An Experiment of Robust Algorithm for the Eigenvalue problem of a Multigroup Neutron Diffusion based on modified FETI-DP – Part 2

KNS automn meeting

2014.10.30~31

Yongpyong

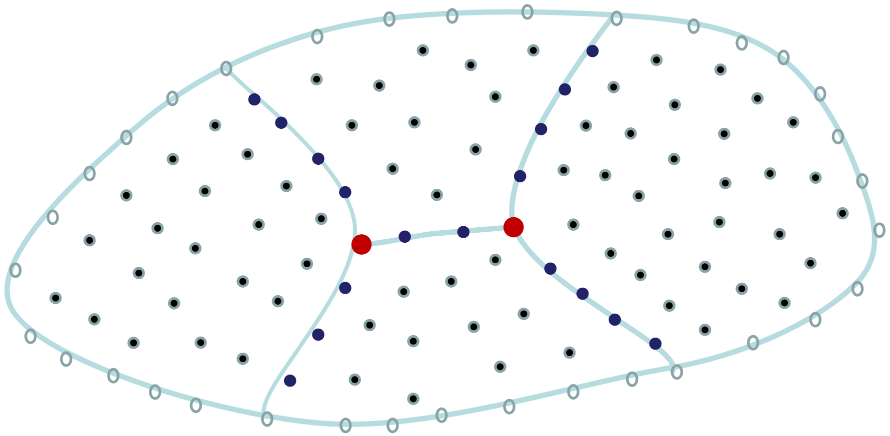
Jonghwa Chang

Korea Atomic Energy Research Institute

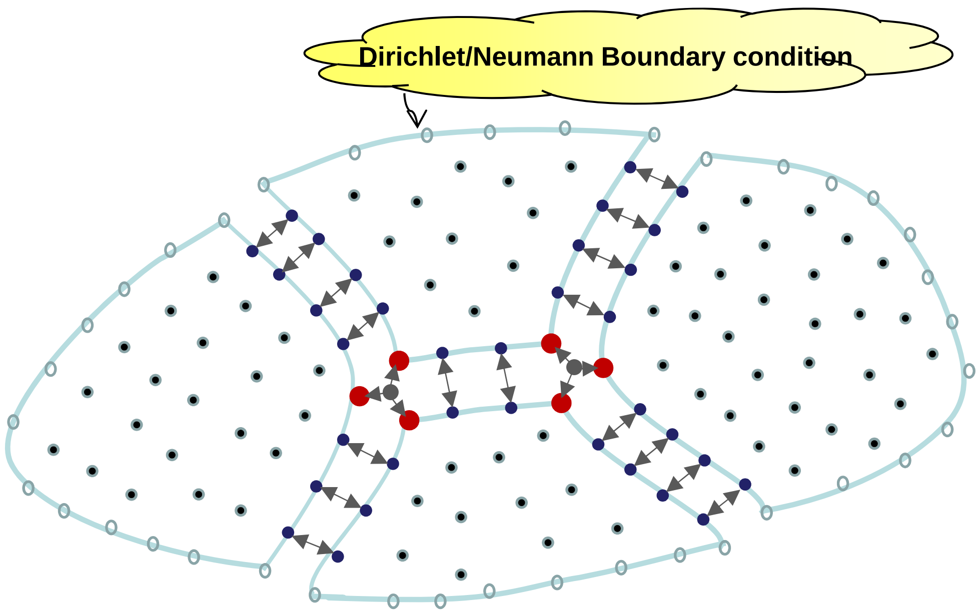
Back ground

- Detailed core modeling requires huge number of unknowns. There is memory size problem as well as long computing time.
- Utilization of cheap parallel processing capability to speed up CPU intensive calculation.
- Modified FETI-DP was successfully applied to Eigenvalue problem of the Neutron Multigroup Diffusion equation.
- Update from last presentation (2014 KNS spring mtg)
  - 3D partitioning implemented for arbitrary number of CPUs.
  - Solution of multigroup source problem for each sub-domain are now using a Krylov method (PGMRES) for faster convergence and little dependency on SOR weight parameter.
  - Experiment on varying no. of CPUs and turn-around time.

