Proceedings of the Korean Nuclear Society Spring Meeting
Cheju, Korea, May 2001

The Additive Angular Rebalance Acceleration Method for Solving Neutron Transport Equations in X-Y Geometry

Chang Je Park and Nam Zin Cho
Korea Advanced Institute of Science and Technology
Department of Nuclear Engineering
373-1 Kusong-dong, Yusong-gu
Taejon, Korea 305-701
Email : nzcho@sorak.kaist.ac.kr

Abstract

The additive angular dependent rebalance (AADR) factor acceleration method proposed by the authors previously is an effective acceleration method for the the discrete ordinates neutron transport equation. For slab geometry problems, it was demonstrated via Fourier analysis that the spectral radius of the AADR method is less than that of diffusion synthetic acceleration (DSA) method. In this paper, a continuous Fourier analysis is performed for x-y geometry to analyze the stability of the additive angular dependent rebalance factor method. As a result, the optimal weighting functions can be obtained. We also suggest an AADR with directional S_2-like weighting functions to get better convergence. The Fourier analysis shows that the AADR with directional S_2-like weighting functions which uses two different rebalance factors for x and y directions per octant provide better results than the AADR with normal S_2-like weighting functions which uses a single weighting function per octant. Numerical tests also confirm our suggestion.

1. Introduction

Various algorithms have been developed in the past to accelerate the source iteration for discrete ordinates equation. Among these algorithms, the diffusion synthetic acceleration (DSA) [1][2] is most popular and unconditionally stable and rapidly convergent. But it needs consistency of discretization scheme between high-order equation and low-order equation. To escape those burdens, various methods have been suggested and tested. Recently boundary projection acceleration (BPA) [3] transport synthetic acceleration (TSA) [4], and angular dependent rebalance (ADR) [5][6] factor acceleration methods were developed. Unlike DSA, these schemes have generality with respect to geometry, discretization scheme, and mesh shape.

We have developed an additive angular dependent rebalance (AADR) factor algorithm [7][8] which is a linearized form of angular dependent rebalance (ADR) factor acceleration. It was found that the effect of acceleration depends on the weighting function and optimal rebalance factor could be obtained from a continuous Fourier analysis in slab geometry. Among various weighting functions, a linear weighting function ($W_n = |\mu_n| + 2.27$) gives spectral radius less than $0.1865c$, which is smaller than that of DSA.

In two-dimensional problems, the formula of AADR becomes more complicated and considers more rebalance factors to get better convergence. The optimal weighting function may be also chosen via standard Fourier analysis which was done in slab geometry. We find that the normal S_2-like rebalance acceleration of AADR does not give any merits compared with existing DSA method but a directional S_2-like rebalance factor gives better results. This result is similar to that of the DP_1 or S_4 acceleration of AADR in slab geometry. That is, the use of more rebalance factors gives the fewer iteration numbers but it might need slightly long computing time to solve the low-order equations. To solve the low-order equation effectively, a preconditoned Bi-CGSTAB algorithm is used. The preconditioner is obtained by a transport sweep incomplete factorization in this study.

In section 2, we briefly describe the differencing schemes which are considered in this paper. The diamond difference (DD) and the linear multiple balance (LMB) schemes [7] are briefly described for two-dimensional geometry. In this linear multiple balance approach, one mesh cell is divided into two subcells with quadratic approximation for the angular flux distribution. Several multiple balance equations are used to relate center angular flux with average angular flux by Simpson's rule. From the analysis of spatial truncation error, the accuracy of the linear multiple balance scheme is better than that of diamond difference scheme. The positivity of the method is also stronger than that of diamond differencing. In section 3, we analyze the additive angular rebalance (AADR) factor acceleration with continuous Fourier analysis. From these results, directional S_2-like rebalance factors in AADR give smaller spectral radius than AADR with normal S_2-like rebalance factors. In section 4, we give the numerical results, and finally in section 5, we present the conclusions.

2. Derivation of LMB Scheme for X-Y Geometry

To apply AADR, we consider several different discretizing schemes - diamond difference (DD) scheme, linear multiple balance (LMB) scheme, nodal schemes (C-C and C-L) [9]. We will describe briefly DD and LMB schemes. We consider the following one group x-y geometry discrete ordinates transport equation in standard notation :

$$\mu_n \frac{d\psi_n(x,y)}{dx} + \eta_n \frac{d\psi_n(x,y)}{dy} + \sigma\psi_n(x,y) = q_n(x,y), \tag{1}$$

where

$$\begin{aligned} q_n(x,y) &= \sigma_s \phi(x,y) + S_n(x,y), \\ \phi(x,y) &= \sum_{n=1}^{N} w_n \psi_n(x,y), \end{aligned} \tag{2}$$

and μ_n, η_n and w_n are a discrete ordinate set and its weight, respectively.

Integrating over $x_{i-1/2} < x < x_{i+1/2}$ and $y_{j-1/2} < y < y_{j+1/2}$, we obtain the spatial balance equation as

$$\frac{\mu_n}{\Delta_i}(\psi_{n,i+1/2,j} - \psi_{n,i-1/2,j}) + \frac{\eta_n}{\Delta_j}(\psi_{n,i,j+1/2} - \psi_{n,i,j-1/2}) + \sigma\psi_{n,i,j} = q_{n,i,j}, \tag{3}$$

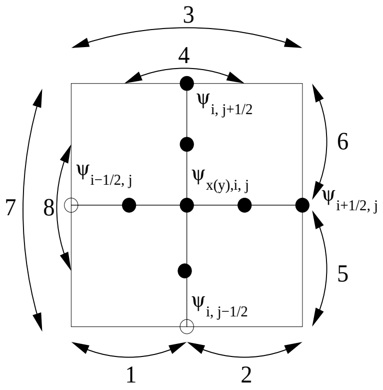

Figure 1: Configuration of unknowns of two dimensional problem.

where

$$
\begin{aligned}
\psi_{n,i+1/2,j} &= \frac{1}{\Delta_j}\int_{y_{j-1/2}}^{y_{j+1/2}} \psi_n(x_{i+1/2},y)dy,\\
\psi_{n,i,j+1/2} &= \frac{1}{\Delta_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \psi_n(x,y_{j+1/2})dx,\\
\psi_{n,i,j} &= \frac{1}{\Delta_i\Delta_j}\int_{x_{i-1/2}}^{x_{i+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}} \psi_n(x,y)dxdy,\\
q_{n,i,j} &= \frac{1}{\Delta_i\Delta_j}\int_{x_{i-1/2}}^{x_{i+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}} q_n(x,y)dxdy,\\
\Delta_i &= x_{i+1/2}-x_{i-1/2},\ \Delta_j = y_{j+1/2}-y_{j-1/2}.
\end{aligned}
\tag{4}
$$

