



Original Article

Comprehensive investigation of the Ronen method in slab geometry

Roy Gross^a, Johan Cufe^a, Daniele Tomatis^b, Erez Gilad^{a,*}^a The Unit of Nuclear Engineering, Ben-Gurion University of the Negev, Beer-Sheva, 8410501, Israel^b Université Paris-Saclay, CEA, DES, Service d'études des réacteurs et de mathématiques appliquées (SERMA), F-91191, Gif-sur-Yvette, France

ARTICLE INFO

Article history:

Received 16 July 2022

Received in revised form

2 September 2022

Accepted 24 September 2022

Available online 30 September 2022

Keywords:

Ronen method

Neutron diffusion

Integral transport

Non-linear transport correction

Sood benchmark

ABSTRACT

A comprehensive investigation of the Ronen method is performed in homogeneous and heterogeneous slab problems from the Sood benchmark, considering isotropic and linearly-anisotropic problems. Three finite differences implementations are exercised and compared. The results are compared to reference solutions using one and two energy groups. The validation is performed for the criticality eigenvalue and the fundamental neutron flux distribution. The results demonstrate the significantly improved accuracy achievable by the Ronen method using a broad set of problems. For standard convergence tolerances, the maximal deviation in criticality eigenvalue is less than ten pcm, and the maximal deviation in the spatial distribution of the flux is less than 2%, always located near sharp interfaces or vacuum boundaries.

© 2022 Korean Nuclear Society, Published by Elsevier Korea LLC. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

The Ronen method (RM), evolved from the hypothesis of Ronen in 2004 [1], is an iterative scheme that provides an improved estimation of the criticality eigenvalue (k_{eff}) and the neutron flux distribution using the diffusion approximation [2–4]. Ronen derived an accurate relation between the integral expression of the net current and the diffusion coefficient. Denote the neutron flux distribution by ϕ , the Fick's law approximation for the neutron current by \mathbf{J}^D , and the exact neutron current by \mathbf{J} . Ronen hypothesized that there exist diffusion coefficients such that

$$\mathbf{J}^D(\mathbf{r}, E) = -D(\mathbf{r}, E)\nabla\phi(\mathbf{r}, E) \cong \mathbf{J}(\mathbf{r}, E), \quad (1)$$

and postulated that by adjusting the degrees of freedom in the diffusion coefficient, i.e., its spatial and spectral dependencies, an improved diffusion approximation could be obtained. Specifically, Ronen suggested that better estimates of the diffusion coefficient could be possible by approximating the current with an integral expression from transport theory, using however an isotropic source determined with the same flux distribution from diffusion theory. A straightforward implementation in slab geometry is to recalculate the diffusion coefficient provided a known flux ϕ , still obtained with neutron diffusion,

$$D(x, E) \leftarrow -J(\phi)/\partial_x\phi. \quad (2)$$

Unfortunately, rigorous mathematical proof for this assumption was not provided and it is yet to be derived.

Tomatis and Dall'Osso [2] were the first to suggest an iterative scheme for the implementation of the RM to obtain a more accurate diffusion coefficient producing an improved neutron flux solution by employing drift corrections to the interface currents. The use of corrections on the net current that are proportional to the local flux value (drift) was meant avoiding possible indeterminate division occurring in regions of flat flux.

A pronounced advantage of the RM is the utilization of standard neutron diffusion approximation codes to achieve transport-level accuracy. The iteration procedure of the RM (Eq. (2)) employs the integral transport expression only to calculate the total current assuming the scalar flux (i.e., the solution of the diffusion approximation) for the neutron source. This way, the diffusion approximation operator is inverted successively throughout the iterations, but no transport operator is inverted at any time. In essence, the RM simulates the transport calculation with a series of diffusion calculations, driving the solution towards transport. For complex or large enough systems, the computational cost of inverting the transport operator is avoided and replaced by a series of much less demanding diffusion calculations.

In 2011, Tomatis and Dall'Osso [2] demonstrated numerically the Ronen postulate utilizing a two-group one-dimensional homogeneous slab problem, representative of a realistic PWR fuel assembly.

* Corresponding author.

E-mail address: gilade@bgu.ac.il (E. Gilad).

They eliminated the possible division by zero in Eq. (2) in cases of vanishing flux gradients by adopting surface current corrections in the form of drift terms, similar to the ones used in Coarse Mesh Finite Difference (CMFD) implementations [5].

Recently, Gross et al. [4] demonstrated the RM's high accuracy (comparing to standard diffusion approximation) utilizing two-group heterogeneous slab problems. In their paper, Gross et al. derived analytic expressions for the accurate relation between the scalar flux, the total current, the angular moments of the angular flux, and the anisotropic scattering source. These expressions allow to consider the first moment of the scattering expansion (i.e., linear anisotropy) in the source when using neutron diffusion. The RM's accuracy was evaluated against other numerical solutions using discrete ordinate S_N code, still using purely isotropic sources.

Gross et al. [4] provided valuable insights into the mechanism underlying the RM and enabling it to achieve improved solutions compared to diffusion ones, i.e., the *localized* nature of the current correction terms. Gross et al. showed that highly localized spatial and spectral variations in the current correction terms enable more accurate reconstruction of flux gradients in regions where the diffusion approximation invalidates, e.g., near bare boundaries, next to interfaces, or in the vicinity of strong absorbers. Away from these regions, where the diffusion approximation is valid, the current corrections vanish. However, the set of numerical problems was rather limited and was chosen mainly to explore the behaviour of the RM's correction terms.

Tomatis et al. [3] studied the RM in curvilinear geometries, and derived analytic expressions of the neutron current, evaluated by integral equations using the framework of the collision probability method. Specifically, all the integrals were written by making use of the first-flight escape probabilities. These expressions were integrated as part of the RM numerical algorithm. As a consequence of the limitation due to probabilities determined by the only isotropic emission, Tomatis et al. noticed that the RM could not reproduce reflection in halved slabs of a few mean free paths, that is where even moments other than the scalar flux can be relevant. A fix-up implemented in the same iterative process was proposed to match the expected vanishing net current. A collision probability solver based on the same escape probabilities was developed providing the reference solutions.

In their work, results were represented for one-group homogeneous critical problems, with isotropic scattering, and the fundamental neutron flux and eigenvalue were compared. Additionally, slow convergence of the neutron flux was noticed necessitating many RM iterations. To overcome this issue, they implemented the Damped Anderson Acceleration with Epsilon Monotonicity (DAAREM) [6,7], which was applied only for the flux, resulting in faster run-time and higher accuracy in the final results. Nevertheless, only a limited amount of test problems were examined.

In this article, the achievable accuracy of the RM is further studied numerically. The solutions obtained by the RM are comprehensively investigated and compared to reference solutions available from the Sood benchmark test suite [8], relying on one and two energy groups in homogeneous and heterogeneous problems. Since the benchmark considers critical systems, the accuracy of the RM is evaluated based on the criticality eigenvalue k_{eff} (effective multiplication factor) and, in some cases, on the neutron flux distribution at some positions with reference values communicated from the benchmark. Given the critical dimensions and material composition of the problem, the deviation of the criticality eigenvalue from unity (in pcm) is considered. Benchmark cases with linearly anisotropic scattering are also studied. This comprehensive study of the RM against reference solutions is an essential and valuable step for the validation and verification of the

RM.

The remainder of the article is organized as follows. In section 2, the mathematical and physical formulation of the RM is given. Section 3 describes the numerical implementation of the RM. In section 4, the results of 18 problems are presented and discussed. Lastly, conclusions are given in section 5.

2. Theoretical background of the Ronen method

In slab geometry with linearly-anisotropic scattering and in the absence of entering particles at the boundaries, the integral expression for the total current is given by [4].

$$J(x, E) = \frac{1}{2} \int_0^a dx' E_2[\tau(x', x, E)] \text{sgn}(x - x') q_0(x', E) + \frac{3}{2} \int_0^a dx' E_3[\tau(x', x, E)] q_1(x', E), \quad (3)$$

where E_2 and E_3 are the second- and third-order exponential integral functions [9], respectively, a is the slab width, $\tau(x', x, E)$ is the optical length

$$\tau(x', x, E) \equiv \text{sgn}(x - x') \int_{x'}^x \sigma(x'', E) dx'', \quad (4)$$

and $\sigma(x, E)$ is the total macroscopic cross-section. The source terms in Eq. (3) represent the isotropic and linearly-anisotropic neutron sources [4],

$$q_0(x, E) = \int_0^\infty dE' \left[\sigma_{s,0}(x, E \leftarrow E') + \frac{\chi(E)}{k_{eff}} \nu \sigma_f(x, E') \right] \phi(x, E'), \quad (5a)$$

$$q_1(x, E) = \int_0^\infty dE' \sigma_{s,1}(x, E \leftarrow E') J(x, E'), \quad (5b)$$

where $\sigma_{s,0}$ and $\sigma_{s,1}$ are the isotropic and linearly-anisotropic scattering cross-sections, $\chi(E)$ is the fission spectrum, k_{eff} is the effective multiplication factor, ν is the average number of neutrons emitted per fission, σ_f is the fission cross-section, and ϕ and J are the first two moments of the angular flux, identified by definition as the scalar flux and the net current, respectively. The presence of the net current at both sides of Eq. (3) is resolved through the RM iterations. Note that this expression does not include contributions of other fixed neutron sources in the slab [4].

Eqs. 2,3,5a and 5b, reveal the fundamental postulate of the Ronen method, extended to problems with linearly-anisotropic scattering. While the expression used to calculate the total current is an accurate expression derived from the integral transport theory (Eq. (3)), the neutron sources used for this (transport) expression are calculated using the scalar flux obtained by diffusion theory and an integral estimation of the net current. This hybrid transport-diffusion approach is iterated (Eq. (2)) to improve the diffusion-based solutions.

3. Numerical implementation of the Ronen method

3.1. Discretization of the current by its integral form

The problems considered in this article assume one-dimensional slab geometries, isotropic and linearly-anisotropic scattering, and homogeneous and heterogeneous media. Hence,

the spatial discretization throughout the article is done according to the notation shown in Fig. 1. The space is divided into I cells, where $i = 0$ and $i = I - 1$ are the leftmost and rightmost cells, respectively. Cell i is centered around x_i , its width is Δx_i , and its left and right interfaces are located at $x_i - \Delta x_i/2$ and $x_i + \Delta x_i/2$, respectively. For brevity and clarity, the left and right interfaces of cell i are denoted by $x_{i\pm 1/2}$.

