structure, heterogeneity of spatial distribution of the materials and dependence of cross-sections on the neutron energy for various materials (fissile and non-fissile) should all be considered. Thus, it is very difficult to obtain an exact analytical solution for the energy-dependent neutron transport equation, even in the steady-state case. Therefore, approximate numerical methods must be adopted to solve the transport equation.

The numerical methods used to solve the neutron transport equation are generally divided into the Monte Carlo method and the deterministic method. The Monte Carlo method can be used to address any complex geometry, and its results are more accurate than those of the deterministic method. In contrast, the deterministic method can quickly obtain the approximate solution of the transport equation by discretizing the direction, energy, space and time. Compared with fission reactors, fusion systems are characterized by complex geometries, complicated neutron spectrum and anisotropic neutron scattering. Because it uses the continuous-energy cross-section and accurate geometric descriptions, the Monte Carlo method has prominent advantages for neutron transport simulations for fusion reactors. However, some problems, such as the slow convergence rate and difficulty in addressing problems of deep penetration, still exist. The deterministic method is faster, but fall short in addressing fusion systems with complex geometries, strong anisotropy of neutron scattering and complicated energy spectrum. In recent years, the method of characteristics (MOC) and the discrete ordinates method with unstructured meshes have been developed with improved geometric processing abilities. However, problems, such as ray effects and the high cost of large-scale systems, still need to be solved. Therefore, currently, it is difficult for the deterministic method to meet the requirements for neutron transport calculations in fusion systems. The Monte Carlo-deterministic coupling method, which combines the advantages of both the Monte Carlo and deterministic methods, is one of the most efficient and accurate methods for solving transport problems in fusion systems [3, 4].

### 2.3.1 Monte Carlo Methods

With the development of science and technology and the improvement of computer performance, the Monte Carlo method was proposed and developed as an independent method [4] and was first applied in nuclear weapon research. The basic idea underlying the Monte Carlo method is as follows: When the solution to a problem is the probability of an event or the mathematical expectation of a random variable, the occurrence frequency of the event or the arithmetical mean of several

specific observations of the random variable can be obtained through numerical experiments. Thus, a solution to that problem is obtained.

The advantages of Monte Carlo methods include realistic descriptions of the characteristics of objects with stochastic natures, the ability to simulate physical experiments, few limitations on geometric conditions and parallel computing adaptability. The disadvantages of Monte Carlo methods include the slow convergence rate and statistical uncertainty [2].

The following section is composed of three parts. The first part discusses the basic principles of Monte Carlo neutron transport calculations. The second part introduces the physical quantity estimation methods, and the last part presents the main acceleration methods, including variance reduction techniques, efficient geometric processing methods, source convergence acceleration methods and parallel computing methods.

### 2.3.1.1 Basic Principles

When the Monte Carlo method is applied to neutron transport simulations, the random motion history of a single neutron in a certain geometry is simulated first, and then, sufficient random experimental values (or sampled values) are obtained by tracking a large number of neutron histories. Finally, an estimator of the numerical characteristics of a random variable is obtained as the solution to the problem.

Three processes are involved in solving the neutron transport problem: (1) source sampling from its probability distribution; (2) tracking of the neutrons' locations, energies and directions; and (3) contribution recording and result analysis.

#### 1. Source sampling

The state of the source neutron is determined by seven variables. Thus, before the transport simulation of a neutron starts, source sampling from its probability distribution is required. In a steady-state transport problem, sampling the source position, energy and direction is considered. The types of source position distribution include point source distribution, surface source distribution and volume source distribution. The types of source energy distribution include single energy distribution and energy spectrum distribution, and the types of source direction distribution include single direction distribution, isotropic distribution and anisotropic distribution. In the following, the source sampling process is demonstrated using an example of an isotropic uniform distribution volume source with a single energy ( $E_0$ ).

Sampling for the source position: Consider a sphere with radius  $R$  and center at the origin, as shown in Fig. 2.5. Three random numbers  $\xi_1$ ,  $\xi_2$ , and  $\xi_3$  are generated from a uniform distribution  $U(0,1)$ , and then,  $r = R \times \max(\xi_1, \xi_2, \xi_3)$  (or  $r = R \times \xi_1^{1/3}$ ).  $\theta$  and  $\varphi$  are sampled from an isotropic distribution, where  $\cos \theta$  and  $\varphi$  obey the uniform distribution  $U(-1, 1)$  and  $U(0, 2\pi)$ , respectively. Random

numbers  $\xi_4$  and  $\xi_5$  are sampled from the uniform distribution  $U(0, 1)$ , and then,  $\varphi = 2\pi\xi_4$ , and  $\cos \theta = 2\pi\xi_5 - 1$ . Thus, the three-dimensional (3D) coordinates of the source are  $(r \sin \theta \cos \varphi, r \sin \theta \sin \varphi, r \cos \theta)$ .

Sampling for the source energy: The source neutron energy is  $E_0$ .

Sampling for the source direction: Assuming that the source direction is  $(u, v, w)$ , let us find the relationship between  $(u, v, w)$  and random numbers. Here, two random number  $\xi_1$  and  $\xi_2$  are generated from the uniform distribution  $U(0, 1)$ . Let  $t_1 = 2\xi_1 - 1$ ,  $t_2 = 2\xi_2 - 1$  and  $s = t_1^2 + t_2^2$ ; if  $s \leq 1$ , then  $u = 2s - 1$ ,  $t_3 = [(1 - u^2)/s]^{1/2}$ ,  $v = t_1 t_3$  and  $w = t_2 t_3$ .

## 2. Tracking neutrons' locations, energies and directions

Neutron interacts with materials after covering a certain flight distance. The flight distance  $l$  is sampled as

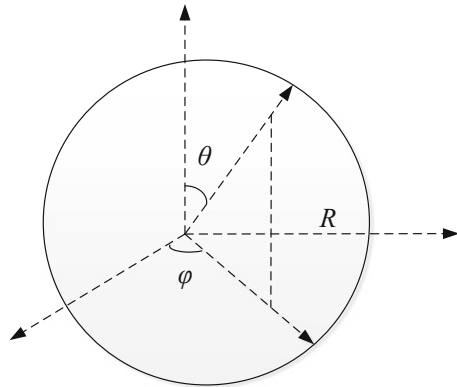
$$l = -\frac{1}{\Sigma_t} \ln(\xi), \quad (2.22)$$

where  $\Sigma_t$  is the total macroscopic cross-section, and  $\xi$  is a uniform distribution random number in the interval  $[0, 1]$ .

If the substance is a compound or mixture, the nuclide that the neutron reacts with must be identified. Assuming that the substance consists of  $n$  species of nuclides with macroscopic cross-sections of  $\Sigma_1, \Sigma_2, \dots, \Sigma_n$ , the total cross-section is  $\Sigma_t = \Sigma_1 + \Sigma_2 + \dots + \Sigma_n$ . Then, the reaction nuclide can be determined by the following sampling method:

$$\text{Reaction nuclide} = \begin{cases} \text{1st nuclide} & 0 \leq \xi < \frac{\Sigma_1}{\Sigma_t} \\ \text{2nd nuclide} & \frac{\Sigma_1}{\Sigma_t} \leq \xi < \frac{\Sigma_1 + \Sigma_2}{\Sigma_t} \\ \dots & \dots \\ \text{nth nuclide} & \frac{\sum_{i=1}^{n-1} \Sigma_i}{\Sigma_t} \leq \xi < 1 \end{cases} \quad (2.23)$$

**Fig. 2.5** Uniformly distributed spherical source



After identifying the collision nuclide, the reaction type (e.g., elastic scattering, inelastic scattering or nuclear fission) must be determined. Assuming that the microscopic cross-sections are  $\sigma_e$ ,  $\sigma_{in}$ , ...,  $\sigma_f$ , the total cross-section is  $\sigma_t = \sigma_e + \sigma_{in} + \dots + \sigma_f$ . Then, the reaction type is sampled by

$$\text{Reaction type} = \begin{cases} \text{Elastic scattering} & 0 \leq \xi < \frac{\sigma_e}{\sigma_t} \\ \text{Inelastic scattering} & \frac{\sigma_e}{\sigma_t} \leq \xi < \frac{\sigma_e + \sigma_{in}}{\sigma_t} \\ \dots & \dots \\ \text{Nuclear fission} & 1 - \frac{\sigma_f}{\sigma_t} \leq \xi < 1 \end{cases} \quad (2.24)$$

After identifying the reaction type, the neutron energy and direction after the reaction can be determined by sampling. If the neutron is absorbed, its history is terminated. Otherwise, the energy and direction of the scattered neutron must be elucidated according to the reaction type. After the elastic scattering, the scattering angle is first obtained by sampling from the differential cross-section, and then, the scattering energy is calculated according to the conservation of momentum and kinetic energy. The method used to identify the neutron energy and direction of inelastic scattering is similar to that used for elastic scattering, but the threshold energy and excited state energies of different energy levels must be considered. The energy of the neutron emitted from the nuclear fission reaction is determined by the fission spectrum, and the direction is sampled according to the isotropic distribution in the laboratory system.

### 3. Contribution recording and result analysis

By tracking a large number of neutron histories and recording the contribution of each neutron, physical quantities, such as the neutron flux, dose rate, energy deposition, reaction rates, and eigenvalues, can be obtained by statistical methods, and the statistical errors on these quantities can be determined. The statistical method is described in the following section, and examples for the estimation of the neutron flux and eigenvalue  $k_{eff}$  are presented.

#### 2.3.1.2 Estimation Methods for Physical Quantities

##### 1. Estimation of the neutron flux

Neutron flux calculations are essential in neutron transport simulations. Three types of flux exist: point flux, surface flux and cell flux. The methods used to estimate point flux include the pointing probability method and the reciprocity method. The estimation methods for surface flux include analytical estimator, the weighting method, the point flux substitution method, and the cell flux substitution method. Finally, those used to estimate the cell flux include analytical estimator, the track

length estimator, the collision estimator, the absorption estimator, and the point flux substitution method. In this section, cell flux estimation is taken as an example.

The cell flux in a given cell (volume  $V$ ) is obtained by

$$\phi(V) = \frac{1}{V} \int_V \phi(\mathbf{r}, E, \boldsymbol{\Omega}) dV dE d\boldsymbol{\Omega} \quad (2.25)$$

where the angular flux  $\phi(\mathbf{r}, E, \boldsymbol{\Omega})$  can be expressed as the sum of the contributions of each scattering.

$$\phi(\mathbf{r}, E, \boldsymbol{\Omega}) = \frac{1}{N} \sum_{i=1}^N \phi_i(\mathbf{r}, E, \boldsymbol{\Omega}) = \frac{1}{N} \sum_{i=1}^N \sum_{n=1}^M \phi_{i,n}(\mathbf{r}, E, \boldsymbol{\Omega}) \quad (2.26)$$

where  $\phi_i(\mathbf{r}, E, \boldsymbol{\Omega})$  represents the track length of the  $i$ th neutron at the phase space  $dV dE d\boldsymbol{\Omega}$ .  $\phi_{i,n}(\mathbf{r}, E, \boldsymbol{\Omega}) dV dE d\boldsymbol{\Omega}$  is the average track length of the  $i$ th neutron between the  $n$ th and  $(n + 1)$ th scatterings at micro-element  $dV dE d\boldsymbol{\Omega}$ .

