

The 21st Century  
Earth Science COE Program  
University of Tokyo

# 局所細分化メッシュに基づく並列有限 要素法における前処理付き反復法

Preconditioned Iterative Methods for Parallel Finite-Element  
Applications with Adaptive Mesh Refinement

中島 研吾<sup>(1)</sup> 兵藤 守<sup>(2)</sup>

(1) 東京大学大学院 理学系研究科 地球惑星科学専攻

(2) 地球シミュレータセンター 固体地球シミュレーション研究グループ

日本応用数学会「行列・固有値問題の解法とその応用」研究部会 第2回研究会

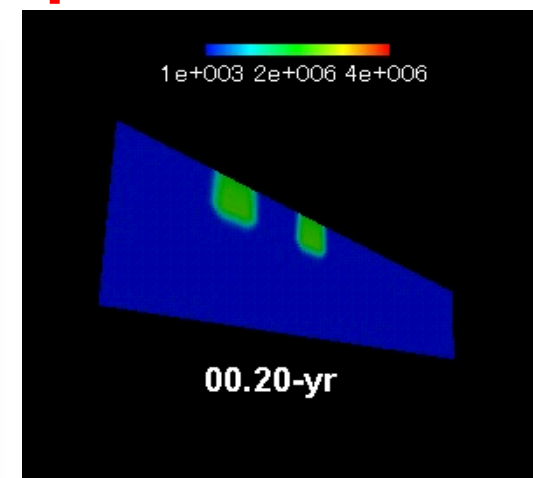
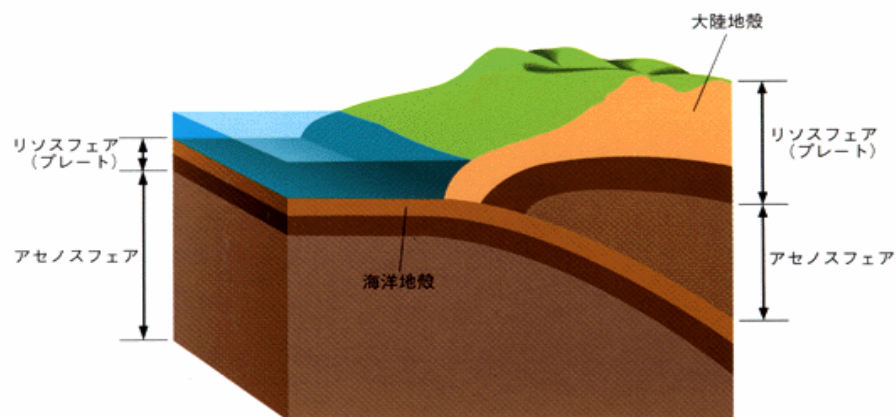
2006年11月16日, 国立情報学研究所

# 本発表の概要

- 背景
  - 地震発生サイクルシミュレーション
  - 局所細分化, 負荷分散
- 問題設定
- 計算結果
- まとめ
  - 今後の予定等

# 地震発生サイクルシミュレーション

- プレート境界における準静的応力蓄積過程
- もともとのアプローチ
  - 非線形接触問題をNewton-Raphson法によって解く
  - ALM法 (Augmented Lagrangean, 拡大ラグランジェ法) による拘束条件: ペナルティ数
  - 領域分割による並列有限要素法
    - GeoFEM, 地球シミュレータ
    - 前処理付き反復法 [Nakajima, 2003], [Nakajima, 2006]

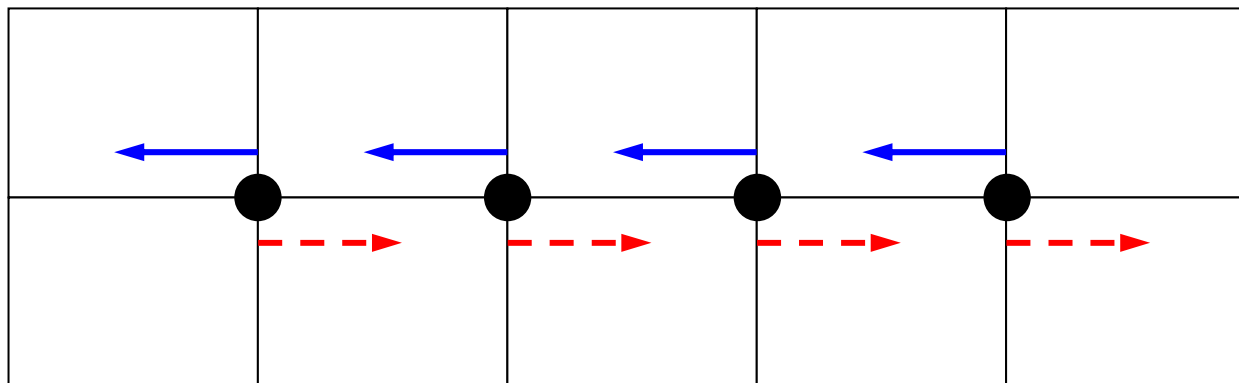


# 現状

- 2003年以降, 全てをFEMで解こうという考え方から方針転換
  - 接触モデリングの困難さ
  - 境界条件
- 伝統的な境界積分法, 境界要素法を使用する
  - 単位すべり応答関数(ある場所 $\xi$ ,  $t=0$ における単位すべり量が時間 $t-s$ , 場所 $x$ における応力変化を与える)による寄与の積分(重ね合わせ)
  - 不均質場における「単位すべり応答関数」を計算する必要がある
    - 有限要素法の使用

# FEMによる「単位すべり応答関数」計算

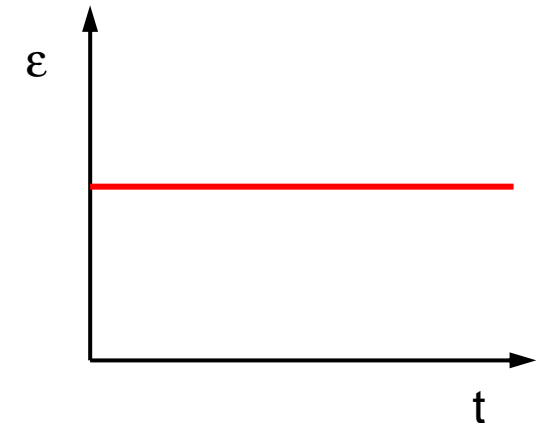
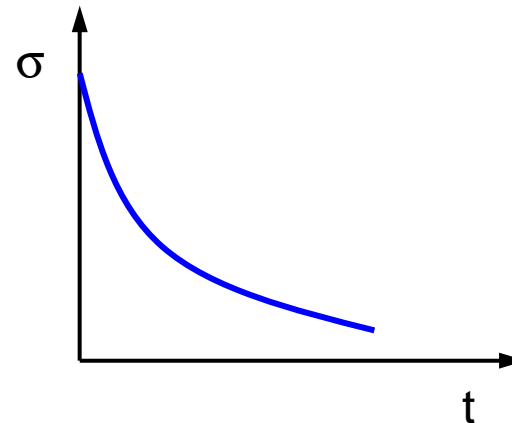
- GeoFEM
- 分割節点法 (Split Node Method) による, 食い違い変位 (Dislocation) モデル
  - 断層をはさんだ変位の食い違い量を仮定し, 仮想変位に相当する節点力を外力項として与える
  - 特殊な要素は不要, 定式化も簡単, 観測値の利用



# FEMによる「単位すべり応答関数」計算 (続き)

- 不均質粘弾性モデル
  - Maxwell粘弾性体

$$\frac{d\varepsilon}{dt} = \frac{1}{E} \frac{d\sigma}{dt} + \frac{1}{\eta} \sigma$$

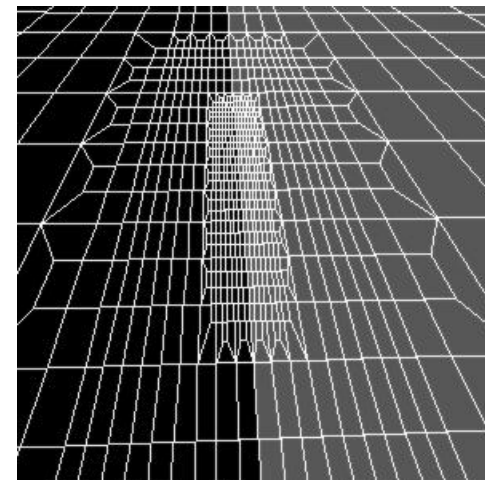
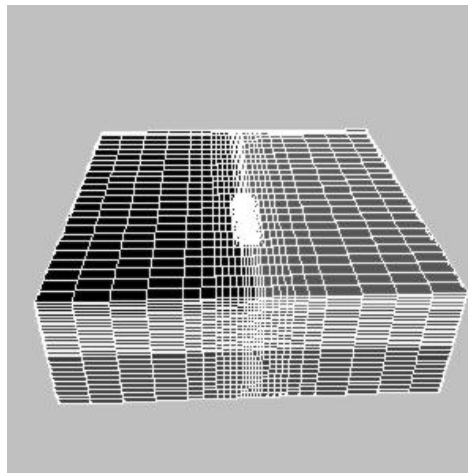
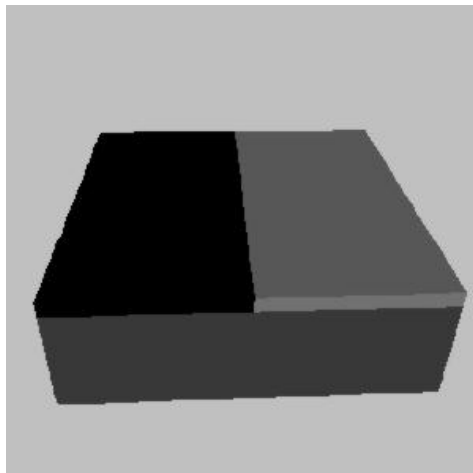


- ちなみに弾性体は・・・

$$\varepsilon = \frac{1}{E} \sigma$$

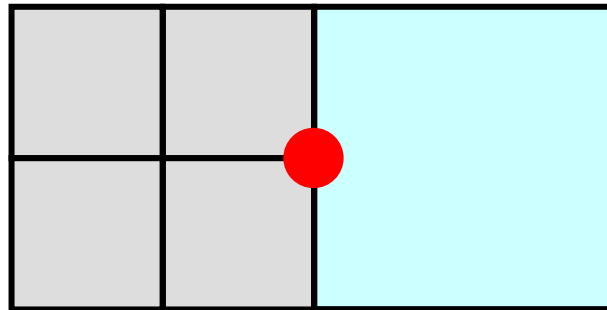
# 局所細分化の必要性

- $1\text{km} \times 1\text{km}$ の応答関数を求めるためには、数十m幅のメッシュが必要
- 数百km  $\times$  数百kmの領域を扱うため、全領域をこの大きさのメッシュで切ることは不可能
  - 局所細分化: Adaptive Mesh Refinement (AMR)



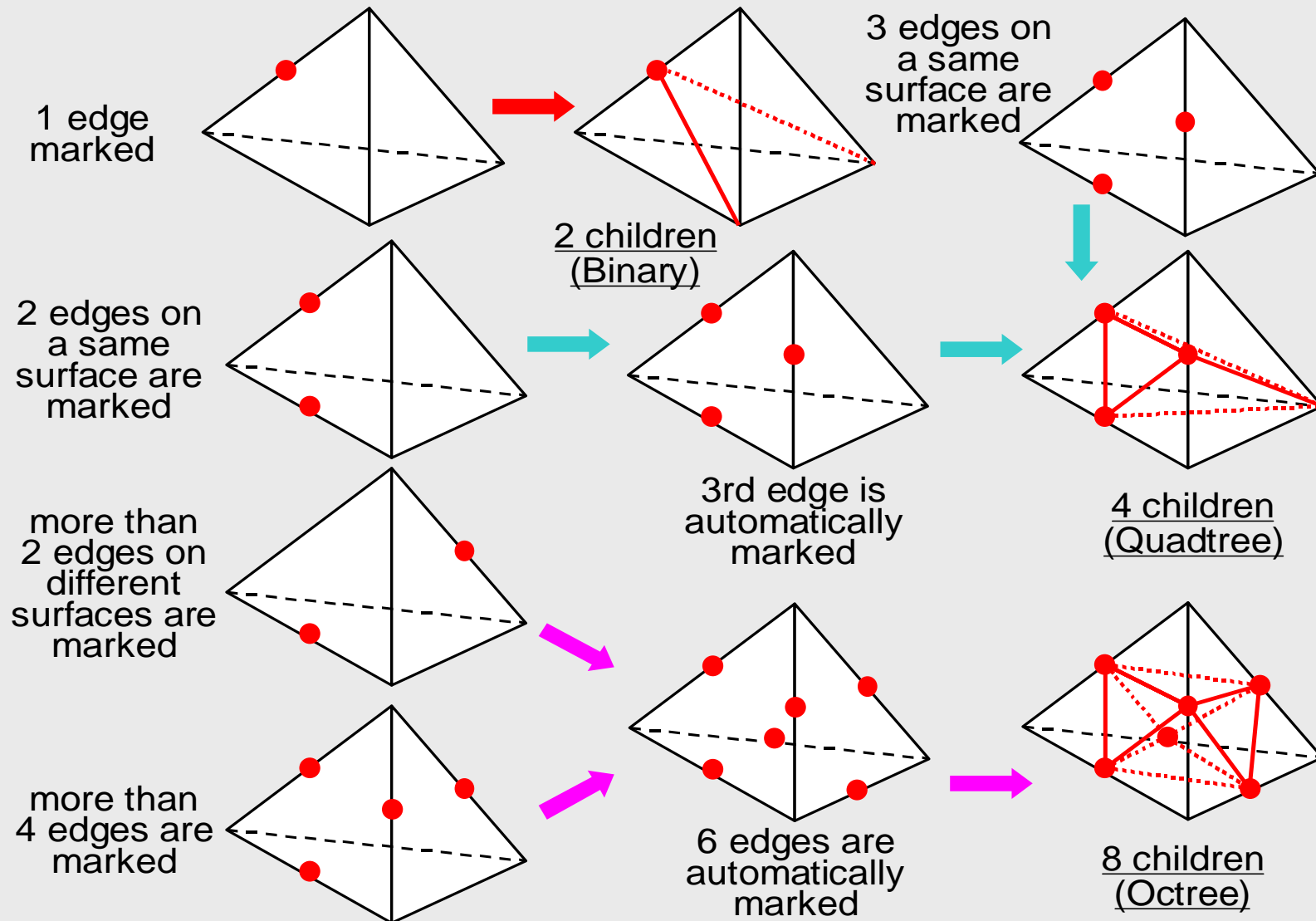
# 六面体要素のAMR

- 実は六面体要素のAMRは難しい
- 八分木(二次元では四分木)は細分化レベルの境界でHanging Nodeを生じてしまう
  - 四面体, 三角形ではこのようなことはないが, 断層のモデリングには六面体要素が望ましく, 六面体要素が一般的に用いられている



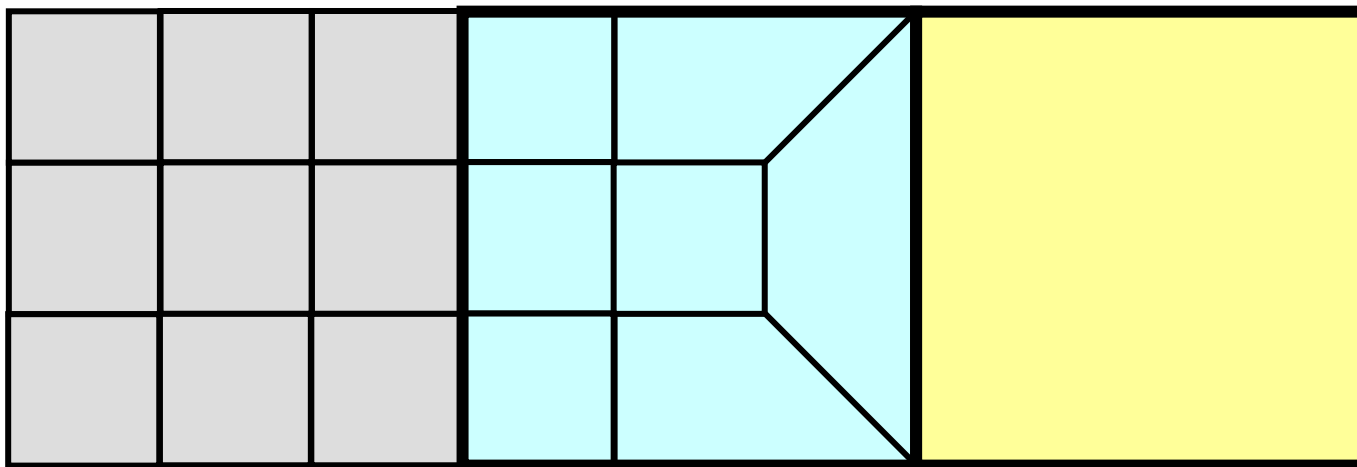


# Refinement Patterns of Tetrahedron

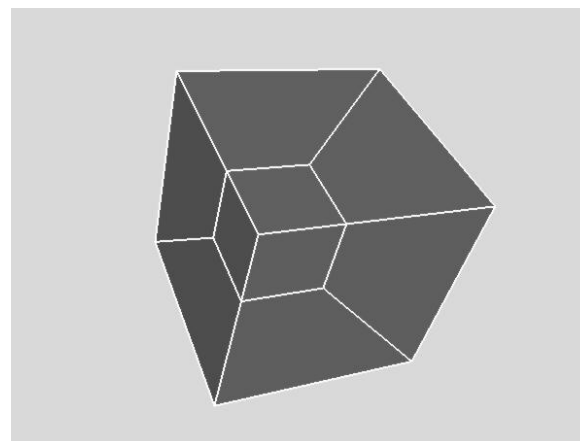
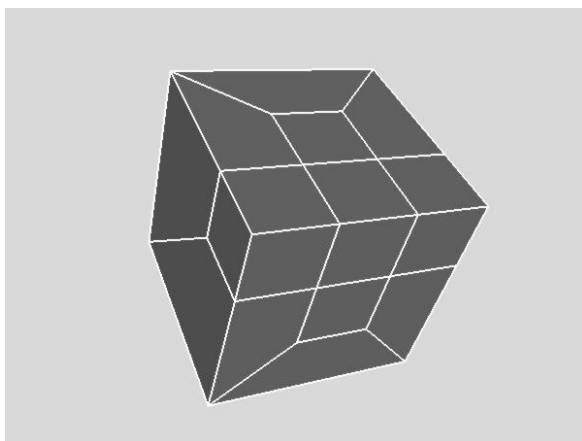
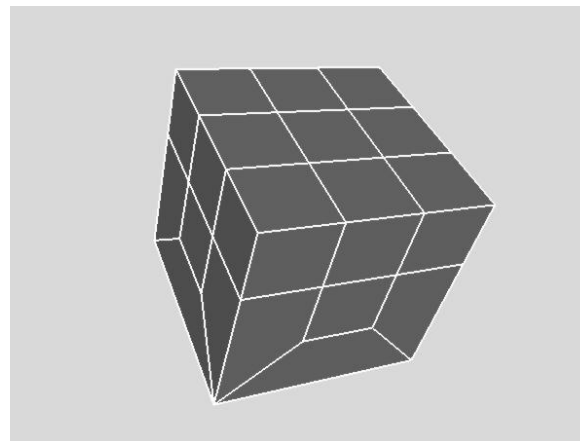
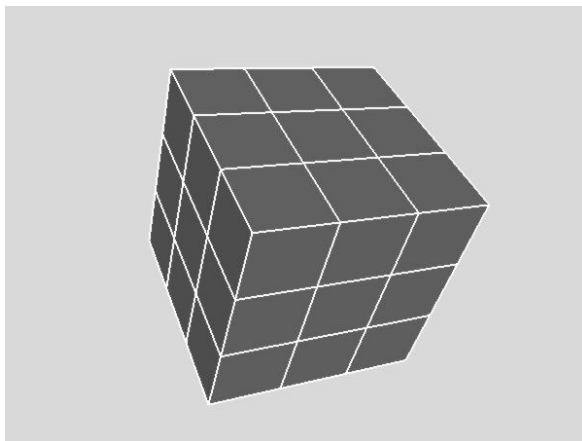