The detailed derivation of a discretized form of the integral expression for the neutron current (Eq. 3) in one-dimensional multigroup notation was reported in a previous work [4], and is

$$J_g(x_{i+1/2}) = \frac{1}{2} \sum_{j=0}^{I-1} \frac{q_{0,g,j}}{\sigma_{g,j}} \left\{ E_3 \left[\tau_g(x_{j+1/2}, x_{i+1/2}) \right] - E_3 \left[\tau_g(x_{j-1/2}, x_{i+1/2}) \right] \right\} \text{sgn}(i-j) + \frac{3}{2} \sum_{j=0}^{I-1} \frac{q_{1,g,j}}{\sigma_{g,j}} \left\{ E_4 \left[\tau_g(x_{j+1/2}, x_{i+1/2}) \right] - E_4 \left[\tau_g(x_{j-1/2}, x_{i+1/2}) \right] \right\}, \quad (6)$$

where $\sigma_{g,j}$ and $q_{0/1,g,j}$ are the cell-averaged total macroscopic cross-section and isotropic and linearly-anisotropic sources of group g in cell j , respectively. The optical length is given by

$$\tau_g(x_i, x_j) = \text{sgn}(i-j) \sum_{p=j}^i \sigma_{g,p} \Delta x_p. \quad (7)$$

For the linearly-anisotropic scattering, the second term of Eq. 6 is present, and the source $q_{1,g,j}$ (Eq. 5b) is computed with a given net current. A proper estimation of this net current is provided for computing the linearly-anisotropic neutron source term. In the latter case, the net current at the first iteration is estimated isotropically; afterward, for successive iterations, the net current is given by the prior value of the integral expression for the current, this time with the linearly-anisotropic source included (see the next section 3.2).

3.2. Numerical schemes

Three different implementations of RM are studied. The first scheme, denoted as RM^D , implements the RM according to Ronen's original suggestion (Eq. 2) and is based on the redefinition of the diffusion coefficient itself through successive iterations. The second implementation, denoted as RM^{net} , implements the RM by adding a correction drift term to the net neutron current at the cell's boundaries. This implementation was previously studied [2,4], so it will be discussed briefly and used for comparison with other implementations. The third scheme, denoted as RM^p , implements the RM by adding corrections as drift terms to the partial neutron currents at the cell's boundaries. It is essentially similar to the RM^{net} implementation, except that the correction term is arbitrarily halved on the two partial currents at each interface. Hence, unlike RM^{net} , which applies a single degree of freedom per interface per energy group, RM^p applies two degrees of freedom per interface per energy group, one for each partial current. RM^p follows the implementation of the pCMFD scheme, which was

introduced to stabilize the CMFD-acceleration in advanced transport solvers [10].

The results for the isotropic cases are compared to a discrete ordinate S_N solver with $N = 128$ discretized angles using Gauss-Legendre quadrature [9], a collision probability method (CPM) transport solver, and a standard diffusion solver. Two neutron transport solvers are used for two main reasons. First, as part of the validation and verification process of the codes, they are compared to existing benchmarks but also to each other. Second, the S_N method is based on the *differential* form of the neutron transport equation, and can take into account any order or scattering anisotropy, reproducing correctly reflection in the halved slab using an appropriate number of directions.

The CPM, on the other hand, is based on the *integral* form of the neutron transport equation and the collision probabilities are calculated only for neutrons emitted isotropically in space [11]. The collision probabilities are computed from the same escape probabilities used to compute the net current in the RM (see Tomatis et al. [3]). These solutions are finally validated with the results of the S_N solver. Therefore, only the S_N solver can be used to validate RM solutions in the presence of linearly-anisotropic scattering.

3.2.1. Standard multigroup diffusion - D_0

The RM is based on obtaining improved solutions for the neutron flux and criticality eigenvalue using a standard multigroup diffusion solver. Hence, the RM iteratively inverts the multigroup diffusion operator with converging correction factors, as detailed in the following sections.

In this article, the implementation scheme for a standard (i.e., no corrections) multigroup diffusion solver is denoted by D_0 . The one-dimensional multigroup neutron balance equation describing the volume-averaged flux in cell i is

$$J_{g,i+1/2} - J_{g,i-1/2} + \sigma_{g,i} \phi_{g,i} \Delta_i = q_{g,i} \Delta_i, \quad (8)$$

where

$$q_{g,i} = \frac{\chi_g}{k_{eff}} \sum_{g'=1}^G \nu \sigma_{f,g',i} \phi_{g',i} + \sum_{g'=1}^G \sigma_{s,g \leftarrow g',i} \phi_{g',i}. \quad (9)$$

Substituting the standard finite difference approximation for Fick's law,

$$J_{g,i+1/2} \cong J_{g,i+1/2}^D \equiv -2D_{g,i+1/2} \frac{\phi_{g,i+1} - \phi_{g,i}}{\Delta_{i+1} + \Delta_i} = -\tilde{D}_{g,i+1/2} (\phi_{g,i+1} - \phi_{g,i}), \quad (10)$$

with $\tilde{D}_{g,i+1/2} \equiv 2D_{g,i+1/2} / (\Delta_{i+1} + \Delta_i)$, yields the (finite) difference equations which constitute the multigroup diffusion operator,

$$-\tilde{D}_{g,i-1/2} \phi_{g,i-1} + \left(\tilde{D}_{g,i+1/2} + \tilde{D}_{g,i-1/2} + \sigma_{t,g,i} \Delta_i \right) \phi_{g,i} - \tilde{D}_{g,i+1/2} \phi_{g,i+1} = q_{g,i} \Delta_i. \quad (11)$$

The P_1 approximation leads to the classic definition of the diffusion coefficient, which is based on the transport cross section $\sigma_{tr,g,i} = \sigma_{t,g,i} - \sigma_{s,1,g,i}$, resulting in $D_{g,i} = 1 / (3\sigma_{tr,g,i})$ [12]. Moreover, the diffusion coefficients are always needed on the cell interfaces, and

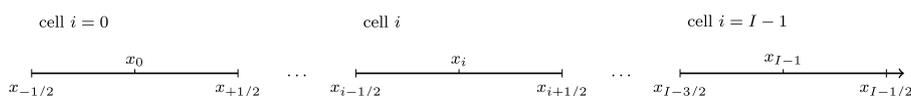


Fig. 1. Notation of the one-dimensional spatial mesh.

first order approximations of Fick currents at both sides of the interface allow to obtain the following expression:

$$D_{g,i+1/2} = \frac{\Delta_i + \Delta_{i+1}}{\Delta_i/D_{g,i} + \Delta_{i+1}/D_{g,i+1}} \quad (12)$$

After the diffusion constants are known, this arrangement forms a set of $G \times I$ linear equations in a three-diagonal matrix at the left hand side. Void boundary conditions were used for the diffusion solver. For the left boundary, e.g.,

$$J_{D,g,-1/2} = -D_{g,-1/2} \frac{\phi_{g,0}}{\Delta_0/2 + \zeta_g} \quad (13)$$

where ζ_g is the extrapolation length of group g . The Marshak boundary condition is obtained if $\zeta_g = 2D_{g,-1/2}$, but a value of $\zeta_g = 2.13D_{g,-1/2}$ is recommended from the literature [13]. The eigenvalue problem is iteratively solved by power iterations to obtain the neutron flux $\phi_{g,i}$ and the multiplication factor (the k_{eff} -eigenvalue). Using the standard notation for the migration operator (\mathcal{M}) and the fission operator (\mathcal{F}), the iterative inversion of the diffusion eigenvalue problem (the power iteration) can be noted as

$$\phi^{(l+1)} = \frac{1}{k_{\text{eff}}^{(l)}} \mathcal{M}^{-1} \mathcal{F} \phi^{(l)}, \quad k_{\text{eff}}^{(l+1)} = k_{\text{eff}}^{(l)} \frac{\langle \phi^{(l+1)}, \mathcal{F} \phi^{(l+1)} \rangle}{\langle \phi^{(l+1)}, \mathcal{F} \phi^{(l)} \rangle} \quad (14)$$

where l is the power iteration index and $\langle \cdot, \cdot \rangle$ stands for the inner product in space and energy. Eq. 14 represents the standard solver of the neutron diffusion problem [14].

3.2.2. Diffusion coefficient correction - RM^D

The computational scheme that implements the original Ronen suggestion is based on iterative redefinition of the diffusion coefficient itself according to Fick's law (Eq. 2). The corrections to the initial diffusion coefficient are local in space and energy in any implementation tested so far and are calculated iteratively according to Eq. 2. This implementation is denoted by RM^D .

The resulting (corrected) diffusion coefficient shows pronounced spatial and spectral dependences. Near sharp interfaces, strong absorbers, and outer boundaries, the diffusion coefficient shows significant local changes. Far from these interfaces, the diffusion coefficient hardly changes since the diffusion approximation is valid, provided that the ratio Σ_a/Σ_s is low. In that sense, the change in the diffusion coefficient (near sharp interfaces) compared to its original value is a measure to the diminishing validity of the diffusion approximation in that local region.

In the diffusion coefficient correction scheme, the RM is implemented as follows. Given a geometry and material composition (i.e., macroscopic cross sections), a first guess for the multigroup flux and k_{eff} -eigenvalue is calculated using standard multigroup diffusion (Eq. 14). Regardless of the initial definition of the diffusion coefficient, the RM will subsequently redefine it. Then, the source term to use in the integral transport equation, Eq. 6, is calculated with the flux obtained by diffusion and the current obtained by the same Eq. 6 at the previous iteration, in order to obtain the next estimate of the current.

Once a new estimation for the current is obtained, a corrected diffusion coefficient is calculated using Ronen's original iterative equation (Eq. 2). Flux gradients are approximated by central finite differences using the cell-averaged fluxes assumed as placed at the cell center. The migration operator \mathcal{M} is updated with the new diffusion coefficient and a new solution for the flux and k_{eff} -eigenvalue is calculated using standard multigroup diffusion.

The operator notation for the RM iterative scheme is

$$\phi^{(r+1)} = \frac{1}{k_{\text{eff}}^{(r)}} \mathcal{M}^{(r)-1} \mathcal{F} \phi^{(r)}, \quad (15)$$

where r is the RM iteration index and the migration operator depends on the corrected diffusion coefficient, $\mathcal{M}^{(r)} \equiv \mathcal{M}[D^{(r)}]$. A flow chart of the RM implementation scheme is shown in Fig. 2.