Diamond difference (DD) approximation is

$$
\begin{aligned}
\psi_{n,i,j} &= \frac{1}{2}(\psi_{n,i+1/2,j}+\psi_{n,i-1/2,j}),\\
\psi_{n,i,j} &= \frac{1}{2}(\psi_{n,i,j+1/2}+\psi_{n,i,j-1/2}).
\end{aligned}
\tag{5}
$$

Thus we can obtain the average angular flux in the case of $\mu_n > 0$ and $\eta_n > 0$.

$$
\begin{aligned}
&\frac{\mu_n}{\Delta_i}(2\psi_{n,i,j}-2\psi_{n,i-1/2,j})+\frac{\eta_n}{\Delta_j}(2\psi_{n,i,j}-2\psi_{n,i,j-1/2})+\sigma\psi_{n,i,j} = q_{n,i,j},\\
&\psi_{n,i,j} = \frac{2\frac{\mu_n}{\Delta_i}\psi_{n,i-1/2,j})+2\frac{\eta_n}{\Delta_j}\psi_{n,i,j-1/2}+q_{n,i,j}}{2\frac{\mu_n}{\Delta_i}+2\frac{\eta_n}{\Delta_j}+\sigma}.
\end{aligned}
\tag{6}
$$

Linear multiple balance (LMB) method is composed of four multiple balance equations for slab geometry problems. To solve rectangular geometry problems, eight multiple balance equations are derived on a mesh cell which is divided into four subcells. For example, the unknowns of linear multiple balance (LMB) equations are given at a cell in Fig. 1. Especially, two balance equations (balance 4 and 8) are set up to overlap over the subcells.

The first four linear multiple balance equations for x-direction are given by

$$
\begin{aligned}
&2\frac{\mu_n}{\Delta_i}(\psi_{nx,i,j}-\psi_{n,i-1/2,j})+\frac{\eta_n}{\Delta_j}(\psi_{n,i,j+1/2}-\psi_{n,i,j-1/2})\\
&\quad+\sigma\frac{\psi_{n,i-1/2,j}+4\psi_{n,i-1/4,j}+\psi_{nx,i,j}}{6}=q^L_{n,x,i,j},\\
&2\frac{\mu_n}{\Delta_i}(\psi_{n,i+1/2,j}-\psi_{nx,i,j})+\frac{\eta_n}{\Delta_j}(\psi_{n,i,j+1/2}-\psi_{n,i,j-1/2})\\
&\quad+\sigma\frac{\psi_{nx,i,j}+4\psi_{n,i+1/4,j}+\psi_{n,i+1/2,j}}{6}=q^R_{n,x,i,j},\\
&2\frac{\mu_n}{\Delta_i}(\psi_{n,i+1/2,j}-\psi_{n,i-1/2,j})+\frac{\eta_n}{\Delta_j}(\psi_{n,i,j+1/2}-\psi_{n,i,j-1/2})\\
&\quad+\sigma\frac{\psi_{n,i-1/2,j}+4\psi_{nx,i,j}+\psi_{n,i+1/2,j}}{6}=q^L_{n,x,i,j}+q^R_{n,x,i,j},\\
&2\frac{\mu_n}{\Delta_i}(\psi_{n,i+1/4,j}-\psi_{n,i-1/4,j})+\frac{\eta_n}{\Delta_j}(\psi_{n,i,j+1/2}-\psi_{n,i,j-1/2})\\
&\quad+\sigma\frac{\psi_{n,i-1/4,j}+4\psi_{nx,i,j}+\psi_{n,i+1/4,j}}{6}=\frac{q^L_{n,x,i,j}+q^R_{n,x,i,j}}{2}.
\end{aligned}
\tag{7}
$$

where

$$
\begin{aligned}
\psi_{nx,i,j}&=\frac{1}{\Delta_j}\int_{y_{j-1/2}}^{y_{j+1/2}}\psi_n(x_i,y)dx\\
\psi_{ny,i,j}&=\frac{1}{\Delta_i}\int_{x_{i-1/2}}^{x_{i+1/2}}\psi_n(x,y_j)dx\\
\psi_{n,i+1/2,j}&=\frac{1}{\Delta_j}\int_{y_{j-1/2}}^{y_{j+1/2}}\psi_n(x_{i+1/2},y)dy,\\
\psi_{n,i,j+1/2}&=\frac{1}{\Delta_i}\int_{x_{i-1/2}}^{x_{i+1/2}}\psi_n(x,y_{j+1/2})dx,\\
\psi_{n,i,j}&=\frac{1}{\Delta_i\Delta_j}\int_{x_{i-1/2}}^{x_{i+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}}\psi_n(x,y)dxdy,\\
q_{n,i,j}&=\frac{1}{\Delta_i\Delta_j}\int_{x_{i-1/2}}^{x_{i+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}}q_n(x,y)dxdy,\\
q^L_{n,x,i,j}&=\frac{2}{\Delta_i\Delta_j}\int_{x_{i-1/2}}^{x_i}\int_{y_{j-1/2}}^{y_{j+1/2}}q_n(x,y)dxdy\\
&=\sigma_s\sum_n w_n\frac{\psi_{n,i-1/2,j}+4\psi_{n,i-1/4,j}+\psi_{nx,i,j}}{6}+\frac{S_{n,i,j}}{2}.
\end{aligned}
\tag{8}
$$

The other four linear multiple balance equations for y-direction may be also derived similarly. An average angular flux can be obtained from neutron balance equation :

$$
\psi_{n,i,j}=\frac{1}{\sigma}\{q_{n,i,j}-\frac{\mu_n}{\Delta_i}(\psi_{n,i+1/2,j}-\psi_{n,i-1/2,j})-\frac{\eta_n}{\Delta_j}(\psi_{n,i,j+1/2}-\psi_{n,i,j-1/2})\}.
\tag{9}
$$

