Overview of methodology for spatial homogenization in the Serpent 2 Monte Carlo code

Leppanen, J.; Pusa, M.; Fridman, E.;

Originally published:

June 2016

Annals of Nuclear Energy 96(2016), 126-136

DOI: https://doi.org/10.1016/j.anucene.2016.06.007

Perma-Link to Publication Repository of HZDR:

https://www.hzdr.de/publications/Publ-23453

Overview of methodology for spatial homogenization in the Serpent 2 Monte Carlo code

Jaakko Leppänen[a,*], Maria Pusa[a], Emil Fridman[b]

[a] *VTT Technical Research Centre of Finland, Ltd., Kivimiehentie 3, Espoo, FI-02044 VTT, Finland*
[b] *Helmholtz-Zentrum Dresden-Rossendorf, Bautzner Landstraße 400, 01328 Dresden, Germany*

Abstract

This paper describes the methods used in the Serpent 2 Monte Carlo code for producing homogenized group constants for nodal diffusion and other deterministic reactor simulator calculations. The methodology covers few-group reaction cross sections, scattering matrices, diffusion coefficients and poison cross sections homogenized in infinite and B_1 leakage-corrected critical spectra, as well the calculation of assembly discontinuity factors, pin-power form factors, delayed neutron parameters and total and partial albedos. Also included is a description of an automated burnup sequence, which was recently implemented for the handling of restart calculations with branch variations. This capability enables covering the full range of local operating conditions required for the parameterization of group constants within a single run. The purpose of this paper is to bring the methodological description provided in earlier publications up to date, and provide insight into the developed methods and capabilities, including their limitations and known flaws.

Keywords: Serpent, Monte Carlo, spatial homogenization, group constants, automated burnup sequence

1. Introduction

The use of the continuous-energy Monte Carlo method for producing homogenized group constants for nodal diffusion and other deterministic reactor simulator calculations has gained considerable interest during the past ten years, even though the practical applications are still limited by the high computational cost of the transport simulation. Covering all assembly types and reactor operating conditions over multiple core cycles requires solving the local heterogeneous transport problem hundreds or even thousands of times, which poses a major challenge for any calculation code. The long running time, however, is thought to be outweighed by the inherent advantages of the Monte Carlo method – the capability the handle interaction physics without major approximations and three-dimensional geometries at an arbitrary level of spatial detail.

Another significant advantage is that Monte Carlo lattice physics codes not only allow performing spatial homogenization at the fuel assembly level, but also running transport simulations for the full-scale heterogeneous system. This provides ideal reference solutions for the validation of the calculation scheme, since all additional discrepancies resulting from evaluated nuclear data libraries and methodological differences can be eliminated. The differences between the homogeneous and the heterogeneous reference solution instead reflect on how well the physics of the transport process is preserved over the calculation chain, which can be extremely valuable for the development of new methods for core calculations.

The Serpent code (Leppänen et al., 2015) has been developed at VTT Technical Research Center of Finland since 2004, and the current user basis includes some 500 users in 150 universities and research organizations in 37 countries around the world. The code is used for a multitude of applications in reactor physics, but spatial homogenization based on the Monte Carlo method was, in fact, the original incentive for starting the work. The progress has been steady over the years, but so far the use of Serpent for group constant generation has mostly been limited to preliminary studies with simplified core models.[1] This is in part because of the complexity and computational cost of producing the full set of group constants for realistic fuel cycle and transient simulator calculations, but no doubt also because of the insufficient documentation of methods and procedures used in the code.

In an effort to correct this deficiency, work on a comprehensive User's Manual in the form of an on-line Wiki[2] was started in late 2015. The purpose of this paper, on the other hand, is to bring the methodological description provided in some earlier publications (Leppänen, 2007; Fridman and Leppänen, 2011, 2012; Leppänen et al., 2014b, 2015) up to date. This part is covered in Section 2, and the description corresponds to code version 2.1.26, distributed to users in March 2016. Recent development includes also an automated burnup sequence capable of performing branch variations, which considerably simplifies setting up the inputs for group constant generation and the management of output data. The procedure is described in Section 3. Since Serpent is still a developing code, there are a number of flaws and limitations in the methodology, as discussed in Section 4. Some future plans are outlined along with the conclusions in Section 5.

*Tel. +358-40-593-9076; fax +358-20-722-5000. E-mail address: Jaakko.Leppanen@vtt.fi

[1] See (Leppänen et al., 2014b) and (Leppänen et al., 2015) for a review of examples.

[2] See: `serpent.vtt.fi/mediawiki`

This paper covers the theory of homogenization and nodal diffusion methods only as far as is considered necessary for understanding the methods used in the Serpent code. The fundamentals can be found in (Koebke, 1978) and (Smith, 1980), as well as most text books on reactor theory. Practical issues, such as the input and output formats are addressed at the Serpent on-line Wiki.

2. Methods used for spatial homogenization

The purpose of spatial homogenization is to preserve the local reaction balance when group constants obtained from the solution of the local heterogeneous transport problem (assembly-level calculation) are used as the building blocks for the global homogeneous system (core-level calculation). Formally, the homogenization of reaction cross section Σ_g can be written as:

$$\Sigma_g = \frac{\int_V d^3r \int_{E_g}^{E_{g-1}} dE\ \Sigma(\boldsymbol{r}, E)\phi(\boldsymbol{r}, E)}{\int_V d^3r \int_{E_g}^{E_{g-1}} dE\ \phi(\boldsymbol{r}, E)}\,, \tag{1}$$

where ϕ is the scalar flux and the integration is carried over the volume of the homogenized region and energy group g. The spatial dependence of Σ reflects the fact that the geometry is heterogeneous, albeit typically composed of discrete uniform material zones. In other words, spatial homogenization implies averaging the physical continuous-energy cross sections over volume and flux spectrum.

Deterministic lattice transport codes, in which the heterogeneous flux solution is obtained in space- and energy-discretized form, apply the procedure as:

$$\Sigma_g = \frac{\sum_{h\in g}\sum_i V_i \Sigma_{i,h}\Phi_{i,h}}{\sum_{h\in g}\sum_i V_i \Phi_{i,h}} \tag{2}$$

where $\Phi_{i,h}$ is the scalar group flux, and i refers to the spatially discretized regions and h to the multi-group structure used in the calculation. Cross section $\Sigma_{i,h}$ is obtained from spectral calculation, taking into account spatial and resonance self-shielding effects.

One of the advantages of using Monte Carlo simulations for spatial homogenization is that stochastic estimates for the integrals in Eq. (1) can be obtained directly, using continuous-energy cross sections. This means that self-shielding effects are automatically accounted for, without relying on various approximations employed by deterministic codes. Consequently, the same methods and cross section libraries can be used for modeling any fuel or reactor type, without any application-specific limitations. Another advantage typically attributed to the method is its capability to handle complicated three-dimensional structures, in which the level of spatial detail can be arbitrarily refined.