# 対処法: 27分木 [Wada, 2000]

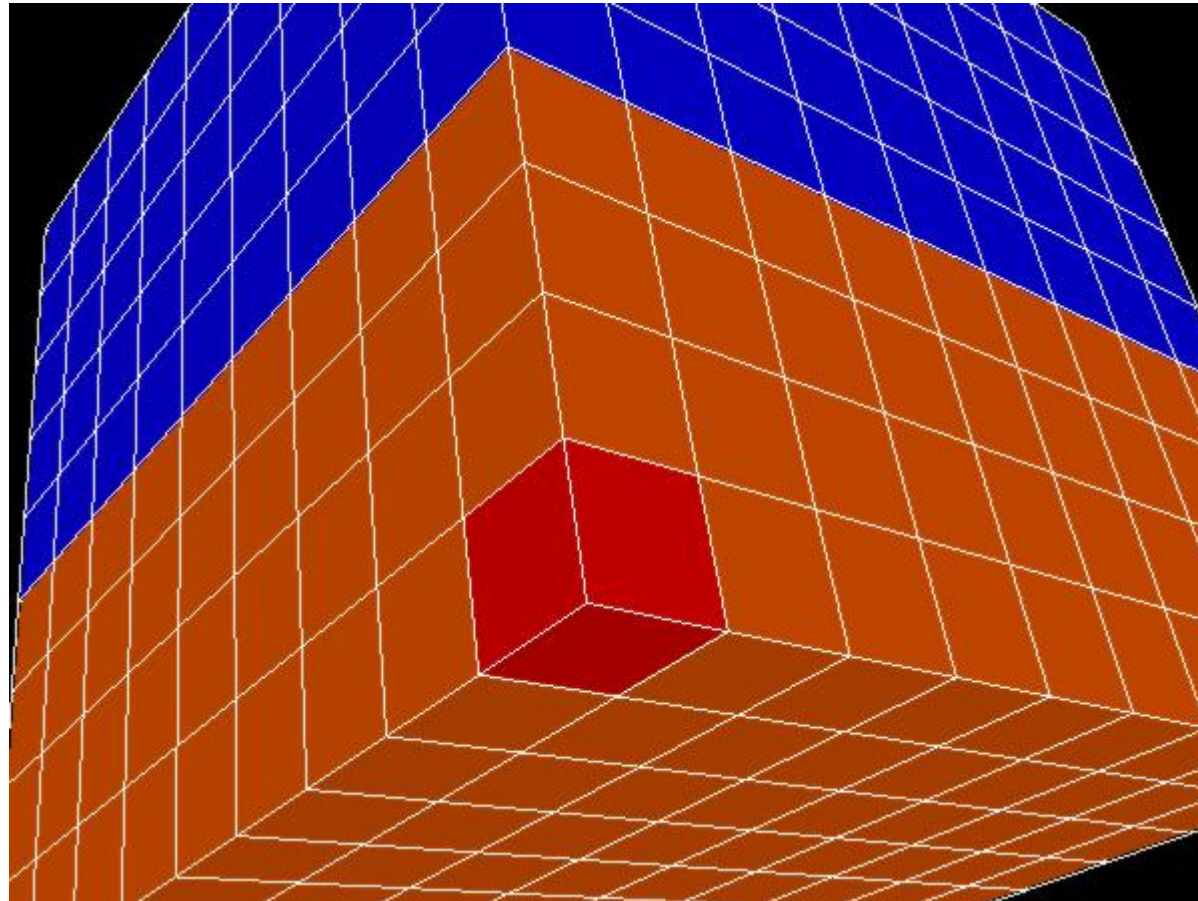
- 各辺の分割単位を2ではなく, 3とする。
  - 細分化レベル境界でHanging Nodeを生じない
  - 二次元: 9分木
  - 三次元: 27分木



