World Academy of Science, Engineering and Technology
International Journal of Physical and Mathematical Sciences
Vol:5, No:5, 2011

A Parallel Algorithm for 2-D Cylindrical Geometry Transport Equation with Interface Corrections

Wei Jun-xia[1] , Yuan Guang-wei , Yang Shu-lin , Shen Wei-dong

***Abstract*—**In order to make conventional implicit algorithm to be applicable in large scale parallel computers , an interface prediction and correction of discontinuous finite element method is presented to solve time-dependent neutron transport equations under 2-D cylindrical geometry. Domain decomposition is adopted in the computational domain.The numerical experiments show that our parallel algorithm with explicit prediction and implicit correction has good precision, parallelism and simplicity. Especially, it can reach perfect speedup even on hundreds of processors for large-scale problems.

***Keywords*—**Transport Equation ; Discontinuous Finite Element ; Domain Decomposition ; Interface Prediction And Correction

Open Science Index, Physical and Mathematical Sciences Vol:5, No:5, 2011 publications.waset.org/8326/pdf

I. Introduction

TIME-dependent neutron transport equation is a kind of important differential equation in nuclear and scientific application. High dimension particle transport calculation is very complex and huge scale scientific calculation problem. The 2-D cylindrical geometry time-dependent neutron transport equations are actually six dimensional, i.e.: two dimensions for the independent variables (x,r) in the geometry space; two dimensions for the independent variables (ξ,μ) or (ξ,ω) in the neutron direction space; one dimension for group g and one dimension for time t, with a large amount of computation and storage. To complete the calculation of time-dependent neutron transport problem, need high performance computer and efficient calculation method. The rapid development of large scale parallel computer has offered favourable condition for efficient parallel algorithm research of high dimension particle transport equations.

Along with the development of parallel computers and parallelizing techniques, the research in neutron transport parallel calculation had got great progress[4-13]. The transport equation calculation include computing of space grid ,angle direction and energy group , The parallelism of angular domain decomposition for Sn approximation is restricted by the number of discrete ordinates, similarly the scalability of parallel algorithm about energy variable is restricted by the number of energy groups for multigroup transport equation. Only base on geometry spatial domain decomposition, then we can realize scalable parallel calculation for two dimension neutron transport problems[7 , 8]. However, the parallel degree of geometry space is limited for the implicit discrete method, since the system of discrete ordinates equation on a cell should not be solved for a particular direction, until these discrete ordinates equations have been solved on all their 'upstream' neighbors. Numerical experiments demonstrate that when the domain number increasing to certain scale, the latency times for idle are longer and parallel efficiency drops. Evidently, for the implicit discrete method of neutron transport problems, the strong data correlation and algorithmic serial essentiality become bottleneck problems of efficient parallel calculation. Therefore, we must research new parallel algorithm to be applicable in large scale parallel computers.

For time-dependent neutron transport equations under 2-D cylindrical geometry, we presented a new parallel algorithm with interface prediction and correction of discontinuous finite element method based on domain decomposition. On the interface , an upwind explicit scheme is applied to give an incident boundary condition ,which enables the subdomain problem iterated independently[9,10]. The interface values are updated by an implicit scheme concurrently in iteration. The scheme shows good precision , parallelism and simplicity in numerical experiments.

The remainder of this paper is organized as follows. In Section 2, the discontinuous finite element method is presented for time-dependent 2-D cylindrical geometric transport equation. In Section 3, we describe the construction of interface prediction and correction method based on spatial domain decomposition. In Section 4, we provide numerical results for parallel computing of 2-D spherical cylindrical geometric transport equation. In the final section, we offer summary and conclusions.

II. 2-D Cylindrical Geometric Transport Equations

The time-dependent, mlui-group neutron transport equations under 2-D cylindrical coordinate are written as follows:

$$\frac{1}{V_g}\frac{D\varphi_g}{Dt}+\frac{\mu}{r}\frac{\partial}{\partial r}(r\varphi_g)+\xi\frac{\partial\varphi_g}{\partial x}-\frac{1}{r}\frac{\partial}{\partial\omega}(\zeta\varphi_g)+\sigma_g^{tr}\varphi_g=Q_{sg}+Q_{fg}+Q_{dg}$$
$$g=1,2,\cdots,G \qquad (1)$$

Where

φ_g is the particle angular flux,

$\varphi_g=\varphi_g(x,r,\xi,\omega,t)$; $(x,r)\in D$; $-1\le\xi\le 1$; $0\le\omega\le\pi$; $0\le t\le T_{max}$

G is the group number, D is system region, V_g is neutron velocity, σ_g^{tr} is the total macroscopic cross-section,

[1] Corresponding author. Institute of Applied Physics and Computational Mathematics , Beijing 100088 , China
E-mail addresses: wei_junxia@iapcm.ac.cn
Supported by the National Basic Research Program under the Grant 2011CB309705, Project 2008A0202008 supported by CAEP .