The following is an example of cell flux estimation using the track length estimator. Assuming that the track length from the  $n$ th to the  $(n + 1)$ th scattering of the  $i$ th neutron is  $s$ , the range of volume element  $V$  is  $s_1 \sim s_2$ , and the flux contribution of the  $n$ th scattering is

$$\phi_{i,n}(V) = \begin{cases} w_n(s - s_1)/V & s_1 \leq s \leq s_2 \\ w_n(s_2 - s_1)/V & s > s_2 \\ 0 & s < s_1 \end{cases} \quad (2.27)$$

where  $w_n$  is the neutron weight after the  $n$ th scattering.

Equations (2.26) and (2.27) can be used to calculate the contribution of a single neutron to the flux ( $\phi_i$ ). Then, the neutron flux in a certain cell can be obtained by Monte Carlo simulation of  $N$  neutrons.

$$\bar{\phi} = \frac{1}{N} \sum_{i=1}^N \phi_i \quad (2.28)$$

In the Monte Carlo calculation, the estimated value of the variance is required, in addition to the estimated flux value for a certain cell. The estimated variance ( $S^2$ ) is

$$S^2 = \frac{1}{N-1} \sum_{i=1}^N (\phi_i - \bar{\phi})^2 \quad (2.29)$$

where  $S$  is the standard deviation of the  $N$  neutrons' contributions. Thus, the statistical error of  $\bar{\phi}$  is

$$S_{\bar{\phi}} = S \frac{1}{\sqrt{N}} \quad (2.30)$$

## 2. Eigenvalue estimation

Theoretically, the effective multiplication factor  $k_{\text{eff}}$  is defined as the ratio of the neutron counts of two adjacent generations. From neutron transport theory, this quantity can be expressed as

$$k_{\text{eff}} = \lim_{n \rightarrow \infty} k^{(n+1)} = \lim_{n \rightarrow \infty} \frac{\int_{4\pi} \int_0^{E_{\text{max}}} \int_V v(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') \phi^{(n+1)}(\mathbf{r}, E', \boldsymbol{\Omega}') d\mathbf{r} dE d\boldsymbol{\Omega}'}{\int_{4\pi} \int_0^{E_{\text{max}}} \int_V v(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') \phi^{(n)}(\mathbf{r}, E', \boldsymbol{\Omega}') d\mathbf{r} dE d\boldsymbol{\Omega}'} \quad (2.31)$$

where

$$\begin{aligned} \phi^{(n+1)}(\mathbf{r}, E, \boldsymbol{\Omega}) = & \int_0^\infty e^{-\tau(\mathbf{r}, E, \boldsymbol{\Omega})} \left[ \frac{1}{k^{(n)}} \frac{\chi(\mathbf{r}, E' \rightarrow E)}{4\pi} \int_0^{E_{\text{max}}} \int_{4\pi} v(\mathbf{r}', E') \Sigma_f(\mathbf{r}', E') \right. \\ & \times \phi^{(n)}(\mathbf{r}', E', \boldsymbol{\Omega}') d\boldsymbol{\Omega}' dE' + \int_0^{E_{\text{max}}} \int_{4\pi} \Sigma_s(\mathbf{r}'; E', \boldsymbol{\Omega}' \rightarrow E, \boldsymbol{\Omega}) \\ & \left. \phi^{(n)}(\mathbf{r}', E', \boldsymbol{\Omega}') d\boldsymbol{\Omega}' dE' \right] d\mathbf{l} \end{aligned} \quad (2.32)$$

The exponential iteration method is widely used to solve  $k$ -eigenvalue problems in Monte Carlo eigenvalue calculations. The neutrons and their respective secondary neutrons are all simulated generation by generation. During the simulation of one generation, the neutrons generated from the fission reaction are stored as the source neutrons for the next generation. Since the fission source spatial distribution of the initial generations (i.e., the non-active generations) is not convergent, and the statistical error is large, the  $k_{\text{eff}}$  of one cycle begins to accumulate from the active generation. The collision estimator, absorption estimator, track length estimator and their combinations are often used to estimate  $k_{\text{eff}}$  [4, 5].

### 2.3.1.3 Monte Carlo Acceleration Method

Considering the slow convergence of the traditional Monte Carlo method, a number of methods have been proposed to accelerate Monte Carlo transport calculations. These methods include variance reduction techniques, efficient geometry processing methods, source convergence acceleration methods and parallel computing methods.



### 1. Variance reduction techniques

During a transport simulation, the statistical error is generally required to be less than a certain value. For example, in a point detector tally, the general requirement is that the statistical error be less than 5%. Because it is proportional to  $1/\sqrt{N}$ , the statistical error can be reduced by increasing the number of simulated particles or using variance reduction techniques. In variance reduction techniques, the stochastic probability model that makes the variance as low as possible is selected to reduce the statistical error while ensure that the calculation result is unbiased. Variance reduction techniques are divided into two varieties: basic variance reduction techniques and adaptive variance reduction techniques.

#### (1) Basic variance reduction techniques

Basic variance reduction techniques are divided into truncation methods, population control methods, modified sampling methods, and partially deterministic methods, among others.

Truncation methods are the simplest variance reduction methods. These methods accelerate calculations by truncating parts of the phase space that do not contribute significantly to the solution. Typical cutoff methods include the energy cutoff and time cutoff methods. In the energy cutoff method, particles with energy less than the cutoff value are killed to reduce the simulation time cost associated with each particle. In contrast, in the time cutoff method, particles with transport simulation times exceeding the cutoff value are killed to reduce the simulation time of each particle.

In population control methods, particle splitting and Russian roulette are used to control the numbers of sampled particles in various regions of the phase space. In important regions, many sampled particles with low weights are tracked, whereas in unimportant regions, few sampled particles with high weights are tracked. Weight adjustment is performed to ensure that the problem solution remains unbiased. Typical population control methods include geometry/energy/time split, Russian roulette, weight cut and weight window. The weight window is a combination of the split and Russian roulette methods. The basic idea is to define the upper and lower bounds of the weight window in space, energy and time. If the particle weight is less than the lower bound of the weight window, Russian roulette occurs; then, the particle is either killed or the particle weight increased to be within the range of the weight window. If the particle weight exceeds the weight window upper limit, the particle is split up so that the weights of all split particles lie within the weight window. If the particle weight is within the weight window, the particle will continue to be transported without any special treatment. Using reasonable settings for the weight window parameters can effectively reduce the variance in the Monte Carlo transport calculation results.

In modified sampling methods, the sampling processes are altered to increase the contribution of each particle to tallies. For any random event, it is possible to sample from any arbitrary distribution rather than the physical probability as long as the particle weights are adjusted to compensate. Thus, in modified sampling

methods, sampling is performed from distributions that send particles in desired directions or into other desired regions of the phase space, such as time or energy, or that change the location or type of collisions. Typical modified sampling methods include exponential transformations, implicit capture, forced collisions, and source biases.

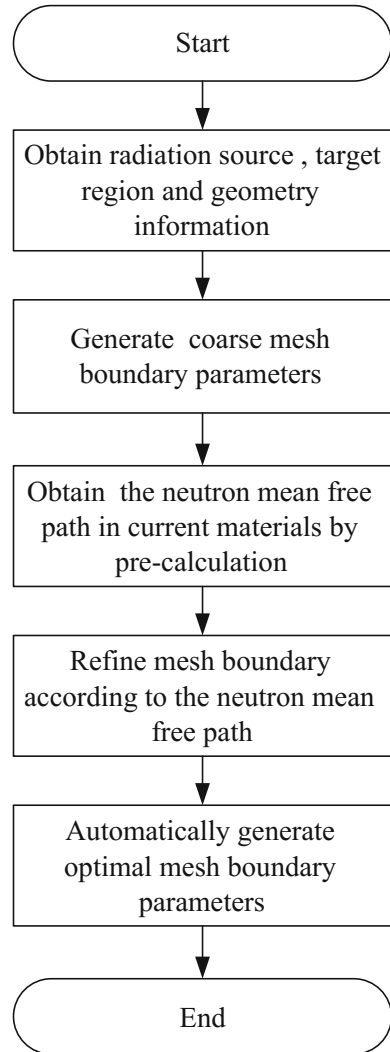
Partially-deterministic methods are the most complicated class of variance reduction techniques. They avoid the normal random walk process by using deterministic-like techniques, such as next-event estimators, or by controlling the random number sequence. Typical partially-deterministic methods include point detectors and correlated sampling.

## (2) Adaptive variance reduction techniques

The basic variance reduction techniques usually require the user to set parameters manually; thus, they are dependent on the user's knowledge and experience. Therefore, adaptive variance reduction techniques that can automatically generate variance parameters and adapt to different problems must be developed. According to the tally targets, adaptive variance reduction techniques can be divided into local adaptive variance reduction techniques and global adaptive variance reduction techniques. The local adaptive variance reduction techniques include Consistent Adjoint Driven Importance Sampling (CADIS) [6], which was developed by Oak Ridge National Laboratory (ORNL); the automatic adaptive mesh generation method, and CADIS based on weight window smoothing [7], which were developed by the Institute of Nuclear Energy Safety Technology (INEST), Chinese Academy of Sciences-FDS Team (referred to as the FDS Team). The global adaptive variance reduction techniques include Forward Weight-CADIS (FW-CADIS) [8] (developed by ORNL), the Method of Automatic Generation of Importance by Calculation (MAGIC) [9] (developed by the Culham Centre for Fusion Energy, CCFE) and the Global Weight Window Generator (GWWG) [10] (developed by the FDS Team). The following sections describe adaptive variance reduction techniques, including the automatic adaptive mesh generation method, CADIS method based on weight window smoothing and GWWG method, as examples.

In the automatic adaptive mesh generation method, the mesh can be generated automatically according to the neutron mean free path during the transport process, thereby eliminating the dependence of the weight window parameters on the user's experience. Additionally, proper numbers of collisions are guaranteed in every weight window mesh to ensure the validity of the statistical results. The flow chart of automatic adaptive mesh generation method is shown in Fig. 2.6. Firstly, according to the geometry characteristics of the target area and radiation source distribution, the interested mesh region is determined, and the parameters of the coarse mesh boundary are obtained. Then, according to the radiation source energy, the spatial scale of the problem and the geometric complexity, the neutron mean free path in the current region is obtained by pre-calculation. Finally, the optimal mesh boundary parameters are automatically generated after refining the mesh boundary according to the neutron mean free path.