To get exiting angular fluxes, the following matrix form of the eight balance equations is obtained:

$$
\mathbf{A}\ \mathbf{x}=\mathbf{B},
\tag{10}
$$

where

$$\mathbf{A} = \begin{pmatrix} 4 & 1+2a & 0 & 0 & 0 & 0 & 0 & b \\ 0 & 1-2a & 4 & 1+2a & 0 & 0 & 0 & b \\ 0 & 4 & 0 & 1+a & 0 & 0 & 0 & b \\ 1-2a & 4 & 1+2a & 0 & 0 & 0 & 0 & b \\ 0 & 0 & 0 & a & 4 & 1+2b & 0 & 0 \\ 0 & 0 & 0 & a & 0 & 1-2b & 4 & 1+2b \\ 0 & 0 & 0 & a & 0 & 4 & 0 & 1+b \\ 0 & 0 & 0 & a & 1-2b & 4 & 1+2b & 0 \end{pmatrix},$$

$$\mathbf{B} = \begin{pmatrix} \frac{6}{\sigma}q^L_{n,x,i,j} + (1-2a)\psi_{n,i-1/2,j} + b\psi_{n,i,j-1/2} \\ \frac{6}{\sigma}q^R_{n,x,i,j} + b\psi_{n,i,j-1/2} \\ \frac{6}{\sigma}(q^L_{n,x,i,j} + q^R_{n,x,i,j}) + (1-a)\psi_{n,i-1/2,j} + b\psi_{n,i,j-1/2} \\ \frac{3}{\sigma}(q^L_{x,i,j} + q^R_{x,i,j}) + b\psi_{n,i,j-1/2} \\ \frac{6}{\sigma}q^L_{n,y,i,j} + (1-2b)\psi_{n,i,j-1/2} + a\psi_{n,i-1/2,j} \\ \frac{6}{\sigma}q^R_{n,y,i,j} + a\psi_{n,i-1/2,j} \\ \frac{6}{\sigma}(q^L_{n,y,i,j} + q^R_{n,y,i,j}) + (1-b)\psi_{n,i,j-1/2} + a\psi_{n,i-1/2,j} \\ \frac{3}{\sigma}(q^L_{n,y,i,j} + q^R_{n,y,i,j}) + a\psi_{n,i-1/2,j} \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} \psi_{n,i-1/4,j} \\ \psi_{nx,i,j} \\ \psi_{n,i+1/4,j} \\ \psi_{n,i+1/2,j} \\ \psi_{n,i,j-1/4} \\ \psi_{ny,i,j} \\ \psi_{n,i,j+1/4} \\ \psi_{n,i,j+1/2} \end{pmatrix}, \tag{11}$$

$$a = \frac{6|\mu_n|}{\sigma\Delta_i}, \quad b = \frac{6|\eta_n|}{\sigma\Delta_j}.$$

The determinant of the system matrix is positive, and the angular fluxes are obtained by direct inversion. So we get the analytic form for the exiting angular fluxes on each direction.

3. Additive Angular Dependent Rebalance (AADR) Acceleration

Standard source iteration (SI) equations for two-dimensional problem are

$$\begin{aligned} &\mu_n \frac{\partial}{\partial x}\psi_n^{l+1/2} + \eta_n \frac{\partial}{\partial y}\psi_n^{l+1/2} + \sigma\psi_n^{l+1/2} = \sigma_s\phi^l + S_n, \\ &\phi^{l+1} = \frac{1}{4}\sum_n w_n\psi_n^{l+1/2} = \phi^{l+1/2}, \end{aligned} \tag{12}$$

where l denotes iteration index. We know that the spectral radius of SI is c (scattering ratio) when the standard Fourier analysis is applied. If c goes to unity, the convergence becomes worse and it might well need many iterations and long computing time. For this reason, many acceleration methods are suggested and developed. Among them, the diffusion synthetic acceleration (DSA) method is most popular but it might some happen instability in two-dimensional problems. Recently, several people have devised the non-linear acceleration scheme with rebalancing the angular domain or spatial domain. We also have developed the linearized form of angular dependent rebalance (ADR) factor method, the additive angular dependent rebalance (AADR) method. These acceleration methods including ADR and AADR, the weighting functions must be chosen to get optimal spectral radius. Recently, we suggested how to determine optimal weighting functions from analytic Fourier analysis in slab geometry. Extending the previous works, we can also get the optimal weighting function in two-dimensional problems.

First, the diffusion synthetic acceleration (DSA), the most popular acceleration method for neutron transport equation, is briefly reviewed using standard Fourier analysis. Iteration equations of DSA are given by

$$
\begin{aligned}
&\mu_n \frac{\partial}{\partial x}\psi_n^{l+1/2} + \eta_n \frac{\partial}{\partial y}\psi_n^{l+1/2} + \sigma\psi_n^{l+1/2} = \sigma_s\phi^l + S_n,\\
&\phi^{l+1/2} = \frac{1}{4}\sum_n w_n \psi_n^{l+1/2},\\
&-\frac{1}{3\sigma}\frac{\partial^2}{\partial x^2}f^{l+1} - \frac{1}{3\sigma}\frac{\partial^2}{\partial y^2}f^{l+1} + \sigma f^{l+1} = \sigma_s(\phi_0^{l+1} - \phi_0^l),\\
&\phi^{l+1} = \phi^{l+1/2} + f^{l+1},
\end{aligned}
\tag{13}
$$

where

$$
f^{l+1} = \phi^{l+1} - \phi^{l+1/2}. \tag{14}
$$

Fourier ansatz are defined as

$$
\begin{aligned}
\psi_n^{l+1/2} &= A_n \omega^l e^{j(\lambda_x x + \lambda_y y)},\\
\phi^l &= B\omega^l e^{j(\lambda_x x + \lambda_y y)},\\
\phi^{l+1/2} &= D\omega^l e^{j(\lambda_x x + \lambda_y y)},\\
f^{l+1} &= F\omega^l e^{j(\lambda_x x + \lambda_y y)}.
\end{aligned}
\tag{15}
$$

Using the above equations, the following relations are derived with assumption of $\sigma = 1,\ \sigma_s = c,\ S_n = 0$.

$$
\begin{aligned}
&(j(\mu_n\lambda_x + \eta_n\lambda_y) + 1)A_n = cB,\\
&\omega D = \frac{1}{4}\sum_n w_n A_n,\\
&(\lambda_x^2/3 + \lambda_y^2/3 + 1)F = c(\omega - 1)B,\\
&\omega B = D + F.
\end{aligned}
\tag{16}
$$

Rearranging the above equations, the eigenvalue (ω) is

$$
\begin{aligned}
&\omega = \frac{1}{4}\sum_n w_n \frac{c}{1 + j(\mu_n\lambda_x + \eta_n\lambda_y)} + \frac{c(\omega - 1)}{\lambda_x^2/3 + \lambda_y^2/3 + 1},\\
&(1 - \frac{c}{\lambda_x^2/3 + \lambda_y^2/3 + 1})\omega = \frac{1}{4}\sum_n w_n \frac{c}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2}\\
&\qquad - \frac{c}{\lambda_x^2/3 + \lambda_y^2/3 + 1}\frac{1}{4}\sum_n w_n \frac{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2}\\
&(1 - c + \lambda_x^2/3 + \lambda_y^2/3)\omega = \frac{c}{4}\sum_n w_n \frac{\lambda_x^2/3 + \lambda_y^2/3 - (\mu_n\lambda_x + \eta_n\lambda_y)^2}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2}
\end{aligned}
\tag{17}
$$