Open Science Index, Physical and Mathematical Sciences Vol:5, No:5, 2011 publications.waset.org/8326/pdf

$\mu = \sqrt{1-\xi^2}\cos\omega$ and $\zeta = \sqrt{1-\xi^2}\sin\omega$ are angular variables, Q_{fg}, Q_{sg}, Q_{dg} are fission source , scattering source and external source respectively ,

Fission source: $Q_{fg} = \sum_{g'=1}^{G} \chi_{g'\to g}(\nu\sigma)_{g'}^{f}\phi_{0,g'}^{0}(x,r)$

Scatter source: $Q_{sg} = \sum_{l=0}^{L}\sum_{g'=1}^{g}\sigma_{g'\to g}^{s,l}\sum_{k=0}^{l}R_l^k(\xi,\omega)\phi_{l,g'}^{k}(x,r)$

$$\phi_{l,g}^{k} = \frac{1}{2\pi}\int_{-1}^{1}d\xi\int_{0}^{\pi}R_l^k(\xi,\omega)\varphi_g(x,r,\xi,\omega)d\omega$$

For the equations (1), the free boundary conditions

$$\varphi_g(x,r,\xi,\mu,t)|_{\Gamma_\Omega} = 0$$

is given on neutron-inward boundary Γ_Ω of each direction $\vec{\Omega}$, where $\Gamma_\Omega = \{(x,r)\in\Gamma;\ \vec{\Omega}\cdot\vec{n}_\Gamma < 0\}$

Γ is the outer boundary of the calculated region.

The discretizing way is:

1) Time discretization: Implicit difference scheme.

2) Direction discretization: Sn discrete ordinate method.

3) Geometry discretization: Discontinuous finite element method.

The discrete equations on a triangle or quadrangle geometry grid are:

$$\begin{aligned}&\sum_{i=1}^{I}\varphi_{g,m,s,i}^{(\nu+1)}\left\{\mu_{m,s}(\frac{\partial N_i}{\partial r},N_j)+\xi_s(\frac{\partial N_i}{\partial x},N_j)+(\mu_{m,s}+\gamma_{m,s})(\frac{N_i}{r},N_j)\right.\\&\left.+(\sigma_g^{tr}+\frac{2}{V_g\Delta t^{n+\frac{1}{2}}})(N_i,N_j)-<N_i,N_j>\right\}\\&=\sum_{i=1}^{I}\left\{(N_i,N_j)\left[\sum_{l=0}^{L}\sum_{g'=1}^{g}\sigma_{g'\to g}^{s,l}\sum_{k=0}^{l}R_l^k\phi_{l,g',m,s,i}^{k,(\nu)}+\sum_{g'=1}^{G_0}\chi_{g'\to g}(\nu\sigma)_{g'}^{f}\phi_{g',m,s,i}^{0,(\nu)}+Q_{dg}+\frac{2}{v_g\Delta t^{n+\frac{1}{2}}}\varphi_{g,m,s,i}^{n}\right]\right.\\&\left.+\delta_{m,s}\frac{\beta_{m+\frac{1}{2},s}+\beta_{m-\frac{1}{2},s}}{w_{m_s,s}}(\frac{N_i}{r},N_j)\varphi_{g,m+\frac{1}{2},s,i}^{(\nu+1)}\right\}-<\bar{\chi}_{g,m,s}^{(\nu+1)},N_j>\end{aligned}\quad(2)$$

Where

$$\gamma_{m,s} = \begin{cases}-\mu_{m,s}, & m_s = M_{s+\frac{1}{2}}, s=1,2,...,N\\ \dfrac{2\beta_{m-\frac{1}{2},s}}{w_{m,s}}, & m=1,2,...,M_s \quad s=1,2,...,N\end{cases},$$

$$\delta_{m,s} = \begin{cases}0, & m=M_{s+\frac{1}{2}}; s=1,2,...,N\\ 1, & m=M_s,...,1; s=1,2,...,N\end{cases},$$

$$\beta_{m\pm\frac{1}{2},s} = \frac{1}{2\pi}\Delta\xi_s\varsigma_{m\pm\frac{1}{2},s}$$

The scalar product of volume integral

$$(A,B) = \iint_{D_p} rABdxdr$$

The scalar product of area integral

$$<A,B> = \oint_{\partial^- D_p}\vec{\Omega}\cdot\vec{n}\ rABds = \oint_{\partial^- D_p} rAB(\xi dr - \mu dx)$$

$\partial^- D_p$ is the neutron-inward boundary of the geometry grid D_p corresponding with the direction $\vec{\Omega}$.