**Fig. 2.6** Flow chart of the adaptive mesh generation method



The CADIS method based on weight window smoothing is a local variance reduction technique that involves deterministic pre-computation. In this method, the computer-aided design (CAD) model is first converted into a deterministic calculation model. Then, the adjoint calculation is performed using the deterministic method to obtain the adjoint flux. During the calculation of the lower bound of the weight window, the particle is split excessively in the region with drastically changing adjoint flux, causing the computational efficiency to decrease. Therefore, the smoothing factor  $\rho$  ( $0 < \rho \leq 1$ ) is introduced, and the lower bound of the weight window  $W'_L$  is calculated as follows:

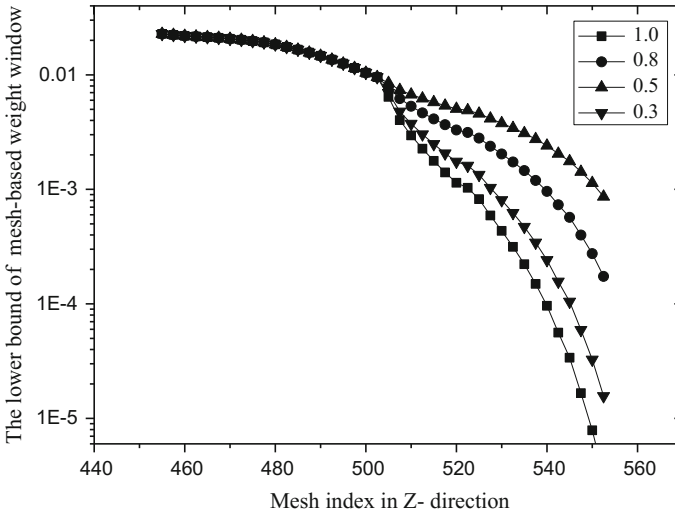
$$W'_L = C \left( \frac{W_L}{W_{L,r}} \right)^\rho \quad (2.33)$$

where  $W_L$  is the lower bound of the weight window in the CADIS method,  $W_{L,r}$  is the lower bound of the weight window at the reference point, and  $C$  is a normalization constant. The lower bound of the weight window calculated in the CADIS method will be decreased significantly in the region where the material shielding performance is strong, as shown in Fig. 2.7. By setting a proper value of  $\rho$ , the tendency of the weights to change rapidly can be avoided, the excessive splitting of the particles through these regions reduced, and the computational efficiency increased while maintaining the computational accuracy.

The GWWG method, which utilizes the particle density non-uniformity and historical mesh importance, is a physical feature prediction-based efficient particle transport simulation method. This method constructs the particle density uniformity function to represent the mesh importance and accumulates historical contributions iteratively to optimize the importance distribution. The mesh importance is defined as

$$I_i = \frac{C_i}{W_i} \quad (2.34)$$

where  $I_i$  is the importance of mesh  $P_i$ ,  $C_i$  represents the contributions of particles entering mesh  $P_i$  to the particle density uniformity, and  $W_i$  is the total weight of the particles entering mesh  $P_i$ . The mesh importance is related to the contributions to the uniformity of the global particle density distribution. With the generated weight



**Fig. 2.7** Lower bound smoothing of the mesh weight window in the z-direction

window, the flattening of the density distribution of particles can be effectively realized, and globally uniform statistical error can be obtained. To enhance the efficiency of iterative weight window optimization, the contribution to the weight window from the current iteration step and the total particle weight values in previous steps are multiplied by discount factor  $q$  and then delivered to the next iteration step. After simulating step  $k$ , the importance of mesh  $P_i$  is

$$I_{i,k} = \frac{\sum_{j=1}^k q^{k-j} c_j}{\sum_{j=1}^k q^{k-j} w_j} \quad (2.35)$$

where  $c_j$  is the cumulated contribution in mesh  $P_i$  in step  $j$  of the simulation, and  $w_j$  is the total weight of the particles entering mesh  $P_i$  in step  $j$  of the simulation. After obtaining the importance of all meshes, the lower bound of the weight window of the whole space can be calculated. Then, the convergent neutron flux of the whole space can be obtained by neutron transport calculation based on the lower bound of the weight window.

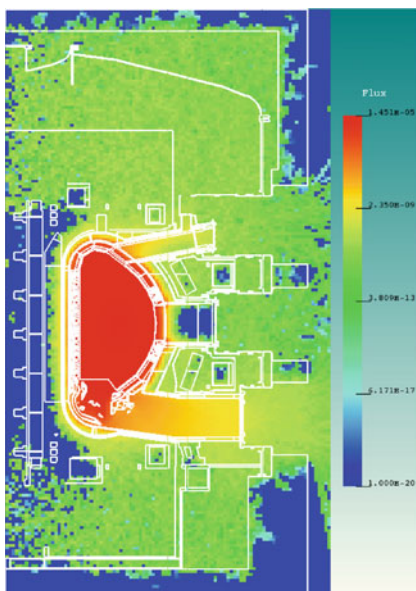
The GWWG method was benchmarked with the International Thermonuclear Experimental Reactor (ITER) Clite model, which is one of the series of basic models released by ITER. This model has a large spatial scale. Thus, the significant deep-penetration problem results in that the flux distribution of the whole space cannot be effectively calculated with the traditional Monte Carlo method. Based on the GWWG method, the average standard error of the neutron flux in the whole space of the Clite model has been reduced from 67.9% to 3.13%, indicating improvement in the Figure of Merit (FOM, also computational efficiency) a factor of 634.8. Figure 2.8 shows the results of the Clite model.

## 2. Source convergence acceleration method

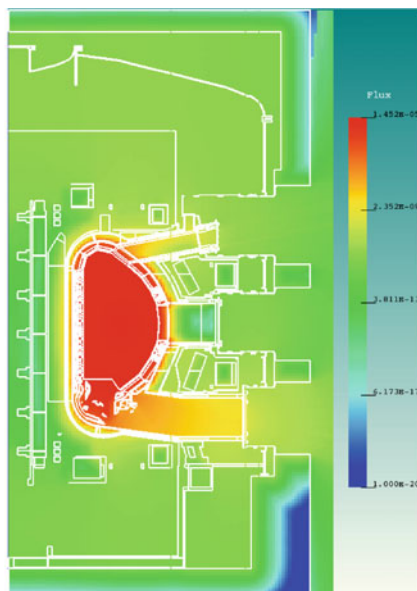
In a fusion-fission hybrid system, the eigenvalues of the breeding and transmutation blanket must be calculated. In Monte Carlo simulations, the fission source distribution of the system must converge within a prescribed tolerance to achieve reliable estimates for  $k_{\text{eff}}$  and the power. To ensure the convergence of the fission source, a large number of non-active generations must be simulated, and the Shannon entropy is employed to evaluate the convergence of the fission source distribution. Simulating non-active generation consumes a large amount of unnecessary time, highlighting the need for fission source convergence acceleration methods.

In Monte Carlo eigenvalue calculations, many methods can be used to accelerate the convergence of the fission source, including the Wieland method, finite difference method, fission matrix method, and zero-variance method. Among them, the Wieland method is most mature and has good acceleration effect. The basic idea underlying the Wieland method is introduced below.

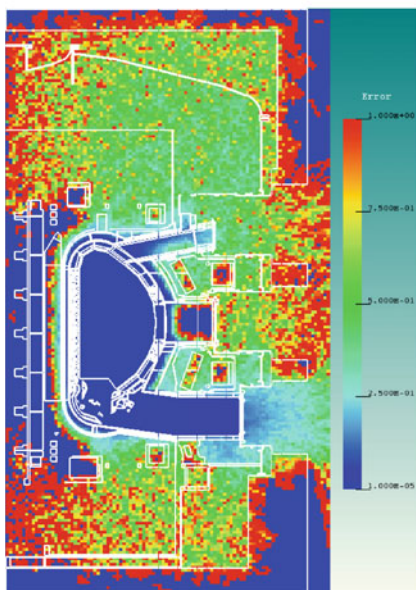
In the Wieland method, the dominance ratio of the system is reduced by subtracting a fixed fission source from both sides of the transport equation (2.20), and



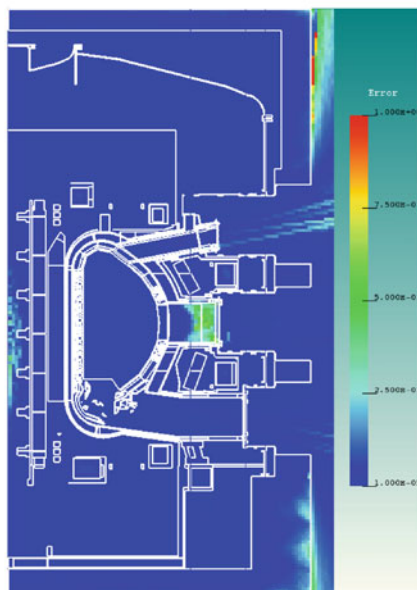
(a) Neutron flux distribution without GWG



(b) Neutron flux distribution with GWG



(c) Standard error distribution without GWG



(d) Standard error distribution with GWG

**Fig. 2.8** Comparison of the calculation results of the ITER Clite model

fast convergence of the fission source distribution is thus achieved. The transformation of the transport equation in the Wieland method is

$$\left(\mathbf{M} - \frac{1}{k_e}\mathbf{F}\right)\phi = \left(\frac{1}{k_{\text{eff}}} - \frac{1}{k_e}\right)\mathbf{F}\phi, \quad k_e > k_{\text{eff}} \quad (2.36)$$

where  $\mathbf{F}\phi/k_e$  is the fission source of the current generation. By transformation, the following equation can be obtained:

$$\phi^{(n+1)} = \frac{1}{k^{(n)}}\mathbf{F}_1\phi^{(n)} \quad (2.37)$$

where  $k^{(n)} = (1/k_{\text{eff}} - 1/k_e)^{-1}$ , and  $\mathbf{F}_1 = \mathbf{F}(\mathbf{M} - \mathbf{F}/k_e)^{-1}$ . Given  $k^{(0)}$  and  $\phi^{(0)}$ , the final  $k_{\text{eff}}$  and  $\phi$  can be obtained by iteration.

According to formula (2.37), using the Wieland method, the dominance of the system is

$$\rho^W = \frac{k_e - k_0}{k_e - k_1}\rho \quad (2.38)$$

However, when using the standard exponential iterative method, the dominant ratio of the system is  $\rho = k_1/k_0$ .  $\rho^W < \rho$  because  $k_e > k_1 > k_0$ . Thus, the eigenfunction of Eq. (2.36) converges faster, indicating that the Wieland method can accelerate the convergence of the fission source.

