Introduction
Overview of the Class

Kengo Nakajima
Information Technology Center

Technical & Scientific Computing II (4820-1028)
Seminar on Computer Science II (4810-1205)
Hybrid Distributed Parallel Computing (3747-111)

Descriptions of Class

- Technical & Scientific Computing II (4820-1028)
  - 科学技術計算Ⅱ
  - Department of Mathematical Informatics
- Seminar on Computer Science II (4810-1205)
  - コンピュータ科学特別講義Ⅱ
  - Department of Computer Science
- Hybrid Distributed Parallel Computing (3747-111)
  - ハイブリッド分散並列コンピューティング
  - Department of Electrical Engineering & Information Systems

- 2009-2014
  - Introduction to FEM Programming
    - FEM: Finite-Element Method：有限要素法
  - Summer (I) : FEM Programming for Solid Mechanics
  - Winter (II): Parallel FEM using MPI
    - The 1st part (summer) is essential for the 2nd part (winter)
- Problems
  - Many new (international) students in Winter, who did not take the 1st part in Summer
  - **They are generally more diligent than Japanese students**
- 2015
  - Summer (I) : Multicore programming using OpenMP
  - Winter (II): FEM + Parallel FEM using MPI for Heat Conduction
  - Part I & II are mostly independent
- 2016
  - Oakleaf-FX (Fujitsu FX10) -> Reedbush-U (Intel Broadwell-EP)

Technical & Scientific Computing-II Seminar on Computer Science-II Hybrid Distributed Parallel Computing

科学技術計算Ⅱ・コンピュータ科学特別講義Ⅱ・ハイブリッド分散並列プログラミング

- Instructor
  - Kengo Nakajima
  - Professor, Information Technology Center, The University of Tokyo
- Topics
  - Finite-Element Method (FEM)
    - Heat Conduction: easier than solid mechanics
  - Parallel FEM using MPI and OpenMP

This class provides introduction to large-scale scientific computing using the most advanced massively parallel supercomputers. Topics cover:

•Finite-Element Method (FEM)
•Message Passing Interface (MPI)
•Parallel FEM using MPI and OpenMP
•Parallel Numerical Algorithms for Iterative Linear Solvers

Several sample programs will be provided and participants can review the contents of lectures through hands-on-exercise/practices using Reedbush-U System with Intel Broadwell-EP at the University of Tokyo.

Finite-Element Method is widely-used for solving various types of real-world scientific and engineering problems, such as structural analysis, fluid dynamics, electromagnetics, and etc. This lecture course provides brief introduction to procedures of FEM for 1D/3D steady-state heat conduction problems with iterative linear solvers and to parallel FEM. **Lectures for parallel FEM will be focused on design of data structure for distributed local mesh files, which is the key issue for efficient parallel FEM.** Introduction to MPI (Message Passing Interface), which is widely used method as "de facto standard" of parallel programming, is also provided.

Solving large-scale linear equations with sparse coefficient matrices is the most expensive and important part of FEM and other methods for scientific computing, such as Finite-Difference Method (FDM) and Finite-Volume Method (FVM). Recently, families of Krylov iterative solvers are widely used for this process. In this class, details of implementations of parallel Krylov iterative methods are provided along with parallel FEM.

Moreover, lectures on programming for multicore architectures will be also given along with brief introduction to OpenMP and OpenMP/MPI Hybrid Parallel Programming Model.

Motivation for Parallel Computing (and this class)

- Large-scale parallel computer enables fast computing in large-scale scientific simulations with detailed models. Computational science develops new frontiers of science and engineering.
- Why parallel computing ?
  - faster & larger
  - "larger" is more important from the view point of "new frontiers of science & engineering", but "faster" is also important.
  - + more complicated
  - Ideal: Scalable
    - Weak Scaling, Strong Scaling

Scalable, Scaling, Scalability

- Solving N^x scale problem using N^x computational resources during same computation time
  - for large-scale problems: **Weak Scaling, Weak Scalability**
  - e.g. CG solver: more iterations needed for larger problems
- Solving a problem using N^x computational resources during 1/N computation time
  - for faster computation: **Strong Scaling, Strong Scalability**

Kengo Nakajima (1/2)

- Current Position
  - Professor, Supercomputing Research Division, Information Technology Center 情報基盤センター
  - Professor, Department of Mathematical Informatics, Graduate School of Information Science & Engineering 数理情報学専攻
  - Professor, Department of Electrical Engineering & Information Systems, Graduate School of Engineering 電気系工学専攻
  - Visiting Senior Researcher, Advanced Institute for Computational Science (AICS), RIKEN
- Research Interest
  - High-Performance Computing
  - Parallel Numerical Linear Algebra (Preconditioning)
  - Parallel Programming Model
  - Computational Mechanics, Computational Fluid Dynamics
  - Adaptive Mesh Refinement, Parallel Visualization