The group constant input for deterministic nodal diffusion codes typically consists of absorption ($\Sigma_{a,g}$), fission neutron production ($\nu\Sigma_{f,g}$) and group transfer ($\Sigma_{s,gg'}$) cross sections, as well fission spectrum χ_g and diffusion coefficients (D_g). These are the constants needed for forming the group diffusion equations. Coupling between adjacent nodes is accomplished using discontinuity factors F_g, and normalization of flux with fission energy production cross sections ($\kappa\Sigma_{f,g}$). Dynamic calculations require additionally inverse neutron speeds $1/v_g$ and effective delayed neutron fractions β_{eff}, divided into a number of precursor groups. Most modern core simulators have the capability to perform pin-power reconstruction and track the concentrations of fission product poisons ^{135}Xe and ^{149}Sm separately. The group constant input then includes also pin-power form factors and production and absorption cross sections for fission product poisons and their precursors. As discussed below, the task of producing all this data becomes much more complicated than just calculating stochastic estimates for the integrals in Eq. (1) using standard reaction rate tallies.

2.1. General procedure applied in Serpent 2

For practical reasons made apparent in Sec. 2.4, Serpent does not evaluate the integrals in Eq. (1) directly. The procedure is instead handled in two parts:

i) A number of multi-group homogenized reaction cross sections are calculated using standard Monte Carlo tallies and analog estimators.
ii) The multi-group cross sections are condensed into few-group cross sections using the infinite and the B_1-leakage corrected critical spectra (see Sec. 2.4).

In this paper the intermediate energy group structure used internally in the calculation routines is referred to as the "multi-group structure", and using group index h. The final group structure to be used in the simulator calculation is correspondingly referred to as the "few-group structure", with group index g. The default multi- and few-group structures used by Serpent are the WIMS 69-group structure and the conventional two-group structure with thermal and fast group separated at 0.625 eV. Both structures can be changed by input options.

Technically this two-stage approach means that instead of Eq. (1), the final energy group condensation is written as:

$$\Sigma_g = \frac{\sum_{h\in g}\Sigma_h\Phi_h}{\sum_{h\in g}\Phi_h}\,. \tag{3}$$

The difference to Eq. (2), applied by deterministic codes, is that Φ_h is integrated and Σ_h averaged over the volume of the homogenized geometry:

$$\Phi_h = \int_V d^3r \int_{E_h}^{E_{h-1}} dE\ \phi(\boldsymbol{r}, E)\,, \tag{4}$$

$$\Sigma_h = \frac{\int_V d^3r \int_{E_h}^{E_{h-1}} dE\ \Sigma(\boldsymbol{r}, E)\phi(\boldsymbol{r}, E)}{\int_V d^3r \int_{E_h}^{E_{h-1}} dE\ \phi(\boldsymbol{r}, E)}\,. \tag{5}$$

The integrals in Eqs. (4) and (5) are obtained using standard reaction rate tallies. Since the stochastic integration is performed using continuous-energy cross sections, all self-shielding effects are automatically taken into account, so even though the calculation is divided in two parts, the most significant advantages of Monte Carlo simulation is preserved.

The calculation proceeds in batches. One batch in criticality source simulation consists of 20 source cycles (by default), during which Monte Carlo integral estimates for parameters of the form (4) and (5) are collected from the transport simulation. Once the batch is completed, the resulting multi-group cross sections are used to calculate batch-wise estimates of the few-group constants. These estimates form the statistical mean values and the associated relative statistical errors printed in the output. It is assumed that the conditions of the central limit theorem are met, i.e. that the sequence of batch-wise estimates follows the normal distribution. This assumption is supported by statistical analyses performed for the group constant data, provided that the source population is of reasonable size (Kaltiaisenaho and Leppänen, 2014).

This general approach applies to simple one-dimensional reaction cross sections, such as total, total absorption and total fission, as well as the inverse neutron speed. Fission neutron and energy production cross sections are obtained by multiplying the fission cross section with the average neutron yield $\overline{\nu}$ and deposited fission energy κ, respectively, when the tallies for the batch-wise multi-group estimates are scored. In the default optimization mode (Leppänen and Isotalo, 2012) Serpent precalculates the corresponding continuous-energy macroscopic cross sections to avoid summation over material compositions during the transport simulation. This is made possible by the use of a single unionized energy grid for all nuclides, and may result in a significant increase in computational performance, especially in calculations involving irradiated fuel composed of hundreds of nuclides (Leppänen, 2009).

2.2. Poison cross sections

Poison cross sections for fission product poisons ^{135}Xe and ^{149}Sm and their precursors ^{135}I, ^{147}Pm, ^{148}Pm, $^{148\mathrm{m}}$Pm and ^{149}Pm are calculated similar to the homogenized reaction cross section, via the intermediate multi-group structure. The production cross section of isotope x is formally defined as:

$$\Sigma_{\mathrm{p},g,x} = \frac{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE \sum_l \gamma_{l,x}(E)\Sigma_{\mathrm{f},l}(\boldsymbol{r},E)\phi(\boldsymbol{r},E)}{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE\, \phi(\boldsymbol{r},E)}, \quad (6)$$

where $\Sigma_{\mathrm{f},l}$ is the macroscopic fission cross section for actinide l and $\gamma_{l,x}$ is the corresponding fission yield of isotope x, obtained from the ENDF format fission yield data file. The yields of the immediate precursors are taken as cumulative, i.e. they include all nuclides higher in the decay chains, most significantly ^{135}Te for ^{135}I and ^{149}Nd for ^{149}Pm. The ^{135}Xe production also includes the meta-stable state $^{135\mathrm{m}}$Xe, which has no separate cross sections in standard nuclear data libraries.

The microscopic absorption cross section for isotope x is defined as:

$$\sigma_{\mathrm{a},g,x} = \frac{\dfrac{V}{V_\mathrm{f}} \displaystyle\int_{V_\mathrm{f}} d^3r \int_{E_g}^{E_{g-1}} dE\, \sigma_{\mathrm{a},x}(E)\phi(\boldsymbol{r},E)}{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE\, \phi(\boldsymbol{r},E)}, \quad (7)$$

where V_f and V refer to the fissile and the total volume of the homogenized region, respectively. In addition to microscopic absorption cross sections, also the flux-volume-averaged macroscopic absorption cross sections of ^{135}Xe and ^{149}Sm are calculated as:

$$\Sigma_{\mathrm{a},g,x} = \frac{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE\, N_x(\boldsymbol{r})\sigma_{\mathrm{a},x}(E)\phi(\boldsymbol{r},E)}{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE\, \phi(\boldsymbol{r},E)}, \quad (8)$$

where N_x is the poison concentration. These parameters reflect the contribution of fission product poisons on total absorption for the given fuel composition.

2.3. Scattering matrices

Group-transfer cross sections form a scattering matrix, reflecting the rate at which neutrons are removed from group g to group g'. Calculation of these matrices requires averaging the differential scattering cross section over incident and emission energies:

$$\Sigma_{\mathrm{s},gg'} = \frac{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE \int_{E_{g'}}^{E_{g'-1}} dE'\, \Sigma_\mathrm{s}(\mathbf{r},E\to E')\phi(\boldsymbol{r},E)}{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE\, \phi(\boldsymbol{r},E)}. \quad (9)$$

However, instead of differential form, the continuous-energy ACE format used by Serpent provides the scattering data as total (1D) cross sections and associated probability distributions for the scattering angle. Inelastic reactions and thermal S(α,β) data also include distributions for the emission energy. Since obtaining the direct integral estimates in Eq. (9) becomes extremely complicated in practice, Serpent resorts to using analog estimators instead.