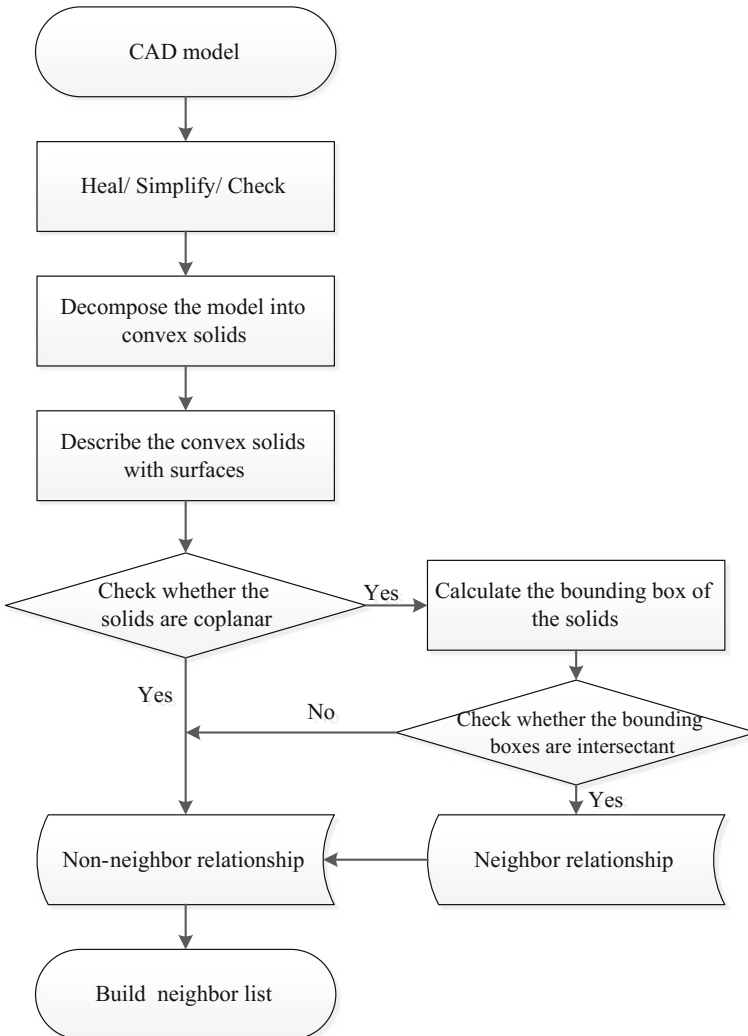
### 3. Efficient geometry processing methods

In Monte Carlo particle transport simulations, significant computation time is usually spent on geometry navigation. Therefore, optimizing the geometry navigation is an important research aim to improve the efficiency of particle transport simulations. In particle transport simulations, both the particle location and step length need to be calculated for every step of the random walk, resulting in low-efficiency calculations. In the traditional method, all geometry objects should be traversed to identify the geometry object in which the particle is located, and then, the shortest distance to the geometry boundary along the particle movement direction is determined. The time complexity of this process grows linearly as the number of geometry objects increases. To accelerate geometry navigation and improve the efficiency of particle transport calculations, efficient probabilistic transport computational methods based on the prediction of position characteristics have been developed. These methods include the neighbor list method based on CAD preprocessing, geometry selection method based bounding boxes, and optimized grid segmentation method based on cost estimation function [11].

#### (1) Neighbor list method based on CAD preprocessing

A neighbor list is constructed for each solid geometry in a complex model by analyzing the topological relationships between the geometry and its neighbor in

the CAD model. The neighbor list includes the geometry that the particle may enter after it leaves the current geometry. When the geometry is located based on the neighbor list, only the neighbor geometry, and not the whole model, is traversed, which reduces the average time needed for the geometry search and improves the efficiency of transport calculations. Figure 2.9 shows the process of constructing a neighbor list based on CAD preprocessing.



**Fig. 2.9** The process of constructing a neighbor list based on CAD preprocessing



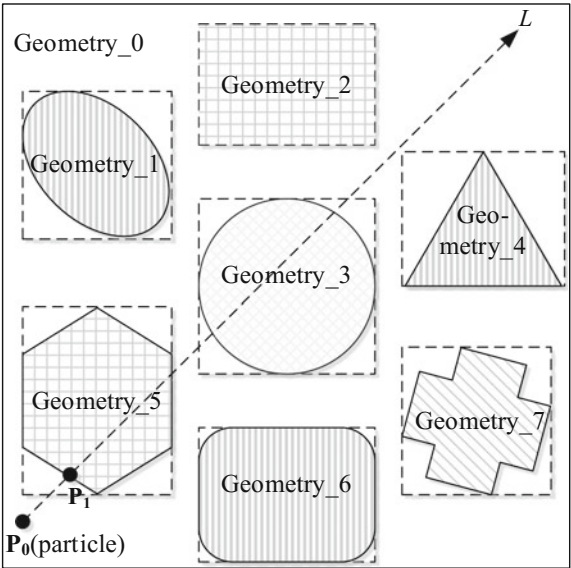
(2) Geometry selection method based on bounding boxes

The bounding box method is widely used to accelerate ray tracing and collision detection in computer graphics. Based on the CAD model, the bounding boxes of all geometries are constructed. When the geometry steps are calculated, the intersection of the ray along the particle motion and the bounding boxes is used to quickly determine whether the particle track intersects the geometry in the bounding box. This process can improve the efficiency of geometry processing, as shown in Fig. 2.10.

(3) Optimal spatial subdivision method based on cost estimation function

In computer graphics, spatial grid subdivision method is widely used for collision detection in virtual reality simulations. The basic idea is to divide the geometry model into a series of virtual space grids according to certain rules. In collision detection, only objects in the certain spatial grid need to be detected; consequently, the number of objects to be detected is reduced, and the efficiency is improved. These methods can be divided into two types: the uniform spatial grid subdivision method and non-uniform spatial grid subdivision method. The advantage of the uniform spatial grid subdivision method is that the subdivision algorithm is simple, and all the grids are exactly the same; thus, the grid location with time complexity  $O(1)$  can be realized according to the particle position. The biggest drawback of this method is that it does not consider the specific distribution of geometry objects in the model, resulting in large differences in geometry object numbers in the model. The advantage of the non-uniform spatial grid subdivision method is that it can consider the spatial distribution of geometry objects in the model, and all the

**Fig. 2.10** Geometry selection method based on bounding boxes



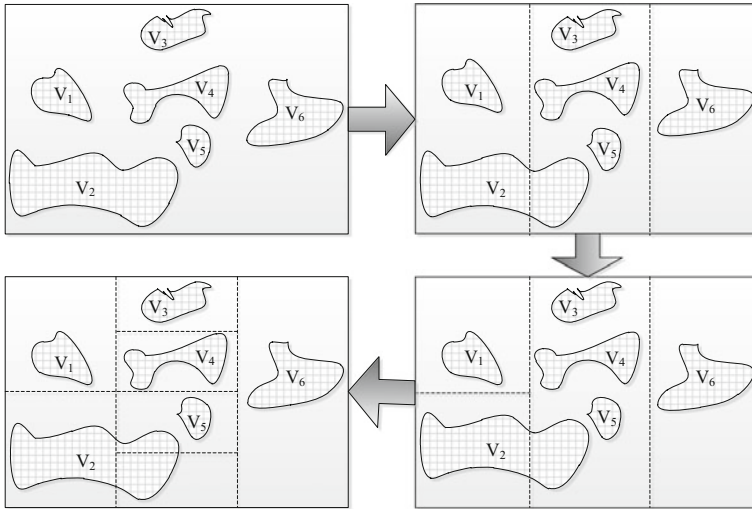
geometry objects are divided into different grids. The disadvantage is that the subdivision algorithm for spatial grid subdivision is relatively complex, and traversing the tree data structure of the model is time consuming. Additionally, this method cannot locate the grid with time complexity  $O(1)$ .

The optimal spatial subdivision method based on cost estimation function can overcome the shortcomings of the traditional spatial grid subdivision method. The cost estimation function is

$$F(C) = \frac{\sum_{J=1 \sim N_C} M_{J,C}}{\sum_{J=1 \sim N_C} N_{J,C}} \quad (2.39)$$

where  $C$  represents the current subdivision axis,  $J(1 \sim N_C)$  is the mesh index,  $N_C$  stands for the total number of grids subdivided along axis  $C$ , and  $M_{J,C}$  is the number of geometry objects in the  $J_{th}$  grid ( $N_{J,C} = 1$  when  $M_{J,C} > 0$ ;  $N_{J,C} = 0$  when  $M_{J,C} = 0$ ).  $F(C)$  is the cost estimation function of the spatial subdivision along coordinate axis  $C$ .  $F(C)$  is used to express the average number of geometry objects in each non-empty grid. As  $F(C)$  decreases, geometry navigation cost decreases, and the efficiency increases.

When the grid is divided, the minimum bounding box size along each axis is selected as the grid width. Then, the cost estimation function value for division along each axis is calculated, and the grid corresponding to the smallest cost estimation function value is selected as the optimal subdivision scheme. The above operations are repeated for each of the divided grids until all obtained grid cost estimation function values are less than the prescribed optimal values. Figure 2.11 shows the basic process of optimal spatial subdivision.



**Fig. 2.11** Optimal spatial subdivision based on cost estimation function

Geometric localization based on the optimal spatial subdivision model is briefly described below. First, according to the position of the particle and the size of the grid, the grid containing this particle can be quickly found. Then, the geometry objects contained in the grid are traversed to determine which geometry object the particle is in. Thus, the particle is in a geometry object or stays in the parent geometry object. In this method, it is necessary only to obtain the geometry information contained in the grid intersecting the particle trajectory. If a grid does not intersect the particle trajectory, there is no need to calculate the distance between the geometry objects in this grid and the particle trajectory; thus, the computational time can be reduced.

The above-mentioned acceleration algorithms have been verified using the ITER benchmark model. The neutron flux of the divertor component was calculated by setting the number of simulated particles to 1,000,000. The above three methods were compared, and Table 2.1 presents the comparison results. The neutron flux calculation results of the divertor are shown in Fig. 2.12. The correctness and efficiency of these methods are demonstrated.

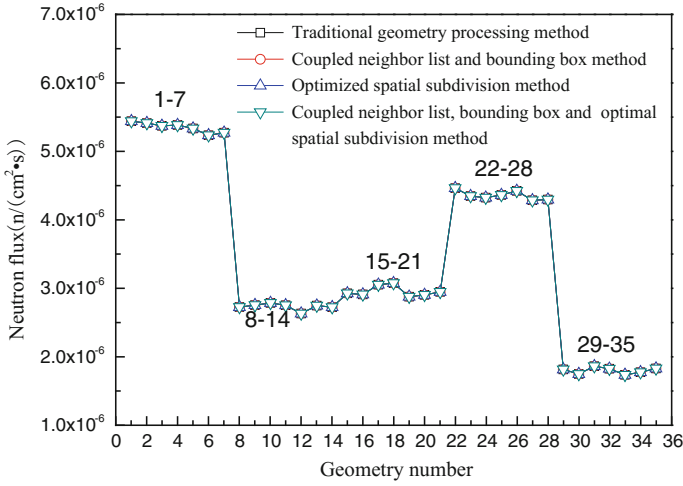
4. Parallel Computing Method

In Monte Carlo particle transport simulations with high fidelity, the computational speed and memory often represent great challenges. The parallel computing method is an effective way to overcome these challenges. Because it performs independent simulations of each particle, the Monte Carlo method has natural parallel characteristics. Three types of typical parallel computing methods exist: parallel on particles, region decomposition and data decomposition.

Parallel on particles is the basic parallel method for Monte Carlo particle transport simulations and can be divided into two varieties: parallel computing for fixed-source problems and parallel computing for eigenvalue problems. The main idea underlying parallel computing for fixed-source problems is to equally distribute particles to each process for independent simulation. After each process is complete, the simulation results are merged into the main process to obtain the final results. In parallel computing for eigenvalue problems, the fission source particles are sampled and stored in each process during the simulation of a certain generation of particles. After this generation is simulated, the fission source particles are redistributed between the processes based on the neighbor list algorithm or bidirectional traversal algorithm for next-generation calculation.