Kengo Nakajima (2/2)

- Education
  - B.Eng (Aeronautics, The University of Tokyo, 1985)
  - M.S. (Aerospace Engineering, University of Texas, 1993)
  - Ph.D. (Quantum Engineering & System Sciences, The University of Tokyo, 2003)
- Professional Background
  - Mitsubishi Research Institute, Inc. (1985-1999)
  - Research Organization for Information Science & Technology (1999-2004)
  - The University of Tokyo
    - Department Earth & Planetary Science (2004-2008)
    - Information Technology Center (2008-)
  - JAMSTEC (2008-2011), part-time
  - RIKEN (2009-), part-time

Scientific Computing = SMASH

Science

Modeling

Algorithm

Software

Hardware

- You have to learn many things.
- Collaboration (or Co-Design) will be important for future career of each of you, as a scientist and/or an engineer.
  - You have to communicate with people with different backgrounds.
  - It is more difficult than communicating with foreign scientists from same area.
- (Q): Your Department ?

This Class ...

<u>S</u>cience

<u>M</u>odeling

<u>A</u>lgorithm

<u>S</u>oftware

<u>H</u>ardware

- Parallel FEM using MPI and OpenMP

- Science: Heat Conduction
- Modeling: FEM
- Algorithm: Iterative Solvers etc.

- You have to know many components to learn FEM, although you have already learned each of these in undergraduate and high-school classes.

Road to Programming for "Parallel" Scientific Computing

Programming for Parallel Scientific Computing (e.g. Parallel FEM/FDM)

Programming for Real World Scientific Computing (e.g. FEM, FDM)

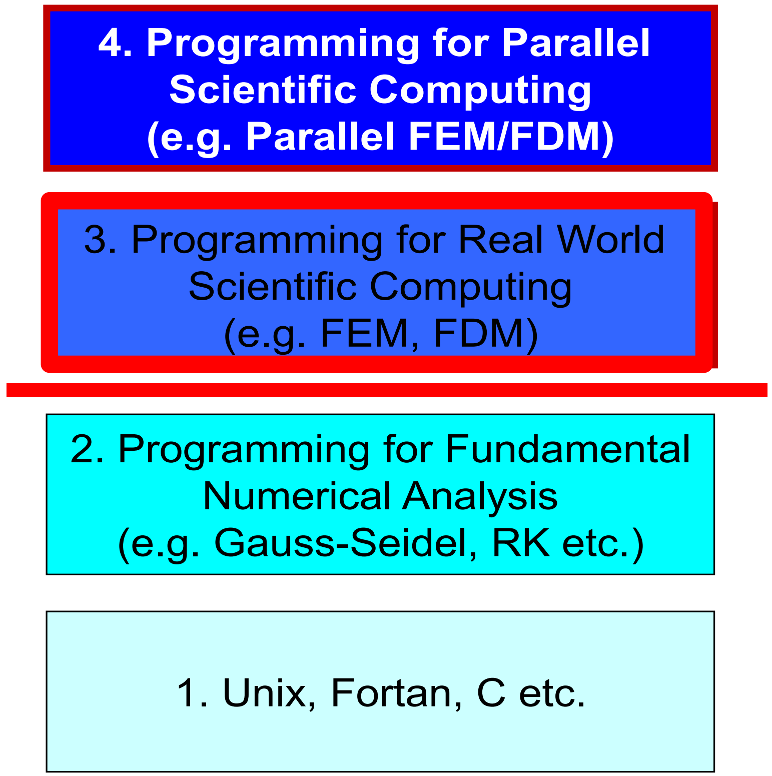
<u>Big gap here !!</u>

Programming for Fundamental Numerical Analysis (e.g. Gauss-Seidel, RK etc.)

Unix, Fortan, C etc.

The third step is important !