An obvious difficulty of this implementation is handling cases in which the denominator of Eq. 2 tends to zero. In this benchmark, vanishing gradients can occur where the scalar flux shows maximum or minimum values, usually at the center of symmetric problems or in heterogeneous problems alternating diffusive and multiplying media, as it happens for the thermal flux in a slab with a fuel-moderator interface. At these points, the code behavior may become unpredictable and numerical instabilities may arise.

In the numerical scheme used for this implementation, the diffusion coefficient is corrected where the neutron current is calculated, i.e., at the cells' interfaces. A threshold (10^{-6}) on small-enough current prevents division by zero with vanishing flux gradients, hence the recalculation of the diffusion coefficient is avoided after detection of negligible currents. In symmetric cases, possible division by zero can happen when the mesh is composed of an even number of cells. An odd number of cells in the mesh is usually enough to avoid this problem. A more robust scheme is currently under development without bypassing the recalculation of the coefficient. Notwithstanding this fixup, the iterative scheme showed instabilities caused by small negative values of the diffusion coefficient arising near the vacuum boundary during the iterations. This led to non-convergence on the flux, or even negative flux, thus invalidating the solution. Further investigations are however needed as future research to ensure numerical robustness.

3.2.3. Net current correction - RM^{net}

The computational scheme that implements the net current correction is based on iterative corrections of the net currents at the cells' interfaces. The correction terms are phenomenological degrees of freedom (one per interface per energy group) that are added to the finite difference neutron diffusion equation (Eq. 11).

This implementation, denoted as RM^{net} , was previously studied and is the first implementation of the RM [2,4]. Similarly to the corrected diffusion coefficient (see section 3.2.2), the resulting correction factors exhibit strong spatial and spectral dependence near sharp interfaces, and vanish where the diffusion approximation is valid.

The new correction terms are introduced for the surface currents at cell interfaces $x_{i+1/2}$, and are denoted by δj ,

$$\delta J_{g,i+1/2} = J_{g,i+1/2} - J_{g,i+1/2}^D \quad (16)$$

where $J_{g,i+1/2}$ is the net current obtained from the integral transport expression for the surface current (Eq. 6), and $J_{g,i+1/2}^D$ is the current obtained from the diffusion approximation by Fick's law (Eq. 10). The net current correction term is defined as [5].

$$\delta J_{g,i+1/2} \equiv -\delta D_{g,i+1/2} (\phi_{g,i+1} + \phi_{g,i}) \quad (17)$$

The term $\delta D_{g,i+1/2}$ has an arbitrary definition, and in [2,4] it was divided by the halved sum of the cell widths, here omitted to have fewer floating point operations in the implementation. In any case, the sum of the fluxes stands for some average flux at the interface to physically represent a drift.

Substituting the net current as defined in Eq. 16 into the one-dimensional multigroup neutron balance equation (Eq. 8) yields

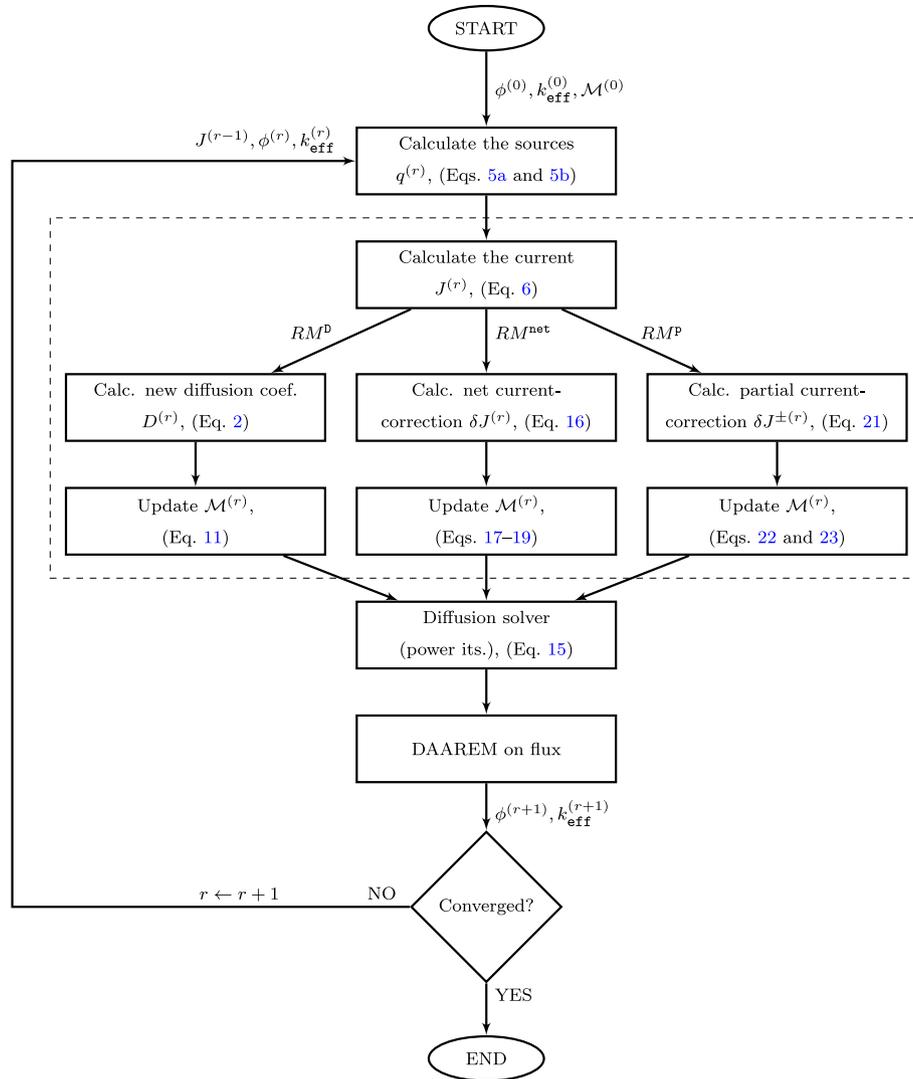


Fig. 2. A flow chart of the Ronen method iterations as implemented using three schemes, RM^D , RM^{net} , and RM^P .

$$\left(J_{g,1+1/2}^D + \delta J_{g,i+1/2} \right) - \left(J_{g,1-1/2}^D + \delta J_{g,i-1/2} \right) + \sigma_{g,i} \phi_{g,i} = q_{g,i}. \tag{18}$$

Substituting Eqs. 10 and 17 into Eq. 18 and rearranging terms gives the discretized form of the *net current-corrected* one-dimensional multigroup neutron diffusion equation,

$$\begin{aligned} & \left(-\tilde{D}_{g,i-1/2} + \delta D_{g,i-1/2} \right) \phi_{g,i-1} \\ & + \left(\tilde{D}_{g,i+1/2} - \delta D_{g,i+1/2} + \tilde{D}_{g,i-1/2} + \delta D_{g,i-1/2} + \sigma_{g,i} \Delta_i \right) \phi_{g,i} \\ & + \left(-\tilde{D}_{g,i+1/2} - \delta D_{g,i+1/2} \right) \phi_{g,i+1} = q_{g,i} \Delta_i. \end{aligned} \tag{19}$$

Note that in case the correction terms vanish, the standard multigroup neutron diffusion operator is recovered (Eq. 11).

In the net current correction scheme, the diffusion coefficient remains unchanged. Instead, at each RM iteration, once a new estimation for the current is obtained, the correction terms δJ are calculated using Eq. 16. The correction factors δD_g on the left boundary take the form of $\delta D_{g,-1/2} = -\delta J_{g,-1/2} / \phi_{g,0}$ and an equivalent form is used for the right boundary. The migration operator $\mathcal{M}^{(r)}$ is reconstructed with the new correction terms (Eqs. 17–19) and a new solution for the flux and k_{eff} -eigenvalue is calculated (Eq. 15). This process is repeated until convergence is achieved for the flux and eigenvalue. A flow chart of the net current correction implementation scheme is shown Fig. 2.

3.2.4. Partial currents correction - RM^P

The computational scheme that implements the *partial currents* correction takes advantage of the fact that partial currents calculation at the cells' interfaces is possible without additional computational overhead. The correction terms in this scheme are two degrees of freedom (per interface per energy group) that are added to the finite difference neutron diffusion equation (Eq. 11).

The development of this implementation, denoted as RM^P , was

mainly motivated by its reported enhanced numerical stability and convergence characteristics in CMFD-accelerated transport [10,15,16].

The new correction terms are introduced for the surface currents at cell interfaces according to

$$J_{i+1/2} = J_{i+1/2}^+ - J_{i+1/2}^- = J_{i+1/2}^D + \delta J_{i+1/2}^+ + \delta J_{i+1/2}^- \quad (20)$$

An upwind definition with respect to the direction of flight of the neutrons is used for the corrections [15,16],

$$\delta J_{i+1/2}^+ = -\delta D_{i+1/2}^+ \phi_i \quad \text{and} \quad \delta J_{i+1/2}^- = -\delta D_{i+1/2}^- \phi_{i+1} \quad (21)$$

Note that the signs before the correction terms are arbitrary, with no need to follow a specific physical meaning. The current calculated by diffusion is equally distributed over the partial currents to compute the corrective coefficients δD^\pm ,

$$J_{i+1/2}^+ - \frac{1}{2} J_{i+1/2}^D = -\delta D_{i+1/2}^+ \phi_i, \quad (22a)$$

$$J_{i+1/2}^- + \frac{1}{2} J_{i+1/2}^D = +\delta D_{i+1/2}^- \phi_{i+1}. \quad (22b)$$

Hence, once the partial currents and the diffusion current are known, the correction terms can be evaluated. One can recover RM^{net} if the correction factors are set equal at all interfaces $\delta D^+ = \delta D^-$.

The substitution of Eqs. 20–22 into the neutron balance equation (Eq. 8) yields the discretized form of the *partial current-corrected* one-dimensional multigroup neutron diffusion equation,

$$\begin{aligned} & \left(-\tilde{D}_{g,i-1/2} + \delta D_{g,i-1/2}^+ \right) \phi_{g,i-1} - \left(\tilde{D}_{g,i+1/2} + \delta D_{g,i+1/2}^- \right) \phi_{g,i+1} \\ & + \left(\tilde{D}_{g,i+1/2} - \delta D_{g,i+1/2}^+ + \tilde{D}_{g,i-1/2} + \delta D_{g,i-1/2}^- + \sigma_{g,i} \Delta_i \right) \phi_{g,i} = q_{g,i} \Delta_i. \end{aligned} \quad (23)$$

Note that also in this scheme, if the correction terms vanish, the standard multigroup neutron diffusion operator is recovered (Eq. 11). Hence, in a weakly absorbing medium and several mean free paths away from the boundary or a strong absorber, all correction factors δJ^\pm are expected to vanish and the net current should be reproduced by the diffusion current. The estimation of the correction factors on the boundaries is similar to the RM^{net} implementation, except that incoming correction currents δJ^\pm are eliminated for each vacuum boundary. A flow chart of the partial currents correction implementation scheme is shown Fig. 2.