$N_i (i=1,\cdots,I)$ are the basis functions (linear or bilinear) on a certain geometry grid.

ν is iterative degree, and the time superscript $n+\frac{1}{2}$ is omitted.

Formula (2) can be abbreviated to :

$$A(\varphi_{g,m,s,k}^{n+\frac{1}{2}(\nu+1)}) = Q_{f+s}(\Phi_{g,k}^{n+\frac{1}{2}(\nu)}) + B\left(\varphi_{g,m+\frac{1}{2},s,k}^{n+\frac{1}{2}(\nu+1)}\right) + C\left(\bar{\chi}_{g,m,s,k}^{n+\frac{1}{2}(\nu+1)}\right) + E \quad (3)$$

Where k is grid index, E is the term independent of $n+\frac{1}{2}$ row neutron flux.

The brief course of the calculation for 2-D discontinuous finite element (DFE) equations: For each direction, the grid queue must be arranged by $\vec{\Omega}_{ms}\cdot\vec{n} < 0$, and the solving according to the queue proceeds. Thus, when a grid is in calculation, the inflow segment of the boundary is either a part of outer boundary or a segment of the boundary belonging to the calculated neighbor grid.

The boundary condition for each grid is from either outer boundary condition or flux on the corresponding inflow grid.

For each energy group, each SN discrete direction and each grid , only an I-order (I=3 or 4) algebraic equation is needed to solve.

III. Parallel Algorithm with Interface Prediction and Correction

According to the continuity condition for neutron flow, the inflow flux of a mesh equal to the corresponding outflow flux of neighbor meshes. It is obvious that, for each angular direction, at first, only these finite elements located at the boundary of domain with known boundary inflow flux can be solved (See Figure 1).

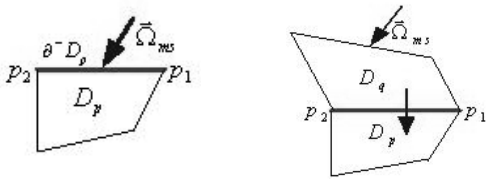


Fig. 1 The boundary relation of meshes

For an inner subdomain, as Figure 2 shows, the solving is dependent on its "upper elements" D1 and D2. Only after solving the DFE equations on D1 and D2, the DFE equations on D can be solved.

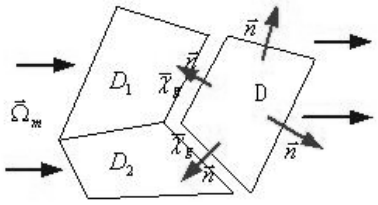


Fig. 2 The boundary relation of subdomains

Thus, it is inevitable that there is the data interrelation among the neighboring meshes. In order to resolve the exchange of message among the processors, a layer of assistant meshes is added along the boundary of each subdomain to store the flux message on these meshes of the neighbor subdomains.

For a certain discrete direction $\vec{\Omega}_m$, the solving sequence of meshes is determined by the flow-line of neutrons. Meanwhile, aimed at the 2-D cylindrical geometry, there are the dependent relations among the directions. Then the parallel SN sweep algorithm is designed, the main steps of the algorithm are:

1) Calculate the in-degree for each cell, and insert the cell with zero in-degree into a queue according to some inserting algorithm.

2) Extract one cell from the queue. For its element and direction, for all groups, solve Equation (3), then subtract 1 from the in-degree of each downstream cell.

3) Send the flux of the cell to the processor owning the downstream cell.

4) Receive all messages from other processors.

5) Insert new cells with zero in-degree into the queue.

For a given direction $\vec{\Omega}_{ms}$,when compute on the cell D, the angle flux inflow to D must be known, i.e. $\overline{\chi}^{n+\frac{1}{2}(\nu+1)}_{g,m,s,k}$ must be known, it is solved in current iterative, so compute on the cell D must wait for the computing on D_1 and D_2. If D is in the different domain with D_1 and D_2, then we must transmit $\overline{\chi}^{n+1/2(\nu+1)}_{g,m,s,D1}$ and $\overline{\chi}^{n+1/2(\nu+1)}_{g,m,s,D2}$ to the domain that include D by communication.