- How to parallelize applications ?
  - How to extract parallelism ?
  - If you understand methods, algorithms, and implementations of the original code, it's easy.
  - "Data-structure" is important



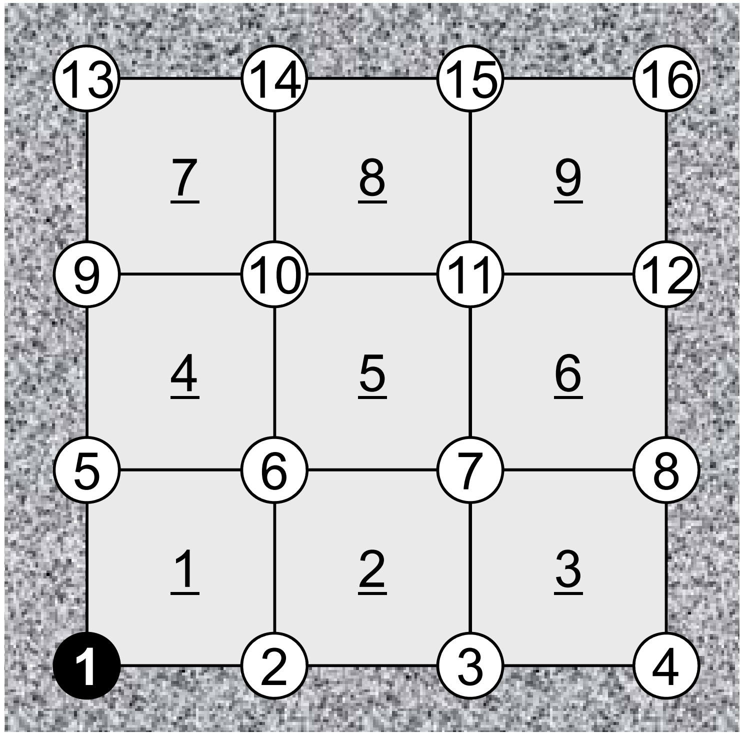
- How to understand the code ?
  - Reading the application code !!
  - It seems primitive, but very effective.
  - In this class, "reading the source code" is encouraged.

Finite-Element Method (FEM)

- One of the most popular numerical methods for solving PDE’s.
  - elements (meshes) & nodes (vertices)
- Consider the following 2D heat transfer problem:

$$\lambda\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}\right)+Q=0$$

  - 16 nodes, 9 bi-linear elements
  - uniform thermal conductivity (λ=1)
  - uniform volume heat flux (Q=1)
  - T=0 at node 1
  - Insulated boundaries



Galerkin FEM procedures

- Apply Galerkin procedures to each element:

$$\int_V [N]^T \left\{ \lambda \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) + Q \right\} dV = 0$$

where $T = [N]\{\phi\}$ in each elem.

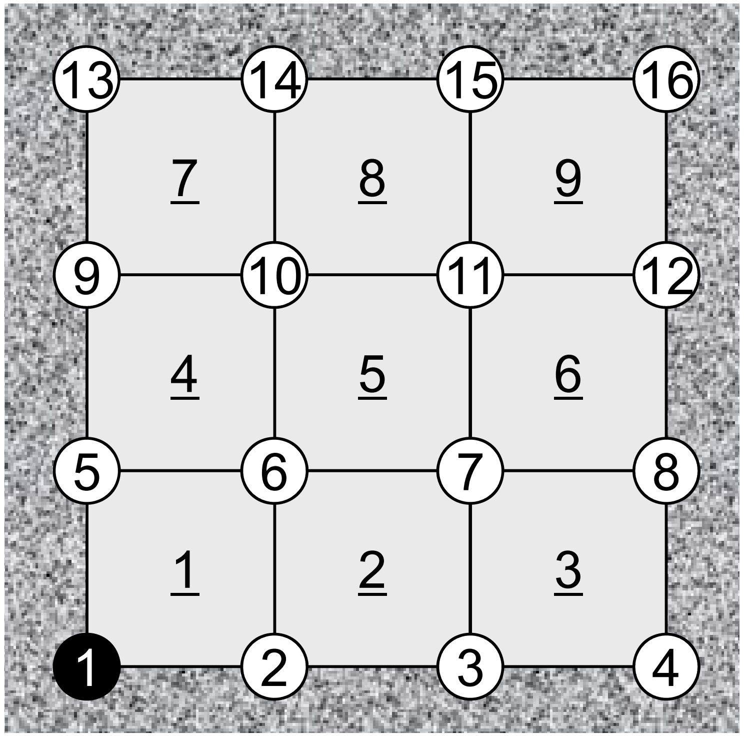
$\{\phi\}$: T at each vertex

$[N]$: Shape function (Interpolation function)

- Introduce the following "weak form" of original PDE using Green's theorem:

