An Experiment of Robust Parallel Algorithm for the Eigenvalue problem of a Multigroup Neutron Diffusion based on modified FETI-DP

Jonghwa Chang
Korea Atomic Energy Research Institute,989-111 Daedeokdaero, Yuseong-gu, Daejeon, Korea, 305-353
jhchang@kaeri.re.kr

1. Introduction

Today, we can build a computer cluster of few hundreds CPUs with reasonable budget. To take advantage of such a cluster, efficient parallel algorithms must be developed. Parallelization of Monte Carlo simulation is widely adpoted. There are also several parallel algorithms developed for the SN transport theory using the parallel wave sweeping algorithm and for the CPM using parallel ray tracing. For practical purpose of reactor physics application, the thermal feedback and burnup effects on the multigroup cross section should be considered. In this respect, the domain decomposition method(DDM) is suitable for distributing the expensive cross section calculation work. Parallel transport code and diffusion code based on the Raviart-Thomas mixed finite element method was developed. However most of the developed methods rely on the heuristic convergence of flux and current at the domain interfaces. Convergence was not attained in some cases.

Mechanical stress computation community has also work on the DDM to solve the stress-strain equation using the finite element methods. The most successful domain decomposition method in terms of robustness is FETI-DP. [1] We have modified the original FETI-DP to solve the eigenvalue problem for the multigroup diffusion problem in this study.

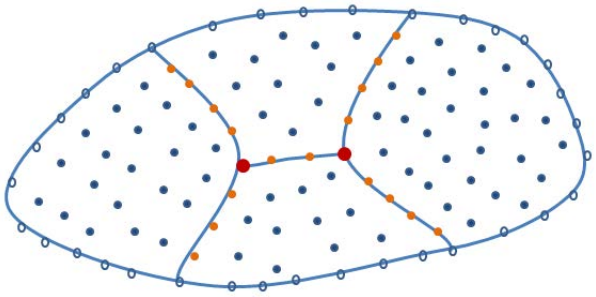

Figure 1. Domain Decomposition and Node Map
(filled: inner, open: boundary, brown: dual, red: primal)

2. Modified FETI-DP

Finite Element Tearing and Interconnection (FETI) – DP (Dual Primal) method is an idea to tear (partition) the large matrix into several sub-matrices. Lagrange multiplier is introduced to interconnect the interface points (dual points). Points at the interfaces where three or more sections are met, are defined as primal points. A global equation, depends on the Lagrange multiplier, can be established to find the value at the primal points. In Figure 1, there are 2 primal points where the value is to be solved globally. The iteration will be continued until convergence in the Lagrange multiplier is achieved. Detailed procedure is described on Farhat et al. [1]. Original FETI-DP assumes a symmetric matrix system, but it can be modified to non-symmetric system such as multi-group up-down scattering diffusion problem easily.

3. Multigroup Diffusion Equation

2.1 Prescription of Algorithm

The multigroup diffusion eigenvalue problem is written in following form.

$$\left(D_g\nabla^2+\sigma_g\right)\phi_g=\lambda\chi_g s+\sum_{g'\neq g}\sigma_{g'\to g}\phi_{g'}$$

where $s=\sum_g \nu\sigma_{fg}\phi_g$

A general eigenvalue problem can be setup to solve above elliptic equation by using appropriate method such as finite element method in following form.

$$A\phi=\lambda B\phi$$

The matrix A is a positive definite real symmetric matrix for each group. For typical slowing down only problem, source iteration method can be used and sub matrix for each group can be solved easily by Cholesky decomposition, i.e, using the positive definite symmetric property. However, for the problems such as gas cooled reactor in which up-scattering is also important, the source iteration strategy converges very slowly.

We considered more general approach considering that the matrix A is a very large sparse matrix. There are several methods known to deal with the eigenvalue problem of sparse matrix system. [2] Approach to solve the eigenvalue problem, by Krylov subspace method, is generating a finite set of orthogonal vectors and compose eigenvector which satisfy maximum (or minimum) eigenvalue. The eigenvalue and corresponding eigenvector in the Krylov space is obtained by solution of a linear system which can be solved by direct method. We have used LU decomposition method using GSL. [3] This cycle repeats until convergence in the eigenvector is achieve. However normal Gram-Schmidt normalization method

(GMRES) reveals oscillation problem due to amplification of the numerical truncation error. To avoid such problem, we adopted more robust Householder projection method to generate the sequence of orthogonal vectors. [4] Original eigenvalue problem is converted to the inverse eigenvalue problem for faster and stable convergence.

$$A^{-1}B\phi = \frac{1}{\lambda}\phi$$

The Krylov iteration requires a product of the matrix for eigenvalue problem or a solution of a linear system for the inverse eigenvalue problem;

$$A\varphi = By$$

Above linear system is solved by FETI-DP parallel algorithm. The global equation on the Lagrange multiplier will be established by collecting results at each subdomain. To solve the resulting linear system of Lagrange multipliers, Farhat [5] suggested PCG (preconditioned conjugate gradient), which use single residual vector, for the stress strain problem, but we found PBiCGSTAB (preconditioned bi conjugate gradient stabilized), which use two residual vector, converges faster for the diffusion problems. PCG or PBiCGSTAB requires solution for inverse eigenvalue problem (or matrix product for eigenvalue problem). This problem can be solved by FETI-DP with DDM. FETI-DP requires solution of the multigroup equation in each subdomain (or partition). This problem is solved by the block successive over-relaxation (bSOR) method [6] with Cholesky decomposition method [7] for each group. Cholesky decomposed matrix can be prepared ahead of iteration.

Suitable preconditioner matrix for the primal variable is required for PBiCGSTAB (or PCG) algorithm. A lumped preconditioner suggested by Farhat [5] was satisfactory.