# 27分木 における分割パターン [Wada, 2000]

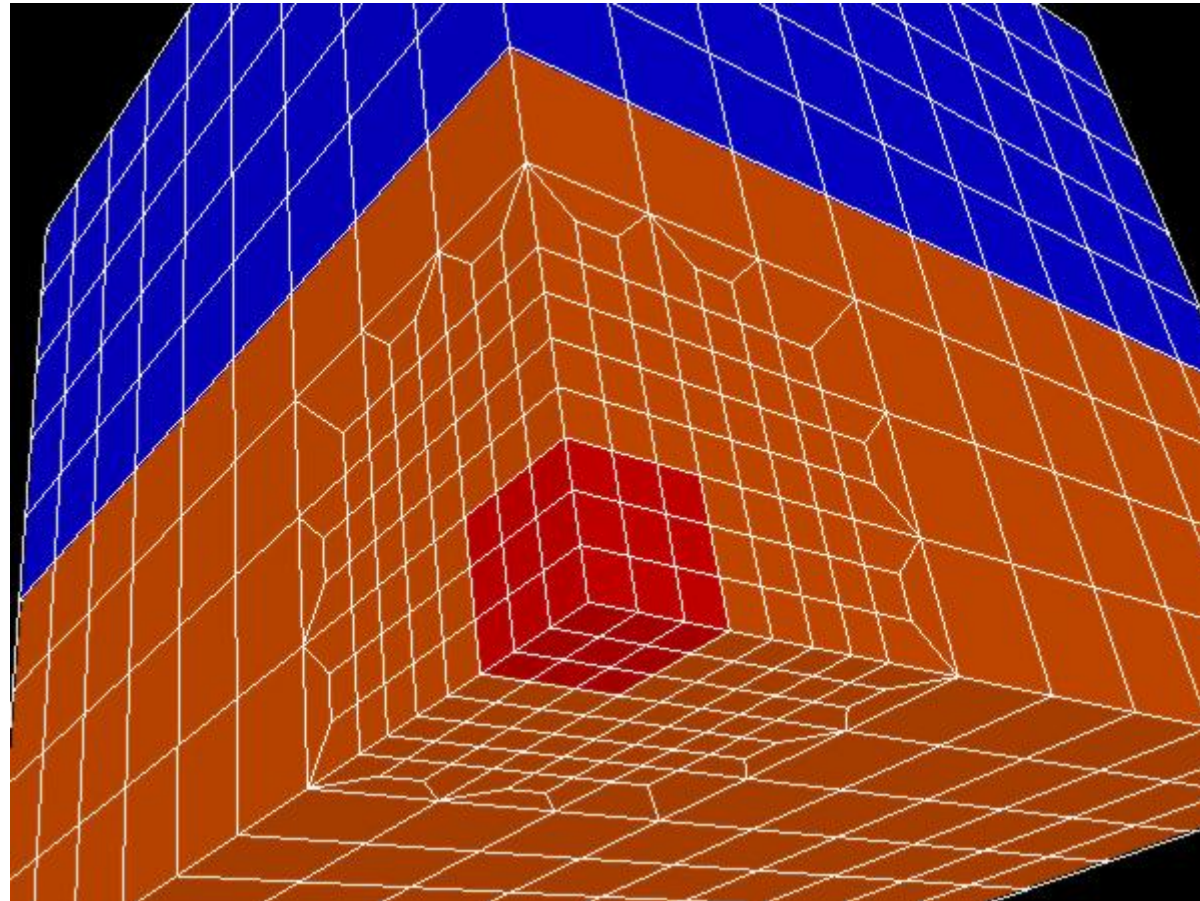


# 27分木 による細分化例 初期レベル



# 27分木 による細分化例

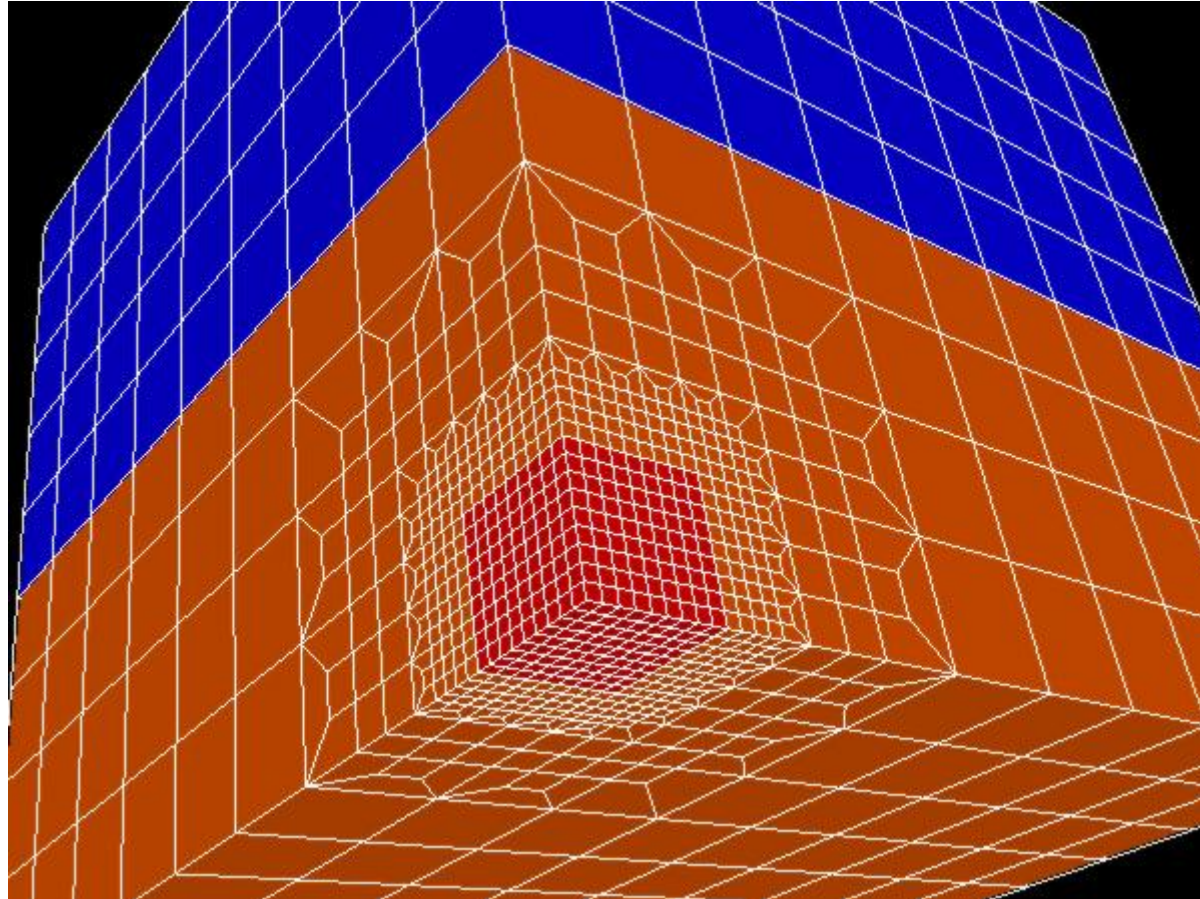
## 細分化レベル1





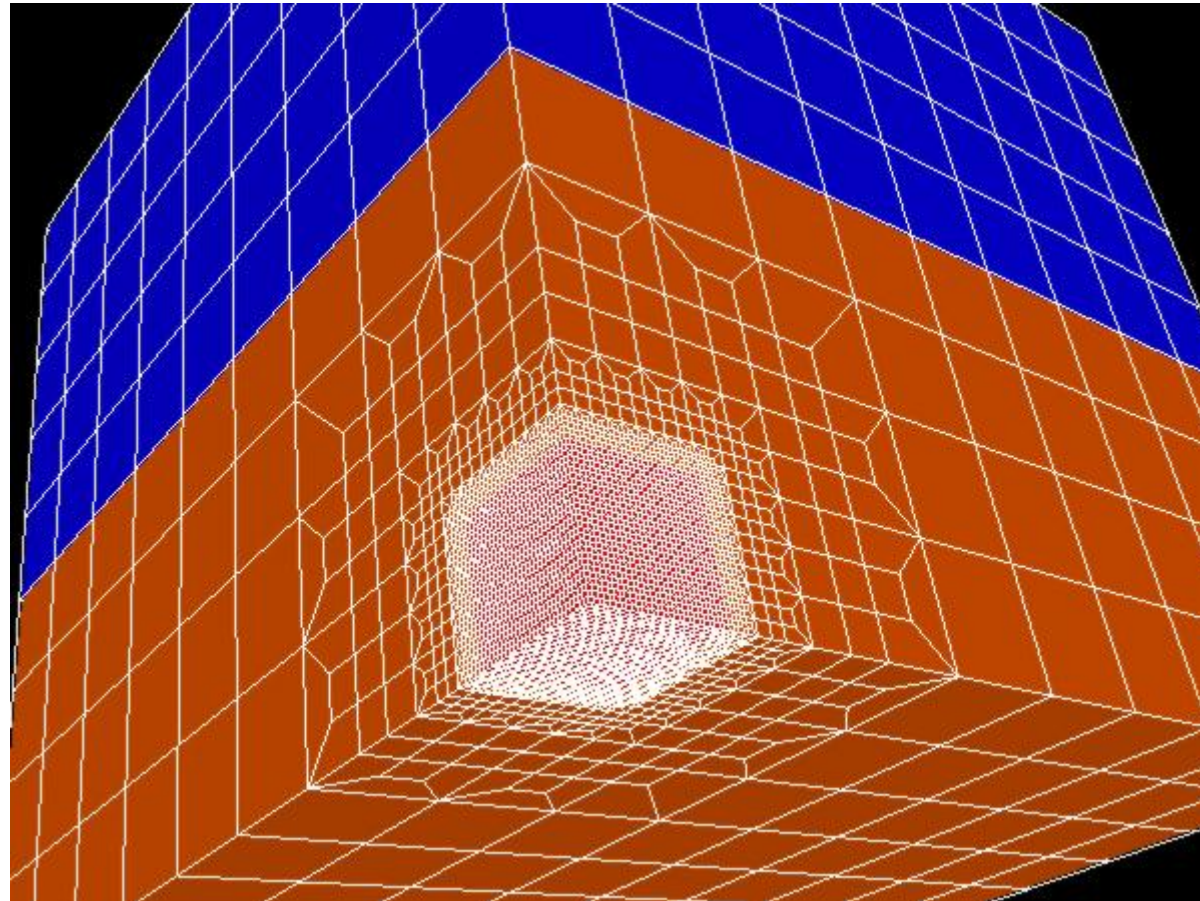
# 27分木 による細分化例

## 細分化レベル2



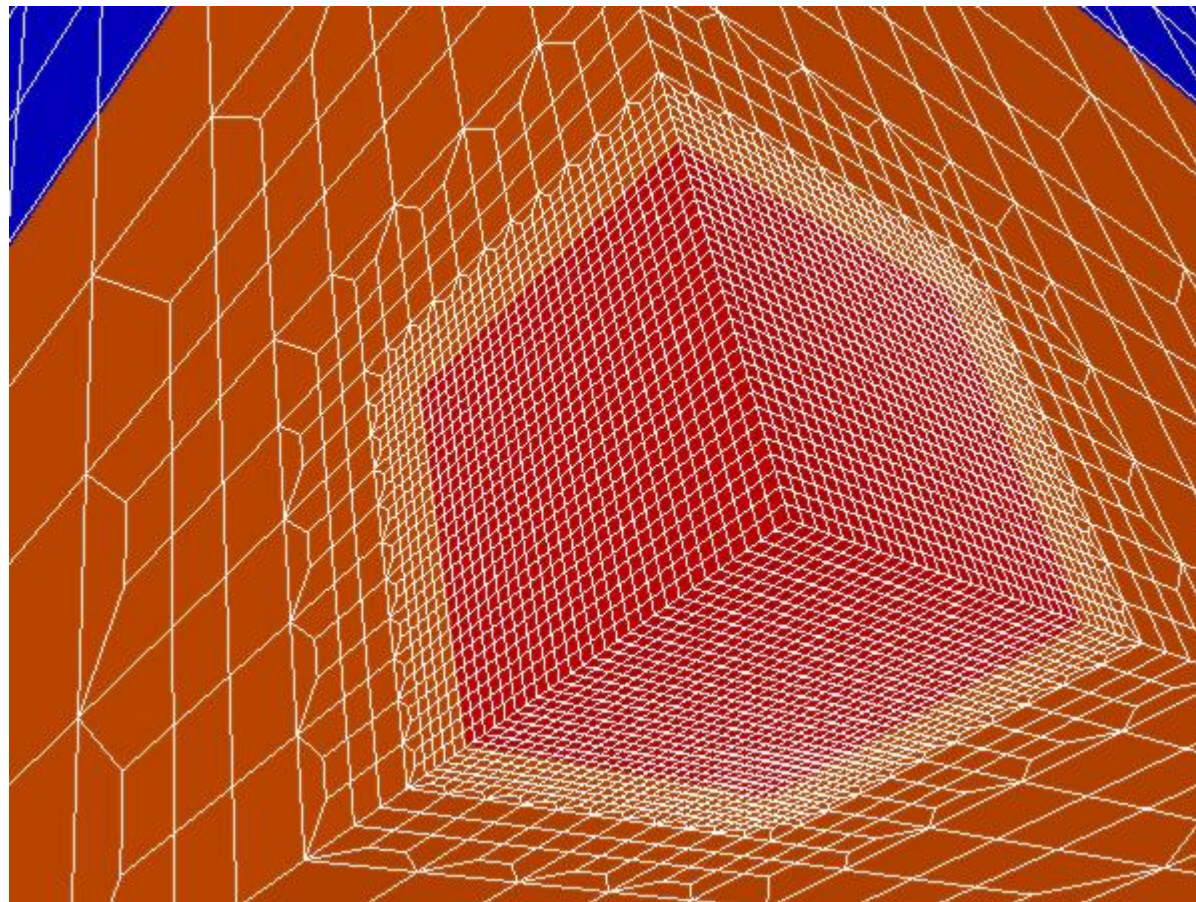
# 27分木 による細分化例

## 細分化レベル3





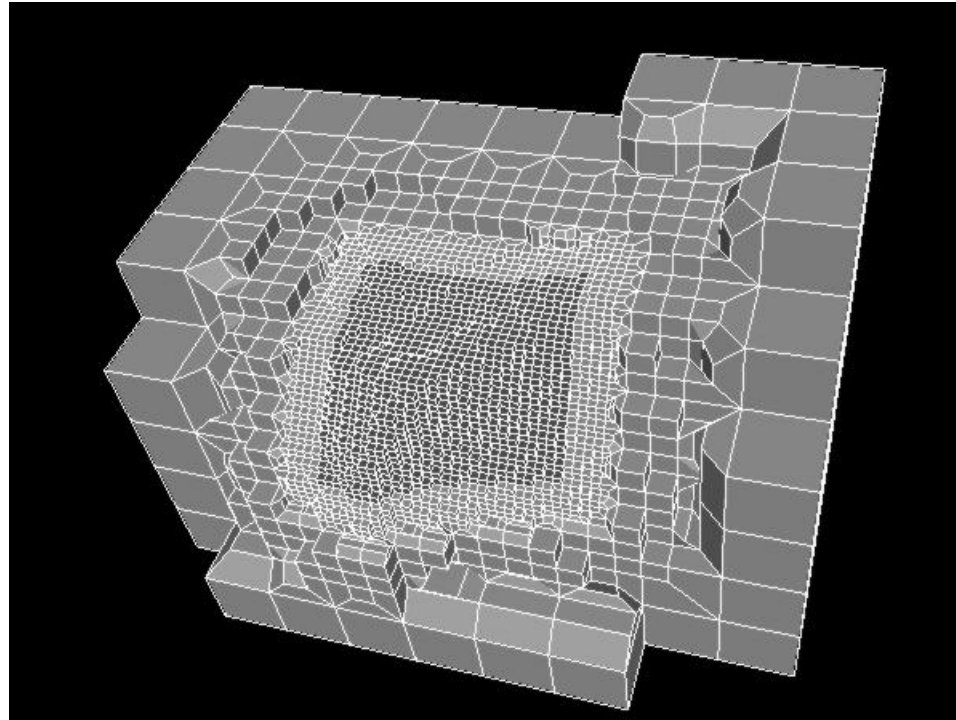
# 27分木 による細分化例 細分化レベル3(拡大図)





# 27分木

- 膨大な細分化メッシュ
- 8000要素  $\Rightarrow$  1つの要素を3レベルの分割しただけで67,000要素以上
  - となりあうメッシュの分割レベルの差は1以下とする場合:精度
- これまであまり使われなかった主な理由

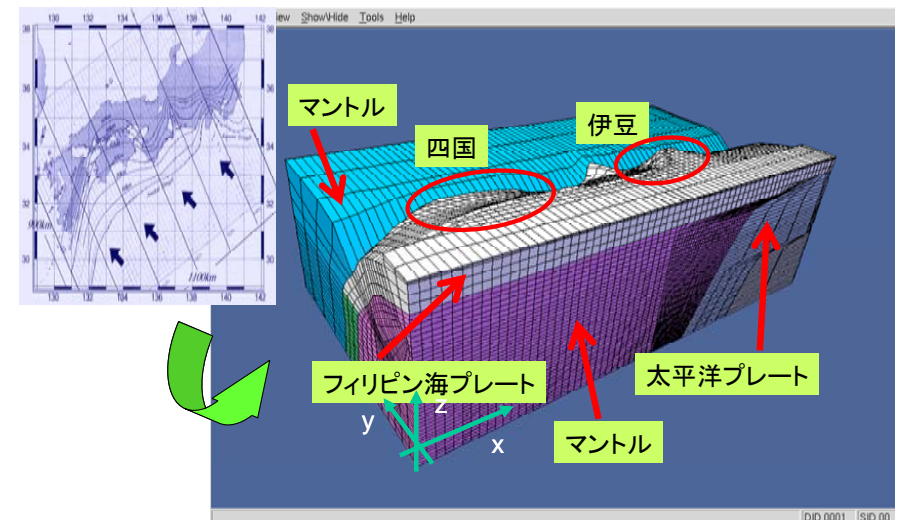


# 「並列」27分木

- 要素数, 節点数が100万を越えるような初期メッシュを局所的に細分化する必要がある
  - 細分化レベルを増やしていくと, 1プロセッサで扱うことは困難
- 色々な細分化パターンを実施する必要がある
  - 「すべり応答関数」を算出するのが目的であるため, 多数のメッシュパターンを「高速に」生成する必要がある

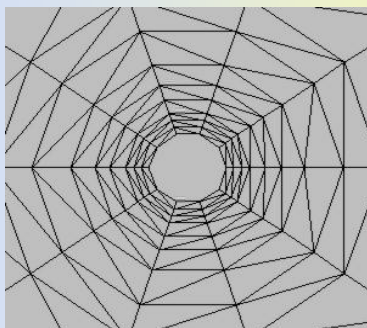
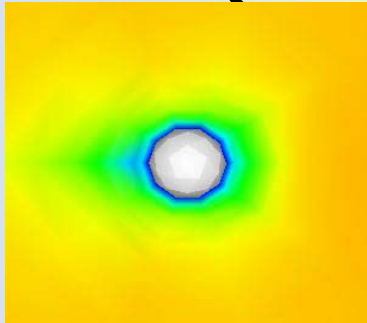
## ● 並列計算

- 初期メッシュ, 領域分割
- 「並列」細分化
- 動的負荷分散 (Dynamic Load Balancing) + データマイグレーション

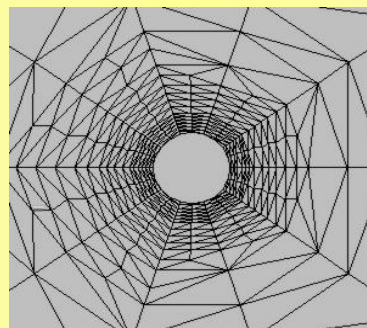
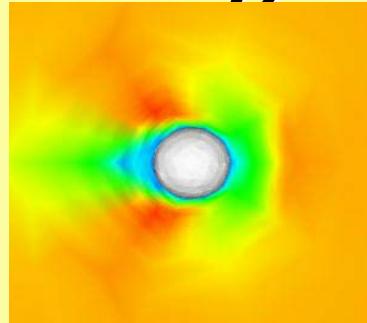


# Supersonic Flow around a Sphere

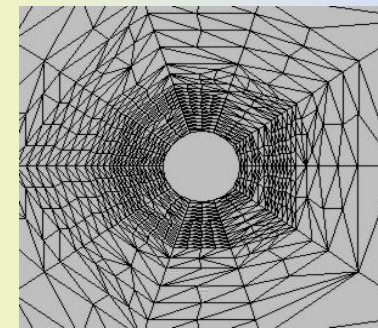
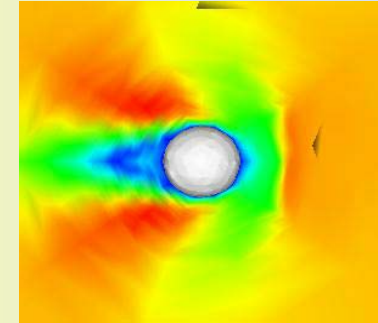
M=1.40 Uniform Flow, Ideal Gas, Re=10<sup>6</sup>  
 4 PE cases (tetrahedron only)



**Initial Grid**



**1-Lev. Adapted**



**2-Lev. Adapted**

	<u>before</u>	<u>DRAMA</u>	<u>before</u>	<u>DRAMA</u>	<u>before</u>	<u>DRAMA</u>
PE0	137	-	793	652	3834	2527
PE1	137	-	696	650	2769	2526
PE2	136	-	668	652	2703	2522
PE3	136	-	448	651	1390	2524

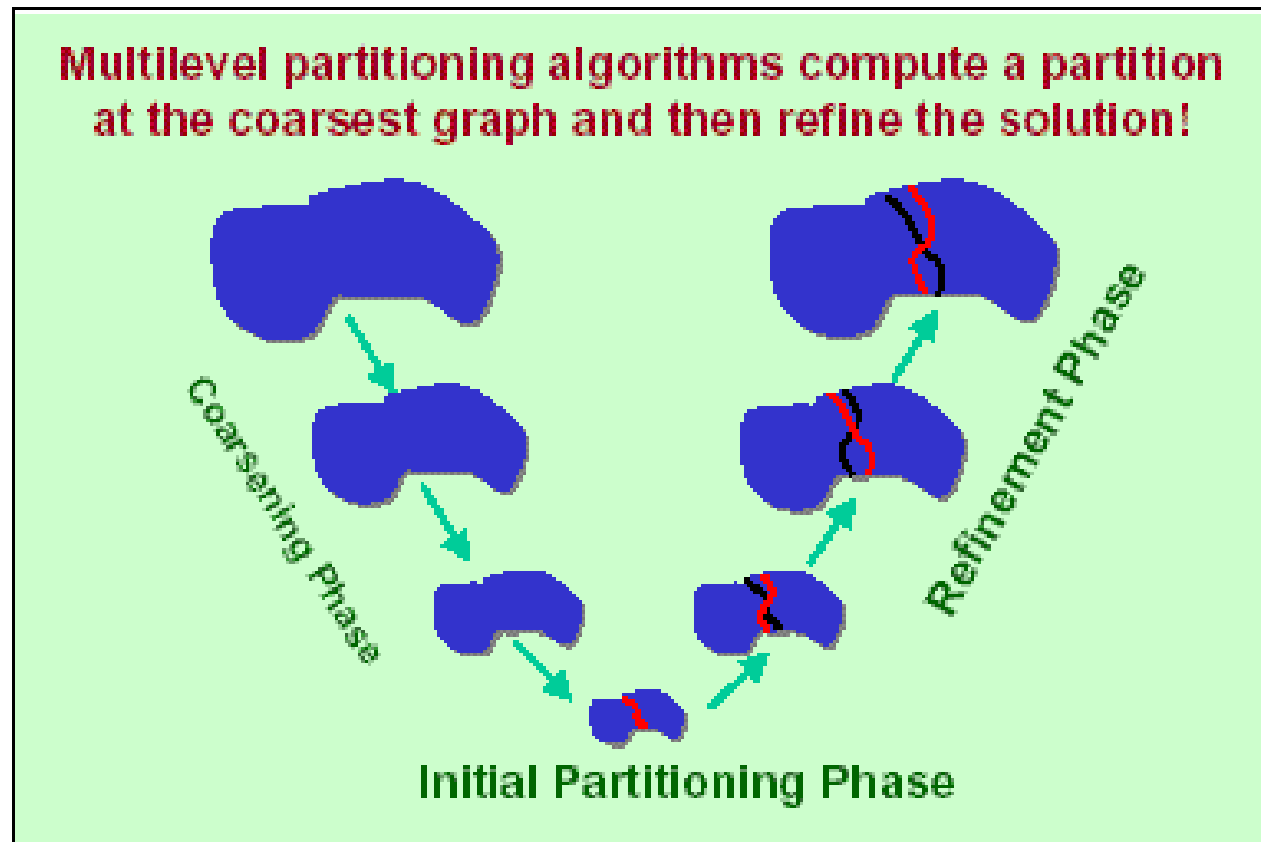
# 「並列」27分木の開発

- 並列細分化
  - Double Numberingの導入 [Nakajima & Fingberg, 2001]
  - 局所的な細分化情報の通信(未実装)
  - 四面体と比較すると膨大な組み合わせを考慮する必要がある
- ParMETISによる動的負荷分散
  - <http://glaros.dtc.umn.edu/gkhome/metis/parmetis/>
  - マルチレベル的な手法, 節点数がバランスするような手法
- データマイグレーション
  - 節点座標, 要素コネクティビティ, 物性値, グループ情報, 通信テーブル等の領域間の移動(実装済み, 最適化はしていない)
  - 領域間オーバーラップ深さの自動変更機能

# METIS

<http://www-users.cs.umn.edu/~karypis/metis/>

- マルチレベルグラフ理論に基づいた方法



# METIS

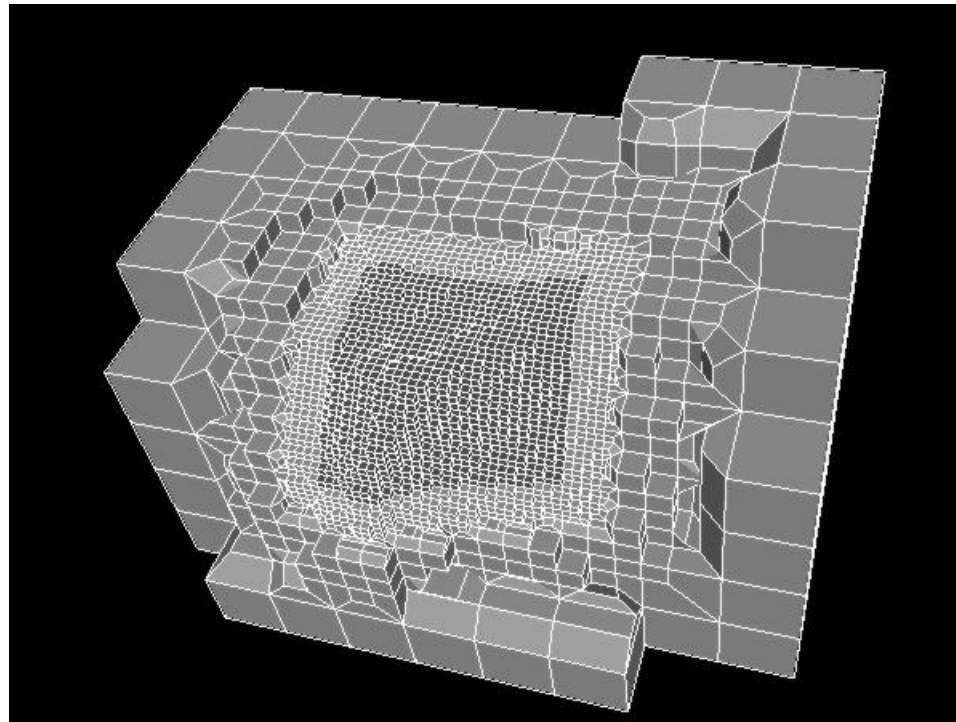
<http://www-users.cs.umn.edu/~karypis/metis/>

- マルチレベルグラフ理論に基づいた方法
  - 特に通信 (edge-cut) が少ない分割を提供する
  - 安定, 高速
  - フリーウェア, 他のプログラムに組み込むことも容易
- 色々な種類がある
  - k-METIS      通信量 (edge-cut) 最小
  - p-METIS      領域間バランス最適化
  - ParMETIS     並列版
  - 領域分割だけでなく, オーダリング, データマイニングなど色々な分野に使用されている
    - 接触, 衝突問題における並列接触面探索