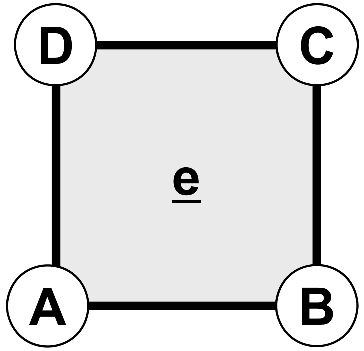
$$-\int_V \lambda \left(\frac{\partial [N]^T}{\partial x} \frac{\partial [N]}{\partial x} + \frac{\partial [N]^T}{\partial y} \frac{\partial [N]}{\partial y} \right) dV \cdot \{\phi\}$$

$$+ \int_V Q[N]^T dV = 0$$



Element Matrix

- Apply the integration to each element and form "element" matrix.



$$-\int_V \lambda \left(\frac{\partial [N]^T}{\partial x} \frac{\partial [N]}{\partial x} + \frac{\partial [N]^T}{\partial y} \frac{\partial [N]}{\partial y} \right) dV \cdot \{\phi\}$$

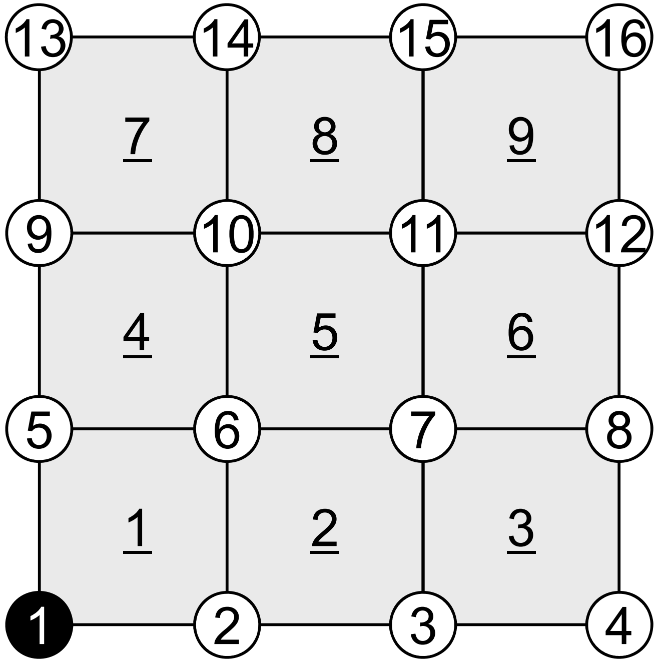
$$+\int_V Q[N]^T dV = 0$$

$$[k^{(e)}]\{\phi^{(e)}\} = \{f^{(e)}\}$$

$$\begin{bmatrix} k_{AA}^{(e)} & k_{AB}^{(e)} & k_{AC}^{(e)} & k_{AD}^{(e)} \\ k_{BA}^{(e)} & k_{BB}^{(e)} & k_{BC}^{(e)} & k_{BD}^{(e)} \\ k_{CA}^{(e)} & k_{CB}^{(e)} & k_{CC}^{(e)} & k_{CD}^{(e)} \\ k_{DA}^{(e)} & k_{DB}^{(e)} & k_{DC}^{(e)} & k_{DD}^{(e)} \end{bmatrix} \begin{Bmatrix} \phi_A^{(e)} \\ \phi_B^{(e)} \\ \phi_C^{(e)} \\ \phi_D^{(e)} \end{Bmatrix} = \begin{Bmatrix} f_A^{(e)} \\ f_B^{(e)} \\ f_C^{(e)} \\ f_D^{(e)} \end{Bmatrix}$$

Global (Overall) Matrix

Accumulate each element matrix to "global" matrix.