The spectral radius (ρ) is given by

$$
\begin{aligned}
\rho &= sup_{\lambda_x,\lambda_y} \frac{c(\lambda_x^2/3 + \lambda_y^2/3)}{1 - c + \lambda_x^2/3 + \lambda_y^2/3}\left|\frac{1}{4}\sum_n w_n \frac{1 - \frac{(\mu_n\lambda_x + \eta_n\lambda_y)^2}{\lambda_x^2/3 + \lambda_y^2/3}}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2}\right|\\
&\leq sup_{\lambda_x,\lambda_y} c\left|\frac{1}{4}\sum_n w_n \frac{1 - \frac{(\mu_n\lambda_x + \eta_n\lambda_y)^2}{\lambda_x^2/3 + \lambda_y^2/3}}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2}\right| = 0.2247c.
\end{aligned}
\tag{18}
$$

However, the general form of AADR in S_2 form is written as

$$\begin{aligned}
&\mu_n \frac{\partial}{\partial x}\psi_n^{l+1/2} + \eta_n \frac{\partial}{\partial y}\psi_n^{l+1/2} + \sigma\psi_n^{l+1/2} = \sigma_s\phi^l + S_n, \\
&\phi^{l+1/2} = \frac{1}{4}\sum_n w_n\psi_n^{l+1/2}, \\
&k_1\frac{\partial}{\partial x}f_1^{l+1} + k_2\frac{\partial}{\partial y}f_1^{l+1} + \sigma f_1^{l+1} = \sigma_s(\phi^{l+1} - \phi^l), \\
-&k_1\frac{\partial}{\partial x}f_2^{l+1} + k_2\frac{\partial}{\partial y}f_2^{l+1} + \sigma f_2^{l+1} = \sigma_s(\phi^{l+1} - \phi^l), \\
-&k_1\frac{\partial}{\partial x}f_3^{l+1} - k_2\frac{\partial}{\partial y}f_3^{l+1} + \sigma f_3^{l+1} = \sigma_s(\phi^{l+1} - \phi^l), \\
&k_1\frac{\partial}{\partial x}f_4^{l+1} - k_2\frac{\partial}{\partial y}f_4^{l+1} + \sigma f_4^{l+1} = \sigma_s(\phi^{l+1} - \phi^l), \\
&\phi^{l+1} = \phi^{l+1/2} + \frac{f_1^{l+1} + f_2^{l+1} + f_3^{l+1} + f_4^{l+1}}{4},
\end{aligned} \tag{19}$$

where

$$\begin{aligned}
k_1 &= \frac{\sum_n w_n\mu_n W(\mu_n,\eta_n)}{\sum_n w_n W(\mu_n,\eta_n)}, \\
k_2 &= \frac{\sum_n w_n\eta_n W(\mu_n,\eta_n)}{\sum_n w_n W(\mu_n,\eta_n)}, \\
f_1 &= \psi_n^{l+1} - \psi_n^{l+1/2}, \;\; \mu_n > 0, \; \eta_n > 0, \\
f_2 &= \psi_n^{l+1} - \psi_n^{l+1/2}, \;\; \mu_n < 0, \; \eta_n > 0, \\
f_3 &= \psi_n^{l+1} - \psi_n^{l+1/2}, \;\; \mu_n < 0, \; \eta_n < 0, \\
f_4 &= \psi_n^{l+1} - \psi_n^{l+1/2}, \;\; \mu_n > 0, \; \eta_n < 0.
\end{aligned} \tag{20}$$

We want to know the optimal spectral radius for various weighting function using Fourier analysis. First, the Fourier ansatz are defined as

$$\begin{aligned}
\psi_n^{l+1/2} &= A_n\omega^l e^{j(\lambda_x x + \lambda_y y)}, \\
\phi^l &= B\omega^l e^{j(\lambda_x x + \lambda_y y)}, \\
\phi^{l+1/2} &= D\omega^l e^{j(\lambda_x x + \lambda_y y)}, \\
f_m^{l+1} &= F_m\omega^l e^{j(\lambda_x x + \lambda_y y)}, \quad m = 1, 2, 3, 4.
\end{aligned} \tag{21}$$

If we assume that $\sigma = 1,\ \sigma_s = c,\ S_n = 0$, then we get

$$\begin{aligned}
&(j(\mu_n\lambda_x + \eta_n\lambda_y) + 1)A_n = cB, \\
&\omega D = \frac{1}{4}\sum_n w_n A_n, \\
&(j(k_1\lambda_x + k_2\lambda_y) + 1)F_1 = c(\omega - 1)B, \\
&(j(-k_1\lambda_x + k_2\lambda_y) + 1)F_2 = c(\omega - 1)B, \\
&(j(-k_1\lambda_x - k_2\lambda_y) + 1)F_3 = c(\omega - 1)B, \\
&(j(k_1\lambda_x - k_2\lambda_y) + 1)F_4 = c(\omega - 1)B, \\
&\omega B = D + \frac{F_1 + F_2 + F_3 + F_4}{4},
\end{aligned} \tag{22}$$

Getting the sum of the rebalance factors with the above relations,

$$F_1 + F_2 + F_3 + F_4 = 2c(\omega - 1)B\Big(\frac{1}{1 + (k_1\lambda_x + k_2\lambda_y)^2} + \frac{1}{1 + (k_1\lambda_x - k_2\lambda_y)^2}\Big). \tag{23}$$