This means that every sampled scattering reaction from group h to group h' is counted, and the result stored in an $h\times h$ matrix. This multi-group data is then condensed into the few-group structure using a procedure very similar to that used for the reaction cross sections:

$$\Sigma_{\mathrm{s}0,gg'} = \frac{\displaystyle\sum_{h\in g}\sum_{h'\in g'} \Sigma_{\mathrm{s}0,hh'}\Phi_h}{\displaystyle\sum_{h\in g}\Phi_h}. \quad (10)$$

Group-transfer cross sections (10) form the P_0 scattering matrix. Serpent also calculates the P_1 matrix:

$$\Sigma_{\mathrm{s1},gg'} = \frac{\sum_{h\in g}\sum_{h'\in g'}\Sigma_{\mathrm{s1},hh'}\Phi_h}{\sum_{h\in g}\Phi_h}, \tag{11}$$

by weighting the scored tallies used to form $\Sigma_{\mathrm{s1},hh'}$ by the scattering cosine μ. The scattering cosine is obtained from the scalar product between the direction vectors of the incident and emitted neutron in the scattering event.[3]

Since neutron-multiplying scattering has a non-negligible impact on neutronics, the contribution of (n,2n), (n,3n), etc. reactions has to be included in the balance equations. In most core simulator codes this is accounted for by replacing the conventional scattering cross sections with scattering multiplication cross sections:

$$\Sigma_{\mathrm{sp}} = \Sigma_{\mathrm{s}} + 2\Sigma_{\mathrm{2n}} + 3\Sigma_{\mathrm{3n}} + \dots, \tag{12}$$

where Σ_{s} includes all elastic and inelastic reactions in which a single neutron is produced, and absorption cross section with the reduced absorption cross section:

$$\Sigma_{\mathrm{ra}} = \Sigma_{\mathrm{t}} - \Sigma_{\mathrm{s}} - \Sigma_{\mathrm{2n}} - 3\Sigma_{\mathrm{3n}} + \dots, \tag{13}$$

where Σ_{t} is the total cross section. Serpent calculates scattering production cross sections in matrix form, similar to the scattering cross sections described above. The reduced absorption cross sections are also calculated from the multi-group data.

Analog reaction rate estimators have the inherent drawback that only reactions sampled during the transport simulation make contributions in the tally scores. This is not considered a major problem for the scattering cross sections, since the reaction mode is so dominant in reactor applications (elastic scattering typically covers more than 90% of all neutron interactions in LWRs). Similar estimator is used for the fission spectra, by counting the number of neutrons emitted in group h. The number of scores is lower compared to scattering, but still sufficient for reasonable statistics. This is ensured by the fact that the k-eigenvalue criticality source simulation preserves the average number of emitted fission neutrons from cycle to cycle.

2.4. B_1 leakage correction

Performing spatial homogenization as an infinite-lattice calculation ignores the fact that neutrons are moving between assemblies, and that the inward or outward current contributes to local neutron balance. When the net current is forced to zero by reflective boundary conditions, there is, in general, an imbalance between the source and loss terms. When the transport problem is formulated into a steady-state k-eigenvalue equation, the balance between source and loss terms is sought by dividing the fission source by multiplication factor k, which differs from unity. In Monte Carlo criticality source simulations the same adjustment is done by scaling the average number of emitted fission neutrons for every source cycle in such way that the average population size is preserved.

This biasing of the fission source introduces a bias in the flux solution. The same problem concerns both deterministic solution methods and Monte Carlo simulation, and the root cause is that a physically sub- or super-critical system is forced into a steady-state condition by adjusting one of the source terms. The result is that the flux spectrum used for calculating the flux-volume averaged cross sections is distorted if the local multiplication factor differs significantly from unity. Leakage effects can be approximated to some extent by including the immediate surroundings in the modeled geometry, but as long as the calculation is performed on a simplified sub-set of the physical system, there is no physically consistent solution to this problem. Since, however, the entire purpose of spatial homogenization is not to simulate physical reality, but rather to produce input parameters for the next stage in the calculation chain, the flux solution can be artificially improved by leakage corrections.

The general idea is to introduce an additional source or loss term, representing the transfer of neutrons across the boundaries in such way that the local neutron balance is restored. The simplest correction, implemented in Serpent 2, is based on the homogeneous B_1 method. The basic assumption of the method is that the homogeneous multi-group flux can be represented as a product of the space-dependent part and the angular- and energy-dependent part. Substitution of a trial function of this form in the multi-group transport equation yields an eigenvalue problem for the spatial modes with material bucklings B_n^2 as the eigenvalues. In general, the greatest eigenvalue corresponds to the spectrum inside a homogenized geometry away from anisotropies and therefore this fundamental mode is chosen as the spectrum used in the method. After substituting the fundamental mode solution to the transport equation, the angular dependence of the solution is expanded in Legendre polynomials. The B_1 equations defining the current and flux spectrum can then be derived by limiting scattering anisotropy to first order (Stamm'ler and Abbate, 1983). The material buckling B^2 is iterated by repeatedly solving the B_1 equations until a solution yielding a critical multiplication factor $k_{\mathrm{eff}} = 1$ is found.

In Serpent the B_1 equations are formed for the homogenized system, using the multi-group cross sections obtained as in (5). The solution of the B_1 equations produces the critical flux spectrum Φ'_h, which can be used similar to the infinite spectrum Φ_h for collapsing cross sections into the final few-group form. The only difference in the procedure is the vector of multi-group fluxes used in summations similar to Eq. (3), which considerably simplifies the practical implementation of the leakage correction. The leakage correction is applied after each batch. The leakage-corrected homogenized parameters are generated along with the infinite-spectrum few-group cross section data according to the user request. Since the method involves solving the critical flux spectrum, it is naturally not applicable to reflectors or other non-multiplying regions.

[3] In fact, Serpent calculates the P_n scattering matrices up to Legendre polynomial order $n = 7$, but the higher-order data has so far not been tested or used by the authors.

2.5. Diffusion coefficients

Diffusion coefficient differs from homogenized reaction cross sections in that it has no continuous-energy counterpart in transport theory. It is instead most commonly defined as the scalar constant that relates flux gradient to neutron current density:

$$\mathbf{J}_g(\boldsymbol{r}) = -D_g \nabla \Phi_g(\boldsymbol{r}) \tag{14}$$

This relation is also known as Fick's law. The derivation of Eq. (14) is not completely trivial, and neither is the definition of D_g. It results from transport theory that the group-wise diffusion coefficient can be calculated from

$$D_g = \frac{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE \; \frac{1}{3\Sigma_{\mathrm{tr}}(\boldsymbol{r}, E)} \phi(\boldsymbol{r}, E)}{\displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE \; \phi(\boldsymbol{r}, E)} , \tag{15}$$

where the transport cross section is defined as:

$$\Sigma_{\mathrm{tr}}(\boldsymbol{r}, E) = \Sigma_{\mathrm{t}}(\boldsymbol{r}, E) - \Sigma_{\mathrm{s1}}(\boldsymbol{r}, E) , \tag{16}$$

and the the P_1 scattering cross section is given by the product of total scattering cross section and the average scattering cosine:

$$\Sigma_{\mathrm{s1}}(\boldsymbol{r}, E) = \overline{\mu} \Sigma_{\mathrm{s}}(\boldsymbol{r}, E) . \tag{17}$$

Serpent first calculates the multi-group transport cross section homogenized over the geometry:

$$\Sigma_{\mathrm{tr},h} = \Sigma_{\mathrm{t},h} - \Sigma_{\mathrm{s1},h} , \tag{18}$$

where $\Sigma_{\mathrm{s1},h}$ is obtained by summing over the columns of the P_1 scattering matrix (see Sec. 2.3). The few-group diffusion coefficient is then obtained by collapsing the inverse value:

$$D_g = \frac{\displaystyle\sum_{h \in g} \frac{1}{3\Sigma_{\mathrm{tr},h}} \Phi_h}{\displaystyle\sum_{h \in g} \Phi_h} . \tag{19}$$

The practical necessity of dividing group constant generation in two parts is again seen in that the P_1 scattering cross section involves the average scattering cosine. This parameter is not provided in the ACE format cross section libraries in continuous-energy form, which means that instead of evaluating the integrals in Eq. (15) directly, the calculation has to resort to analog estimators, which are used for calculating the average on the intermediate multi-group structure.