Domain Decomposition

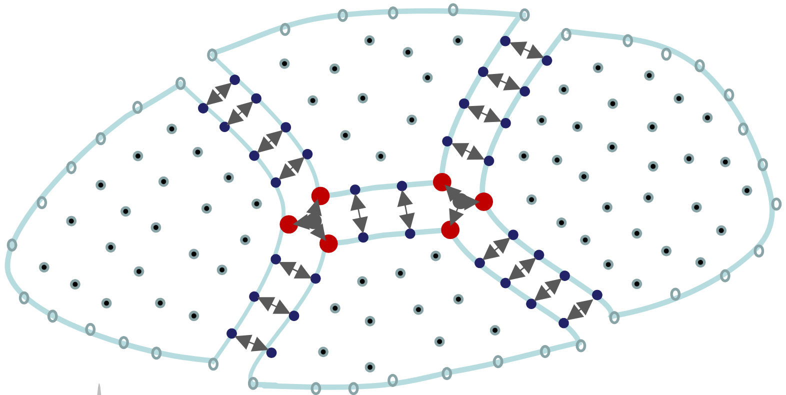


Conventional approach
- guess flux (or current) at Interface
- solve subdomain equation
- find current (or flux)
- adjust flux (or current)
- repeat until convergence



FETI approach
- setup global equation
- solve subdomain equation
- find l.h.s. of global equation
- repeat until convergence

Ref) C. Farhat et. al, Numer.Linear Algebra Appl. 2000; 7:687-714



Linear system

$$\mathbf{Ku} = \mathbf{f}$$

Subdomain problem

$$\begin{bmatrix} K_{ii}^s & K_{ib}^s \\ K_{bi} & K_{bb}^s \end{bmatrix} \begin{bmatrix} u_i^s \\ u_b^s \end{bmatrix} = \begin{bmatrix} f_i^s \\ f_b^s \end{bmatrix}$$

Continuity condition at interface

$$u_b^S - u_b^q = 0 \quad \text{on} \quad \partial\Omega^S \cap \partial\Omega^q$$

$$\sum_{s=1} B^s u^s = 0 \qquad B^s u^s = \pm u_b^s$$

Nodes in a subdomain Ω^S

- Internal nodes (*i*)
- — Interface boundary nodes (*b*)
  - Cornor nodes (b_c)
  - Remainder nodes (b_r)

Partitioning subdomain matrix

$$\begin{bmatrix} K_{rr}^s & K_{rc}^s \\ K_{cr}^s & K_{cc}^s \end{bmatrix} \begin{bmatrix} u_r^s \\ u_{b_c}^s \end{bmatrix} = \begin{bmatrix} f_r^s \\ f_{b_c}^s \end{bmatrix}$$

original FETI-DP requires symmeteric K

$$u_r^s = \begin{bmatrix} u_i^s \\ u_{b_r}^s \end{bmatrix} \qquad f_r^s = \begin{bmatrix} f_i^s \\ f_{b_r}^s \end{bmatrix}$$

$$u_c = \begin{bmatrix} u_c^1 \\ \vdots \\ u_c^j \\ \vdots \\ u_c^{N_c} \end{bmatrix}$$

global cornor vector

Interface condition

$$B_r^s u_r^s = \pm u_{b_r}^s \qquad B_c^s u_c = u_{b_c}^s$$

using Lagrange multiplier

local

$$K_{rr}^s u_r^s + K_{rc}^s B_c^s u_c + B_r^{sT} \lambda = f_r^s$$

$$\sum_s B_c^{sT} K_{cr}^s u_r^s + \sum_s B_c^{sT} K_{cc}^s B_c^s u_c = \sum_s B_c^{sT} f_{b_c}^s = f_c$$

$$\sum_s B_r^s u_r^s = 0$$

global

global matrix

$$K_{cc} \equiv \sum_s B_c^{sT} K_{cc}^s B_c^s$$

Recipe

- FEM matrix with triangular pipe element. Linear or Quadratic Lagrangian base in both direction.
- Subdomain partition using MeTIS with non-zero points weighted FEM element.
- Eigenvalue problem $K\phi=\mu B\phi$ using Householder-Arnoldi.
- Linear system, $A\lambda=b$ using PBiCGSTAB with Lumped preconditioner.
- Subdomain multigroup equation $K_i\phi=s$ using PGMRES with SSOR preconditioner.

PGMRES algorithm

Ref) Y. Saad, Iterative Methods for Sparse Linear Systems.