The eigenvalue (ω) is written as

$$\begin{aligned}
&\omega = \frac{1}{4}\sum_n w_n \frac{c}{1 + j(\mu_n\lambda_x + \eta_n\lambda_y)} + (\frac{c(\omega - 1)/2}{1 + (k_1\lambda_x + k_2\lambda_y)^2} + \frac{c(\omega - 1)/2}{1 + (k_1\lambda_x - k_2\lambda_y)^2}), \\
&\Big(1 - (\frac{c/2}{1 + (k_1\lambda_x + k_2\lambda_y)^2} + \frac{c/2}{1 + (k_1\lambda_x - k_2\lambda_y)^2})\Big)\omega = \frac{1}{4}\sum_n w_n \frac{c}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2} \\
&\quad - \Big(\frac{c/2}{1 + (k_1\lambda_x + k_2\lambda_y)^2} + \frac{c/2}{1 + (k_1\lambda_x - k_2\lambda_y)^2}\Big) \times \frac{1}{4}\sum_n w_n \frac{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2},
\end{aligned} \tag{24}$$

leading to

$$\begin{aligned}
&\Big(1 - c + (2 - c)((k_1\lambda_x)^2 + (k_2\lambda_y)^2) + ((k_1\lambda_x)^2 - (k_2\lambda_y)^2)^2\Big)\omega \\
&= \Big((k_1\lambda_x)^2 + (k_2\lambda_y)^2 + ((k_1\lambda_x)^2 - (k_2\lambda_y)^2)^2\Big) \times \frac{1}{4}\sum_n w_n \frac{c}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2} \\
&\quad \times \Big(1 - \frac{(1 + (k_1\lambda_x)^2 + (k_2\lambda_y)^2)(\mu_n\lambda_x + \eta_n\lambda_y)^2}{(k_1\lambda_x)^2 + (k_2\lambda_y)^2 + ((k_1\lambda_x)^2 - (k_2\lambda_y)^2)^2}\Big).
\end{aligned} \tag{25}$$

The spectral radius (ρ) is given by

$$\rho \leq sup_{\lambda_x,\lambda_y} c\Big|\frac{1}{4}\sum_n w_n \frac{1}{1 + (\mu_n\lambda_x + \eta_n\lambda_y)^2} \times \Big(1 - \frac{(1 + (k_1\lambda_x)^2 + (k_2\lambda_y)^2)(\mu_n\lambda_x + \eta_n\lambda_y)^2}{(k_1\lambda_x)^2 + (k_2\lambda_y)^2 + ((k_1\lambda_x)^2 - (k_2\lambda_y)^2)^2}\Big)\Big|. \tag{26}$$

To get the optimal spectral radius, weighting coefficients (k_1, k_2) are obtained. Weighting functions are also evaluated to satisfy this weighting coefficients. First, we suggest a linear type of weighting function such as

$$\begin{aligned}
&k_1 = k_2 = 0.7, \quad \rho < 0.4469c, \\
&W(\mu, \eta) = |\mu| + |\eta| - 0.5.
\end{aligned} \tag{27}$$

The results do not provide good enough spectral radii compared with those of DSA. Thus we next suggest the directional S_2-like rebalance factor of AADR, where different directional rebalance factors are used to update different directional scalar fluxes. The iteration equations of AADR with directional S_2-like rebalance factors are given by

$$\begin{aligned}
&\mu_n \frac{\partial}{\partial x}\psi_n^{l+1/2} + \eta_n \frac{\partial}{\partial y}\psi_n^{l+1/2} + \sigma\psi_n^{l+1/2} = \sigma_s\phi^l + S_n,\\
&\phi^{l+1/2} = \frac{1}{4}\sum_n w_n \psi_n^{l+1/2},\\
&k_u \frac{\partial}{\partial x} f_{1,u}^{l+1} + k_v \frac{\partial}{\partial y} f_{1,u}^{l+1} + \sigma f_{1,u}^{l+1} = \sigma_s(\phi^{l+1} - \phi^l),\\
&-k_u \frac{\partial}{\partial x} f_{2,u}^{l+1} + k_v \frac{\partial}{\partial y} f_{2,u}^{l+1} + \sigma f_{2,u}^{l+1} = \sigma_s(\phi^{l+1} - \phi^l),\\
&-k_u \frac{\partial}{\partial x} f_{3,u}^{l+1} - k_v \frac{\partial}{\partial y} f_{3,u}^{l+1} + \sigma f_{3,u}^{l+1} = \sigma_s(\phi^{l+1} - \phi^l),\\
&k_u \frac{\partial}{\partial x} f_{4,u}^{l+1} - k_v \frac{\partial}{\partial y} f_{4,u}^{l+1} + \sigma f_{4,u}^{l+1} = \sigma_s(\phi^{l+1} - \phi^l),\\
&\quad u = x, v = y \text{ or } u = y, v = x\\
&\phi^{l+1} = \phi^{l+1/2} + \frac{1}{8}\sum_{m=1}^{4}(f_{m,x}^{l+1} + f_{m,y}^{l+1}),
\end{aligned} \tag{28}$$

where

$$\begin{aligned}
&f_{m,x} = \psi_n^{l+1} - \psi_n^{l+1/2}, \quad W = W_x,\\
&f_{m,y} = \psi_n^{l+1} - \psi_n^{l+1/2}, \quad W = W_y, \quad m = 1,2,3,4,\\
&k_x = \frac{\sum_n w_n\mu_n W_x(\mu_n,\eta_n)}{\sum_n w_n W_x(\mu_n,\eta_n)} = \frac{\sum_n w_n\eta_n W_y(\mu_n,\eta_n)}{\sum_n w_n W_y(\mu_n,\eta_n)},\\
&k_y = \frac{\sum_n w_n\mu_n W_y(\mu_n,\eta_n)}{\sum_n w_n W_y(\mu_n,\eta_n)} = \frac{\sum_n w_n\eta_n W_x(\mu_n,\eta_n)}{\sum_n w_n W_x(\mu_n,\eta_n)}.
\end{aligned} \tag{29}$$

To choose directional rebalance factors properly, several conditions are given for directional weighting functions such as

$$\begin{aligned}
&W_x + W_y = 2,\\
&\sum_n w_n W_x(\mu_n,\eta_n) = \sum_n w_n W_y(\mu_n,\eta_n) = 1,\\
&\sum_n w_n\mu_n W_x(\mu_n,\eta_n) = \sum_n w_n\eta_n W_y(\mu_n,\eta_n),\\
&\sum_n w_n\eta_n W_x(\mu_n,\eta_n) = \sum_n w_n\mu_n W_y(\mu_n,\eta_n),\\
&W_x(\mu_n,\eta_n) = a(|\mu_n| - |\eta_n|) + 1, \quad W_y(\mu_n,\eta_n) = -a(|\mu_n| - |\eta_n|) + 1,
\end{aligned} \tag{30}$$

and

$$\begin{aligned}
\phi^{l+1} - \phi^{l+1/2} &= \sum_n w_n (\psi_n^{l+1} - \psi_n^{l+1/2}), \\
&= \frac{1}{4} \sum_n w_n \frac{W_x + W_y}{2} (\psi_n^{l+1} - \psi_n^{l+1/2}), \\
&= \frac{1}{4} \sum_n w_n \left\{ \frac{W_x}{2} (\psi_n^{l+1} - \psi_n^{l+1/2}) + \frac{W_y}{2} (\psi_n^{l+1} - \psi_n^{l+1/2}) \right\}, \\
&= \frac{1}{4} \sum_n w_n \left\{ \frac{W_x}{2} f_{m,x}^{l+1} + \frac{W_y}{2} f_{m,y}^{l+1} \right\}, \\
&= \frac{f_{m,x} + f_{m,y}}{8}, \quad m = 1, 2, 3, 4.
\end{aligned} \tag{31}$$

