HYBRID S_N /DIFFUSION and S_N /P_3 NEUTRONICS CALCULATIONS

A Thesis

by

SERGIY MANOLOV

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Jim E. Morel
Committee Members,	Marvin L. Adams
	Raytcho Lazarov
	Jean C. Ragusa
Head of Department,	Yassin Hassan

May 2013

Major Subject: Nuclear Engineering

ABSTRACT

In this thesis we investigate coupling and preconditioning techniques for 1-D hybrid neutronics calculations. Each problem is represented by two spatial regions with Sn in one region and either Diffusion (P_1) or P_3 in the other region. For each of these two cases we define one coupling scheme and two different preconditioned systems. These systems are solved with both fixed-point iteration and the GMRES Krylov method. The solution techniques are compared in terms of iteration count and computational cost. Preconditioning with a global diffusion operator is found to be very effective for the most difficult problems.

DEDICATION

To my wife Anastasiia

ACKNOWLEDGEMENTS

I would like to thank my committee chair, Dr. Morel for his guidance throughout the course of this research. Thanks to my committee members, Dr. Adams, Dr. Ragusa and Dr. Lazarov for their time and support.

Thanks also to my friends and colleagues and the department faculty and staff for making my time at Texas A&M University a great experience.

Finally, thanks to my mother and father for their encouragement and to my wife for her patience and love.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

Very little previous work has been done with respect to coupling different neutronics methods in the same calculation and solving the resulting equations. The purpose of the present work is to investigate various strategies for coupling the Sn and Pn methods in 1-D slab geometry and solving the associated equations via preconditioned Krylov methods. We couple the Sn equations with the diffusion (P_1) and P_3 equations. The problem is represented by two spatial regions each of which uses a different approximation of the transport problem. The regions are connected and the exchange of information across the interface affects the solution in each region.

Each method for solving the neutron transport problem has its advantages and disadvantages. The Pn approximation is accurate and efficient for homogenized reactor calculations, but the discrete-ordinates (S_N) method or the method of characteristics (MOC) are preferred in non-homogenized calculations due to the rapid variation of the angular flux. A full-core non-homogenized treatment is currently not practical even with the latest computer systems. Thus it has been proposed [1, 2] that improved accuracy without a fully non-homogenized calculation might be obtained by homogenizing only certain assemblies while leaving others non-homogenized. In this case, improved efficiency might be obtained in a calculation by using Pn in homogenized assemblies and S_N or MOC in non-homogenized assemblies [1, 2]. Both Idaho National Laboratory and Argonne National Laboratory are developing neutronics codes that are intended to use this approach [3], and Idaho National Laboratory has sponsored this thesis research.

The method of discrete ordinates was developed for reactor applications in the late 50’s at Los Alamos and was well established by 1968 when Carlson and Lathrop documented the scheme as it existed at that time [4]. The concept of diffusion-synthetic acceleration was proposed in 1963 [5], but the synthetic method for discrete problems was originally seen to be unstable for some problems [6]. Later the instability was eliminated by making the spatial discretization of DSA consistent with the spatial discretization of the Sn equations. The application of the Pn method for 1-D radiative transfer was considered in late 40’s [7].

2. TRANSPORT APPROXIMATIONS AND DISCRETIZATIONS

In this section the diffusion (P_1), P_3 and Sn approximations for a steady-state neutron transport problem are described. A 1-D problem is simplified further by assuming uniform cross sections over the whole domain, mono-energetic particles, isotropic scattering and isotropic distributed source. The neutron transport problem we consider in this research can be presented by the following equation:

$$\mu\frac{\partial\psi(x,\mu)}{\partial x}+\sigma_t\psi(x,\mu)=\frac{\sigma_s(x)\varphi(x)}{4\pi}+\frac{Q(x)}{4\pi}, \tag{2.1}$$

where $\psi(x,\mu)$ is an angular flux and μ is the cosine that defines particle direction, σ_t is the total cross section, σ_s is the scattering cross section, and $\varphi(x)$ is the scalar flux:

$$\varphi(x)=2\pi\int_{-1}^{1}\psi(\mu,x)d\mu. \tag{2.2}$$

For the simplicity, we assume uniform cell size.

2.1 Diffusion Approximation

The Diffusion approximation implies that the angular flux $\psi(\vec{r},E,\Omega,t)$ has a linear dependence in direction, i.e. $\psi(\mu)=\frac{\Phi+3J\mu}{4\pi}$, where Φ is the scalar flux and J is the net current. This approximation in slab geometry is the same as the S_2 approximation with Gauss quadrature. The diffusion approximation works well for a medium without strong or localized absorbers and for materials where cross sections are constant or have weak spatial dependencies.

However, the diffusion approximation is not always adequate for nuclear power analysis. For example, it will produce high error if applied to calculate the distribution of a neutron field inside a fuel rod in a reactor core. In order to make it work for a reactor problem, the homogenization of some areas of the core needs to be performed, so the level of detail of the problem is reduced.

We solve the following diffusion equation in the diffusion region:

$$-\frac{\partial}{\partial x}D\frac{\partial\varphi(x)}{\partial x}+\ \sigma_a\varphi(x)=Q(x). \tag{2.3}$$

We use the cell-centered discretization for equation (2.3) with spatial indexing defined in Fig.1.

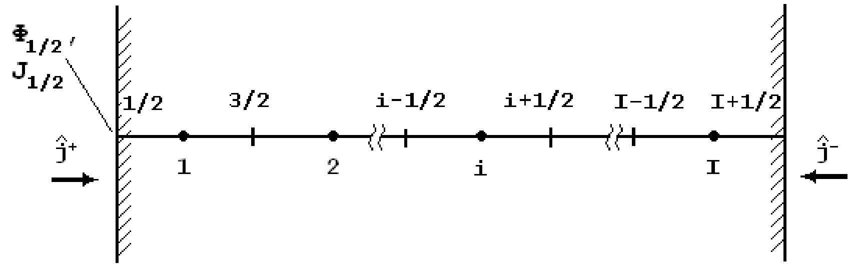

Fig.1. Spatial indexing for the diffusion approximation.

Note that $\Phi_{1/2}$ and $J_{1/2}$ lie on the left boundary. The balance equation for the i-th cell is given by the following equation:

$$J_{i+\frac{1}{2}}-J_{i-\frac{1}{2}}+\ \sigma_a h\,\varphi_i=\mathrm{q}_i h\,. \tag{2.4}$$

Here J is net current, h is cell size, σ_a is macroscopic absorption cross section and q_i is the distributed source at the i-th cell. Fick's law can be used to calculate the net currents:

$$J_{i+\frac{1}{2}} = -\frac{D}{h}(\varphi_{i+1} - \varphi_i). \tag{2.5}$$

For the interior cells, the balance equation expressed in terms of the scalar fluxes is

$$-\left(\frac{D}{h}\right)(\varphi_{i+1} - \varphi_i) + \left(\frac{D}{h}\right)(\varphi_i - \varphi_{i-1}) + \sigma_a h \varphi_i = q_i h. \tag{2.6}$$

At the left boundary, the boundary equation has the following form:

$$-\frac{D}{h}(\varphi_2 - \varphi_1) - J_{\frac{1}{2}} + \sigma_a \varphi_1 h = q_1 h\,. \tag{2.7}$$

We will use the Fick's law

$$J_{\frac{1}{2}} = -\frac{2D}{h}\left(\varphi_1 - \varphi_{\frac{1}{2}}\right), \tag{2.8a}$$

and the Marshak boundary conditions to obtain an expression for the net current at the boundary

$$J_{\frac{1}{2}} = \frac{1}{2}\left(4\hat{\jmath}^{+} - \varphi_{\frac{1}{2}}\right). \tag{2.8b}$$

Here $\hat{\jmath}^{+}$ is the incoming half-range current which is set to zero in the case of a vacuum boundary condition. We obtain an equation for $\varphi_{\frac{1}{2}}$ by equating (2.8a) and (2.8b):

$$-\frac{2D}{h}\left(\varphi_1 - \varphi_{\frac{1}{2}}\right) = \frac{1}{2}\left(4\hat{\jmath}^{+} - \varphi_{\frac{1}{2}}\right). \tag{2.9}$$

From the equation (2.9) follows:

$$\varphi_{\frac{1}{2}} = \frac{\left(\frac{2D}{h}\right)\varphi_1 + 2\hat{J}^+}{\frac{1}{2} + \left(\frac{2D}{h}\right)}. \tag{2.10}$$

Substituting the expression for $\varphi_{\frac{1}{2}}$ back into either (2.8b) or (2.8a), we obtain the expression for $J_{\frac{1}{2}}$:

$$J_{\frac{1}{2}} = \frac{2D(4\hat{J}^+ - \varphi_1)}{\mathrm{h} + 4D}. \tag{2.11}$$

For the first cell, the balance equation becomes:

$$-\frac{D}{h}(\varphi_2 - \varphi_1) + \left(\frac{2D}{h + 4D}\right)\varphi_1 + \sigma_a \mathrm{h}\,\varphi_1 = q_1 h + \frac{8D\hat{J}^+}{\mathrm{h} + 4D}. \tag{2.12}$$

The same approach is used to obtain the equation for the right boundary.

2.2 P$_3$ Approximation

The P$_n$ approximation assumes that the angular flux can be represented as a superposition of spherical harmonics or Legendre polynomials for the case of slab geometry. In plane geometry, the angular flux for a P$_3$ approximation is represented as follows:

$$\psi(\mu, x) = \sum_{n=0}^{3} \frac{2n+1}{4\pi} P_n(\mu)\phi_n(x), \tag{2.13}$$

where P$_n$(μ) is the Legendre polynomial of degree n and ϕ$_n$(*x*) is the n-th moment of the angular flux. The steady-state P$_1$ equations are equivalent to diffusion theory.

The P$_3$ equations for 1-D slab geometry have the form:

$$\left(\frac{k+1}{2k+1}\right)\varphi'_{k+1} + \left(\frac{k}{2k+1}\right)\varphi'_{k-1} + \beta_k\varphi_k = 0, \qquad k = 0,..,3. \tag{2.14}$$

For a problem with isotropic scattering, the coefficient β_k can be presented as

$$\beta_k = \begin{cases} \sigma_a & \text{if } \mathrm{k} = 0, \\ \sigma_t & \text{if } \mathrm{k} > 0. \end{cases}$$

Solving (2.14) for the P$_3$ odd moments, we obtain the expressions

$$\varphi^{(1)}(x) = -\frac{1}{\sigma_t}\left[\frac{2}{3}\cdot\frac{d}{dx}\varphi^{(2)}(x) + \frac{1}{3}\cdot\frac{d}{dx}\varphi^{(0)}(x)\right] \tag{2.15}$$

and

$$\varphi^{(3)}(x) = -\frac{1}{\sigma_t}\left[\frac{3}{7}\cdot\frac{d}{dx}\varphi^{(2)}(x)\right]. \tag{2.16}$$

The P$_3$ equations are put in second-order form by substituting expressions from (2.15) and (2.16) into the equations (2.14) for even *k*. The expressions for the zeroth and second moments are:

$$\frac{1}{\sigma_t}\left[\frac{2}{3}\frac{d^2}{dx^2}\varphi^{(2)}(x) + \frac{1}{3}\frac{d^2}{dx^2}\varphi^{(0)}(x)\right] + \sigma_a\varphi^{(0)}(x) = q^{(0)} \tag{2.17}$$

and

$$-\frac{3}{5\sigma_t}\left[\frac{3}{7}\frac{d^2}{dx^2}\varphi^{(2)}(x)\right] - \frac{2}{5\sigma_t}\left[\frac{2}{3}\frac{d^2}{dx^2}\varphi^{(2)}(x) + \frac{1}{3}\frac{d^2}{dx^2}\varphi^{(0)}(x)\right] + \sigma_t\varphi^{(2)}(x) = 0. \tag{2.18}$$

In this research, we use edge-centered discretization for our P$_3$ equations. The spatial indexing is shown on Fig. 2:

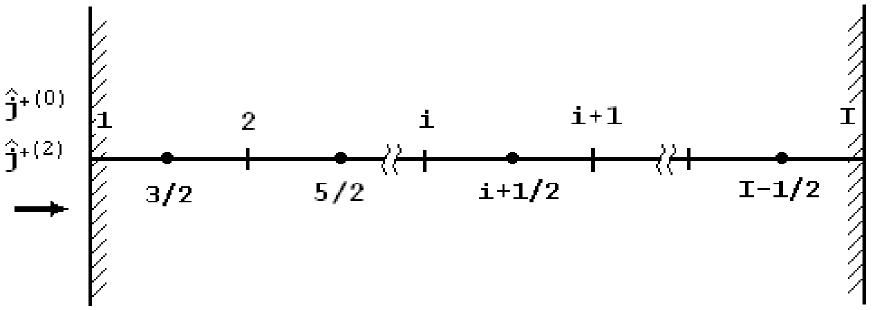

Fig.2. Spatial indexing for P$_3$ problem.

We define $h_t = \sigma_t h$, $h_s = \sigma_s h$. We will use the central-differencing approximation of a spatial derivative:

$$\frac{d}{dx}\phi^{(l)}_{i+1/2} = \frac{1}{h}\left(\phi^{(l)}_{i+1} - \phi^{(l)}_{i}\right) \text{ and } \frac{d}{dx}\phi^{(l)}_{i} = \frac{1}{h}\left(\phi^{(l)}_{i+1} - \phi^{(l)}_{i-1}\right).$$

After spatial discretization, the equations (2.17) and (2.18) on the mesh interior become

$$-\frac{1}{3}\left(\frac{\varphi^{(0)}_{i+1} - 2\varphi^{(0)}_{i} + \varphi^{(0)}_{i-1}}{h_t}\right) - \frac{2}{3}\left(\frac{\varphi^{(2)}_{i+1} - 2\varphi^{(2)}_{i} + \varphi^{(2)}_{i-1}}{h_t}\right) + h_a\varphi^{(0)}_{i} = q_0 h\,, \quad (2.19)$$

and

$$-\frac{11}{35}\left(\frac{\varphi^{(2)}_{i+1} - 2\varphi^{(2)}_{i} + \varphi^{(2)}_{i-1}}{h_t}\right) - \frac{2}{15}\left(\frac{\varphi^{(0)}_{i+1} - 2\varphi^{(0)}_{i} + \varphi^{(0)}_{i-1}}{h_t}\right) + h_t\varphi^{(2)}_{i} = 0, \quad (2.20)$$

and the equations (2.15) and (2.16) become

$$\varphi^{(1)}(x) = -\frac{1}{h_t}\left[\frac{2}{3}\cdot\left(\varphi^{(2)}_{i+1} - \varphi^{(2)}_{i-1}\right) + \frac{1}{3}\cdot\left(\varphi^{(0)}_{i+1} - \varphi^{(0)}_{i-1}\right)\right] \quad (2.21)$$

and

$$\varphi^{(3)}(x) = -\frac{1}{h_t}\left[\frac{3}{7}\cdot\frac{d}{dx}\left(\varphi^{(2)}_{i+1} - \varphi^{(2)}_{i-1}\right)\right]. \quad (2.22)$$

The boundary conditions for the P_{N-1} approximation are usually different from the S_N approximation and more complex. For a Pn approximation, there are (n+1) unknowns and (n+1) boundary conditions are needed. The Marshak boundary conditions require preservation of half-range current. For P_3, the Marshak boundary conditions can be expressed as follows at the left face of the slab:

$$2\pi \int_0^{+1} \left(\sum_{n=0}^{3} \frac{2n+1}{4\pi} \phi_1^{(n)} P_n(\mu) \right) \mu P_k(\mu) d\mu = \hat{\jmath}^{+\,(k)}, \qquad k = 0, 2. \tag{2.23}$$

Here $\hat{\jmath}^{+,(0)}$ is positive half-range current and $\hat{\jmath}^{+,(2)}$ is so-called second-order half-range current. For vacuum boundary conditions, we set these currents to zero. On the right boundary, the boundary conditions are similar. For a reflective boundary, the odd flux moments are set to zero.