2.2 Domain Decomposition

Algorithm described above requires conformal geometry at the interface. The continuity of flux values at the subdomain interface are satisfied through the Lagrange multipliers and that at the corner point where more than 3 subdomain, are satisfied by a global equation on the flux. Total computing time is sum of 1) the CPU time spent to solve the global equation for primal nodes and the Krylov iteration for dual nodes, 2) the highest CPU time spent to solve the FEM for internal points and dual points, and 3) the time required to exchange the interface values with global equation. So, we can choose following criteria for minimum computing time with given number of subdomains.

a. Number of internal and dual points are even,
b. Total number of dual points is small,
c. Total number of primal points is small.

This is an NP optimization problem in the graph theory. Criterion a means to minimize the supremum of number of internal and dual points of subdomain. Criterion b is sum of number of dual points. We can impose appropriate weighting factor to define the cost of segmentation. This is a NP problem in graph theory. For 3D parallel pipe geometry, we separate planar direction and axial direction to simplify partitioning.

2.3 Experiment with IAEA3D benchmark problem

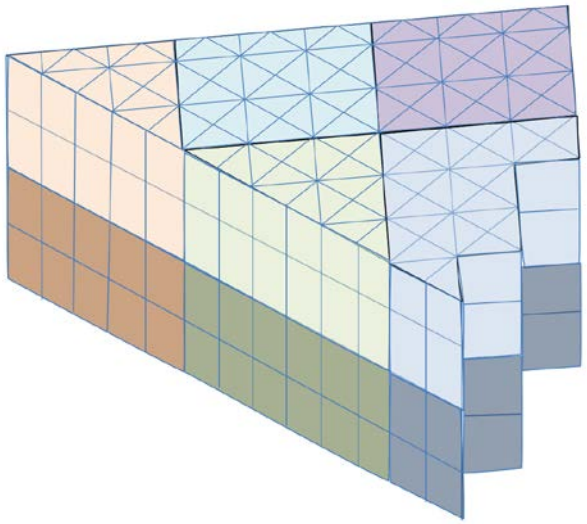

Figure 2. Partitioning of IAEA 3D benchmark

For experiment with IAEA3D benchmark problem, we also made simple partition strategy to have similar number of assemblies in a subdomain. As a result we have 10 subdomains with 5 planar and 2 axial partitions shown as in Figure 2. 11 CPUs was used in this experiment to know the effect of CPU speed and LAN speed. Table 1 summarizes the consumed time and data transfer between CPUs.

Table 1. Summary of IAEA3D benchmark problem

ns/nord/ nz/nordz	nprim/ndual/ np3glob	MPI s/r (MWords)	User(s) min/max	Real time(s)
1/1/19/1	66/720/10160	9.1/16.4	19/59	65
1/1/19/2	104/1213/19812	15.9/28.9	59/198	208
1/2/19/1	94/1960/39600	29.3/54.7	218/1006	1080
2/1/38/1	132/2986/77220	48.5/90.6	617/2126	2151
2/1/38/2	208/5038/152460	101.6/191.2	2908/9348	9627
2//12/38/1	188/7976/304824	140.7/265.3	9048/31261	31909

In above table, the first column shows ns and nz which are the number of grids per assembly, nord and nordz are the order of Legendre polynomial. Second column shows nprim, ndual, and np3glob which are the number of primal, dual, and total global points, respectively. The total global points are the number of finite element nodes multiplied by the number of energy group. Third column shows the number of data words transferred between the processors. These amounts are proportional to the number of interface points (primal and dual) and the number of FETI-DP iterartions. Fourth column shows minimum and maximum of CPU times used by each process. Last column shows actual CPU time used to solve the problem. It should be notified that the actual CPU time is similar to the maximum CPU time used by a process. The effect of

network communication is not signification in our experiment with Gigabit Ethernet cluster.

3. Conclusion and Further Study

FETI-DP algorithm developed for a simple tree network configuration did not show significant effect in the Ethernet configuration which is most widely deployed cluster configuration.

Even distribution of computational load (criterion a) is more important. Domain Decomposition into given number of CPU is an NP optimization problem of graph partitioning in graph theory. There are several general computer code such as Chaco [8], METIS [9], etc. Further study is required to find an optimum partition.

During the experiment, we found a convergence problem in the bSOR iteration to solve the multigroup equation in a segment after several Householder iterations. There may be more robust method to solve the segment-wise multigroup problem.

REFERENCES

[1] C. Farhat, M. Lesoinne, and K. Pierson, "A scalable dual-primal domain decomposition method," *Numer. Linear Algebra Appl.*, vol. 7, pp. 687-714, 2000.

[2] Y. Saad, *Numerical Methods for Large Eigenvalue Problems, 2nd ed.*: SIAM, 2011.

[3] M. Galassi and et al., "GNU Scientific Library," The GSL Team, 2011.

[4] Y. Saad, *Iterative Methods for Sparse Linear Systems.*: PWS Pub., 1996.

[5] C. Farhat and et al., "FETI-DP: a dual- primal unified FETI method - part I: A faster alternative to the two-level FETI method," *Int. J. Numer. Meth. Engng*, vol. 50, pp. 1523-1544, 2001.

[6] S. Nakamura, *Computational Methods in Engineering and Science*.: John Wiely & Sons, 1977.

[7] T.A. Davis, *Direct Methods for Sparse Linear Systems.*: SIAM, 2006.

[8] B. Hendrickson. (2014, Mar.) Chaco: Software for Partitioning Graphs. [Online]. http://www.sandia.gove/~bahendr/chaco.html

[9] G. Karypis. (2014, Mar.) METIS - Serial Graph and Hypergraph Partitioning Software. [Online]. http://glaros.dtc.umn.edu/gkhome/views/metis/index.html