4. Results

4.1. Benchmark description

The analytic benchmark of Sood et al. (2003) [8] presents one-dimensional homogeneous and heterogeneous critical configurations, with the geometrical properties, shown in Table 1, including the associated one- and two-group cross sections, given here in Appendix A (Tables A1–A4). The nomenclature of the problem identifiers follows Sood’s notations, specifically, the first digit after the material identifier denotes the energy group, and the second is the scattering order. The fast and thermal energy groups are

denoted as 1 and 2, respectively. The scattering order is identified as 0 for isotropic and 1 for linearly-anisotropic scattering. For some problems, the spatial distribution of the flux is given at specific points (Table A5–A6). The test suite contains cases with scattering anisotropy (linearly and quadratically). Sood’s benchmark contains also spherical and cylindrical geometries, which are beyond the scope of his paper.

A total of 18 problems are solved in this study. The problems can be categorized into four different slab configurations shown in Fig. 3. These include symmetric bare and reflected slab (Fig. 3a and b), asymmetric reflected slab (Fig. 3c), and asymmetric three-media slab (Fig. 3d). The critical dimensions of the fuel and water reflector, marked as L_c and L_{H_2O} , respectively, are given for each problem in Table 1.

The L_i points mark the locations where a reference solution for the normalized flux is provided for some problems (see Tables A5 and A6). The fluxes are normalized by the value at the center of the slab, except for problem #30, where the flux is normalized by the flux at the left boundary, and the L_i correspond to the interfaces between different materials. For two-energy groups, the scalar fluxes are normalized with the fast group flux at the center. The two-energy groups cross sections given in Table A3 are similar to ^{239}Pu (Prob. #45), ^{235}U (Prob. #48), a realistic enriched uranium-aluminum-water assembly (prob. 51), and a 93% enriched ^{235}U bare slab model of a university research reactor (Prob. #54) and water reflected (Probs. #58–#59).

4.2. The local behavior of the RM correction terms

Three test cases are considered to demonstrate the spatial behavior of the correction terms in the different implementations in presence of vacuum boundary. The selected test cases are one-dimensional one-group problems, of which two are homogeneous, and one is heterogeneous. Problem #22 is a large uranium-heavy water mixture slab with a width of ~ 10 mean free paths (mfp). Problem #2 is a small plutonium slab with ~ 1 mfp width. Problem #4 is a small heterogeneous problem of water-reflected plutonium slab, with a typical width of ~ 0.5 mfp.

Fig. 4 shows the RM’s neutron flux, the flux deviation from the reference flux and the spatial distribution of the correction factors for the selected problems. The flux is shown for the right part of the plane due to symmetry, and the reference solutions are calculated using the collision probability code (CPM). The spatial discretization for all problems is $\Delta \approx 0.01$ mfp (equidistant mesh), and the flux is normalized by the total reaction rate.

The standard diffusion underestimates the flux at the slab center and overestimates it near the boundary. This is less pronounced in the larger and more diffusive problem (#22) but very noticeable at the smaller less diffusive problems #2 and #4. The flux obtained by the three implementations of the RM matches well the reference solutions, with a maximal deviation of $\sim 1\%$.

The correction factors of the RM exhibit interesting spatial behavior. For the large slab (problem #22), the corrected diffusion coefficient $D^{NEW}(x)$ (of the RM^D implementation) is close to the original diffusion coefficient D_0 away from the boundaries but changes rapidly towards the boundary, forcing a steep gradient on the flux. For the smaller problems (problem #2 and #4), the corrected $D^{NEW}(x)$ is smaller compared to D_0 at the slab center and decreases even further towards the interfaces or the boundary.

The correction factor $\delta D^{NET}(x)$ (of the RM^{net} implementation)

Table 1
Critical dimensions of each problem [8].

Config.	Prob. # / ID	L_c [cm/mfp]	Comments			
3a	2 / PUa-1-0-SL	1.853722 / 0.605055	Isotropic			
	6 / PUB-1-0-SL	2.256751 / 0.73660355				
	12 / Ua-1-0-SL	2.872934 / 0.93772556				
	22 / UD2O-1-0-SL	10.371065 / 5.6655054562				
	45 / PU-2-0-SL	1.795602 / 0.396469				
	48 / U-2-0-SL	3.006375 / 0.649377				
	51 / UAL-2-0-SL	7.830776 / 2.099940				
	54 / URRa-2-0-SL	7.566853 / 4.971120				
	32 / PUa-1-1-SL	0.77032 / 0.77032		Linearly anisotropic		
	34 / PUB-1-1-SL	0.79606 / 0.79606				
71 / URRa-2-1-SL	9.49590 / 6.23840					
Config.	Prob. # / ID	L_c [cm/mfp]	L_{H_2O} [cm/mfp]			
3c	3 / PUa-H2O(1)-1-0-SL	1.478450 / 0.482566	3.063725 / 1.0			
3b	4 / PUa-H2O(0.5)-1-0-SL	1.317862 / 0.43015	1.531863 / 0.5			
	25 / UD2O-H2O(1)-1-0-SL	9.214139 / 5.03350	1.830563 / 1.0			
	26 / UD2O-H2O(10)-1-0-SL	8.428096 / 4.60410	18.30563 / 10.0			
	58 / URRb-H2Oa(1)-2-0-SL	6.696802 / 5.94147	1.126152 / 1.0			
	59 / URRb-H2Oa(5)-2-0-SL	4.863392 / 4.31485	5.630757 / 5.0			
Config.	Prob. # / ID	Units	Fe (left)	U-235	Fe (right)	Na
3d	30 / Ue-Fe-Na-1-0-SL	cm	0.317337461	5.119720083	0.317337461	2.002771002
		mfp	0.0738	2.0858098	0.0738	0.173

effectively vanishes where diffusion is valid, i.e., away from interfaces or boundaries. However, it changes rapidly in the vicinity of interfaces and boundaries.

The tandem correction factors, $\delta D^\pm(x)$ (of the RM^P implementation), exhibit similar qualitative behavior as the $\delta D^{NET}(x)$ correction factor. However, they do not vanish where diffusion is valid. Rather, they asymptotically approach ± 0.25 , which is explained as follows. Consider a cell interface $i + 1/2$ and the partial current on that interface $J_{i+1/2}^+$. The first-order diffusion approximation for the current is

$$J_{i+1/2}^+ \approx \frac{\phi_{i+1/2}}{4} + \frac{J_{i+1/2}}{2}. \tag{24}$$

Subtracting Eq. 22a from Eq. 24 yields

$$\frac{\phi_{i+1/2}}{4} + \frac{J_{i+1/2}}{2} - \left[\frac{J_{i+1/2}^D}{2} - \delta D_{i+1/2}^+ \phi_i \right] \approx 0. \tag{25}$$

Assuming that the diffusion approximation is valid, i.e., $J_{i+1/2}^D \approx J_{i+1/2}$ and that $\phi_{i+1/2} \approx \phi_i$ results in $\delta D_{i+1/2}^+ \approx -1/4$. The

same process for J^- using Eq. 22b yields $\delta D_{i+1/2}^- \approx 1/4$. Under the same assumptions, the correction factors of RM^{net} will vanish, i.e., $\delta D \approx 0$, as shown in Fig. 4.

The physics that governs the RM corrections follows the principle that the larger the diffusion coefficient, the smoother the flux is. The value of the diffusion coefficient represents the strength of the spatial coupling between the flux in nearby cells (physically realized by neutron leakage). A zero diffusion coefficient results in complete spatial decoupling of the flux, i.e., the flux in each cell does not depend on adjacent cells. The RM takes advantage of this feature and spatially modifies the diffusion coefficient. By decreasing the value of the diffusion coefficient near interfaces and boundaries, the effective diminished diffusion coefficient (in all implementations) enables steeper gradients of the flux. Since the corrections are derived using the integral transport expression for the neutron current, the diffusion solution is driven towards the transport one, thus providing an improved diffusion solution.

Table 2 shows the flux deviations at 4 locations for a simple diffusion solver and the three implementations of the RM. Cases #32 and #34 are small (~1.5 mfp) homogenous one-group linearly-anisotropic problems. Results were compared to S_{128} with spatial discretization of $\Delta \approx 0.01$ mfp. Maximal flux deviations were

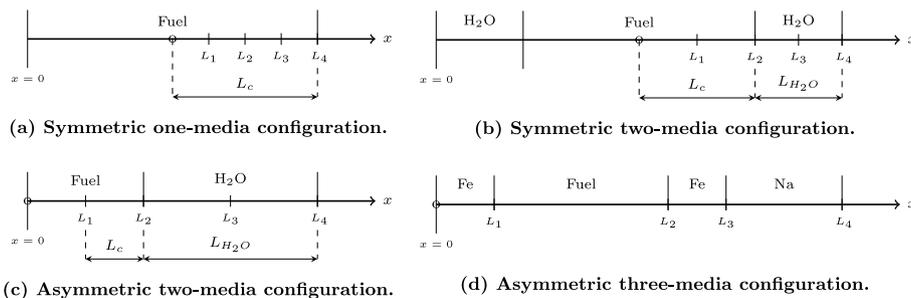
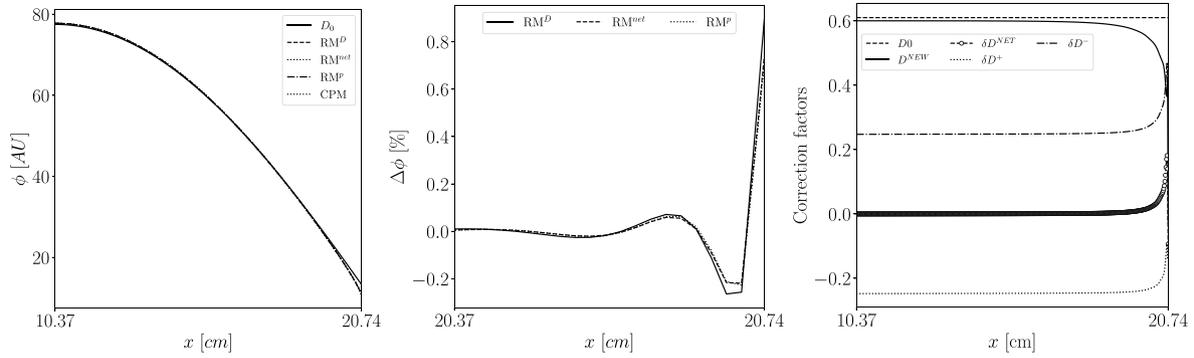
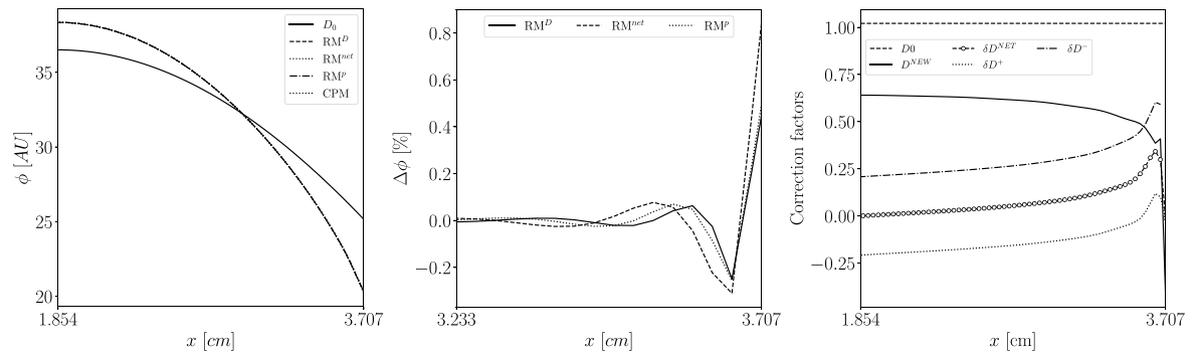