Since the solution for a P_3 problem is acquired in terms of moments $\vec{\phi} = \left(\phi^{(0)}, \phi^{(1)}, \phi^{(2)}, \phi^{(3)}\right)^T$, four equations are needed to solve for the moments on the left boundary. Two of these equations are the zeroth and second moment equations in the first half-cell:

$$\phi_{3/2}^{(1)} - \phi_1^{(1)} + \left(\frac{\sigma_a\, h}{2}\right) \phi_1^{(0)} = \frac{q^{(0)}\, h}{2} \tag{2.24}$$

and

$$\frac{3}{5} \cdot \left[\phi_{3/2}^{(3)} - \phi_1^{(3)}\right] + \frac{2}{5} \cdot \left[\phi_{3/2}^{(1)} - \phi_1^{(1)}\right] + \left(\frac{\sigma_a\, h}{2}\right) \phi_1^{(2)} = 0. \tag{2.25}$$

The expressions for the odd moments in the center of the first cell are:

$$\phi_{3/2}^{(1)} = -\frac{1}{h_t}\left[\left(\frac{2}{3}\right)\left(\phi_2^{(2)} - \phi_1^{(2)}\right) + \left(\frac{1}{3}\right)\cdot\left(\phi_2^{(0)} - \phi_1^{(0)}\right)\right], \tag{2.26}$$

and

$$\phi_{3/2}^{(3)} == -\frac{3}{7h_t}\left(\phi_2^{(2)} - \phi_1^{(2)}\right). \tag{2.27}$$

For reasons explained later, we use Sn quadrature to calculate half-range currents. After discretization, equation (2.23) becomes:

$$\sum_{\mu_m>0}\left(\sum_{l=0}^{3}\left(\frac{2l+1}{4\pi}\right)\phi_1^{(l)}P_l(\mu_m)\right)P_k(\mu_m)\mu_m w_m = \hat{\jmath}^{+(k)}, \qquad k = 0, 2. \tag{2.28}$$

These equations can be presented in the following form:

$$\sum_{l=0}^{3}\left(\frac{2l+1}{4\pi}\right)\phi_1^{(l)}a_{kl} = \hat{\jmath}^{+(k)},\ \ k = 0, 2, \tag{2.29}$$

where

$$a_{kl} = \sum_{\mu_m>0} P_l(\mu_m)\mu_m P_k(\mu_m)w_m,\ \ l = 0,..,3,\ \ k = 0, 2.$$

Using these notations, the equations (2.35) can be reexpressed as:

$$\begin{pmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{20} & a_{21} & a_{22} & a_{23} \end{pmatrix}\cdot\begin{pmatrix} \phi_1^{(0)} \\ \phi_1^{(1)} \\ \phi_1^{(2)} \\ \phi_1^{(3)} \end{pmatrix} = \begin{pmatrix} \hat{\jmath}^{+(0)} \\ \hat{\jmath}^{+(2)} \end{pmatrix}. \tag{2.30}$$

We can rearrange (2.30) as follows:

$$\begin{pmatrix} a_{01} & a_{03} \\ a_{21} & a_{23} \end{pmatrix} \begin{pmatrix} \phi_1^{(1)} \\ \phi_1^{(3)} \end{pmatrix} = \begin{pmatrix} \hat{j}^{+(0)} \\ \hat{j}^{+(2)} \end{pmatrix} - \begin{pmatrix} a_{00} & a_{02} \\ a_{20} & a_{22} \end{pmatrix} \begin{pmatrix} \phi_1^{(0)} \\ \phi_1^{(2)} \end{pmatrix}. \tag{2.31}$$

Expression (3.31) can be represented as:

$$\boldsymbol{A}\vec{\phi}_{odd} = \vec{j} + \boldsymbol{B}\vec{\phi}_{even}, \tag{2.32}$$

where

$$\boldsymbol{A} \equiv \begin{pmatrix} a_{01} & a_{03} \\ a_{21} & a_{23} \end{pmatrix}, \qquad \boldsymbol{B} \equiv - \begin{pmatrix} a_{00} & a_{02} \\ a_{20} & a_{22} \end{pmatrix},$$

$$\vec{j} = \begin{pmatrix} \hat{j}^{+(0)} \\ \hat{j}^{+(2)} \end{pmatrix}, \qquad \vec{\phi}_{even} = \begin{pmatrix} \phi_1^{(0)} \\ \phi_1^{(2)} \end{pmatrix}, \qquad \vec{\phi}_{odd} = \begin{pmatrix} \phi_1^{(1)} \\ \phi_1^{(3)} \end{pmatrix}. \tag{2.33}$$

We can solve (2.32) for the odd moments

$$\vec{\phi}_{odd} = \boldsymbol{A^{-1}}\left(\vec{j} + \boldsymbol{B}\vec{\phi}_{even}\right). \tag{2.34}$$

Finally, we can substitute from (2.34) into (2.24) and (2.25) to close the equations for $\phi_1^{(0)}$ and $\phi_1^{(2)}$. The right boundary is treated in a similar manner.

As n → ∞, the Pn approximation converges to the solution of the corresponding transport problem. The S_N approximation with Gauss quadrature and the Pn approximation are equivalent for slab geometry. However, they are not equivalent for multi-D geometries.

2.3 S$_N$ Approximation

The S$_N$ approximation implies that there are N discrete directions in which the particles travel. These directions correspond to those of an angular quadrature $\{\mu_m, w_m\}_{m=1}^{N}$.

In slab geometry, the S$_N$ approximation with isotropic scattering and sources produces the following equation:

$$\mu_m \frac{\partial\psi(x,\mu_m)}{\partial x} + \sigma_t\psi(x,\mu_m) = \frac{\sigma_t\varphi(x)}{4\pi} + \frac{q\,(x)}{4\pi}. \tag{2.35}$$

All angular integrals are calculated using the quadrature. For instance, the scalar flux is given by:

$$\varphi(x) = \sum_{m=1}^{N} \psi_m(x) w_m. \tag{2.36}$$

All our calculations are performed using Gauss quadrature. The spatial grid for the discretized Sn problem is shown on the Fig.3:

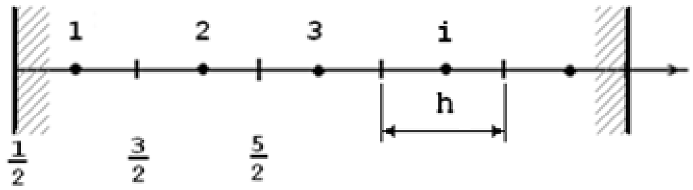

Fig.3. Spatial indexing for Sn problem.

After the spatial discretization, the balance equation for a single cell is:

$$\mu_m \left(\psi_{i+\frac{1}{2}} - \psi_{i-\frac{1}{2}}\right) + \sigma_t h\, \psi_i = \frac{\sigma_t \varphi_i h}{4\pi} + \frac{q_i h}{4\pi}. \tag{2.37}$$

There are two unknowns in equation (2.37): the average angular flux in the cell center ψ_i and the outflow angular flux at one of the edges. The inflow is known either from the solution in the upstream cell or from boundary conditions. The outflow is $\psi_{i-\frac{1}{2}}$ for μ<0 and $\psi_{i+\frac{1}{2}}$ for μ>0. To close the system of equations, we use the diamond difference expression:

$$\psi_i = \frac{1}{2}\left(\psi_{i+\frac{1}{2}} + \psi_{i-\frac{1}{2}}\right). \tag{2.38}$$

Substituting (2.38) into (2.37), we get

$$\mu_m \left(\psi_{i+\frac{1}{2}} - \psi_{i-\frac{1}{2}}\right) + \frac{\sigma_t h}{2}\left(\psi_{i+\frac{1}{2}} + \psi_{i-\frac{1}{2}}\right) = \frac{\sigma_s h}{8\pi}\left(\varphi_{i+\frac{1}{2}} + \varphi_{i-\frac{1}{2}}\right) + \frac{q_i}{4\pi}. \tag{2.39}$$

The S_N approximation is one of the most widely used approaches for solving the transport equation. The solution of the S_N equations converges to the exact solution of the corresponding transport problem as N→∞. However, the discrete ordinates approximation in a multi-D geometry can produce non-physical oscillations called ray effects, but this effect does not occur in 1-D steady-state calculations.

3. SOLUTION TECHNIQUES FOR INDIVIDUAL APPROXIMATIONS

In this section we describe the numerical solution techniques for the three individual approximations. The section also includes the Fourier analysis results for the P$_3$ Gauss-Seidel iterative scheme.

3.1 Diffusion Approximation

The diffusion equations are solved by the direct inversion of the tridiagonal matrix representing this system.

3.2 P$_3$ Approximation

The Gauss-Seidel scheme is used to solve the system of the P$_3$ equations for the even moments $\vec{\phi}^{(2)}$and $\vec{\phi}^{(0)}$. The solution for $\vec{\phi}^{(2)}$ is acquired first, then we solve for $\vec{\phi}^{(0)}$. The Gauss-Seidel iterative scheme used in this research proceeds as follows. We first solve the equations for the second moment lagging the zeroth moments. For the interior cell, the descretized interior equation with iteration indices has the following form:

$$-\left[\frac{11}{35\cdot\sigma_t}\right]\left(\frac{\varphi_{i+1}^{2.(l+1)}-2\varphi_i^{2.(l+1)}+\varphi_{i-1}^{2.(l+1)}}{h}\right) \\ -\left[\frac{2}{15\cdot\sigma_t}\right]\left(\frac{\varphi_{i+1}^{0.(l)}-2\varphi_i^{0.(l)}+\varphi_{i-1}^{0.(l)}}{h}\right)+\sigma_t h\varphi_i^{2.(l+1)}=0, \tag{3.1}$$

where *l* is the iteration index. After the solution for the second moment is obtained, the zeroth moment equations are solved:

$$-\frac{1}{3\sigma_t}\left(\frac{\varphi_{i+1}^{0.(l+1)}-2\varphi_i^{0.(l+1)}+\varphi_{i-1}^{0.(l+1)}}{h}\right)$$
$$-\frac{2}{3\sigma_t}\left(\frac{\varphi_{i+1}^{2.(l+1)}-2\varphi_i^{2.(l+1)}+\varphi_{i-1}^{2.(l+1)}}{h}\right)+h_a\varphi_i^{0.(l+1)} = \; q_0 h \tag{3.2}$$

For reasons explained later, it is important to note that the balance equation (3.2) is satisfied by the (l+1) solution iterate. The second moment is solved before the zeroth moment equation to specifically achieve conservation.

3.3 S$_\text{N}$ Approximation

We use the technique called source iteration to solve an Sn problem. Since the fluxes in different directions couple only on the right side of equation (2.42) and the operator on the left side of this equation is easy to invert, the equations are solved by lagging the scattering source. The iteration process for solving a 1-D discrete ordinates problem with isotropic scattering is described by the equation:

$$\mu_m\frac{\partial\psi_m^{(l+1)}}{\partial x}+\Sigma_\text{t}\psi_m^{(l+1)}=\frac{\sigma_\text{S}}{4\pi}\varphi^{(l)}+\frac{q_0}{4\pi}, \tag{3.3}$$

where *l* is the iteration index. In case of a fixed source problem, the convergence rate is dependent on the scattering ratio

$$c=\frac{\sigma_s}{\sigma_t}. \tag{3.4}$$

If the scattering ratio is close to unity, the convergence rate is low, and vice versa. For c = 0 it takes only one iteration to converge.

The solution of an S_N problem can be acquired by performing a series of forward and backward sweeps. The backward sweep consists of two loops – over the angles and over the cells. Before each iteration, the source is calculated using the following equation:

$$q_i = \frac{1}{4\pi}\sigma_s \cdot \varphi_i^{(l-1)} + \frac{Q}{4\pi}; \quad i = 1..N_{cells}. \tag{3.5}$$

The backward sweep computes the edge angular fluxes using the following equation:

$$\psi_{i-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}(n) = h \cdot q_i - \psi_{i+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left[\frac{\sigma_t h}{2} + \mu_n\right] \cdot \left[\frac{\sigma_t h}{2} - \mu_n\right]^{-1}. \tag{3.6}$$

Here the index ‘i’ refers to the cell number and ‘n’ refers to an angle. The calculations start at the right edge for $\mu<0$ and proceed to solve one cell and direction at a time until we reach the left edge. Similarly, we perform the forward sweep starting on the left edge of the domain and calculating the angular fluxes for $\mu>0$.

The equation for a forward sweep is the following:

$$\psi_{i+\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}(n) = h \cdot q_i - \psi_{i-\frac{1}{2}}^{\left(l+\frac{1}{2}\right)}\left[\frac{\sigma_t h}{2} + \mu_n\right] \cdot \left[\frac{\sigma_t h}{2} - \mu_n\right]^{-1}. \tag{3.7}$$

The scalar fluxes at the cell centers are calculated using the diamond difference expression from the scalar fluxes on the cell edges:

$$\varphi_i = \frac{1}{2}\left(\varphi_{i+\frac{1}{2}} + \varphi_{i-\frac{1}{2}}\right). \tag{3.8}$$

The scalar fluxes on the cell edges are computed by numerical integration of the discrete angular fluxes using the Gauss quadrature set.

Sweep is a relatively effective technique and doesn't consume too much of the memory because only angular fluxes for one edge and a center of a cell can be stored to calculate the fluxes on the other edge using diamond-difference relationship.

When source iteration is slow to converge, this situation can be improved by using the diffusion-based acceleration technique. Diffusion-Synthetic Acceleration (DSA) improves the situation and helps S_N to converge if the scattering ratio is near unity.

DSA uses diffusion approximation to estimate the error in the scalar flux after a source iteration:

$$\mu_m \frac{\partial \psi_m^{\left(l+\frac{1}{2}\right)}}{\partial x} + \sigma_t \psi_m^{\left(l+\frac{1}{2}\right)} = \frac{\sigma_S}{4\pi} \varphi^{(l)} + q_0, \tag{3.9}$$

$$-\frac{\partial}{\partial x} D \frac{\partial \delta\varphi^{\left(l+\frac{1}{2}\right)}}{\partial x} + \sigma_a \delta\varphi^{\left(l+\frac{1}{2}\right)} = \sigma_S \left(\varphi^{\left(l+\frac{1}{2}\right)} - \varphi^{(l)}\right). \tag{3.10}$$

After the DSA equations are solved and the error estimated, the scalar fluxes can be corrected:

$$\varphi^{(l+1)} = \varphi^{\left(l+\frac{1}{2}\right)} + \delta\varphi^{\left(l+\frac{1}{2}\right)}. \tag{3.11}$$

For the S_N interior, the corrective equations look like regular diffusion balance equations. Our equations are consistently derived from the Sn equations, which ensures stability and effectiveness of the algorithm with isotropic scattering. The spatial indexing is shown on the figure 2. In terms of cell-centered unknowns, the equation can be expressed as follows:

$$\delta J_i - \delta J_{i-1} + \frac{\sigma_a h}{2}(\delta\phi_{i-1} + \delta\phi_i) = \frac{h}{2}(q_{i-1} + q_i). \tag{3.12}$$

However, the fundamental scalar flux unknowns are located at the vertices:

$$\delta\phi_i = \frac{\delta\phi_{i+\frac{1}{2}} + \delta\phi_{i-\frac{1}{2}}}{2} \tag{3.13}$$

and

$$\delta J_i = -\frac{D}{h}\left(\delta\phi_{i+\frac{1}{2}} - \delta\phi_{i-\frac{1}{2}}\right). \tag{3.14}$$

Substituting these expressions into (3.12), we get

$$-\frac{D}{h}\left(\delta\phi_{i+\frac{1}{2}} - \delta\phi_{i-\frac{1}{2}}\right) + \frac{D}{h}\left(\delta\phi_{i-\frac{1}{2}} - \delta\phi_{i-\frac{3}{2}}\right)$$
$$+\frac{\sigma_a h}{4}\left(\delta\phi_{i-\frac{3}{2}} + 2\delta\phi_{i-\frac{1}{2}} + \delta\phi_{i+\frac{1}{2}}\right) \tag{3.15}$$
$$= \left(\frac{\sigma_s h}{2}\right)\left(\left[\varphi_i^{\left(l+\frac{1}{2}\right)} - \varphi_i^{(l)}\right] + \left[\varphi_{i-1}^{\left(l+\frac{1}{2}\right)} - \varphi_{i-1}^{(l)}\right]\right).$$

The balance equation at the left boundary is

$$\delta J_1 - \delta J_{\frac{1}{2}} + \frac{\sigma_{at} h}{2}\delta\phi_1 = \frac{1}{2}q_1. \tag{3.16a}$$

Assuming a vacuum condition on the left boundary:

$$\frac{\delta\phi_{\frac{1}{2}}\langle\mu\rangle}{2} + \frac{\delta J_{\frac{1}{2}}}{2} = 0. \tag{3.16b}$$

From (3.14) and (3.16) it follows that

$$\left(\frac{\mathrm{D_T}}{\mathrm{h}}+\langle\mu\rangle\right)\delta\phi_{\frac{1}{2}}-\left(\frac{\mathrm{D_T}}{\mathrm{h}}\right)\delta\phi_{\frac{3}{2}}+\frac{\sigma_a \mathrm{h}}{4}\left(\delta\phi_{\frac{1}{2}}+\delta\phi_{\frac{3}{2}}\right)=\left(\frac{\sigma_s \mathrm{h}}{4}\right)\left(\varphi_1^{\left(l+\frac{1}{2}\right)}-\varphi_1^{(l)}\right). \quad (3.17)$$

The boundary equation for the right boundary is derived similarly.

In general the diffusion boundary conditions depend upon the transport boundary conditions. Assuming the boundary conditions are met by the Sn solution after each iteration, the diffusion conditions for a transport boundary source or vacuum conditions is vacuum and the diffusion conditions for a transport reflective conditions is reflective.

Using the corrective equations on the interior and boundaries, the tridiagonal matrix is composed and inverted to acquire a vector of corrections for the scalar fluxes at cell edges. After the DSA correction for the current iteration is known, the following steps follow:

1) Calculate correction in the cell centers (Diamond Differencing): $\delta\vec{\varphi}_i$.

2) Update $\vec{\varphi}_i^{(l+1)} = \vec{\varphi}^{\left(l+\frac{1}{2}\right)} + \delta\vec{\varphi}_i$.

3) Update the outgoing angular fluxes on the boundaries.

The implementation of the S_N scheme with DSA can be verified by checking the balance. The balance must be zero after each performed iteration. Without DSA, the S_N scheme guarantees zero balance only after convergence.