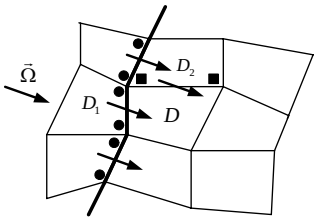


Fig. 3 Incidence boundary of sub-domain

So, for each sub-domain, the data relativity of grids is represented by $\overline{\chi}^{n+\frac{1}{2}(\nu+1)}_{g,m,s,k}$ of current iterative, then we can adopt interface prediction and correction method for inflow $\overline{\chi}^{n+\frac{1}{2}(\nu+1)}_{g,m,s,k}$ of the domain boundary cells, .ie., for the domain boundary cells D, if the 'upstream' cell D_1 are in the different domain ,but the 'upstream' cell D_2 are in the same domain with D(Fig.3), substitute $\overline{\chi}^{n+\frac{1}{2}(\nu+1)}_{g,m,s,D}$ with $\overline{\chi}^{*}_{g,m,s,D}$, then (3) become

$$A(\varphi^{n+\frac{1}{2}(\nu+1)}_{g,m,s,D}) = Q_{f+s}(\Phi^{n+\frac{1}{2}(\nu)}_{g,D}) + B\left(\varphi^{n+\frac{1}{2}(\nu+1)}_{g,m+\frac{1}{2},s,D}\right) + C\left(\overline{\chi}^{*}_{g,m,s,D}\right) + E \tag{4}$$

$$\overline{\chi}^{*}_{g,m,s,D} = \overline{\chi}^{*}_{g,m,s,D_1\to D} + \overline{\chi}^{*}_{g,m,s,D_2\to D}$$

We use originally implicit method (3) to compute $\varphi^{*,n+\frac{1}{2}(\nu+1)}_{g,m,s,D_2}$ on D_2 to get $\overline{\chi}^{*}_{g,m,s,D_2\to D}$:

$$A(\varphi^{n+\frac{1}{2}(\nu+1)}_{g,m,s,D_2}) = Q_{f+s}(\Phi^{n+\frac{1}{2}(\nu)}_{g,D_2}) + B\left(\varphi^{n+\frac{1}{2}(\nu+1)}_{g,m+\frac{1}{2},s,D_2}\right) + C\left(\overline{\chi}^{*}_{g,m,s,D_2}\right) + E$$

For $\overline{\chi}^{*}_{g,m,s,D_1\to D}$, we use interface prediction and correction method:

1) Prediction: for the first iterative of t^{n+1} time layer, solve the flux $\varphi^{*,n+\frac{1}{2}(\nu+1)}_{g,m,s,D_1}$ on D_1 by using explicit method to get $\overline{\chi}^{*}_{g,m,s,D_1\to D}$:

$$A(\varphi^{*,n+\frac{1}{2}(\nu+1)}_{g,m,s,D_1}) = Q_{f+s}(\Phi^{n}_{g,D_1}) + B\left(\varphi^{*,n+\frac{1}{2}(\nu+1)}_{g,m+\frac{1}{2},s,D_1}\right) + C\left(\overline{\chi}^{n}_{g,m,s,D_1}\right) + E \tag{5}$$

2)Correction: after the fisrt iterative of t^{n+1} time layer, solve the flux $\varphi^{*,n+\frac{1}{2}(\nu+1)}_{g,m,s,D_1}$ on D_1 by using implicit scheme to get $\overline{\chi}^{*}_{g,m,s,D_1\to D}$ for correction.

$$A(\varphi^{*,n+\frac{1}{2}(\nu+1)}_{g,m,s,D_1}) = Q_{f+s}(\Phi^{n+\frac{1}{2}(\nu)}_{g,D_1}) + B\left(\varphi^{*,n+\frac{1}{2}(\nu+1)}_{g,m+\frac{1}{2},s,D_1}\right) + C\left(\overline{\chi}^{n+\frac{1}{2}(\nu)}_{g,m,s,D_1}\right) + E \tag{6}$$

Then, the strategy of the parallel method with interface prediction and correction of discontinuous finite element method to solve 2-D cylindrical geometry time-dependent neutron transport equations is as follows

Open Science Index, Physical and Mathematical Sciences Vol:5, No:5, 2011 publications.waset.org/8326/pdf

Open Science Index, Physical and Mathematical Sciences Vol:5, No:5, 2011 publications.waset.org/8326/pdf