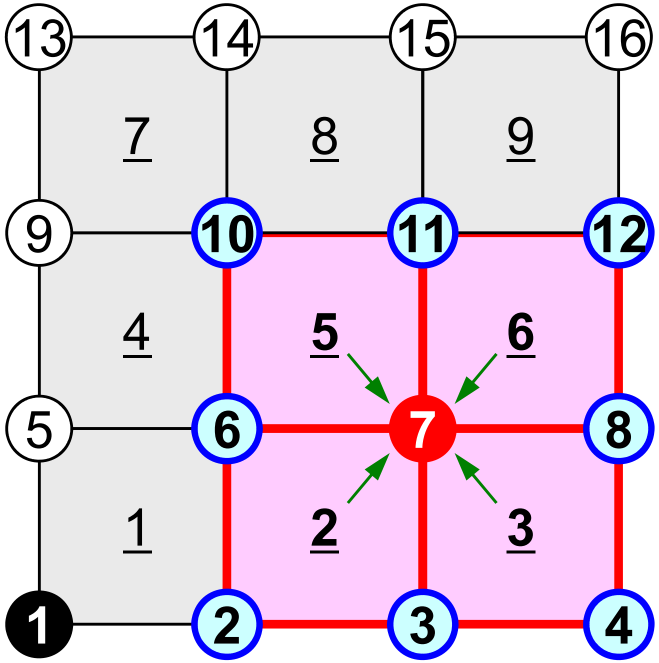


$$[K]\{\Phi\}=\{F\}$$

$$\begin{bmatrix}
D & X & & & X & X & & & & & & & & & & \\
X & D & X & & X & X & X & & & & & & & & & \\
& X & D & X & & X & X & X & & & & & & & & \\
& & X & D & & & X & X & & & & & & & & \\
X & X & & & D & X & & & X & X & & & & & & \\
X & X & X & & X & D & X & & X & X & X & & & & & \\
& X & X & X & & X & D & X & & X & X & X & & & & \\
& & X & X & & & X & D & & & X & X & & & & \\
& & & & X & X & & & D & X & & & X & X & & \\
& & & & X & X & X & & X & D & X & & X & X & X & \\
& & & & & X & X & X & & X & D & X & & X & X & X \\
& & & & & & X & X & & & X & D & & & X & X \\
& & & & & & & & X & X & & & D & X & & \\
& & & & & & & & X & X & X & & X & D & X & \\
& & & & & & & & & X & X & X & & X & D & X \\
& & & & & & & & & & X & X & & & X & D
\end{bmatrix}
\begin{Bmatrix}\Phi_1\\\Phi_2\\\Phi_3\\\Phi_4\\\Phi_5\\\Phi_6\\\Phi_7\\\Phi_8\\\Phi_9\\\Phi_{10}\\\Phi_{11}\\\Phi_{12}\\\Phi_{13}\\\Phi_{14}\\\Phi_{15}\\\Phi_{16}\end{Bmatrix}
=
\begin{Bmatrix}F_1\\F_2\\F_3\\F_4\\F_5\\F_6\\F_7\\F_8\\F_9\\F_{10}\\F_{11}\\F_{12}\\F_{13}\\F_{14}\\F_{15}\\F_{16}\end{Bmatrix}$$

To each node …

Effect of surrounding elem's/nodes are accumulated.