**Table 2.1** Comparison of the calculation times with the ITER benchmark model

Test cases	Calculation time/min	Acceleration rate
Traditional geometry processing method	130.60	—
Coupled neighbor list and bounding box method	62.91	2.08
Optimal spatial subdivision method	14.10	9.26
Coupled neighbor list, bounding box and optimal spatial subdivision methods	13.05	10.0



**Fig. 2.12** Simulation results of ITER benchmark model

When using the Monte Carlo method for high-fidelity calculations, especially in burnup calculations, the memory required may exceed the physical memory of a single computer node. To solve this problem, region decomposition and data decomposition algorithms can store memory data in the memory of different nodes. The main idea of a region decomposition algorithm is that the entire model is divided into several regions, which are assigned to each process. Each process stores only the geometry, material, and tally data for that region. When the particles travel across boundaries from one region to another, the particles will be passed to the new region by data communication between processes and then simulated by the corresponding process. Region decomposition algorithms can effectively solve the problem of memory, but other problems, such as load unbalance, still need to be overcome. The main idea underlying data decomposition algorithms is that the tally cells are uniformly distributed to each process and then numbered. During the simulation, the particles that move through the tally cell generate tally information, which will be sent to the process for storage according to the process number corresponding to the tally mesh. Compared with the region decomposition algorithm, the data decomposition algorithm has higher parallel efficiency. However, currently, the data decomposition algorithm cannot decompose geometry and material data.

### 2.3.2 Deterministic Methods

In deterministic methods, the continuous variables in transport equations are replaced by a set of discrete values. Thus, a matrix equation can be obtained, which

can be solved using the matrix-calculation method. The energy is usually discrete with the multi-energy-group approximation. Angle discretization can be performed by direct discretization or function expansion. Space discretization is achieved based on the MOC and the finite difference, finite element, nodal and other methods. For time-dependent problems, the direct discrete method is also used for time variables. In deterministic transport methods, several strategies for the discretization of the angle variable, such as the discrete ordinates method, the spherical harmonics method, the collision probability method, and the transmission probability method, can be used. Regarding the discretization of spatial variables, the neutron transport equation can be solved using the MOC, finite difference method, finite element method, nodal method and other methods.

Compared with the Monte Carlo method, the numerical calculation process of the deterministic method is simpler and has a faster convergence rate. However, the deterministic method has poor adaptability to complex geometries, and its computational time increases significantly with the dimension of the problem.

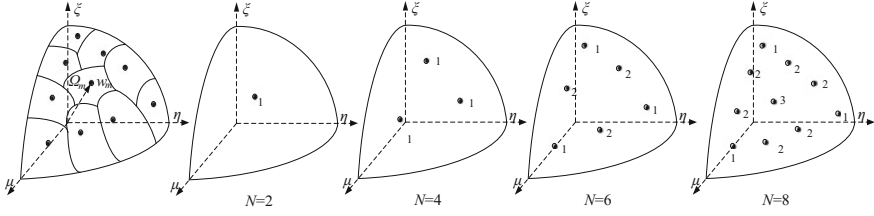
In this section, some deterministic methods, including the discrete ordinates method, the spherical harmonics method, the MOC, and some typical deterministic acceleration methods, are introduced.

### 2.3.2.1 The Discrete Ordinates Method

The discrete ordinates method (also known as the  $S_N$  method) was first implemented by B. Carlson to solve the neutron transport equation and has become one of the most effective methods for studying neutron and photon transport problems in recent years [2].

The basic idea underlying the discrete ordinates method is that the angular distribution of the neutron flux is evaluated in a number of discrete directions instead of using spherical harmonics in phase space ( $\mathbf{r} \times E \times \boldsymbol{\Omega}$ ). Specifically, for the angular neutron flux  $\phi(\mathbf{r}, E, \boldsymbol{\Omega})$ , the direction  $\boldsymbol{\Omega}$  is discretized into a sequence of discrete points ( $\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \dots, \boldsymbol{\Omega}_N$ ) or subdomains ( $\Delta\boldsymbol{\Omega}_i, i = 1, 2, \dots, N$ ). Then, after obtaining  $\phi(\mathbf{r}, E, \boldsymbol{\Omega}_i)$  values for these discrete points or subdomains, the flux  $\phi(\mathbf{r}, E, \boldsymbol{\Omega})$  is determined by approximately weighting these values. The solution of the transport equation using the discrete ordinates method can be divided into three steps [4]:

The first step is to choose a reasonable and accurate quadrature set  $\{w_m, \boldsymbol{\Omega}_m\}$ , in which  $w_m$  is the quadrature weight coefficient in direction  $\boldsymbol{\Omega}_m$ . In a one-dimensional (1D) problem, only one variable is needed to represent direction  $\boldsymbol{\Omega}$ , whereas in two-dimensional (2D) or 3D problems, two variables are required. The choice of the quadrature sets influences the computational accuracy and efficiency of the discrete ordinates method. No “best” quadrature set suitable for all problems is available. The reader can refer to the related literature [2] to understand the principle of selecting quadrature sets in detail. Taking the fully symmetric quadrature



**Fig. 2.13** Quadrature points on a 1/8 spherical surface when  $N = 2, 4, 6,$  and  $8$

sets as an example, Fig. 2.13 depicts the distribution of quadrature points on a 1/8 spherical surface when  $N$  equals 2, 4, 6, and 8.  $N$  represents the discrete number of angular directions in one ordinate. Different numbers on the spherical surface represent different types of quadrature points, and the maximum value is equal to the number of quadrature weight coefficients  $w_m$  to be determined.

The second step is to establish the difference equation. After determining the quadrature sets, the direction variables are then discretized. Next, multi-group transport equation (2.40) is integrated over  $\Delta\Omega_m$ , which is near each selected discrete direction  $\Omega_m(\mu_m, \eta_m, \zeta_m)$ . Ultimately, the discrete ordinates equation (2.44), in which the energy group symbol  $g$  is omitted for brevity, can be obtained using Eqs. (2.41)–(2.43).

$$\Omega \cdot \nabla \phi_g(\mathbf{r}, \Omega) + \Sigma_{t,g} \phi_g(\mathbf{r}) = Q_g(\mathbf{r}, \Omega) \quad (2.40)$$

$$\int_{\Delta\Omega_m} \Omega \cdot \nabla \phi(\mathbf{r}, \Omega) d\Omega = w_m [\Omega \cdot \nabla \phi(\mathbf{r}, \Omega)]_m \quad (2.41)$$

$$\int_{\Delta\Omega_m} \phi(\mathbf{r}, \Omega) d\Omega = w_m \phi_m(\mathbf{r}) \quad (2.42)$$

$$\int_{\Delta\Omega_m} Q(\mathbf{r}, \Omega) d\Omega = w_m Q_m(\mathbf{r}) \quad (2.43)$$

$$[\Omega \cdot \nabla \phi(\mathbf{r}, \Omega)]_m + \Sigma_t \phi_m(\mathbf{r}) = Q_m(\mathbf{r}) \quad (2.44)$$

Then, the discretization of the spatial variables is performed (the finite difference method is generally used). In a 3D spatial geometry, the difference equation for a discrete space mesh  $(i, j, k)$  is

$$\begin{aligned}
& w_m \mu_m \left( A_{i+1/2} \phi_{i+1/2,j,k,m} - A_{i-1/2} \phi_{i-1/2,j,k,m} \right) + w_m \eta_m B_{i,k} \left( \phi_{i,j+1/2,k,m} - \phi_{i,j-1/2,k,m} \right) \\
& + w_m \xi_m C_{i,j} \left( \phi_{i,j,k+1/2,m} - \phi_{i,j,k-1/2,m} \right) + (A_{i+1/2} - A_{i-1/2}) \left( a_{m+1/2} \phi_{i,j,k,m+1/2} \right. \\
& \left. - a_{m-1/2} \phi_{i,j,k,m-1/2} \right) + w_m \Sigma_{t,i,j,k} \phi_{i,j,k,m} V_{i,j,k} = w_m Q_{i,j,k,m} V_{i,j,k}
\end{aligned} \tag{2.45}$$

where  $Q$  is the neutron source term;  $\mu_m$ ,  $\eta_m$ ,  $\xi_m$  constitute the direction cosine of the discrete direction  $\Omega_m$ ; and  $a_{m\pm 1/2}$  is the coefficient to be determined, which satisfies the condition  $a_{m+1/2} - a_{m-1/2} = -w_m \mu_m$ . Coefficients  $A$ ,  $B$ ,  $C$  and  $V$  have different expressions in different coordinate systems, as shown in Table 2.2.

The third step is to solve the difference equation (2.45) using the inner and outer iteration method.

The advantage of the discrete ordinates method is that the numerical process is relatively simple. When the iterative method is used to solve the equation, the neutron source term is already known, and the equations in each discrete direction are independent of each other. The numerical processes for solving the equations in each direction are similar; thus, they are easy to code. The disadvantages of the method are that it requires large storage and long computational time when applied to complex 3D problems.

The “ray effects” problem exists in the discrete ordinates method, especially for media with relatively small values of the scattering ratio,  $\Sigma_s/\Sigma_t$ . Unrelated to the error of the numerical process, this problem is caused by approximating the original continuous direction of the transport equation with a finite number of discrete directions. The most direct method to eliminate the ray effect is to increase the number of discrete directions. However, increasing the number of discrete directions increases the computational time, which is considered not effective. Another more effective method is to convert the discrete ordinate equation into an equivalent form of the spherical harmonics approximation equation.

In general, the structured mesh is used in the discrete ordinates method to discretize the spatial variable  $\mathbf{r}$ . However, for irregular cells in complex models, using regular structured mesh for space discretization may result in large errors. Additionally, for a fusion system with complicated geometry, solving the problem using a structured mesh  $S_N$  is difficult. In recent years, the unstructured mesh  $S_N$  method has been developed to discretize irregular geometries using tetrahedral

**Table 2.2** Area and volume elements in commonly used coordinate systems

Ordinates		$A_{i\pm 1/2}$	$B_{i,k}$	$C_{i,j}$	$V_{i,j,k}$
Rectangular	$x, y, z$	$\Delta y_j \Delta z_k$	$\Delta x_i \Delta z_k$	$\Delta x_i \Delta y_j$	$\Delta x_i \Delta y_j \Delta z_k$
Cylindrical	$r, \varphi, z$	$2\pi r_{i\pm 1/2} \Delta \varphi_j \Delta z_k$	$\Delta r_i \Delta z_k$	$R_i \Delta \varphi_j$	$R_i \Delta \varphi_j \Delta z_k$

Notes (1)  $R_i = r_{i+1/2}^2 - r_{i-1/2}^2$ ,  $\Delta \varphi$  is normalized by  $2\pi$

meshes. This method has a stronger capability to deal with complex geometry models, especially in fusion reactors, but long computational time and large computational resources are required.

### 2.3.2.2 The Method of Characteristics

The MOC, which combines the advantages of both the discrete ordinates method and the collision probability method, was proposed by Askew [12]. In this method, the differential neutron transport equation is used. This differential equation is then solved by integrating along the characteristics of the differential operator that correspond to the tracking lines.