The calculation of D_g from (19) corresponds to the infinite flux spectrum. The solution of the B_1 equations provides the leakage-corrected diffusion coefficient:

$$D_g = \frac{\displaystyle\sum_{h \in g} J'_h}{|B| \displaystyle\sum_{h \in g} \Phi'_h} , \tag{20}$$

where B is the critical buckling, and J'_h and Φ'_h are the corresponding leakage-corrected current and flux spectra, respectively. As discussed in Sec. 2.4, these values are obtained as the result of the critical buckling iteration.

2.6. Effective delayed neutron parameters

Serpent 2 has the capability to calculate adjoint-weighted point kinetics parameters, i.e. effective delayed neutron fractions, precursor decay constants and prompt neutron lifetimes using the iterated fission probability (IFP) method (Leppänen et al., 2014a). The methodology was implemented mainly for the purpose of full-core Monte Carlo calculations, and it has certain limitations when it comes to spatial homogenization. Most importantly, these time constants are always calculated over the entire modeled geometry, which may not produce the correct result if the homogenized region covers only a part of the system, which is the case, for example, in assembly colorset configurations.

Effective delayed neutron fractions and precursor yields for the purpose of group constant generation are currently calculated separately, using the Meulekamp method (Meulekamp and van der Marck, 2006), in which the delayed neutron fraction is defined as the fraction of new fissions initiated by neutrons emitted as delayed. This is basically an approximation of the IFP method, taking into account only the first generation of descendants. The calculation is limited to the homogenized region by counting only neutron histories that originate from inside it.

2.7. Assembly discontinuity factors

The coupling between adjacent calculation nodes in nodal diffusion codes is accomplished via continuity conditions for neutron current and heterogeneous flux. The homogeneous flux, which forms the actual global solution, is discontinuous at the node boundaries, but coupled to the heterogeneous flux. This coupling is handled using assembly discontinuity factors (ADFs), which by definition are calculated as the ratio of two flux integrals:

$$F_{g,k} = \frac{\dfrac{1}{S_k} \displaystyle\int_{S_k} d^2r \int_{E_g}^{E_{g-1}} dE \; \phi(\boldsymbol{r}, E)}{\dfrac{1}{S_k} \displaystyle\int_{S_k} d^2r \; \Phi_g(\mathbf{r})} , \tag{21}$$

where ϕ is the heterogeneous flux, Φ is the homogeneous flux, and the integration is carried over boundary surface k.

In equivalence theory (Koebke, 1978; Smith, 1980), the global heterogeneous and homogeneous flux are represented by the corresponding local flux solutions at the fuel assembly level. When the lattice calculation is performed using reflective boundary conditions, the net boundary currents are reduced to zero, and the local homogeneous flux in Eq. (21) becomes constant. The constant flux shape and the preservation of reaction rate balance in the homogenized region imply that the surface-averaged homogeneous flux is equal to the volume-averaged heterogeneous flux, which means that Eq. (21) is reduced into:

$$F_{g,k} = \frac{\dfrac{1}{S_k} \displaystyle\int_{S_k} d^2r \int_{E_g}^{E_{g-1}} dE \; \phi(\boldsymbol{r}, E)}{\dfrac{1}{V} \displaystyle\int_V d^3r \int_{E_g}^{E_{g-1}} dE \; \phi(\boldsymbol{r}, E)} . \tag{22}$$

In practice this means that the ADFs can be obtained directly from the Monte Carlo simulation, using standard cell and surface flux tallies.

This, however, is no longer the case when the homogenized region forms only a part of the modeled geometry, and the boundary currents become non-zero. The homogeneous flux assumes a non-uniform shape, which has to be solved explicitly, and integrated over the boundary surface as in Eq. (21). Examples of such geometries include reflectors, where the neutron source is comprised of an inward boundary current from the core, and assembly colorsets, in which the homogenized assembly is modeled together with its immediate surroundings. For these cases Serpent provides a built-in diffusion flux solver based on an analytic solution corresponding to specified boundary conditions. The solution scheme is described in the following.

The diffusion equation in a homogenized medium can be written in matrix form as

$$-\mathbf{D}\triangle\mathbf{\Phi}(x,y,z) = \left(\mathbf{\Sigma}_{\text{sp}} - \mathbf{\Sigma}_{\text{t}}\right)\mathbf{\Phi}(x,y,z) + \frac{1}{k_{\text{eff}}}\mathbf{F}\mathbf{\Phi}(x,y,z)\,, \quad (23)$$

where $\mathbf{D} \in \mathbb{R}^{G\times G}$ and $\mathbf{\Sigma}_{\text{t}} \in \mathbb{R}^{G\times G}$ are diagonal matrices containing the multi-group diffusion coefficients and total cross-sections, $\mathbf{\Sigma}_{\text{sp}} \in \mathbb{R}^{G\times G}$ is the scattering production matrix and $\mathbf{F} = \chi(\nu\mathbf{\Sigma}_{\text{f}})^T \in \mathbb{R}^{G\times G}$ is the fission source term, with G being the number of energy groups in the few-group structure. Let

$$\widetilde{\mathbf{\Sigma}} = -\mathbf{D}^{-1}\left(\mathbf{\Sigma}_{\text{sp}} - \mathbf{\Sigma}_{\text{t}} + \frac{1}{k_{\text{eff}}}\mathbf{F}\right)\,. \quad (24)$$

Diffusion equation can now be written

$$\triangle\mathbf{\Phi}(x,y,z) = \widetilde{\mathbf{\Sigma}}\,\mathbf{\Phi}(x,y,z)\,. \quad (25)$$

It can be seen that any matrix function of the form

$$\mathbf{f}(x,y) = e^{\mathbf{\Sigma_1}x+\mathbf{\Sigma_2}y+\mathbf{\Sigma_3}z}\,, \quad \mathbf{\Sigma}_1^2 + \mathbf{\Sigma}_2^2 + \mathbf{\Sigma}_3^2 = \widetilde{\mathbf{\Sigma}} \quad (26)$$

is a solution of Eq. (25). Matrix functions according to Eq. (26) are called the *basis functions* of the diffusion equation in the homogenized medium. The solution of Eq. (25) can be written using an arbitrary combination of basis functions and the selection depends on the geometry and boundary conditions. Boundary conditions are typically specified as net currents over segments of the boundary surface. In this case, an unambiguous solution can be constructed when the number of basis functions equals the number of boundary conditions. The most recent version of Serpent 2 supports two-dimensional geometries bounded by planes and square, rectangular and hexagonal prisms. The net currents are calculated using standard surface current tallies. Corners of the prismatic geometries can be included in the calculation, which doubles the number of basis functions and boundary conditions. Net currents at corners are calculated on segments extending 10% of the surface width in both directions.