- 背景
  - 地震発生サイクルシミュレーション
  - 局所細分化, 負荷分散
- **問題設定**
- 計算結果
- まとめ
  - 今後の予定等

# 懸念

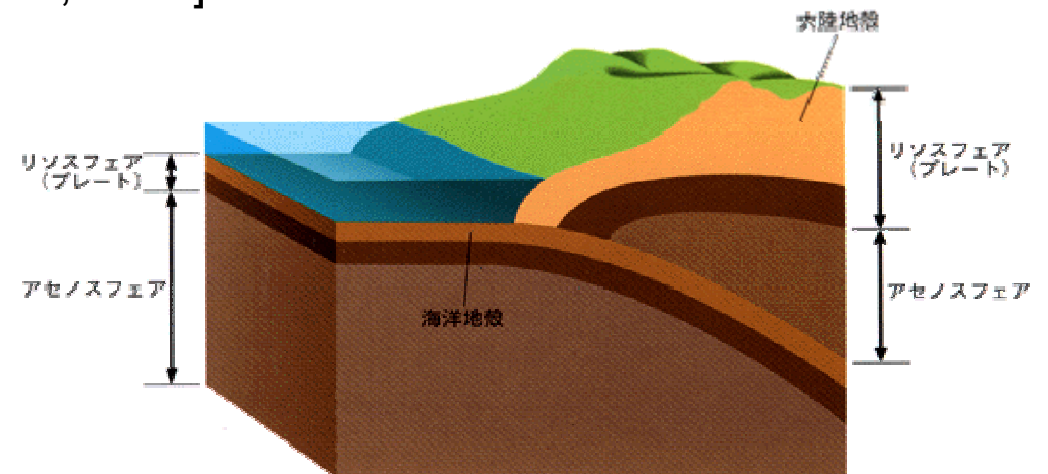
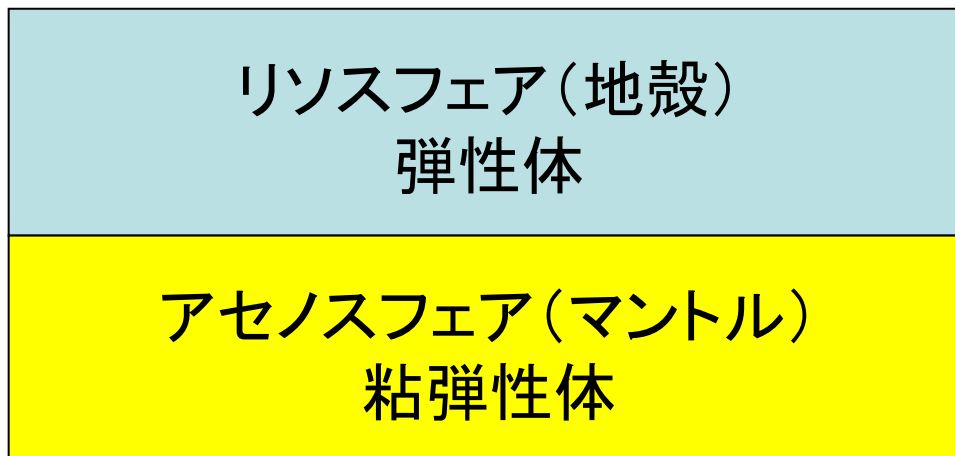
- 細分化によって、要素の幅が大きく分布するため、反復法の収束が悪化するのではないか・・・
- ParMETISを使用して、ブラックボックス的に領域分割を実施すると、並列時の収束が悪化するのではないか・・・





# 例題 (1/3)

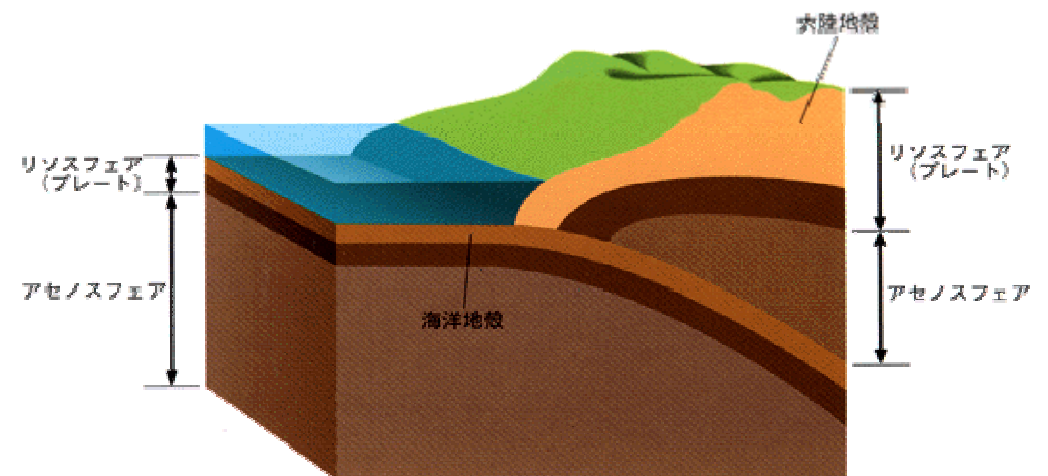
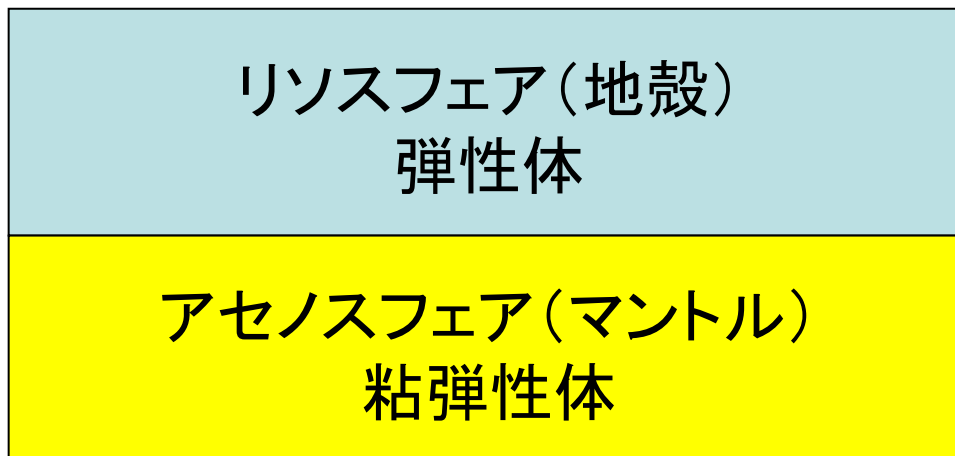
- 三次元弾性問題
- 初期メッシュ
  - 1領域  $20^3$ 要素, 64領域(約150万自由度)
  - z方向
    - 上10層:弾性体(リソスフェア, 地殻)
    - 下10層:粘弾性体(アセノスフェア, マントル)
      - 剛性率の低い弾性体(ヤング率:小さい, ポアソン比:大きい)としてモデル化
      - $t \rightarrow \infty$ で成立する近似 [Fukahata, 2006]



# 例題 (2/3)

- 細分化メッシュ

- 4領域に1領域の割合で, 1要素を3段階に細分化
- 70,000メッシュ近い領域が出てくる
- 全体として, 約150万要素(節点), 450万自由度(正確には1,480,960節点, 4,442,880自由度)
- ParMETISによる動的負荷分散+データマイグレーション



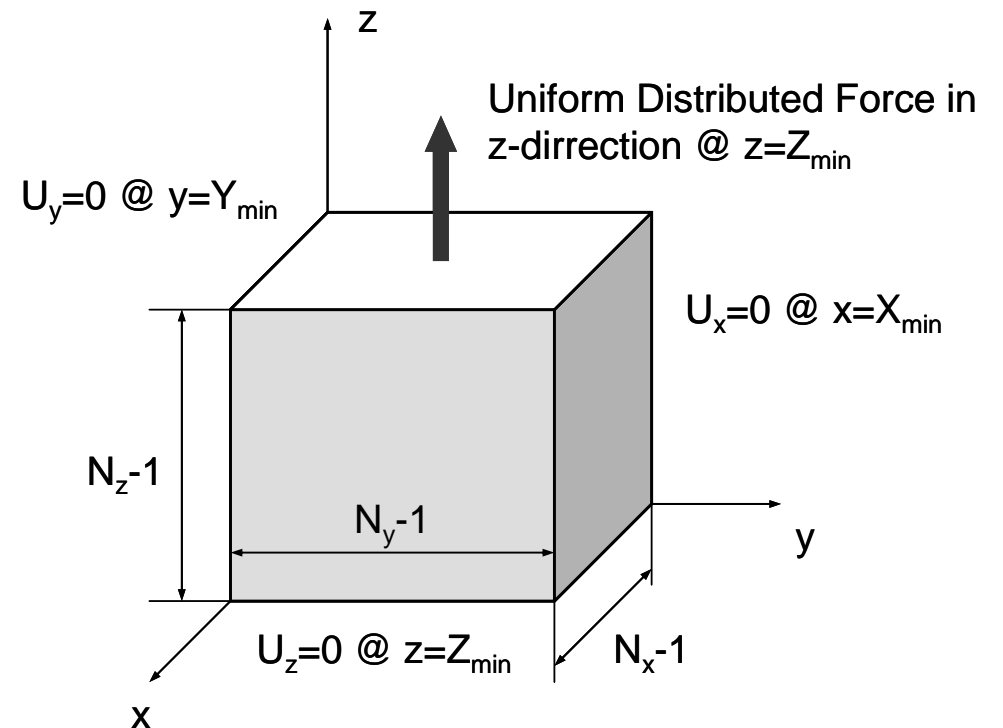
# 例題 (3/3)

- 境界条件

- 右図参照
- 現実的なもの(食い違い変位)ではない

- 物性値

- リソスフェア
  - ヤング率 1.00, ポアソン比:0.24
- アセノスフェア
  - ヤング率 0.01, ポアソン比:0.47  
を基準



# つりあい方程式

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \bar{X} = 0$$

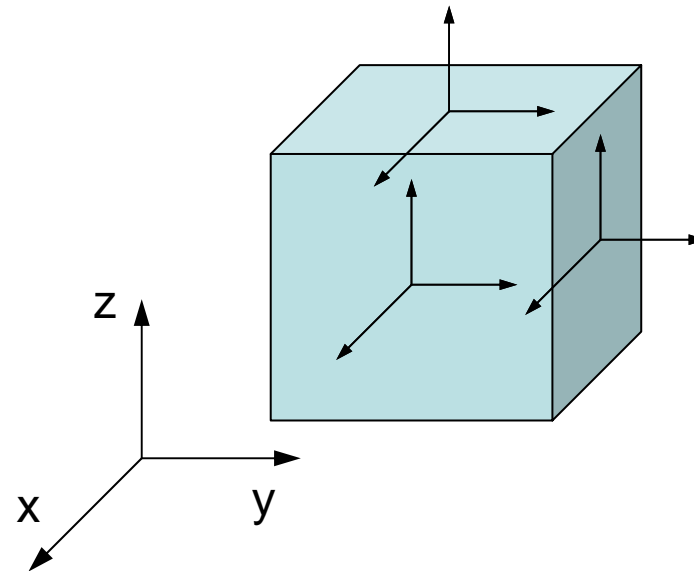
$$\frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yz}}{\partial z} + \bar{Y} = 0$$

$$\frac{\partial \sigma_z}{\partial z} + \frac{\partial \tau_{zx}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \bar{Z} = 0$$

$\sigma_x, \sigma_y, \sigma_z$  : 垂直応力

$\tau_{xy}, \tau_{yz}, \tau_{zx}$  : せん断応力

$\bar{X}, \bar{Y}, \bar{Z}$  : 物体力



# 変位－ひずみ関係式

$$\varepsilon_x = \frac{\partial u}{\partial x}, \quad \varepsilon_y = \frac{\partial v}{\partial y}, \quad \varepsilon_z = \frac{\partial w}{\partial z}$$

$$\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}, \quad \gamma_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}, \quad \gamma_{zx} = \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}$$

# 応力－ひずみ関係式(弾性)

$$\sigma_{x_j} = \lambda e + 2\mu \varepsilon_{x_j} \quad (x_j = x, y, z)$$

$$\tau_{x_i x_j} = \mu \gamma_{x_i x_j} \quad (x_i, x_j = x, y, z)$$

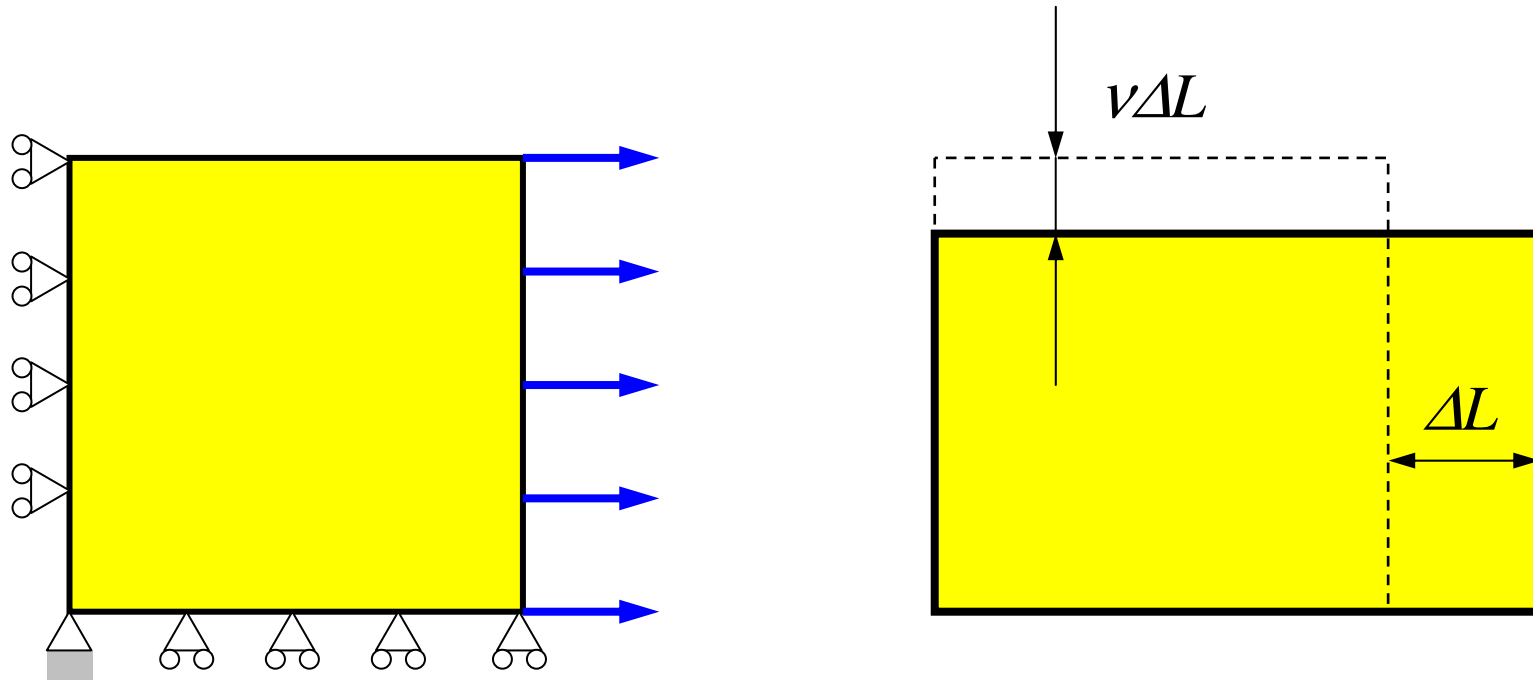
$$e = \varepsilon_x + \varepsilon_y + \varepsilon_z$$

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \quad \mu = \frac{E}{2(1+\nu)}$$

$E$ : ヤング率

$\nu$ : ポアソン比 (0~0.5)

# ポアソン比 $\nu$



ポアソン比=0.50:

伸びた分だけ縮む  $\Rightarrow$

非圧縮: 解きにくい

特殊な要素を使うことがある ( $\nu=0.4999\dots$  とすると非物理的挙動)

# 応力－ひずみ関係式(弾性)

$$\{\sigma\} = [D]\{\varepsilon\}$$

$$[D] = \frac{\nu E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \nu/(1-\nu) & \nu/(1-\nu) & 0 & 0 & 0 \\ & 1 & \nu/(1-\nu) & 0 & 0 & 0 \\ & & 1 & 0 & 0 & 0 \\ & & & \frac{1-2\nu}{2(1-\nu)} & 0 & 0 \\ & & & & \frac{1-2\nu}{2(1-\nu)} & 0 \\ & & & & & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix}$$

(Sym.)

ポアソン比=0.50:

伸びた分だけ縮む ⇒

非圧縮: 解きにくい

特殊な要素を使うことがある ( $\nu=0.4999\dots$  とすると非物理的挙動)



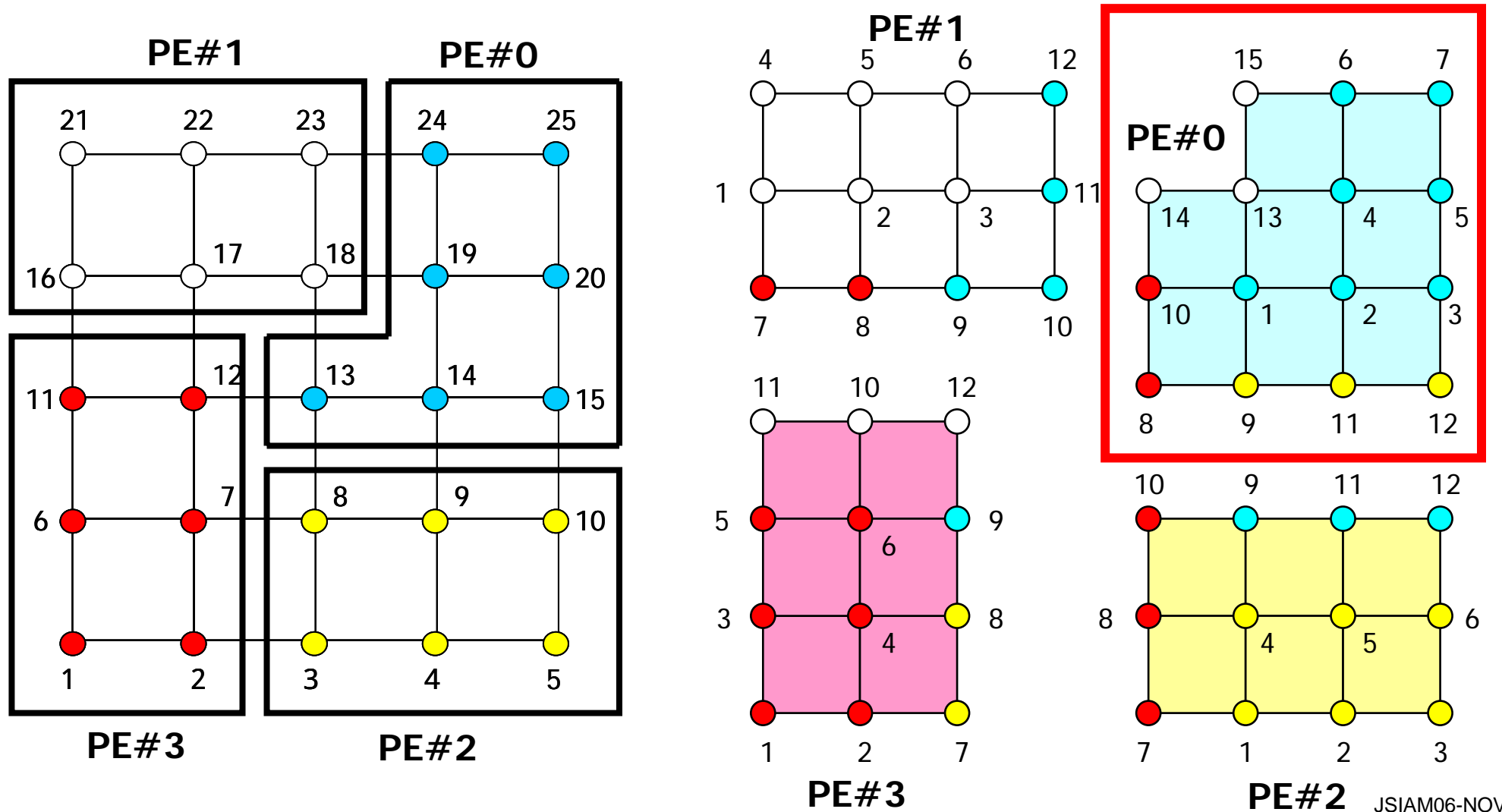
# 計算手法, 環境等

- 前処理付き反復法
  - GPBiCG
  - Block SGS (Symmetric Gauss-Seidel)
    - 並列, ポアソン比大の場合BILU(0), BILU(1), BILUT等では解けない場合あり
- 並列計算
  - 一層のオーバーラップ(次ページ参照)
  - ブロックヤコビ型局所前処理
- オーダリング
  - ベクトルプロセッサ, 多段階並列化
  - 初期
  - Multicolor, Cuthill-McKee
- PCクラスタ
  - Opteron 275 (2.2GHz) × 64コア, PGIコンパイラ+MPI