Fourier ansatz are also given as

$$\begin{aligned}
\psi_n^{l+1/2} &= A_n \omega^l e^{j(\lambda_x x + \lambda_y y)}, \\
\phi^l &= B \omega^l e^{j(\lambda_x x + \lambda_y y)}, \\
\phi^{l+1/2} &= D \omega^l e^{j(\lambda_x x + \lambda_y y)}, \\
f_{m,u}^{l+1} &= F_{m,u} \omega^l e^{j(\lambda_x x + \lambda_y y)}, \quad m = 1, 2, 3, 4, \ u = x, y.
\end{aligned} \tag{32}$$

If we assume that $\sigma = 1, \ \sigma_s = c, \ S_n = 0$, the following equations are obtained:

$$\begin{aligned}
&(j(\mu\lambda_x + \eta\lambda_y) + 1)A = cB, \\
&\omega D = \frac{1}{4} \sum_n w_n A, \\
&(j(k_u\lambda_x + k_v\lambda_y) + 1)F_{1,u} = c(\omega - 1)B, \\
&(j(-k_u\lambda_x + k_v\lambda_y) + 1)F_{2,u} = c(\omega - 1)B, \\
&(j(-k_u\lambda_x - k_v\lambda_y) + 1)F_{3,u} = c(\omega - 1)B, \\
&(j(k_u\lambda_x - k_v\lambda_y) + 1)F_{4,u} = c(\omega - 1)B, \\
&\omega B = D + \frac{F_{1,x} + F_{2,x} + F_{3,x} + F_{4,x}}{8} + \frac{F_{1,y} + F_{2,y} + F_{3,y} + F_{4,y}}{8},
\end{aligned} \tag{33}$$

leading to

$$\begin{aligned}
&F_{1,u} + F_{2,u} + F_{3,u} + F_{4,u} = 2c(\omega - 1)B \times \Big(\frac{1}{1 + (k_u\lambda_x + k_v\lambda_y)^2} + \frac{1}{1 + (k_u\lambda_x - k_v\lambda_y)^2} \Big), \\
&\quad u = x, v = y \text{ or } u = y, v = x.
\end{aligned} \tag{34}$$

To get the optimal spectral radius, weighting coefficients (k_x, k_y) are obtained. Weighting functions are also evaluated to satisfy this weighting coefficients.

$$k_x = 0.8, \ k_y = 0.2, \quad W_x = 2.4(|\mu_n| - |\eta_n|) + 1, \ W_y = -2.4(|\mu_n| - |\eta_n|) + 1, \quad \rho < 0.1275c, \tag{35}$$

We find that the directional S_2-like weighting functions help to converge faster that the normal S_2-like weighting in AADR.

Finally, the spectral radius (ρ) is expressed in a general form as

$$\rho = sup_{\lambda_x,\lambda_y}\Big|\frac{c}{4}\sum_n w_n \frac{1-K(\lambda_x,\lambda_y)(\mu_n\lambda_x+\eta_n\lambda_y)^2}{1+(\mu_n\lambda_x+\eta_n\lambda_y)^2}\Big|, \tag{36}$$

where $K(\lambda_x,\lambda_y)$ represents a weighting parameter and it is given in different forms depending on the methods.

$$\begin{aligned}
&\text{(i) SI}: \quad K(\lambda_x,\lambda_y)=0,\\
&\text{(ii) DSA}: \quad K(\lambda_x,\lambda_y)=\frac{1}{\mathrm{\lambda_x^2/3+\lambda_y^2/3}},\\
&\text{(iii) AADR1}: \quad K(\lambda_x,\lambda_y)=\frac{\mathrm{1+(k_1\lambda_x)^2+(k_2\lambda_y)^2}}{\mathrm{(k_1\lambda_x)^2+(k_2\lambda_y)^2+((k_1\lambda_x)^2-(k_2\lambda_y)^2)^2}},\\
&\text{(iv) AADR2}: \quad K(\lambda_x,\lambda_y)=\frac{1}{2}\frac{\mathrm{B(C+2D-1)+D(A+2B-1)}}{\mathrm{(A+2B-1)(C+D-1)+(C+2D-1)(A+B-1)}},
\end{aligned} \tag{37}$$

where

$$\begin{aligned}
&A=((k_x\lambda_x)^2-(k_y\lambda_y)^2)^2, \quad B=1+(k_x\lambda_x)^2+(k_y\lambda_y)^2,\\
&C=((k_y\lambda_x)^2-(k_x\lambda_y)^2)^2, \quad D=1+(k_y\lambda_x)^2+(k_x\lambda_y)^2.
\end{aligned} \tag{38}$$

The spectral radii of various methods are then given as

$$\begin{aligned}
&\text{(i) SI}: \quad \rho=\mathrm{c},\\
&\text{(ii) DSA}: \quad \rho<0.2247\mathrm{c},\\
&\text{(iii) AADR1}: \rho<0.4469\mathrm{c}\ (\text{when } \mathrm{k_1=k_2=0.7}),\\
&\text{(iv) AADR2}: \rho<0.1275\mathrm{c}\ (\text{when } \mathrm{k_x=0.8,\ k_y=0.2}).
\end{aligned} \tag{39}$$

Here, AADR1 denotes AADR with normal S_2-like rebalance and AADR2 denotes AADR with directional S_2-like rebalance. The spectral radius of AADR2 is much less than those of AADR1 and DSA. Considering directional S_2-like rebalance factors, AADR2 may need slightly longer computing time for low-order equation than AADR1 but it is compensated by fast convergence with reduced spectral radius. The preconditioned Bi-CGSTAB algorithm used also reduces computational burden in AADR. The Bi-CGSTAB method is a variation of the conjugate gradient square (CGS) method which was developed to remedy the substantial buildup of rounding errors or possibly even overflow of the CGS method in cases of irregular convergence and it does not require a transpose of the operator.