Diffusion-synthetic acceleration is a very effective way to accelerate convergence. A loss of effectiveness of the diffusion synthetic acceleration has been observed under certain conditions [1]. For example, for a multidimensional problem, the effectiveness of the DSA decreases if a strong discontinuity of the material properties is present in the medium. The effectiveness also decreases as the scattering becomes increasingly forward-peaked. Whether or not the DSA is effective for a particular problem depends on the number of dimensions, discretization of S_N and DSA and material properties.

3.4 Fourier (Von Neumann) Stability Analysis

There are several methods for the stability analysis of numerical iterate solution techniques for differential equations.

1) Fourier (Von Neumann) expansion method.

2) Eigenfunction method.

3) Matrix method.

4) Modified equation method.

These methods can be applied to PDEs of parabolic and hyperbolic types.

The Fourier expansion method is most widely used [4]. In this method, stability of the iterative method is investigated by expanding the error in a Fourier series. For the Fourier analysis to be applicable for the particular PDE, the following conditions must be met:

1) The PDE is linear.

2) The domain of interest is infinite.

3) The grid spacing is constant.

4) The PDE has constant coefficients.

In the Fourier stability analysis, the actual boundary conditions can be ignored. The source term of the PDE is also ignored because it does not arise in the error equation. If the error increases, the scheme is unstable. As mentioned above, the Fourier stability analysis can be applied to a parabolic or hyperbolic type PDE under the stated conditions.

The Fourier analysis for the P_3 schemes serves several purposes. The main purpose of the Fourier analysis is to determine the spectral radius of a particular scheme, which is the factor by which the error is asymptotically multiplied per single iteration. This can be useful in a case when the decision has to be made whether a particular scheme is worth implementing. For example, in the course of this work the results of Fourier analysis were obtained for the P_3 Gauss-Seidel scheme.

The Fourier analysis can be used to verify whether or not the scheme is implemented correctly. After the scheme is implemented, the numerical estimate for the spectral radius $\rho_i \equiv \frac{\sum \varphi_j^{(i+1)} - \sum \varphi_j^{(i)}}{\sum \varphi_j^{(i)} - \sum \varphi_j^{(i-1)}}$ must converge to the spectral radius. Here 'i' denotes the iteration index and 'j' refers to a cell number. By measuring ρ_i after each iteration, it can be determined whether or not it converges to the value obtained from the Fourier analysis. If it doesn't, either the Fourier analysis was performed incorrectly, or the implementation contains an error. However, it has to be noted that ρ_i has to be measured with random initial guess and no distributed sources present. Otherwise it might not converge before inaccuracies arise due to lack of precision. The measured spectral radii for the implemented P_3 Gauss-Seidel scheme are compared with those predicted by the Fourier Analysis in the Appendix A.

3.5 Krylov Methods

A Krylov method can be used to solve a linear system $\boldsymbol{A}\vec{x} = \vec{b}$. The matrix **A** is usually (but not necessarily) sparse. All Krylov methods are iterative methods, but they are not of the fixed-point type. A detailed description of the theory behind Krylov methods is beyond the scope of this thesis. It is sufficient for our purposes to describe how one uses a Krylov subroutine to solve $\boldsymbol{A}\vec{x} = \vec{b}$. The user first provides a Krylov routine with the source vector and an initial guess for the solution. At the beginning of each iteration the Krylov routine sends a vector to the user. Let us denote this vector by $\vec{z}^{\,(l)}$ where l is the iteration index. The user must then provide the Krylov routine with the vector $\vec{y} = \boldsymbol{A}\vec{z}^{\,(l)}$. This vector is called the "action" of $\boldsymbol{A}$ on $\vec{z}^{\,(l)}$. The Krylov routine then produces an approximation to the solution of the equation at the end of each iteration and checks for convergence.

In many cases, the Krylov iterations converge very slowly. This often happens in case of large real-world problems. For this reason, the systems solved with a Krylov method are often preconditioned. The problem $\boldsymbol{A}\vec{x} = \vec{b}$ is replaced by $\boldsymbol{C}\boldsymbol{A}\vec{x} = \boldsymbol{C}\vec{b}$ (Left preconditioner) or $\boldsymbol{A}\boldsymbol{C}\vec{y} = \vec{b}, \vec{x} = \boldsymbol{C}\vec{y}$ (Right preconditioner). In our research, we use only left preconditioning and the Krylov solver is used to solve all our preconditioned systems of transport equations. Krylov space methods are included in the list of the 10 most important classes of numerical methods.

Essentially any fixed-point iteration scheme can be represented as follows:

$$\boldsymbol{A}\vec{x}^{l+1} = \boldsymbol{B}\vec{x}^{l} + \vec{q},$$

where **A** and **B** are matrices or operators and $\vec{x}^{l}$ is the solution after the l-th iteration. The vector of unknowns can be found as follows

$$\vec{x}^{l+1} = \boldsymbol{A}^{-1}\boldsymbol{B}\vec{x}^{l} + \boldsymbol{A}^{-1}\vec{q}.$$

The preconditioned system corresponding to the iteration scheme is

$$(I - \boldsymbol{A}^{-1}\boldsymbol{B})\vec{x} = \boldsymbol{A}^{-1}\vec{q}.$$

It is important to note that obtaining the action of the preconditioned system on a vector is very similar to performing a fixed-point iteration. This is why using a Krylov method to solve a preconditioned system corresponding to a fixed-point iteration scheme is often referred to as "wrapping" the iteration scheme in a Krylov method. The details are described in the next section. In the course of this work, the Matlab Generalized Minimum Residual Method solver was used to solve all systems solved via a Krylov method.

3.5.1 Adapting Point Iterative Methods for Krylov Solver

When adapting a particular point iterative scheme for the Krylov solver, the following principles must apply:

1. The Krylov vector chosen for a particular fixed-point iteration scheme need not necessarily be the full vector of unknowns associated with the fixed-point iteration

scheme. The minimal vector is that for which the full solution can be reconstructed from the data it contains in a single fixed-point iteration. For example, for the S_N-diffusion scheme where S_N fully converges before the diffusion solver is called, the Krylov vector contains only positive half-range current and the interface scalar flux. However, once the Krylov solver provides the converged solution for these two quantities, a single fixed-point iteration scheme must be performed to obtain the full solution.

2. The source vector for the Krylov solver $\vec{q}$ is acquired by performing a single iteration of the scheme with the distributed sources present in the problem and zero initial guess for all other parameters.
3. The action of the preconditioned operator on a Krylov vector is obtained into two steps. First one performs a fixed point iteration with the Krylov vector playing the role of the previous iterate and the distributed source set to zero. Then subtracts the "new" iterate from the Krylov vector.

4. COUPLING TECHNIQUES

In this section, we discuss techniques for coupling different methods to solve a transport problem.

4.1. Location of the Unknowns

For the purpose of this research work, we consider 1D slab geometry. The medium consists of two separate regions. Each region has uniform properties which do not vary within that region as shown on Fig.4:

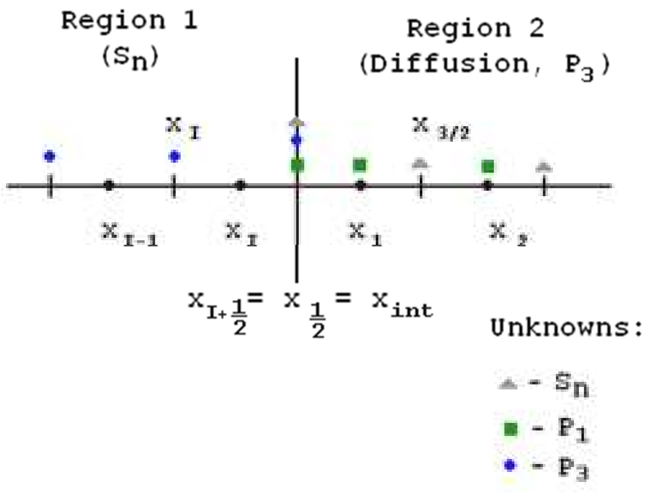

Fig.4. Geometry of the problem.

Different numerical schemes are applied to achieve solution in each region. In the region on the left, the S_N method of arbitrary order with diamond differencing is used. On the right, the Diffusion or P_3 approximation is used. The boundary between two regions is of particular interest because the appropriate communication of the data between the regions is important for the total solution over the whole domain.

4.2 S_N /Diffusion

This section describes the Sn-P$_1$ coupling technique. As mentioned earlier, on the Sn side of the problem, the angular fluxes are edge-centered. On the diffusion side we have the cell-centered scalar fluxes and the scalar flux $\phi_{\frac{1}{2}}$ and the net current $J_{\frac{1}{2}}$ at the interface. The unknowns from both regions overlap at the interface. The idea behind the coupling technique is to make the half-range current entering the diffusion region continuous. The half-range positive current is calculated using the Sn angular fluxes at the interface:

$$\hat{\mathrm{J}}_{\mathrm{t}}^{+} = \sum_{\mu_n > 0} \psi_{Sn}(x_{int}, \mu_n) \cdot w_n \mu_n, \tag{4.1}$$

where w_n and μ_n are the n-th weight and cosine from the Gauss quadrature set. For all the numerical integrations, the Gauss quadrature set of order n is used, where n is equal to the Sn order. This is done to ensure consistency between the half-range currents for different approximations. On the diffusion side, the half-range current entering the region is described by the equation

$$\sum_{\mu_m > 0} \frac{1}{2}\left(\Phi_{1/2} + 3J_{1/2}\mu_m\right) \cdot w_m \mu_m = \hat{\mathrm{J}}_{\mathrm{t}}^{+}. \tag{4.2}$$

The incoming Sn angular fluxes for negative directions μ_n<0 are calculated on the interface using the diffusion scalar flux $\phi_{\frac{1}{2}}$ and the net current $J_{\frac{1}{2}}$ at the interface:

$$\psi_{Sn}(x_{int}, \mu_m) = \frac{\Phi_{1/2} + 3J_{1/2}\mu_m}{4\pi}, \qquad \mu_m < 0. \tag{4.3}$$

On the picture below the communication of data between the regions is shown schematically on the Fig.5:

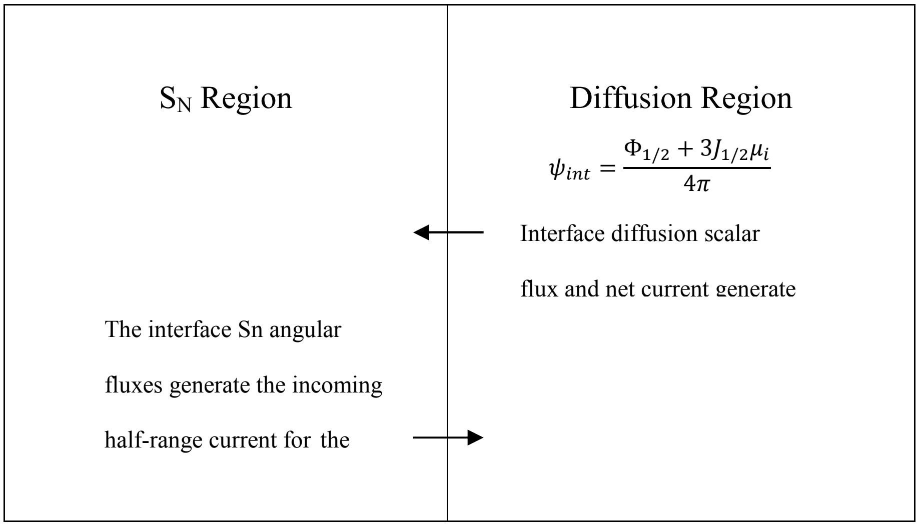
S_N Region
Diffusion Region
$\psi_{int} = \frac{\Phi_{1/2} + 3J_{1/2}\mu_i}{4\pi}$
Interface diffusion scalar
flux and net current generate
The interface Sn angular
fluxes generate the incoming
half-range current for the

Fig.5. Communication between the S_N and diffusion regions.

4.3 S_N/P_3

This section describes the Sn/P_3 coupling. In the diffusion region, the edge-centered P_3 approximation is used. There are Sn angular fluxes and the P_3 flux moments at the interface. The coupling technique preserves the positive half-range current and so-called second-order half-range current entering the P$_3$ region. Two half-range positive currents are calculated using the Sn angular fluxes at the interface:

$$\hat{\mathrm{j}}_{\mathrm{t}}^{+(0)} = \sum_{\mu_n>0} \psi_{Sn}(x_{int},\mu_n)\cdot w_n\mu_n P_0(\mu_n),$$
$$\hat{\mathrm{j}}_{\mathrm{t}}^{+(2)} = \sum_{\mu_n>0} \psi_{Sn}(x_{int},\mu_n)\cdot w_n\mu_n P_2(\mu_n). \tag{4.4}$$

On the P$_3$ side, the currents entering the region are set equal to the Sn-generated half-range currents:

$$\sum_{\mu_m>0}\left(\sum_{l=0}^{3}\left(\frac{2l+1}{4\pi}\right)\phi_{1/2}^{(l)}P_l(\mu_m)\right)P_k(\mu_m)\mu_m w_m = \hat{\mathrm{j}}_{\mathrm{t}}^{+(\mathrm{k})}, \qquad k = 0, 2. \tag{4.5}$$

At the interface in the P$_3$ region, the angular fluxes in the negative direction are calculated using the equation:

$$\psi_{P3}(x_{int},\mu_n) = \sum_{l=0}^{3}\left(\frac{2l+1}{4\pi}\right)\phi_l(x_{int})P_l(\mu_n); \quad \mu_n < 0. \tag{4.6}$$

Incoming Sn angular fluxes at the interface are set to the P$_3$-generated angular fluxes:

$$\psi_{P3}(x_{int},\mu_n) = \psi_{Sn}(x_{int},\mu_n); \quad \mu_n < 0. \tag{4.7}$$

5. SOLUTION TECHNIQUES FOR HYBRID APPROXIMATIONS

The solution techniques used for hybrid calculations are discussed in this section. These techniques use the global diffusion correction to accelerate the calculations. This diffusion correction applies to both regions and corrects the scalar fluxes after initial solutions produce the results in each region. We also call this correction ‘Global DSA’.

5.1 S_N/Diffusion

Two Sn/diffusion hybrid schemes were implemented. They will be referred to as ‘scheme one’ and ‘scheme two’. Scheme one fully converge the solution in the Sn region before solving diffusion equations in the diffusion region. Scheme two performs a single Sn sweep and then solves a set of diffusion equations which includes the DSA equations in the Sn region and diffusion equations for the diffusion region.

5.1.1 Hybrid Scheme with Full Sn Convergence

The hybrid scheme converges the solution in the Sn region before calculating the scalar fluxes in the diffusion region. After the solution in the Sn region has converged and the outcoming angular fluxes on the interface between the regions are known, the positive half-range current is calculated for the diffusion region. After the incoming into the diffusion region is known, the diffusion equations are solved. The detailes on the coupling of Sn and Diffusion can be found in the previous chapter of this thesis.

The problem with this scheme is that a large number of iterations may be needed to converge the solution as scattering dominates absorption in both the S_N and diffusion regions. However, it is important to recognize that one can "wrap" this iteration process with a Krylov method. Furthermore, one can formulate the equations such that there are only two unknowns in the Krylov system: the interface diffusion scalar flux and the S_N half-range current entering the diffusion subdomain. This means that with an error minimizing method such as GMRES, only two Krylov iterations (and thus two independent S_N and diffusion solves) are required for convergence regardless of the problem characteristics.

To solve a preconditioned system using the Krylov solver, the following steps have to be taken:

1) Generate a Krylov source vector $\vec{q}$ by performing a fixed-point iteration after assuming that the distributed source is present in both regions and that the initial iterate in both regions is zero for all scalar fluxes. The vector $\vec{q}$ is then passed to the Krylov solver routine.
2) The flux entering the Sn region are computing using Φ_D and from the Krylov vector $\hat{j}_t^+$ from the Krylov vector $\vec{x}$.
3) The Sn equations are solved assuming a zero distributed source.
4) The half-range current entering the diffusion region is calculated using the Sn solution.
5) The diffusion equation is solved.

6) Intermediate Krylov vector $\vec{y}$ is formed from the solution for Φ_{D} and $\hat{\jmath}_t^+$ from the previous step.

7) The final Krylov vector is generated as $\vec{z} = \vec{x} - \vec{y}$ and returned to the Krylov routine.

Once the Krylov solution is converged, one additional iteration of the fixed-point iteration scheme can be performed using the Krylov vector as initial iterate to obtain the angular fluxes in the Sn region and the scalar fluxes in the diffusion region.

5.1.2 Hybrid Scheme with Global Diffusion Correction

The main idea behind the Global Diffusion Correction technique for the Sn/diffusion hybrid scheme is to make the overall solution conservative after the correction is performed. The solution for the system of corrective equations consists of corrections for the cell-edge scalar fluxes in the Sn region and the cell-centered scalar fluxes in the diffusion region. The set of corrective equations in the Sn region is represented by regular DSA equations described in Section 3.3 of this thesis. The cell-centered diffusion equations are shown in Section 3.1.

To connect two sets of diffusion equations, we need to derive equations that would include the correction for the last cell edge in the Sn region and solution for the first cell center in the diffusion region. In this section we will consider only the cells on the interface between the regions. To simplify the indexing of the cells and make them shorter, we define the space indexing as shown on the Fig.6.