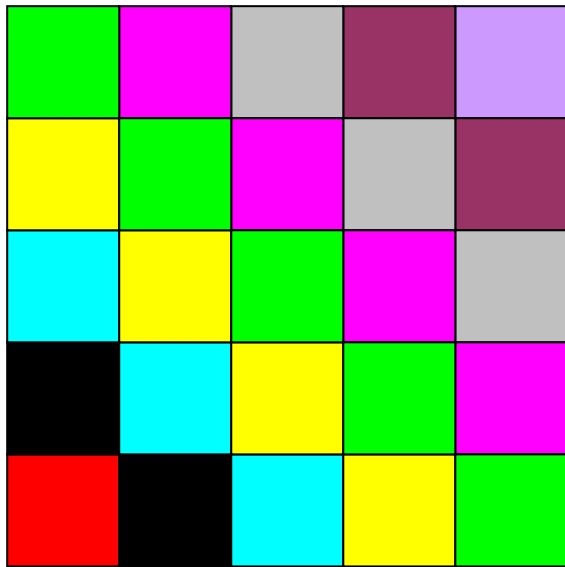
# Local Data Structure

## Node-based Partitioning

internal nodes - elements - external nodes

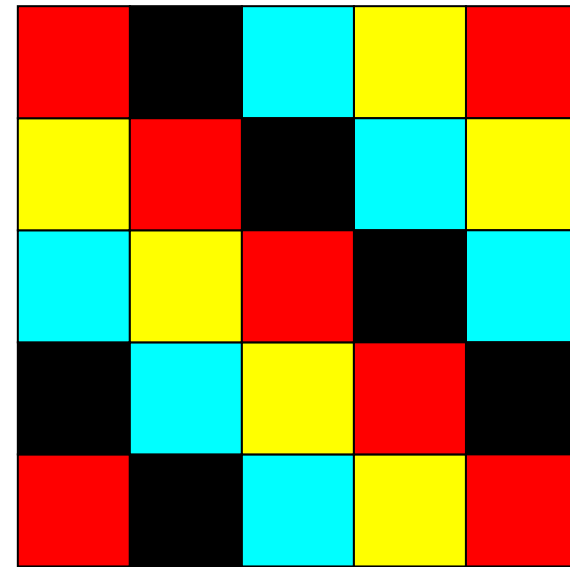


# オーダリング手法



CM (Cuthill-McKee) :

- 要素間結合グラフをもとにオーダリング
- 色間の依存性も考慮

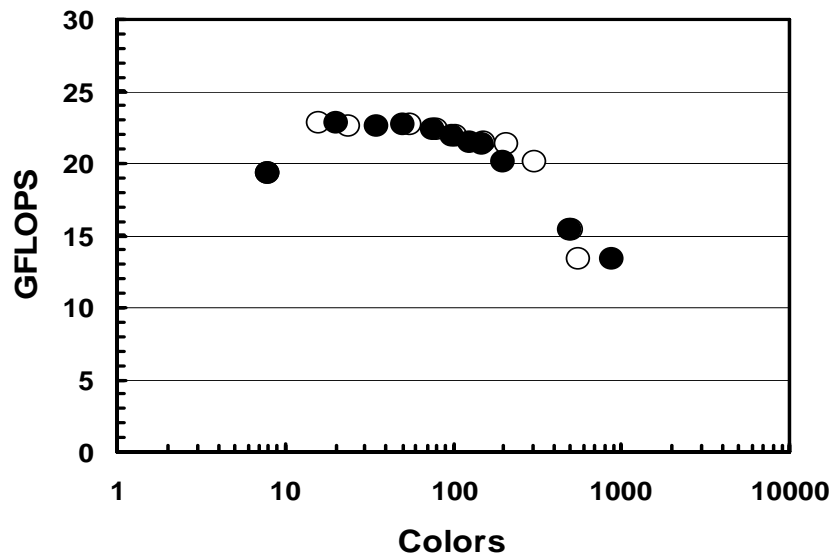
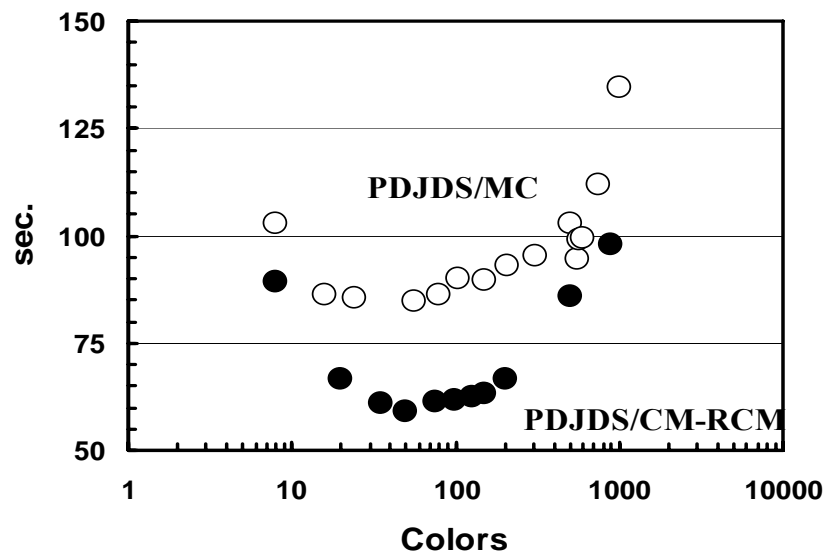
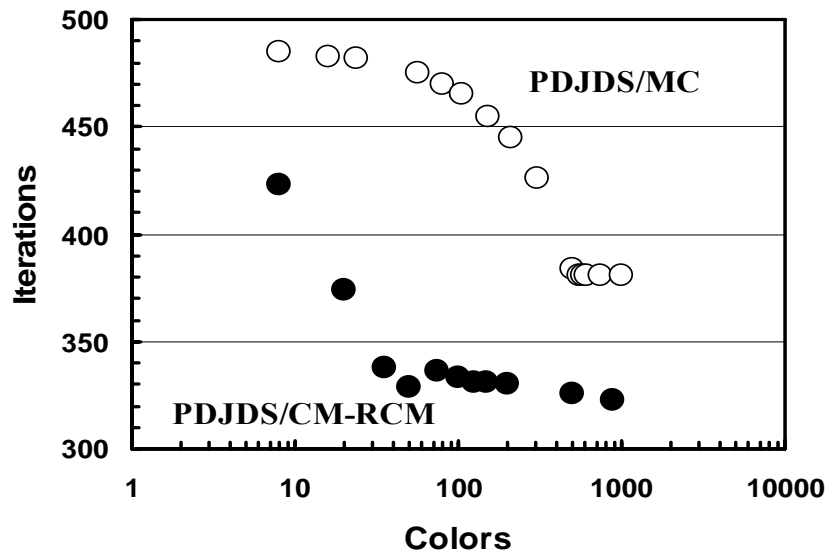


MC (Multicolor, 4色) :

- 要素間結合グラフをもとにオーダリング
- 並列性, 独立性重視

# 色数と計算効率 (ICCG) 3D弾性問題

128<sup>3</sup>節点, 「地球シミュレータ」1ノード

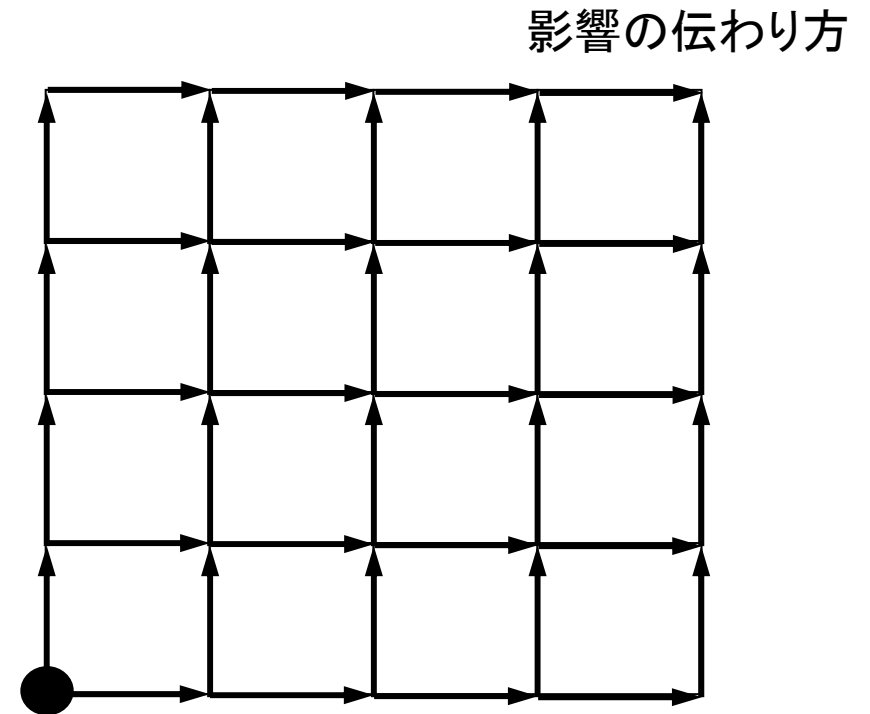
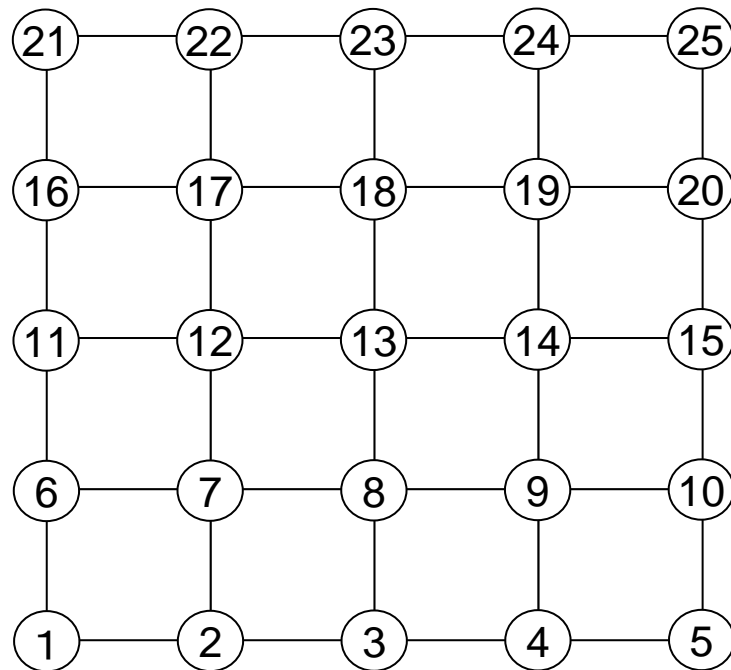


- オーダリング手法
  - PDJDS/MC: マルチカラー
  - PDJDS/CM-RCM
- 色数が増えると反復回数は減る
- FLOPS値は低下, したがって, 色数が増えても計算時間が減るとは限らない。

# 反復回数と色数の関係

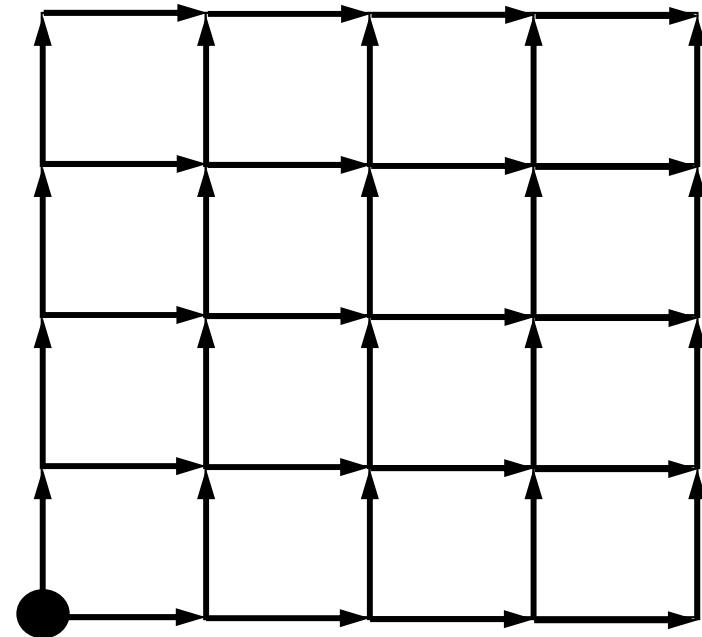
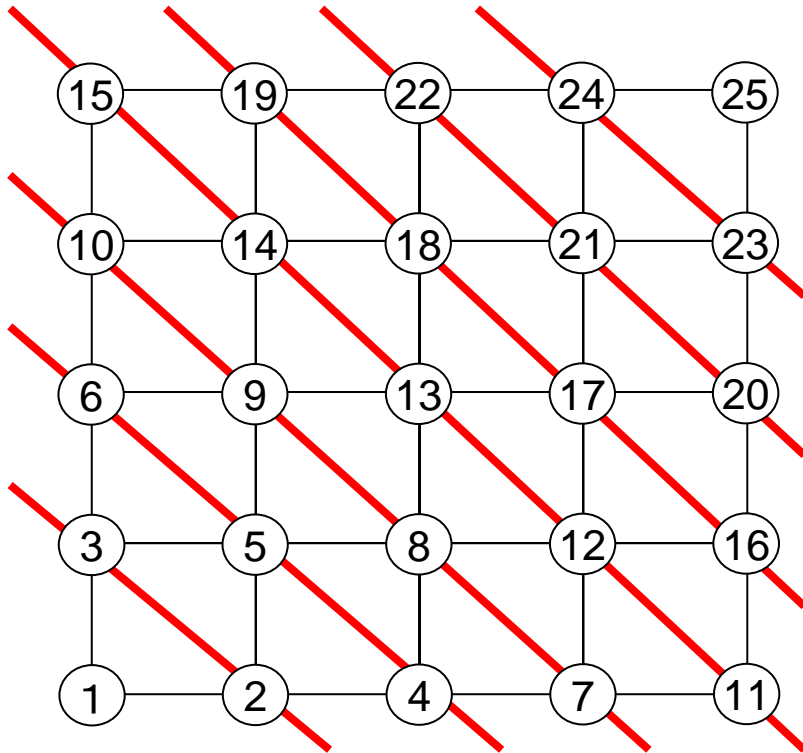
## Incompatible Nodesとは?

Doi, S. (NEC) et al.