The low-order operator ($\mathbf{T}$) is defined as

$$\mathbf{T}\vec{\phi}=(\mathbf{I}-\mathbf{M})\vec{\phi}=\mathbf{M}_s\vec{s}, \tag{40}$$

and rewritten as

$$\mathbf{T}\vec{\phi}=\vec{b}, \quad \vec{b}=\mathbf{M_s}\vec{s}. \tag{41}$$

The linear operator $\mathbf{M}$ representing sweeping with the scattering source is given by

$$\begin{aligned}
\vec{f}_\gamma^{out}&=\mathbf{U}_\gamma\vec{f}_\gamma^{in}+\mathbf{V}_\gamma(\sigma_s\vec{x}),\\
\vec{x}_\gamma&=\mathbf{X}_\gamma\vec{f}_\gamma^{in}+\mathbf{Y}_\gamma(\sigma_s\vec{x}),\ \gamma=1,2,3,4,\\
\mathbf{M}\vec{x}&=\sum_{\gamma=1}^{4}\vec{x}_\gamma.
\end{aligned} \tag{42}$$

The preconditioned matrix **B** is determined as the following "transport sweep" incomplete factorization. For general cases,

$$\begin{aligned} \mathbf{T}\vec{x} &= (\mathbf{I} - \mathbf{M})\vec{x}, \\ \mathbf{M}\vec{x} &= \sum_{\gamma=1}^{4} \vec{x}_\gamma, \end{aligned} \tag{43}$$

where

$$\begin{aligned} \vec{x}_\gamma &= (\mathbf{X}_\gamma(\mathbf{I} - \mathbf{U}_\gamma)^{-1}\mathbf{V}_\gamma + \mathbf{Y}_\gamma)\vec{x}, \\ &= \mathbf{G}_\gamma^{-1}\vec{x}, \quad \gamma = 1, 2, 3, 4. \end{aligned} \tag{44}$$

Thus, we get the original operator **T** such as

$$\mathbf{T}\vec{x} = (\mathbf{I} - \mathbf{G}_1^{-1} - \mathbf{G}_2^{-1} - \mathbf{G}_3^{-1} - \mathbf{G}_4^{-1})\vec{x}. \tag{45}$$

We take an incomplete factorization matrix from **T** as a preconditioner **B**.

$$\mathbf{B}\vec{x} = (\mathbf{I} - \mathbf{G}_1^{-1})(\mathbf{I} - \mathbf{G}_2^{-1})(\mathbf{I} - \mathbf{G}_3^{-1})(\mathbf{I} - \mathbf{G}_4^{-1})\vec{x}. \tag{46}$$

It can be solved in the same way as the standard LU decomposition problem.

$$\begin{aligned} (\mathbf{I} - \mathbf{G}_1^{-1})\vec{z} = \vec{y}, &\quad (\mathbf{G}_1 - \mathbf{I})\vec{z} = \mathbf{G}_1\vec{y}, \\ (\mathbf{I} - \mathbf{G}_2^{-1})\vec{v} = \vec{z}, &\quad (\mathbf{G}_2 - \mathbf{I})\vec{v} = \mathbf{G}_2\vec{z}, \\ (\mathbf{I} - \mathbf{G}_3^{-1})\vec{u} = \vec{v}, &\quad (\mathbf{G}_3 - \mathbf{I})\vec{u} = \mathbf{G}_3\vec{v}, \\ (\mathbf{I} - \mathbf{G}_4^{-1})\vec{x} = \vec{u}, &\quad (\mathbf{G}_4 - \mathbf{I})\vec{x} = \mathbf{G}_4\vec{u}. \end{aligned} \tag{47}$$

4. Numerical Tests and Results

The first test problem is the McCoy-Larsen problem which is homogeneous and simple as shown in Fig. 2. It consists of a uniform, isotropically scattering with a scattering ratio(c) of 1. The problem is divided into 8×8 meshes and the total cross section σ is varied (i.e., $0.01 \leq \sigma \leq 6$). Table I shows the number of iterations and computing times. The criterion for average scalar flux is given 10^{-4} and S_8 quadrature is used. The results of AADR1 with normal S_2-like weighting functions indicate that it needs more iterations and takes slightly longer computing time. But AADR2 with directional S_2-like weighting functions is competitive with or outperform DSA, that are consistent with the results of Fourier analysis.

The second test is heterogeneous iron-water benchmark problem[9]. This problem is a diagonally symmetric, isotropic scattering $30cm \times 30cm$ rectangular box as shown in Fig. 3. Table II shows the material properties of iron-water benchmark problem. We solve it with S_8 angular quadrature set, 10×10 mesh division and a convergence criterion of 10^{-4}. The results for AADR and DSA methods are given in Table III. Unfortunately, DSA method in DANTSYS does not converge, which may result from negativeness of diamond difference scheme. So we compare the solutions of nodal methods (C-C, C-L) with those of LMB for AADR. To get the optimal convergence, a weighting functions is selected as $W_x = 0.6(|\mu| - |\eta|) + 1$, $W_y =$

$-0.6(|\mu|-|\eta|)+1$. The computing time and number of iterations shows that the LMB scheme with AADR provides good enough results.

5. Conclusions

We have described the additive angular dependent angular factor rebalance (AADR) method to accelerate the neutron transport equations in rectangular geometry. From Fourier analysis, we know that the normal S_2-like weighted AADR method is not competitive with DSA methods. But directional S_2-like weighted AADR method gives good results from numerical tests as well as from Fourier analysis. We also know that AADR methods strongly depend on weighting functions and suitable weighting functions provide an optimal convergence of given problem. From continuous Fourier analysis, AADR with directional weighting method is unconditionally stable with spectral radius $< 0.1275c$ when a proper weighting function is used. As a concluding remark, the additive angular dependent rebalance (AADR) acceleration method is useful and can offer various advantages over the existing methods.

Acknowledgement

This work was supported in part by the Ministry of Science and Technology of Korea through the National Research Laboratory (NRL) Program.

References

[1] R.E. Alcouffe, "Diffusion Synthetic Acceleration Methods for the Diamond-Differenced Discrete-Ordinates Equations," *Nucl. Sci. Eng.*, **64**, 344 (1977).

[2] E.W. Larsen, "Unconditionally Stable Diffusion Synthetic Acceleration Method for the Slab Geometry Discrete Ordinates Equations. Part I : Theory," *Nucl. Sci. Eng.*, **82**, 47 (1982).

[3] M.L. Adams and W.R. Martin, "Boundary Projection Acceleration : A New Approach to Synthetic Acceleration of Transport Calculations," *Nucl. Sci. Eng.*, **100**, 177 (1988).