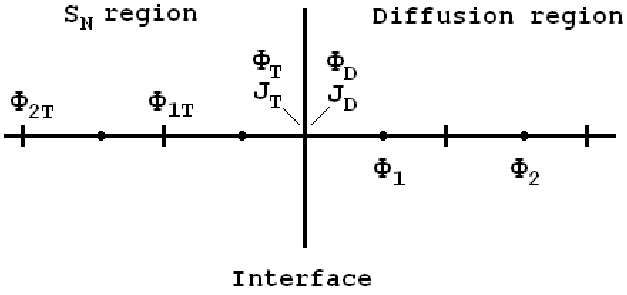

Fig.6. Spatial indexing on Sn/diffusion interface.

The right boundary DSA equation is

$$\delta J_T + \frac{D_T}{h}(\delta\varphi_T - \delta\varphi_{1T}) + \frac{\sigma_a h}{4}(\delta\varphi_T + \delta\varphi_{1T}) \\ = \frac{\sigma_s h}{4}\left(\varphi_T^{\left(l+\frac{1}{2}\right)} + \varphi_{1T}^{\left(l+\frac{1}{2}\right)} - \varphi_T^{(l)} - \varphi_{1T}^{(l)}\right). \tag{5.1}$$

Left boundary diffusion equation:

$$\frac{D_D}{h}\left(\varphi_2^{(l+1)} - \varphi_1^{(l+1)}\right) - J_D^{(l+1)} + \sigma_a h \varphi_1^{(l+1)} = q_D h. \tag{5.2}$$

The fundamental unknowns at the interface between the regions are the diffusion scalar flux $\varphi_D^{(l+1)}$, diffusion net current $J_D^{(l+1)}$ and the correction for the transport net current δJ_T. Thus three additional equations are needed to close (5.1) and (5.2). They are: Fick's law

$$J_D^{(l+1)} = \frac{-2D_D}{h}\left(\varphi_1^{(l+1)} - \varphi_D^{(l+1)}\right), \tag{5.3}$$

the continuity of positive half-range current:

$$j_T^{+,(l+1)} = j_D^{+,(l+1)}, \tag{5.4}$$

or

$$j_T^{+,\left(l+\frac{1}{2}\right)} + \frac{\delta\varphi_T\langle\mu\rangle}{2} + \frac{\delta J_T}{2} = \frac{\varphi_D^{(l+1)}\langle\mu\rangle}{2} + \frac{J_D^{(l+1)}}{2} \tag{5.5}$$

and the continuity of the negative half-range current:

$$j_T^{-,(l+1)} = j_D^{-,(l+1)}, \tag{5.6}$$

or

$$j_T^{-,\left(l+\frac{1}{2}\right)} + \frac{\delta\varphi_T\langle\mu\rangle}{2} - \frac{\delta J_T}{2} = \frac{\varphi_D^{(l+1)}\langle\mu\rangle}{2} - \frac{J_D^{(l+1)}}{2}. \tag{5.7}$$

After expressing the negative half-range current into the transport region in terms of the diffusion scalar flux and net current on the interface, the expression (5.7) becomes:

$$\frac{\phi_D^l\langle\mu\rangle}{2} - \frac{J_D^l}{2} + \frac{\delta\varphi_T\langle\mu\rangle}{2} - \frac{\delta J_T}{2} = \frac{\varphi_D^{(l+1)}\langle\mu\rangle}{2} - \frac{J_D^{(l+1)}}{2}. \tag{5.8}$$

Adding (5.5) and (5.8), we get:

$$j_T^{+,\left(l+\frac{1}{2}\right)} + \frac{\varphi_D^{(l)}\langle\mu\rangle}{2} - \frac{J_D^{(l)}}{2} + \delta\varphi_T\langle\mu\rangle = \varphi_D^{(l+1)}\langle\mu\rangle, \tag{5.9}$$

From this expression $\phi_D^{(l+1)}$ can be presented as a function of $\delta\varphi_T$:

$$\phi_D^{(l+1)} = \frac{1}{\langle\mu\rangle}\left(j_T^{+,\left(l+\frac{1}{2}\right)} + \frac{\varphi_D^{(l)}\langle\mu\rangle}{2} - \frac{J_D^{(l)}}{2}\right) + \delta\varphi_T. \tag{5.10}$$

Substituting from (5.10) into (5.3), we can express $J_D^{(l+1)}$ as:

$$J_D^{(l+1)} = \frac{-2D_D}{h}\left[\varphi_1^{(l+1)} - \delta\varphi_T - \frac{1}{\langle\mu\rangle}\left(j_T^{+,\left(l+\frac{1}{2}\right)} + \frac{\varphi_D^{(l)}\langle\mu\rangle}{2} - \frac{J_D^{(l)}}{2}\right)\right]. \tag{5.11}$$

Substituting from equation (5.11) into equation (5.2), we get the left boundary diffusion equation:

$$-\frac{D_D}{h}\left(\varphi_2^{(l+1)}-\varphi_1^{(l+1)}\right)+\frac{2D_D}{h}\left(\varphi_1^{(l+1)}-\delta\varphi_T\right)+\sigma_a h\varphi_1^{(l+1)} = q_1 h-\frac{2D_D}{h\langle\mu\rangle}\left[-j_T^{+,\left(l+\frac{1}{2}\right)}-\frac{\varphi_D^{(l)}\langle\mu\rangle}{2}+\frac{J_D^{(l)}}{2}\right]. \tag{5.12}$$

Subtracting (5.8) from (5.5), we get

$$j_T^{+,\left(l+\frac{1}{2}\right)}-\frac{\varphi_D^{(l)}\langle\mu\rangle}{2}+\frac{J_D^{(l)}}{2}+\delta J_T=J_D^{(l+1)}. \tag{5.13}$$

Substituting from (5.13) into (5.1), we get:

$$J_D^{(l+1)}-j_T^{+,\left(l+\frac{1}{2}\right)}+\frac{\varphi_D^{(l)}\langle\mu\rangle}{2}-\frac{J_D^{(l)}}{2}+\frac{D_T}{h}(\delta\varphi_T-\delta\varphi_{1T}) +\frac{\sigma_a h}{4}(\delta\varphi_T+\delta\varphi_{1T}) =\frac{\sigma_s h}{4}\left(\varphi_T^{\left(l+\frac{1}{2}\right)}+\varphi_{1T}^{\left(l+\frac{1}{2}\right)}-\varphi_T^{(l)}-\varphi_{1T}^{(l)}\right). \tag{5.14}$$

Substituting from (5.11) into the right boundary DSA equation (5.1), we get:

$$\frac{-2D_D}{h}\left[\varphi_1^{(l+1)}-\delta\varphi_T-\frac{1}{\langle\mu\rangle}\left(j_T^{+,\left(l+\frac{1}{2}\right)}+\frac{\varphi_D^{(l)}\langle\mu\rangle}{2}-\frac{J_D^{(l)}}{2}\right)\right]-j_T^{+,\left(l+\frac{1}{2}\right)} +\frac{\varphi_D^{(l)}\langle\mu\rangle}{2}-\frac{J_D^{(l)}}{2}+\frac{D_T}{h}(\delta\varphi_T-\delta\varphi_{1T}) +\frac{\sigma_a h}{4}(\delta\varphi_T+\delta\varphi_{1T}) =\frac{\sigma_s h}{4}\left(\varphi_T^{\left(l+\frac{1}{2}\right)}+\varphi_{1T}^{\left(l+\frac{1}{2}\right)}-\varphi_T^{(l)}-\varphi_{1T}^{(l)}\right). \tag{5.15}$$

Further algebraic manipulations of (5.15) yield the right boundary equation in the Sn region:

$$\frac{-2D_D}{h}\left(\varphi_1^{(l+1)} - \delta\varphi_T\right) + \frac{D_T}{h}(\delta\varphi_T - \delta\varphi_{1T}) + \frac{\sigma_a h}{4}(\delta\varphi_T + \delta\varphi_{1T})$$

$$= \frac{\sigma_s h}{4}\left(\varphi_T^{\left(l+\frac{1}{2}\right)} + \varphi_{1T}^{\left(l+\frac{1}{2}\right)} - \varphi_T^{(l)} - \varphi_{1T}^{(l)}\right) + \left(1 - \frac{2D_D}{\langle\mu\rangle h}\right) j_T^{+,\left(l+\frac{1}{2}\right)} \tag{5.16}$$

$$+ \left(\frac{2D_D}{\langle\mu\rangle h} + 1\right)\left(\frac{J_D^{(l)}}{2} - \frac{\varphi_D^{(l)}\langle\mu\rangle}{2}\right).$$

Thus the equations (5.16) and (5.12) hold in the two cells sharing the interface. On the interior of the diffusion region, the global diffusion equation is

$$-\left(\frac{D_D}{h}\right)(\varphi_{i+1} - \varphi_i) + \left(\frac{D_D}{h}\right)(\varphi_i - \varphi_{i-1}) + \sigma_a h \varphi_i = 0. \tag{5.17}$$

On the interior of the Sn region, the global diffusion equation is:

$$-\frac{D_T}{h}\left(\delta\phi_{i+\frac{1}{2}} - \delta\phi_{i-\frac{1}{2}}\right) + \frac{D_T}{h}\left(\delta\phi_{i-\frac{3}{2}} - \delta\phi_{i-\frac{1}{2}}\right)$$

$$+ \frac{\sigma_a h}{4}\left(\delta\phi_{i-\frac{3}{2}} + 2\delta\phi_{i-\frac{1}{2}} + \delta\phi_{i+\frac{1}{2}}\right) = 0. \tag{5.18}$$

The global diffusion equations satisfy vacuum boundary conditions on the outer boundaries

The fixed-point iterative lgorithm for solving the S$_N$-Diffusion scheme with the global diffusion correction is as follows:

1) Guess $\phi_D^{(0)}, J_D^{(0)}$and all transport scalar fluxes.

2) Set $\psi^{\left(l+\frac{1}{2}\right)}(\mu_n) = \frac{\varphi + 3J_D^{(l)}\mu_n}{4\pi}$, $\mu_n < 0$, *l = 0;*

3) Perform transport sweep and compute outgoing half-range current $j_T^{+,\left(l+\frac{1}{2}\right)}$.

4) Solve global diffusion equation for $\overrightarrow{\delta\varphi}$ and calculate $\vec{\varphi}^{\left(l+\frac{1}{2}\right)}$ by adding the correction to the Sn scalar fluxes.

5) Compute $\phi_D^{(l+1)}$ using equation (5.10).

6) Compute , $J_D^{(l+1)}$ using (5.3).

7) Return to step 1 and repeat until convergence is achieved.

For the case when a Krylov solver is used, the Krylov vector includes the scalar fluxes in the transport region, φ_D and J_D. To solve a preconditioned system using the Krylov solver, the following steps have to be taken:

1) Generate a Krylov source vector $\vec{q}$ by assuming that the distributed source is present in both regions and assume a zero initial guess for all the scalar fluxes, φ_D and J_D.

2) When the Krylov routine returns the Krylov vector $\vec{x}$, the action of the preconditioned operator on this vector must be calculated. To do this, we set the distributed sources in both regions to zero and calculate the incoming Sn angular fluxes on the interface using φ_D and J_D from the Krylov vector $\vec{x}$. Then a single Sn sweep is performed after the scattering sources are calculated using the Sn scalar fluxes from $\vec{x}$. Next, the global diffusion correction step is performed. The Krylov vector $\vec{y}$ is composed using the results of the calculations and the difference $\vec{z} = \vec{x} - \vec{y}$ is sent back into the Krylov routine.

Once the Krylov solution is converged, one additional iteration of the fixed-point iteration scheme is performed using the Sn scalar fluxes, φ_D and J_D from the Krylov vector as initial iterate to obtain the total solution.

5.2 S_N/P_3

There are two Sn/P$_3$ hybrid schemes available. They can be referred to as 'Sn/P$_3$ Scheme 1' and 'Sn/P$_3$ Scheme 2'. The only difference between these hybrid schemes is that Scheme 2 uses the technique called Global Diffusion Acceleration to improve the convergence rate.

5.2.1. S_N/P_3 Scheme without Acceleration

This Sn/P$_3$ scheme consist of a single Sn iteration with DSA, then a P$_3$ Gauss-Seidel iteration. The coupling technique is described in the section 4.2 of this thesis and the solution techniques for solving Sn and P$_3$ problems are described in the sections 3.2 and 3.3.

The Sn/P$_3$ fixed-point iteration scheme was wrapped in a Krylov solver. The Krylov source vector includes the scalar fluxes in both regions and all the interface moments in the Pn region. First, the Krylov source vector has to be calculated by performing a single iteration of the scheme with a zero initial guess. When the Krylov routine passes a Krylov vector $\vec{x}$ to the user, the action of the preconditioned operator on $\vec{x}$ is calculated. To accomplish this, a single iteration of the fixed-point iteration is performed using the scalar fluxes in both regions and the moments on the interface acquired from $\vec{x}$. After the iteration is performed, the resultant vector $\vec{y}$ is subtracted from $\vec{x}$ and $\vec{z} = \vec{x} - \vec{y}$ is sent back to the Krylov routine.

After the Krylov solution has converged, the scalar fluxes are known because the Krylov vector includes them. If a user wants to calculate the angular fluxes as well, an additional iteration of the fixed-point iteration scheme has to be performed using the data from the Krylov vector as the initial iterate to obtain the total solution.

5.2.2. S_N/P_3 Scheme with Global Diffusion Acceleration

The main idea behind the Global Diffusion Correction technique for the Sn/ P_3 hybrid scheme is to make the overall solution iterate conservative after the correction step is performed. The Sn and P_3 equations are locally conservative after a single Sn iteration with DSA and a P_3 Gauss-Seidel iteration ((l+1/2)-th step). But the overall solution is not conservative because the net current on the interface is not continuous. Therefore, the source for the diffusion corrective equations sharing the interface will be a function of the difference between Sn and P_3 net currents on the interface. On the interior of both regions, the corrective diffusion equations have no sources.

DSA is applied to accelerate the solution in the S_N region, and after that the Pn solution is produced based on the half-range currents coming from the S_N region into the Pn region. Next, the GDC corrects the net currents at the interface to make them continuous, and ensures that balance is maintained everywhere. Spatial indexing for the interface between the regions is shown on the Fig.7.

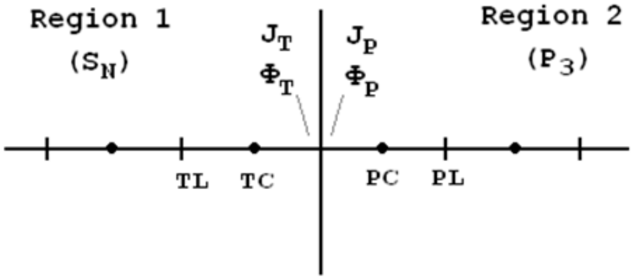

Fig.7. Variables on the interface between the S_N and P_3 regions.

Below is the detailed description of the GDC equations:

1) We have four corrective unknowns at the interface: $\delta\phi_T$, $\delta\phi_P$, δJ_T, and δJ_P. We now derive equations for all of them.