$$[K]\{\Phi\}=\{F\}$$

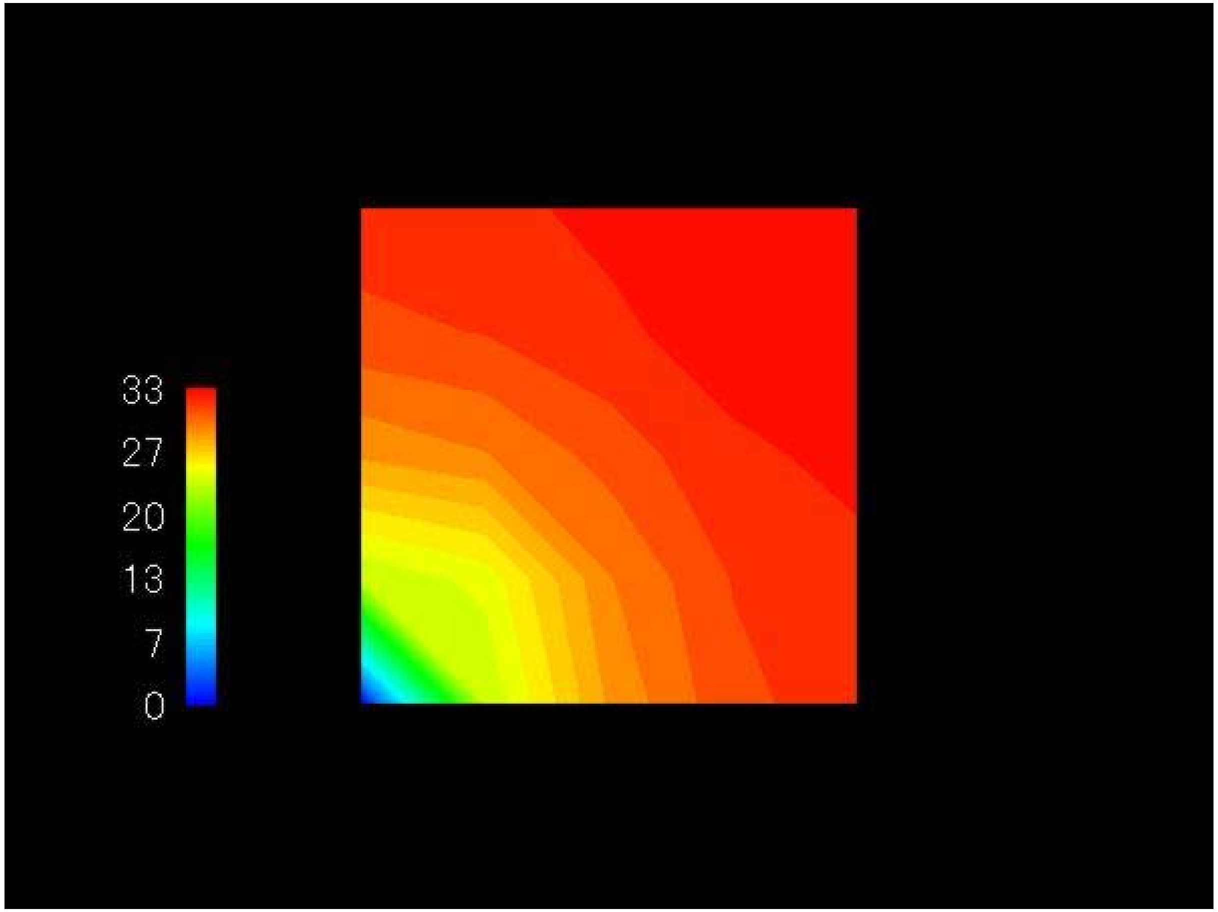
$$
\begin{bmatrix}
D & X & & & X & X & & & & & & & & & & \\
X & D & X & & X & X & X & & & & & & & & & \\
& X & D & X & & X & X & X & & & & & & & & \\
& & X & D & & & X & X & & & & & & & & \\
X & X & & & D & X & & & X & X & & & & & & \\
X & X & X & & X & D & X & & X & X & X & & & & & \\
& X & X & X & & X & D & X & & X & X & X & & & & \\
& & X & X & & & X & D & & & X & X & & & & \\
& & & & X & X & & & D & X & & & X & X & & \\
& & & & X & X & X & & X & D & X & & X & X & X & \\
& & & & & X & X & X & & X & D & X & & X & X & X \\
& & & & & & X & X & & & X & D & & & X & X \\
& & & & & & & & X & X & & & D & X & & \\
& & & & & & & & X & X & X & & X & D & X & \\
& & & & & & & & & X & X & X & & X & D & X \\
& & & & & & & & & & X & X & & & X & D
\end{bmatrix}
\begin{Bmatrix}
\Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \Phi_4 \\ \Phi_5 \\ \Phi_6 \\ \Phi_7 \\ \Phi_8 \\ \Phi_9 \\ \Phi_{10} \\ \Phi_{11} \\ \Phi_{12} \\ \Phi_{13} \\ \Phi_{14} \\ \Phi_{15} \\ \Phi_{16}
\end{Bmatrix}
=
\begin{Bmatrix}
F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \\ F_7 \\ F_8 \\ F_9 \\ F_{10} \\ F_{11} \\ F_{12} \\ F_{13} \\ F_{14} \\ F_{15} \\ F_{16}
\end{Bmatrix}
$$

Solve the obtained global eqn's

under certain boundary conditions
(Φ_1=0 in this case)

$$\begin{bmatrix}
D & X & & & X & X & & & & & & & & & & \\
X & D & X & & X & X & X & & & & & & & & & \\
& X & D & X & & X & X & X & & & & & & & & \\
& & X & D & & & X & X & & & & & & & & \\
X & X & & & D & X & & & X & X & & & & & & \\
X & X & X & & X & D & X & & X & X & X & & & & & \\
& X & X & X & & X & D & X & & X & X & X & & & & \\
& & X & X & & & X & D & & & X & X & & & & \\
& & & & X & X & & & D & X & & & X & X & & \\
& & & & X & X & X & & X & D & X & & X & X & X & \\
& & & & & X & X & X & & X & D & X & & X & X & X \\
& & & & & & X & X & & & X & D & & & X & X \\
& & & & & & & & X & X & & & D & X & & \\
& & & & & & & & X & X & X & & X & D & X & \\
& & & & & & & & & X & X & X & & X & D & X \\
& & & & & & & & & & X & X & & & X & D
\end{bmatrix}
\begin{Bmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \Phi_4 \\ \Phi_5 \\ \Phi_6 \\ \Phi_7 \\ \Phi_8 \\ \Phi_9 \\ \Phi_{10} \\ \Phi_{11} \\ \Phi_{12} \\ \Phi_{13} \\ \Phi_{14} \\ \Phi_{15} \\ \Phi_{16} \end{Bmatrix}
=
\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \\ F_7 \\ F_8 \\ F_9 \\ F_{10} \\ F_{11} \\ F_{12} \\ F_{13} \\ F_{14} \\ F_{15} \\ F_{16} \end{Bmatrix}$$