Left preconditioned GMRES

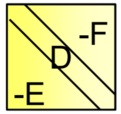
$$M^{-1}Ax = M^{-1}b$$

1. Compute $\mathbf{r}_0 = M^{-1}(\mathbf{b} - A\mathbf{x}_0)$ $\beta = \|\mathbf{r}_0\|_2$ $\mathbf{v}_1 = \mathbf{r}_0 / \beta$
2. For j=1,…,m Do:
3. Compute $\mathbf{w} = M^{-1}A\mathbf{v}_j$
4. For i=1,…,j Do:
5. $h_{i,j} = (\mathbf{w}, \mathbf{v}_i)$
6. $\mathbf{w} = \mathbf{w} - h_{i,j}\mathbf{v}_i$
7. EndDo
8. Compute $h_{j+1,j} = \|\mathbf{w}\|_2$ $\mathbf{v}_{j+1} = \mathbf{w} / h_{j+1,j}$
9. EndDo
10. Define $V_m = [\mathbf{v}_1, \cdots, \mathbf{v}_m]$ $\bar{H}_m = \{h_{i,j}\}_{1 \le i \le j+1; 1 \le j \le m}$
11. Compute $y_m = \arg\min_y \|\beta\hat{e}_1 - \bar{H}_m y\|_2$ $\mathbf{x}_m = \mathbf{x}_0 + V_m y_m$
12. If converged Stop, else set $\mathbf{x}_0 = \mathbf{x}_m$ and GoTo 1

when $h_{j+1,j}=0$ break loop

The multigroup equation

$$\begin{pmatrix} A_1 & -S_{2\to1} & -S_{3\to1} & -S_{4\to1} \\ -S_{1\to2} & A_2 & -S_{3\to2} & -S_{4\to2} \\ -S_{1\to3} & -S_{2\to3} & A_3 & -S_{4\to3} \\ -S_{1\to4} & -S_{2\to4} & -S_{3\to4} & A_4 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{pmatrix} = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{pmatrix}$$



We can solve above eqn. using block symm. SOR.

$$M_{SSOR} = \frac{1}{\omega(2-\omega)}(D-\omega E)^{-1} D^{-1} (D-\omega F)$$

~200 iter. for IAEA3D prob.

How to compute : $M^{-1}x = y$

1+m times per GMRES loop

$$(D-\omega E)D^{-1}(D-\omega F)y = x$$

forward sweep $(I - \omega ED^{-1})z = x$

backward sweep $(D-\omega F)y = z$

D^{-1} by Cholesky decomp.

Hessenberg matrix H : (m+1) x m

$$H = \begin{pmatrix} h_{00} & h_{01} & h_{02} & h_{03} \\ h_{10} & h_{11} & h_{12} & h_{13} \\ & h_{21} & h_{22} & h_{23} \\ & & h_{32} & h_{33} \\ & & & h_{43} \end{pmatrix}$$

$$\arg\min_y \|\beta\hat{e}_1 - \bar{H}_m y\|_2$$

find a vector *y* which gives minimum norm of $\beta\hat{e}_1 - \bar{H}_m y$

over determined

Hessenberg rotation

$$y_m = \arg\min_y \left\| b - Hy \right\|_2$$

Can be solved by Least Squares method
or, Hessenberg rotation method

$$H = \begin{pmatrix} h_{00} & h_{01} & h_{02} & h_{03} \\ h_{10} & h_{11} & h_{12} & h_{13} \\ & h_{21} & h_{22} & h_{23} \\ & & h_{32} & h_{33} \\ & & & h_{43} \end{pmatrix} \qquad b = \begin{pmatrix} \beta \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\Omega_0 = \begin{pmatrix} c_0 & s_0 & & & \\ -s_0 & c_0 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{pmatrix}$$

$$\Omega_0 H = \begin{pmatrix} c_0 h_{00} + s_0 h_{10} & c_0 h_{01} + s_0 h_{11} & c_0 h_{02} + s_0 h_{12} & c_0 h_{02} + s_0 h_{12} \\ -s_0 h_{00} + c_0 h_{10} & -s_0 h_{01} + c_0 h_{11} & -s_0 h_{02} + c_0 h_{12} & -s_0 h_{02} + c_0 h_{12} \\ & h_{21} & h_{22} & h_{23} \\ & & h_{32} & h_{33} \\ & & & h_{43} \end{pmatrix}$$

$$\Omega_0 b = \begin{pmatrix} c_0 \beta \\ -s_0 \beta \\ \\ \\ \\ \end{pmatrix} \qquad s_0 = \frac{h_{10}}{\sqrt{h_{00}^2 + h_{10}^2}} \qquad c_0 = \frac{h_{00}}{\sqrt{h_{00}^2 + h_{10}^2}}$$

Successive rotation to upper matrix

$$\tilde{H} \equiv \Omega_3 \Omega_2 \Omega_1 \Omega_0 H \qquad \tilde{b} \equiv \Omega_3 \Omega_2 \Omega_1 \Omega_0 b$$

$$\tilde{H} y = \tilde{b}$$

$$\begin{pmatrix} \tilde{h}_{00} & \tilde{h}_{01} & \tilde{h}_{02} & \tilde{h}_{03} \\ & \tilde{h}_{11} & \tilde{h}_{12} & \tilde{h}_{13} \\ & & \tilde{h}_{22} & \tilde{h}_{23} \\ & & & \tilde{h}_{33} \\ & & & \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} \tilde{b}_0 \\ \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \end{pmatrix}$$