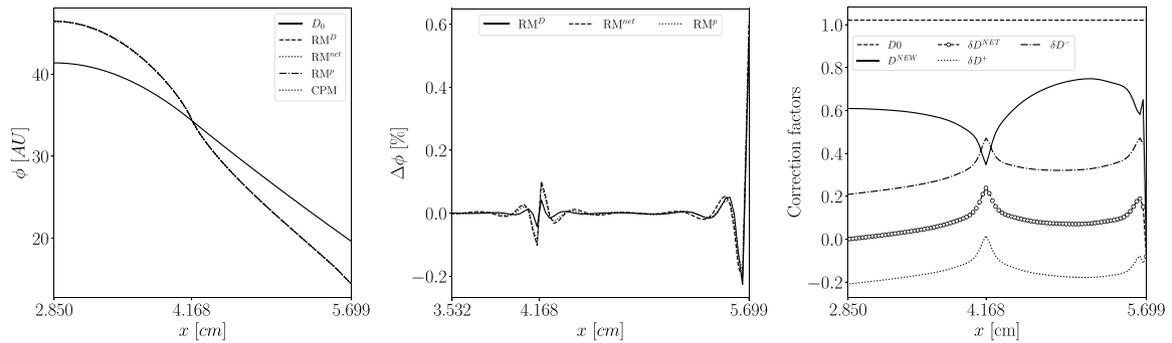
Fig. 3. Four different geometry configurations of the problems solved in this study.



(a) Problem #22 UD2O-1-0-SL.



(b) Problem #2 PUa-1-0-SL.



(c) Problem #4 PUa-H2O(0.5)-1-0-SL.

Fig. 4. Results of three one-group symmetrical problems, obtained by standard diffusion, RM^D , RM^{net} , RM^P , and a reference CPM codes, with spatial discretization of $\Delta \approx 0.01$ mfp.

Table 2

Flux deviation in [%] for problems #32 and #34 compared to S_{128} .

L_i	PUa-1-1-SL (#32)				PUB-1-1-SL (#34)			
	D_0	RM^D	RM^{net}	RM^P	D_0	RM^D	RM^{net}	RM^P
$L_1 = 0.25$	-0.669	0.003	0.001	0.000	-0.695	0.000	0.002	0.000
$L_2 = 0.5$	-3.035	0.005	0.002	0.000	-3.145	0.000	-0.001	-0.001
$L_3 = 0.75$	-8.854	0.008	0.001	-0.007	-9.133	0.002	-0.006	-0.003
$L_4 = 1.0$	-32.231	-1.127	-1.353	-1.620	-32.991	-0.916	-1.570	-1.543

observed in the proximity of vacuum boundary (L_4), where simple diffusion gives the largest deviation of around ~32%, while the three implementations of the RM range between 0.9 – 1.6%.

4.3. Accuracy of the criticality eigenvalue

Table 3 presents the differences in reactivity calculated with the three implementation of the Ronen method (RM^D , RM^{net} , and RM^P) and standard diffusion (D_0). For comparison and scale, the results of a discrete ordinates code with 128 angles (S_{128}) and a collision probability method code (CPM) are also shown for the selected set of problems. The deviations are shown in units of reactivity ($\Delta\rho = (k_{ref}^{-1} - k^{-1}) \times 10^5$ [pcm]), where k_{ref} equals unity. In all problems, the cell size is $\Delta \approx 0.01$ mfp.

Looking at Table 3, it is clear that standard diffusion with classic extrapolated distance boundary conditions fails in evaluating the criticality eigenvalue. The smaller deviations appear in problems

Table 3

Deviation of the criticality eigenvalue from unity for the benchmark problems [8], using three RM implementations, standard diffusion (D_0), S_{128} and CPM. RM implementations also show in round brackets the number of required iterations. In all problems, the cell size is $\Delta \approx 0.01$ mfp. Results are in pcm.

Prob. #/ID	D_0	RM^D	RM^{net}	RM^P	S_{128}	CPM
2/PUa-1-0-SL	-12,109	-2 (120)	-3 (90)	-3 (90)	-4	-2
3/PUa-H2O(1)-1-0-SL	-13,488	-5 (130)	-5 (129)	-5 (138)	-5	-4
4/PUa-H2O(0.5)-1-0-SL	-15,926	-2 (130)	-7 (150)	-5 (150)	-2	-2
6/PUB-1-0-SL	-9,440	-2 (98)	-1 (67)	-1 (80)	-2	-1
12/Ua-1-0-SL	-7,553	-2 (110)	-2 (77)	-2 (95)	-2	-1
22/UD2O-1-0-SL	-164	-1 (136)	-1 (130)	-1 (136)	0	0
25/UD2O-H2O(1)-1-0-SL	-260	0 (126)	0 (126)	0 (130)	0	-1
26/UD2O-H2O(10)-1-0-SL	-355	0 (136)	0 (166)	0 (129)	0	0
30/Ue-Fe-Na-1-0-SL	-6,744	-1 (140)	-2 (140)	-1 (138)	-1	-2
45/PU-2-0-SL	-14,664	-4 (126)	-4 (139)	-4 (140)	-5	-1
48/U-2-0-SL	-8,414	-3 (150)	-3 (140)	-3 (150)	-3	-1
51/UAL-2-0-SL	-3,759	-7 (140)	-7 (138)	-8 (117)	0	-2
54/URRa-2-0-SL	-644	-3 (210)	-3 (165)	-3 (156)	0	0
58/URRb-H2Oa(1)-2-0-SL	-278	-3 (135)	-3 (167)	-3 (175)	0	-2
59/URRb-H2Oa(5)-2-0-SL	-305	-	-2 (219)	-2 (134)	1	-1
32/PUa-1-1-SL	-10,157	1 (165)	1 (190)	1 (120)	0	-
34/PUB-1-1-SL	-10,230	-2 (184)	-2 (230)	-2 (262)	-2	-
71/URRa-2-1-SL	-18,167	-3 (227)	-1 (256)	-3 (229)	0	-

Table 4

Criticality eigenvalue accuracy of D_0 , RM^{net} , and S_{128} with various spatial discretizations for three representative benchmark problems [8]. Results are in pcm.

Δ [mfp]	PUa-1-0-SL (#2)			UD2O-H2O(1)-1-0-SL (#25)			PUa-1-1-SL (#32)		
	D_0	RM^{net}	S_{128}	D_0	RM^{net}	S_{128}	D_0	RM^{net}	S_{128}
0.1	-11,942	-147	-109	-258	-25	-1	-10,036	-128	-77
0.05	-12,065	-41	-31	-250	-6	0	-10,130	-29	-17
0.01	-12,109	-3	-4	-260	0	0	-10,157	1	0
0.005	-12,110	-1	-2	-260	0	0	-10,158	2	0

which are larger (≥ 5 mfp) and more “diffusive”, e.g., problems #22 and #54, where the uranium fuel is mixed with heavy water, or heavily reflected and moderated problems, e.g., #25, #26, #58, and #59.

The deviations of the RM implementations are less than 10 pcm for all problem, close to the transport codes’ results. In addition, the number of Ronen iterations required to fulfill the convergence tolerance (10^{-6}) is shown in Table 3 in round brackets for all the RM implementations; no significant differences in the required number of iterations have been observed among the three implementations. It should be stressed that the RM does not invert a transport operator, it rather uses the standard diffusion solver iteratively with appropriate correction factors. The RM and transport codes underestimate the criticality eigenvalue in all cases, even though by only a few pcm.

The accuracy of the eigenvalue (k_{eff}) in pcm with respect to different values of the spatial discretization is shown in Table 4. D_0 , RM^{net} , and S_{128} are tested for three representative problems, bare slab with isotropic (#2) and linearly-anisotropic (#32) scattering and reflected and moderated slab (#25). Simple diffusion shows

large deviations using finer meshes, while RM^{net} and S_{128} show opposite trends. For the less diffusive problems (#2 and #32), tighter meshes are needed to reach the accuracy of less than 10 pcm; the water-reflected problem (#25) reaches this accuracy with coarser meshes.

4.4. Accuracy of the flux distribution - comparison to the reference solutions

The deviation of the calculated flux from the reference value given by Sood et al. 2003 [8] (Table A5 and A6), with $\Delta\phi = (\phi_{ref} - \phi_{RM})/\phi_{ref} \times 100$ [%], are shown in Tables 5–10 for problems #6, #12, #30, #54, and #71 at positions L_i . The fluxes at L_i are normalized against the flux at a specific location; the normalization positions of problems # 6 and # 12 are at the center of the slab, problems #54 and #71 is two group problem and the fluxes are normalized by the fast flux at the slab’s center, and problem #30 at the left-most boundary (all normalization locations are marked by a blank circle in Fig. 3).