Result …



Features of FEM applications

- Typical Procedures for FEM Computations
  - Input/Output
  - Matrix Assembling
  - Linear Solvers for Large-scale Sparse Matrices
  - Most of the computation time is spent for matrix assembling/formation and solving linear equations.
- **HUGE** “indirect” accesses
  - memory intensive
- Local “element-by-element” operations
  - sparse coefficient matrices
  - suitable for parallel computing
- Excellent modularity of each procedure

Information of this Class

- Instructor
  - Kengo Nakajima (Information Technology Center)
    - Information Technology Center (Asano) Annex 3F #36 ex: 22719
    - e-mail: nakajima(at)cc.u-tokyo.ac.jp
- Schedule
  - Monday's and some additional days
  - 08:30-10:15
  - **http://nkl.cc.u-tokyo.ac.jp/15w/**
- Practice
  - Time for exercise
- Lecture Room
  - Information Technology Center (Asano) Seminar Room #2 (1F)
    - No Foods, No Drinks

1	Sep.26(M)	0830-1015	Introduction, Introduction to FEM
2	Oct.03 (M)	0830-1015	1D FEM (1/2)
3	Oct.17 (M)	0830-1015	1D FEM (2/2)
4	Oct.24 (M)	0830-1015	3D FEM (1/2)
5	Oct.31 (M)	0830-1015	3D FEM (2/2)
6	Nov.07 (M)	0830-1015	Introduction to Parallel FEM, MPI (1/4)
-	Nov.14 (M)	-	(No Class)
7	Nov.21 (M)	0830-1015	MPI (2/4)
8	Nov.28 (M)	0830-1015	Report S1, MPI (3/4)
9	Dec.05 (M)	0830-1015	MPI (4/4)
10	Dec.12 (M)	0830-1015	Report S2, Parallel FEM (1/3)
11	Dec.19 (M)	0830-1015	Parallel FEM (2/3)
12	Jan.12 (Th)	0830-1015	Parallel FEM (3/3)
13	Jan.16 (M)	0830-1015	Hybrid OpenMP/MPI (1/2)
14	Jan.23 (M)	0830-1015	Hybrid OpenMP/MPI (2/2)

Prerequisites

- Knowledge and experiences in fundamental methods for numerical analysis (e.g. Gaussian elimination, SOR)
- Knowledge and experiences in UNIX
- Experiences in programming using FORTRAN or C

- Account for Educational Campuswide Computing System (ECC System) should be obtained in advance:
  - http://www.ecc.u-tokyo.ac.jp/ENGLISH/index-e.html
  - http://www.ecc.u-tokyo.ac.jp/user.html

Grading by Reports ONLY

- MPI (Collective Communication) (S1)
- MPI (1D Parallel FEM) (S2)
- Parallel FEM (P1)

- Sample solutions will be available
- Deadline: January 31st (Tue), 2017, 17:00
  - By E-mail: nakajima(at)cc.u-tokyo.ac.jp
  - You can bring hard-copy's to my office ...

Homepage

- **http://nkl.cc.u-tokyo.ac.jp/16w/**
  - General information is available
  - No hardcopy of course materials are provided (Please print them by yourself)

参考文献(1/2)

- 菊地「有限要素法概説(新訂版)」, サイエンス社, 1999.
- 竹内, 樫山, 寺田(日本計算工学会編)「計算力学:有限要素法の基礎」, 森北出版, 2003.
- 登坂, 大西「偏微分方程式の数値シミュレーション 第2版」, 東大出版会, 2003.
  - 差分法, 境界要素法との比較
- 福森「よくわかる有限要素法」, オーム社, 2005.
  - ヘルムホルツ方程式
- 矢川, 宮崎「有限要素法による熱応力・クリープ・熱伝導解析」, サイエンス社, 1985. (品切)
- Segerlind, L.(川井監訳)「応用有限要素解析 第2版」, 丸善, 1992.(品切)

参考文献(より進んだ読者向け)