Basis functions used in the solution are of the form

$$\mathbf{f}_i = e^{\sqrt{\widetilde{\Sigma}}\,\mathbf{n}_i\cdot\mathbf{r}}\,, \quad (27)$$

where the square root of a matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is defined as any matrix in $\mathbb{C}^{n\times n}$ satisfying

$$\left(\sqrt{\mathbf{A}}\right)^2 = \mathbf{A}\,, \quad (28)$$

and $\mathbf{n}_i$ is the direction vector of basis function i. The directions are selected to match the surface normals. If corners are included, the additional basis functions are oriented in the direction of the diagonals. In an infinite square prism, for example, this approach corresponds to employing the following basis functions:

- $\mathbf{f}_{\text{x}}^{\pm} = e^{\pm\sqrt{\widetilde{\Sigma}}x}$
- $\mathbf{f}_{\text{y}}^{\pm} = e^{\pm\sqrt{\widetilde{\Sigma}}y}$
- $\mathbf{f}_{\text{x+y}}^{\pm} = e^{\pm\frac{\sqrt{\widetilde{\Sigma}}}{2}(x+y)}$
- $\mathbf{f}_{\text{x-y}}^{\pm} = e^{\pm\frac{\sqrt{\widetilde{\Sigma}}}{2}(x-y)}$

The first four basis functions correspond to boundary surfaces and the last four to corners.

According to Fick's law, $J_g = -D_g\nabla\phi_g$, and the boundary conditions can be written using the normal derivatives of the basis functions. Let

$$\mathbf{F}_{i,\text{x}}^{\pm} = \int_{S_i}\frac{\partial\mathbf{f}_{\text{x}}^{\pm}}{\partial n}\,dS \;\in \mathbb{C}^{G\times G}\,. \quad (29)$$

Assuming both boundary and corner net currents are used as boundary conditions, the net current across S_i can be written

$$\mathbf{J}_{\text{net},S_i} = -\mathbf{D}\left(\mathbf{F}_{i,\text{x}}^{+}\mathbf{c}_1 + \mathbf{F}_{i,\text{x}}^{-}\mathbf{c}_2 + \ldots\mathbf{F}_{\text{x-y}}^{-}\mathbf{c}_8\right)\,. \quad (30)$$

Writing down the boundary conditions for all surfaces $S_1,\ldots,S_8$, we obtain a linear system from which coefficients $\mathbf{c}_1,\ldots,\mathbf{c}_8 \in \mathbb{C}^G$ can be solved. After solving the coefficients, the solution can be computed simply as

$$\mathbf{\Phi}(x,y) = \mathbf{f}_{\text{x}}^{+}(x,y)\mathbf{c}_1 + \mathbf{f}_{\text{x}}^{-}(x,y)\mathbf{c}_2 + \ldots + \mathbf{f}_{\text{x-y}}^{-}(x,y)\mathbf{c}_8\,. \quad (31)$$

The procedure is similar for hexagonal geometries with the only difference being that $\mathbf{n}_i, i = 1,\ldots,12$, in Eq. (27) are chosen as the normal vectors of the hexagonal prism.

The numerical solution scheme can be summarized as:

1. Matrix $\widetilde{\mathbf{\Sigma}}$ is formed.
2. The complex Schur form $\widetilde{\mathbf{\Sigma}} = \mathbf{TUT}^*$ is computed. Numerical implementation is based on Hessenberg reduction and QR decomposition using Householder transformations.
3. Basis functions are formed using the Parlett method (Parlett, 1976). In 2D, the number of basis functions is 4 or 8 in rectangular geometry and 6 or 12 in hexagonal geometry, depending on whether net currents over corners are included in the boundary conditions.
4. Normal derivatives of basis functions are integrated over assembly boundaries and corners.

5. Coefficient vectors of the basis functions are solved from the resulting linear system using Gaussian elimination.

The solution provides the unknown coefficients for Eq. (31), i.e. the homogeneous diffusion flux, which can then be integrated over boundary surfaces and corners to obtain the assembly discontinuity factors from (21).

2.8. Pin-power form factors

The intra-nodal homogeneous flux solution carries no information about the heterogeneity of flux and power distributions inside the node, which may be imperative for the estimation of peak cladding temperatures and other parameters relevant for safety analyses. The information lost in the process of homogenization can, however, be recovered by projecting the local power distribution obtained from the assembly-level calculation on top of the global flux solution. This is done in a process of pin-power reconstruction, using form factors:

$$p_{g,j} = \frac{\dfrac{1}{V_j}\displaystyle\int_{V_j} d^3r \int_{E_g}^{E_{g-1}} dE\ \kappa\Sigma_\mathrm{f}(\boldsymbol{r},E)\phi(\boldsymbol{r},E)}{\dfrac{1}{V_j}\displaystyle\int_{V_j} d^3r\ \Phi_g(\mathbf{r})}\,, \qquad (32)$$

where $\kappa\Sigma_\mathrm{f}$ is the fission energy production cross section, ϕ is the heterogeneous flux, Φ is the homogeneous flux, and the integrals are extended over the volume of pin j.

Serpent produces these form factors by relating the pin indices to a lattice structure. Similar to the calculation of ADFs, the procedure depends on the boundary conditions. If the homogenized geometry is limited to a reflective boundary, the net currents are reduced to zero, and the homogeneous flux becomes uniform. The integral in the denominator of Eq. (32) can then be replaced by the integral of the heterogeneous flux:

$$p_{g,j} = \frac{\dfrac{1}{V_j}\displaystyle\int_{V_j} d^3r \int_{E_g}^{E_{g-1}} dE\ \kappa\Sigma_\mathrm{f}(\boldsymbol{r},E)\phi(\boldsymbol{r},E)}{\dfrac{1}{V}\displaystyle\int_{V} d^3r \int_{E_g}^{E_{g-1}} dE\ \phi(\boldsymbol{r},E)}\,, \qquad (33)$$

and both values can be obtained directly using standard Monte Carlo tallies.

When the net boundary current is non-zero, the calculation of form factors requires the explicit solution of the homogeneous flux, which is integrated over the pin volume as in Eq. (32). This solution is provided by the same deterministic solver as described in Sec. 2.7.

2.9. Albedos

Albedos and partial albedos are used in some core simulator codes to set up the boundary conditions between reflectors and other non-multiplying regions with the active core. The total albedo is a matrix describing the fraction of escaped neutrons that return into the fuel, i.e. the ratio of neutrons passing through boundary surface S in group g and returning in group g':

$$\alpha_{gg',k} = \frac{\displaystyle\int_{E_{g'}}^{E_{g'-1}} dE \int_S d\mathbf{S}\cdot\mathbf{J}_g^-(\boldsymbol{r},E)}{\displaystyle\int_{E_g}^{E_{g-1}} dE \int_S d\mathbf{S}\cdot\mathbf{J}^+(\boldsymbol{r},E)}\,, \qquad (34)$$

where $\mathbf{J}^+$ is the escaping current and $\mathbf{J}_g^-$ is the returning current component formed by neutrons that have escaped the active core in energy group g. The current integrals in Eq. (34) are easily evaluated using standard Monte Carlo tallies and energy group flagging.