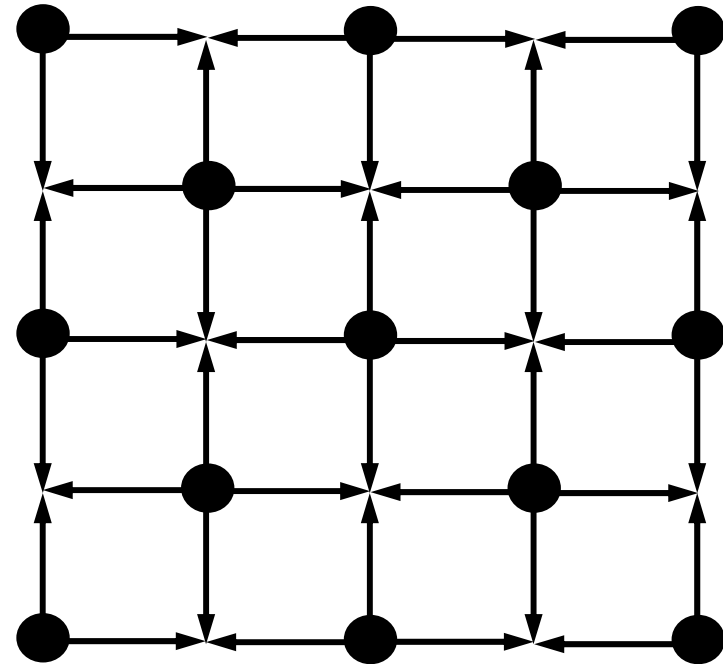
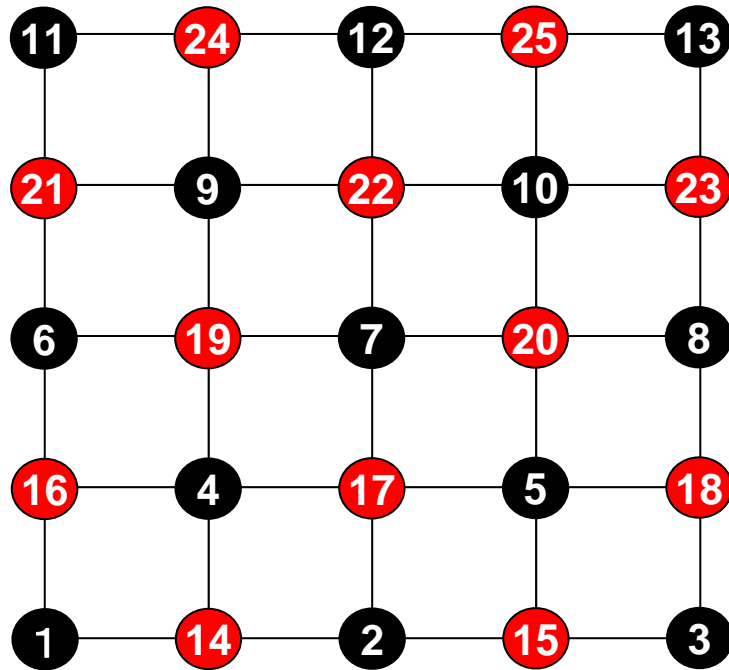
他から影響を受けない点が「Incompatible Node」  
少ない方が収束が早い

# RCMの場合



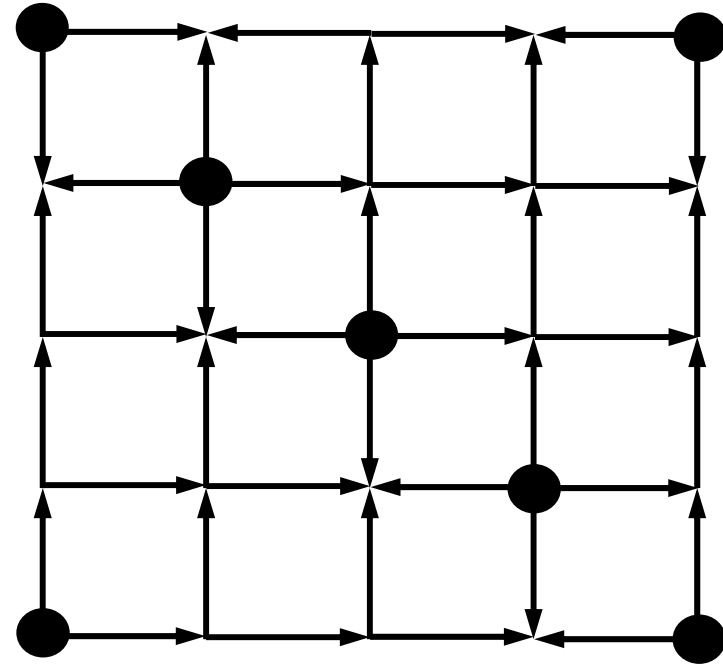
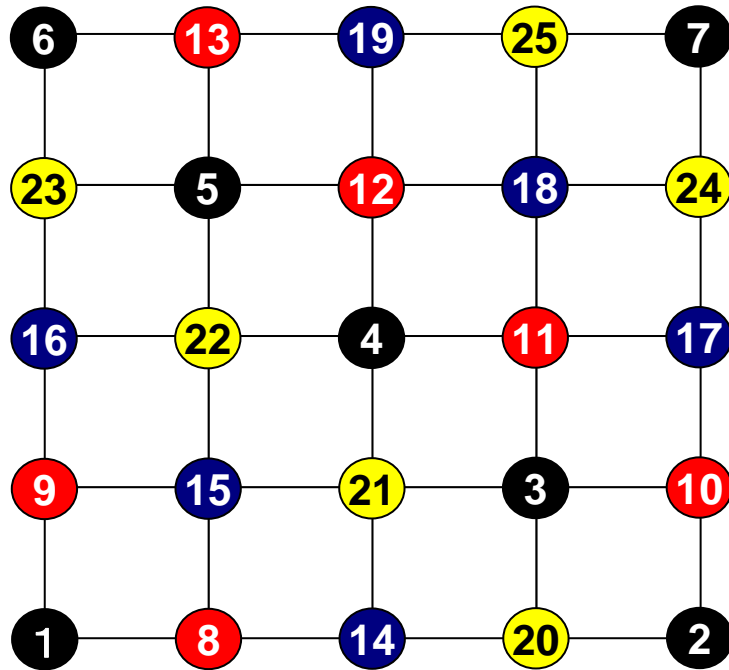
# Red-Blackの場合

並列性は高いがincompatible node多い  
ILU系前処理, Gauss-Seidelで反復回数増加



# 4色の場合

並列性は弱まるが incompatible node は減少  
 ILU系前処理, Gauss-Seidel で反復回数減少

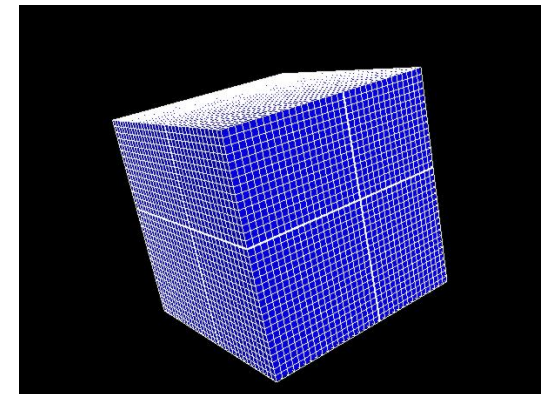
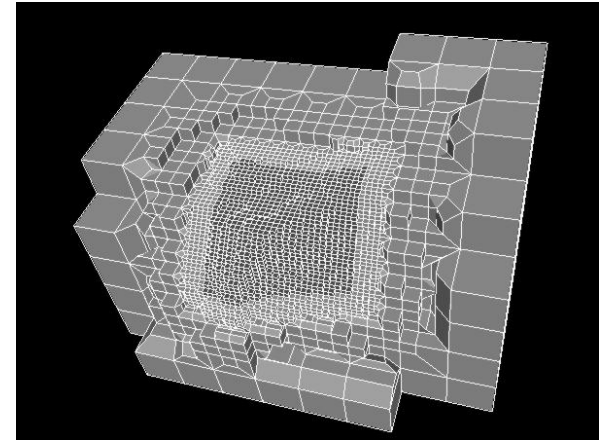




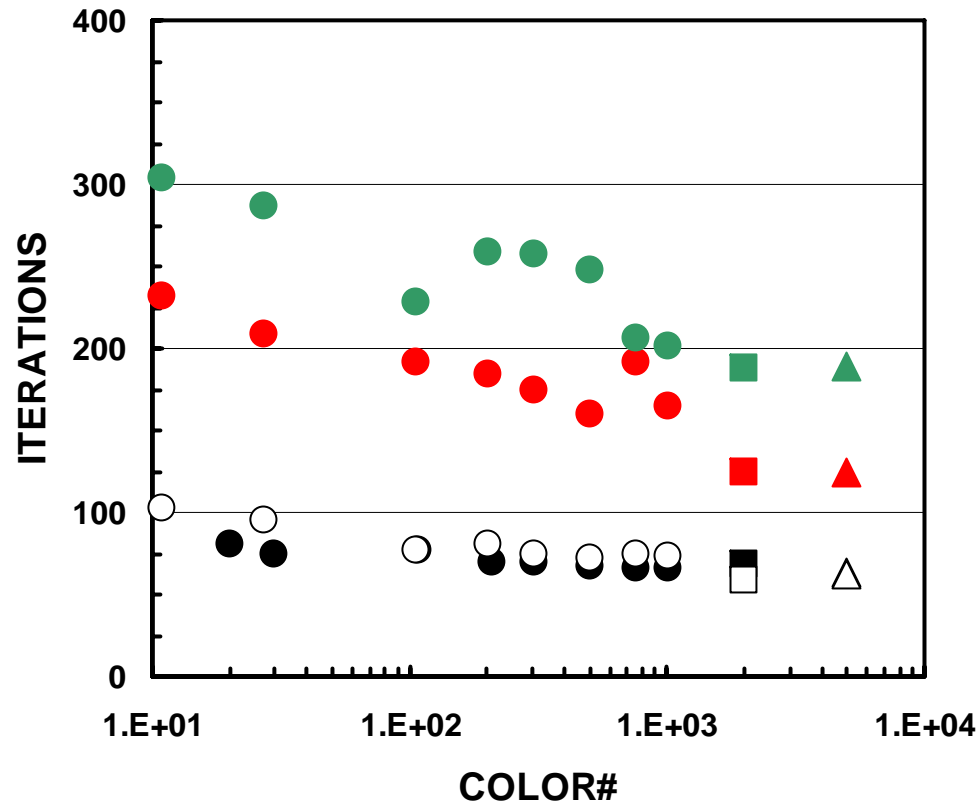
- 背景
  - 地震発生サイクルシミュレーション
  - 局所細分化, 負荷分散
- 問題設定
- **計算結果**
- まとめ
  - 今後の予定等

# 予備計算:1領域

- メッシュ1:なだらかな細分化
  - $20^3$ から初めて1要素を3段階に分割
  - 約68,600節点, 67,000要素
  - ヤング率:1.00, ポアソン比:0.24
- メッシュ2:均等
  - メッシュ数がほぼ「メッシュ1」と同じ均等な $40^3$ メッシュ
  - 約69,000節点, 64,000要素
  - ヤング率:1.00, ポアソン比:0.24, 0.47, 0.495
- メッシュ3:急激な細分化
  - メッシュ2と同じ規模だが, 要素長さを急激に変化
    - 隣接した部分で:100:3(ほぼ3段階の細分化に相当)
    - x, y, z方向に5層ずつ
  - ヤング率:1.00, ポアソン比:0.24



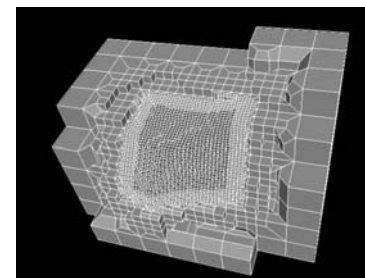
# 予備計算: 反復回数



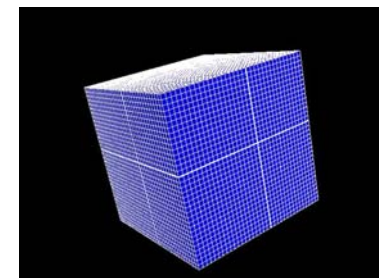
- ■ ▲ メッシュ1 (v=0.24)
- □ △ メッシュ2 (v=0.24)
- ■ ▲ メッシュ2 (v=0.47)
- ■ ▲ メッシュ3 (v=0.24)

- Multicolor オーダリング
- Initial オーダリング
- △ Cuthill-McKee

メッシュ1と2(均質)の差は  
ほとんど無い

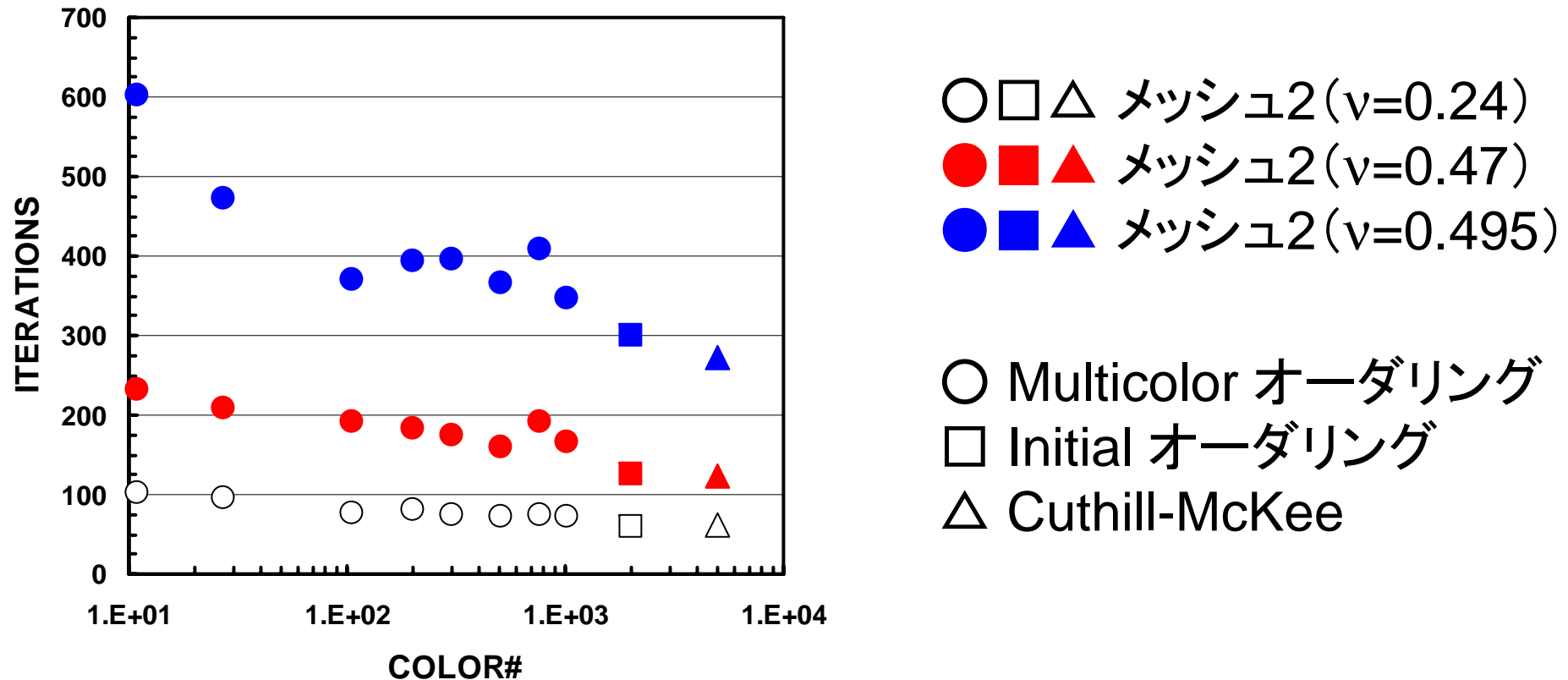


メッシュ1



メッシュ3

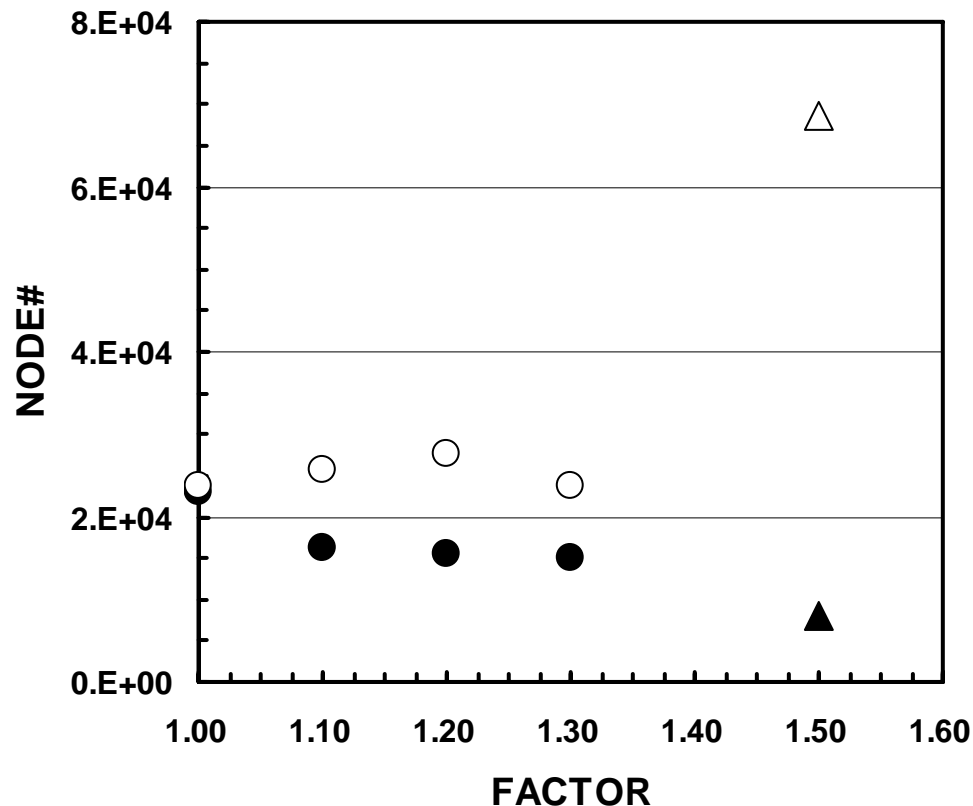
# 予備計算: 反復回数 ポアソン比の影響



ポアソン比が0.50に近づくと急激に収束悪化  
色数増加による収束の向上顕著: incompatible nodes  
ただ、解けても物理的にはあまり意味がない