The calculated flux was obtained using standard diffusion (D_0), three implementations of the RM (RM^D , RM^{net} , RM^P), a discrete ordinates code (S_{128}), and a Collision probability method (CPM). The spatial discretization in all the calculations is $\Delta \approx 0.01$ mfp, and void boundary conditions implied. All the codes above use central finite difference discretization scheme for the flux, which is described as cell-centered. Hence, in order to obtain an estimation of the flux at inner interfaces and the boundary, a linear interpolation is performed.

For standard diffusion (denoted by D_0) in slab problems (#6, #12, #22, #54, and #71), the flux deviation increases from the slab’s center towards the outer-most boundary. The deviations range between 0.5-2% close the center and 23–83% at the boundary (L_4).

Table 5

Flux deviation in [%] for problem #6.

L_i	PUB-1-0-SL (#6)					
	D_0	RM^D	RM^{net}	RM^P	S_{128}	CPM
$L_1 = 0.25$	-0.635	0.002	0.004	0.004	0.000	0.000
$L_2 = 0.5$	-2.884	0.000	0.003	0.004	0.000	0.000
$L_3 = 0.75$	-8.457	0.000	-0.004	0.001	0.002	0.000
$L_4 = 1.0$	-31.234	-0.998	-1.333	-1.242	-0.060	-0.239

Table 6
Flux deviation in [%] for problem #12.

Ua-1-0-SL (#12)						
L_i	D_0	RM^D	RM^{net}	RM^P	S_{128}	CPM
$L_1 = 0.25$	-0.525	0.001	0.001	0.000	0.001	0.000
$L_2 = 0.5$	-2.427	0.000	0.000	0.000	0.001	0.000
$L_3 = 0.75$	-7.386	-0.001	0.001	0.003	0.002	0.000
$L_4 = 1.0$	-30.708	-0.940	-1.148	-1.128	-0.173	-0.222

Table 7
Flux deviation in [%] for problem #22.

UD20-1-0-SL (#22)						
L_i	D_0	RM^D	RM^{net}	RM^P	S_{128}	CPM
$L_1 = 0.25$	-0.014	0.000	0.000	0.000	0.000	0.000
$L_2 = 0.5$	-0.082	0.000	0.000	0.000	0.000	0.000
$L_3 = 0.75$	-0.470	0.000	0.000	0.000	0.000	0.000
$L_4 = 1.0$	-24.157	-0.934	-1.065	-0.984	-0.133	-0.173

Table 8
Flux deviation in [%] for problem #30.

Ue-Fe-Na-1-0-SL (#30)						
L_i	D_0	RM^D	RM^{net}	RM^P	S_{128}	CPM
Fe-U	10.054	0.482	0.543	0.550	-0.205	-0.004
U-Fe	16.240	0.552	0.632	0.646	-0.111	0.087
Fe-Na	12.544	0.405	0.474	0.491	-0.261	-0.052
Na	-1.853	-0.027	0.011	0.012	0.000	0.003

Table 9
Flux deviation in [%] for problem #54. All values are normalized with the fast group flux at the center. The upper and lower values are of the fast and thermal groups, respectively.

URRa-2-0-SL (#54)							
L_i	g	D_0	RM^D	RM^{net}	RM^P	S_{128}	CPM
$L_1 = 0.241394$	1	-1.920	0.000	0.000	0.000	0.000	0.000
	2	8.4470	0.002	0.001	0.002	0.000	0.000
$L_2 = 0.502905$	1	-8.050	0.000	0.000	0.000	0.000	0.000
	2	-6.463	0.002	0.001	0.002	0.000	0.000
$L_3 = 0.744300$	1	-16.155	0.000	0.000	0.000	0.000	0.000
	2	-31.506	0.003	0.002	0.002	-0.001	0.000
$L_4 = 1.0$	1	-49.261	-0.917	-1.174	-1.198	-0.142	-0.179
	2	-83.587	-1.303	-1.305	-1.283	-0.795	-0.665

Table 10
Flux deviation in [%] for problem #71. All values are normalized with the fast group flux at the center. The upper and lower values are of the fast and thermal groups, respectively.

URRa-2-1-SL (#71)						
L_i	g	D_0	RM^D	RM^{net}	RM^P	S_{128}
$L_1 = 0.2$	1	-1.182	0.003	0.003	0.003	0.003
	2	9.568	0.005	0.005	0.005	0.003
$L_2 = 0.5$	1	-7.132	0.024	0.024	0.024	0.024
	2	-5.202	0.028	0.028	0.028	0.026
$L_3 = 0.8$	1	-16.463	0.091	0.091	0.091	0.091
	2	-34.388	0.127	0.127	0.127	0.124
$L_4 = 1.0$	1	-48.500	-0.768	-1.343	-1.343	-0.027
	2	-75.369	-1.379	-1.403	-1.372	-0.882

In the case of problem #30, which is an asymmetric problem (moderated on the right side), the deviations at three positions within the slab range between 10–16%, and 1.8% at the right-most boundary (thanks to the wide sodium region).

The results obtained by the three RM implementations give significantly improved results for the slab problems (#6, #12, #22, #54, and #71). In the RM case, the flux deviations inside the slab are in the order of milli-percent and can be considered negligible. The

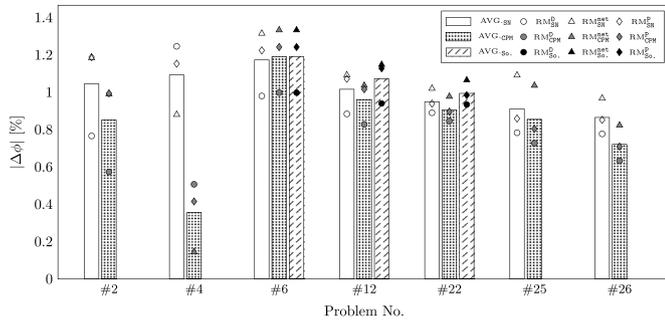


Fig. 5. One-group flux deviation in symmetric problems, compared against S_{128} and CPM codes. Results shown for the right-most boundary (L_4) and cell width of $\Delta \approx 0.01$ mfp.

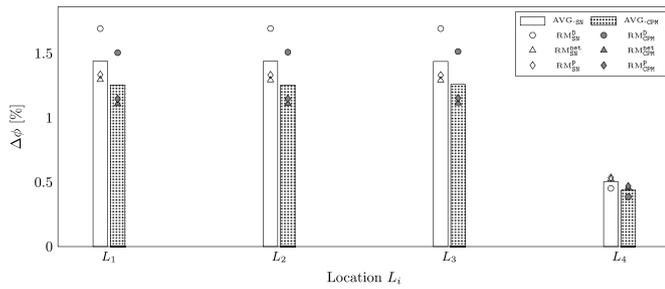


Fig. 6. One-group flux deviation for asymmetric problem #3, compared against S_{128} and CPM codes. Results shown for 4 locations, shown in Fig. 3c. Cell size is $\Delta \approx 0.01$ mfp.

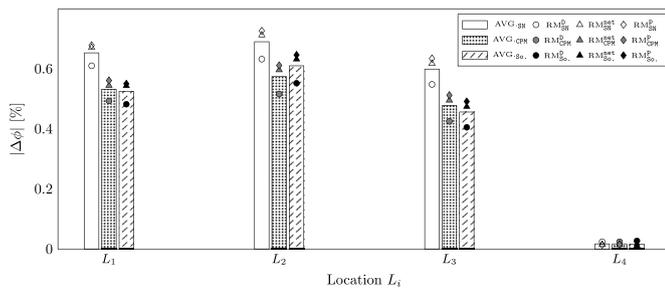


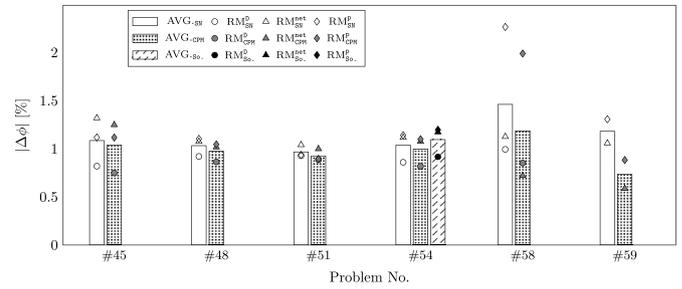
Fig. 7. One-group flux deviation for asymmetric problem #30, compared against S_{128} and CPM codes. Results shown for four locations, shown in Fig. 3d. Cell size is $\Delta \approx 0.01$ mfp.

only significant deviations, of around 1%, are observed on the slab's boundary. The RM also gives more accurate results within the heterogeneous problem #30, with flux deviations of around 0.5% on the inner material interfaces and about 0.1% on the right-most boundary.

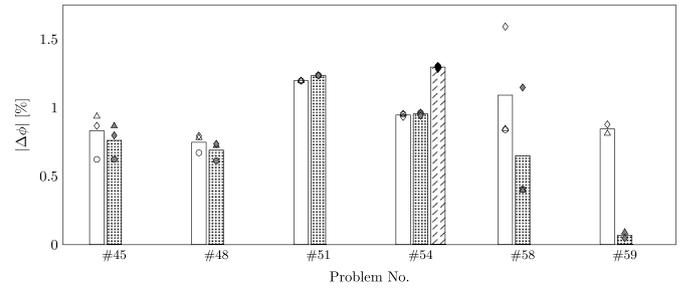
Finally, it is observed that the RM^D implementation yields smaller flux deviations compared to RM^{net} and RM^P for most of the problems. The transport solutions of S_{128} and CPM shows even smaller deviations (of around 0.1%), with the S_{128} resulting in more accurate solutions compared to CPM when the same mesh is used.

4.5. Accuracy of the flux distribution - comparison to the S_N and CPM solutions

Reference values for the flux are not given for all the problems in Table 1. Therefore, a reference flux was calculated at four points for all problems using S_{128} and CPM codes. Finer spatial discretization was used for all problems, keeping $\Delta \approx 0.005$ mfp.



(a) Fast-group flux deviation of various problems.



(b) Thermal-group flux deviation of various problems.

Fig. 8. Two-group flux deviation for various problems, compared against S_{128} and CPM codes. Results shown for the right-most boundary (L_4), and cell average of $\Delta \approx 0.01$ mfp.

Visual representations of the results are shown in Figs. 5–8. The results are categorized into homogeneous and heterogeneous problems and energy groups. Additionally, the results for asymmetric problems contain deviation at four positions (L_1 – L_4). Looking at Figs. 5–8, round, triangle, and diamond markers represent the results of RM^D , RM^{net} , and RM^P , respectively. White and gray filled markers represent the deviation against S_{128} and CPM, respectively. The bars represent the average deviation of the three RM implementations for each reference, where the white filled bar is against S_{128} and the dotted filled bar against CPM. When available, additional comparisons against Sood et al. [8] reference solutions are given, visible with black filled markers and bars filled with diagonal lines.