2) Before the global diffusion solve we satisfy the following equation in the right half of the last Sn cell:

$$J_T^{\left(l+\frac{1}{2}\right)} - J_{TC}^{\left(l+\frac{1}{2}\right)} + \left(\frac{\sigma_{at}h_t}{4}\right)\left(\phi_T^{\left(l+\frac{1}{2}\right)} + \phi_{TL}^{\left(l+\frac{1}{2}\right)}\right) = \frac{Q_{TC}h_t}{2}, \tag{5.19}$$

and the equation satisfied in the left half cell in the P_3 region is:

$$J_{PC}^{\left(l+\frac{1}{2}\right)} - J_P^{\left(l+\frac{1}{2}\right)} + \left(\frac{\sigma_{ap}h_P}{2}\right)\phi_P^{\left(l+\frac{1}{2}\right)} = \frac{Q_{PC}h_p}{2}. \tag{5.20}$$

We add (5.19) and (5.20)

$$\begin{aligned} J_{PC}^{\left(l+\frac{1}{2}\right)} - J_{TC}^{\left(l+\frac{1}{2}\right)} + \left(\frac{\sigma_{at}h_t}{4}\right)\left(\phi_T^{\left(l+\frac{1}{2}\right)} + \phi_{TL}^{\left(l+\frac{1}{2}\right)}\right) + \left(\frac{\sigma_{ap}h_P}{2}\right)\phi_P^{\left(l+\frac{1}{2}\right)} - \frac{Q_{TC}h_t}{2} \\ - \frac{Q_{PC}h_p}{2} = J_P^{\left(l+\frac{1}{2}\right)} - J_T^{\left(l+\frac{1}{2}\right)}. \end{aligned} \tag{5.21}$$

Note that (5.21) is not conservative unless the right side is zero. After the global diffusion correction we want to achieve conservation as follows:

$$J_{PC}^{(l+1)} - J_{TC}^{(l+1)} + \left(\frac{\sigma_{at} h_t}{4}\right)\left(\phi_T^{(l+1)} + \phi_{TL}^{(l+1)}\right) + \left(\frac{\sigma_{ap} h_P}{2}\right)\phi_P^{(l+1)} \\ = \frac{Q_{TC} h_t}{2} + \frac{Q_{PC} h_p}{2}. \tag{5.22}$$

We can express (5.22) in terms of the iterate at $\left(l + \frac{1}{2}\right)$ and the corrective diffusion solution:

$$\delta J_{PC} + J_{PC}^{\left(l+\frac{1}{2}\right)} - \delta J_{TC} - J_{TC}^{\left(l+\frac{1}{2}\right)} + \left(\frac{\sigma_{at} h_t}{4}\right)\left(\delta\phi_T + \phi_T^{\left(l+\frac{1}{2}\right)} + \delta\phi_{TL} + \phi_{TL}^{\left(l+\frac{1}{2}\right)}\right) \\ + \left(\frac{\sigma_{ap} h_P}{2}\right)\left(\delta\phi_P + \phi_P^{\left(l+\frac{1}{2}\right)}\right) = \frac{Q_{TC} h_t}{2} + \frac{Q_{PC} h_p}{2}. \tag{5.23}$$

By subtracting (5.23) from (5.22), we obtain a diffusion corrective equation spanning right half of the last Sn cell and left half of the first Pn cell:

$$\delta J_{PC} - \delta J_{TC} + \left(\frac{\sigma_{at} h_t}{4}\right)(\delta\phi_T + \delta\phi_{TL}) + \left(\frac{\sigma_{ap} h_P}{2}\right)\delta\phi_P = J_T^{\left(l+\frac{1}{2}\right)} - J_P^{\left(l+\frac{1}{2}\right)}. \tag{5.24}$$

where

$$\delta J_{PC} = -\left(\frac{D_p}{h_p}\right)(\delta\phi_{PL} - \delta\phi_P), \tag{5.25}$$

$$\delta J_{TC} = -\left(\frac{D_t}{h_t}\right)(\delta\phi_T - \delta\phi_{TL}). \tag{5.26}$$

We have equations for the $\delta\phi_T$ and $\delta\phi_T$. Now we derive another equations for the $\delta\phi'$s. Now we can consider the interface conditions that we want satisfied after the global diffusion solve. The first is continuity of the positive half-range currents

$$\hat{j}_T^{+\left(l+\frac{1}{2}\right)} + \frac{\delta\phi_T\langle\mu\rangle}{2} + \frac{\delta J_T}{2} = \hat{j}_T^{+\left(l+\frac{1}{2}\right)} + \frac{\delta\phi_P\langle\mu\rangle}{2} + \frac{\delta J_P}{2},$$

$$\frac{\delta\phi_T\langle\mu\rangle}{2} + \frac{\delta J_T}{2} = \frac{\delta\phi_P\langle\mu\rangle}{2} + \frac{\delta J_P}{2}. \tag{5.27}$$

The second is continuity of the negative half-range current

$$\hat{j}_T^{-(l)} + \frac{\delta\phi_T\langle\mu\rangle}{2} - \frac{\delta J_T}{2} = \hat{j}_P^{-\left(l+\frac{1}{2}\right)} + \frac{\delta\phi_P\langle\mu\rangle}{2} - \frac{\delta J_P}{2}, \tag{5.28}$$

$$\frac{\delta\phi_T\langle\mu\rangle}{2} - \frac{\delta J_T}{2} = \frac{\delta\phi_P\langle\mu\rangle}{2} - \frac{\delta J_P}{2}. \tag{5.29}$$

Adding (5.27) and (5.28) we get:

$$\hat{j}_T^{-(l)} + \delta\phi_T\langle\mu\rangle = \hat{j}_P^{-\left(l+\frac{1}{2}\right)} + \delta\phi_P\langle\mu\rangle. \tag{5.30}$$

This is our second equation. We use (5.30) to eliminate $\delta\phi_P$ from (5.24). This will allow us to solve a global diffusion that contains all corrective scalar fluxes except $\delta\phi_P$. After we solve the global diffusion equations, we use (5.30) again to get $\delta\phi_P$.

By subtracting (5.27) from (5.29) we get:

$$-\hat{j}_T^{-(l)} + \delta J_T = -\hat{j}_P^{-\left(l+\frac{1}{2}\right)} + \delta J_P. \tag{5.31}$$

This is our third equation. Now that we have all the $\delta\phi's$ in the transport region, we can write down the equation for $\phi_T^{(l+1)}$ in the last transport cell:

$$J_T^{\left(l+\frac{1}{2}\right)} + \delta J_T - J_{TC}^{\left(l+\frac{1}{2}\right)} - \delta J_{TC} + \left(\frac{\sigma_{at}h_t}{4}\right)(\delta\phi_T + \delta\phi_{TL}) = \frac{Q_T h_t}{2}, \tag{5.32}$$

Which gives us δJ_T. This is our fourth equation.

For the S_N interior, the corrective equations look like regular DSA equations (unknowns are at the vertices) with 0 source:

$$-\frac{D_T}{h_T}\left(\delta\phi_{i+\frac{1}{2}}-\delta\phi_{i-\frac{1}{2}}\right)+\frac{D_T}{h_T}\left(\delta\phi_{i-\frac{1}{2}}-\delta\phi_{i-\frac{3}{2}}\right)$$
$$+\frac{\sigma_{aT}h_T}{4}\left(\delta\phi_{i-\frac{3}{2}}+2\delta\phi_{i-\frac{1}{2}}+\delta\phi_{i+\frac{1}{2}}\right)=0. \qquad (5.33)$$

In the P_3 interior region, we solve the regular diffusion equations with the unknowns at the vertices with a zero interior source:

$$-\left(\frac{D_p}{h_p}\right)(\varphi_{i+1}-\varphi_i)+\left(\frac{D_p}{h_p}\right)(\varphi_i-\varphi_{i-1})+\sigma_{ap}h_P\varphi_i=0. \qquad (5.34)$$

For the first P_3 cell, the equation must include $\delta\phi_P$. However, since we expressed $\delta\phi_P$through $\delta\phi_T$, the equation is expressed as:

$$-\left(\frac{D_p}{h_p}\right)(\delta\phi_{PL+1}-\delta\phi_{PL})+\left(\frac{D_p}{h_p}\right)(\delta\phi_{PL}-\delta\phi_T)+\sigma_{ap}h_P\delta\phi_{PL}=0. \qquad (5.35)$$

According to (5.35), the ‘Global DSA’ matrix does not include $\delta\phi_P$. This shown on the Fig.8:

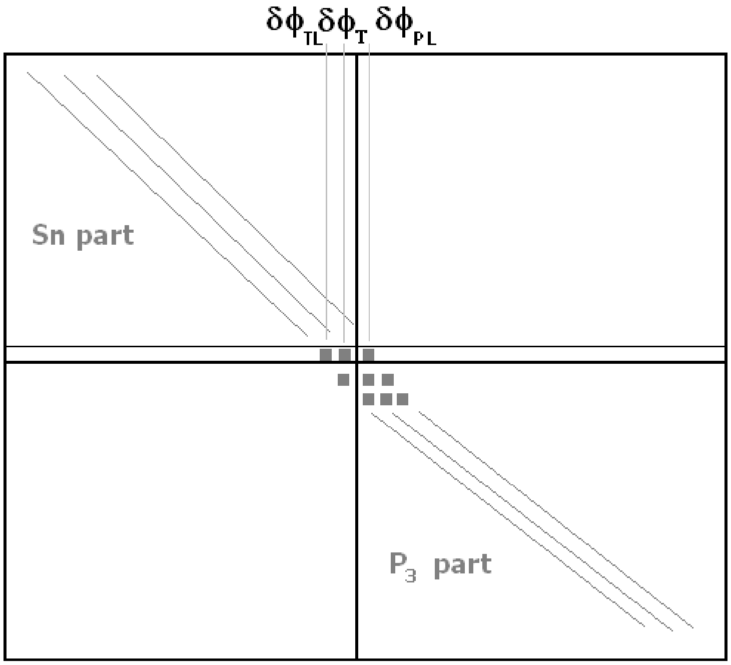

Fig.8. The ‘Global DSA’ matrix

The source vector contains only two elements – the right-hand sides of the equations (5.33) and (5.35).

Additional computations after the global diffusion correction are being performed. Since the solution of the corrective set of equations doesn’t contain $\delta\phi_P$, it will be calculated later using the following equation (5.30).

After the correcting the moments on the interface as follows:

$$\phi_0^{(l+1)} = \phi_0^{\left(l+\frac{1}{2}\right)} + \delta\phi_0,$$

$$\phi_1^{(l+1)} = \phi_1^{\left(l+\frac{1}{2}\right)} + \delta J_P,$$

$$\phi_2^{(l+1)} = \phi_2^{\left(l+\frac{1}{2}\right)},$$

$$\phi_3^{(l+1)} = \phi_3^{\left(l+\frac{1}{2}\right)}.$$

Now we can calculate the Sn incoming angular fluxes:

$$\psi_{I+\frac{1}{2}}(\mu_n) = \sum_{l=0}^{3} \left(\frac{2l+1}{4\pi}\right) \phi_l^{(l+1)}(x_{int}) P_l(\mu_n); \quad \mu_n < 0. \tag{5.36}$$

Algorithm of the solution for an S_N-P_3 problem using a GDC step is this:

1) Initialize $\varphi^{(0)}$, $\varphi^{(1)}$, $\varphi^{(2)}$, $\varphi^{(3)}$ on the boundary (all zeros).

2) Compute incoming angular fluxes ψ^{-} on the S_N boundary.

3) S_N sweep with DSA – returns $\vec{\psi}_T^{\left(l+\frac{1}{2}\right)}$ on the boundary.

4) Compute $\hat{J}^{+(0),\left(l+\frac{1}{2}\right)}$, $\hat{J}^{+(2),\left(l+\frac{1}{2}\right)}$.

5) A Gauss Seidel iteration using $\hat{J}^{+(0),\left(l+\frac{1}{2}\right)}$ and $\hat{J}^{+(2),\left(l+\frac{1}{2}\right)}$ - returns $\varphi^{(0), (l+1/2)}$, $\varphi^{(2), (l+1/2)}$.

6) Compute interface residual.

7) Perform global $\delta\varphi$ calculations.

8) Global diffusion solve $\varphi^{(0),(l+1)} = \varphi^{(0),(l+1/2)} + \delta\varphi^{(0)}$ for all φ's.

9) Update P_3 net current at the interface $\phi_{1,P}^{(l+1)} = \phi_{1,P}^{\left(l+\frac{1}{2}\right)} + \delta J_P$.

The data we have at each step is this:

1) Before S_N, DSA:

Incoming angular fluxes from Pn into S_N region.

2) After S_N and DSA step:

Angular fluxes $\psi^{\left(l+\frac{1}{2}\right)}$ on the interface,

Can compute:

Currents $J_T^{\left(\mathrm{l}+\frac{1}{2}\right)}$, $\hat{J}_T^{+\left(\mathrm{l}+\frac{1}{2}\right)}$, $J_{TC}^{\left(\mathrm{l}+\frac{1}{2}\right)}$.

3) After P_3 (Gauss-Seidel) step:

The moments $\phi^{\left(l+\frac{1}{2}\right)}$.

Can compute:

Currents $J_P^{\left(\mathrm{l}+\frac{1}{2}\right)}$, $\hat{J}_P^{+\left(\mathrm{l}+\frac{1}{2}\right)}$, $\hat{J}_P^{-\left(\mathrm{l}+\frac{1}{2}\right)}$.

4) After P_3 (Gauss-Seidel) step:

The corrective solution $\delta\vec{\phi}$ in both regions (except $\delta\phi_P$).

Can compute:

Correction $\delta\phi_P$ and correction for currents: δJ_T and $\hat{J}_T^{-(\mathrm{l}+1)}$

For a case when Krylov solver is used, a Krylov vector includes the scalar fluxes in both regions and all the moments at the boundary between the regions. To solve a preconditioned system using the Krylov solver, a Krylov source vector $\vec{q}$ must be calculated and passed to the Krylov routine. The Krylov source vector is computed by performing a single iteration of the hybrid scheme including the distributed sources, and assuming a zero initial guess for the scalar fluxes and moments in both regions. Each iteration, the Krylov routine passes the user a Krylov vector $\vec{x}$. The action of the preconditioned operator on this vector must be calculated and passed to the Krylov solver routine. This action is generated by performing a single iteration of the hybrid scheme with the distributed sources in both regions set to zero. First, the incoming Sn angular fluxes on the interface are calculated using the four boundary moments from the vector $\vec{x}$. Then a single Sn sweep with DSA is performed for which the initial scattering sources are calculated using the Sn scalar fluxes from $\vec{x}$. After this, the global diffusion correction is performed. The Krylov vector $\vec{y}$ is composed using the results of these calculations and the difference $\vec{z} = \vec{x} - \vec{y}$ is sent back into the Krylov routine. Once the Krylov solution is converged, one additional iteration of the fixed-point iteration scheme can be performed using the Krylov vector as initial iterate to obtain the angular fluxes in both regions.

6. COMPUTATIONAL RESULTS

The computational results produced in this work include the number of iterations needed to converge each hybrid scheme to the specified tolerance. Each hybrid scheme is tested in a geometry with two regions. Different transport approximations are used in each region. In the first region, the Sn approximation is used and for the second region we use either P_3 or the Diffusion approximation. For the test problems, we use uniform properties in both regions. All the schemes are stable regardless of cell size. If we use thicker cells, the number of iterations decreases. Therefore our test problems use highly refined mesh.

The set of constructed solutions used for code verification with the description of the anticipated result is shown in the appendix B. The computational results also include the execution time for each scheme and the execution time per iteration.

6.1 Tolerance and Measurement of Residual

Traditionally, for the neutron transport calculations the convergence criteria based on the relative change of the current iterate is used:

$$\varepsilon_j = \sqrt{\frac{\left(\sum_i \varphi_i^{(\mathrm{j+1})} - \sum_i \varphi_i^{(\mathrm{j})}\right)^2}{\sum_i \left(\varphi_i^{(\mathrm{j+1})}\right)^2}}\,. \tag{6.1}$$

In the equation (6.1),the index 'j' refers to the iteration number and 'i' refers to the spatial point. However, the GMRES solver used by MATLAB uses different criteria to stop iterations.

We start the description of the convergence criteria used by the Krylov solvers implemented in MATLAB by defining the L$_2$ norm of a vector $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}$ as

$$L_2(\vec{x}) = \sqrt{\sum_{i=1}^{n} (x_i)^2}\,.$$

For the problem that can be expressed as $\boldsymbol{A}\vec{x} = \vec{b}$ where **A** can be a matrix or an operator, the solution stops when

$$\frac{L_2\left(\boldsymbol{A}\vec{x}_l - \vec{b}\right)}{L_2\left(\vec{b}\right)} < \varepsilon_{tol}. \tag{6.2}$$

In the course of this research, a number of hybrid schemes was implemented. Some of these schemes are fixed-point iteration schemes and others were adapted to be used with a Krylov solver. To compare the schemes in terms of number of iterations needed to converge to a particular tolerance, the same convergence criteria must be used. To accomplish this, both the fixed point and Krylov calculations used the Krylov convergence criteria:

$$\varepsilon_{Krylov} = \frac{L_2\left(\boldsymbol{A}\vec{x}_l - \vec{b}\right)}{L_2\left(\vec{b}\right)}, \tag{6.3}$$

where $(A\vec{x} - \vec{b})$ is the system solved with the Krylov method. This measurement adds extra work for the fixed point schemes, so we did not include this extra work in the timing results presented later in this chapter.

6.2 S_N/Diffusion

Below the number of iterations needed to converge the two S_N/Diffusion hybrid schemes to the specified tolerance ε_{tol} is shown. The Krylov-based residual is calculated using the equation (6.4). All the schemes involve using DSA in the first region for acceleration of the Sn.

Table 1.

Number of iterations for S_8/P_1 schemes.

Tolerance $\varepsilon_{tol}=10^{-10}$, 100:100 cells, total thickness of each region is 10 mean free paths.

Problem description				$\sigma_t = 1$ in both regions		
Scheme No.	Region 1	Region 2	Krylov	c = 0	c = 0.5	c = 1
1	S_8	Diffusion	No	2	9	97
1	S_8	Diffusion	Yes	2	2	2
2	S_8 (GDC)	Diffusion GDC	No	5	8	11
2	S_8 (GDC)	Diffusion GDC	Yes	4	8	9

There is a difference between the way the iterations are counted for the Scheme 1 and Scheme 2. For the scheme with the full Sn convergence (Scheme 1), the number of outer iterations is shown in the Table 1. An outer iteration in this context is an iteration over the interface unknowns. It is performed after full convergence of the Sn solution. For the hybrid scheme with the GDC (Scheme 2) the total number of iterations is shown. For the schemes using a Krylov solver, the number of iterations does not include calculation of the Krylov source vector.

It can be seen from the results shown in the Table 1 that introducing the GDC step shows good improvement of convergence rate of Scheme 1 for the problems with high scattering ratio. For the Scheme 2, the improvement is less significant. The preconditioned Scheme 1 with a GMRES Krylov solver converges in two iterations. This fact shows that the implementation of the preconditioned scheme is correct because the Krylov solver for this scheme consists of two elements. The preconditioned Scheme 2 with a Krylov solver doesn't show much improvement compared to the corresponding fixed-point iteration scheme. The Krylov method shows less improvement when the fixed point iteration scheme is most efficient.

6.3 S_N/P_3

In the Table 2, the number of iterations needed to converge the hybrid Sn/P_3 schemes to the specified tolerance ε_{tol} is shown. The Krylov-based residual is calculated using equation (6.4). All the schemes involve using DSA in the region 1 for acceleration of the Sn.

Table 2.

Number of iterations for S_8/P_3 schemes.