Partial albedos are used in a similar way, but they allow accounting for the fact that neutrons escaped from one node may return into the active core through the boundary of a completely different node. The non-multiplying region coupled to core nodes is represented by a response matrix, in which each element gives the probability of transition from energy group g to g' and face k to k':

$$\alpha_{gg',kk'} = \frac{\displaystyle\int_{E_{g'}}^{E_{g'-1}} dE \int_{S_{k'}} d\mathbf{S}\cdot\mathbf{J}_{gk}^-(\boldsymbol{r},E)}{\displaystyle\int_{E_g}^{E_{g-1}} dE \int_{S_k} d\mathbf{S}\cdot\mathbf{J}^+(\boldsymbol{r},E)}\,, \qquad (35)$$

where $\mathbf{J}_{gk}^-$ is the outward current component formed by neutrons that have entered the volume in energy group g through face k. Partial albedos are used, for example, in VTT's HEXTRAN transient code, for the modeling of VVER-440 control elements (Siltanen et al., 2003).

Serpent calculates albedos and partial albedos for user-specified surfaces within the geometry. The supported surface types are the same as in the ADF calculation, i.e. planes and infinite square, rectangular and hexagonal prisms.

3. Automated burnup sequence

The direct connection between neutronics and state variables is lost in the process of homogenization. The same applies to fuel burnup, as the group constant data carries no information on the detailed isotopic compositions. Feedback effects and fuel depletion in core calculations are instead accounted for by interpolating between discrete states, which means that the building blocks of the full-scale model have to cover the complete range of operating conditions. The interpolation and the way the data is parameterized depends on the core simulator, but it is not uncommon that the production of the full set of group constants requires repeating the assembly-level calculation thousands of times. This poses a major computational challenge, not only because of the high CPU cost of the Monte Carlo simulation, but also because of the sheer volume of data that needs to be handled and processed.

Since the procedure involves burnup calculation, and the local operating conditions inside the homogenized fuel assembly also affect how the materials are depleted, the state-points by which the group constant data is parameterized are not completely independent. The calculations are instead divided into:

i) Branch variations, taking into account the momentary changes in the operating conditions, such as fuel temperature, moderator density and temperature, boron concentration and insertion of control rods inside the core

ii) History variations, taking into account conditions that persist for an extended period of time, thus affecting the way the fuel is burnt, such as moderator temperature and density, boron concentration and positioning of control rods

Preparing each input and setting up the calculations by hand is not a viable option, and Serpent 2 offers an automated burnup sequence for performing the branch variations. It should be noted that even though this is a relatively new feature, built in burnup capability has been available in Serpent for years. An overview of the general methodology is provided in (Leppänen et al., 2015), and the depletion solver based on the Chebyshev Rational Approximation Method (CRAM) is introduced in (Pusa and Leppänen, 2010).

The automated sequence works by running a burnup calculation for a single history case, after which a number of restart calculations are performed for selected burnup points. For each restart the code invokes a number of user-specified variations in the input, corresponding to the branches to different state points. The available variations in version 2.1.26 include:

- Change in material temperature and density
- Replacement of one material with another
- Replacement of one universe with another
- Application of a universe transformation
- Adjustment of normalization

The changes in material temperatures and densities can be used to account for variations in the thermal hydraulic state. The adjustment of cross section library temperatures is handled using the built-in Doppler-broadening preprocessor routine (Viitanen, 2009),[4] and moderator temperature effects by interpolation between S(α,β) tables (Viitanen and Leppänen, 2016). Coolant boron branches can be invoked by changing the entire material, and control rod branches in 2D calculations by replacing the universe, for example, an empty guide tube with a rodded tube. The capability to apply universe transformations allows moving and rotating different parts of the geometry, which is practical for positioning the control rods if the homogenization is performed in 3D. Adjustment in normalization changes the level of flux and fission power.

The standard Matlab-format output of Serpent 2 contains all calculated results, including the homogenized few-group constants. However, because of the large amount of data, these files can be difficult to post-process. When the automated calculation sequence is run, the code produces another output file, in which the group constant data is organized in a way easily read by processing scripts. The parameters included in the output can be selected by the user. It is also possible to define variables, which are passed as-is into the output file. This allows including additional information on each calculation case, which may be useful when the output is read and processed.

[4] Serpent 2 also provides an on-the-fly temperature treatment routine (Viitanen, 2015), which is used especially for modeling non-uniform temperature distributions in coupled multi-physics simulations. The temperature branches in group constant generation typically involve uniform changes throughout the homogenized geometry, for which the Doppler preprocessor is well sufficient.

4. Known limitations and flaws

The procedures used in Serpent 2 for group constant generation involve several subroutines and large quantities of intermediate data. The methodology has its limitations and there are several compromises that had to be made during the course of development. It is important that the code user is aware of the potential pitfalls, and does not apply the code as a black box. Some input options, in particular related to the batching of results and selection of the intermediate multi-group structure, may have an impact on computational performance and memory footprint, but also on the results of the calculation.

4.1. General procedures and multi-group structure

The amount of data collected and stored during the transport simulation depends on the number of energy groups in the intermediate multi-group structure. The discretization of the energy variable has no effect on most group constants homogenized in the infinite flux spectrum, but insufficient energy resolution may be reflected in the leakage-corrected values. This is also the case for the infinite-spectrum diffusion coefficient. Using too many energy groups, on the other hand, may result in excessive memory usage. The distribution of collected scores over a large number of energy groups may also lead to deteriorated statistics, and consequently, convergence problems with the deterministic B_1 solver.

Similar convergence problems caused by insufficient statistics are encountered when modeling fast-spectrum systems, as the energy groups in the thermal region may be left completely without scores. A practical workaround is to merge the lowest groups together both in the multi- and the few-group structure. This is not likely to affect the outcome of the core simulations, as demonstrated by Fridman and Shwageraus (2013). Even so, the root cause of the problem is not in the B_1 method, but rather in the implementation of the numerical solver in Serpent 2, which cannot handle zeros in the input data. This flaw is planned to be corrected in future updates.

The homogeneous diffusion flux solver used for calculating ADFs and pin-power form factors may also fail because of poor statistics, although the calculation is based on the few-group, not the multi-group structure. An efficient way to improve the batch-wise statistics without increasing the population size is to increase the batch interval from the default of 20 source cycles, but it should be noted that if the total number of batches falls too low, the statistical error estimates may no longer be reliable. It is even possible to include the entire simulation in a single batch. This minimizes the stochastic noise in parameters

passed to the deterministic solvers, but the drawback is that all information on statistical errors is lost.

Both deterministic solvers run into problems when the impact of the associated correction becomes negligible – the B_1 solver when the system is close to criticality and the homogeneous diffusion flux solver when the net boundary currents are close to zero. Serpent tries to identify such conditions and fall back to the simplified methodology. For ADFs and pin-power form factors this means assuming a uniform distribution for the homogeneous flux and for leakage corrected cross sections using the infinite spectrum instead.