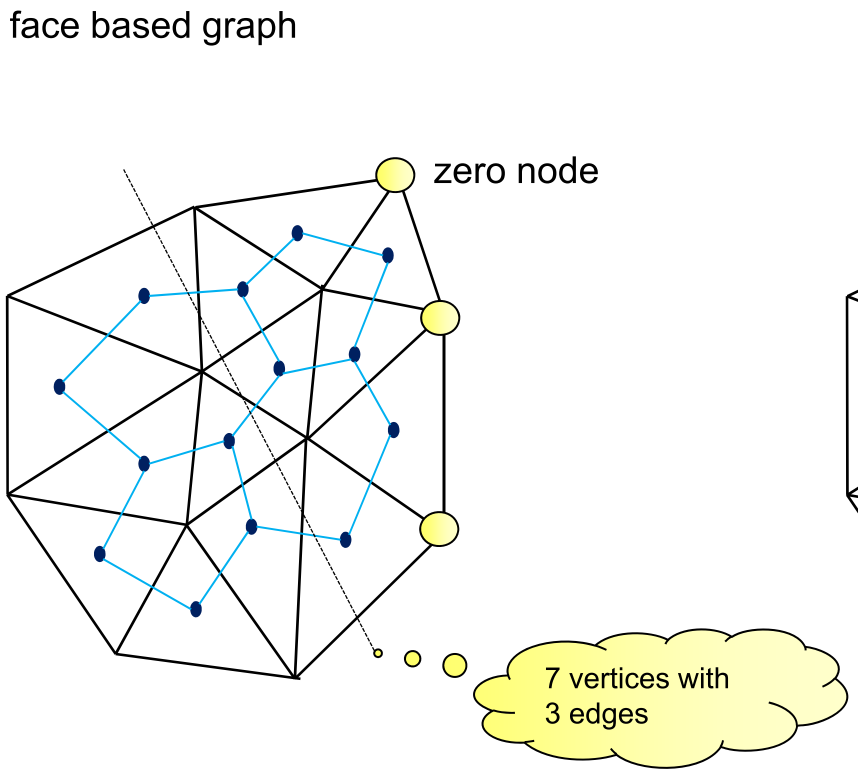
argmin$_y$: $y_m = (y_0, y_1, y_2, y_3)^T$