Step 1. Given the discrete initial value and incident flux on boundary;

Step 2. Divide the physical space region into nonoverlapping sub-domains and note the information of boundary grids ;

Step 3. Arrange the grid solving queue according to each angle direction in each sub-domain, and get the boundary grids' relations of neighbouring domains;

Step 4. Compute the downflow flux values on the boundary cells of each sub-domain by explicit prediction scheme (5);

Step 5. Send the flux of the cell to the processor owning the downstream cell, and receive all messages from other processors;

Step 6. Solve the implicit equation (3) using explicit prediction values as inflow boundary conditions to get $\varphi_{g,m,s,k}^{n+\frac{1}{2}(\nu+1)}$ on each sub-domain in parallel;

Step 7. The iterative procedure stop when the iterative precision at t^{n+1} time layer is satisfied, else go to step 8;

Step 8. Compute the downflow flux values on the boundary cells of each sub-domain by implicit scheme (6), go to step 5.

IV. Numerical results

Two test problems in 2-D cylindrical geometry are calculated on large parallel computers. The first problem has an analytic solution which is related to time variable only. The second problem describes a typical transport problem without known analytic solution.

4.1. Test 1

Given an analytic solution on a sphere as follows:

$\varphi_h = e^{-t}$ with the source item

$$Q_{dg} = (-\frac{1}{v_g} + \sigma_g^{tr})\varphi_h - Q_{fd}(\varphi_h) - Q_{sd}(\varphi_h).$$

This problem is about a pure absorber.

We assume the macroscopic total cross-section σ_g^{tr} to be 1 cm and particle velocity to be $10^3 cm/\mu s$, and use the S4 quadrature set and the number of spatial grid is 16384. The computing end time T is $0.1\mu s$

Let $S_{g,m,s,k,i} = (\varphi_{g,m,s,k,i} - \varphi_{h,g,m,s,k,i}) / \varphi_{h,g,m,s,k,i}$

Then define the maximum error:

$$Maxerr = \max_{g,m,s,k,j} \{ | S_{g,m,s,k,j} | \}$$

the average error:

$$Average-err = \sqrt{\frac{\sum_{g,m,s,k,i} S^2_{g,m,s,k,i}}{4 \times Net \times \max ms \times G_0}}$$

The numerical results of numerical precision and the speedup are presented in Table 1. The parallel computations are implemented individually with 1, 16, 32, 64, 128, 256 and 512 processors. From these data we can observe that the parallel numerical precision of the interface prediction and correction of discontinuous finite element method maintain the precision of single processor. From 1 to 512 processors, the new method can reach exceed linear speedup profiting from the Cache .

TABLE I
NUMERICAL RESULTS FOR PROBLEM 1

CPU	Maxerr ($\times 10^{-4}$)	Average-err($\times 10^{-6}$)	Iterative number	compute time (s)	speed up
1	0.1535	0.8680	2000	47160	1.0
16	0.1535	0.8680	2000	1790	26.3
64	0.1535	0.8680	2000	414.5	113.8
128	0.1535	0.8680	2000	202.6	232.8
256	0.1535	0.8680	2000	100.3	470.2
512	0.1535	0.8680	2000	64.89	726.8

4.2. Test 2

This problem is a time-dependent sphericity model , the material is uranium.

$r = 8.809cm$, $\rho = 18.71\, g/cm^3$,

$n^{235} = 0.247286 \times 10^{-2} / g$,

Define λ is the system neutron increasing constant of time:

$$\lambda = \frac{\frac{d}{dt}(\sum_{g=1}^{G} N_g)}{\sum_{g=1}^{G} N_g} ,$$

$$N_g = \Phi_g / V_g , \quad \Phi_g = 2\pi \sum_k \int_{D_k} \int_0^{2\pi} \int_{-1}^{1} \varphi_g r d\xi d\omega dx dr$$

Where N_g is the number of group g and time t

Define relatively error is $err = abs\left(1 - \lambda_{new} / \lambda_{old}\right)$

The numerical results are presented in Table 2.The compute region is divided into 16384 spatial grids. The parallel computations are implemented individually with 1, 16, 32, 64, 128, 256 and 512 processors. For this problem, the parallel numerical precision of the interface prediction and correction of discontinuous finite element method maintain the precision of single processor. The speedups, which increase exceed linearly with the number of processors, are shown as Table II.For 512 processors, the new method causes some degration in the rate of convergence,but the fractional increase in the num ber of iterations it takes to converge is usually substantially smaller than the relative speedup achieved by simultaneously beginning sweeps from all the sub-domains.