Serpent uses OpenMP threading to distribute the simulation of neutron histories over multiple CPU cores. The processing of intermediate multi-group data after each batch, on the other hand, is currently done in serial. This is also the case for the deterministic solvers. All CPU time spent outside the threaded loop limits the parallel scalability, so the selection of multi-group structure and batching interval also affect computational performance. Group constant generation always adds some computational overhead, and if the output data is not needed, it is advised to switch the procedure off.

The production of homogenized cross sections is currently limited to macroscopic data, but there have been several requests from the user community to include microscopic cross sections, for example, for the purpose of micro-depletion in fuel cycle simulations. The reason why this is currently not an option is in part related to the way group constant generation is handled. Storing the intermediate multi-group data for a large number of isotopic microscopic reaction cross sections may lead to very large memory footprint. It would be possible to produce the few-group data directly, as in Eqs. (1) and (9), but only in the infinite spectrum. Another problem is related to the use of the analog estimator for the calculation of scattering matrices. It was noted in Sec. 2.3 that poor statistics is not a problem for the macroscopic cross sections because scattering is the dominant reaction mode. This is not the case for microscopic cross sections, since the atomic density of the nuclide may be very low or even zero.

4.2. Leakage correction

The homogeneous B_1 method is only one option to perform a leakage correction for the flux spectrum, and the methodology used in Serpent was adopted from deterministic transport theory. This approach has the advantage that both infinite- and critical-spectrum-averaged group constants can be produced within a single run, but it also ignores some of the advantages of Monte Carlo simulation, including the capability to obtain the integral estimates in Eq. (1) without discretizing the energy variable. The leakage correction is applied to a multi-group flux, and the group structure sets a limit for the resolution of spectral effects. There exist also Monte Carlo specific methods in which the correction is applied during the transport simulation, for example, based on the use of complex weights (Yamamoto, 2012) or albedo iteration (Yun and Cho, 2010). One of such methods was developed for Serpent by Dorval (2016a), but at the time of this writing the capability was not yet included in the distributed version (update 2.1.26).

It should also be noted that no leakage correction can currently be applied to transmutation cross sections during burnup calculation, even though it is known that fuel depletion is strongly influenced by spectral effects. This is mainly because the leakage-corrected spectrum is calculated separate from the transport simulation, during which the data for the depletion solver is collected. A natural solution for this problem would be to apply one of the Monte Carlo specific corrections discussed above, in which the spectral effects are accounted for within the transport simulation.

4.3. Effective delayed neutron fractions

The number of delayed neutron precursor groups is fixed in the evaluated nuclear data files. Libraries based on JEFF-3.1 and later versions use a structure of 8 groups, while other evaluations typically rely on 6 groups. This group structure cannot be changed, which may limit the applicability of the produced data. It should also be noted that Serpent fixes the structure to either 6 or 8 groups, and if the number of groups for some nuclide differs from the fixed value, the data is discarded altogether. The selection is primarily based on the major actinides: ^{235}U, ^{239}Pu, ^{233}U and ^{238}U. If none of these isotopes are included in the initial composition, the structure used by the first listed actinide is selected.

As mentioned in Sec. 2.6, the calculation of effective delayed neutron parameters relies on the Meulekamp method. This is an intermediate solution, planned to be replaced later by a deterministic solver and importance weighting by the multi-group adjoint flux.

4.4. Homogeneous diffusion flux solver

The local homogeneous flux solution used in the calculation of ADFs and pin-power form factors when the net boundary currents for the homogenized region differ from zero should match the diffusion flux solution in the nodal code. The routine in Serpent 2 was developed specifically for the ARES code (Mattila, 2003). It is important to realize that if the data is used in a nodal diffusion code with boundary conditions and intra-nodal flux solution significantly different from what is described in Sec. 2.7, the results of the core calculation may be compromised. For the moment the solution is limited to two-dimensional rectangular and hexagonal geometries.

4.5. Other methodological limitations

One methodological issue that extends beyond group constant generation is the question of how to normalize reaction rates to fission power? The recoverable energy released in fission is divided between the kinetic energy of the fission fragments, neutrons and prompt fission gammas. This is supplemented by a delayed component following the decay of radioactive fission products, as well as additional energy released in other neutron reactions, in particular (n,γ). Even though the Monte Carlo method allows very accurate modeling of the direct heating effects of neutrons and photons, such approach may not be the best choice for lattice physics calculations. The fact

that the homogenized assembly is separated from its actual surroundings by reflective boundary conditions ignores in any case the energy transfer beyond the geometry boundaries.

Fission power in Serpent 2 is currently calculated based on a simple empirical model, by assuming that each ^{235}U fission deposits 202.27 MeV of energy directly in the fuel, and the corresponding values for other actinides are scaled according to the ratios of the fission Q-values. Since this normalization fixes the ratio of reaction rates to fission power, as well as the relation between irradiation time and burnup, the approximation does have an effect on the rate at which the initial composition is depleted and new isotopes are accumulated in the fuel. The user should be aware of this approximation, and if necessary, adjust the default values by input options.

It was briefly mentioned in Sec. 2.1 that the meta-stable state of fission product poison ^{135}Xe is lumped together with the ground state in the calculation of poison cross sections. This is a practical necessity, as ^{135m}Xe lacks the cross sections in all major nuclear data evaluations, and because of its relatively short half-life of 15 minutes, the effect of this simplification is not considered very significant. Kim and Kim (2014) have pointed out, however, that including the meta-stable state may have a noticeable effect in transient analyses. Including separate absorption and production cross sections for ^{135m}Xe should therefore be an option for group constant generation as well, provided that the data is included in the cross section library.

4.6. Automated burnup sequence

The automated burnup sequence is currently capable of handling only branch variations, and setting up the history cases has to be done separately. The possibility of including histories in the calculation sequence has been considered, but at this point such approach does not seem practical. Creating the history inputs with a simple pre-processing script, or even by hand, is relatively straightforward. Implementing such capability in Serpent would only lead to overly complicated input structures without much added value to the user.

The Doppler-broadening preprocessor routine used for invoking the temperature variations cannot currently adjust the unresolved resonance probability table data,[5] which may have a noticeable effect in fast-spectrum systems. The problem can be avoided by selecting the branch temperatures in such way that they match the library temperatures, which are typically provided in 300K intervals. The adjustment is made based on the closest available temperature below the given value, and if the temperatures match, there is no need to broaden the cross sections or adjust the probability tables at all.

5. Conclusions, discussion and plans for future work

The Serpent Monte Carlo reactor physics burnup calculation code has been developed for the purpose of spatial homogenization and other reactor physics applications since 2004. The code has been successfully used for producing group constants for various deterministic core simulators, although most of these studies have so far been limited to simplified applications, most typically initial core zero-power calculations. In recent years, however, some experienced Serpent users have started moving from proof-of-concept type of studies towards more challenging practical applications, involving fuel cycle and transient simulations.

The continuous-energy Monte Carlo method does have some clear advantages compared to traditional deterministic lattice transport codes, such as the capability to produce the ideal reference solution for the validation of the multi-stage calculation scheme. Even so, the scope of applications is still limited by the high computational cost of the transport simulation. The practical applicability of Serpent has also been limited by insufficient documentation, of both the input syntax and the methodology used for spatial homogenization. In an effort to correct this deficiency, an on-line Serpent Wiki was recently set up to fill in for the missing User Manual.[6] The purpose of this paper, on the other hand, was to collect the methodological description scattered through various earlier publications into a single up-to-date document, and to fill in the missing pieces.