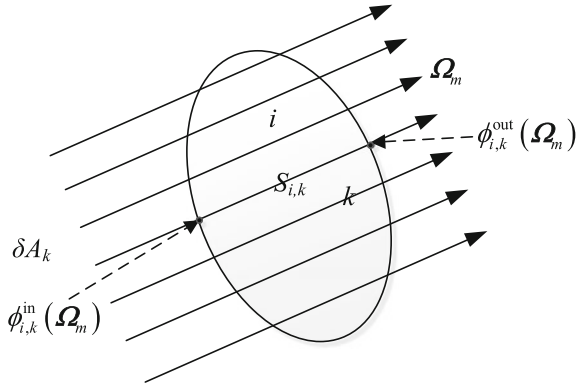
The MOC can be divided into two types: the MOC with the flat source approximation and the MOC with the linear source approximation. Furthermore, several types of sub-methods have been developed. The step MOC developed from the flat source approximation MOC is the most intuitive, stable and widely used. This MOC requires the system to be divided into several sub-regions, and the total cross-section is assumed to be constant. The neutron source is also assumed to be constant over a certain characteristic line segment  $k$ .

Similar to the discrete ordinates method, the first step in the MOC is to choose a reasonable, accurate quadrature group  $\{w_m, \Omega_m\}$  for discretizing the direction variable. Then, in each discrete direction  $\Omega_m$ , a large number of parallel rays pass through all regions, as shown in Fig. 2.14. The neutron flux is finally calculated by integrating the differential neutron transport equations along these rays. In Fig. 2.14,  $S_{i,k}$  is the  $k$ th characteristic line in region  $i$ .  $\delta A_k$  is the projection area of region  $i$  along the direction of neutron motion.  $\phi_{i,k}^{\text{in}}(\Omega_m)$  is the incoming neutron flux, and  $\phi_{i,k}^{\text{out}}(\Omega_m)$  is the outgoing neutron flux.

The multi-group differential neutron transport equation in the steady state is

$$\Omega \cdot \nabla \phi_g(\mathbf{r}, \Omega) + \Sigma_{t,g} \phi_g(\mathbf{r}, \Omega) = Q_g(\mathbf{r}, \Omega) \quad (2.46)$$

Fig. 2.14 Characteristic line





According to the assumptions of the MOC, in region  $i$ , both the total cross-section  $\Sigma_{t,i}$  and neutron source term  $Q_i(\Omega)$  are constant.  $\phi_{i,k}(s, \Omega_m)$  is the neutron flux along characteristic line  $k$  in region  $i$ . Thus, the equation in direction  $\Omega_m$  and energy group  $g$  is

$$\frac{d\phi_{i,k}(s, \Omega_m)}{ds} + \Sigma_{t,i}\phi_{i,k}(s, \Omega_m) = Q_{i,k}(\Omega_m) \quad (2.47)$$

Equation (2.47) is a first-order ordinary differential equation, and the general solution is shown in Eq. (2.48), in which the energy index  $g$  is omitted for simplicity.

$$\phi_{i,k}(s, \Omega_m) = C \cdot \exp(-\Sigma_{t,i}s) + \frac{Q_{i,k}(\Omega_m)}{\Sigma_{t,i}} \quad (2.48)$$

where  $C$  is a constant. The constant  $C$ , the neutron flux along the characteristic line  $\phi_{i,k}(s, \Omega_m)$ , and the outgoing neutron flux  $\phi_{i,k}^{\text{out}}(\Omega_m)$  can be obtained as in Eqs. (2.49)–(2.51), respectively, according to the incoming neutron flux  $\phi_{i,k}^{\text{in}}(\Omega_m)$ .

$$C = \phi_{i,k}^{\text{in}}(\Omega_m) - \frac{Q_{i,k}(\Omega_m)}{\Sigma_{t,i}} \quad (2.49)$$

$$\phi_{i,k}(s, \Omega_m) = \phi_{i,k}^{\text{in}}(\Omega_m) \exp(-\Sigma_{t,i}s) + \frac{Q_{i,k}(\Omega_m)}{\Sigma_{t,i}} [1 - \exp(-\Sigma_{t,i}s)] \quad (2.50)$$

$$\phi_{i,k}^{\text{out}}(\Omega_m) = \phi_{i,k}^{\text{in}}(\Omega_m) \exp(-\Sigma_{t,i}s_{i,k}) + \frac{Q_{i,k}(\Omega_m)}{\Sigma_{t,i}} [1 - \exp(-\Sigma_{t,i}s_{i,k})] \quad (2.51)$$

The average neutron angular flux along the characteristic line can be obtained by integrating the above equation along the characteristic line.

$$\begin{aligned} \bar{\phi}_{i,k}(\Omega_m) \cdot s_{i,k} &= \int_0^{s_{i,k}} \phi_{i,k}(s, \Omega_m) ds \\ &= \frac{Q_{i,k}(\Omega_m)}{\Sigma_{t,i}} s_{i,k} + \frac{\phi_{i,k}^{\text{in}}(\Omega_m)}{\Sigma_{t,i}} - \frac{\phi_{i,k}^{\text{out}}(\Omega_m)}{\Sigma_{t,i}} \\ &= \frac{Q_{i,k}(\Omega_m)}{\Sigma_{t,i}} s_{i,k} + \frac{\Delta\phi_{i,k}(\Omega_m)}{\Sigma_{t,i}} \end{aligned} \quad (2.52)$$

The average neutron angular flux in region  $i$  is volume weighted, as shown in Eq. (2.53).

$$\bar{\phi}_i(\boldsymbol{\Omega}_m) = \frac{\sum_k \bar{\phi}_{i,k}(\boldsymbol{\Omega}_m) s_{i,k} \delta A_k}{V_i} \approx \frac{\sum_k \bar{\phi}_{i,k}(\boldsymbol{\Omega}_m) s_{i,k} \delta A_k}{\sum_k s_{i,k} \delta A_k} \quad (2.53)$$

After calculating the neutron angular flux in a certain direction  $\boldsymbol{\Omega}_m$ , energy group  $g$  and region  $i$ , the neutron flux is obtained by angular integration of the neutron angular flux over all directions using the following numerical integral formula, where  $M$  is the number of discrete directions:

$$\bar{\phi}_i = \int_{4\pi} \bar{\phi}_i(\boldsymbol{\Omega}) d\boldsymbol{\Omega} \approx \sum_{m=1}^M \omega_m \bar{\phi}_i(\boldsymbol{\Omega}_m) \quad (2.54)$$

The main advantage of the MOC compared to the many other deterministic transport methods is that its geometric description is rather flexible. Therefore, more attention has been paid to the MOC in recent years. In the MOC, to achieve high computational accuracy, a large number of parallel lines must pass through each region, resulting in long computational time and very large memory requirements. Thus, the 3D MOC transport calculations consume substantial amounts of computer resources.

The 3D MOC is very costly for fusion systems because of their complex geometry. Researchers have attempted to improve the performance of the MOC in recent years. For example, the neutronics analysis of ITER test blanket module was performed with the 3D MOC and the spherical harmonics coupling method, in which the MOC was used only for local 3D mesh computations. An efficient 3D MOC for whole systems needs to be further developed.

### 2.3.2.3 The Spherical Harmonics Method

The spherical harmonics method (also known as the  $P_N$  method) was the first method developed for neutron transport calculations. Some functions containing the direction variable  $\boldsymbol{\Omega}$ , such as the neutron angular flux  $\phi(\mathbf{r}, E, \boldsymbol{\Omega})$  in the original transport equation, are expanded by the spherical harmonic function  $Y_n^m(\mathbf{r}, E, \boldsymbol{\Omega})$  [13]. Then, the transport equation can be transformed into a group of differential equations. Each coefficient of expansion series is determined by solving these differential equations. The neutron angular flux is expanded in terms of spherical harmonics functions as follows:

$$\phi(\mathbf{r}, E, \boldsymbol{\Omega}) = \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} \sum_{m=-n}^n a_{n,m} \phi_{n,m}(\mathbf{r}, E) Y_n^m(\theta, \varphi) \quad (2.55)$$

where

$$a_{n,m} = \frac{(n - |m|)!}{(n + |m|)!} \frac{2}{(1 + \delta_{0m})} \quad (2.56)$$

$$Y_n^m(\theta, \varphi) = \begin{cases} P_n^{|m|}(\cos \theta) \sin^{|m|} \varphi, & m = -1, \dots, -n \\ P_n^m(\cos \theta) \cos m\varphi, & m = 0, 1, 2, \dots, n \end{cases} \quad (2.57)$$

In Eq. (2.57),  $\delta_{0m}$  is the Kronecher symbol.  $\delta_{0m} = 1$  when  $m = 0$ , and  $\delta_{0m} = 0$  when  $m \neq 0$ .  $\phi_{n,m}(\mathbf{r}, E)$  is a coefficient to be determined and can be expressed as

$$\phi_{n,m}(\mathbf{r}, E) = \int_{\Omega} \phi(\mathbf{r}, E, \boldsymbol{\Omega}) Y_n^m(\theta, \varphi) d\boldsymbol{\Omega} \quad (2.58)$$

During calculations, we usually take the expansion of the former  $N + 1$  item; that is,  $n$  ranges from 0 to the order  $N$ , where  $N$  is the expansion order. The neutron angular flux is

$$\phi(\mathbf{r}, E, \boldsymbol{\Omega}) = \sum_{n=0}^N \frac{2n+1}{4\pi} \sum_{m=-n}^n a_{n,m} \phi_{n,m}(\mathbf{r}, E) Y_n^m(\theta, \varphi) \quad (2.59)$$

Similarly, the scattering cross-section is expanded using Legendre polynomials.

$$\Sigma_s(\mathbf{r}; E' \rightarrow E, \mu_0) = \sum_{n=0}^N \frac{2n+1}{2} \Sigma_{sn}(\mathbf{r}; E' \rightarrow E) P_n(\mu_0) \quad (2.60)$$

where  $\mu_0 = \boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}$ .  $\Sigma_{sn}(\mathbf{r}; E' \rightarrow E)$  is the Legendre expansion coefficient, which is expressed by

$$\Sigma_{sn}(\mathbf{r}; E' \rightarrow E) = \int_{-1}^{+1} \Sigma_s(\mathbf{r}; E' \rightarrow E, \mu_0) P_n(\mu_0) d\mu_0 \quad (2.61)$$

The scattering source is

$$Q_s(\mathbf{r}, E, \boldsymbol{\Omega}) = \int_0^\infty dE' \int_{\Omega'} \Sigma_s(\mathbf{r}; E' \rightarrow E, \mu_0) \phi(\mathbf{r}, E', \boldsymbol{\Omega}') d\boldsymbol{\Omega}' \quad (2.62)$$

By substituting Eqs. (2.59) and (2.60) in Eq. (2.62) and using the additive formula and orthogonal relation of Legendre polynomials, the scattering source can be derived as follows:

$$Q_s(\mathbf{r}, E, \boldsymbol{\Omega}) = \int_0^\infty dE' \sum_{n=0}^N \Sigma_{sn}(\mathbf{r}; E' \rightarrow E) \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} \phi_{n,m}(\mathbf{r}, E') Y_n^m(\theta, \varphi) \quad (2.63)$$

Substituting Eqs. (2.59) and (2.63) in the neutron transport equation and using the additive formula and orthogonal relation of spherical harmonic functions, a set of closely coupled equations can be obtained. Taking a 1D Cartesian coordinate system and the single energy group as an example,  $N + 1$  equations are written, as follows:

$$\frac{n+1}{2n+1} \frac{d\phi_{n+1}(z)}{dz} + \frac{n}{2n+1} \frac{d\phi_{n-1}(z)}{dz} + \Sigma_n \phi_n(z) = S(z) \delta_{0n}, \quad n = 0, 1, 2, \dots, N \quad (2.64)$$

where  $\Sigma_n = \Sigma_t - \Sigma_{sn}$ ,  $S(z)$  is the neutron source term, and the undetermined coefficients of the neutron flux can be found after applying the appropriate boundary conditions.