Tolerance $\varepsilon_{tol}=10^{-10}$, 100:100 cells, total thickness of each region is 10 mean free paths.

Problem description				Scattering ratio		
Scheme No.	Region 1	Region 2	Krylov	c = 0	c = 0.5	c = 1
3	S_8	P_3 -Gauss Seidel	No	22	24	96
3	S_8	P_3 -Gauss Seidel	Yes	16	18	38
4	S_8 (GDC)	P_3 -Gauss Seidel GDC	No	21	23	21
4	S_8 (GDC)	P_3 -Gauss Seidel GDC	Yes	13	14	13

As described in the section 6.1, the results for number of iterations are produced using the Krylov-based tolerance. Fig.9 is an illustration of the fact that if Krylov-based calculation of the residual is used, the parameter ε representing the relative error at each iteration decreases faster than the relative change of scalar fluxes defined by the equation (6.1).

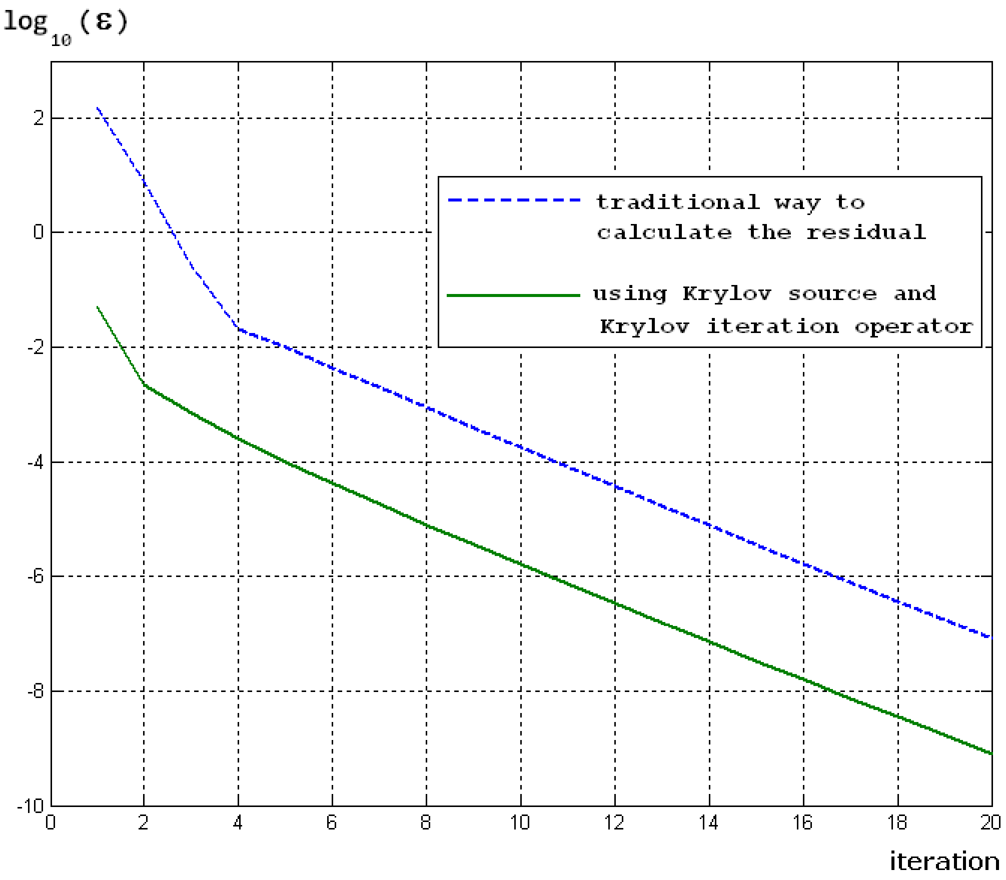

Fig.9. Relative change for S_4-P_3 scheme with global acceleration calculated using different methods. 100:100 cells, c = 1, $h_t = 0.1$.

Introducing the GDC step shows good improvement of convergence for the problems with high scattering ratio. The relative change of the scalar flux for the problem with c=1 in both regions is shown on the Fig.10. Along the vertical axis, the relative change of scalar flux ε is shown. Along the horizontal axis, the iteration number is shown.

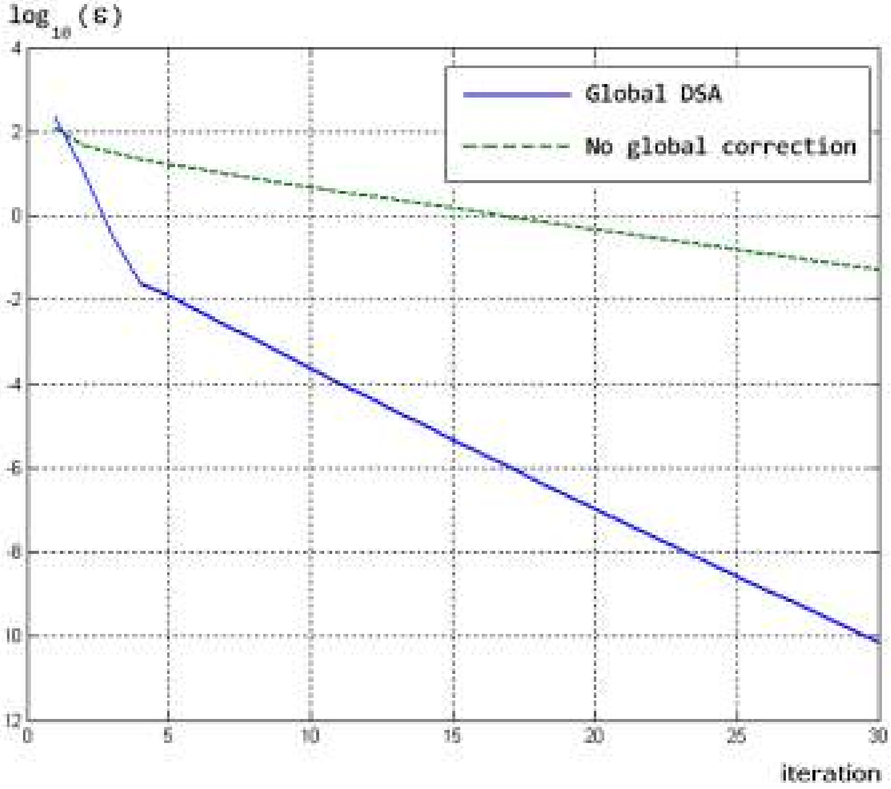

Fig.10. Convergence ratios for the pure scattering problem with and without the global DSA. S_4/ P_3 scheme, 100 cells, h_t=0.1.

The global diffusion correction shows slight improvement of the convergence rate for pure absorber problem. For the problems with the scattering ratio approaching unity, the improvement is significant.

6.4 Execution Time

The execution time has been measured for the S_8/Diffusion and S_8/P_3 schemes. The execution time is acquired using the Matlab routines ‘cputime’ (the value is shown on the upper row) and ‘tic’ and ‘toc’ (lower row). For each configuration, the measurements are taken five times and then the average value is computed. The results are shown in the Table 3.

Table 3.

Execution time for S_8/Pn schemes.

Tolerance $\varepsilon_{tol}=10^{-10}$, 100:100 cells, total thickness of each region is 10 mean free paths.

Problem description				$\sigma_t = 1$ in both regions		
Scheme #	Region 1	Region 2	Krylov	c = 0	c = 0.5	c = 1
1	S_8	Diffusion	No	0.23	7.04	292.49
				0.22	7.04	292.53
1	S_8	Diffusion	Yes	0.59	4.04	15.37
				0.59	4.04	15.39
2	S_8 (GDC)	Diffusion GDC	No	0.29	0.46	0.67
				0.30	0.46	0.67
2	S_8 (GDC)	Diffusion GDC	Yes	0.36	0.70	0.72
				0.36	0.70	0.72

Table 3 Continued.

Problem description				$\sigma_t = 1$ in both regions		
Scheme #	Region 1	Region 2	Krylov	c = 0	c = 0.5	c = 1
3	S_8	P_3 -Gauss Seidel	No	2.35	2.53	10.12
				2.35	2.53	10.12
3	S_8	P_3 -Gauss Seidel	Yes	2.09	2.12	4.63
				2.09	2.12	4.64
4	S_8 (GDC)	P_3 -Gauss Seidel	No	2.12	2.30	2.10
		GDC		2.12	2.31	2.10
4	S_8 (GDC)	P_3 -Gauss Seidel	Yes	1.61	1.72	1.62
		GDC		1.62	1.72	1.62

The cost of one iteration in terms of execution time is close for S_8/P_3 and S_8/P_3 with GDP schemes. The approximate execution time per iteration is shown below:

S_8/P_3: 0.106 ± 0.002 seconds.

S_8/P_3 (Krylov): 0.122 ± 0.003 seconds.

S_8/P_3 with GDC: 0.111 ± 0.003 seconds.

S_8/P_3 with GDC (Krylov): 0.117 ± 0.005 seconds.

S_8/P_1: 0.054 ± 0.006 seconds.

S_8/P_1(Krylov): 0.059 ± 0.007 seconds.

$S_8/P_1$with GDC: 0.059 ± 0.002 seconds.

$S_8/P_1$with GDC (Krylov): 0.065 ± 0.005 seconds.

The results show that the overhead of Krylov is no more than about 12% for our calculations, but there is considerable noise in the measurements.

7. CONCLUSIONS

As can be seen from the Tables 1 and 2, the Krylov solver is proven to be most effective for the schemes for which the corresponding fixed-point iteration schemes are less effective. For instance, the S_N-Diffusion scheme with the global diffusion-based acceleration rapidly converges, and the number of iteration obtained using the Krylov solver is close to the number of iteration for the fixed-point iteration scheme. In general, the Krylov solver is most effective when the FPIS is least effective.

Sn-P_3 Gauss Seidel scheme without the global DSA converge relatively fast for the problem where absorption dominates scattering. However, for pure scattering problem the number of iteration increases dramatically. The global DSA improves the convergence rate to the level of that of a pure absorber. The most significant improvement can be seen when global DSA is applied to the pure scattering medium.

This research demonstrated several efficient ways to simultaneously use different numerical transport approximations for the solution of a transport problem. The next step for future research would be to investigate the GDSA schemes in 2-D and 3-D geometries. Another topic for the future work is the effect of local block solves about the interface. The idea behind this approach is to use the local block solves to improve the shape of the solution in the area around the interface. The diffusion correction can only correct the magnitude of the solution, not its shape. Thus a significant improvement might be possible.

REFERENCES

1. Enrico Giradi, Jean Michael Ruggieri, and Simone Santandrea. Unstructured Characteristic Method Embedded with Variational Nodal Method Using Domain Decomposition Techniques. Proceedings of the International Meeting on Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications, Palais des Papes, Avignon, France, 12-15 September 2005, American Nuclear Society, LaGrange Park, IL (2005).

2. Personal Communication, Dr. Cristian Ribitti, Idaho National Laboratory, (2009).

3. James S.Warsa, Todd A. Wareing, Jim E. Morel. On the Degraded Effectiveness of Diffusion Synthetic Acceleration for Multidimensional SN Calculations in the Presence of Material Discontinuities. Proceedings of the 2003 Nuclear Mathematical and Computational Science, 6-11 April 2003, Gatlinburg, TN (2003).

4. B.G. Carlson and K.D.Lathrop. Transport Theory – The Method of Discrete-Ordinates, pp.165-266. Computing Methods in Reactor Physics, Gordon and Breach, New York (1968).

5. H.J. Kopf. Synthetic Method Solution of the Transport Equation. Nuclear Science and Engineering, pp.17, 65 (1963).

6. R.E. Alcouffe. Diffusion Sinthetic Acceleration Methods for the Diamond-Differenced Discrete-Ordinates Equations. Nuclear Science and Engineering, pp.64, 344-350 (1971).

7. W. H. Reed. The Effectiveness of Acceleration Techniques for Iterative Methods in Transport Theory. Nuclear Science and Engineering, pp.45, 245-254 (1971).

APPENDIX A

FOURIER ANALYSIS FOR P$_3$ GAUSS-SEIDEL SCHEME

Initial equations:

$$\begin{cases} \left(\frac{k+1}{2k+1}\right)\varphi'_{k+1} + \left(\frac{k}{2k+1}\right)\varphi'_{k-1} + \beta_k\varphi_k = 0, \qquad k = 0..N, \\ \varphi'_{-1} = \varphi'_{N+1} = 0, \\ \varphi_k = -\frac{1}{\beta_k}\left[\left(\frac{k+1}{2k+1}\right)\varphi'_{k+1} + \left(\frac{k}{2k+1}\right)\varphi'_{k-1}\right]. \end{cases}$$

$$\Rightarrow \varphi_{k-1} = -\frac{1}{\beta_{k-1}}\left[\left(\frac{k}{2k-1}\right)\varphi'_k + \left(\frac{k-1}{2k-1}\right)\varphi'_{k-2}\right].$$

$$\varphi_{k+1} = -\frac{1}{\beta_{k+1}}\left[\left(\frac{k+2}{2k+3}\right)\varphi'_{k+2} + \left(\frac{k+1}{2k+3}\right)\varphi'_k\right].$$

Substituting expressions for φ_{k-1} and φ_{k+1} back into the original equation, we get:

$$-\left(\frac{k+1}{2k+1}\right)\frac{1}{\beta_{k+1}}\left[\left(\frac{k+2}{2k+3}\right)\varphi''_{k+2} + \left(\frac{k+1}{2k+3}\right)\varphi''_k\right] - \left(\frac{k}{2k+1}\right)\frac{1}{\beta_{k-1}}\left[\left(\frac{k}{2k-1}\right)\varphi''_k + \left(\frac{k-1}{2k-1}\right)\varphi''_{k-2}\right] + \beta_k\varphi_k = 0.$$

If we consider P$_3$, we can get expressions for k=0 and k=2:

K=0:

$$-\left(\frac{k+1}{2k+1}\right)\frac{1}{\beta_{k+1}}\left[\left(\frac{k+2}{2k+3}\right)\varphi''_{k+2} + \left(\frac{k+1}{2k+3}\right)\varphi''_k\right] - \left(\frac{k}{2k+1}\right)\frac{1}{\beta_{k-1}}\left[\left(\frac{k}{2k-1}\right)\varphi''_k + \left(\frac{k-1}{2k-1}\right)\varphi''_{k-2}\right] + \beta_k\varphi_k =$$

$$\boldsymbol{-\frac{1}{\beta_1}\left[\frac{2}{3}\varphi''_2 + \frac{1}{3}\varphi''_0\right] + \beta_0\varphi_0 = 0}.$$

K=2:

$$-\left(\frac{k+1}{2k+1}\right)\frac{1}{\beta_{k+1}}\left[\left(\frac{k+2}{2k+3}\right)\varphi''_{k+2}+\left(\frac{k+1}{2k+3}\right)\varphi''_{k}\right]$$

$$-\left(\frac{k}{2k+1}\right)\frac{1}{\beta_{k-1}}\left[\left(\frac{k}{2k-1}\right)\varphi''_{k}+\left(\frac{k-1}{2k-1}\right)\varphi''_{k-2}\right]+\beta_k\varphi_k=$$

$$-\frac{3}{5}\,\frac{1}{\beta_3}\left[\frac{3}{7}\varphi''_2\right]-\frac{2}{5}\,\frac{1}{\beta_1}\left[\frac{2}{3}\varphi''_2+\frac{1}{3}\varphi''_0\right]+\beta_2\varphi_2=$$

$$-\frac{9}{35\cdot\beta_3}\ \varphi''_2-\left[\frac{4}{15\cdot\beta_1}\varphi''_2+\frac{2}{15\cdot\beta_1}\varphi''_0\right]+\beta_2\varphi_2$$

$$\boldsymbol{=-\left[\frac{9}{35\cdot\beta_3}+\frac{4}{15\cdot\beta_1}\right]\varphi''_2-\left[\frac{2}{15\cdot\beta_1}\right]\varphi''_0+\beta_2\varphi_2=0}.$$

Gauss-Seidell Scheme

1. We first lag the 0-th moment and solve for the second moment:

$$-\left[\frac{9}{35\cdot\beta_3}+\frac{4}{15\cdot\beta_1}\right]\cdot\left(\frac{\varphi_{i+1}^{2.(l+1)}-2\varphi_i^{2.(l+1)}+\varphi_{i-1}^{2.(l+1)}}{h}\right)-\left[\frac{2}{15\cdot\beta_1}\right]\left(\frac{\varphi_{i+1}^{0.(l)}-2\varphi_i^{0.(l)}+\varphi_{i-1}^{0.(l)}}{h}\right)+\beta_2 h\varphi_i^{2.(l+1)}=0, \tag{A.1}$$

Next, we update the 0-th moment with new values of the second moment:

$$-\frac{2}{3\beta_1}\left(\frac{\varphi_{i+1}^{2.(l+1)}-2\varphi_i^{2.(l+1)}+\varphi_{i-1}^{2.(l+1)}}{h}\right)-\frac{1}{3\beta_1}\left(\frac{\varphi_{i+1}^{0.(l+1)}-2\varphi_i^{0.(l+1)}+\varphi_{i-1}^{0.(l+1)}}{h}\right)+\beta_0\cdot h\cdot\varphi_i^{0.(l+1)}=0. \tag{A.2}$$

Now we make the Fourier anzatz:

$\varphi_i^k \equiv \varphi^k\cdot e^{j\lambda x_i}$.