In symmetrical problems, the error within the slab is negligible, so only those on the boundary (L_4) are considered. Fig. 5 shows the absolute flux deviations for seven symmetrical one-group problems at L_4 . For most of the problems, it can be seen that RM^D (round marker) is more accurate than the other implementation of the RM, and against all references. The flux errors are of the order of $\sim 1\%$. The same trend also appears in the comparison to reference solutions (see Sec. 4.4). The average deviation against CPM is lower for most of the problems (dotted bar), compared to S_{128} (white bar), and Sood et al. (dashed lines bar).

Figs. 6 and 7 show one-group flux deviation for asymmetrical problems #3 and #30. The neutron flux in both problems is normalized using the flux on the left-most boundary. As before, the deviation of RM against S_{128} is generally higher compared to the deviation against CPM. The deviation of RM^{net} and RM^P is the same in all positions for both problems. Within the slab, the deviation of RM^D in problem #3 is higher than the other implementation, unlike the other problems.

Fig. 8 shows the flux deviations for two-group problems with the same marking as for one-group. The average deviations of the RM compared to the S_{128} and CPM are almost the same for the homogeneous problems #45, #48, #51, and #54, with slightly higher deviation against S_{128} . A reference flux for problem # 54 is

Table 11

CPU runtime (wall clock seconds) of RM, S_{128} , and CPM simulating two-group, bare and reflected problems with spatial discretization of $\Delta \approx 0.01$ mfp. Results shown for 10 realizations of each implementation.

Prob. #	RM^D	RM^{net}	RM^P	S_{128}	CPM
54	35.1 ± 0.7	30.1 ± 0.3	28.7 ± 0.5	263.0 ± 3.2	100.8 ± 0.5
58	71.7 ± 1.1	54.5 ± 0.6	44.1 ± 1.4	475.8 ± 2.9	268.9 ± 2.3

provided by Sood et al. (Table A6). The relatively large differences are noticed with the heterogeneous problems #58 and #59, favoring CPM. The RM^D implementation could not produce results for problem #59 due to the small size of Δx resulting in a convergence problem. Larger values of Δx indeed resume convergence.

The differences in flux distribution accuracy between the different RM implementations are relatively small for the homogeneous two-group problems, similar to the trend observed in one-group problems (less than 0.5% difference). The most pronounced difference is shown in problem #58, where the RM^P flux deviation is larger by ~1% compared to RM^D and RM^{net} . The behavior of the flux deviation for the different RM implementations is qualitatively and quantitatively similar for the fast and thermal groups.

Considering asymmetrical problems, the deviation of the RM compared to S_{128} was higher than of against CPM, resulting in closer neutron flux obtained by RM, to the one of CPM. Over all, the RM tends to provide closer results to CPM than S_{128} , due to the fact that both methods use the integral expressions of neutron transport.

4.6. RM convergence performances

4.6.1. Run times

Average CPU run times for RM, S_{128} , and CPM calculations of a bare and reflected slab, are shown in Table 11. A total of ten realizations for each implementation were carried out. These run times were obtained using an intel core-i5 4.1GHz x12 CPU, with 32GB DDR4 memory, and Ubuntu OS. The spatial discretization cell size is $\Delta \approx 0.01$ mfp. Both the relative flux and eigenvalue residuals were below 10^{-6} .

For these problems, the RM is three times faster than CPM (both calculate the scalar flux) and eight times faster than S_{128} . Looking at RM implementations, the RM^D implementation is slightly slower

than RM^{net} and RM^P . Note that the computer programs are written in Python which is an interpreted language. As such, it is difficult to provide accurate runtime estimates of the implemented algorithms. A reprogram in low level languages may change the current outcome. A more thorough investigation into the RM's computational time performances is beyond the scope of this paper and will be published elsewhere.

4.6.2. Convergence rate of the RM iterations

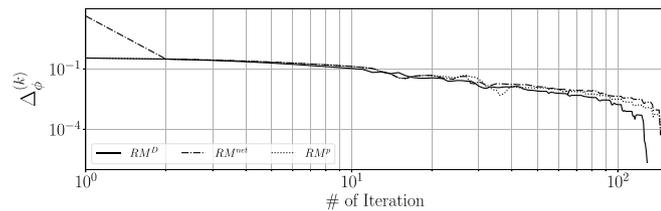
The convergence of the RM can be evaluated in two ways. First, measure the "distance" of the solution in the RM k^{th} iteration from the final one. Second, another aspect of RM convergence can be evaluated by measuring the "distance" between the solutions of two successive RM iterations. Both aspects relate closely to the Anderson Acceleration (AA) through the DAAREM algorithm and its parameters. In this study, the parameters of this acceleration algorithm were taken as the recommended values for the general case and were not optimized or studied [6,7]. Nonetheless, it would be interesting to study the effect of the acceleration algorithm's parameters on the convergence rate of the RM, and we intend to study it in future work.

The convergence rate of the neutron flux ($\Delta_{\phi}^{(k)}$) and criticality eigenvalue ($\Delta_{k_{eff}}^{(k)}$) with respect to the final RM solution are shown in Fig. 9 for the three implementations of the RM for a reflected problem (#4, shown in Fig. 4c). The convergence rate is calculated according to

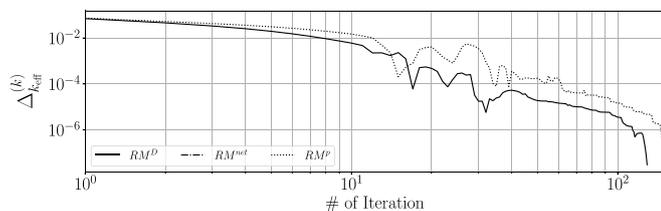
$$\Delta_{\phi}^{(k)} \equiv \max \left| \frac{\phi^{(k)} - \phi^{RM_{final}}}{\phi^{RM_{final}}} \right|, \quad \Delta_{k_{eff}}^{(k)} \equiv \left| \frac{k_{eff}^{(k)} - k_{eff}^{RM_{final}}}{k_{eff}^{RM_{final}}} \right|, \quad (26)$$

where $\phi^{RM_{final}}$ and $k_{eff}^{RM_{final}}$ are the final scalar flux distribution and criticality eigenvalue calculated by the RM.

The convergence rate of the neutron flux, at the very first few iterations, is relatively slow. For the last few tens of iterations, the convergence rate increases gradually with a sharp increase toward the final value of the RM. Small fluctuations have been noticed after the 10th RM iteration typical of Anderson accelerated algorithms applied to fixed-point iterative problem [17,18]. Furthermore, this trend is similar for the three implementations. The behavior of the flux through the RM iterations also affects the convergence rate of



(a) Scalar flux convergence rate.



(b) Criticality eigenvalue convergence rate.

Fig. 9. The convergence rate toward the final RM solution of the scalar flux and the criticality eigenvalue for benchmark problem #4 [8].



(a) Scalar flux convergence rate.



(b) Criticality eigenvalue convergence rate.

Fig. 10. The convergence rate between two successive RM iterations of the scalar flux and the criticality eigenvalue for benchmark problem #4 [8].

the criticality eigenvalue. A similar trend is observed for $\Delta_{k_{\text{eff}}}^{(k)}$ but with larger fluctuations with respect to $\Delta_{\phi}^{(k)}$.

The convergence rate of the neutron flux ($\varepsilon_{\phi}^{(k)}$) and criticality eigenvalue ($\varepsilon_{k_{\text{eff}}}^{(k)}$) with respect to two successive RM iterations are shown in Fig. 10 for the three implementations of the RM for a reflected problem (#4, shown in Fig. 4c). The convergence rate is calculated according to

$$\varepsilon_{\phi}^{(k)} \equiv \max \left| \frac{\phi^{(k)} - \phi^{(k-1)}}{\phi^{(k-1)}} \right|, \quad \varepsilon_{k_{\text{eff}}}^{(k)} \equiv \left| \frac{k_{\text{eff}}^{(k)} - k_{\text{eff}}^{(k-1)}}{k_{\text{eff}}^{(k-1)}} \right|. \quad (27)$$

The convergence rate measured between two successive iterations does not change dramatically in the first few iterations, and the rate of solution improvement is relatively steady initially. Second, the DAAREM algorithm accelerates the Ronen method's convergence by order of magnitude ($\sim 10^2$ iterations instead of $\sim 10^3$). Third, the convergence trend exhibits fluctuations typical of Anderson accelerated algorithms.

5. Conclusions

The RM is implemented as an iterative scheme that employs an integral expression to introduce an equivalence between approximated physical expressions for the current, like using Fick's law in the diffusion approximation, and its integral expression from transport. Such equivalence is established by means of new free parameters, enabling corrections into the balance equation, either by letting the diffusion coefficient to change locally through the iterations, or by introducing new terms, like in the case of adding drift currents. These corrections are used in a diffusion solver to obtain more accurate solutions. So far, the RM has only been developed and implemented using the correction of the *net* currents (RM^{net}).

In this paper, two new implementations are presented. The first is the so-called RM *partial* currents corrections (denoted as RM^{p}), which applies the correction on the partial currents. Both RM^{net} and RM^{p} eliminate the possible division by zero when the flux gradient tends to zero thanks to the drift-like correction terms. The RM^{p} follows the pCMFD scheme used to accelerate advanced transport solvers, and was introduced with the aim of studying convergence with the implementation schemes based on drift corrections on the current, thanks to the higher stability reported by pCMFD. However, both RM^{net} and RM^{p} never showed non-convergence problems with Sood's test cases presented in this work. The differences observed between the RM^{net} and RM^{p} in the criticality and fundamental flux distribution are minor.

The second new implementation is the spatial correction of the diffusion coefficients (RM^{D}), close to the original form proposed by Ronen [1]. This implementation exhibits slightly better accuracy in predicting the flux distribution near the vacuum boundary compared to RM^{net} and RM^{p} . Indeed, the RM^{D} can apply a correction also to the extrapolation distance used to reproduce zero incoming current with the vacuum boundary because such distance is defined as proportional to the diffusion coefficient evaluated at the boundary. This explains somewhat the better agreement noticed in most cases, and suggests to study further in the future the closure relation used to reproduce vacuum at the boundary.