For the angular flux expanded with spherical harmonics, the direction variable is continuous. This expansion has characteristics of rotational invariance at any angle. Thus, the spherical harmonics method does not suffer from ray effects. Moreover, the variables can be treated with the finite element method, and the spherical harmonics method is very effective for unstructured meshes of irregular geometry. However, for complex multidimensional geometry, it is far more complicated to obtain the solution using the spherical harmonics method. Both the equations and the processing of the numerical solution are different for different expansion orders. Therefore, writing general codes for equations with different orders is difficult and eventually leads to substantial inconvenience under practical engineering calculations. These shortcomings hinder the application of the high-order spherical harmonics method.

In the simplified spherical harmonics method ( $SP_N$ ), by replacing the 1D spatial differential operators of spherical harmonic equations with 3D spatial differential operators, the  $SP_N$  equations in a 3D Cartesian coordinate system can be obtained. The simplified spherical harmonics approximation reduces the number of coupled differential equation without resulting in substantial loss of computational accuracy.

### 2.3.2.4 Acceleration Methods

The neutron transport equations are transformed into linear equations after numerical discretization and solved by matrix-calculation methods. Because the order of the matrix is so high, an iterative method is required. The convergence rate of the commonly used source iteration method is very slow, and sometimes, the

solution is not converged. Therefore, it is time consuming to solve the neutron transport equation for practical problems, and acceleration methods are necessary. Acceleration can be achieved in several ways, such as by adopting a new iterative strategy, speeding up the iterative convergence rate or adopting parallel computing. A variety of acceleration methods have been developed, such as the Discrete Nodal Transport Method (DNTM) [14], the coarse mesh rebalance method, the diffusion synthetic acceleration method [2], the transport synthetic acceleration method, and the Krylov subspace method [15]. This section briefly introduces several acceleration methods.

### 1. DNTM

The discrete ordinates method discretizes continuous direction variables with a finite number of discrete directions. To ensure accuracy, the spatial mesh cannot be too large; thus, the traditional fine mesh finite difference method is used. However, the computational cost of the fine mesh finite difference in the discrete ordinates method is very high. Discretizing the spatial mesh with the nodal method can greatly improve the computational efficiency. DNTM combines the ideas of the nodal Green's function method and the traditional discrete ordinates method; thus, the neutron transport equation can be solved effectively in a Cartesian coordinate system. However, in actual engineering problems, curve coordinate systems (i.e., cylindrical and spherical systems) are often used, and the neutron scattering is generally anisotropic. Problems arise for the DNTM method due to the scattering anisotropy and the angular redistribution. DNTM-related theory and solution method in the curvilinear coordinate system were developed by the FDS Team. This method can solve the above problems and perform more efficient simulation for engineering problems while consuming less resources. The following is an example of the DNTM method in a 1D curvilinear coordinate system.

In a 1D spherical coordinate system,  $\mu = \boldsymbol{\Omega} \cdot \mathbf{e}_r$ , and

$$\frac{\mu}{r^2} \frac{\partial(r^2 \phi)}{\partial r} + \frac{1}{r} \frac{\partial[(1 - \mu^2)\phi]}{\partial \mu} + \Sigma_t(r)\phi(r, \mu) = q(r, \mu) \quad (2.65)$$

In a 1D cylindrical coordinate system,  $\mu = \boldsymbol{\Omega} \cdot \mathbf{e}_r$ ,  $\eta = \boldsymbol{\Omega} \cdot \mathbf{e}_\theta$ , and  $\xi = \boldsymbol{\Omega} \cdot \mathbf{e}_z$ ; then, we have

$$\frac{\mu}{r} \frac{\partial(r\phi)}{\partial r} - \frac{1}{r} \frac{\partial(\eta\phi)}{\partial \chi} + \Sigma_t(r)\phi(r, \xi, \chi) = q(r, \xi, \chi) \quad (2.66)$$

Note that the second terms in the left-hand sides of Eqs. (2.65) and (2.66) are the so-called angular redistribution terms (ARTs) and appear only in curvilinear geometries. Suppose that there are several nodes along the radial direction  $r$  and that the width of the  $i$ th node is  $\Delta r_i$ . Equations (2.65) and (2.66) are multiplied by  $r^j$  ( $j = 1$  for cylindrical geometry;  $j = 2$  for spherical geometry), and the variable transformation  $r = r_i + a_i r'$  ( $r' \in [-1, 1]$ ,  $a_i = \Delta r_i/2$ ) is performed. When Eqs. (2.65) and (2.66) are rearranged (i.e., by moving the ART from the left side to the right

side) and combined, the following equation is obtained (where  $r'$  is still represented by  $r$  for convenience).

$$\mu \frac{\partial}{\partial r} F(r, \boldsymbol{\Omega}) + \Sigma_i F(r, \boldsymbol{\Omega}) = a_i R(r, \boldsymbol{\Omega}) \quad (2.67)$$

where the dimensionless (local) variable  $r$  is written in terms of the node half-width  $a_i$ .  $r_i$  is the radius of the  $i$ th node midpoint, and  $\Sigma_i$ ,  $F(r, \boldsymbol{\Omega})$  and  $R(r, \boldsymbol{\Omega})$  are represented, respectively by

$$\boldsymbol{\Omega} = \begin{cases} \mu, & \text{for spherical geometry} \\ (\xi, \chi), & \text{for cylindrical geometry} \end{cases} \quad (2.68)$$

$$\Sigma_i = a_i \Sigma_t \quad (2.69)$$

$$F(r, \boldsymbol{\Omega}) = (r_i + a_i r)^j \phi(r_i + a_i r, \boldsymbol{\Omega}) \quad (2.70)$$

$$R(r, \boldsymbol{\Omega}) = \begin{cases} (r_i + a_i r)^2 q(r_i + a_i r, \mu) - (r_i + a_i r) \frac{\partial}{\partial \mu} [(1 - \mu^2) \phi(r_i + a_i r, \mu)], & \text{for spherical geometry} \\ (r_i + a_i r) q(r_i + a_i r, \xi, \chi) + \frac{\partial}{\partial \chi} [\eta \phi(r_i + a_i r, \xi, \chi)], & \text{for cylindrical geometry} \end{cases} \quad (2.71)$$

After obtaining the nodal equation, the equation is discretized, and then, the Green's function method is used to find the solution. Finally, the neutron flux in node is calculated by Legendre polynomial expansion. This method can achieve very high accuracy on coarse spatial meshes and can greatly enhance the computational efficiency of neutron transport in curvilinear coordinate systems.

## 2. Coarse mesh rebalance method

The coarse mesh rebalance method is an effective and widely adopted acceleration method. The basic idea underlying this method is as follows: Based on the original refined finite difference mesh, a set of regular and non-overlapping coarse meshes containing several fine meshes is built up with a coincident boundary between the coarse mesh and the fine mesh. After iterations for the fine mesh, a "coarse mesh rebalance factor" is calculated according to the neutron balance principle. The final solution is achieved by multiplying the coarse mesh rebalance factor by the neutron flux obtained by iteration. This approach can force the solution to satisfy the neutron balance relationship in the coarse mesh and accelerate the convergence rate.

## 3. Diffusion synthetic acceleration method

The diffusion synthesis acceleration method aims to accelerate the convergence of the source iteration for the  $S_N$  method. The diffusion equation is a good

approximation of the transport equation based on transport theory. In this method, a modified diffusion equation is obtained by replacing the original leakage term with the leakage term calculated by the transport equation. Thus, the solution of the modified diffusion equation has the accuracy of the transport equation. Then, the neutron flux obtained in each iteration is rectified by the modified diffusion equation and used to obtain the next iteration source, thereby effectively accelerating the source iteration.

#### 4. Krylov subspace method

The Krylov subspace method is an accelerated method that was developed in recent years. Its core idea is to find a subspace  $k$  from  $n$ -dimensional vector space and then to find an approximate solution in subspace  $k$ . Assume that the subspace dimension  $\dim k = m$ . Then,  $m$  constraints are required to derive an approximate solution from subspace  $k$ . Usually,  $m$  orthogonal conditions are adopted to solve the linear equation sets. In the solution of the eigenvalue problem, an  $m$ -order small matrix is usually obtained with the subspace, the eigenvalues of the large matrix are approximately obtained by solving the small matrix, and  $m$  coefficients are determined from the eigenvectors of the small matrix to linearly combine the subspaces. In addition to its high computational speed, the Krylov subspace method can solve many eigenvalues and eigenvectors simultaneously and play an important role in solving the high-order harmonics of neutron transport equations.

## 2.4 Transport Simulation Codes

### 2.4.1 Monte Carlo Codes

The Monte Carlo method can handle neutron transport problems of complex geometry, complex neutron spectrum and anisotropic neutron scattering and has been widely employed in neutron transport calculations of fusion reactors. Currently, the most common codes for neutron transport in fusion systems are based on the Monte Carlo method. Some typical Monte Carlo transport simulation codes are introduced in following section.

#### 2.4.1.1 SuperMC

The Super Monte Carlo Program for Nuclear and Radiation Simulation (SuperMC), which was developed by the FDS Team, is a general, intelligent, accurate and precise software system for nuclear design and safety evaluation of nuclear systems. It can be applied in nuclear reactor physics, radiation protection and dosimetry, radiation shielding, medical physics, nuclear criticality safety, nuclear oil-well logging and other fields [4, 16]. SuperMC will be introduced in detail in Chap. 7.

SuperMC supports transport simulations of various types of particles, such as neutrons, photons, electrons, and protons. The physical reactions of neutrons include inelastic scattering, elastic scattering and absorption. The thermal scattering effect of neutrons is considered, and the epithermal energy range of neutrons is treated. The self-shielding effect in the unresolved resonance range and prompt neutrons are also considered. The reactions of photons with nuclides include Compton scattering, coherent scattering, the photoelectric effect, the electron pair effect and photonuclear reactions. Sources include general sources, critical sources and user-defined sources. Common physical quantities for nuclear design and analysis, such as the cell flux, point flux, energy deposition, and reaction rate, and special quantities, such as  $k_s$  for an external source-driven system, can be calculated in SuperMC. A criticality search aiming for a specified eigenvalue within a prescribed confidence interval is designed exclusively for determining assembly configurations, fuel loadings, and soluble boron concentrations, among others. Considering the multi-physics feedback effect, the functions of structured mesh, unstructured mesh, and continuous tallies, among others, are developed. For shielding calculations, some basic variance reduction techniques, including the weight window, forced collision, and exponential transformation, and adaptive variance reduction techniques, such as GWWG, are supported.