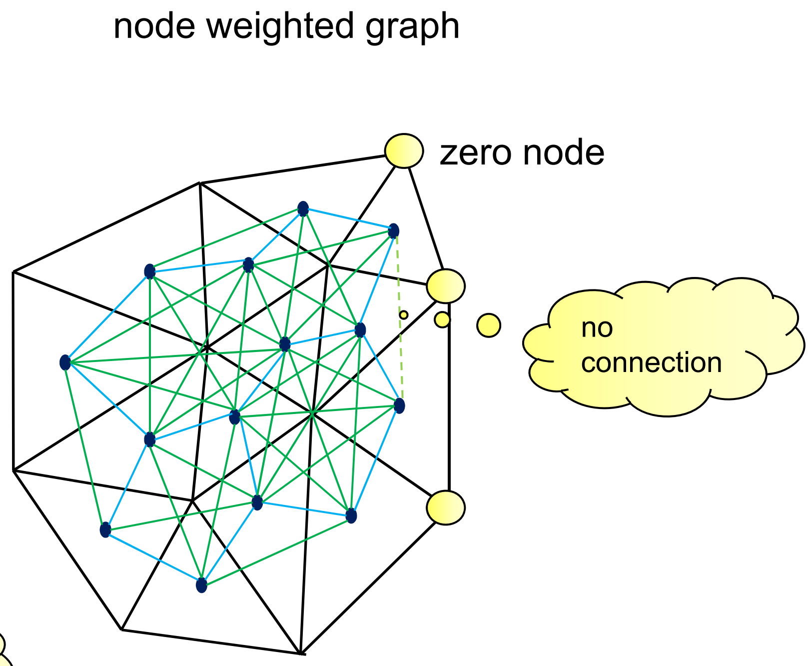
residual : $\tilde{b}_4$

Result (for IAEA3D problem)
 outer iterations : ~2 (m=5)
 no. of $M^{-1}x$: ~ 6

✓ cf. conventional SOR ~150 iterations

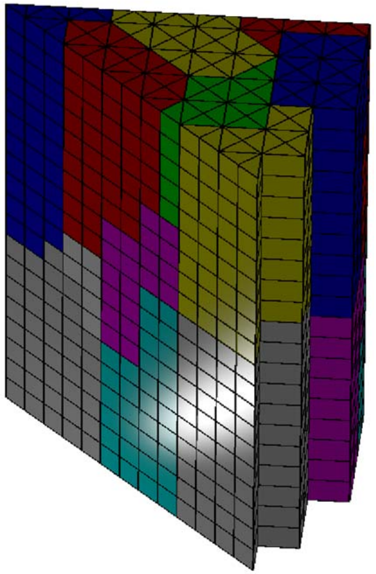
Vertice graph



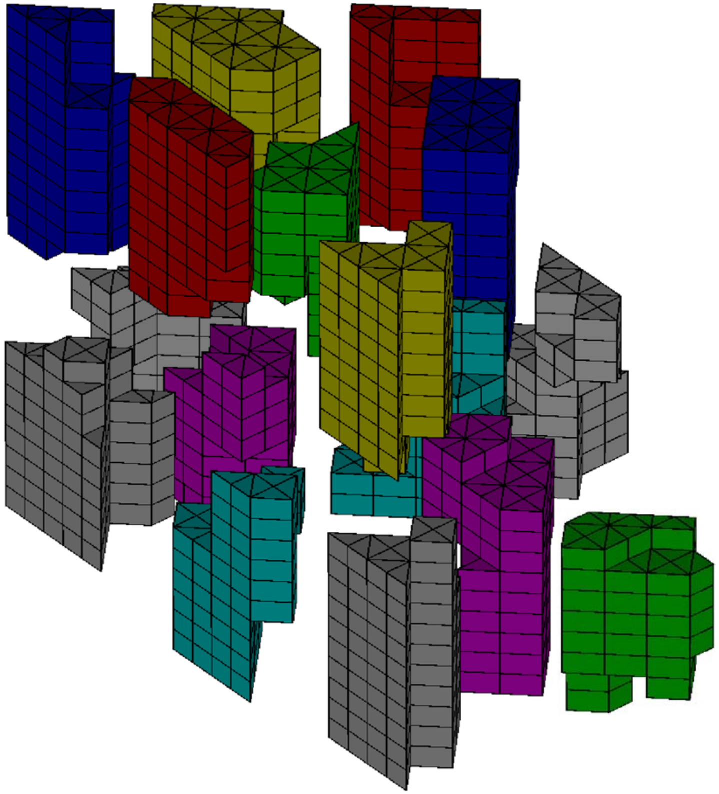


Find minimun edge cuts with same vertice weight using METIS.

IAEA problem - Nonzero point weight of vertices



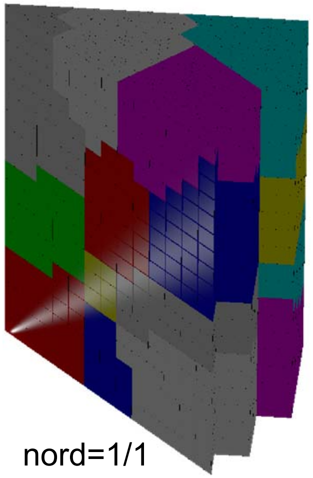
Graph node weight using non-zero points improve evenness a little.
- from 1.28 to 1.16 for low order
- not much for higher order



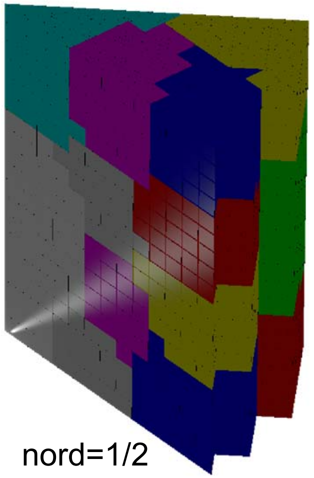
more elements for zero boundary.