Then the equations become:

$$\begin{aligned}
&-\left[\frac{9}{35\cdot\beta_3}+\frac{4}{15\cdot\beta_1}\right]\left(\varphi^{2,(l+1)}\cdot e^{j\lambda(x_i+h)} - 2\varphi^{2,(l+1)}\cdot e^{j\lambda x_i} + \varphi^{2,(l+1)}\cdot e^{j\lambda(x_i-h)}\right)\\
&\qquad -\left[\frac{2}{15\cdot\beta_1}\right]\left(\varphi^{0,(l)}\cdot e^{j\lambda(x_i+h)} - 2\varphi^{0,(l)}\cdot e^{j\lambda x_i} + \varphi^{0,(l)}\cdot e^{j\lambda(x_i-h)}\right)\\
&\qquad + \beta_2 h^2\varphi^{2,(l+1)}\cdot e^{j\lambda x_i} = 0,
\end{aligned} \tag{A.3}$$

and

$$\begin{aligned}
&-\frac{2}{3\beta_1}\left(\varphi^{2,(l+1)}\cdot e^{j\lambda(x_i+h)} - 2\varphi^{2,(l+1)}\cdot e^{j\lambda x_i} + \varphi^{2,(l+1)}\cdot e^{j\lambda(x_i-h)}\right)\\
&\qquad -\frac{1}{3\beta_1}\left[\varphi^{0,(l+1)}\cdot e^{j\lambda(x_i+h)} - 2\varphi^{0,(l+1)}\cdot e^{j\lambda x_i} + \varphi^{0,(l+1)}\right.\\
&\qquad \left.\cdot e^{j\lambda(x_i-h)}\right] + \beta_0 h^2\varphi^{0,(l+1)}\cdot e^{j\lambda x_i} = 0.
\end{aligned} \tag{A.4}$$

In the equations, $\beta_n = \sigma_t - \sigma_n = \sigma_t - 2\pi\int_{-1}^{1}\sigma_s(\mu)\cdot P_n(\mu)d\mu$.

Here σ_a and σ_t are macroscopic crossections and have units of 1/cm^3. Therefore,

$$\beta_n h = \begin{cases}\sigma_a h & if\ n = 0,\\ \sigma_t h & if\ n \neq 0.\end{cases}$$

After a series of algebraic manipulations, the equations (A.3) and (A.4) become

$$\left(h_t^2 - \frac{22}{21}(\cos(\lambda h)-1)\right)\varphi^{2,(l+1)} = \left[\frac{4}{15}\right](\cos(\lambda h)-1)\varphi^{0,(l)} \tag{A.5}$$

and

$$\left[(1-c){h_t}^2 - \frac{2}{3}(\cos(\lambda h)-1)\right]\varphi^{0,(l+1)} - \left[\left(\frac{4}{3}\right)(\cos(\lambda h)-1)\right]\varphi^{2,(l+1)} = 0. \tag{A.6}$$

In the matrix form, the equations (A.5) and (A.6) look like the following:

$$\begin{pmatrix} 0 & k1 \\ m1 & m2 \end{pmatrix} \begin{pmatrix} \varphi^{0,(l+1)} \\ \varphi^{2,(l+1)} \end{pmatrix} = \begin{pmatrix} k2 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \varphi^{0,(l)} \\ \varphi^{2,(l)} \end{pmatrix},$$

where the coefficients defined as

$$k1 \equiv \left(h_t^2 - \frac{22}{21}(\cos(\lambda h) - 1)\right),$$

$$k2 \equiv \left[\frac{4}{15}\right](\cos(\lambda h) - 1),$$

$$m1 \equiv \left[\left(\frac{4}{3}\right)(\cos(\lambda h) - 1)\right],$$

$$m2 \equiv \left[(1 - c){h_t}^2 - \frac{2}{3}(\cos(\lambda h) - 1)\right].$$

Finally, expression for the even moments in the matrix form becomes:

$$\begin{pmatrix} \varphi^{0,(l+1)} \\ \varphi^{2,(l+1)} \end{pmatrix} = \begin{pmatrix} 0 & k1 \\ m1 & m2 \end{pmatrix}^{-1} \begin{pmatrix} k2 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \varphi^{0,(l)} \\ \varphi^{2,(l)} \end{pmatrix},$$

If we define the matrix **H** as

$$\boldsymbol{H} \equiv \begin{pmatrix} 0 & k1 \\ m1 & m2 \end{pmatrix}^{-1} \begin{pmatrix} k2 & 0 \\ 0 & 0 \end{pmatrix},$$

the expression for the even moments looks like

$$\vec{\varphi}^{\,(l+1)} = \boldsymbol{H}\vec{\varphi}^{(l)}.$$

The matrix H has two eigenvalues. The maximal eigenvalue is equal to the spectral radius of the Gauss-Seidel scheme.

Results

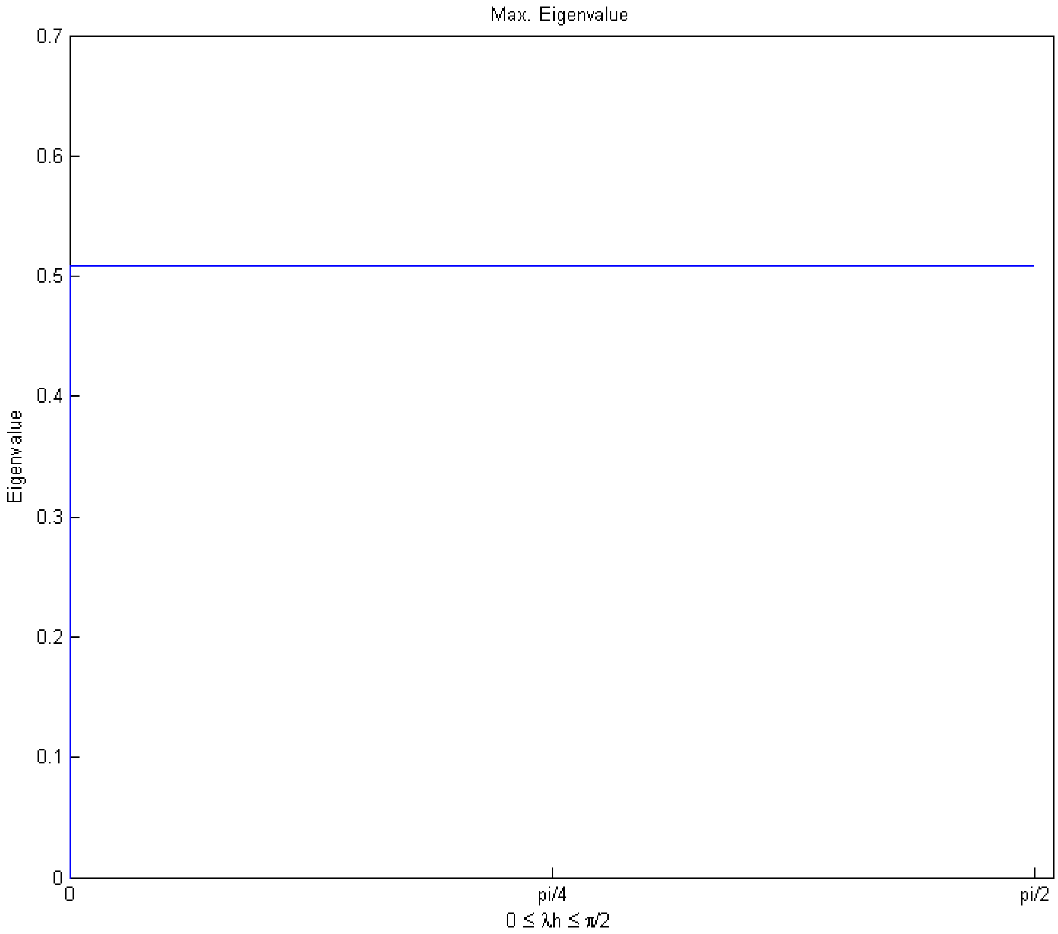

Fig. A1. Maximal eigenvalue of the matrix **H** as a function of λh for $h_t = 10^{-6}$ (the size of a cell in terms of σ_t). The results for c = 0, 0.5 and 1 are the same at such h_t.

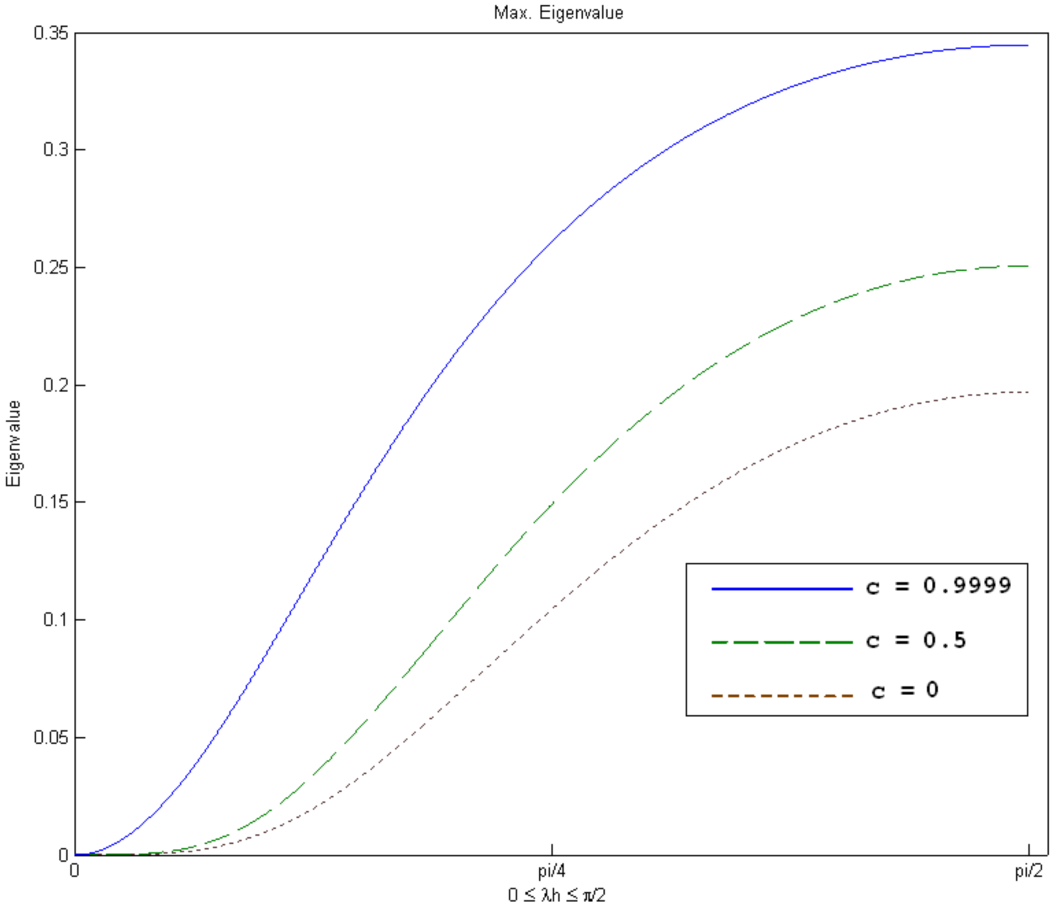

Fig. A2. Maximal eigenvalue of the matrix **H** as a function of λh for $h_t = 1$ (the size of a cell in terms of σ_t).

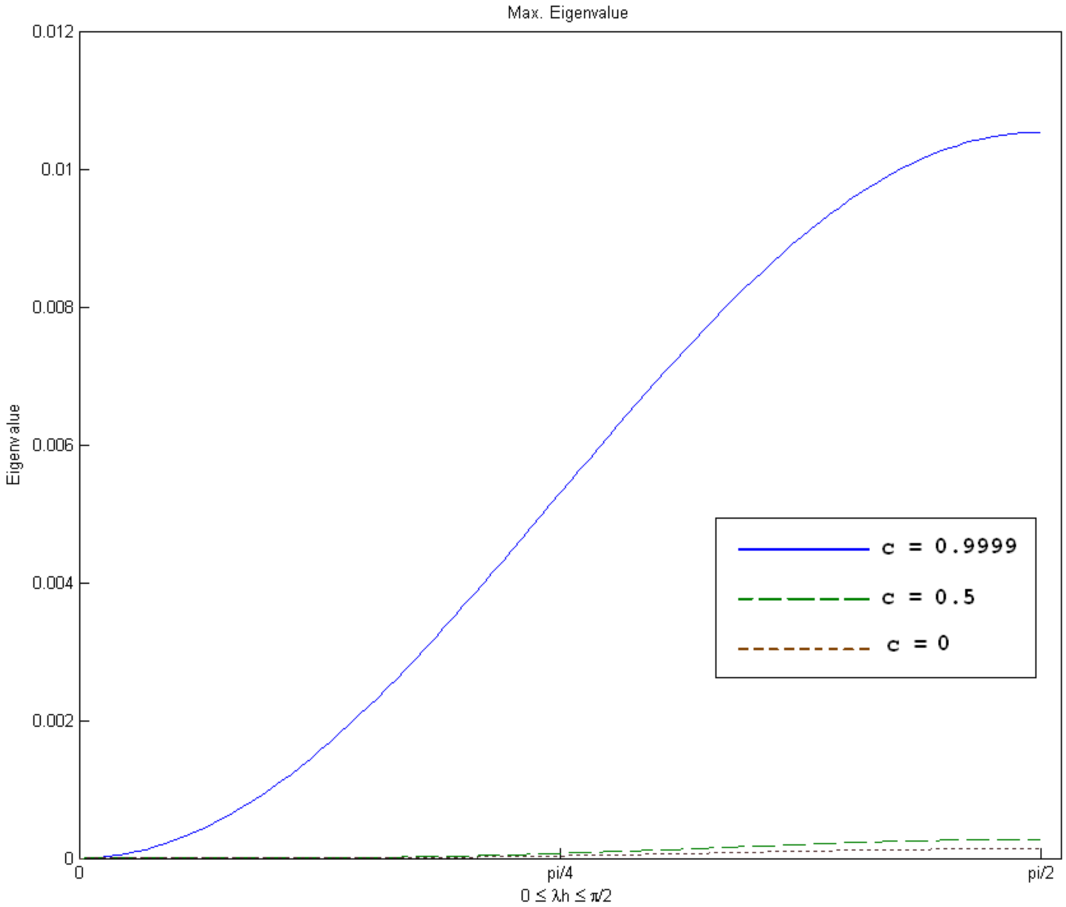

Fig. A3. Maximal eigenvalue of the matrix **H** as a function of λh for $h_t = 10$ (the size of a cell in terms of σ_t).

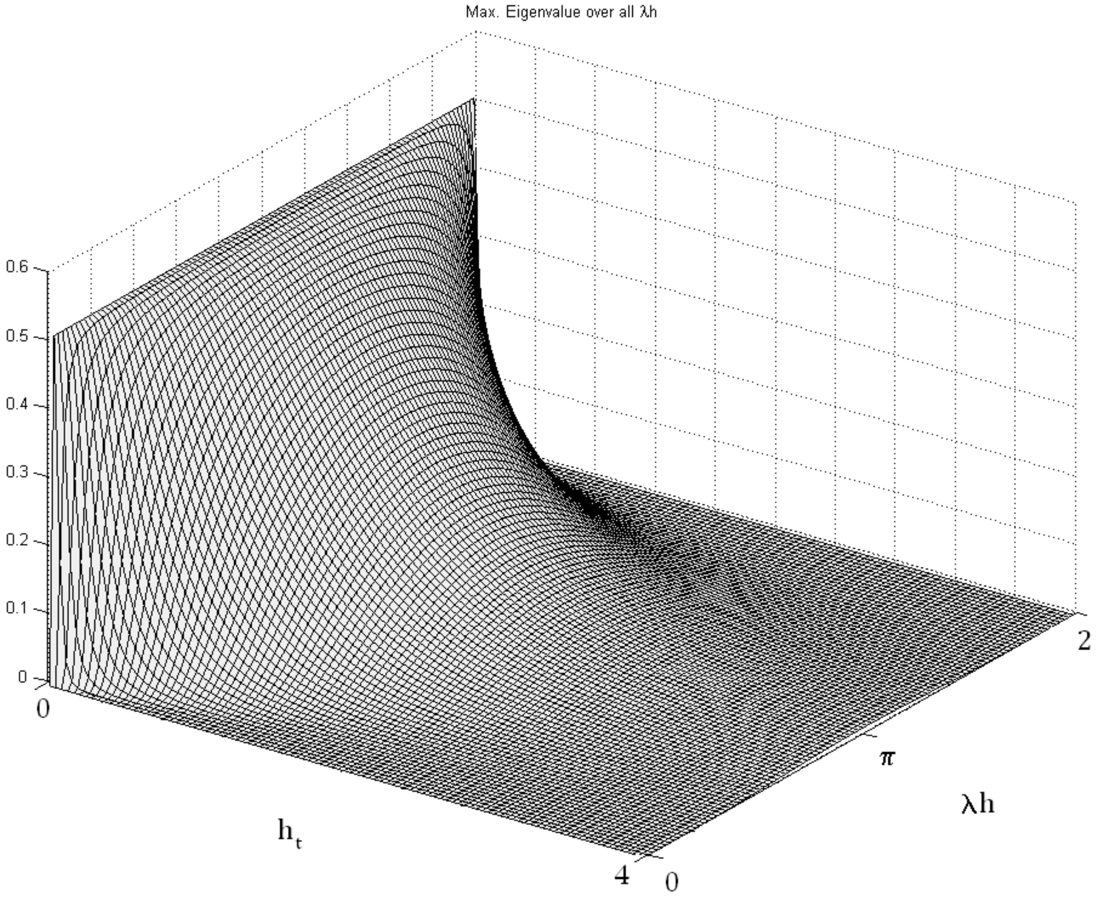

Fig. A4. Maximal Eigenvalue of the matrix **H** as a function of λh and h_t for c = 0.5.