For fusion systems, SuperMC has already been widely applied to nuclear design and analysis, including the accurate evaluation of nuclear heat deposition in the Toroidal Field (TF) Coils of ITER, nuclear analysis and shielding optimization of observation systems in vacuum vessels, assessment of radioactive waste, and tritium breeding rate of blankets.

#### 2.4.1.2 MCNP

MCNP, a General Monte Carlo N-Particle Transport Code, was developed by Los Alamos National Laboratory (LANL) in the USA [17]. It was coded using the FORTRAN and C programming languages, and the latest version is MCNP6, which combines the features of both MCNP5 and MCNPX. MCNP6 can be used to perform transport computations of heavy charged particles and has been widely applied in many fields, such as fusion/fission reactor design, radiation protection and dosimetry, medical physics, and nuclear criticality safety. It supports transport simulations of neutrons, photons, electrons and heavy charged particles and can perform eigenvalue and fixed-source calculations. Simulations of neutrons with energies ranging from  $10^{-5}$  eV to 150 MeV and photons with energies ranging from 1 keV to 100 GeV are supported. In the latest version of MCNP6, multi-type descriptions of sources, various tallies and variance reduction techniques are included. MCNP has already been applied in the neutronics analysis of ITER, including calculations of neutron wall loading, neutron/photon fluence rate and nuclear heat deposition.



### 2.4.1.3 TRIPOLI

TRIPOLI is a 3D, general-purpose continuous-energy Monte Carlo transport code developed by the French Alternative Energies and Atomic Energy Commission (CEA) [18]. It was coded in the C/C++ language and has been applied to the radiation protection and shielding design of fission reactor core and the design of fusion reactors and other nuclear facilities. The current version TRIPOLI-4 supports neutron and photon transport simulations and has eigenvalue and fixed-source computation capabilities. The energy range of the simulated neutrons is  $10^{-5}$  eV–20 MeV, and that of the simulated photons is 1 keV–100 MeV. Physical quantities, such as the flux, reaction rate, energy deposition, perturbation calculation for the density/differential cross-section and basic variance reduction techniques, are all supported in TRIPOLI-4. This code has already been applied to neutronics analyses of fusion systems, such as the calculation of neutron flux in the ITER blanket, nuclear heat deposition and tritium breeding rate.

### 2.4.1.4 Serpent

Serpent is a continuous-energy Monte Carlo reactor physics burnup calculation code developed by the National Technical Research Center of Finland (VTT) [19]. It was programmed in the C language. The current version is Serpent 2, and it is mainly applied for physics analyses of fission reactors. Serpent 2 supports neutron and photon transport simulations within the neutron energy range of  $10^{-5}$  eV–20 MeV. The flux, reaction rate, energy deposition and other physical quantities can be calculated. For fusion systems, Serpent 2 has been applied to the design and analysis of fusion systems, such as neutron flux calculations of the ITER Clite model. Variance reduction techniques for fusion applications are under development.

### 2.4.1.5 Geant4

Geant4 is a toolkit for the simulation of the passage of particles through matter that was developed by the European Organization for Nuclear Research (CERN) [20]. It is programmed in the C++ language, and the current version (Geant4 10.3) has been widely applied in high-energy particle physics, heavy ion physics, astrophysics, space science, radiation medicine and other fields. Geant4 supports transport simulations of more than 100 types of particles, including gluons, muons, neutrons, protons and ions. It contains a variety of physical models and simple variance reduction techniques (section bias and implicit capture). Geant4 has been applied to the design and analysis of fusion systems.

#### 2.4.1.6 FLUKA

FLUKA is a fully integrated particle physics Monte Carlo simulation package developed by CERN and the Italian Institute for Nuclear Physics (INFN) [21]. It is programmed in the FORTRAN language and has mainly been applied in high-energy physics, radiation protection, radiation therapy, accelerator design and other fields. FLUKA supports transport simulations of more than 60 types of particles, including neutrons, photons, electrons, neutrinos and hadrons, and can calculate the particle flux, particle yield, energy deposition and other related physical quantities of charged particles. Basic variance reduction techniques, such as Russian roulette, particle splitting and source bias, are also supported. FLUKA has been applied to the design and analysis of fusion systems, such as the calibration of neutron dosimeters.

#### 2.4.1.7 PHITS

PHITS is a particle and heavy ion transport code developed by the Japan Atomic Energy Agency (JAEA), Research Organization for Information and Technology (RIST) and High Energy Accelerator Research Organization (KEK). It was programmed in the FORTRAN language, and the latest version is PHITS2.88. PHITS is mainly applied in accelerator design, radiation therapy, space radiation and other fields [22]. It supports transport simulations of multiple particles, such as neutrons, photons, electrons, protons and mesons, and the energy range of the simulated neutrons is  $10^{-5}$  eV–20 MeV. It can calculate the flux, energy deposition and other physical quantities and support simple variance reduction techniques, such as importance, weight window and forced collisions. It has been applied to the design and analysis of fusion systems.

### 2.4.2 *Deterministic Codes*

The discrete ordinates method is the main deterministic calculation method applied in fusion neutron transport calculations. In this section, two deterministic codes, DOORS and ATILA, based on the discrete ordinates method are introduced.

#### 2.4.2.1 DOORS

DOORS is a 1D, 2D, and 3D discrete ordinates transport code system for the deep-penetration transport of neutrons and photons developed by ORNL. The latest version is DOORS3.2a, which contains independent codes, such as ANISN, DORT and TORT. DOORS has been widely applied in nuclear reactor physics and nuclear radiation shielding [23].

In DOORS, the discrete ordinates method, the multi-group approximation, the finite difference method and the nodal method are all used to process the variables of angle, energy and space, respectively. In DOORS, ANISN can calculate the neutron/photon flux and eigenvalue in 1D coordinate system. DORT can calculate the neutron/photon flux and eigenvalue in 1D or 2D systems. TORT can be used to address the deep-penetration problem in radiation shielding and calculate neutron/photon flux and eigenvalues in 2D and 3D coordinate systems.

DOORS has been applied to the design and analysis of fusion systems, such as the 3D calculation of radiation field distribution of HT-7U Superconducting Tokamak to support design optimization of shielding systems and associated components.

### 2.4.2.2 ATTILA

ATTILA is a 3D, unstructured tetrahedral mesh discrete ordinates neutron/photon transport code [24] developed by the Transpire company. It was coded in the FORTRAN language. It can be used to address complex geometrical transport problems and has been applied to analyses of fission and fusion reactors, radiation therapy, nuclear oil-well logging and other fields.

The discrete ordinates method, the multi-group approximation and the tetrahedral mesh are used to process angle, energy and space variables, respectively. For fusion system, ATTILA has been applied to neutronics analyses, such as the nuclear design and shielding of ITER diagnostics port plugs.

## References

1. Lamarsh JR (1966) Introduction to nuclear reactor theory. Addison-Wesley Publication Company, Massachusetts
2. Lewis EE, Miller WF Jr (1993) Computational methods of neutron transport. American Nuclear Society, La Grange Park, Illinois
3. Wilson RPH, Feder R, Fischer U et al (2008) State-of-art 3-D radiation transport methods for fusion energy system. *Fusion Eng Des* 83:824–833
4. Wu YC, Song J, Zheng HQ et al (2015) CAD-Based Monte Carlo Program for integrated simulation of nuclear system SuperMC. *Ann Nucl Energy* 82:161–168
5. Song J, Sun GY, Chen ZP et al (2015) Study on Monte Carlo K-effective calculation method. *Nuclear Sci Eng* 35(2): 241–245 (in Chinese)
6. Haghighat A, Wagner JC (2003) Monte Carlo variance reduction with deterministic importance functions. *Prog Nuclear Energy* 42(1):25–53
7. Zhao JB, Li XM, Wu B et al (2016) An automatic adaptive mesh generation method for weight window in Monte Carlo particle transport. *Ann Nuclear Energy* 91:105–110
8. Wagner JC, Blakeman ED, Peplow DE (2007) Forward-weighted CADIS method for global variance reduction. *Trans Am Nuclear Soc* 97:630–633
9. Davis A, Turner A (2011) Comparison of global variance reduction techniques for Monte Carlo radiation transport simulations of ITER. *Fusion Eng Des* 86(9–11):2698–2700

10. Zhang S, Yu SP, He P (2016) Verification of SuperMC with ITER C-Lite neutronic model. *Fusion Eng Des* 113:126–130
11. Chen ZP, Song J, Wu B et al (2015) Optimal spatial subdivision method for improving geometry navigation performance in Monte Carlo particle transport simulation. *Ann Nuclear Energy* 76:479–484
12. Askew JR (1972) A characteristics formulation of the neutron transport equation in complicated geometries. Report AEEW-M 1108. United Kingdom Atomic Energy Establishment, Winfrith, England
13. Bell GI, Glasstone S (1970) Nuclear reactor theory. Van Nostrand Reinhold Company, New York
14. Wu YC, Xie ZS, Fischer U (1999) A discrete ordinates nodal method for one-dimensional neutron transport calculation in curvilinear geometries. *Nuclear Sci Eng* 133:350–357
15. Azmy Y, Sartori E (2010) Nuclear computational science: a century in review. Springer, Berlin
16. Wu YC, Team FDS (2009) CAD-based interface programs for fusion neutron transport simulation. *Fusion Eng Des* 84(7–11):1987–1992
17. X-5 Monte Carlo Team (2003) MCNP-A general Monte Carlo N-particle Transport Code, Version 5. LA-UR-03-1987, Los Alamos National Laboratory
18. Brun E, Damian F, Diop CM et al (2015) TRIPOLI-4, CEA, EDF and AREVA reference Monte Carlo code. *Ann Nucl Energy* 82:151–160
19. Leppanen J, Pusa M, Vitanen T (2015) The serpent Monte Carlo code: status, development and applications in 2013. *Ann Nucl Energy* 82:142–150
20. Tabbakh F (2016) Particles transportation and nuclear heating in a tokamak by MCNPX and GEANT4. *J Nucl Energy* 35:401–406
21. Ferrari A, Sala PR, Fasso A et al (2011) Fluka: a multi-particle transport code. CERN-2005-010, Geneva
22. Niita K, Sato T, Iwase H et al (2006) PHITS-A particle and heavy ion transport code system. *Rad Meas* 41(9–10):1080–1090
23. Oak Ridge National Laboratory (1998) DOORS3.2 one, two- and three dimensional discrete ordinates neutron/photon transport code system. CCC-650, Oak Ridge, Tennessee
24. Youssef M, Feder R, Batistoni P et al (2013) Benchmarking of the 3-D CAD-based discrete ordinates code “ATTILA” for dose rate calculations against experiment and Monte Carlo calculations. *Fusion Eng Des* 88:3033–3040

Fusion Neutronics

Wu, Y.

2017, XX, 393 p. 192 illus., 110 illus. in color.,

Hardcover

ISBN: 978-981-10-5468-6