Node base graph partitioning

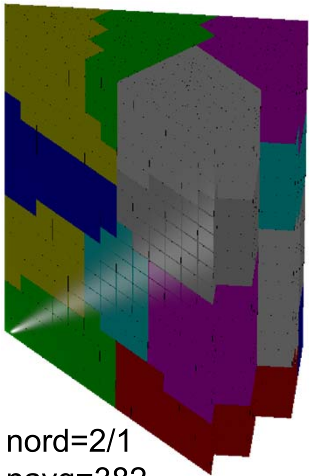
ns=1



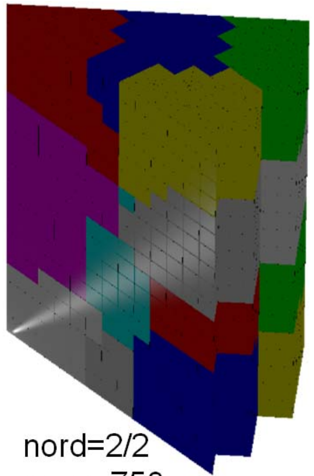
nord=1/1
navg=102
nmax=128
peak=1.26



nord=1/2
navg=208
nmax=262
peak=1.26

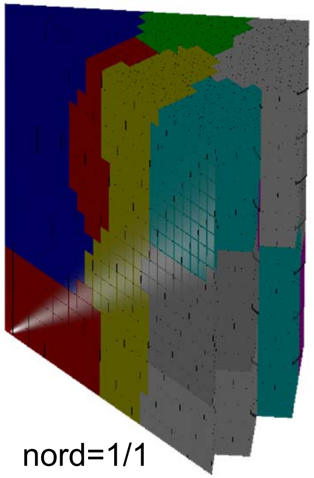


nord=2/1
navg=382
nmax=422
peak=1.11

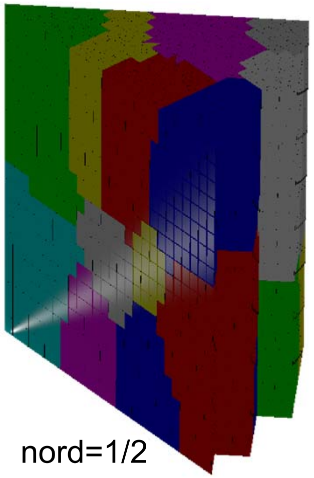


nord=2/2
navg=758
nmax=791
peak=1.04

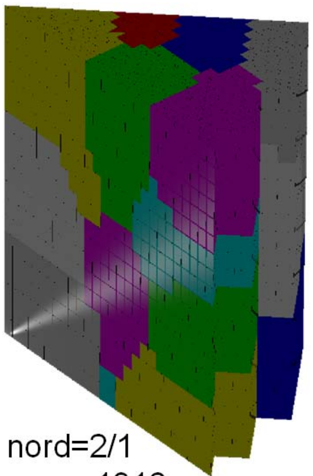
ns=2



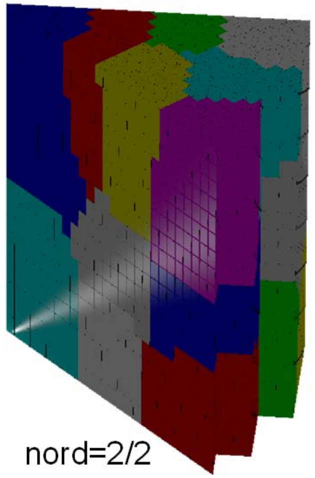
nord=1/1