[4] G.L. Ramone and M.L. Adams, M. L., "A Transport Synthetic Acceleration Method for Transport Iterations," *Nucl. Sci. Eng.*, **125**, 257 (1997).

[5] S.G. Hong and N.Z. Cho, "A Rebalance Approach to Nonlinear Iteration for Solving the Neutron Transport Equations," *Ann. Nucl. Energy*, **24**, 147 (1997).

[6] S.G. Hong and N.Z. Cho, "Angle-Dependent Rebalance Factor Method for Nodal Transport Problems in X-Y Geometry," *Trans. Am. Nucl. Soc.*, **79**, 139 (1998).

[7] C.J. Park and N.Z. Cho, "A Linear Multiple Balance Method with High Order Accuracy for Discrete Ordinates Neutron Transport Equations," *Ann. Nucl. Energy*, 2000 (accepted).

[8] N.Z. Cho and C.J. Park, "An Additive Angular Dependent Rebalance Acceleration Method for Neutron Transport Equations," *Trans. Am. Nucl. Soc.*, 2001, to appear.

[9] H. Khalil, "A Nodal Diffusion Technique for Synthetic Acceleration of Nodal S_N Calculations," *Nucl. Sci. Eng.*, **90**, 263 (1985).

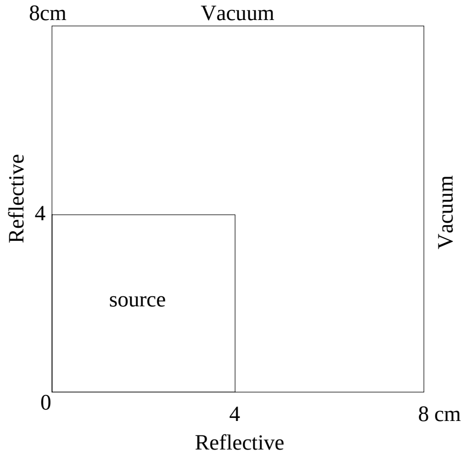

Figure 2: Configuration of 2D version of McCoy-Larsen problem.

Table I. Comparison of Number of Iterations and Computing Time[a] ($c = 1.0$)

σ	SI[b]		AADR1[c]		AADR2[d]		DSA[e]
	DD	LMB	DD	LMB	DD	LMB	DD
0.01	6[f] (0.06[g])	6 (0.09)	4 (0.08)	5 (0.19)	4 (0.10)	4 (0.12)	4 (0.99)
0.1	17 (0.14)	17 (0.23)	8 (0.17)	8 (0.41)	5 (0.14)	5 (0.17)	6 (1.04)
1.0	269 (1.88)	236 (2.91)	8 (0.28)	8 (0.41)	6 (0.27)	6 (0.33)	5 (1.01)
2.0	755 (5.29)	537 (6.58)	9 (0.43)	7 (0.43)	7 (0.43)	5 (0.32)	5 (1.01)
4.0	2038 (15.71)	1083 (13.40)	10 (0.65)	6 (0.38)	7 (0.51)	6 (0.37)	8 (1.08)
6.0	3473 (27.94)	1617 (20.15)	10 (0.72)	5 (0.35)	7 (0.59)	6 (0.38)	8 (1.08)

[a]: SUN-ULTRA1, [b]: Source iteration,
[c]: AADR1 with normal S_2-like weighting functions ($W = |\mu| + |\eta| - 0.5$),
[d]: AADR2 with directional S_2-like weighting functions
($W_x = 2.4(|\mu| - |\eta|) + 1,\ W_y = -2.4(|\mu| - |\eta|) + 1$),
[e]: DANTSYS code system,
[f]: Number of iterations, [g]: Computing time (sec).

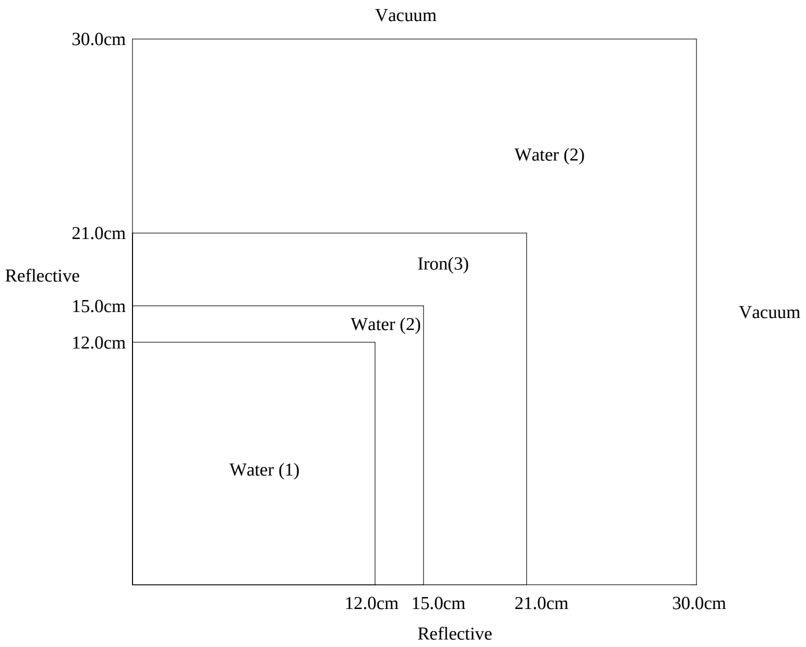

Figure 3: Configuration of the iron-water benchmark problem.

Table II. Material Properties for Iron-Water Benchmark Problem.

Composition	$\sigma(cm^{-1})$	c	Source Strength
1(water)	3.33	0.994	1.0
2(water)	3.33	0.994	0.0
3(iron)	1.33	0.831	0.0

Table III. Comparison of Number of Iterations and Computing Time.

Methods	AADR1[a]			AADR2[b]			DSA[c]
Schemes	LMB	C-C[d]	C-L[e]	LMB	C-C	C-L	DD
Number of Iterations	7	5	N.C.[f]	6	5	5	N.C.
Computing Time(sec)[g]	1.10	2.34	-	0.98	2.97	16.1	-

[a]: AADR1 with normal S_2-like weighting functions ($W = |\mu| + |\eta| - 0.5$),
[b]: AADR2 with directional S_2-like weighting functions
($W_x = 0.6(|\mu| - |\eta|) + 1,\ W_y = -0.6(|\mu| - |\eta|) + 1$),
[c]: DANTSYS code system, [d]: Constant-constant nodal scheme,
[e]: Constant-linear nodal scheme, [f]: Not converged, [g]: SUN-ULTRA1.