The three implementations are tested on 18 benchmark cases

provided by Sood [8]. The benchmark suite also includes problems showing scattering anisotropy, which allowed testing the new implementation of the current and the source used in the integral expression for the current. Note that RM can reproduce only problems with linearly scattering anisotropy because it relies on a diffusion problem where higher flux moments cannot be estimated.

As expected and known from the theory, the classic diffusion solver failed in general to provide precise results for the criticality eigenvalue and the flux distribution. The standard diffusion's deviations from criticality range from around 160 to 16,000 pcm, and the flux deviation ranges between 16% for diffusive problems and 83% for non-reflected slab problems.

The results indicate that the three implementations of the RM achieve significantly better accuracy in this set of problems. The maximal deviation in criticality is less than ten pcm, and the maximal deviation in the spatial distribution of the flux is less than 2% on the boundary (and generally $\sim 1\%$). This deviation trend is common to both isotropic and linearly-anisotropic problems.

In addition to the reference by Sood, the RM was also verified against S_{128} and CPM codes. The RM flux distribution tends to be slightly closer to CPM than S_{128} , which is attributed to the fact that both methods employ the same formalism of the integral transport equation.

Finally, the run times of the RM and the reference codes were evaluated for two benchmark problems using CPU time with ten realizations per problem. The RM, which is basically an iterative sequence of diffusion calculations, shows significantly faster runtime for one-dimensional, two-group homogeneous and heterogeneous problems. On average, the RM is three times faster than CPM (both calculating the scalar flux) and eight times faster than S_{128} , which calculates the angular fluxes using the typical sweep procedure on directions.

The convergence rate of the RM is relatively slow and is significantly accelerated by the DAAREM algorithm, enabling the RM to converge within $\sim 10^2$ iterations instead of $\sim 10^3$. The fluctuating nature of the convergence trend is typical of Anderson accelerated algorithms applied to fixed-point iterative problem.

Future work will include a more thorough investigation into the RM's computational time performances, stability and robustness of the implementations with respect to alternative formulations describing the vacuum boundary, and convergence properties of the RM.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

RG thanks the Israel Ministry of Energy for its support as part of the scholarship program for undergraduate and graduate students in energy-related fields, contract no. 219-11-045. JC thanks FRA-MATOME for its support as part of a collaboration agreement regarding this research.

Appendix A. Supplementary data

Supplementary data related to this article can be found at <https://doi.org/10.1016/j.net.2022.09.026>.

Appendix A. Collected data from Sood et al. 2003 [8].

Table A1
One-group macroscopic cross sections for critical problems with isotropic scattering [8].

Material	ν	Σ_f	Σ_s	Σ_t	c
Pu-239 (a)	3.24	0.08160	0.225126	0.3264	0.689724265
Pu-239 (b)	2.84	0.08160	0.225126	0.3264	0.689724265
H ₂ O	0.00	0.0	0.293760	0.3264	0.90
U-235 (a)	2.70	0.065280	0.248064	0.3264	0.76
U-D ₂ O	1.7	0.054628	0.464338	0.546280	0.85
H ₂ O (refl)	0.0	0.0	0.491652	0.546280	0.90
U-235 (e)	2.5	0.06922744	0.328042	0.407407	0.805194805
Fe (refl)	0.0	0.0	0.232094880	0.232560	0.998
Na (mod)	0.0	0.0	0.086368032	0.086368032	1.0

Table A2
One-group macroscopic cross sections for critical problems with linearly anisotropic scattering [8].

Material	ν	Σ_f	$\Sigma_{s,0}$	$\Sigma_{s,1}$	Σ_t	c
Pu-239 (a)	2.5	0.266667	0.733333	0.2	1.0	1.4
Pu-239 (b)	2.5	0.266667	0.733333	0.333333	1.0	1.4

Table A3
Two-group macroscopic cross sections for critical problems with isotropic scattering [8].

Material	g	ν_g	$\Sigma_{f,g}$	$\Sigma_{c,g}$	$\Sigma_{s,g \leftarrow g'}$	$\Sigma_{t,g}$	χ_g
PU	1	3.10	0.09360	0.00480	0.07920	0.	0.575
	2	2.93	0.08544	0.01440	0.04320	0.23616	0.425
U-235	1	2.5	0.06192	0.00384	0.07824	0.	0.575
	2	2.5	0.06912	0.01344	0.07200	0.26304	0.423
ResearchReactor (a)	1	2.5	1.04840×10^{-3}	1.00460×10^{-3}	0.625680	0.	1.
	2	2.5	5.06320×10^{-2}	2.57880×10^{-2}	0.029227	2.44383	0.
ResearchReactor (b)	1	2.5	8.36×10^{-4}	1.104×10^{-3}	0.838920	7.67×10^{-3}	1.
	2	2.5	2.9564×10^{-2}	2.4069×10^{-2}	0.046350	2.91830	0.
H ₂ O (a)(refl.)	1	0.	0.	7.4×10^{-4}	0.839750	3.36×10^{-4}	0.
	2	0.	0.	1.8564×10^{-2}	0.047490	2.9676	0.
U-D ₂ O	1	2.5	2.817×10^{-3}	8.7078×10^{-3}	3.1980×10^{-1}	0.	1.
	2	2.5	9.7×10^{-2}	2.5180×10^{-2}	4.5552×10^{-3}	0.424100	0.
U-Al	1	0.	0.	2.170×10^{-4}	0.247516	0.020432	1.
	2	2.830023	6.0706×10^{-2}	3.143×10^{-3}	1.213127	0.	0.

Table A4
Two-group macroscopic cross sections for critical problems with linearly anisotropic scattering [8].

Material	g	ν_g	$\Sigma_{f,g}$	$\Sigma_{c,g}$	$\Sigma_{s,g \leftarrow g'}^0$	$\Sigma_{s,g \leftarrow g'}^1$	$\Sigma_{t,g}$	χ_g
ResearchReactor (a)	1	2.5	0.0010484	0.0010046	0.625680	0.0	0.2745900	1.0
	2	2.5	0.0506320	0.0257880	0.029227	2.44383	0.0075737	0.0

Table A5
Values of the 1-group normalized scalar flux at four spatial positions. The flux is normalized by the value at the center [8].

Prob. #	$L_i = x/L_c$			
	$L_1 = 0.25$	$L_2 = 0.5$	$L_3 = 0.75$	$L_4 = 1.0$
6	0.97017340	0.88105400	0.73181310	0.49025920
12	0.96695060	0.86862590	0.70552180	0.44619120
22	0.93945236	0.76504084	0.49690627	0.13893858
Prob. #	Fe-U	U-Fe	Fe-Na	Na
30	1.229538	1.49712	1.324899	0.91227

Table A6
Values of the two-group normalized scalar flux at four spatial positions. The flux is normalized by the value of the fast-group flux at the center [8].

Prob. #	Group	$L_i = x/L_c$			
		$L_1 = 0.241394$	$L_2 = 0.502905$	$L_3 = 0.744300$	$L_4 = 1.0$
54	1	0.943363	0.761973	0.504012	0.1475980
	2	0.340124	0.273056	0.173845	0.0212324
Prob.#	Group	$L_1 = 0.2$	$L_2 = 0.5$	$L_3 = 0.8$	$L_4 = 1.0$
71	1	0.963873	0.781389	0.472787	0.1895780
	2	0.349006	0.280870	0.157376	0.0277639

References

- [1] Y. Ronen, Accurate relations between the neutron current densities and the neutron fluxes, *Nucl. Sci. Eng.* 146 (2) (2004) 245–247.
- [2] D. Tomatis, A. Dall'Osso, Application of a numerical transport correction in diffusion calculations, in: *Int. Conf. on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2011)*, Rio de Janeiro, RJ, Brazil, May 8–12, 2011, 05 2011.
- [3] Daniele Tomatis, Roy Gross, Erez Gilad, On the ronon method in simple 1-D geometries for neutron transport theory solutions, *J. Comput. Theor. Trans.* 50 (2) (2021) 134–157.
- [4] Roy Gross, Daniele Tomatis, Erez Gilad, High-accuracy neutron diffusion calculations based on integral transport theory, *Eur. Phys. J. Plus* 135 (2) (2020) 235.
- [5] Kord S. Smith, Nodal method storage reduction by nonlinear iteration, *Trans. Am. Nucl. Soc.* 44 (1983) 265–266.
- [6] Nicholas C. Henderson, Ravi Varadhan, Damped anderson acceleration with restarts and monotonicity control for accelerating em and em-like algorithms, *J. Computational Graphical Stat.* 28 (4) (2019) 834–846.
- [7] Homer F. Walker, Peng Ni, Anderson acceleration for fixed-point iterations, *SIAM J. Numer. Anal.* 49 (4) (2011) 1715–1735.
- [8] Avneet Sood, R. Arthur Forster, D. Kent Parsons, Analytical benchmark test set for criticality code verification, *Prog. Nucl. Energy* 42 (1) (2003) 55–106.
- [9] Milton Abramowitz, Irene A. Stegun, *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, Dover Publications New York, 1972.
- [10] Nam Zin Cho, Gil Soo Lee, Comparison of cmfd and p-cmfd acceleration methods for neutron transport calculations, *Proc. Korean Nucl. Soc. Gyeongju, Korea* (2003) 53.1.
- [11] E.E. Lewis, W.F. Miller, *Computational Methods of Neutron Transport*, John Wiley and Sons, Inc, New York, NY, La Grange Park, Ill., USA, 1984.
- [12] W.M. Stacey, *Nuclear Reactor Physics*, John Wiley & Sons, 2007.
- [13] J.R. Lamarsh, *Introduction to Nuclear Reactor Theory*, Addison-Wesley, Reading, Mass, 1966.
- [14] J.J. Duderstadt, L.J. Hamilton, *Nuclear Reactor Analysis*, Wiley, 1976.
- [15] M. Jarrett, et al., Analysis of stabilization techniques for CMFD acceleration of neutron transport problems, *Nucl. Sci. Eng.* 184 (2) (2016) 208–227.
- [16] A. Zhu, et al., An optimally diffusive coarse mesh finite difference method to accelerate neutron transport calculations, *Ann. Nucl. Energy* 95 (2016) 116–124.
- [17] Junzi Zhang, Brendan O'Donoghue, Stephen Boyd, Globally convergent type-I Anderson acceleration for nonsmooth fixed-point iterations, *SIAM J. Optimization* 30 (4) (2020) 3170–3197.
- [18] Ansar Calloo, Le Tellier, Romain, Couyras, David, Comparison of Chebyshev and Anderson accelerations for the neutron transport equation, *EPJ Web Conf.* 247 (2021), 03001.