# 64領域の計算: 負荷分散の効果: 節点数

ParMETISの係数の影響, アセノスフェアの $\nu=0.47$



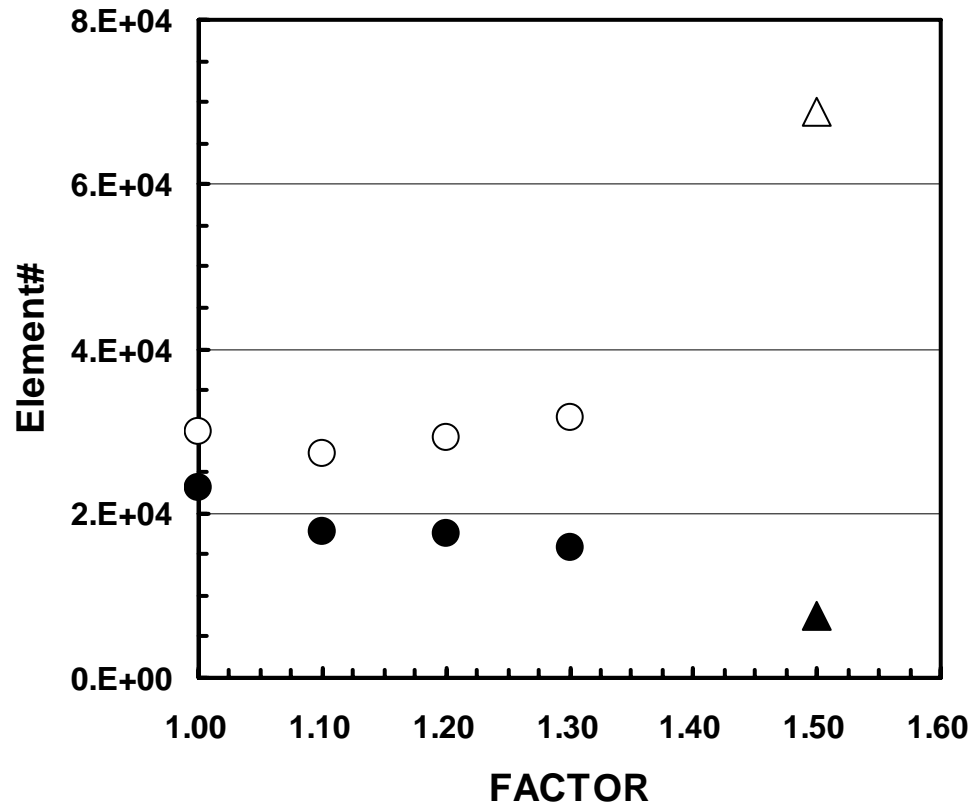
△▲ 負荷分散前の最大, 最小  
○● 負荷分散後の最大, 最小

係数が1.0に近いほど  
厳密な負荷分散を適用する

本ケースの場合, 節点数が  
バランスするように設定してある

# 64領域の計算: 負荷分散の効果: 要素数

ParMETISの係数の影響, アセノスフェアの $\nu=0.47$

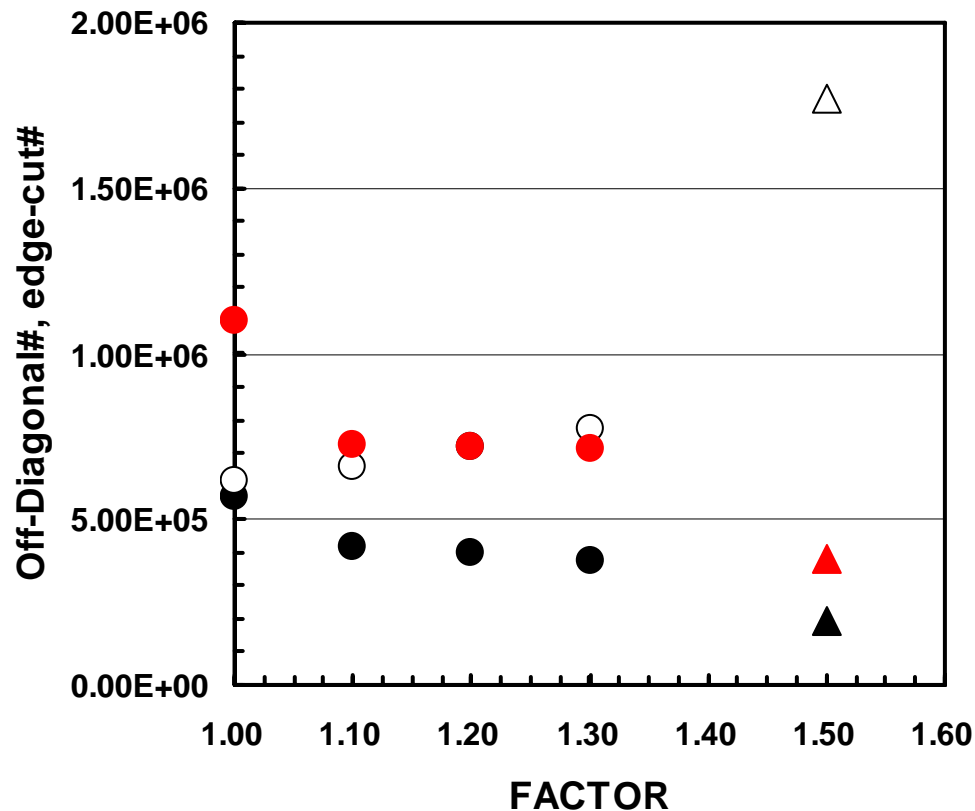


△▲ 負荷分散前の最大, 最小  
○● 負荷分散後の最大, 最小

係数が1.0に近いほど  
厳密な負荷分散を適用する

# 64領域の計算: 負荷分散の効果: 非ゼロ成分数, edge-cut

ParMETISの係数の影響, アセノスフェアの $\nu=0.47$



非ゼロ成分数

△▲ 負荷分散前の最大, 最小

○● 負荷分散後の最大, 最小

edge-cut

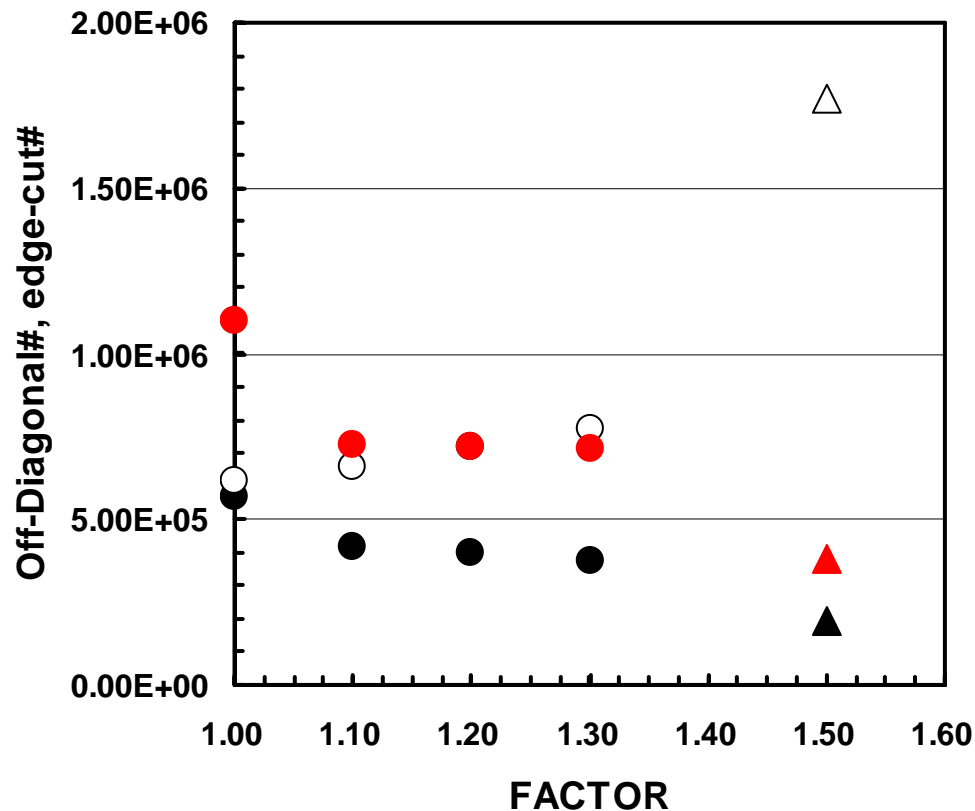
▲ 負荷分散前

● 負荷分散後

負荷バランスが悪いほど, edge-cut  
数は少ない: 並列計算の収束は良い

# 64領域の計算: 負荷分散の効果: 非ゼロ成分数, edge-cut

ParMETISの係数の影響, アセノスフェアの $\nu=0.47$



非ゼロ成分数

△▲ 負荷分散前の最大, 最小

○● 負荷分散後の最大, 最小

edge-cut

▲ 負荷分散前

● 負荷分散後

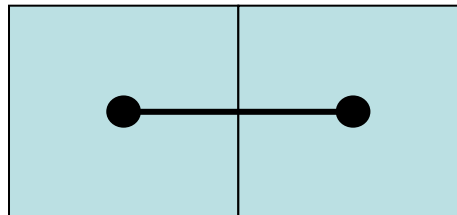
負荷バランスが悪いほど, 非ゼロ成分の最大値が増加⇒全体として計算時間がかかる



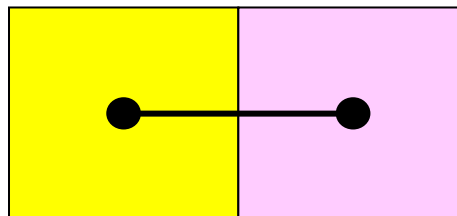
# EDGE-CUTとは？

- 辺の両端の節点（または要素）が異なった領域に属している場合、「EDGE-CUTが生じている。」という。
- EDGE-CUTが少ないほど、通信は少ない。

EDGE-CUT無し

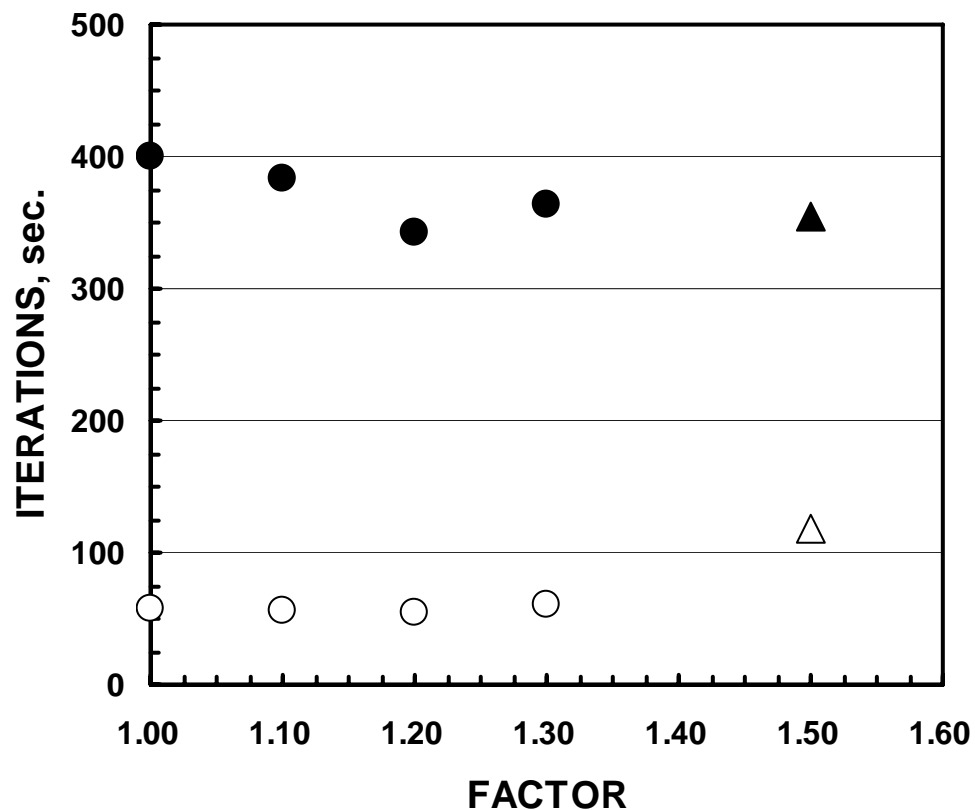


EDGE-CUT有り



# 64領域の計算: 負荷分散の効果: 反復回数, 計算時間

ParMETISの係数の影響, アセノスフェアの $\nu=0.47$



反復回数

- ▲ 負荷分散前
- 負荷分散後

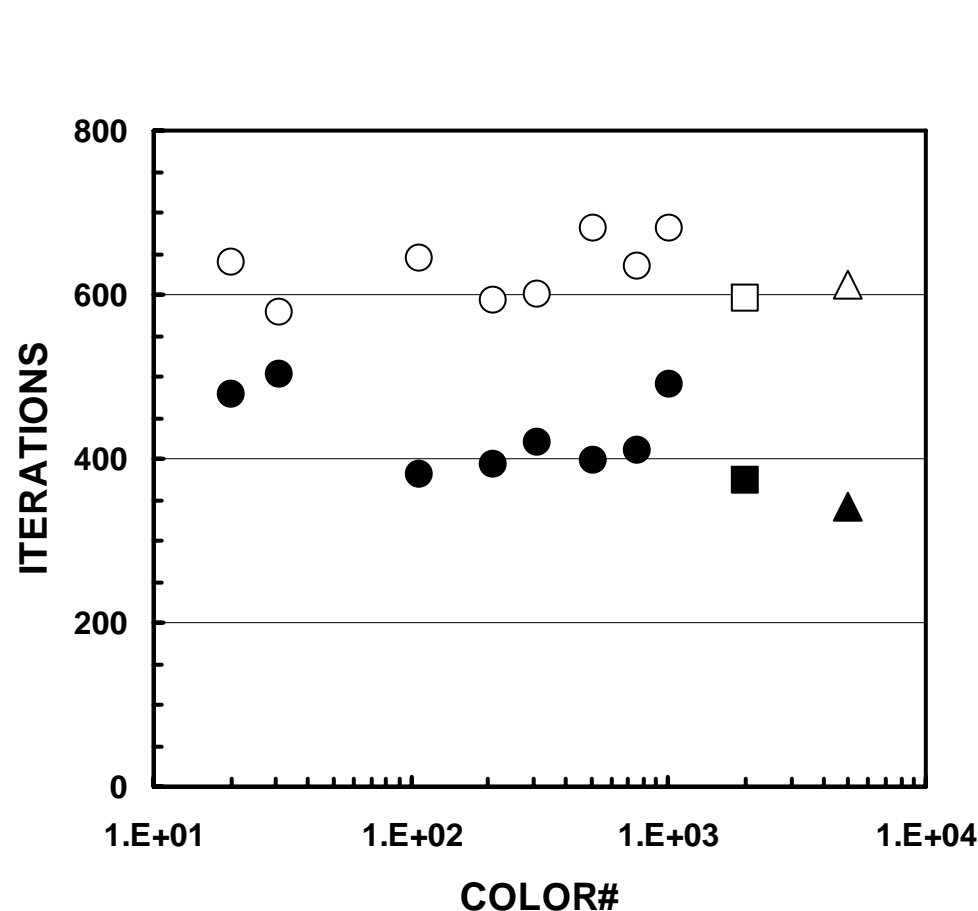
計算時間 (sec.)