On the figure below, the maximal eigenvalue for all λh as a function of the scattering ratio and cell size is shown; $h_t \in (0, 4]$, $c \in [0, 1)$.

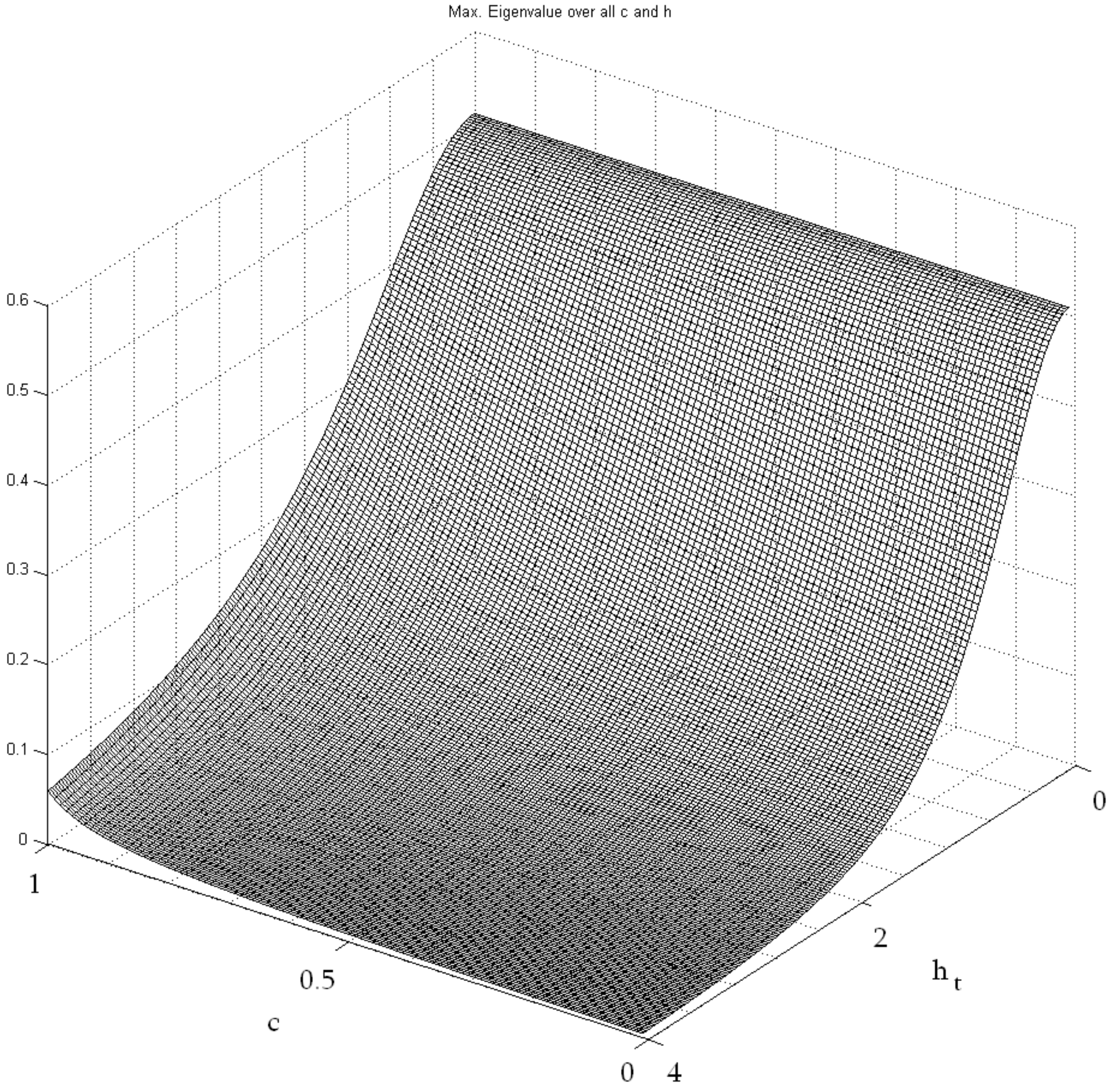

Fig. A5. Maximal Eigenvalue for the Gaus-Seidel scheme as a function of h_t and c.

The spectral radius for the Sn with DSA is well known and approximately equal to 0.23c. The figure A6 shows the spectral radius for the P_3 Gauss-Seidel iterative scheme plotted with the spectral radius for Sn.

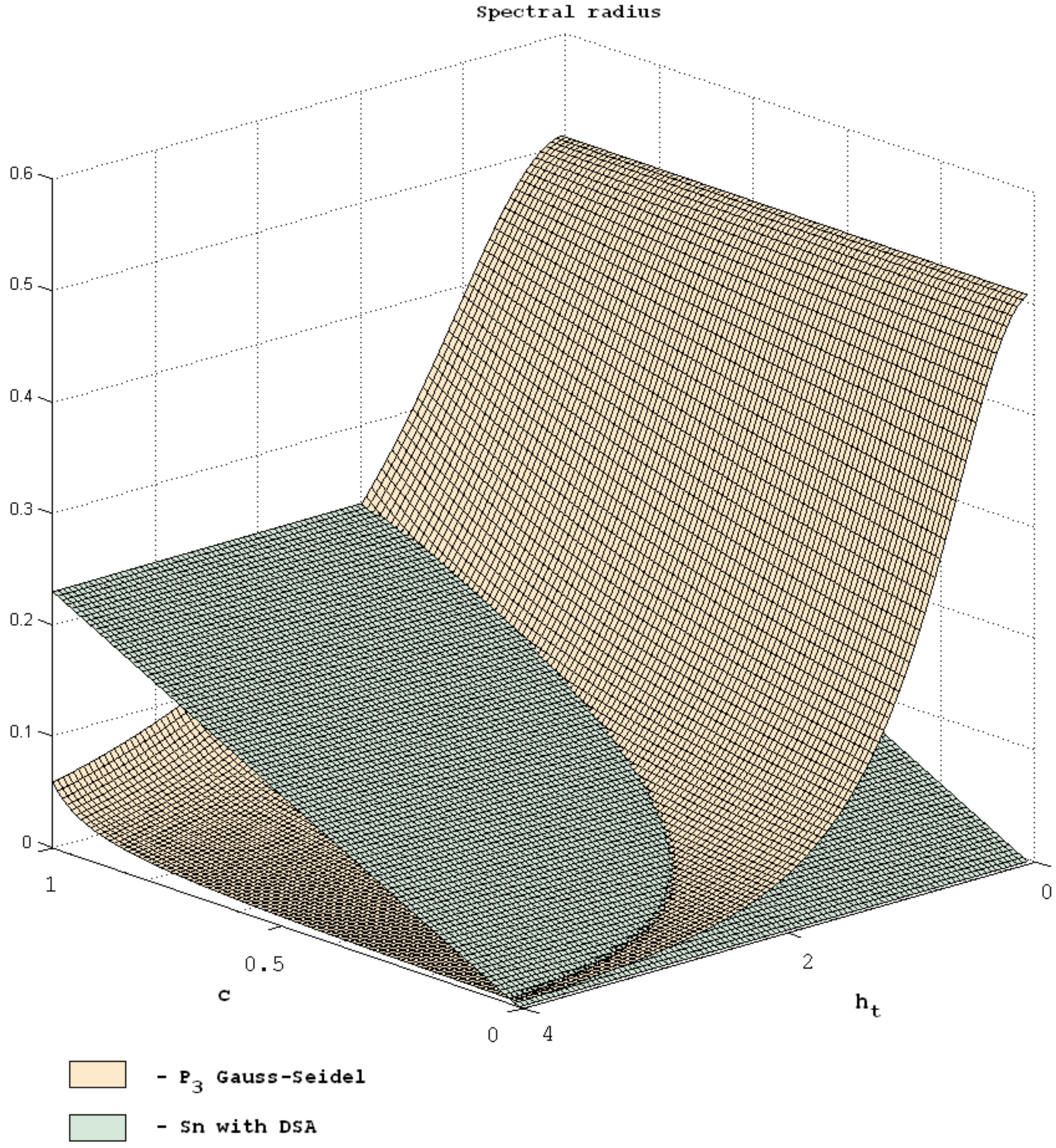

Fig. A6. Spectral radius for the Sn and P_3 Gauss-Seidel schemes as functions of h_t and c.

From this data, the regions in (h_t,c) phase space can be identified where the spectral radius of one of the schemes is smaller. These regions are shown on the figure A7.

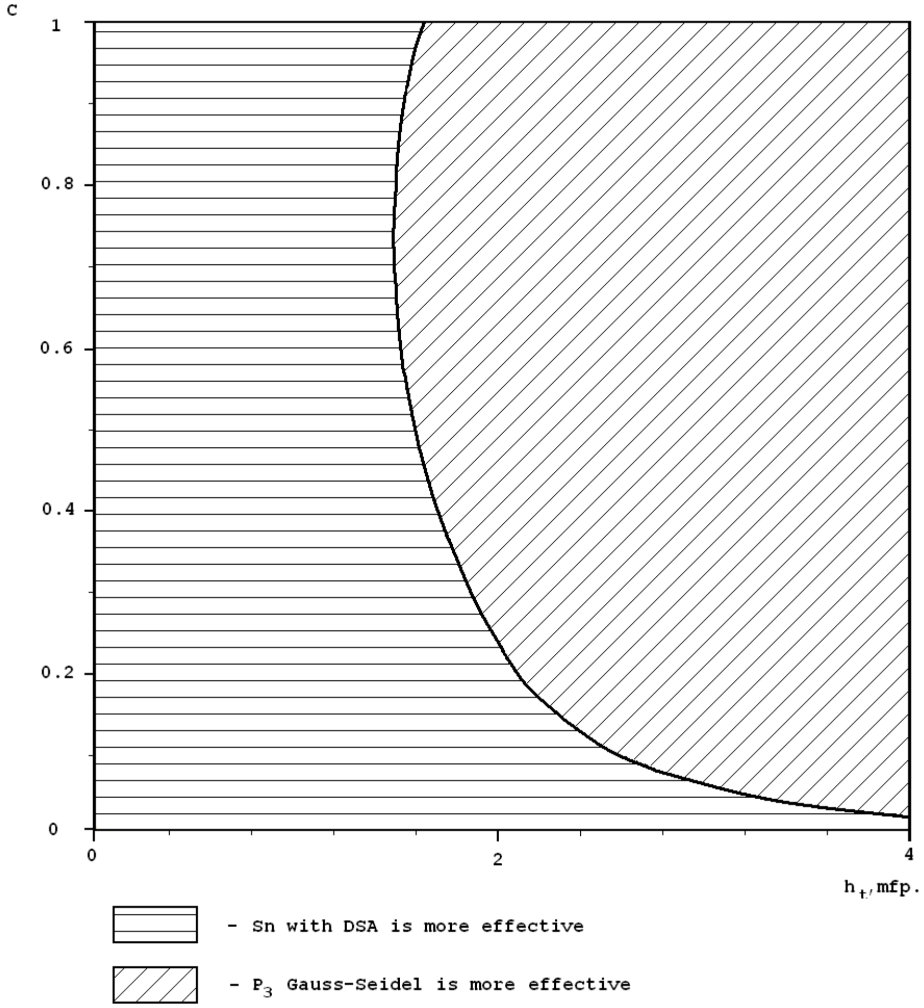

Fig. A7. Regions in (h_t,c) phase space where the spectral radius of particular scheme is smaller.

The computational results for the spectral radius can be acquired by measuring the value

$$\rho_i \equiv \frac{\sum\varphi_j^{(i+1)} - \sum\varphi_j^{(i)}}{\sum\varphi_j^{(i)} - \sum\varphi_j^{(i-1)}}. \qquad \text{(A.7)}$$

after each iteration. If the scheme implemented correctly, ρ_i converges to the theoretical prediction for the spectral radius. The theoretical results and measured spectral radius values for the implemented Gauss-Seidel scheme is shown below:

Table A1

Measured Spectral Radius for Gauss-Seidel P_3 scheme

	c = 0		c = 0.5		c = 1	
	Experiment	Theory	Experiment	Theory	Experiment	Theory
$h_t = 10^{-6}$	0.5091	0.5091	0.5091	0.5091	0.5091	0.5091
$h_t = 1$	0.1910	0.1969	0.2439	0.2506	0.3371	0.3446
$h_t = 10$	$1.366 \cdot 10^{-4}$	$1.375 \cdot 10^{-4}$	$2.574 \cdot 10^{-4}$	$2.713 \cdot 10^{-4}$	0.0109	0.0104

As can be seen from the table A1, there is a good correlation between the theoretically predicted spectral radius and its measured numerical approximation. This fact proves that the implementation of the P_3 Gauss-Seidel scheme is correct.

APPENDIX B

VERIFICATION OF THE SN/DIFFUSION AND SN/P_3 HYBRID SCHEMES

The main approach to verification of an implementation of a particular numerical solition technique is the concept of a benchmark solution. The idea is to compare the resuls of the numerical simulations to the analytical solution. If the scheme doesn't produce the anticipated result, then the implementation is incorrect.

The first problem is a pure scattering medium with no distributed sources present and incoming current on the left boundary of the S_N region. The theory predicts that for such a problem the spatial distribution of scalar flux looks like a straight line. On the extrapolated boundary where the boundary source is present, the numerical angular fluxes must be close to $\frac{\varphi}{4\pi}$, where φ is the scalar flux. On the side that doesn't containe a source, the scalar flux must be close to 0 on the extrapolated bpundary.

For the second case, the uniform crossections and distributed sources result in the flat scalar flux distribution in a relatively thick medium. The theory predicts that for such problem the spatial distribution of scalar fluxes will have a plato in the central region of the medium. The distance between an edge of the central region and the nearest boundary is about 3 mean free paths. The magnitude of the scalar flux will not exceed $\left(\frac{q}{\sigma_a}\right)$, where q is the distributed source and σ_a is the macroscopic cross section.

The verification of the code has been performed for the following parameters of the 100:100 cells problem, total thickness of each region is 10 mean free paths and total cress section σ_{tot} = 1 cm^{-1} in both regions. Below the results are shown for two constructed solutions. On the figure the plots for the computational results are shown and look like expected.

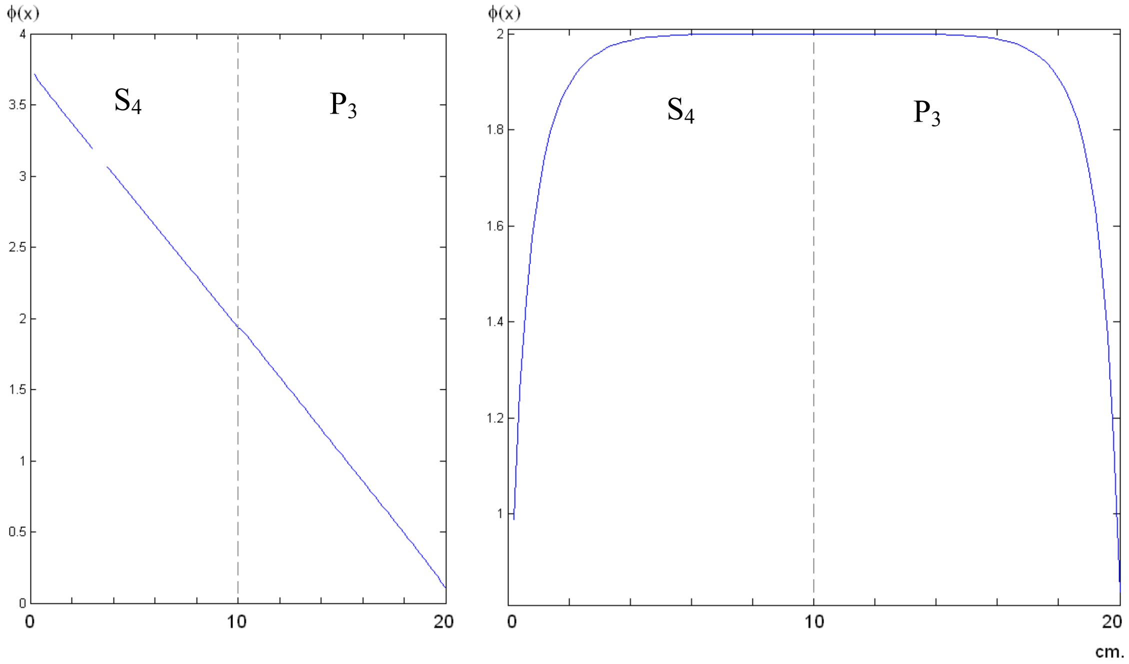

Fig.B1. Result for the scalar flux in an S_4 region and P_3 region.

From the figure B1 it can be seen that the numerical solution for the two benchmark problems behave exactly as it had been predicted by the theory. For the Sn/Diffusion problem the plots are similar.