TABLE II
NUMERICAL RESULTS FOR PROBLEM 2

CPU number	λ	err	Iterative number	compute time (s)	speed up
1	-0.25298	--	1000	26418.5	1.0
16	-0.25286	0.05%	1000	813.7	32.5
64	-0.25265	0.13%	1000	184.4	143.3
128	-0.25223	0.29%	1000	87.0	303.7
256	-0.25208	0.35%	1000	44.54	593.1
512	-0.25305	0.03%	1082	32.7	807.9

V. SUMMARY AND CONCLUSIONS

This paper is motivated by the need to obtain precise and efficient solutions to time-dependent 2-D cylindrical geometric transport equation in large scale parallel computers . We have presented a parallel algorithm that uses a combination of spatial decomposition with interface prediction and correction of discontinuous finite element method. On the interface , an upwind explicit scheme is applied to give an incident boundary condition ,which enables the subdomain problem iterated independently. The interface values are updated by an implicit scheme concurrently in iteration. Moreover, the parallel method is easy to reach load balance, and needs only local communication. Numerical results shows good precision , parallelism and simplicity for 2-D cylindrical geometric transport equation. Especially, it can reach perfect speedup even on hundreds of processors for large-scale problems.The further study will aim at constructing higher order numerical scheme to improve computational precision and efficiency with less computing time for interface prediction and correction.

REFERENCES

[1] [1] W.H. Reed and T.R. Hill, Triangular mesh methods for the neutron transport equation,
[2] Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
[3] [2] F.Lianxiang and Y.Shulin. Researches on 2-D neutron transport solver NTXY2D. Technical report, Institute of Applied Physics and Computational Mathematics in Beijing, 1999.10.
[4] [3] E.E. Lewis, W.F. Miller. Computational Methods of Neutron Transport [M]. New York: John Wiley & Sons Publisher, 1984.
[5] [4] R.S.Baker, K.R.Koch. An Sn algorithm for the massively parallel CM-200 computer [J]. Nuclear Science and Engineering, 1998, Vol.128 : 312-320.
[6] [5] S.Plimpton, B.Hendrickson, S.Burns, W.McLendon. Parallel algorithms for radiation transprt on unstructured grids [A]. Proceeding of SuperComputing'2000.
[7] [6] Shawn D Pautz. An Algorithm for Parallel Sn Sweeps on Unstructured Meshes [J]. Nuclear Science and Engineering ,2002 ,140 (2) :111-136.
[8] [7] Mo Z, Fu L. Parallel flux sweeping algorithm for neutron transport on unstructured grid. Journal of Supercomputing, 2004, 30(1): 5-17.
[9] [8] WEI Jun-xia , YANG Shu-lin, FU Lian-xiang. A Parallel Domain Decomposition Method for Neutron Transport Equations Under 2-D Cylindrical Geometry[J]. Chinese J Comput Phys, 2010, 27(1):1-7.
[10] [9] Yuan Guangwei, Hang Xudeng. A parallel algorithm for the particle transport SN method with interface corrections [J]. Chinese J Comput Phys, 2006, 23(6):637-641.
[11] [10] Zhenying Hong,Guangwei Yuan, Aparallel algorithm with interface prediction and correction for spherical geometric transport equation. Prog.Nucl.Energy,51,268-273(2009).
[12] [11] Zhang Aiqing , Mo Zeyao. Parallelization of the 2D multi-group radiation transport code LARED-R-1 [J]. Chinese J Comput Phys, 2007, 24(2): 146-152.
[13] [12] Yang Shulin, Mo Zeyao, Shen Longjun. The Domain Decomposition Parallel Iterative Algorithm foe the 3-D Transport Issue [J]. Chinese J Comput Phys, 2004, 21(1): 1-9.
[14] [13] T.A. Wareing, J.M. McGhee, J.E. Morel and S.D. Pautz, Discontinuous Finite Methods Sn Methods on 3-D unstructured Grids, in Proceeding of International Conference on Mathematics and Computation, Reactor Physics and Environment Analysis in Nuclear Applications, 1999, Madrid, Spain.
[15] [14] SWEEP3D:3D Discrete Ordinates Neutron Transport Benchmark Codes, http://www.llnl.gov/asci_benchmarks/asci/limited/sweep3d/sweep3d_readme.html.

Open Science Index, Physical and Mathematical Sciences Vol:5, No:5, 2011 publications.waset.org/8326/pdf