navg=346
nmax=372
peak=1.08



nord=1/2
navg=708
nmax=788
peak=1.11



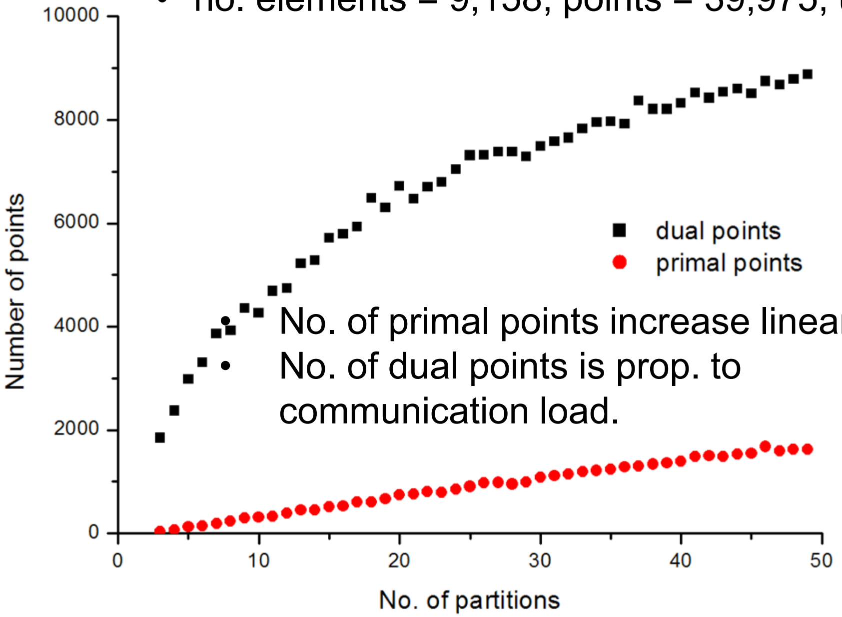
nord=2/1
navg=1316
nmax=1380
peak=1.05



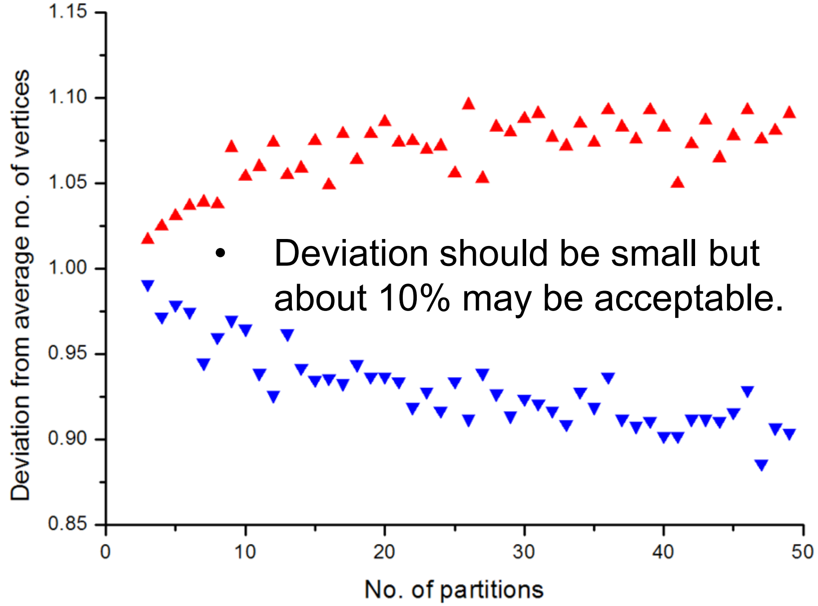
nord=2/2
navg=2618
nmax=2713
peak=1.04

Result (ns=2, nord=2/2)

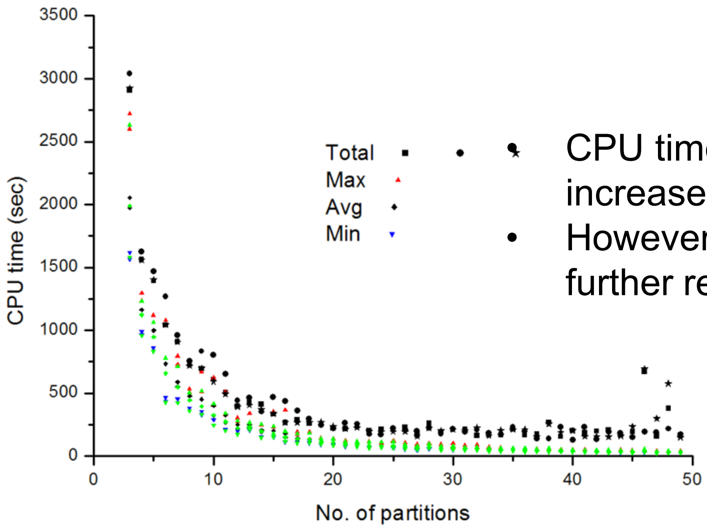
- no. elements = 9,158, points = 39,975, unknowns = 73,260



- No. of primal points increase linearly.
- No. of dual points is prop. to communication load.