- △ 負荷分散前
- 負荷分散後

edge-cut数が少ないほど並列計算の収束は良い: 計算時間は負荷分散している状態でほぼ同じ

# 64領域の計算: 負荷分散の効果: 反復回数

係数=1.20, アセノスフェアポアソン比と色数の影響



ポアソン比: 0.470

- Multicolor
- 初期オーダリング
- ▲ Cuthill-McKee

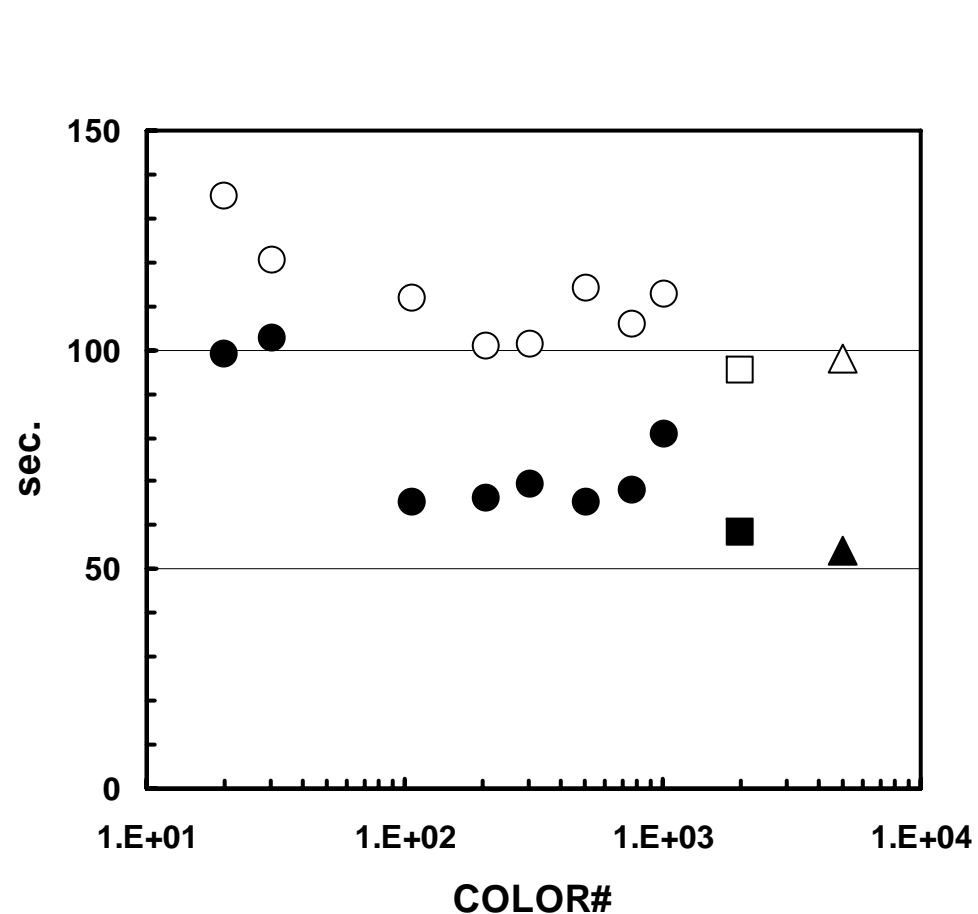
ポアソン比: 0.495

- Multicolor
- 初期オーダリング
- △ Cuthill-McKee

Multicolorにおける色数と反復回数の関係は一様ではない

# 64領域の計算: 負荷分散の効果: 計算時間

係数=1.20, アセノスフェアポアソン比と色数の影響



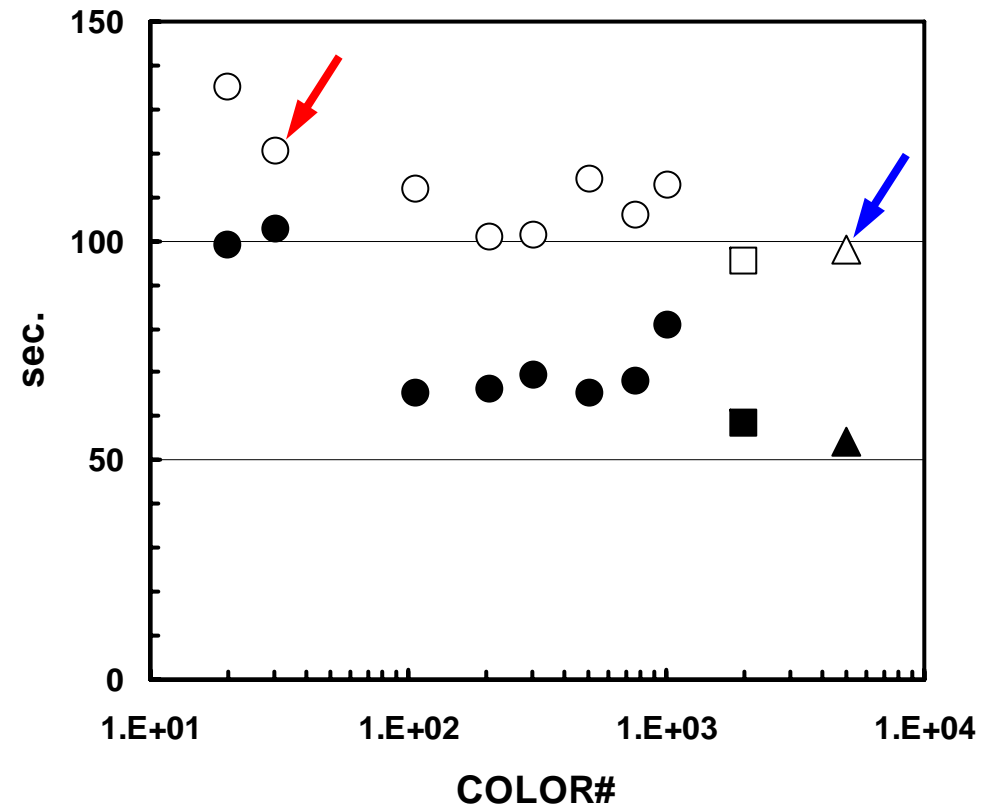
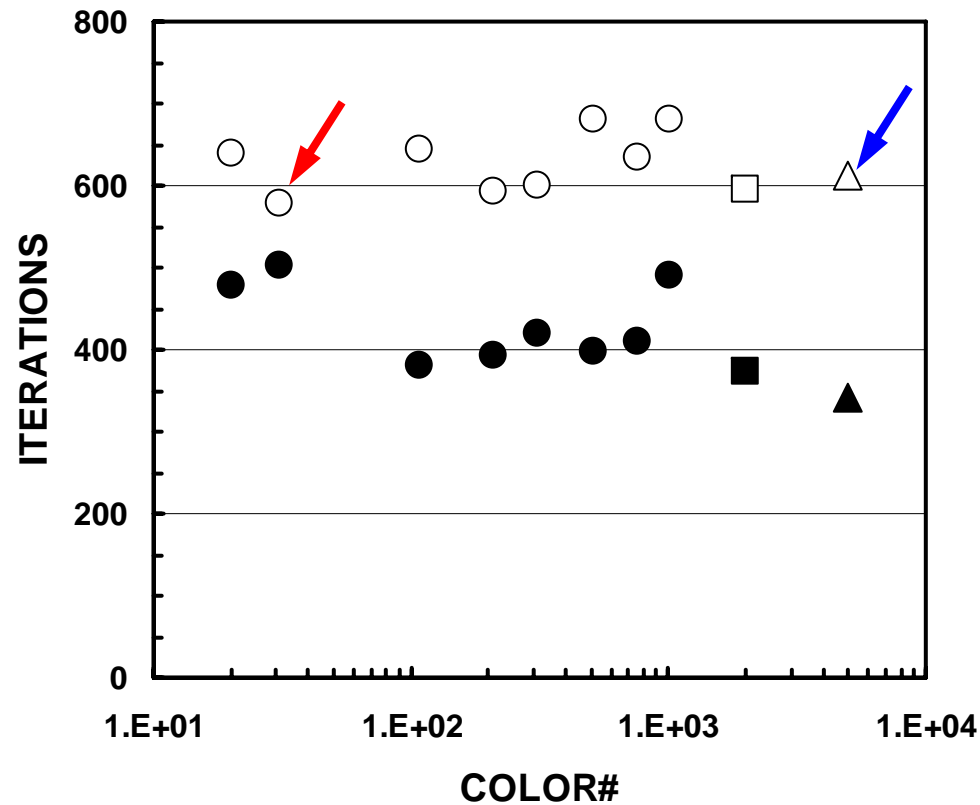
ポアソン比: 0.470

- Multicolor
- 初期オーダリング
- ▲ Cuthill-McKee

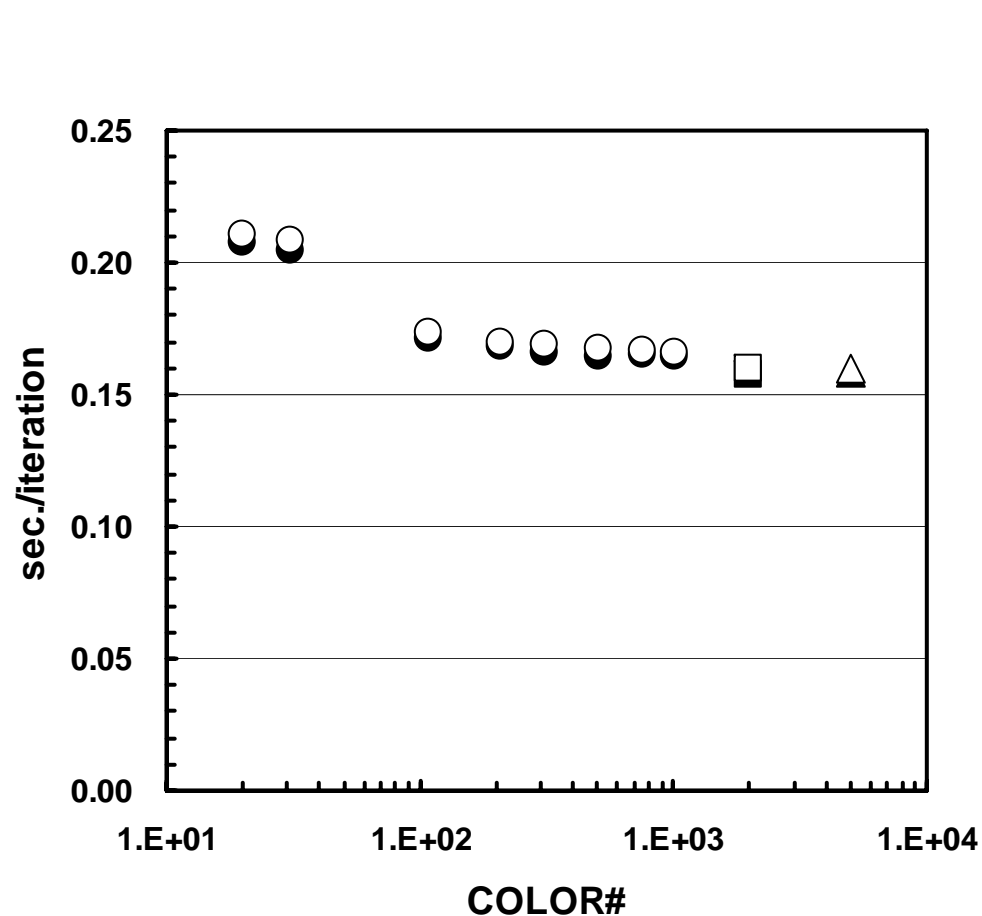
ポアソン比: 0.495

- Multicolor
- 初期オーダリング
- △ Cuthill-McKee

# 反復回数が減っているのに 計算時間が増加している？



# 1反復あたりの計算時間を測定してみると



ポアソン比: 0.470

- Multicolor
- 初期オーダリング
- ▲ Cuthill-McKee

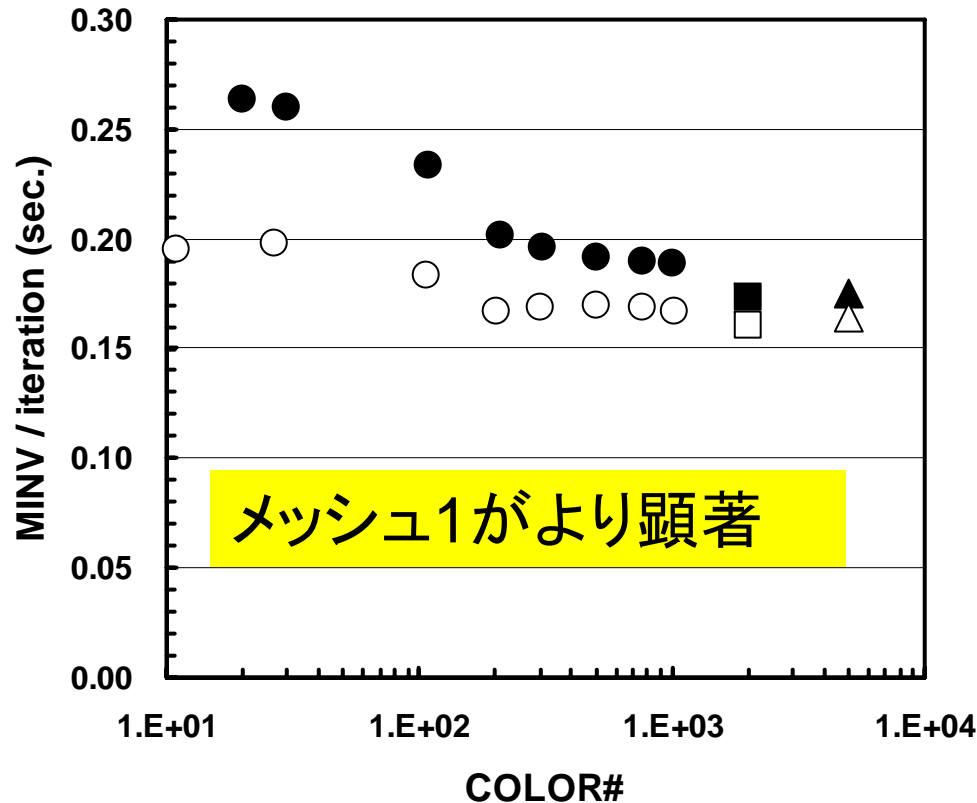
ポアソン比: 0.495

- Multicolor
- 初期オーダリング
- △ Cuthill-McKee

色数が少ないときの性能が悪い

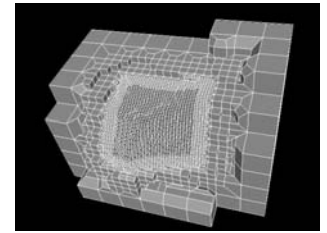
# 1領域の計算について測定

前処理部分(全体の50%以上)1反復あたり計算時間



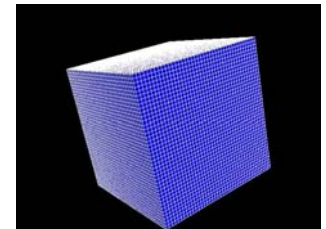
## メッシュ1: 細分化メッシュ

- Multicolor
- 初期オーダリング
- ▲ Cuthill-McKee



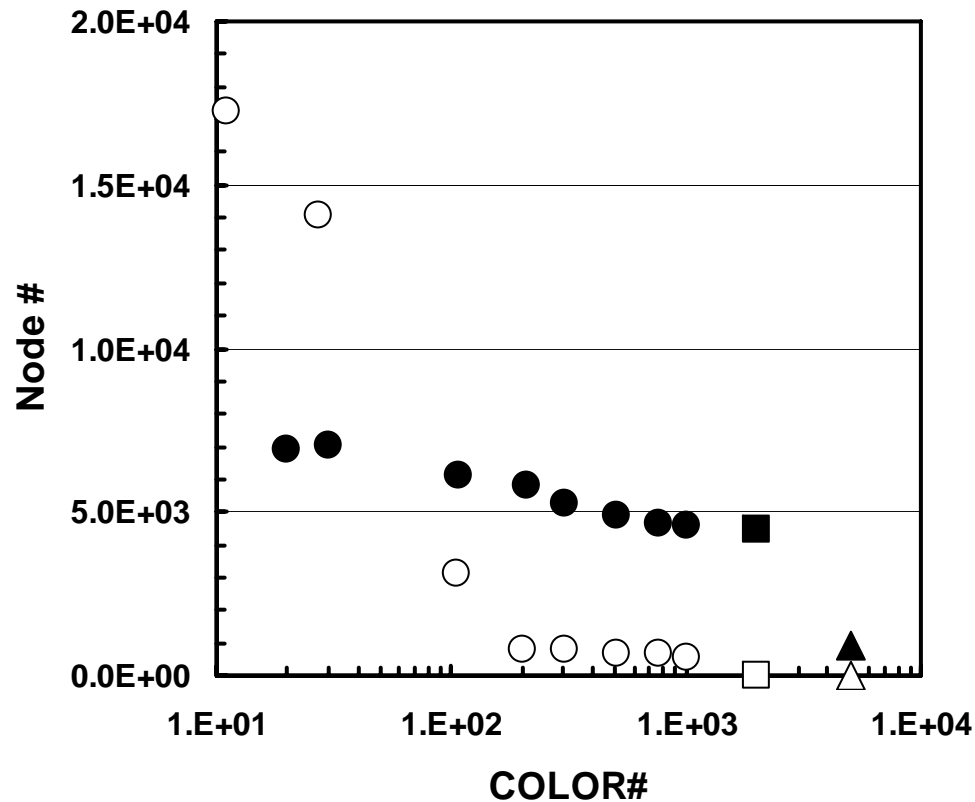
## メッシュ2: 均質メッシュ

- Multicolor
- 初期オーダリング
- △ Cuthill-McKee



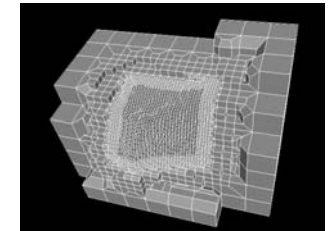
# 1領域の計算について測定

上三角成分または下三角成分が無い行の総数



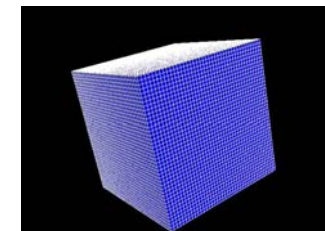
## メッシュ1: 細分化メッシュ

- Multicolor
- 初期オーダリング
- ▲ Cuthill-McKee



## メッシュ2: 均質メッシュ

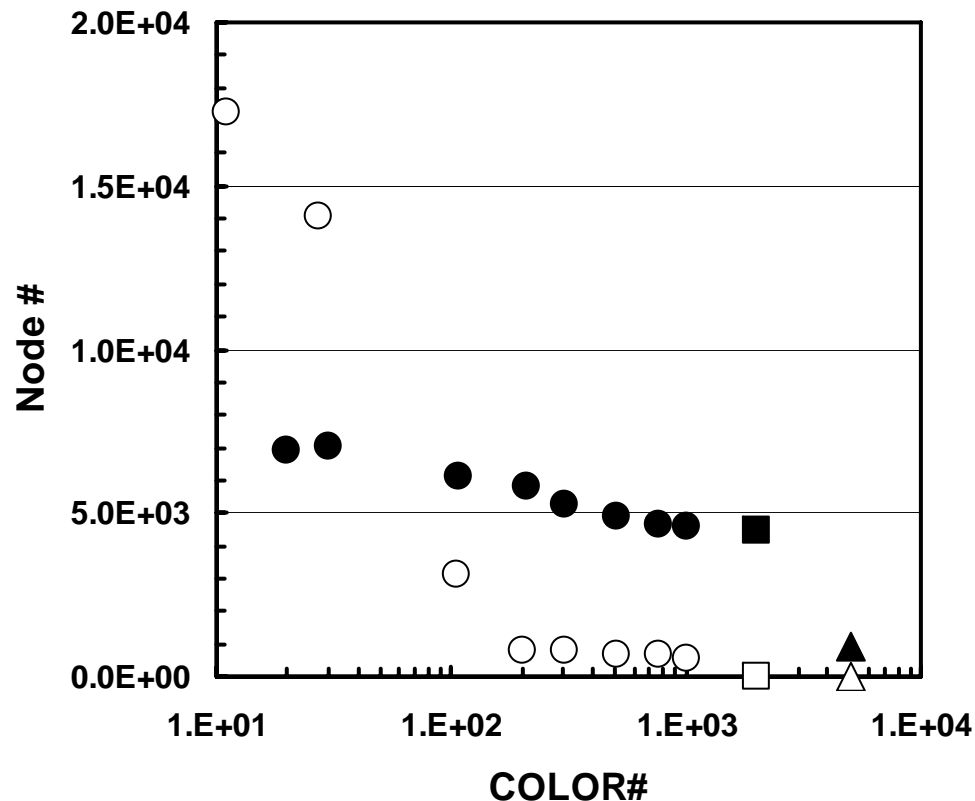
- Multicolor
- 初期オーダリング
- △ Cuthill-McKee





# 1領域の計算について測定

上三角成分または下三角成分が無い行の総数



上三角成分, または下三角成分のいずれかしか無い行では, 最も内側のループの計算量が増加しメモリに負担がかかる。メッシュ1は非対角成分の数がもともと多いため, この傾向が顕著になるとと思われる。  
⇒ESでは別の挙動？

```
do i= 1, N
  do j= 1, INL(i)
    Y(i)= Y(i) + AL(j,i)*X(IAL(j,i))
  enddo
  do j= 1, INU(i)
    Y(i)= Y(i) + AU(j,i)*X(IAU(j,i))
  enddo
enddo
```

- 背景
  - 地震発生サイクルシミュレーション
  - 局所細分化, 負荷分散
- 問題設定
- 計算結果
- **まとめ**
  - **今後の予定等**

# まとめ

- 並列FEMによる「すべり応答関数」計算
  - AMR
  - 並列27分木
  - ParMETISによる負荷分散 + データマイグレーション
- 大規模（でもないが）弾性問題
  - ポアソン比増加による収束性能悪化はあるが顕著では無い
  - 局所Block SGSで十分解ける
    - メッシュ幅変化がなだらかなため
- 負荷分散と収束性のトレードオフ
- 色数と「性能」の関係

# 今後の予定

- 並列27分木の完全実装
- 「地球シミュレータ」への移植，実計算
- ポアソン比の扱い
- 「高級な」前処理の適用
  - 不均質な場合
  - 領域間オーバーラップ