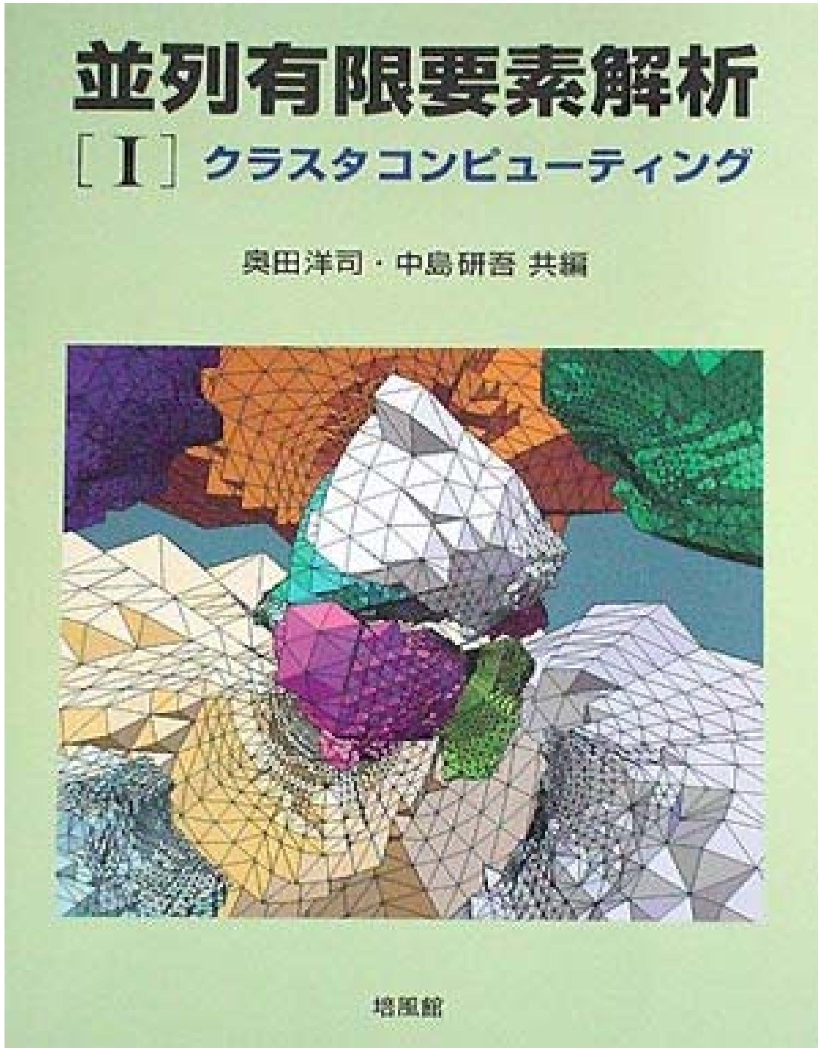
- 菊池, 岡部「有限要素システム入門」, 日科技連, 1986.
- 山田「高性能有限要素法」, 丸善, 2007.
- 奥田, 中島「並列有限要素法」, 培風館, 2004.

- Smith, I. 他「Programming the Finite Element Method (4th edition)」, Wiley.

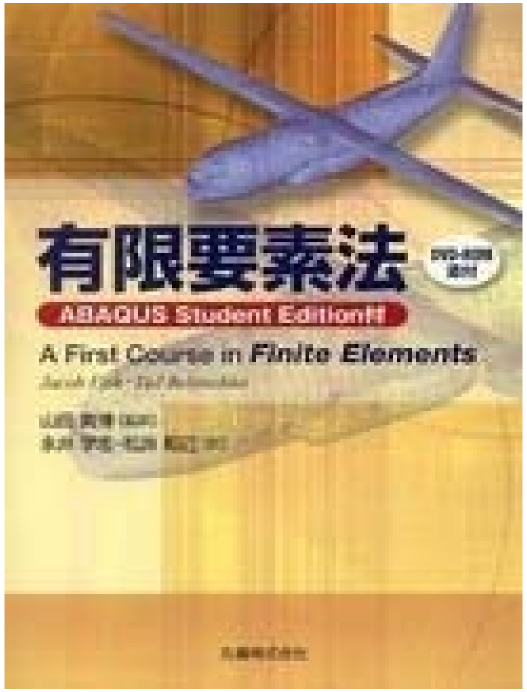
奥田，中島編「並列有限要素解析〔I〕クラスタコンピューティング」

培風館，2004.

- 「GeoFEM」の成果のまとめ
  - http://geofem.tokyo.rist.or.jp
- 「地球シミュレータ」上での最適化，シミュレーション結果を紹介
- 初心者向けでは無い
- 高い・・・
  - 若干残部があるので希望者には貸し出します。



References



- Fish, Belytschko, A First Course in Finite Elements, Wiley, 2007
  - Japanese version is also available
  - “ABAQUS Student Edition” included
- Smith et al., Programming the Finite Element Method (4th edition), Wiley, 2004
  - Parallel FEM
- Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Dover, 2000