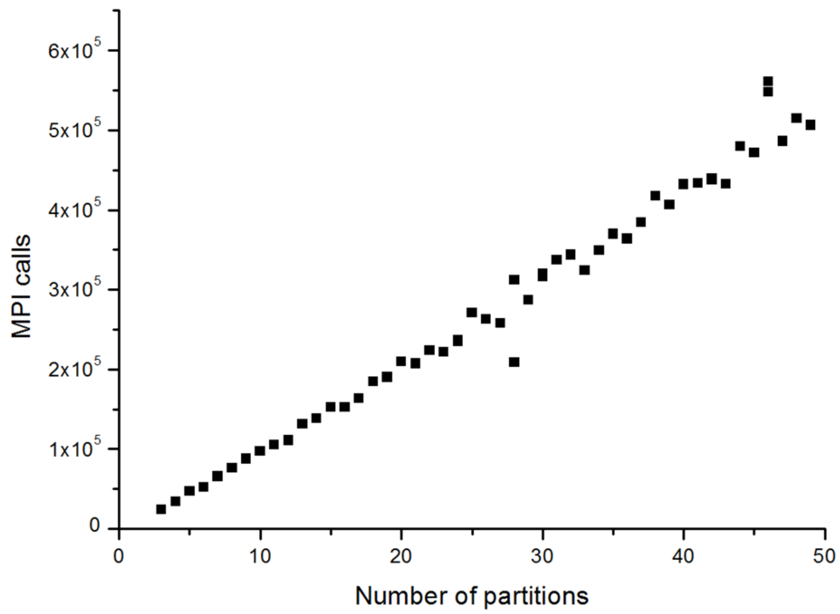


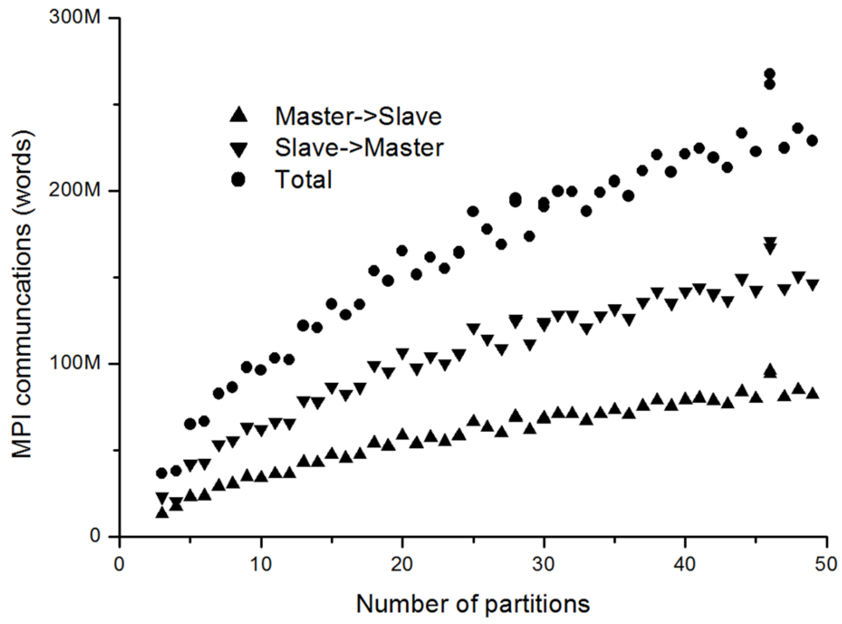
- Deviation should be small but about 10% may be acceptable.



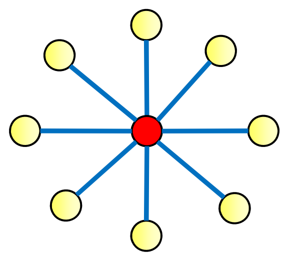
- CPU time is decreasing as no. of proc. increase
- However, communication bottleneck limits further reduction in CPU time.

Communication load

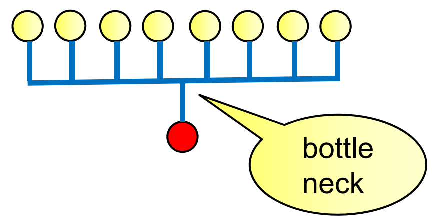




- Communication load increases almost linearly as number of partitions is increasing.
- MPI communication load can be significant for slow network as the number of partition increase.
  - There may be a maximum number of partitions where speed up is limited by the network configuration.



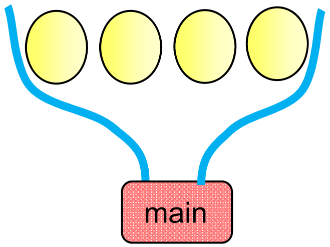
optimum network configuration



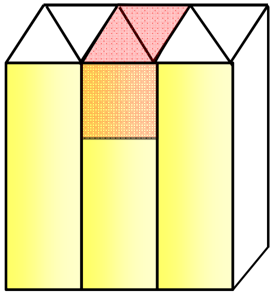
switched network configuration

Conclusion and Further works

- FETI-DP extension to block symmetric and global nonsymmetric, neutron multigroup eigenvalue problem works well.
- Graph theory is used for arbitrary number of CPU partitioning with element vertice graph.
- Effect of network speed is not so significant (Gigabit) for few tens partitions. However, there is a bottleneck for large central communication.
  - ➢ Need confirmation at larger and faster network cluster.



- Non-conforming FEM for partitioning.
  - Important for
    - Time dependent problem
    - Soluble boron free reactors such as GCR, SFR
  - Interior Penalty Galerkin method can be used for non-conforming FEM.



Thank you for attention. Any question or comment ?