Serpent is still a developing code, and advanced methods for spatial homogenization is one of the major topics for future work. As discussed above, there are several flaws and problems in the methodology waiting to be corrected. Plans for future work also include implementing new features requested by Serpent users, in particular the capability to produce microscopic cross sections and axial discontinuity factors.

Many of the limitations and weak points in the methodology are ultimately related to the fact that the energy and angular transfer functions of scattering reactions are not provided in double-differential form, but rather as probability distributions separate from the cross sections. This complicates the calculation of scattering matrices and higher scattering moments, and practically necessitates resorting to the use of analog estimators. An elegant solution to the problem would be to extract this information from the ACE data and pre-process it into a form directly usable with the reaction rate tallies, as was done by Nelson and Martin (2014) for the OpenMC code. Whether or not this is a viable solution for Serpent as well is not clear at this point.

As noted in Sec. 2.5, the most conventional approach to diffusion theory is to use a scalar constant to relate neutron current density to flux gradient by Fick's law (14), but it is also possible to include directional dependence in the diffusion process. The currently distributed version of Serpent 2 (version 2.1.26) does not yet support the calculation of directional diffusion coefficients, but recent work on the topic has been carried out in a related doctoral thesis (see Dorval and Leppänen, 2015; Dorval, 2016b). The methodology, together with the new leakage model developed as part of the same thesis work (Dorval, 2016a), will be included in the official version of Serpent 2 in the future.

[5] It should also be noted that probability table sampling is switched off by default in Serpent 2.

[6] The User Manual for Serpent 1 (Leppänen, 2015) is considered outdated, and in any case is missing most of the new features and capabilities in Serpent 2.

Acknowledgments

This work was funded in part by the Finnish National Research Programme on Nuclear Power Plant Safety (SAFIR 2018 / MONSOON).

References

Dorval, E., 2016a. A comparison of Monte Carlo methods for neutron leakage at assembly level. Ann. Nucl. Energy 87, 591–600.

Dorval, E., 2016b. Directional diffusion coefficients and leakage-corrected discontinuity factors: Implementation in Serpent and tests. Ann. Nucl. Energy 87, 101–112.

Dorval, E., Leppänen, J., 2015. Monte Carlo current-based diffusion coefficients: Application to few-group constants generation in Serpent. Ann. Nucl. Energy 78, 104–116.

Fridman, E., Leppänen, J., 2011. On the use of the Serpent Monte Carlo code for few-group cross section generation. Ann. Nucl. Energy 38, 1399–1405.

Fridman, E., Leppänen, J., 2012. Revised methods for few-group cross section generation in the Serpent Monte Carlo code. In: PHYSOR 2012. Knoxville, TN, USA, Apr. 15-20, 2012.

Fridman, E., Shwageraus, E., 2013. Modeling of SFR cores with Serpent-DYN3D codes sequence. Ann. Nucl. Energy 53, 354–363.

Kaltiaisenaho, T., Leppänen, J., 2014. Analysing the statistics of group constants generated by Serpent 2 Monte Carlo code. In: PHYSOR 2014. Kyoto, Japan, Sep. 28 - Oct. 3, 2014.

Kim, J., Kim, Y., 2014. Preliminary Study of the Impact of Xe-135m on the PCR of CANDU. In: PHYSOR 2014. Kyoto, Japan, Sep. 28 - Oct. 3, 2014.

Koebke, K., 1978. A new approach to homogenization and group condensation. IAEA-TECDOC-231, International Atomic Energy Agency, 1978.

Leppänen, J., 2007. Development of a new Monte Carlo reactor physics code. D.Sc. Thesis, Helsinki University of Technology, 2007.

Leppänen, J., 2009. Two practical methods for unionized energy grid construction in continuous-energy Monte Carlo neutron transport calculation. Ann. Nucl. Energy 36, 878–885.

Leppänen, J., 2015. Serpent – a continuous-energy Monte Carlo reactor physics burnup calculation code. VTT Technical Research Centre of Finland (June 18, 2015).

Leppänen, J., Aufiero, M., Fridman, E., Rachamin, R., van der Marck, S., 2014a. Calculation of effective point kinetics parameters in the Serpent 2 Monte Carlo code. Ann. Nucl. Energy 65, 272–279.

Leppänen, J., Isotalo, A., 2012. Burnup calculation methodology in the Serpent 2 Monte Carlo code. In: PHYSOR 2012. Knoxville, TN, USA, Apr. 15-20, 2012.

Leppänen, J., Mattila, R., Pusa, M., 2014b. Validation of the Serpent-ARES code sequence using the MIT BEAVRS benchmark - initial core at HZP conditions. Ann. Nucl. Energy 69, 212–225.

Leppänen, J., Pusa, M., Viitanen, T., Valtavirta, V., Kaltiaisenaho, T., 2015. The Serpent Monte Carlo code: Status, development and applications in 2013. Ann. Nucl. Energy 82, 142–150.

Mattila, R., 2003. Three-dimensional analytic function expansion nodal model. In: ANFM 2003. Hilton Head Island, SC, USA, Oct. 5-8, 2003.

Meulekamp, R., van der Marck, S., 2006. Calculating the effective delayed neutron fraction with Monte Carlo. Ann. Nucl. Energy 152, 142–148.

Nelson, A., Martin, W., 2014. Improved Monte Carlo Tallying of Multi-Group Scattering Moment Matrices . Trans. Am. Nucl. Soc. 110, 15–19.

Parlett, B. N., 1976. A Recurrence among the Elements of Functions of Triangular Matrices. Linear Algebra and its Applications 14, 117–121.

Pusa, M., Leppänen, J., 2010. Computing the matrix exponential in burnup calculations. Nucl. Sci. Eng. 164, 140–150.

Siltanen, P., Kaloinen, E., Wasastjerna, F., 2003. Simulation of reactor scram experiments in the Loviisa and Mochovce VVER reactors. In: XI Reactor Physics Meeting in the Nordic Countries. Helsinki, Finland, Apr. 9-10, 2003.

Smith, K., 1980. Spatial homogenization methods for light water reactor analysis. Ph.D. Thesis, Massachusetts Institute of Technology, 1980.

Stamm'ler, R., Abbate, M., 1983. Methods of Steady-State Reactor Physics in Nuclear Design. Academic Press Inc.

Viitanen, T., 2009. Implementing a doppler-preprocessor of cross section libraries in reactor physics code serpent. M.Sc. Thesis, Helsinki University of Technology.

Viitanen, T., 2015. Development of a stochastic temperature treatment technique for Monte Carlo neutron tracking. D.Sc. Thesis, Aalto University.

Viitanen, T., Leppänen, J., 2016. New interpolation capabilities for thermal scattering data in Serpent 2. In: PHYSOR 2016. Sun Valley, ID, USA, May 1-5, 2016.

Yamamoto, T., 2012. Monte Carlo method with complex weights for neutron leakage-corrected calculations and anisotropic diffusion coefficient generations. Ann. Nucl. Energy 50, 141–149.

Yun, S., Cho, N., 2010. Monte Carlo depletion under leakage-corrected critical spectrum via albedo search. Nucl. Eng. Technol. 42, 271–278.